

SISTEMA DE RECOMENDACIONES DE CONTENIDOS EDUCATIVOS DE VOD CENTRADO EN LAS COMPETENCIAS EDUCATIVAS



DIEGO FABIAN DURAN DORADO

Tesis de Doctorado en Ingeniería Telemática

Director:

Jose Luis Arciniegas Herrera

Doctor en Ingeniería de Sistemas Telemáticos

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Línea de Investigación Servicios Avanzados de Telecomunicaciones

Popayán, Abril de 2019

DIEGO FABIAN DURAN DORADO

SISTEMA DE RECOMENDACIONES DE
CONTENIDOS EDUCATIVOS DE VOD CENTRADO
EN LAS COMPETENCIAS EDUCATIVAS

Tesis presentada a la Facultad de Ingeniería Electrónica y
Telecomunicaciones de la Universidad del Cauca para la obtención del Título
de:

Doctor en:
Ingeniería Telemática

Director:
Jose Luis Arciniegas Herrera
Doctor en Ingeniería de Sistemas Telemáticos

Popayán 2019



Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Programas de Posgrado

FORMATO I
ACTA DE SUSTENTACIÓN DE
TRABAJO DE GRADO DE MAESTRÍA/TESIS DOCTORAL

Los jurados de:

TRABAJO DE GRADO DE MAESTRÍA () TESIS DOCTORAL (X)

Titulado:

Sistema de Recomendaciones de Contenidos Educativos de VoD centrado en las Competencias Educativas

Bajo la dirección de: PhD Jose Luis Arciniegas Herrera

HACEN CONSTAR:

Que siendo las 9:00 am del día 15 del mes de Febrero de 2019, el(la) estudiante:
Diego Fabian Duran Dorado

realizó la Sustentación del Trabajo de Grado de Maestría/Tesis Doctoral, obteniendo la calificación de:

APROBADO (X) APROBADO CON OBSERVACIONES ()
APLAZADO () NO APROBADO ()

Para constancia, se firma en Popayán, a los 15 días del mes de Febrero de 2019.

JURADO 1:

Nombre: Pedro Javier Álvarez Pérez

JURADO 2 (Doctorado):

Nombre: Néstor Darío Duque Méndez

JURADO COORDINADOR:

Nombre: Gustavo Adolfo Ramírez
González

COORDINADOR DEL PROGRAMA (*):

Nombre:

(*) Si es requerido porque algún jurado no pueda firmar, firma el Coordinador del Programa con autorización del mismo.

Dedicado a mi esposa Eliana y mi hijo Santiago, quienes me enseñaron que el amor es el motor que mueve los sueños.

En la memoria de mi padre Julio Duran, quien me enseñó el valor de la paciencia.

Dedicado a mi madre Myriam Dorado, de quien aprendí que para obtener grandes recompensas se deben hacer grandes esfuerzos.

A mis hermanos Mónica y Julio por su amistad y cariño.

A mis sobrinos David y Sara, a quienes amo profundamente.

Resumen estructurado

El Video bajo Demanda (VoD) es en la actualidad una importante fuente de información para el apoyo de procesos educativos de enseñanza y aprendizaje. Especialmente, plataformas web como Youtube y WatchKnowLearn gozan de gran popularidad entre las personas debido a la facilidad en el acceso y reproducción de los recursos que ofrecen.

Sin embargo, las personas se deben enfrentar a dos problemas que dificultan la búsqueda efectiva de información. El primero, es la *sobrecarga de información*. De acuerdo con Levy [1], es “la exposición a, o provisión de, demasiada información; una situación problemática o estado de estrés mental aparece a raíz de esto”. El segundo, relacionado con la *metacognición* [2], consiste en que los estudiantes generalmente no son conscientes de sus estados y procesos cognitivos, así como de su habilidad para monitorear y ajustar conscientemente tales aspectos. Así, la ausencia de cierto nivel de metacognición dificulta la selección de recursos relevantes [3].

Con el propósito de proveer una alternativa de solución a estos problemas, el presente trabajo de investigación propone un Sistema de Recomendaciones (SR) centrado en competencias educativas, el cual utiliza técnicas de la web semántica para usar conocimiento aseverado e inferido dentro del proceso de obtención de recomendaciones.

El SR propuesto integra una serie de productos derivados de la investigación que se resumen a continuación:

1. Un esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas (Capítulo 3), el cual establece atributos y vínculos lógicos entre contenidos y competencias.
2. Una ontología del dominio de los contenidos de VoD y las competencias (Capítulo 4), la cual adapta ontologías existentes en la literatura de acuerdo al esquema de metadatos con el propósito de permitir la aplicación de técnicas de la web en la inferencia de conocimiento útil en el proceso de recomendación.
3. Un método computacional basado en información cualitativa (Capítulo 5) que permite evaluar competencias aplicando un modelo de computación lingüístico

sobre las relaciones de subsunción entre competencias formalizadas en los mapas de competencias.

4. Una estrategia de recomendación de contenidos de VoD enfocada en el cubrimiento de las necesidades educativas de los estudiantes (Capítulo 6). Para esto, la estrategia construye perfiles de usuario que formalizan las falencias y fortalezas en el aprendizaje, y a partir de ellos, aplica técnicas de ponderación de asociaciones semánticas (adaptadas al contexto del trabajo) sobre la ontología para inferir conocimiento propicio para mantener el carácter personalizado de las recomendaciones.
5. Una evaluación del SR enfocada en la precisión (Capítulo 7), entendiéndose ésta como una medida de la pertinencia de los contenidos de VoD para apoyar el aprendizaje de los estudiantes según el criterio de los expertos. Esto se realizó comparativamente con relación a un SR construido con base en el coeficiente Sørensen-Dice.

Entre los resultados más importantes de la investigación, se comprobó que el SR, en una configuración por defecto determinada experimentalmente, alcanza una precisión del 65% que aumenta directamente proporcional a la cantidad de asociaciones semánticas descubiertas en una base de conocimiento construida a partir de la ontología propuesta. En concreto, supera en un 38% el rendimiento del sistema basado en el coeficiente Sørensen-Dice. Esta diferencia es estadísticamente significativa con una seguridad del 99%, de acuerdo a un estudio de significancia estadística. Esto se debe a que el error estándar inherente a la experimentación (0,0008) es menor que 0,01.

Structured abstract

Nowadays, Video on Demand (VoD) is an important source of information to support teaching and learning educational processes. Particularly, web platforms such as Youtube and WatchKnowLearn are highly recognized among people due to the ease in accessing and reproducing of the content they offer.

However, people must face two issues that hinder the effective search for information. The first one, is the *information overload*. According to Levy [1], it is “The exposure to, or provision of too much information; a problematic situation or state of mental stress arising from this”. The second one is that students are usually unaware of their cognitive states and processes, as well as their ability to consciously monitor and adjust such aspects. Consequently, the lack of certain degree of meta-cognition [2] makes it difficult the selection of relevant resources [3].

In order to address these issues, the present research proposes a Recommender System (RS) focused on educational competencies, which uses semantic web techniques to explore asserted and inferred knowledge in the process of obtaining recommendations.

The proposed RS integrates a set of products derived from this research, which are summarized as follows:

1. A metadata scheme for describing VoD contents according to competencies (Chapter 3), which establishes attributes and logic links between contents and competencies.
2. An ontology of the domain of VoD contents and competencies (Chapter 4), which adapts existing literature ontologies according to the metadata scheme to allow the application of semantic web techniques in the inference of useful knowledge in the recommendation process.
3. A computational method based on qualitative information (Chapter 5) that allows to evaluate competencies by applying a linguistic computation model on subsumption relationships between competencies formalized in competency maps.
4. A recommendation strategy of VoD contents focused on students' educational needs coverage (Chapter 6). For this, such a strategy builds user profiles that formalize

weaknesses and strengths in learning, and from them, it applies semantic association ranking techniques (adapted to the context of this work) on the ontology to infer appropriate knowledge to maintain the personalized character of recommendations.

5. A RS evaluation focused on accuracy (Chapter 7), which refers to a measure of the relevance of the VoD contents to support the students' learning based on expert opinion. This is done comparatively with respect to a RS based on the Sørensen-Dice similarity coefficient.

Among the most important results of this research is that the proposed RS (in a default configuration) reaches an accuracy of 65%, which increases directly proportional to the number of discovered semantic associations. Specifically, it exceeds the performance of a system based on the coefficient Sørensen-Dice by 38%. According to a statistical significance study, this difference is statistically significant with a 99% reliability. This is because the standard error inherent in the experiment (0.0008) is less than 0.01.

Contenido

Introducción.....	1
1.1 Motivación, planteamiento del problema y solución	1
1.2 Objetivos	11
1.2.1 Objetivo general.....	11
1.2.2 Objetivos específicos.....	11
1.3 Organización de la monografía	12
Base conceptual y estado del arte	17
2.1 Base conceptual.....	17
2.1.1 Sistemas de recomendaciones.....	18
2.1.2 Video bajo Demanda	20
2.1.3 Competencias educativas.....	21
2.1.4 Mapas de competencias	22
2.1.5 Metadatos y calidad de los metadatos.....	23
2.1.6 Ontologías	24
2.1.7 Modelos computacionales lingüísticos.....	26
2.2 Estado del arte	29
2.2.1 Sistemas de recomendaciones educativos.....	31
2.2.2 Esquemas de metadatos del ámbito de las competencias	32
2.2.3 Esquemas de metadatos para contenidos multimedia	34
2.2.4 Ontologías existentes	35
2.2.5 Modelos computacionales lingüísticos para la evaluación de competencias	37
Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas	41

3.1	Perfil aplicación de TV-Anytime para la descripción de contenidos de VoD.	45
3.2	Extensión del IEEE RCD para la descripción de competencias	51
3.3	Modelo descriptivo integral ofrecido por el esquema.....	57
3.4	Evaluación de la extensión del IEEE RCD para la descripción de competencias	58
Una ontología sobre el dominio de los contenidos de VoD y las competencias		
	educativas	65
4.1	Adaptación de ontologías existentes	70
4.1.1	Adaptación de “ <i>Learning Object Metadata Ontology</i> ”	71
4.1.2	Adaptación de “ <i>Ontology for Media Resources</i> ”	80
4.1.3	Adaptación de “ <i>Competency Model Ontology</i> ”	83
4.2	OntoRCM: una ontología para el modelado de mapas de competencias.....	86
4.3	Integración de ontologías	91
4.4	Evaluación de la ontología	97
Método computacional basado en información cualitativa para la evaluación de		
	competencias	101
5.1	Estructura de los mapas de competencias.....	102
5.2	Descripción del método computacional	104
5.2.1	Fase A: Construcción del perfil del estudiante	106
5.2.2	Fase B: Cómputo de los niveles de competencia	106
5.3	Evaluación del método computacional	110
Estrategia de recomendación		
		117
6.1	Modelo del sistema de recomendaciones	120
6.2	Construcción de perfiles de usuario	125
6.2.1	El perfil inicial de usuario	126
6.2.2	Construcción de perfiles-ontología.....	126
6.2.3	Adaptación de los perfiles de usuario frente a cambios en las necesidades	
	de los estudiantes	129
6.3	Descubrimiento de asociaciones semánticas y secuencias de nodos.....	132
6.4	Filtro semántico	137

6.4.1 Ponderación y filtrado de asociaciones semánticas	138
6.5 Construcción de listas de recomendaciones	147
Evaluación del Sistema de recomendaciones	151
7.1 Prototipos	153
7.2 Colección de datos.....	155
7.3 Métricas.....	157
7.4 Fases de la evaluación.....	159
7.5 Resultados	159
7.5.1 Resultados de precisión	160
7.5.2 Resultados de tiempo de ejecución	162
7.5.3 Determinación de parámetros	163
Conclusiones y trabajos futuros	165
8.1 Conclusiones.....	165
8.2 Trabajos futuros	168
Referencias bibliográficas	173
Aspectos complementarios de la base conceptual.....	185
A.1 Características de calidad de los metadatos	185
A.2 Ejemplos de aplicación de modelos computacionales lingüísticos	186
A.3 Asociaciones semánticas complejas	188
A.3.1 Secuencias de propiedades.....	190
A.3.2 Tipos de asociaciones semánticas complejas	190
A.3.3 Ponderación de asociaciones semánticas complejas	192
Análisis de esquemas de metadatos del ámbito de las competencias.....	201
Versión extendida de la evaluación de la extensión del IEEE RCD	207
C.1 Características de calidad evaluadas.....	207
C.2 Evaluadores e instrumentos de evaluación	208
C.3 Resultados	209
C.3.1 Completitud	209
C.3.2 Accesibilidad.....	210

C.3.3 Conformidad con las expectativas	210
Versión extendida del estudio de ontologías existentes del dominio del VoD y las competencias	213
D.1 Fase 1: descubrimiento de ontologías.....	213
D.2 Fase 2: evaluación de ontologías existentes.....	214
Definición formal del perfil de aplicación de TV-Anytime.....	221
Información complementaria de la extensión del IEEE RCD.....	229
F.1 Tipos de información integrados al IEEE RCD	229
F.2 Definición formal de la extensión del IEEE RCD	233
Aspectos complementarios de la adaptación de “ <i>Learning Object Metadata Ontology</i> ”	239
G.1 Representaciones intermedias después de la adaptación	239
G.2 Cotejamiento ontología-esquema LOM.....	240
G.3 Modelo ontológico final.....	241
Versión extendida de la adaptación de “ <i>Ontology for Media Resources</i> ”	243
Versión extendida de la adaptación de “ <i>Competency Model Ontology</i> ”	253
Versión extendida de la construcción de OntoRCM	265
Versión extendida de la evaluación de la ontología propuesta.....	279
K.1 Declaración de URI.....	280
K.1.1 Método: Chequear declaraciones de nombres	280
K.2 Literales y tipos de datos	282
K.2.1 Método: Chequeo de literales y tipos de datos	282
K.3 Validación XML.....	283
K.3.1 Método: Validar con respecto a XML	283
K.4 Profundidad máxima de la jerarquía de conceptos.....	284
K.4.1 Método: Claridad de la jerarquía de subsunción	284
K.5 Preguntas de competencia formalizadas.....	285
K.5.1 Método: Chequear preguntas de competencia con relación a resultados	286
Versión extendida de la evaluación del método computacional propuesto	291

L.1 Prototipo del método computacional.....	292
L.2 Colección de datos	294
L.3 Métricas	295
L.4 Fases de la evaluación	296
L.5 Resultados.....	297
L.5.1 Resultados de precisión	297
L.5.2 Resultados de tiempo de ejecución y consumo de memoria	298
Pseudoalgoritmos propuestos	301
M.1 DFS-CA: “ <i>Depth First Search-Competency Assessment</i> ”	301
M.2 NSD: “ <i>Node Sequence Discovery</i> ”	302
M.3 RCS: “ <i>Relevant Competencies Search</i> ”	304
M.4 VTR: “ <i>Video Tracks Ranking</i> ”	304
M.5 VR: “ <i>Video Ranking</i> ”	306
Construcción de listas de recomendaciones para la estrategia de recomendación	309
N.1 Identificación de competencias relevantes.....	310
N.2 Consulta y ponderación de segmentos de video.....	310
N.3 Consulta y ponderación de contenidos de VoD	312
N.4 Ordenamiento de contenidos de VoD	313
Prototipos	315
O.1 CMA-Tool: prototipo del método computacional para la evaluación de competencias.....	315
O.1.1 Generador de interfaces.....	316
O.1.2 Mapa base.....	317
O.1.3 Procesador de mapas	318
O.1.4 Vista dinámica del prototipo	319
O.2 Prototipos de sistemas de recomendaciones.....	320
O.2.1 EduCompRS: prototipo del sistema de recomendaciones semántico	320
O.2.2 Prototipo del sistema de recomendaciones sintáctico	328
Resultados complementarios de tiempos de ejecución del prototipo del sistema de recomendaciones propuesto	331

Discusión	337
Q.1 Uso esperado de lo propuesto	337
Q.2 Resultados de precisión	339
Q.3 Limitaciones	339
Resumen de productos derivados de la investigación.....	341
Versión previa de la extensión del IEEE RCD.....	343

Lista de figuras

Figura 1.1 Representación del escenario problemático	3
Figura 1.2 Solución planteada del Sistema de Recomendaciones.....	10
Figura 2.1. Fragmento de un mapa de competencias basado en el modelo de SRCM. Fuente: propia.	23
Figura 2.2 Un grupo de siete términos lingüísticos y su semántica. Fuente: [57].....	27
Figura 2.3 Esquemas de metadatos relacionados con competencias. Fuente: propia.	34
Figura 3.1. Proceso desarrollado para la generación del esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias.....	44
Figura 3.2 Ejemplo del modelo descriptivo de los contenidos desde el punto de vista del servicio de VoD. Fuente: propia.	48
Figura 3.3 Estructura del perfil de aplicación de TV-Anytime enfocado en el VoD....	50
Figura 3.4 Modelo descriptivo de las competencias educativas. Fuente: propia.....	54
Figura 3.5 Estructura de la extensión del IEEE RCD	57
Figura 3.6 Modelo descriptivo integral. Fuente: propia.....	58
Figura 3.7 Resultados para las preguntas guía: (a) P2, basados en el porcentaje de evaluadores que consideran que los elementos son innecesarios; y (b) P3, basados en el porcentaje de evaluadores que consideran que los elementos son confusos. Fuente: propia.	61
Figura 3.8 Resultados para las preguntas guía: (a) P4, basados en el porcentaje de evaluadores que están de acuerdo con que el esquema cumple sus expectativas; (b) P5, de acuerdo con que los elementos cumplen sus funciones; (c) P6, basados en el porcentaje de evaluadores que no están de acuerdo con que los valores deben ser asignados por medio de vocabularios; y (d) P7, de acuerdo con el porcentaje de evaluadores que no están de acuerdo con los vocabularios propuestos. Fuente: propia.	62
Figura 4.1 Proceso desarrollado para el diseño de la ontología del dominio de los contenidos de VoD y las competencias.....	67
Figura 4.2 Red semántica identificada de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	73
Figura 4.3 Red semántica de la adaptación de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	77

Figura 4.4 Ejemplo de aplicación de la adaptación de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.....	80
Figura 4.5 Red semántica para la adaptación de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	81
Figura 4.6 Ejemplo de inferencia entre los contenidos de VoD y los segmentos de video. Fuente: propia.....	82
Figura 4.7 Ejemplo de aplicación de “ <i>Ontology for Media resources</i> ”. Fuente: propia.	83
Figura 4.8 Red semántica de la adaptación de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	85
Figura 4.9 Ejemplo de aplicación de la adaptación de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	85
Figura 4.10 Red semántica para OntoRCM. Fuente: propia.....	87
Figura 4.11 Modelo para inferir competencias generales y específicas. Fuente: propia.	89
Figura 4.12 Ejemplo de un mapas de competencias.....	90
Figura 4.13 Ejemplo de aplicación de la ontología para el modelado de mapas de competencias. Fuente: propia.	91
Figura 4.14 Resultado gráfico del <i>matching</i> de ontologías por medio de COMA. Fuente: propia.	92
Figura 4.15 Resultado explícito del “ <i>matching</i> ” de ontologías por medio de COMA. Fuente: propia.	93
Figura 4.16 Vista 1 del modelo de integración de ontologías. Fuente: propia.	96
Figura 4.17 Vista 2 del modelo de integración de ontologías. Fuente: propia.	96
Figura 4.18 Extracto de la base de conocimiento. Fuente: propia.....	97
Figura 5.1 Modelo de los mapas de competencias aplicado a través de un ejemplo. Se añadieron las actividades de evaluación al modelo de SRCM. Fuente: propia.	103
Figura 5.2 Método computacional propuesto. Fuente: propia.....	105
Figura 5.3 Modelo de los mapas de acuerdo con: (a) el modelo ontológico propuesto; (b) el modelo de un grafo etiquetado. Fuente: propia.....	105
Figura 5.4 Asignación de índices para términos lingüísticos. Fuente: propia.	108
Figura 5.5 Interfaz principal de la herramienta CMA-Tool. Fuente: propia.	111
Figura 5.6 Interfaces que presentan los resultados de la evaluación de competencias. Fuente: propia.	112
Figura 5.7 Formato usado para la descripción de la estructura de los mapas. Fuente: propia.	113
Figura 5.8 Formato usado para la descripción de los mapas de competencias. Fuente: propia.	113

Figura 5.9 Tiempos de ejecución del modelo propuesto y el MBPE para cada una de las estructuras de los mapas. Fuente: propia.....	116
Figura 5.10 Consumo de memoria del modelo propuesto y el MBPE para cada una de las estructuras de los mapas. Fuente: propia.....	116
Figura 6.1 Flujo de trabajo del sistema de recomendación propuesto	119
Figura 6.2 Modelo del sistema de recomendaciones propuesto. Fuente: propia. ...	121
Figura 6.3 Diagrama de secuencias que explica la inter-relación de los componentes durante la ejecución de la estrategia de recomendación. Fuente: propia.	124
Figura 6.4 Modelo para la construcción de perfiles. Fuente: propia.	128
Figura 6.5 Ejemplo de un perfil-ontología. Fuente: propia.....	131
Figura 6.6 Ejemplo de ponderación de segmentos de video y contenidos de VoD. Fuente: propia.	149
Figura 7.1 Interfaz del prototipo del sistema de recomendaciones propuesto. Fuente: propia.	154
Figura 7.2 Formato usado para la descripción de competencias. Fuente: propia. ..	156
Figura 7.3 Formato usado para la descripción de contenidos de VoD y segmentos. Fuente: propia.	156
Figura 7.4 Resultados de precisión: sistema de recomendaciones propuesto vs. sistema de recomendaciones sintáctico. Fuente: propia.....	160
Figura 7.5 Resultados de tiempo de ejecución para: (a) obtener la lista de recomendaciones para una competencia; (b) ejecutar procesos propios del sistema propuesto. Fuente: propia.	163
Figura A.1 Extracto de una base de conocimiento de un portal cultural. Fuente: [4].	189
Figura B.1 Esquemas de metadatos relacionados con competencias. Fuente: propia.	201
Figura C.1 Resultados para las preguntas guía: (a) P2, basados en el porcentaje de evaluadores que consideran que los elementos son innecesarios; y (b) P3, basados en el porcentaje de evaluadores que consideran que los elementos son confusos. Fuente: propia.	210
Figura C.2 Resultados para las preguntas guía: (a) P4, basados en el porcentaje de evaluadores que están de acuerdo con que el esquema cumple sus expectativas; (b) P5, de acuerdo con que los elementos cumplen sus funciones; (c) P6, basados en el porcentaje de evaluadores que no están de acuerdo con que los valores deben ser asignados por medio de vocabularios; y (d) P7, de acuerdo con el porcentaje de evaluadores que no están de acuerdo con los vocabularios propuestos. Fuente: propia.	211
Figura H.1 Red semántica de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	244

Figura H.2 Red semántica para la adaptación de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	249
Figura H.3 Ejemplo de inferencia entre los contenidos de VoD y los segmentos de video. Fuente: propia.	252
Figura I.1 Red semántica de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	254
Figura I.2 Red semántica de la adaptación de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	259
Figura J.1 Red semántica para OntoRCM. Fuente: propia.	268
Figura J.2 Modelo para inferir competencias generales y específicas. Fuente: propia.	276
Figura J.3 Modelo para inferir competencias de las cuales depende una en particular. Fuente: propia.	276
Figura J.4 Modelo para inferir competencias posibilitadas por una específica. Fuente: propia.	278
Figura K.1 Jerarquía de la ontología integrada. Fuente: propia.	285
Figura L.1 Interfaz principal de la herramienta para la evaluación de competencias. Fuente: propia.	293
Figura L.2 Interfaces que presentan los resultados de la evaluación de competencias. Fuente: propia.	294
Figura L.3 Formato usado para la descripción de la estructura de los mapas. Fuente: propia.	295
Figura L.4 Formato usado para la descripción de los mapas de competencias. Fuente: propia.	295
Figura L.5 Tiempos de ejecución del modelo propuesto y el MBPE para cada una de las estructuras de los mapas. Fuente: propia.	299
Figura L.6 Consumo de memoria del modelo propuesto y el MBPE para cada una de las estructuras de los mapas. Fuente: propia.	299
Figura O.1 Vista funcional del prototipo del método computacional para la evaluación de competencias	315
Figura O.2 Interfaz principal del prototipo.....	316
Figura O.3 Interfaz para la introducción de los Niveles de Competencia	317
Figura O.4 Interfces que presentan los resultados de la evaluación de los mapas de competencia, tiempos de ejecución y consumo de memoria.....	317
Figura O.5 Ejemplo sobre la representación de la estructura de un mapa usando el Formato de Grafos JSON.....	319
Figura O.6 Vista dinámica de CMA-Tool	320
Figura O.7 Vista funcional de los prototipos EduCompRS y SR sintáctico.....	321
Figura O.8 Interfaz del prototipo del sistema de recomendaciones semántico.....	325

Figura O.9 Vista dinámica de la aplicación de usuario del sistema de recomendaciones semántico.....	327
Figura O.10 Vista dinámica del prototipo del sistema de recomendaciones sintáctico	330
Figura P.1 Resultados de tiempo de ejecución para la construcción de perfiles-ontología	332
Figura P.2 Resultados de tiempo de ejecución para el descubrimiento de asociaciones semánticas complejas	333
Figura P.3 Resultados de tiempo de ejecución para el cálculo de la popularidad personalizada	333
Figura P.4 Resultados de tiempo de ejecución para el cálculo de la similitud semántica	334
Figura P.5 Resultados de tiempo de ejecución para el cálculo de la longitud de la asociación	334
Figura P.6 Resultados de tiempo de ejecución para la ponderación del peso de la subsunción en los mapas	335
Figura Q.1 Comparación entre categorías de Youtube con base en: (a) el número de suscriptores; (b) el número de vistas. Fuente: propia.....	339
Figura S.1 Versión evaluada de la extensión del IEEE RCD	343

Lista de tablas

Tabla 2.1. Clasificación de trabajos sobre sistemas de recomendaciones educativos. Fuente: propia.	32
Tabla 2.2. Comparación entre <i>TV-Anytime</i> y <i>MPEG-7</i> . Fuente: propia.	36
Tabla 3.1. Relación entre la información de los contenidos que se busca describir y los elementos de datos de <i>TV-Anytime</i> . Fuente: propia.	50
Tabla 3.2. Relación entre la información de las competencias que se busca describir y los elementos de datos correspondientes. Fuente: propia.	55
Tabla 3.3. Vocabularios para los elementos del esquema de metadatos para competencias. Fuente: propia.	56
Tabla 3.4 Características de calidad, interpretación y preguntas guía adaptadas a los esquemas de metadatos. Fuente: propia.	60
Tabla 4.1. Diccionario de conceptos de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	72
Tabla 4.2. Relaciones binarias de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	72
Tabla 4.3. Tabla de atributos de las instancias de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	72
Tabla 4.4. Fragmento de la relación entre LOM y “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	74
Tabla 4.5. Diccionario de conceptos para la adaptación de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	76
Tabla 4.6. Relaciones binarias para la adaptación de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	76
Tabla 4.7. Tabla de atributos de instancias para la adaptación de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	77
Tabla 4.8. Fragmento de la tabla de instancias para la adaptación de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	78
Tabla 4.9. Fragmento del modelo ontológico final para la adaptación de “ <i>Learning Object Metadata Ontology</i> ”. Fuente: propia.	79
Tabla 5.1 Estadística de la colección de datos usada para la evaluación del método computacional para la evaluación de competencias. Fuente: propia.	112

Tabla 5.2 Precisión del método para la evaluación de competencias propuesto y del MBPE para cada estructura de los mapas de competencias. Fuente: propia.	114
Tabla 6.1 Definición de los valores del factor de olvido. Fuente: propia.	130
Tabla 6.2 Ejemplo del cálculo de grados de necesidad por competencias. Fuente: propia.	131
Tabla 6.3 Ejemplo del cálculo de grados de necesidad por propiedades semánticas. Fuente: propia.	131
Tabla 6.4 Ejemplo de descubrimiento de asociaciones semánticas complejas. Fuente: propia.	137
Tabla 6.5 Ejemplo sobre la ponderación de asociaciones semánticas complejas. Fuente: propia.	147
Tabla 7.1 Estadística de la colección de datos para la evaluación del sistema de recomendaciones propuesto. Fuente: propia.	157
Tabla A.1. Resultados de la evaluación de expertos para el ejemplo. Fuente: [60].	186
Tabla A.2. Resultados de la evaluación de expertos convertidos a <i>2-tuplas</i> para el ejemplo. Fuente: [60].	186
Tabla B.1. Resultados del análisis comparativo sobre la información descrita por los esquemas de metadatos. Fuente: propia.	205
Tabla B.2. Fortalezas y debilidades de los esquemas de metadatos. Fuente: propia.	206
Tabla C.1 Características de calidad, interpretación y preguntas guía adaptadas a los esquemas de metadatos. Fuente: propia.	208
Tabla D.1. Ontologías pre-seleccionadas para su reutilización. Fuente: propia.	215
Tabla D.2. Resultado de evaluación de las ontologías pre-seleccionadas. Fuente: propia.	219
Tabla E.1. Definición de elementos y tipos de datos para la descripción de contenidos de VoD de acuerdo a <i>TV-Anytime</i> . Fuente: propia.	227
Tabla F.1. Definición de elementos y tipos de datos para la descripción de competencias. Fuente: propia.	237
Tabla G.1. Tabla de instancias para la adaptación de " <i>Learning Object Metadata Ontology</i> ". Fuente: propia.	240
Tabla G.2. Relación entre LOM y " <i>Learning Object Metadata Ontology</i> ". Fuente: propia.	241
Tabla G.3. Modelo ontológico final para la adaptación de " <i>Learning Object Metadata Ontology</i> ". Fuente: propia.	242
Tabla H.1. Diccionario de conceptos de " <i>Ontology for Media Resources</i> ". Fuente: propia.	243

Tabla H.2. Relaciones binarias de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	244
Tabla H.3. Tabla de atributos de instancias de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	244
Tabla H.4. Relación entre elementos <i>TV-Anytime</i> del esquema y “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	247
Tabla H.5. Diccionario de conceptos para la adaptación de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	248
Tabla H.6. Relaciones binarias para la adaptación de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	249
Tabla H.7. Tabla de atributos de instancias para la adaptación de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	250
Tabla H.8. Tabla de instancias de la adaptación de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	250
Tabla H.9. Modelo ontológico final para la adaptación de “ <i>Ontology for Media Resources</i> ”. Fuente: propia.	252
Tabla I.1. Diccionario de conceptos de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	253
Tabla I.2. Relaciones binarias de “ <i>Competency Model Ontology</i> ”. Fuente: propia. .	253
Tabla I.3. Diccionario de conceptos de la adaptación de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	258
Tabla I.4. Relaciones binarias para la adaptación de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	259
Tabla I.5. Tabla de atributos de instancias de la adaptación de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	260
Tabla I.6. Instancias definidas en la adaptación de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	261
Tabla I.7. Modelo ontológico para la adaptación de “ <i>Competency Model Ontology</i> ”. Fuente: propia.	263
Tabla J.1. Diccionario de conceptos para <i>OntoRCM</i> . Fuente: propia.	267
Tabla J.2. Relaciones binarias para <i>OntoRCM</i> . Fuente: propia.	267
Tabla J.3. Tabla de atributos para <i>OntoRCM</i> . Fuente: propia.	268
Tabla J.4. Tabla de instancias para <i>OntoRCM</i> . Fuente: propia.	269
Tabla J.5. Clases de la ontología para <i>OntoRCM</i> . Fuente: propia.	271
Tabla J.6. Tabla de atributos de instancias para <i>OntoRCM</i> . Fuente: propia.	272
Tabla J.7. Modelo ontológico de <i>OntoRCM</i> . Fuente: propia.	273
Tabla K.1 Clases e instancias con el mismo nombre entre ontologías. Fuente: propia.	281

Tabla L.1 Estadística de la colección de datos usada para la evaluación del método computacional para la evaluación de competencias. Fuente: propia.	295
Tabla L.2 Precisión del método para la evaluación de competencias propuesto y del MBPE para cada estructura de los mapas de competencias. Fuente: propia.	298
Tabla S.1 Vocabulario para el elemento <code>typical_age</code>	343
Tabla S.2 Definiciones de la versión previa del IEEE RCD que fueron modificadas en la versión propuesta	344

Lista de algoritmos

Algoritmo M.1. Depth First Search-Competency Assessment. Fuente: propia.....	301
Algoritmo M.2. Node Sequence Discovery. Fuente: propia.....	302
Algoritmo M.3. Relevant Competencies Search. Fuente: propia.....	304
Algoritmo M.4. Video Tracks Ranking. Fuente: propia.....	305
Algoritmo M.5. Video Ranking. Fuente: propia.....	306

Lista de siglas

AE: Actividad de Evaluación

AVDP: “*Audio Visual Description Profile*”

BFS: “*Breadth First Search*”, Búsqueda Primero en anchura

ADM: “*Audio Model*”, Modelo de Audio

CB: “*Content-Based*”, Basado en Contenido

CDP: “*Core Description Profile*”, Perfil de Descripción Base

CF: “*Collaborative Filtering*”, Filtrado Colaborativo

DFS: “*Depth First Search*”, Búsqueda Primero en Profundidad

DFS-CA: “*Depth First Search-Competency Assessment*”, Búsqueda Primero en Profundidad-Evaluación de competencias

CINE: Clasificación Internacional Normalizada de la Educación

CM: Calidad de los Metadatos

CRID: “*Content Reference Identifier*”, Identificador de Referencia de Contenido

DAE: Datos Abiertos Enlazados

EPG: “*Electronic Program Guide*”, Guía Electrónica de Programación

EQF: “*European Qualifications Framework*”, Marco Europeo de Cualificaciones

GAD: Grafo Acíclico Dirigido

GN: Grado de Necesidad

IA: Inteligencia Artificial

JSON: “*JavaScript Object Notation*”, Notación de Objetos JavaScript

RCD: “*Reusable Competency Definition*”, Definición Reusable de Competencia

KMRG: “*Knowledge Management Research Group*”, Grupo de Investigación de Gestión de Conocimiento

LOCO: “*Learning Object Context Ontologies*”, Ontologías de Contexto para Objetos de Aprendizaje

LOD: “*Learning Outcome Definition*”, Definición de Resultado de Aprendizaje

LOM: “*Learning Object Metadata*”, Metadatos para Objetos de Aprendizaje

LOV: “*Linked Open Vocabularies*”, Vocabularios Abiertos Enlazados

MBPE: Modelo Basado en el Principio de Extensión

MCFS: “*MedBiquitous Competency Framework Specification*”, Especificación del Marco de Competencia MedBiquitous

MCOS: “*MedBiquitous Competency Object Specification*”, Especificación de Objeto de Competencia de MedBiquitous.

MRM: “*Music Reporting Metadata*”, Metadatos de Informes de Música

MVC: Modelo Vista Controlador

NC: Nivel de Competencia

NSD: “*Node Sequence Discovery*”, Descubrimiento de Secuencias de Nodos

OA: Objetos de Aprendizaje

OCDE: Organización para la Cooperación y el Desarrollo Económico

RCS: “*Relevant Competencies Search*”, Búsqueda de Competencias Relevantes

RDF: “*Resource Description Framework*”, Marco de Descripción de Recursos

RDCEO: “*Reusable Definition of Competency or Educational Objective*”, Definición Reusable de Competencia u Objetivo Educativo.

RI: Representación Intermedia

SemDis: “*SEMantic DIScovery*”

SKOS: “*Simple Knowledge Organization System*”, Sistema de Organización del Conocimiento Simple

SLOR: “*Semantic Learning Objects Repository*”, Repositorio Semántico de Objetos de Aprendizaje

SMP: “*Simple Metadata Profile*”, Perfil de Metadatos Simple

SR: Sistema de Recomendaciones

SRCM: “*Simple Reusable Competency Map*”, Mapa de Competencia Reusable Simple

VAL: Valor de Agregación Lingüístico

VAN: Valor de Agregación Numérico

VoD: Video bajo Demanda

VR: “*Video Ranking*”, Ranking de Video

VTR: “*Video Tracks Ranking*”, Ranking de segmentos de video

W3C: “*World Wide Web Consortium*”, Consorcio Mundial de la Web

Capítulo 1

Introducción

El presente capítulo introduce el trabajo de investigación aquí descrito. Se mencionan aspectos motivantes, tales como la necesidad de asistir a docentes y estudiantes en la selección de contenidos de VoD relevantes para apoyar la enseñanza y aprendizaje de competencias educativas. Se describe el planteamiento del problema, el cual se relaciona con las dificultades a las que se enfrentan docentes y estudiantes durante la búsqueda efectiva de contenidos en las plataformas de VoD actuales. Se realiza una descripción de la solución propuesta, la cual es un Sistema de Recomendaciones (SR) semántico que explota tanto las capacidades de inferencia de las técnicas de la web semántica para descubrir conocimiento a partir del análisis de una ontología; como las técnicas del enfoque difuso, usado para la evaluación cualitativa de competencias, siendo esta una tarea importante en la construcción de perfiles de usuario. Además, se definen los objetivos trazados para el desarrollo del trabajo de investigación. Finalmente, se resume la organización de la monografía.

1.1 Motivación, planteamiento del problema y solución

En los últimos años, las competencias han venido ganando relevancia en el ámbito educativo, haciendo referencia a la capacidad individual demostrada para ejecutar, por ejemplo, la posesión del conocimiento, destrezas y características personales que se necesitan para satisfacer las demandas especiales o requerimientos de una situación particular [4]. De ahí que éstas fomentan y benefician la construcción de conocimiento propio por parte del estudiante, que le permitirá no sólo resolver ejercicios tipo, sino tener competencias (conocimientos, habilidades, actitudes y valores) para cualquier momento en el que deba enfrentarse a la solución de situaciones reales [4]. Dada su importancia, las competencias pueden ser definidas por entidades tales como países (a través de sus sistemas educativos), organizaciones (por ejemplo, universidades y escuelas) e individuos (por ejemplo, docentes). Según la UNESCO [5], el desarrollo del impacto de la calidad de la educación y la efectividad del aprendizaje se actualiza a través de la aplicación de competencias definidas como esenciales en el apoyo del desarrollo en contextos específicos. Éstas son adquiridas a través de los ciclos de

aprendizaje y durante todo el ciclo de vida de las personas, en la educación formal, no formal e informal.

Independientemente de su tipología y enfoque, las competencias son especificadas como resultados de aprendizaje, por lo cual, su evaluación es una tarea clave. Ésta permite identificar qué va a desempeñar o construir el estudiante y se basa en la comprobación de qué es capaz de construir o desempeñar. En este proceso se asignan calificaciones, llamadas *Niveles de Competencia* (NC), los cuales son determinados por un valor lingüístico o numérico, con base en el rendimiento del estudiante [6].

En esta dinámica, las plataformas de *Video bajo Demanda* (VoD) web, tales como Youtube y WatchKnowLearn, gozan de popularidad porque proveen una gran cantidad de recursos educativos que pueden ser usados para el apoyo de los procesos de enseñanza y aprendizaje (contenidos entretenidos, coherentes en las explicaciones y que favorecen los diferentes estilos de aprendizaje), lo cual constituye el factor motivador del presente trabajo. De acuerdo con García y Gairín [7], existen varios beneficios en el uso del VoD en la educación. Por ejemplo, facilita el razonamiento y la solución de problemas, asiste con dominio del aprendizaje, e incrementa la motivación del estudiante; al tiempo que puede ser accedido en el momento que se requiera.

Aunque estas plataformas facilitan el acceso al conocimiento, durante la tarea de búsqueda de contenidos relevantes se desarrolla un escenario como el mostrado en la Figura 1.1, en el cual, estudiantes y docentes se enfrentan a una serie de preguntas problemáticas (señaladas en diferentes colores) que no tienen la capacidad de abordar, ya sea porque generalmente no conocen sus estados y procesos cognitivos (o de sus estudiantes), o porque los servicios de VoD no proveen herramientas que faciliten la selección de contenidos relacionados con las competencias. Estas situaciones se deben a dos problemas principalmente. El primero es la *sobrecarga de información*. De acuerdo con Levy [1], es “la exposición a, o provisión de, demasiada información; una situación problemática aparece a raíz de esto”. En una plataforma de VoD, los docentes y estudiantes podrían no tener el conocimiento o el tiempo para identificar un contenido adecuado a las características de su enseñanza y aprendizaje. Entonces, episodios de estrés pueden surgir. El segundo problema, relacionado con la *metacognición* [2], consiste en que los estudiantes generalmente no son conscientes de sus estados y los procesos desarrollados en su aprendizaje, ni de su habilidad para monitorear y ajustar conscientemente tales aspectos. Así, la ausencia de cierto nivel de metacognición dificulta la selección de recursos relevantes [3].

1. Introducción

Además, los servicios de VoD actuales de acceso libre no ofrecen un modelo descriptivo formal que permita señalar que un contenido aporta en el aprendizaje de una competencia. Éste en realidad no es un problema del contenido en sí, ya que aunque posea características propicias para la enseñanza (en cuanto a contenido y edición), los esquemas de metadatos de los servicios de VoD no facilitan su vinculación con las competencias que desarrolla, dificultando la formulación de soluciones para la recuperación de información. Esto se evidencia, por ejemplo, analizando un video tomado de Youtube titulado *Teorema de Pitágoras súper fácil*¹. Este contenido está descrito por un grupo de metadatos tales como el título, una sinopsis en lenguaje natural (*Que onda aquí les dejo este vídeo, espero que les guste!!*) y una categoría (*Personas y blogs*). Aunque en ninguno de estos metadatos el creador menciona cómo el video puede contribuir al desarrollo de ciertas competencias, es evidente que éste cuenta con la suficiente información para ello. De hecho, el contenido no sólo trata aspectos del teorema sino también otros asociados a los *triángulos rectángulos y sus partes*, algunos ejemplos, y a las *potencias de segundo grado con ejemplos*, los cuales son tratados de forma preliminar al desarrollo del modelo matemático del teorema. Esto además sugiere que los contenidos pueden ser segmentados de acuerdo a competencias, lo cual puede contribuir a la recuperación efectiva de información.

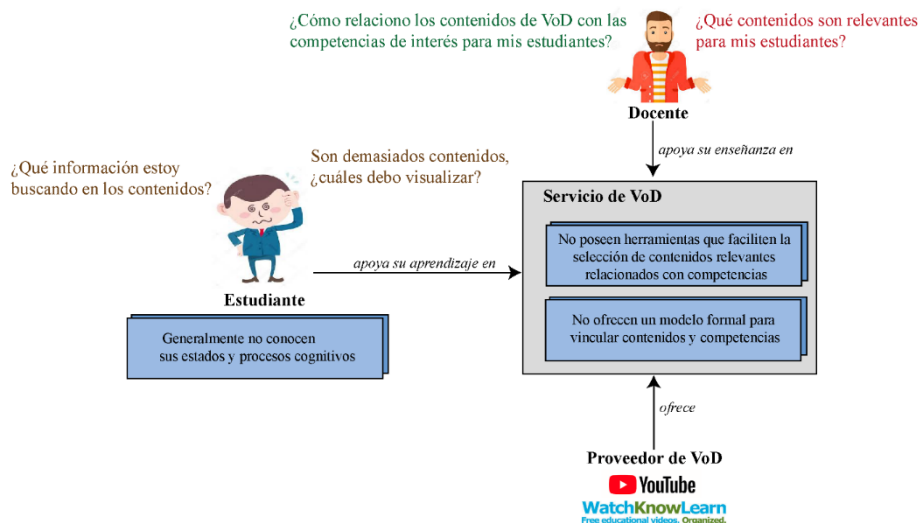


Figura 1.1 Representación del escenario problemático

Todas las situaciones problemáticas mencionadas provocan que estudiantes y docentes deban visualizar un número de contenidos antes de hallar la información más relevante para el apoyo de los procesos educativos. Esto representa una gran

¹ Este contenido puede visualizarse en el enlace: <https://www.youtube.com/watch?v=2yfkEAt2ew0>

inversión de tiempo, minando la confianza en el VoD como herramienta de apoyo al aprendizaje.

Es así como el problema que se aborda en el presente trabajo de investigación es:

Existe una dificultad inherente a la búsqueda efectiva de contenidos educativos asociados a competencias en servicios de VoD.

Con el reto de abordar el problema, como hipótesis primaria de esta investigación se plantea:

Un sistema de recomendaciones que considere el nivel de cumplimiento de las competencias permitirá mejorar la búsqueda de contenidos educativos de VoD a través de sugerencias precisas.

Esta solución es acorde con múltiples estudios que sugieren que los efectos de la *sobreespecialización* y la falta de *metacognición* pueden ser mitigados incorporando un *Sistema de Recomendaciones* (SR) a los procesos educativos [3], [8], [9], [10], [11], [12], [13]. De acuerdo con Tarus y Yousif [8], los SR filtran recursos de aprendizaje irrelevantes y recomiendan automáticamente los que pueden ser usados para satisfacer las necesidades educativas y los intereses de los estudiantes. La literatura sugiere que, con la aparición de diversos tipos de recursos multimedia en la web, los SR han venido siendo adaptados para la sugerencia, no solo de recursos de aprendizaje (por ejemplo objetos de aprendizaje, enlaces web, lecciones y cursos) [11], [14], [8], [15], sino también de personas con intereses comunes [3] y estrategias de apoyo a la enseñanza y el aprendizaje [15], [16], [17], [18].

La selección de un enfoque consolidado para los SR no es una tarea trivial. Existen enfoques que podrían ser inapropiadamente aplicados al aprendizaje enfocado en competencias. Por ejemplo, tanto el *Filtrado Basado en Contenido* (CB) como el *Filtrado Colaborativo* (CF) se enfocan en dos entidades: el usuario y el ítem a recomendar [9], [10], [19], [20]; utilizando técnicas basadas en matrices y algoritmos de *machine learning* sobre historiales de valoración. Entonces, estos enfoques no son apropiados cuando se consideran entidades adicionales como, en este caso, las *competencias*.

Por otra parte, el enfoque *consciente del contexto* es usado en diversos trabajos enmarcados en ambientes de aprendizaje, dándose a conocer como el más apropiado. Dentro del enfoque consciente del contexto se destacan iniciativas que incorporan ontologías para relacionar el perfil de usuario con los conceptos educativos formalizados en ellas [11], [8], [21], [22]. Sin embargo, en los trabajos de la literatura revisados como parte de la investigación, no ha sido posible hallar soluciones dirigidas a integrar aspectos del contexto del estudiante como su historial de NC, sus fortalezas

1. Introducción

y debilidades, las competencias, los contenidos de VoD y sus relaciones. Por consiguiente, se prescinde de una gran cantidad de conocimiento que puede ser útil en el proceso de personalización.

Precisamente, dado que en el presente trabajo se busca aprovechar el conocimiento en torno a las competencias educativas como información contextual (más allá de la información de los usuarios y los contenidos) para obtener recomendaciones más inteligentes y útiles, el SR propuesto adopta el enfoque consciente del contexto. Específicamente se incorpora como información de entrada el conocimiento sobre el *dominio de aprendizaje* (competencias educativas), del *contenido* (enfocado en el VoD) y del *usuario* (centrado en su rendimiento) al modelo funcional y matemático para estimar si un contenido puede ser útil en el proceso de adquisición de competencias de un estudiante. Así, se presume que tal utilidad no es solo una función de los contenidos en sí misma, sino también una función del contexto en el cual los contenidos pueden ser relevantes. En concreto, la propuesta se enmarca en el paradigma *modelado contextual*, dentro las categorías *conocimiento/rendimiento* y *metas de aprendizaje* de la dimensión del usuario [23].

Para la definición de los aspectos particulares de la presente investigación, se analizaron los problemas registrados en el ya mencionado escenario representado en la Figura 1.1 (especialmente en las preguntas problemáticas). Esto, apoyado en la revisión del estado del arte mostrada más adelante en el Capítulo 2, permitió identificar retos y brechas específicas que debieron ser abordadas para, en conjunto, afrontar el problema planteado. Una vez identificadas las brechas, se formularon hipótesis secundarias² que permitieron determinar los componentes del SR que, según la hipótesis primaria, permitirá mejorar la búsqueda de contenidos educativos de VoD.

La primera brecha abordada en la investigación es:

En los trabajos de la literatura revisados se observa una ausencia de atributos específicos en la descripción de los contenidos de VoD que permitan establecer vínculos lógicos entre éstos y las competencias.

En respuesta, con el reto inicial de establecer vínculos lógicos entre competencias y contenidos, se plantea la primera hipótesis secundaria:

Un esquema de metadatos permitirá definir elementos descriptivos propicios para formalizar relaciones lógicas entre competencias y contenidos de VoD.

² Las hipótesis secundarias se refieren a hipótesis más específicas que la primaria, que en conjunto abordan el problema planteado

Por otra parte, tal como se mencionó anteriormente las ontologías son importantes en los SR del ámbito educativo debido a que permiten inferir conocimiento útil en los procesos de recuperación de información [24], especialmente cuando se involucran entidades adicionales al binomio ítem-usuario. Para aprovechar tal ventaja es indispensable contar con una estructura propicia para la formalización de conocimiento sobre la cual sea posible la aplicación de técnicas de inferencia. Especialmente, la definición de atributos específicos con el suficiente significado dentro de un contexto facilita el establecimiento de vínculos lógicos entre las entidades, propiciando el descubrimiento de conocimiento inferido adicional al aseverado.

Al respecto, la revisión del estado del arte evidenció la segunda brecha abordada en el trabajo:

En los trabajos de la literatura revisados no fue posible hallar una estructura para la representación de conocimiento (ontología o taxonomía) que vincule competencias y contenidos de VoD.

Así, con el reto crear una representación de conocimiento propicia para inferir conocimiento útil en el proceso de personalización, se plantea la segunda hipótesis secundaria:

Una ontología sobre la cual el sistema infiera conocimiento a partir del descubrimiento y análisis de las asociaciones entre entidades permitirán la aplicación de técnicas de la web semántica, de tal forma que la información inferida favorezca la obtención de sugerencias de contenidos de VoD asociados a las competencias educativas.

Como se adoptó el enfoque *consciente del contexto* para modelar funcional y matemáticamente el SR propuesto, se ha decidido incorporar como entrada el conocimiento sobre el *dominio de aprendizaje*, el *contenido* (enfocado en el VoD) y el *usuario* a través de ontologías. Así, es posible aprovechar el conocimiento aseverado e inferido adyacente a los vínculos lógicos entre los usuarios, contenidos y las competencias formalizadas en bases de conocimiento como alternativa al uso de *historiales de valoración*. Al respecto, éste tipo de información puede carecer de significado en el ámbito educativo dada la falta de metacognición de algunos estudiantes. Precisamente, la oportunidad de incorporar información contextual distinta a las valoraciones hace innecesario el uso de grandes volúmenes de datos para obtener un mejor rendimiento del sistema, siendo ésta una ventaja inherente a los SR semánticos [24]. A lo anterior se suma que éstos abordan de mejor manera los problemas “*cold start*” y “*data sparsity*”, lo cual difiere de las técnicas basadas en matrices o en algoritmos de *machine learning*, para las cuales su rendimiento depende del tamaño y densidad de los datos de entrada [25], [26].

1. Introducción

La importancia del descubrimiento y análisis de asociaciones entre entidades se ilustra con el siguiente ejemplo: dos competencias como *Reconozco un triángulo rectángulo* y *Reconozco el diámetro de un círculo* están asociadas semánticamente por medio del verbo *Reconocer*, por lo tanto, éste puede ser considerado como un atributo que asocia competencias. Para un estudiante que tiene dificultades para *reconocer* figuras geométricas y sus partes, la asociación es relevante y por lo tanto sería conveniente recomendar contenidos que apoyen tales competencias.

Del citado ejemplo se puede deducir que el reconocimiento de las necesidades de los estudiantes tiene un rol importante dentro de la recomendación. De hecho, éste es un aspecto importante en la hipótesis primaria formulada. De manera general, la construcción de perfiles de usuario precisos, a partir de los cuales se pueda obtener información relevante y con el suficiente significado dentro del contexto, es un proceso inherente al modelado de cualquier SR.

Al respecto, en la dinámica de la evaluación de competencias, trabajos como [27], [28] y [29] plantean que la mejor forma de asignar NC es a través de términos lingüísticos tales como *bajo*, *medio* y *alto*. Según Fung [27], en los procesos de evaluación de aspectos educativos o empresariales (por ejemplo, para tomar decisiones), los expertos se exponen a una serie de problemas ocasionados por la asignación de juicios de valor numéricos (por ejemplo, al usar escalas de *Likert*), tales como la incertidumbre, restricciones, y el vago conocimiento que pueda tener el experto. Incluso, es difícil para él asegurar que los criterios sean igualmente aplicados al evaluar todos los aspectos. En el caso de los entornos educativos, de acuerdo con Montero, Alias, Badía, Fonseca y Vicent [28], estos problemas se dan especialmente si la evaluación involucra habilidades y capacidades observables en la solución de problemas (por ejemplo, una mesa redonda, experimentos químicos, etc.) o si implica aspectos de naturaleza subjetiva, como el *trabajo en equipo*.

Estos problemas pueden resultar críticos al considerar entidades como los *mapas de competencias* en la evaluación de las competencias [30], [7], [31] debido a toda la información que se debe ser considerada. Los *mapas de competencias* son herramientas usadas por los docentes para observar los NC alcanzados por los estudiantes en el currículo de los cursos [7]. Según el borrador de estándar “*Simple Reusable Competency Map*” (SRCM), un mapa vincula competencias y subcompetencias, indicando que una subcompetencia apoya el desarrollo de una más general; al tiempo que muestra los NC alcanzados por los estudiantes en las competencias. Es así como un mapa muestra qué competencias están más desarrolladas por los estudiantes, las carencias de espacios curriculares, los

solapamientos y las fortalezas [7]; facilitando a los docentes tomar decisiones sobre su enseñanza dirigidas al apoyo del aprendizaje de forma personalizada.

Considerando lo anterior, después de la revisión del estado del arte se pudo identificar la tercera brecha abordada en el trabajo, la cual se relaciona con la evaluación de competencias:

En los trabajos de la literatura revisados no fue posible hallar una herramienta de análisis de información cualitativa que permita construir perfiles de usuario enfocados en representar las fortalezas de los estudiantes en el aprendizaje de competencias.

Con el reto de aprovechar la información sobre el rendimiento de los estudiantes ofrecida por los mapas de competencias, haciendo énfasis en sus fortalezas y debilidades, se plantea la tercera hipótesis secundaria:

Un método computacional para la evaluación de mapas de competencias basado en un modelo computacional lingüístico permitirá construir perfiles de usuario a partir de los cuales será posible una recomendación enfocada en las necesidades educativas de los estudiantes.

En resumen, el SR planteado en la hipótesis primaria engloba aspectos como: 1) el establecimiento de vínculos lógicos entre competencias y contenidos de VoD a través de un esquema de metadatos; 2) el descubrimiento y análisis de conocimiento derivado de asociaciones entre competencias y contenidos formalizadas en una ontología; y 3) la construcción de perfiles de usuario que formalizan los NC de estudiantes a través de un método computacional para la evaluación cualitativa de mapas de competencias. Con el reto final de aprovechar todo lo anterior, se plantea la cuarta hipótesis secundaria de la investigación:

Una estrategia de recomendación que orqueste tanto la inferencia de conocimiento sobre la ontología como las necesidades de los estudiantes, representadas por los Niveles de Competencia, permitirá enfocar las sugerencias en el aprendizaje de éstas.

Con respecto a la anterior hipótesis, la incorporación de ontologías requiere el uso de técnicas de la web semántica en la definición de una estrategia de recomendación apropiada. Como era de esperar, las técnicas tradicionalmente usadas en el enfoque consciente del contexto (por ejemplo, matrices y clasificación por *machine learning*) no pudieron ser utilizadas en la presente investigación ya que sus algoritmos están diseñados para realizar tareas de predicción o cómputo de la ponderación (*ranking*) a partir de historiales de valoración de los usuarios por los ítems. En cambio, se propone una estrategia de recomendación caracterizada por el cálculo de la relevancia semántica entre las competencias en las cuales los estudiantes tienen necesidades educativas y otras competencias formalizadas en las ontologías a partir de

conocimiento aseverado e inferido, siendo esta la base para el descubrimiento y ponderación de los contenidos de VoD. A este efecto, se han incorporado las características del contexto de las competencias y de las ontologías en el cálculo de la relevancia semántica. Precisamente, la incorporación de técnicas tales como las semánticas en la presente investigación y en otros trabajos [11], [8], [21], [22] o como las de lógica fuzzy en [32], [33], ha requerido la definición de modelos funcionales y matemáticos novedosos que permitan aprovechar las ventajas que proveen, por ejemplo, para aumentar la precisión del sistema o reducir los efectos negativos del cold start y el data sparsity [34], [33]. Por esto y, teniendo en cuenta que tales técnicas se presentan como una alternativa de uso a las basadas en matrices y *machine learning*, los mencionados sistemas, incluido el propuesto en la presente investigación, aparentemente no se enmarcan en lo comúnmente establecido en la literatura de base, pero sin duda han aportado al conocimiento en el área de los SR en la comunidad investigativa en los últimos años.

La idea de ponderar la relevancia semántica entre las competencias para ponderar la utilidad de un contenido para un usuario indica que el SR propuesto puede ser también clasificado dentro del enfoque *Basado en Contenido*. Esto no excluye al enfoque *consciente del contexto*, ya que una forma de solución consiste en condicionar el funcionamiento del enfoque *basado en contenido* o *colaborativo* por medio de información contextual.

Como resultado de las hipótesis planteadas hasta aquí, se establece la solución del SR propuesta en el presente trabajo, la cual se presenta en la Figura 1.2. Como puede observarse, los componentes abordan puntualmente las preguntas señaladas en el escenario problemático de la Figura 1.1, existiendo una correspondencia entre los colores. Por ejemplo, el *Esquema de Metadatos para la descripción de contenidos de VoD de acuerdo a las competencias* (marcado en verde) aborda la pregunta del *Docente: ¿Cómo relaciono lo contenidos de VoD con las competencias de interés para mis estudiantes?*, ya que generalmente, los esquemas de metadatos ofrecen elementos y tipos de datos, y vocabularios controlados que en conjunto constituyen una guía en el proceso de marcación de contenidos y en la vinculación de las entidades involucradas.

En síntesis, el SR considera elementos pedagógicos propios de los sistemas educativos tales como las competencias y los mapas de competencias con el fin de integrar a la solución propuesta cierta dimensión del contexto del estudiante, posibilitando así enfocar el proceso de recomendación en las necesidades de éste. Entonces, estos elementos son explotados por los componentes del SR, siguiendo un

proceso explicado de forma general como sigue (todos los componentes de la solución son desarrollados con detalle en lo que resta de la presente monografía):

1. Con base en las *Competencias* y *Mapas de competencias* definidos en los currículos de los cursos y en su experiencia en la visualización de contenidos de video, los *Docentes* tienen un rol importante en la creación de descripciones de, y asociaciones entre, contenidos dispuestos en los servicios de VoD, competencias y mapas tomando como base el *Esquema de Metadatos para la descripción de contenidos de VoD de acuerdo a las competencias*.
2. Toda la información producida es almacenada en una base de datos, llamada aquí *Descripción de contenidos, competencias y mapas*.

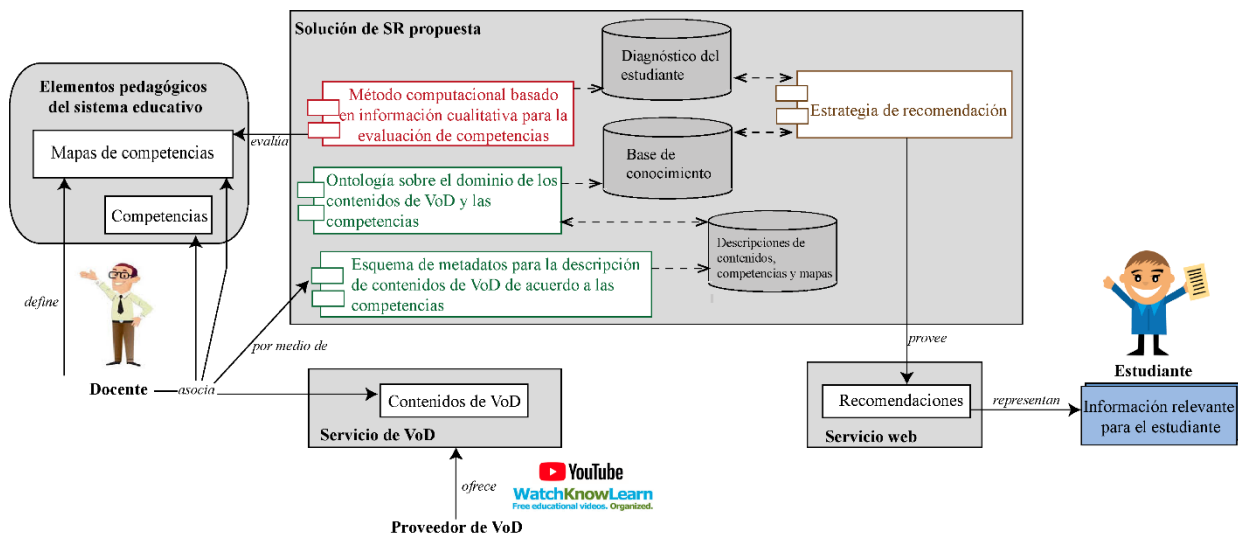


Figura 1.2 Solución planteada del Sistema de Recomendaciones

3. Posteriormente, la información de la base de datos es interpretada y adaptada a la *Ontología sobre el dominio de los contenidos de VoD y las competencias* para generar una *Base de conocimiento*, la cual es factible de procesar por medio de técnicas de la web semántica. Este enfoque es principalmente útil para reutilizar los repositorios de metadatos existentes. En caso de no ser necesaria tal reutilización, otro enfoque consiste en generar la *Base de conocimiento* directamente sin pasar antes por la creación de las descripciones a través de metadatos. Sea de una u otra forma, es importante lograr una correspondencia entre el modelo de descripción del esquema de metadatos y la ontología; esto teniendo en cuenta que la consideración de esquemas de metadatos puede reducir los costos en el diseño de ontologías [35].

1. Introducción

4. Con el propósito de enfocar las recomendaciones en las necesidades de los estudiantes, el *Método computacional basado en información cualitativa* toma la información consignada en los mapas de competencias para realizar un diagnóstico del aprendizaje del estudiante en términos de los NC, todo esto determinado por la precisión y objetividad ofrecida por la lógica difusa.
5. La información del diagnóstico, que no es otra cosa que la identificación precisa de las fortalezas y debilidades de los estudiantes, es almacenada en una base de datos llamada *Diagnóstico del estudiante*.
6. Finalmente, la *Estrategia de recomendación* orquesta el diagnóstico de los estudiantes y la base de conocimiento para generar recomendaciones de contenidos de VoD por medio de la aplicación de técnicas de inferencia propias de la web semántica.

Dada la dificultad que supone la inserción de las recomendaciones en los servicios ofrecidos por los proveedores de VoD, la solución planteada sugiere presentar las recomendaciones en un servicio web independiente que vincule los contenidos a través de enlaces web. Esto teniendo en cuenta la facilidad que existe hoy en día para embeber contenidos de video en cualquier tipo de página o aplicación por medio de las URL.

1.2 Objetivos

A partir de los retos mostrados en la motivación, a continuación, se presenta el objetivo general y los objetivos específicos del trabajo de investigación.

1.2.1 Objetivo general

Proponer un sistema de recomendaciones de contenidos educativos de VoD centrado en las necesidades en torno a las competencias educativas.

1.2.2 Objetivos específicos

1. Adaptar uno o más esquemas de metadatos para la descripción de contenidos educativos de VoD en torno a las competencias.
2. Diseñar una representación del conocimiento basada en Ontologías que describa las relaciones entre las competencias educativas y los contenidos de VoD.
3. Proveer un método computacional basado en información cualitativa para la identificación del nivel de cumplimiento de las competencias individuales a partir de las actividades de evaluación.

4. Diseñar al menos una estrategia para la recomendación de contenidos educativos de VoD basada en medidas de similitud semántica, a partir del nivel de cumplimiento de las competencias educativas.
5. Evaluar experimental y comparativamente la precisión de las recomendaciones a través de un prototipo, considerando un grupo de competencias de un área de conocimiento específica.

1.3 Organización de la monografía

La presente monografía está organizada de la siguiente manera:

Capítulo 1: Introducción. En este capítulo se explica la motivación para el desarrollo de la investigación, el planteamiento del problema y la solución propuesta; además, se especifican los objetivos y la organización de la monografía.

Capítulo 2: Base conceptual y estado del arte. Capítulo que presenta inicialmente una base conceptual que introduce información relevante en la presente investigación. Esto de acuerdo a núcleos temáticos tales como: *sistemas de recomendaciones*, el *video bajo demanda*, *las competencias y mapas de competencias*, *metadatos y calidad de metadatos*, *ontologías*, y *modelos computacionales lingüísticos*. Posteriormente, el capítulo describe un estado del arte, mostrando un estudio del conocimiento acumulado en la literatura sobre los siguientes aspectos: *sistemas de recomendaciones educativos*, *esquemas de metadatos del ámbito de las competencias y contenidos multimedia*, *ontologías existentes relacionadas con la investigación*, y *modelos computacionales lingüísticos para la evaluación de competencias*.

Capítulo 3: Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas. Este capítulo propone un esquema de metadatos que provee un modelo descriptivo que integra el modelado de contenidos de VoD, competencias y mapas de competencias; haciendo especial énfasis en sus atributos y relaciones lógicas. En concreto, el esquema es el resultado de integrar: 1) un perfil de aplicación del estándar *TV-Anytime* [36], el cual es un aporte que enfoca la descripción de los contenidos de video en el servicio de VoD; 2) una extensión del estándar IEEE RCD ("*Reusable Competency Map*") [37], propuesta aquí para el modelado de competencias educativas a través de atributos específicos; y 3) el borrador de estándar SRCM ("*Simple Reusable Competency Map*") [38], el cual ha sido tomado para la descripción de mapas de competencias. Además, el capítulo introduce una evaluación de la extensión del IEEE RCD para competencias basada en la Calidad de los Metadatos (CM).

Capítulo 4: Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas. Este capítulo presenta una ontología que formaliza los conceptos y relaciones asociadas a las competencias, mapas de competencias, segmentos de video y contenidos de VoD. La ontología es el resultado del mapeo del esquema de metadatos que se propone en el Capítulo 3 en elementos ontológicos (propiedades de datos y objetos, clases e instancias) a partir ontologías existentes. La ontología busca articular de manera natural (desde el punto de vista de las tecnologías existentes) y a través de sus atributos las relaciones entre las siguientes entidades: los contenidos de VoD, las competencias y los mapas de competencias. En consecuencia, la ontología permite la generación y aplicación de estrategias de recomendación personalizadas basadas en el descubrimiento y ponderación de asociaciones semánticas entre las entidades que ésta formaliza. Además, este capítulo introduce una evaluación de la ontología enmarcada en la característica de *completitud* [39], a partir del cual se mide si se cubre el dominio de interés de forma apropiada.

Capítulo 5. Método computacional basado en información cualitativa para la evaluación de competencias. Se presenta un método computacional basado en el modelo de *2-tuplas lingüísticas*, el cual es consciente de la estructura de los mapas de competencias para el cómputo de *niveles de competencia*. Éstos son usados en el sistema de recomendaciones para construir perfiles de usuario precisos, utilizados para enfocar las sugerencias en las fortalezas y debilidades de los estudiantes en la adquisición de competencias. Además, este capítulo introduce una evaluación de precisión, tiempo de ejecución y consumo de memoria, de manera comparativa con el *Modelo Basado en el Principio de Extensión*.

Capítulo 6: Estrategia de recomendación basada en ontologías. Se presenta una estrategia de recomendación enfocada en competencias y consciente del contexto que obtiene listas de recomendaciones de contenidos de VoD ajustadas a las necesidades de los estudiantes. Ésta combina la lógica de los enfoques tradicionales y la capacidad de descubrimiento de conocimiento propia de las tecnologías de la web semántica, específicamente, la relacionada con el descubrimiento y ponderación de asociaciones semánticas complejas. Para esto, la estrategia “orquesta” la ontología propuesta en el Capítulo 4 y el método computacional del Capítulo 5 para descubrir conocimiento relacionado con las necesidades educativas, manteniendo así el carácter personalizado de las recomendaciones. Concretamente, la estrategia consiste en: 1) calcular los grados de necesidad de los usuarios por los conceptos educativos formalizados en la ontología; 2) inferir en una base de conocimiento (construida a partir de la ontología propuesta) las competencias relevantes vinculadas con las necesidades del estudiante por medio del descubrimiento y ponderación de

asociaciones semánticas complejas; 3) computar la ponderación de segmentos de video relevantes; 4) construir listas de recomendación basadas en la ponderación de contenidos de video. Aquí, se hace uso de los avances alcanzados por la comunidad investigativa en la web semántica en cuanto al descubrimiento de conocimiento útil en la recuperación de información.

Capítulo 7. Evaluación del Sistema de Recomendaciones. Este capítulo describe una evaluación del sistema de recomendaciones propuesto, con base en la precisión de las sugerencias (de acuerdo al criterio de expertos) y los tiempos de ejecución; todo a partir de un prototipo construido como aporte de la investigación. Esta evaluación se realiza comparativamente con relación a un enfoque de emparejamiento sintáctico soportado en el coeficiente Sørensen-Dice.

Capítulo 8. Conclusiones y trabajos futuros y productos de la investigación. Este capítulo presenta inicialmente las conclusiones derivadas de la investigación. Finalmente, muestra las líneas de trabajos futuros propuestas.

Anexo A. Aspectos complementarios de la base conceptual. Anexo que muestra información complementaria a la base conceptual del Capítulo 2 acerca de: *características de calidad de los metadatos, ejemplos de aplicación de modelos computacionales lingüísticos y asociaciones semánticas complejas.*

Anexo B. Análisis de esquemas de metadatos del ámbito de las competencias. Anexo que presenta con detalle el análisis de los esquemas de metadatos del ámbito de las competencias, el cual se introduce resumidamente en el estado del arte del Capítulo 2.

Anexo C. Versión extendida de la evaluación de la extensión del IEEE RCD. Anexo que presenta de manera extendida la evaluación de la extensión del IEEE RCD mostrado en el Capítulo 3.

Anexo D. Versión extendida del estudio de ontologías existentes del dominio del VoD y las competencias. Anexo que presenta detalladamente el análisis de ontologías existentes relacionadas con la investigación, el cual se introduce de forma resumida en el Capítulo 2.

Anexo E. Definición formal del perfil de aplicación de TV-Anytime. Anexo que presenta la definición del perfil de aplicación de *TV-Anytime* de acuerdo a la nomenclatura de este estándar y la descripción de los elementos en cuanto a su significado.

Anexo F. Información complementaria de la extensión del IEEE RCD. Anexo que presenta la definición de la extensión del IEEE RCD de acuerdo a la nomenclatura de este esquema y la descripción de los elementos en cuanto a su significado.

Anexo G. Aspectos complementarios de la adaptación de “Learning Object Metadata Ontology”. Anexo que presenta aspectos complementarios acerca del proceso de adaptación de la ontología “*Learning Object Metadata Ontology*” introducido en el Capítulo 4.

Anexo H. Versión extendida de la Adaptación de “Ontology for Media Resources”. Anexo que presenta en detalle el proceso de adaptación de la ontología “*Ontology for Media Resources*” resumido en el Capítulo 4.

Anexo I. Versión extendida de la Adaptación de “Competency Model Ontology”. Anexo que presenta en detalle el proceso de adaptación de la ontología “*Competency Model Ontology*” resumido en el Capítulo 4.

Anexo J. Versión extendida de la construcción de OntoRCM. Anexo que muestra detalladamente el proceso de construcción de la ontología para el modelado de mapas de competencias definido resumidamente en el Capítulo 4.

Anexo K. Versión extendida de la evaluación de la ontología propuesta. Anexo que presenta todo el proceso desarrollado en la evaluación de la ontología propuesta, el cual se introduce resumidamente en el Capítulo 4.

Anexo L. Versión extendida de la evaluación del método computacional propuesto. Anexo que muestra en detalle la evaluación del método computacional propuesto en el Capítulo 5.

Anexo M. Pseudoalgoritmos propuestos. Anexo que muestra los pseudoalgoritmos de todos los algoritmos propuestos en la investigación.

Anexo N. Construcción de listas de recomendaciones para la estrategia de recomendación. Anexo que presenta el proceso de la estrategia de recomendación encargado de construir las listas de recomendaciones.

Anexo O. Prototipos. Anexo que detalla aspectos técnicos (diagramas de componentes y de flujo, bases de datos, tecnologías) de los prototipos construidos como aporte de la investigación (prototipos del método computacional y del sistema de recomendaciones).

Anexo P. Resultados complementarios de tiempos de ejecución del prototipo del sistema de recomendaciones propuesto. Anexo que muestra resultados complementarios de tiempos de ejecución obtenidos a partir de la evaluación del sistema de recomendación presentada en el Capítulo 7.

Anexo Q. Discusión. Anexo que establece una discusión acerca del uso esperado de lo propuesto, sus limitaciones y los resultados de precisión del sistema de recomendaciones

Anexo R. Resumen de productos derivados de la investigación. Anexo que resume productos tales como: publicaciones, prototipos, bases de datos y bases de conocimiento.

Anexo S. Versión previa de la extensión del IEEE RCD. Anexo que presenta una versión previa de la extensión del IEEE RCD propuesta en el Capítulo 3, la cual fue evaluada por expertos con el propósito de establecer mejoras. Éstas ya están incluidas en el esquema propuesto.

Capítulo 2

Base conceptual y estado del arte

La motivación, planteamiento del problema y solución presentados en el Capítulo 1 definieron una serie de hipótesis que, en términos generales, juegan el papel de requisitos a cumplir en el trabajo de investigación. Tales hipótesis, representadas por componentes del *Sistema de Recomendaciones* en la Figura 1.2, permitieron identificar unos núcleos temáticos que contienen toda la información manejada a lo largo del desarrollo de la investigación. Además, permitieron vislumbrar los aspectos a tener en cuenta durante la revisión del estado del arte.

Con lo anterior, y como resultado del estudio de la literatura, se obtuvo una base conceptual y un estado del arte que se presentan en las siguientes secciones.

2.1 Base conceptual

Tal como se mencionó anteriormente, los componentes del SR planteado permitieron identificar un grupo de núcleos temáticos relevantes que involucran información manejada a lo largo de la investigación. Debido a la naturaleza del trabajo, los primeros núcleos temáticos de interés para la investigación son: 1) los *Sistemas de recomendaciones*, haciendo énfasis en sus enfoques; 2) el *VoD*, señalando su utilidad dentro de los procesos educativos; y 3) el concepto de *Competencias educativas y Mapas de competencias*, desde dos puntos de vista: pedagógico, en el cual se especifican definiciones formales halladas en la literatura; y el de los esquemas de metadatos, los cuales definen elementos de descripción propicios para generar, compartir y reutilizar conceptos comunes a través de medios como internet.

Por otra parte, la necesidad de construir un *Esquema de Metadatos para la descripción de contenidos de VoD de acuerdo a las competencias* obligó a dirigir la atención hacia los *Metadatos* en cuanto a su definición. Dado que el rendimiento de los SR depende en gran parte de la calidad de las descripciones de los ítems, fue necesario abordar el tema de la *Calidad de los Metadatos*.

Así mismo, debido al componente *Ontología sobre el dominio de VoD y las competencias* surgió la necesidad de definir las *Ontologías* desde el campo de la informática.

Debido a que la *Estrategia de recomendación* se basa en la aplicación de técnicas de la web semántica para la inferencia de conocimiento, en esta sección también se introduce el concepto de *Asociación semántica compleja*, el cual cobra relevancia cuando en las ontologías se definen relaciones entre las entidades involucradas a través de sus atributos. Más información sobre este tipo de estructuras se puede hallar en el Anexo A (Sección A.3). En concreto, debido a que en la presente investigación se busca establecer vínculos lógicos entre contenidos de VoD, competencias y mapas de competencias, las *Asociaciones semánticas complejas* pueden ser aprovechadas para inferir conocimiento útil dentro del proceso de personalización de las recomendaciones.

Finalmente, el componente *Método computacional basado en información cualitativa* se soporta en los modelos matemáticos enmarcados dentro del *Enfoque lingüístico difuso*, por lo cual también es tratado en la presente base conceptual.

2.1.1 Sistemas de recomendaciones

Según Ricci, Rokach y Shapira [40] los Sistemas de Recomendaciones (SR) son “herramientas y técnicas software que proveen sugerencias de ítems para ser usadas por el usuario”. Las sugerencias se relacionan con diferentes procesos de toma de decisión, tales como *qué productos comprar, qué música escuchar o qué noticia leer*. Los SR están dirigidos principalmente a personas que carecen de la experiencia personal o competencia suficiente para evaluar la gran cantidad de ítems que un sitio web, por ejemplo, puede ofrecer. Por tal razón, son una alternativa para abordar los problemas acaecidos por la gran cantidad de contenidos educativos de VoD disponibles en la web o la televisión.

Los SR se pueden clasificar de acuerdo a diferentes enfoques. A continuación, se listan los más populares y se explica la elección de uno de ellos para el SR propuesto.

- *Basado en Contenido* (CB): el sistema recomienda ítems similares a los que han sido del gusto del usuario en el pasado [25]. La similitud de los ítems es calculada con base en los atributos asociadas a los ítems comparados. En este enfoque, la conformación de las listas de recomendación se realiza con base en el cálculo de una medida del interés del usuario por un ítem no visualizado a través de técnicas de predicción (de valoraciones o *ratings*), clasificación (con algoritmos de *machine learning*) o *ranking*.

2. Base conceptual y estado del arte

- *Filtrado Colaborativo (CF)*: el sistema recomienda al usuario ítems que han sido del gusto de otros usuarios con gustos similares [25]. La similitud en el gusto de dos usuarios se calcula con base en la similitud en el historial de calificaciones de los usuarios. En este enfoque se calcula una medida del interés del usuario por un ítem usando técnicas de predicción.
- *Consciente del contexto*: el sistema explota la información del contexto que afecta las situaciones y preferencias de los usuarios con el propósito de recomendar ítems que son realmente relevantes para cambiar sus necesidades [41]. A diferencia de los enfoques CB y CF, en este enfoque la información contextual puede ser incorporada en la recomendación de tres modos:
 - *Prefiltrado contextual*: en este paradigma la información sobre el contexto es usada para seleccionar el conjunto de datos relevante, y las recomendaciones son obtenidas a través de cualquier sistema tradicional con el enfoque CB o CF.
 - *Postfiltrado contextual*: en este paradigma la información contextual es ignorada inicialmente, y las recomendaciones son obtenidas a través de cualquier sistema CB o CF usando el conjunto de datos completo. Posteriormente, el conjunto de recomendaciones es ajustado (contextualizado) por cualquier usuario usando la información contextual.
 - *Modelado contextual*: en este paradigma la información contextual es usada directamente en la técnica de modelado en el proceso de obtención de recomendaciones. La información se toma como parte del conocimiento de entrada del SR, por lo cual, la función que define la medida de la utilidad o preferencia (calculadas en la práctica como rating o ranking) de un ítem para un usuario depende de tres dimensiones: usuario, ítem, contexto. Entonces, se consideran entidades adicionales (parte del contexto) a las tradicionales (ítem y usuario).

La selección de uno de estos enfoques para la propuesta no es una tarea trivial. Se ha decidido reconocer las implicaciones de cada uno con respecto a su uso en el ámbito educativo. Por ejemplo, de acuerdo con Jannach, Zanker, Felfernig y Friedrich [25] y Desrosiers y Karypis [26] el rendimiento de los enfoques *Basado en Contenido* (CB) y *Colaborativo* (CF) en las recomendaciones depende de la cantidad de valoraciones que los usuarios hayan expresado por los contenidos, por ejemplo, a través de una escala numérica o un *like* o *dislike* (siendo éste el caso de Youtube y Netflix). Con respecto al CB, es necesario que un usuario exprese explícitamente un gran número de valoraciones por los contenidos visualizados, mientras que para el CF es necesario que un gran número de usuarios coincida en la valoración de los mismos contenidos para que los algoritmos sean precisos. Cuando estas condiciones no se dan, es posible que la precisión de las recomendaciones sea baja o que se produzcan los problemas

data sparsity o *cold start*, los cuales consisten en la incapacidad del sistema para generar recomendaciones porque no cuenta con las suficientes valoraciones explícitas. Con respecto al CB, además del *cold start*, es usual el problema de la *sobreespecialización*, el cual consiste en recomendar contenidos muy parecidos entre sí [34]. Al respecto, dado que no se creyó conveniente utilizar las valoraciones de los estudiantes por los contenidos como una fuente confiable de información debido a la posible falta de metacognición, además de que no se contó con una colección de datos consolidada que representara la efectividad de los contenidos en la adquisición de competencias, y ante la dificultad de crear una con tales características debido a los plazos de la investigación, no se vio como la mejor opción el uso de los enfoques CB o CF para el sistema de recomendaciones propuesto. Por lo anterior, se decidió adoptar un enfoque *consciente del contexto*, el cual ofrece la oportunidad de compensar la falta de información sobre valoraciones confiables de los contenidos o del impacto de éstos en la adquisición de competencias a través de la incorporación de la información contextual en torno a las competencias.

Con el enfoque *consciente del contexto* se busca aprovechar el conocimiento en torno a las competencias educativas como información contextual (más allá de la información de los usuarios y los ítems) para obtener recomendaciones más inteligentes y útiles. Específicamente se incorpora como información de entrada el conocimiento sobre el dominio de aprendizaje (competencias educativas), del contenido (enfocado en el VoD) y del usuario (centrado en su rendimiento) al modelo funcional y matemático para estimar el interés de un estudiante por un contenido. De tal manera, se presume que tal interés no es solo una función de los contenidos en sí misma, sino también una función del contexto en el cual los contenidos pueden ser relevantes. En concreto, la propuesta se enmarca en el paradigma *modelado contextual*, dentro las categorías *conocimiento/rendimiento* y *metas de aprendizaje* de la dimensión del usuario [23].

2.1.2 Video bajo Demanda

El Video bajo Demanda (VoD) es un servicio que permite a los usuarios elegir contenidos de video desde una *Guía Electrónica de Programación* (GEP) en el momento que se requiera [42]. Una característica fundamental de este servicio es que posibilita al usuario el control de reproducción (por ejemplo pausar, adelantar, y reproducir) [43].

En la actualidad, el VoD en la web ha ganado popularidad debido a que facilita el acceso a los contenidos a cualquier persona que tenga una suscripción a internet, sin

2. Base conceptual y estado del arte

la implicación de los operadores tradicionales en el control o la distribución del contenido. Por tal ventaja, el VoD se presenta como una herramienta para apoyar la educación. Según [44], los servicios de VoD están posicionados idealmente para impulsar la búsqueda de la calidad de la educación e incluso ayudar a subsidiar el costo de ella. Este es el caso plataformas como Youtube y WatchKnowLearn, las cuales ofrecen un punto de encuentro para quienes exhiben y visualizan un video; siendo esta una circunstancia favorable para realizar actividades de enseñanza y aprendizaje.

Estudios sobre el uso del VoD en los procesos educativos consideran que el video es un poderoso medio para movilizar un espectro amplio de contenidos, desde conocimientos científicos hasta emociones humanas a través del drama [44], [45]. Así, los docentes pueden hacer uso de los videos para realizar una intervención *centrada en la enseñanza*, en la cual el docente puede ilustrar conceptos, mostrar la aplicación de un video en el mundo real, facilitar discusiones y análisis colectivos de contenidos, seguir procedimientos, motivar o inspirar [44], [45], [46]; o una intervención *centrada en el aprendizaje*, en la cual el docente puede hacer uso de plataformas de VoD para formar habilidades en los estudiantes en la búsqueda, selección y exposición de un tema a tratar [44], [47]. Dadas estas utilidades, la presente investigación se centra en el VoD como una fuente de objetos de aprendizaje que pueden ser ofrecidos pertinentemente a los estudiantes manteniendo el carácter personalizado de los SR.

2.1.3 Competencias educativas

En la literatura se pueden hallar diferentes definiciones sobre *competencias*. Por ejemplo, de acuerdo con [48], es el “grupo de habilidades que deben aprender los pupilos y para las cuales los docentes deben estar preparados para enseñar”. Para el proyecto *TENCompetence* [49] una *competencia* es la “disposición o atributo latente de un individuo, equipo u organización que es situacional, identificado y definido en una comunidad de práctica”. Así mismo, para *MedBiquitous* [50], es la “posesión de conocimiento, habilidad y actitud suficiente y necesaria de un individuo para permitirle desarrollar un trabajo específico de forma segura y efectiva”. Esta definición es reforzada por el “*European Qualifications Framework*” (EQF) [51], el cual afirma que las competencias se “refieren a habilidades y conocimientos”. De acuerdo con lo anterior, el concepto de *competencia* sugiere la internalización de conocimiento y habilidad a tal punto de que ellas y la selección instintiva de los conocimientos y habilidades más apropiadas, se transforman en una parte inherente a la respuesta de una persona ante un problema que requiere solución.

Ahora bien, desde la informática, las competencias han sido conceptualizadas a través de estándares de metadatos como el “*Reusable Definition of Competency or*

Educational Objective” (RDCEO) [52], el “*IEEE Reusable Competency Definition*” (IEEE RCD) [37]. Éstos han sido definidos para establecer un “puente” entre el área de la informática y el de la educación. Debido a la gran variedad de conceptos, definiciones y aspectos relacionados con las competencias, estos estándares son “agnósticos” de una definición exacta de éstas, y optan por ofrecer una forma estándar (hasta cierto punto), que puede ser extensible, para formalizar cualquier aspecto relacionado. Entonces, incorporan diferentes elementos de datos con el propósito de proporcionar un modelo general lo suficientemente específico como para describir las competencias de la forma más precisa posible, y lo suficientemente minimalista como para permitir la adaptación a los diferentes conceptos de competencia.

En general, los estándares pretenden abarcar cuatro tipos de información propios para construir descripciones de competencias y hacerlas intercambiables entre sistemas: 1) *definición genérica y reusable de la competencia*; 2) *contexto*; 3) *evidencias*; y 4) *dimensiones*. Considerando éstos tipos de información, se presume que los esquemas existentes pueden ser el punto de partida para la integración de nuevos elementos de datos, vocabularios y tipos de datos a través de extensiones. Para esto, se realizó un análisis más profundo acerca de las posibilidades, fortalezas y debilidades de los esquemas existentes a partir del cual se establecieron las bases para la construcción del esquema de metadatos propuesto (Sección 2.2.2).

2.1.4 Mapas de competencias

Según García y Gairín [7], los mapas de competencias son “herramientas usadas por los docentes para observar los NC alcanzados por los estudiantes en el currículo de una materia, área, programa o curso”. Una buena aproximación a la estructura de un mapa de competencias es ofrecida por el borrador de estándar “*Simple Reusable Competency Map*” (SRCM) [38]. Éste define un modelo de información simple y reusable, para ser usado en la descripción, referencia e intercambio de datos que definen relaciones entre competencias en el contexto educativo. Cada mapa es descrito por un *Grafo Acíclico Dirigido* (GAD), en el cual las competencias son representadas por nodos que formalizan relaciones de subsunción entre ellos, entendiendo entonces que una competencia puede ser descompuesta en subcompetencias, o que tales subcompetencias contribuyen a ella. La Figura 2.1 muestra un fragmento de un mapa de competencias basado en el modelo del SRCM, el cual fue provisto por un docente de matemáticas que usa los *Estándares Básicos de Competencias de Colombia* [53]. Como puede observarse, los mapas tienen las siguientes características:

- Las competencias se descomponen en otras competencias (subcompetencias).

2. Base conceptual y estado del arte

- La estructura establece relaciones de subsunción entre sus elementos, las cuales son etiquetadas con un peso que representa la importancia de una subcompetencia en el desarrollo de una competencia de un nivel superior.
- El mapa define *niveles de competencia* para las competencias.

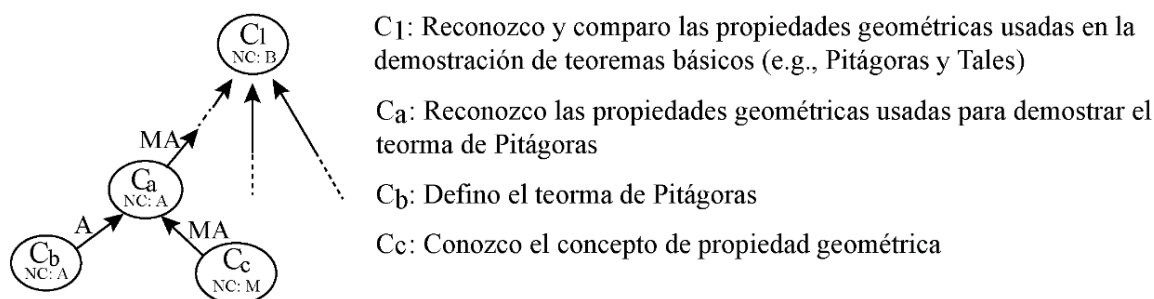


Figura 2.1. Fragmento de un mapa de competencias basado en el modelo de SRCM.
Fuente: propia.

En el ejemplo presentado, los NC y pesos fueron asignados de acuerdo a un grupo de siete términos lingüísticos como el siguiente:

$$S = \{N: Nulo, MB: Muy Bajo, B: Bajo, M: Medio, A: Alto, MA: Muy Alto, T: Total\} \quad (2.1)$$

Es importante mencionar que, aunque la misma estructura puede ser usada para varios estudiantes, cada mapa de competencia corresponde a un solo estudiante porque registra sus propios NC.

La incorporación de un esquema de metadatos para la descripción de competencias en el esquema propuesto es un aspecto importante en este trabajo, ya que, además de permitir la vinculación ponderada de competencias, posibilita la formalización de los mapas a través de la ontología propuesta; siendo esto aprovechado para: 1) establecer vínculos lógicos entre competencias; 2) generar un grupo de modelos de inferencia que aportan conocimiento explotable por el SR sobre las relaciones de subsunción entre las competencias.

2.1.5 Metadatos y calidad de los metadatos

De acuerdo con Baca [54], los *metadatos* son “datos adicionales que se relacionan con un contenido existente y clarifican las propiedades y semántica del contenido anotado”. Principalmente, los metadatos permiten la correcta recuperación de información, para lo cual es esencial la construcción de registros y repositorios de calidad. Al respecto, la *Calidad de los Metadatos* (CM) refleja el grado con el que los metadatos realizan sus funciones esenciales bibliográficas de búsqueda, localización, uso, procedencia, autenticación y administración [55].

Los metadatos son usados en este trabajo por las siguientes razones:

1. Permiten describir de forma organizada los atributos específicos de las entidades involucradas en esta investigación: los contenidos de VoD, las competencias y los mapas de competencias; y facilitan la definición de vínculos lógicos entre éstas.
2. Los esquemas de metadatos son adaptables o mapeables en ontologías. Ejemplos son el esquema “*Learning Object Metadata*” (LOM) [56] y la ontología “*Learning Object Metadata Ontology*” [57], y el esquema *TV-Anytime* [36] y la ontología “*Ontology for Media Resources*” [58]. Esta característica permite añadir semántica a los metadatos, lo cual permite aprovechar el uso de técnicas de la web semántica para la inferencia de conocimiento.

Precisamente, estas características son aprovechadas aquí para definir una ontología según las características del esquema de metadatos propuesto.

Por otra parte, trabajos hallados en la literatura han definido diversos criterios usados para medir la calidad de los metadatos, [59] y [55]. En concreto, en [55] se definen siete características de la calidad de los metadatos: *completitud, precisión, procedencia, conformidad con las expectativas, consistencia y coherencia lógica, atemporalidad y accesibilidad*. Una descripción resumida de éstas puede consultarse en el Anexo A (Sección A.1).

Una evaluación enfocada en la CM de una versión preliminar del *esquema de metadatos propuesto para el modelado de competencias* fue realizada para determinar en qué medida éste posee las características de calidad y realizar las modificaciones necesarias para mejorarlo, obteniendo así como resultado el esquema descrito en este trabajo. Esta información se extiende durante la definición del esquema en la Sección 3.2 y la descripción de la evaluación en la Sección 3.4.

2.1.6 Ontologías

El concepto de *ontología* fue traído al área de la Inteligencia Artificial (IA), en el que Gruber [60] definió una ontología como una “especificación explícita y formal de una conceptualización compartida”. En esta definición, convertida ya en estándar, *conceptualización* se refiere a un modelo abstracto del dominio de aplicación específico que se representa en la ontología; *explícita* hace referencia a la necesidad de especificar los diferentes conceptos incluidos en la misma; *formal* indica que la especificación debe llevarse a cabo mediante un lenguaje formalizado, y *compartida* alude al hecho de que una ontología debe representar conocimiento consensuado, ampliamente aceptado, al menos en la comunidad de usuarios en la que ésta vaya a ser utilizada.

Una definición más concreta de ontología fue dada por Weigand [61] en la que se refería a este tipo de estructuras como “bases de datos que describen los conceptos

2. Base conceptual y estado del arte

de un determinado dominio de aplicación, así como sus propiedades y las relaciones existentes entre ellos”. Ésta es precisamente la noción de ontología utilizada en el dominio de la web semántica y, por lo tanto, la que se acoge en este trabajo.

Además de ofrecer la capacidad de formalizar un dominio específico, las ontologías permiten que el conocimiento que en ellas se representa pueda ser reutilizado fácilmente, entre usuarios y entre sistemas informáticos. Esta característica es la que ha convertido a las ontologías en la piedra angular de la *web semántica*, en lo que a conceptualización de conocimiento compartido se refiere.

Como se mencionó en la Sección 2.1.1, se ha adoptado el enfoque consciente del contexto para modelar funcional y matemáticamente el SR propuesto. Específicamente, se incorpora como información de entrada el conocimiento sobre el dominio de aprendizaje (competencias educativas), el contenido (enfocado en el VoD) y el usuario (centrado en su rendimiento). Se ha decidido representar tal información a través de ontologías. Esto con el propósito de aprovechar el conocimiento aseverado e inferido adyacente a los vínculos lógicos entre los usuarios, contenidos y las competencias (incluidos los mapas de competencias) formalizados en bases de conocimiento como alternativa al uso de historiales de valoraciones, los cuales pueden carecer de importancia en el ámbito educativo. Precisamente, la oportunidad de incorporar información distinta a las valoraciones hace innecesario el uso de grandes volúmenes de datos para obtener un mejor rendimiento del sistema, siendo ésta una ventaja inherente a los SR semánticos [24]. Esto difiere de las técnicas basadas en matrices o en algoritmos de *machine learning*, para las cuales su rendimiento depende del tamaño de los datos de entrada [34].

En concreto, el uso de ontologías se fundamenta en este trabajo por las siguientes razones:

1. Aportan información aseverada e inferida a partir de vínculos lógicos entre usuarios, competencias y contenidos de VoD; lo cual puede ser aprovechado en la estrategia de recomendación.
2. Posibilitan la creación de nuevas aplicaciones del campo de los *Datos Abiertos Enlazados* (DAE) que exploten los resultados aquí obtenidos.

El uso de ontologías en la recomendación ha requerido el uso de técnicas de la web semántica en la definición de una estrategia de recomendación apropiada. Como era de esperar, las técnicas tradicionalmente usadas en el enfoque consciente del contexto (matrices y clasificación a través de algoritmos) no pudieron ser utilizadas en la presente investigación ya que están diseñadas para realizar tareas de predicción o cómputo de ponderaciones (“*ranking*”) a partir de conjuntos de valoraciones de los usuarios por los ítems. En cambio, se ha propuesto una estrategia de recomendación

caracterizada por el cálculo de la relevancia semántica entre las competencias en las cuales los estudiantes tienen necesidades educativas y otras competencias formalizadas por las ontologías, siendo esta la base para el descubrimiento y ponderación de los contenidos de VoD. A este efecto, se han incorporado las características del contexto de las competencias y las características de las ontologías en el cálculo de la relevancia semántica. En este sentido, la idea de ponderar los vínculos entre competencias para calcular la relevancia de un usuario por un contenido indica que el SR propuesto puede ser también clasificado dentro del enfoque CB.

En el ámbito de la web semántica, los vínculos formalizados entre instancias de una base de conocimiento son conocidos como *asociaciones semánticas complejas*. Éstas son esencialmente secuencias de propiedades que enlazan dos entidades en una consulta, por ejemplo, en el contexto de una búsqueda semántica. Una de las principales ventajas de las asociaciones semánticas es que, al ser descubiertas en una *base de conocimiento*, permiten inferir conocimiento a partir de los vínculos lógicos establecidos entre las entidades. De acuerdo con el proyecto *SemDis* (“*Semantic Discovery*”) [62], las *asociaciones semánticas complejas* pueden ser filtradas con el propósito de identificar las más relevantes a partir de un proceso de *ponderación*. Precisamente, esta posibilidad representa una oportunidad para inferir conocimiento útil a partir de los vínculos entre competencias, asegurando el carácter personalizado a partir de un proceso de *ponderación* delineado por el contexto en torno a las competencias. Más información sobre las asociaciones semánticas complejas, incluyendo sus principios, modelo, tipos y ejemplos aplicados, pueden hallarse en el Anexo A (Sección A.3).

2.1.7 Modelos computacionales lingüísticos

Hay situaciones en las cuales no es apropiado el uso de una evaluación cuantitativa, por ejemplo, en las cuales los expertos deben establecer un valor de percepción de acuerdo al rendimiento de algo o alguien [6]. Según Fung [27], en los procesos de evaluación de aspectos educativos o empresariales (por ejemplo, para tomar decisiones), los expertos se exponen a una serie de problemas ocasionados por la asignación de juicios de valor numéricos (por ejemplo, al usar escalas de Likert), tales como la incertidumbre, restricciones, y el vago conocimiento que pueda tener el experto. Incluso, es difícil para él asegurar que los criterios sean igualmente aplicados al evaluar todos los aspectos. En el caso de los entornos educativos, de acuerdo con Montero, Alias, Badía, Fonseca y Vicent [28], estos problemas se dan especialmente si la evaluación involucra habilidades y capacidades observables en la solución de problemas (por ejemplo, una mesa redonda, experimentos químicos, etc.) o si implica aspectos de naturaleza subjetiva, como el *trabajo en equipo*.

2. Base conceptual y estado del arte

En estos casos, la utilización de un enfoque lingüístico puede ser más conveniente, en el cual, son usados un conjunto de términos como los mostrados en la expresión (2.1). De acuerdo con el criterio de expertos que participaron en esta investigación, éste es el conjunto más apropiado en la práctica educativa en cuanto a términos y granularidad (esto es, la cardinalidad del conjunto de términos).

Por otro lado, en este trabajo son usadas las funciones de pertenencia triangulares (Figura 2.2) porque son las más utilizadas en la literatura [63], [64] [65]; además de que en los trabajos revisados no fue posible hallar evidencia de que otras funciones aportan más precisión en los resultados de los sistemas difusos.

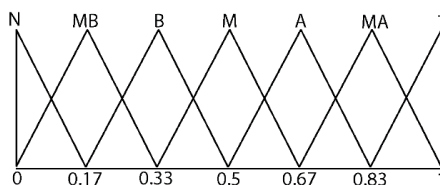


Figura 2.2 Un grupo de siete términos lingüísticos y su semántica. Fuente: [63].

De acuerdo con Herrera y Martínez [66], la semántica de los siete términos de la expresión (2.1) se define de la siguiente forma:

$$N = (0,0,0.17), \quad MB = (0,0.17,0.33), \quad B = (0.17,0.33,0.5), \quad M = (0.33,0.5,0.67), \quad A = (0.5,0.67,0.83), \quad MA = (0.67,0.83,1), \quad T = (0.83,1,1).$$

Dentro de los modelos computacionales lingüísticos se encuentran modelos como el de *2-tuplas* y el *basado en el principio de extensión*, los cuales son tratados en las siguientes secciones.

2.1.7.1 Modelo de 2-tuplas

El modelo de *2-tuplas* [63] está basado en el concepto de translación simbólica, la cual se define como sigue: sea $S = \{s_0, \dots, s_g\}$ un conjunto de términos lingüísticos, si un método simbólico agregando información lingüística obtiene un valor $\beta \in [0, g]$, y $\beta \notin \{0, \dots, g\}$ entonces una función de agregación ($app_2(\cdot)$) es usada para expresar el índice del resultado en S . La translación simbólica de un término lingüístico s_i es un valor numérico evaluado en $[-0.5, 0.5)$ que apoya la *diferencia de información* entre una cantidad de información $\beta \in [0, g]$ obtenida después de una operación de agregación simbólica y el valor más cercano en $\{0, \dots, g\}$ que indica el índice del término lingüístico más cercano en $S (i = round(\beta))$. Este enfoque representa la información lingüística por medio de 2-tuplas de la forma (s_i, α_i) , donde $s_i \in S$ y $\alpha_i \in [-0.5, 0.5)$, siendo s_i una representación de la etiqueta lingüística centro de la información y α_i un valor numérico que expresa el valor de la translación desde el resultado original β a la etiqueta del índice más cercana i en el conjunto de términos

lingüísticos s_i , esto es, la translación simbólica. Además, se define un conjunto de funciones de transformación entre términos lingüísticos y 2-tuplas, y entre valores numéricos y 2-tuplas. Las 2-tuplas que expresan la información equivalente a β es obtenida con las siguientes funciones:

$$\Delta: [o, g] \rightarrow Sx[-0.5, 0.5], \quad (2.2)$$

$$\Delta(\beta) = \begin{cases} s_i, & i = \text{round}(\beta) \\ \alpha = \beta - i, & \alpha \in [-0.5, 0.5] \end{cases} \quad (2.3)$$

donde $\text{round}(\cdot)$ es la operación de redondeo común, s_i tiene la etiqueta más cercana a β y α es el valor de la translación simbólica. Como un comentario importante, la conversión de un término lingüístico a una 2-tupla consiste en añadir un valor cero de la siguiente forma:

$$s_i \in S \Rightarrow (s_i, 0), \quad (2.4)$$

El modelo de *2-tuplas* fue generado para ser usado en contextos en los que se requiera *agregar valores lingüísticos* (equivalente a promediar valores numéricos) después de un proceso de evaluación llevado a cabo por expertos en el cual los valores de percepción son expresados en lenguaje natural. Por tal razón, este enfoque es ideal para ser aplicado en la evaluación de competencias educativas, considerando que los *Niveles de Competencia* (NC) son valores lingüísticos y que se busca realizar un proceso de fusión teniendo en cuenta la estructura de los mapas (es un proceso equivalente al de promediar valores numéricos). El modelo de *2-tuplas* es generalmente usado para abordar el problema de la toma lingüística de decisiones.

En el Anexo A (Sección A.2) se muestra un ejemplo de aplicación del modelo *2-tuplas* para la toma de decisiones (Ejemplo A.1) en un caso en el que una compañía necesita renovar su sistema operativo. Para esto, contrata una consultora que realiza una revisión de las diferentes posibilidades (UNIX, WINDOWS, AS/400 y VMS). Después de aplicar el modelo *2-tuplas*, se puede ver que éste permite identificar la opción más conveniente de entre varias posibilidades (UNIX). Además, se evidencia la facilidad con la que se aplica el modelo, lo cual supone un bajo consumo de recursos computacionales.

2.1.7.2 Modelo basado en el principio de extensión

El *Modelo Basado en el Principio de Extensión* (MBPE) [67] ha sido introducido para generalizar operaciones matemáticas numéricas a los conjuntos difusos. Para esto, se usan operadores de agregación lingüísticos, los cuales actúan de acuerdo con:

$$S^n \xrightarrow{\bar{F}} F(\mathcal{R}) \xrightarrow{\text{app}_1(\cdot)} S, \quad (2.5)$$

2. Base conceptual y estado del arte

donde S^n simboliza el producto cartesiano n de S , \tilde{F} es un operador de agregación basado en el principio de extensión, $F(\mathcal{R})$ es el conjunto de conjuntos difusos sobre el conjunto de números reales \mathcal{R} , $app_1: F(\mathcal{R}) \rightarrow S$ es una función de aproximación lingüística que retorna una etiqueta del conjunto de términos lingüísticos S cuyo significado es el más cercano al número difuso no etiquetado obtenido y S es el conjunto de términos inicial. Al igual que el enfoque de *2-tuplas*, el basado en el principio de extensión también es usado para agregar valores lingüísticos provenientes de la evaluación de expertos.

En el Anexo A (Sección A.2) se muestra el comportamiento de MBPE en el mismo caso en el que una compañía debe seleccionar un sistema de cómputo de entre varias posibilidades (Ejemplo A.2). En él se puede verificar que a diferencia del modelo *2-tuplas*:

1. El MBPE decretó un empate entre UNIX, WINDOW y VMS, lo cual sugiere que *2-tuplas* es un modelo más preciso.
2. MBPE implica una mayor complejidad de aplicación que el modelo *2-tuplas*, lo cual sugiere que su consumo de memoria y tiempo de ejecución también es mayor.

Estas conclusiones permitieron seleccionar al enfoque de *2-tuplas* como la base para la construcción del método computacional para la evaluación de competencias propuesto en el Capítulo 5.

2.2 Estado del arte

Tal como se introdujo en el Capítulo 1, la solución planteada en la presente investigación es un SR de contenidos educativos de VoD centrado en las necesidades en torno a las competencias educativas. Con el propósito de establecer los antecedentes del trabajo se realizó un estudio de la literatura que, como es de esperar, se enfocó inicialmente en los SR. Dado que la solución representada en el Capítulo 1 (Figura 1.2) involucra otros aspectos como los *esquemas de metadatos*, las *ontologías*, la *computación lingüística* y *estrategias de recomendación* basadas en técnicas de la web semántica, en primera instancia se hizo una revisión en búsqueda de antecedentes que los considerara a todos al tiempo. Sin embargo, en los trabajos de la literatura revisados no fue posible hallar información al respecto. En cambio, se pudo constatar que existen trabajos sobre los siguientes temas: *SR educativos*, *competencias*, *mapas de competencias*, *esquemas de metadatos del ámbito de la educación*, *esquemas de metadatos para contenidos multimedia*, *ontologías para recursos multimedia y del ámbito educativo*, *modelos difusos para la evaluación de competencias*, y *técnicas de la web semántica basadas en asociaciones semánticas*

para la inferencia de conocimiento. Así las cosas, la revisión debió centrarse en el estudio individual de los antecedentes de todos los aspectos mencionados, con lo cual se generó un estado del arte que identifica los aportes de la literatura a la presente investigación y las brechas que debieron ser abordadas.

El estado del arte trata los siguientes aspectos:

Inicialmente, se presenta una revisión de los trabajos en los que se proponen *SRs del ámbito de la educación*, señalando un grupo de brechas relacionadas con: 1) la no consideración de las competencias y los *Niveles de Competencia* (NC) en los procesos de recomendación; y 2) el uso de las ontologías como fuentes de datos y no como estructuras a partir de las cuales se puede descubrir conocimiento acerca de las necesidades educativas de los estudiantes. Precisamente, estas brechas se abordan en la estrategia de recomendación semántica propuesta en el Capítulo 6.

Segundo, se presenta un análisis de los *esquemas de metadatos del ámbito de las competencias* (incluyendo los *mapas de competencias*), a partir del cual identifica la información que éstos describen, sus fortalezas y debilidades. Esto con el fin tomar algunas decisiones básicas para la definición del esquema de metadatos para competencias que se propone en el Capítulo 3.

Tercero, se retoman algunos trabajos hallados en la literatura enfocados en el análisis de los *esquemas de metadatos para la descripción de contenidos multimedia*. Esto se hace con el propósito de establecer los principios para la definición del esquema de metadatos para de contenidos de VoD presentado en el Capítulo 3.

Cuarto, de acuerdo con la metodología acogida para el diseño de ontologías [68], se hace un análisis de las ontologías existentes relacionadas con las *competencias*, los *Objetos de Aprendizaje* (OA) y los *contenidos de video*; con el fin de establecer los principios elementales para la construcción de la ontología que se propone en el Capítulo 4.

Quinto, se presenta una revisión de la literatura acerca de los trabajos relacionados con los *modelos difusos usados en la evaluación lingüística de competencias*, a partir del cual se identificaron las brechas existentes con respecto a las características del presente trabajo y los aportes del método computacional para la evaluación de competencias propuesto en el Capítulo 5.

Para la construcción del estado del arte, se tomó como referencia lo definido por Kitchenham, Pretorius y Budgen [69], donde se establece un grupo de directrices para la revisión sistemática de la literatura. Siguiendo éstas, entre los criterios de búsqueda utilizados más importantes se encuentran:

- Competency metadata
- Video ontology

2. Base conceptual y estado del arte

- Video metadata
- Competency map metadata
- Competency assessment
- Computing with words
- Competency ontology
- Learning object metadata
- Learning object ontology
- Educational recommender system
- Learning recommender system
- Ontology-based recommender system

2.2.1 Sistemas de recomendaciones educativos

Hoy en día, la era digital provee herramientas que facilitan la creación y publicación de una gran cantidad de recursos en televisión e internet. Éstos pueden ser usados por docentes, instructores y estudiantes para apoyar sus procesos educativos. Con el propósito de ofrecer mecanismos efectivos para la búsqueda y recuperación de información útil y apropiada de entre una extensa variedad de opciones, los SR educativos han atraído el esfuerzo investigativo en los últimos años.

Al respecto, se realizó una revisión exhaustiva de la literatura con base en las directrices para la revisión sistemática propuestas en [69]. Inicialmente, se hizo una búsqueda centrada en las expresiones: “*learning recommender system*” y “*educational recommender system*”; en bases bibliográficas como *IEEE Xplore Digital Library*, *Science Direct*, *ACM Digital Library* y *Google Scholar*. Como resultado, 48 artículos fueron encontrados en un análisis inicial, de los cuales 18 fueron considerados relevantes. Después de un segundo análisis, los artículos fueron clasificados de acuerdo a tres criterios: 1) el enfoque utilizado; 2) el tipo de ítem recomendado; y 3) el tipo de datos de entrada. Los resultados se presentan en la Tabla 2.1.

La revisión demuestra que la mayoría de trabajos adoptan los enfoques *colaborativo* (CF) y *consciente del contexto*, sugiriendo recursos de aprendizaje y utilizando intereses educativos expresados explícitamente.

Los autores de los trabajos enfatizan en que hay una necesidad de ayudar a los estudiantes a cubrir sus necesidades educativas ofreciendo recursos de forma personalizada. Sin embargo, en los trabajos revisados no se encontró evidencia de la incorporación de competencias y mapas de competencias de forma complementaria en los procesos de recomendación. Entonces, con base en la necesidad de proveer mecanismos para apoyar entornos educativos formales, el presente trabajo incluye las características de competencias y mapas para definir un modelo funcional y matemático para construir perfiles de usuario, ejecutar procesos de filtrado y obtener recomendaciones.

Por otra parte, cuatro de los trabajos relevantes presentan SR basados en ontologías [11], [8], [21], [22]. En ellos, las ontologías son usadas como fuentes de datos

proveedoras de conocimiento ligado a la educación. Al respecto, de acuerdo con Vesin [22], éstas facilitan el intercambio, construcción y reutilización de conocimiento. Por lo tanto, los SR basados en ontologías son apropiados para ambientes de DAE. En la presente investigación, las ontologías no sólo son usadas como fuentes de datos, sino también como herramientas para descubrir: 1) conocimiento acerca del aprendizaje del estudiante, provisto por estructuras denominadas *perfiles-ontología* [34]; y 2) *asociaciones semánticas* entre competencias y contenidos de VoD. Esto, orquestado por una estrategia de recomendación propuesta en el Capítulo 6.

		Trabajos	
Criterios	Enfoque utilizado	Colaborativo (CF)	[3], [8], [9], [10], [70], [18]
		Basado en contenido (CB)	[19], [20] [71]
		Híbrido (CF-CB)	[14], [17]
		Consciente del contexto	[11], [12], [13], [21], [16], [15], [22]
	Ítem recomendado	Recursos de aprendizaje (OA, enlaces web, lecciones y cursos)	[11], [14], [8], [15], [9], [12], [20], [13], [21], [10], [70], [22], [71]
		Personas con intereses similares	[3]
		Estrategias de apoyo a la enseñanza y el aprendizaje	[16], [15], [17], [18]
	Tipo de datos de entrada	Estilos de aprendizaje	[8], [9], [13], [70]
		Calificaciones en test	[10], [15]
		Niveles de competencia	[20], [16]
		Patrones de navegación en la web	[22]
		Actividad en foros	[18]
		Intereses educativos expresados explícitamente	[11], [14], [3], [19], [12], [21], [17], [71]

Tabla 2.1. Clasificación de trabajos sobre sistemas de recomendaciones educativos.
Fuente: propia.

2.2.2 Esquemas de metadatos del ámbito de las competencias

Debido a que los procesos de aprendizaje están cada vez más ligados a plataformas tecnológicas, por ejemplo, para facilitar el acceso a comunicaciones entre estudiantes y profesores y asignar recursos de aprendizaje adaptados a las necesidades, se han venido desarrollando a través de los años diversos esquemas de metadatos relacionados con las competencias. Dado que uno de los propósitos del presente

2. Base conceptual y estado del arte

trabajo es definir un esquema de metadatos para la descripción de contenidos de VoD de acuerdo con las características de las competencias, se realizó un estudio de la literatura para conocer y estudiar los antecedentes. En la presente sección se presentan los resultados de tal estudio. Una versión extendida con información detallada puede ser consultada en el Anexo B.

Inicialmente se extrajo de la literatura un grupo de esquemas de metadatos del ámbito de la educación. Como resultado se identificaron esquemas para *Objetos de Aprendizaje (OA)* tales como “*Learning Object Metadata*” (LOM) [56]; esquemas para competencias tales como “*eCOTOOL Competence Model*” (eCOTOOL) [72], “*MedBiquitous Competency Object Specification*” (MCOS) [50], “*Reusable Definition of Competency or Educational Objective*” (RDCEO) [52], “*IEEE Reusable Competency Definition*” (IEEE RCD) [37] y “*Learning Outcome Definition*” (LOD) [73]; y esquemas para mapas de competencias tales como “*Simple Reusable Competency Map*” (SRCM) [38] y “*MedBiquitous Competency Framework Specification*” (MCFS) [74]. Otro esquema para objetos de aprendizaje es *Objetos de Aprendizaje Abiertos (OBAA)* desarrollado por la *Universidade Federal do Rio Grande do Sul* de Brasil [75]. Este esquema no fue considerado en la presente investigación, dadas las semejanzas con LOM en cuanto al significado de sus metadatos. Además de que LOM es un estándar aceptado a nivel mundial. Sin embargo, tales semejanzas suponen cierta ventaja para adaptar los resultados aquí expuestos a OBAA y a otros esquemas basados en LOM como el perfil de aplicación de LOM para España LOM-ES [76].

En la primera fase del análisis se buscaron relaciones entre los esquemas (Figura 2.3), tratando de identificar similitudes y equivalencias en los elementos de descripción, versiones simplificadas entre esquemas (perfiles de aplicación), vínculos entre las entidades involucradas (OA, competencias y mapas). A partir de este análisis se estableció que el esquema IEEE RCD permite definir un marco de referencia para vincular OAs, competencias y mapas de competencias; lo cual es pertinente teniendo en cuenta que estos vínculos pueden aportar información relevante en el proceso de recomendación. Por ejemplo, en una visión simple de una estrategia de recomendación, conviene identificar los contenidos de VoD educativos asociados a competencias en las que los estudiantes tienen necesidades educativas.

En la segunda fase del análisis se hizo una comparación de los *esquemas de metadatos para competencias* con base en la información que describen. A partir de esta, se estableció una taxonomía de fortalezas y debilidades relacionadas con el uso de lenguaje natural, la posibilidad de crear extensiones, entre otras. Como resultado se halló que IEEE RCD es el esquema con más fortalezas y menos debilidades.

En resumen, éste es un estándar aceptado, lo cual es apropiado para la proliferación, intercambio y reutilización de descripciones. Además, puede ser extendido sin afectar su estructura y cuenta con SRCM para describir relaciones entre competencias a través de mapas de competencias, manteniendo la dependencia entre entidades y facilitando así la reutilización de descripciones. Sin embargo, IEEE RCD depende del lenguaje natural, lo cual puede dificultar la construcción de conceptos comunes debido a la variabilidad sintáctica y léxica entre las personas. Al respecto, eCOTOOL, MCOS y LOD dan un paso adelante en la definición de elementos específicos (por ejemplo, un verbo de acción, categoría KSC y niveles de cualificación) como una alternativa al lenguaje natural.

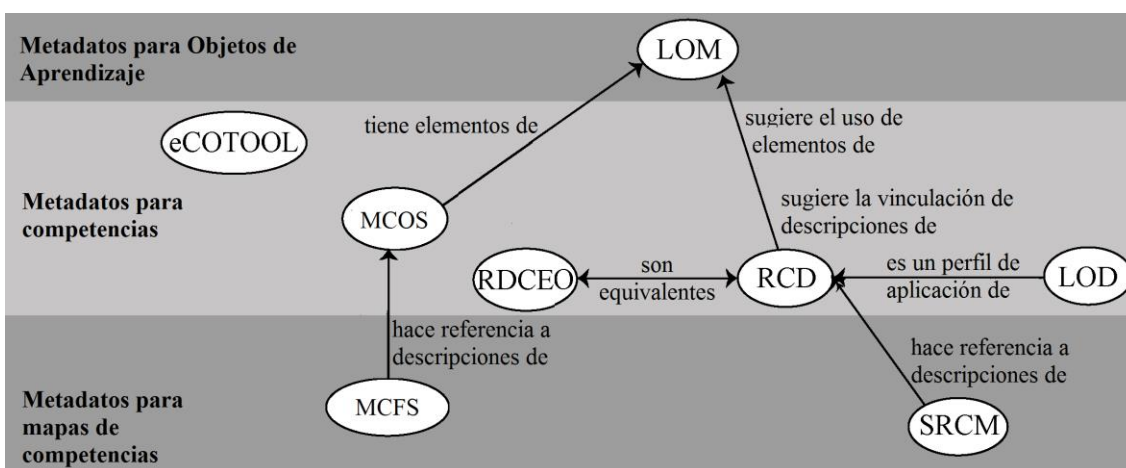


Figura 2.3 Esquemas de metadatos relacionados con competencias. Fuente: propia.

Considerando estos resultados, el presente trabajo provee como contribución una *extensión de IEEE RCD*, la cual suma las fortalezas de los esquemas estudiados y aborda sus dificultades añadiendo nuevos elementos de datos, vocabularios y tipos de datos para la descripción de competencias, algunos definidos como resultado de una revisión de literatura asociada y otros a partir de los resultados de un proceso de evaluación llevado a cabo por expertos y usuarios con base en características de la CM. La extensión del IEEE RCD se propone en el Capítulo 3 (Sección 3.2).

Por otra parte, considerando la posibilidad de crear relaciones entre competencias descritas por IEEE RCD, se adopta el SRCM como el esquema para el modelado de mapas de competencias. Al respecto, SRCM es el esquema más expresivo hallado en la literatura.

2.2.3 Esquemas de metadatos para contenidos multimedia

En servicios de VoD como Youtube, cada vez es más común ver que los usuarios ejecutan prácticas de producción más elaboradas para generar sus contenidos. En

2. Base conceptual y estado del arte

este sentido, hay definidos metadatos en diferentes estándares tales como *MPEG-7*, *TV-Anytime*, *PBCore*, *SMPTE*, especificaciones como *P-META*, y perfiles de aplicación como *EBUCore*.

El propósito aquí es seleccionar uno de estos esquemas de metadatos, de tal forma que pueda ser utilizado para describir los contenidos educativos desde el punto de vista del servicio de VoD. Al respecto, Vargas, Baldassarri y Arciniegas [77] realizaron un análisis comparativo detallado de los estándares ya mencionados con miras a la descripción de contenidos de IPTV desde el punto de vista educativo, por lo que conviene retomar sus conclusiones:

1. *TV-Anytime*, *MPEG-7* y *P-META* facilitan la segmentación de contenidos. *TV-Anytime* en tiempo, *MPEG-7* en tiempo y espacio, y *P-META* en tiempo.
2. A partir de *MPEG-7*, se han construido diferentes perfiles de aplicación como el “*AudioVisual Description Profile*” (AVDP), “*Simple Metadata Profile*” (SMP), “*Core Description Profile*” (CDP) y *ProtoSpace*. De la misma forma, con base en *P-META* se ha construido “*EBU Music Reporting Metadata*” (MRM). A partir de *EBUCore*, se generaron los perfiles “*EBU Audio Model*” (ADM), *EgtaMETA*, y “*EUScren Metadata Schema*”. De los otros esquemas, no se han encontrado perfiles de aplicación en la literatura.
3. *TV-Anytime* y *MPEG-7* son esquemas que pueden ser usados de forma precisa para la descripción de contenidos de VoD. El primero, se especializa en experiencias televisivas, mientras que el segundo es aplicable a cualquier tipo de contenido multimedia en general.

Después del análisis anterior, *TV-Anytime* y *MPEG-7* son los esquemas más apropiados para ser usados en el presente trabajo. Con el fin de seleccionar uno de los dos, se han analizado otros aspectos que se resumen en la Tabla 2.2. Específicamente, conviene que el esquema sea fácil de aplicar y que posea elementos educativos. De acuerdo con el análisis, éstas características son propias de *TV-Anytime*, por lo cual ha sido elegido para ser usado en el presente trabajo. A partir de tal esquema, la presente investigación contribuye al conocimiento proponiendo un perfil de aplicación que selecciona y define un grupo de elementos y tipos de datos para describir los contenidos desde el punto de vista del servicio de VoD y vincularlos con las competencias. El perfil de aplicación se propone en el Capítulo 3 (Sección 3.1).

2.2.4 Ontologías existentes

La reutilización de ontologías existentes es un aspecto importante dentro del proceso de desarrollo de una nueva ontología. Según Noy y MacGuinness [68] la reutilización de ontologías se define como “el proceso en el cual el conocimiento ontológico

existente es utilizado como una entrada para generar nuevas ontologías”. Esto tiene beneficios, tanto para ingenieros como usuarios, como el de posibilitar el despliegue de modelos existentes, reduciendo costos de implementación, mejorando la calidad de las fuentes ontológicas, las cuales son, gracias a la reutilización, sujeto de continuas revisiones y refinamientos; e incrementando la interoperabilidad entre aplicaciones basadas en ontologías.

Característica	TV-Anytime	MPEG-7
Complejidad de la estructura	Sencilla	Compleja
Complejidad de la sintaxis	Sencilla	Compleja
Segmentación	Sencilla	Compleja
Esquema de metadatos modular	Si	Si
Posee elementos de datos educativos	Si	No

Tabla 2.2. Comparación entre *TV-Anytime* y *MPEG-7*. Fuente: propia.

Con el propósito de llevar a cabo un estudio de las ontologías existentes, se siguió la metodología definida [78] para la *reutilización de ontologías*. Específicamente, se desarrollaron dos fases: *descubrimiento de ontologías* y *evaluación de ontologías existentes*. A continuación, se realiza una breve descripción del estudio, haciendo énfasis en los resultados alcanzados. Una versión extendida se puede encontrar en el Anexo D.

En la primera fase, se definió un grupo de *palabras clave* para realizar la búsqueda en buscadores convencionales y repositorios de ontologías. Algunas son: “*Competency*”, “*Competency map*”, “*video on demand*”, entre otras. Como resultado de la búsqueda se encontraron seis ontologías que abarcan: la formalización del esquema de metadatos LOM, el modelado de recursos multimedia y competencias. En lo revisado, ontologías relacionadas con los mapas de competencias no fueron encontradas.

En la segunda fase, las ontologías fueron evaluadas de acuerdo a cinco criterios [78]: *contenido*, *representación de conocimiento*, *técnica*, *aplicación*, y *disponibilidad*. De acuerdo con los resultados, se estableció una taxonomía de fortalezas y debilidades relacionadas con: 1) la consistencia identificada por razonadores (*Hermit*, *Pellet* y *jcel*); 2) soporte de APIS Java de Apache Jena (centradas en web semántica); 3) consistencia con esquemas de metadatos reconocidos (LOM, *TV-Anytime*); entre otros.

Como resultado del estudio se concluyó que existen ontologías útiles para el desarrollo del presente trabajo, que deben pasar por un proceso de adaptación e integración para proveer las condiciones que imponen los esquemas de metadatos expuestos en las

2. Base conceptual y estado del arte

anteriores secciones. A continuación, se listan las ontologías que son reutilizadas y la respectiva justificación de forma resumida:

- “*Learning Object Metadata Ontology*”: modela todos los componentes de LOM. Es de fácil comprensión, lo cual facilita su integración con otras ontologías y la inclusión de nuevas clases y propiedades. Ésta ontología permite proveer a los contenidos de VoD las características propias de los OA.
- “*Ontology for Media Resource*”: ontología que permite modelar los contenidos de VoD, adaptando los conceptos representados a los elementos de *TV-Anytime*. En su documentación se encuentra explícitamente cómo realizar el mapeo respectivo usando propiedades OWL. Además, ofrece una gran cantidad de comentarios que facilitan su comprensión.
- “*Learning Object Context Ontologies*” (“*The LOCO framework*”): ofrece un marco modular de ontologías pequeñas y fácilmente comprensibles en torno a los OA. De entre el grupo de ontologías que define, hay dos que serán reutilizadas:
 - “*Competency Model Ontology*”: ontología útil para modelar las competencias.
 - “*Learning Context Ontology*”: ontología que integra las competencias con otros aspectos del contexto de los OA, por lo cual puede ser útil para asociar otras ontologías.

Tal como se adelantó en el capítulo introductorio (Sección 1.1), en el presente trabajo se provee una ontología para la formalización de relaciones entre competencias, contenidos de VoD y mapas de competencias. Para esto, las ontologías seleccionadas son integradas y usadas para mapear todos los elementos y relaciones que se definen en el esquema de metadatos propuesto para *describir contenidos de VoD y competencias*. Tal como se mencionó anteriormente, en los trabajos analizados como parte de la revisión bibliográfica no fue posible hallar una ontología para modelar mapas de competencias. Entonces, fue necesario diseñar una nueva. La descripción completa de la ontología, la cual es el producto de la integración de ontologías para contenidos de VoD, competencias y mapas de competencias, se presenta en el Capítulo 4.

2.2.5 Modelos computacionales lingüísticos para la evaluación de competencias

De acuerdo a lo que se planteó en la Base conceptual (Sección 2.1.7) existen situaciones educativas y empresariales (por ejemplo, la evaluación de competencias y la toma de decisiones) que requieren el desarrollo de procesos evaluativos lingüísticos. Así las cosas, la comunidad investigativa ha desarrollado modelos basados en

computación lingüística que involucran términos cualitativos, siendo éstos una alternativa al uso de valores numéricos.

De acuerdo con Montero et al., el enfoque difuso es usado porque permite una mejor representación del lenguaje natural [28]. Específicamente, en la literatura hay trabajos centrados en evaluar competencias usando enfoques lingüísticos difusos. Pengshun y Jianguo [79] presentan una metodología para evaluar competencias laborales de los estudiantes. Tal propuesta aplica un método de evaluación difuso basado en una estructura de competencias profesionales, la cual es una estructura en árbol de tres niveles con competencias pre-definidas.

Serrano, Romero y Olivas [80] proponen un enfoque difuso para recomendar actividades educativas. Inicialmente, define un modelo basado en *2-tuplas* que posibilita la evaluación lingüística de competencias sobre una estructura compuesta por un grupo de competencias y actividades. Según la estructura, cada competencia está relacionada con un grupo de actividades. Después, se define un modelo de recomendación que usa un filtro basado en los resultados de la evaluación. Zhaojun, Xiao y Xiaoqing [29] presentan un modelo para evaluar competencias educativas basado en un enfoque difuso. En un caso de estudio, el modelo es aplicado a un sistema de índices de competencias para vendedores, el cual es una estructura en árbol compuesta por 3 niveles. Jia y Zhao [81] proponen un método de evaluación de competencias básicas para empresas. Éste se basa en la entropía y los números difusos triangulares, posibilitando una evaluación objetiva de empresas. Éste puede ser útil en un proceso de decisión, en el cual un grupo de empresas son comparadas entre sí de acuerdo al criterio de expertos. Zhang [82] propone un método de decisión basado en operadores de agregación. Éste usa el modelo de *2-tuplas* para agregar los desempeños de las alternativas medidas por un grupo de evaluadores. El resultado de la agregación es usado para tomar una decisión de entre el grupo de alternativas.

Todos los trabajos relacionados revelan la importancia de la evaluación de competencias por medio de términos lingüísticos. Para este propósito, varios modelos y enfoques han sido propuestos para facilitar el proceso de evaluación con un alto nivel de efectividad. Entre ellos se destaca el de *2-tuplas* difusas. Sin embargo, en la literatura revisada no se halló evidencia de investigaciones basadas en la estructura y características de los mapas de competencias. Tampoco fue posible encontrar un modelo formal para definir las estructuras de competencias compuestas por niveles, como por ejemplo los grafos. Generalmente se basan en vectores, definiendo algoritmos adaptados a éstos. Adicionalmente, aunque todas las propuestas son efectivas, no ha sido realizada una evaluación de su precisión.

2. Base conceptual y estado del arte

En el presente trabajo se define un método computacional basado en *2-tuplas* para la evaluación de competencias, el cual es consciente de la estructura de los mapas de competencias definida en el esquema de metadatos SRCM. El enfoque de *2-tuplas* ha sido seleccionado porque, de acuerdo con los resultados de los ejemplos A.1 y A.2 mostrados en el Anexo A (Sección A.2), tiene un modelo matemático preciso y fácilmente aplicable. Con respecto al uso del SRCM, el cual fue tratado en la Sección 2.2.2, éste es el más expresivo. Entre sus ventajas está que es lo suficientemente explícito como para incorporar sus características (nodos y aristas etiquetadas) al método computacional. Las estructuras de los mapas se han definido formalmente como grafos, por lo cual, en el presente trabajo se ha creado un algoritmo que fusiona: 1) el algoritmo de *Búsqueda de primero en profundidad* para recorrer los grafos; y 2) las características del enfoque de *2-tuplas* difusas, incluyendo la representación y organización de la información, la creación de perfiles de usuario y el proceso de agregación de términos lingüísticos. El método computacional se presenta en detalle en el Capítulo 5 y es fundamental para crear perfiles de usuario propicios para el SR propuesto.

Capítulo 3

Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

Las plataformas de VoD ofrecen circunstancias favorables para realizar actividades de enseñanza y aprendizaje [44]. Por tal razón, cada vez es más popular la producción de videos en torno a objetivos educativos.

Aunque generar contenidos de video de calidad es uno de los propósitos más importantes para los creadores, si éstos no pueden ser recuperados por las personas especialmente interesadas, el trabajo de producción podría no tener el efecto deseado. En este sentido, los metadatos juegan un papel importante porque permiten organizar y recuperar los contenidos, ya sea en labores administrativas o en servicios de búsqueda y recomendación. De ahí que, conseguir descripciones adecuadas y precisas de los contenidos de VoD de acuerdo a las competencias tiene una importancia vital en el desarrollo de este trabajo.

Sumado a esto, está la relevancia de los esquemas de metadatos en la construcción de ontologías. En la literatura ligada a la web semántica, en áreas específicas como los *Datos Abiertos Enlazados* (DAE), es común encontrar ontologías cuya construcción se basa en esquemas de metadatos. Por ejemplo, “*Dublin Core Ontology*” se basa en el esquema “*Dublin Core*” [83]; “*Learning Object Metadata Ontology*” se basa en el esquema “*Learning Object Metadata*” [57]; y “*Ontology for Media Resources*” se basa en esquemas como *MPEG-7* y *TV-Anytime* [58]. Ésta práctica supone ventajas como: 1) la facilidad en el diseño de las ontologías, teniendo en cuenta que iniciar el diseño de una desde cero puede ser una tarea compleja [35]; 2) se añade semántica a los metadatos [84]. Siguiendo este enfoque, la construcción de un *esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas* es crucial para definir posteriormente una ontología que permita ejecutar estrategias

de recomendación basadas en técnicas de la web semántica. Tal esquema se presenta en este capítulo, haciendo énfasis en el proceso llevado a cabo para obtenerlo.

Como primera instancia, es importante mencionar que el desarrollo del esquema de metadatos se basó en los resultados obtenidos en el análisis de la base conceptual y el estado del arte presentado en el Capítulo 2, los cuales se resumen a continuación:

- Existe una tendencia a generar un “*framework*” de esquemas en el que las descripciones de los *Objetos de Aprendizaje* (OA), las competencias y los mapas de competencias se generan individualmente para después vincularlas por metadatos que hacen referencia a sus identificadores [38].
- *TV-Anytime* se ha elegido como el esquema de referencia para formalizar las descripciones de los contenidos desde el punto de vista del servicio de VoD.
- El esquema “*IEEE Reusable Competency Map*” (IEEE RCD) cuenta con más ventajas frente a otros esquemas especializados en la descripción de competencias (ver Sección 2.2.2). Sin embargo, éste no facilita la construcción de conceptos comunes debido a que no define atributos específicos. En cambio, depende del lenguaje natural, sobre el cual pesa la variabilidad léxica existente entre las personas. Al respecto, el esquema *eCOTOOL* da pasos hacia la definición de atributos y vocabularios específicos como una alternativa al uso del *lenguaje natural* y el *texto libre*.
- El esquema “*Simple Reusable Competency Map*” (SRCM) permite la formalización de mapas de competencias, definiendo nodos relacionados entre sí, los cuales a su vez se relacionan con descripciones de competencias construidas con el IEEE RCD.

A partir de éstos resultados, se tomaron las siguientes decisiones básicas para generar el *esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias*:

- El esquema de metadatos propuesto es un “*framework*” que vincula tres esquemas: uno para la descripción de contenidos desde el punto de vista del VoD; uno para competencias y otro para mapas de competencias. En la práctica, al indicar que una competencia es desarrollada en el video, y que en el currículo educativo la competencia está formalizada en un mapa, el esquema propuesto permitirá describir cada una de estas entidades de forma individual y proveerá los elementos necesarios para vincularlas a través de referencias a sus identificadores. Esta decisión va en concordancia con lo definido en [38], en donde se establecen sinergias entre SRCM e IEEE RCD para vincular competencias a los mapas. Esto permitirá que en el futuro se puedan aprovechar los repositorios de cada uno de los esquemas para después

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

vincularlos de manera natural. Además, permitirá que la información descrita pueda ser mapeada en la ontología sobre la cual se apoya la estrategia de recomendación. Lo anterior, teniendo en cuenta que en los ambientes de DAE los recursos, provenientes de diferentes fuentes de datos producto de la instanciación de ontologías, son enlazados por medio de sus URL; lo cual favorece el descubrimiento de conocimiento nuevo a partir de ontologías conocidas [85]. En síntesis, el modelo descriptivo provisto por el esquema que aquí se propone y las ontologías comparten un enfoque similar.

- Dado que *TV-Anytime* ofrece todas las posibilidades descriptivas, el esquema de metadatos para la descripción de los contenidos de VoD es un *perfil de aplicación de TV-Anytime* que incluye elementos de datos con significado y función dentro del VoD. Por ejemplo, si el contenido es tomado de un servicio de VoD fuente como Youtube, es importante tener en cuenta su URL.
- Como el IEEE RCD no proporciona elementos descriptivos específicos sobre los atributos de las competencias, el esquema de metadatos para la descripción de competencias es una extensión (información adicional a lo definido en un esquema) del IEEE RCD que incorpora: 1) nuevos elementos de datos, vocabularios y tipos de datos tomados de esquemas para OA, de los otros esquemas para competencias y de literatura asociada; 2) nuevos elementos considerados de interés para la comunidad de uso después de un proceso de evaluación enmarcado en la CM (Sección 3.4).
- Dado que en los trabajos de la literatura revisados no fue posible hallar esquemas con mayor expresividad para la descripción de mapas de competencias, se hace uso del SRCM tal como se define originalmente [38].

Todo el proceso llevado a cabo para la determinación del esquema de metadatos está representado en el diagrama de flujo de la Figura 3.1, el cual nace de una adaptación de la *Orientación sobre la elaboración de un esquema de metadatos* (Norma UNE-ISO 23081) [86] y está constituido por un grupo de fases que tienen como salida diferentes productos. Las fases A, B y C se ejecutan para determinar tanto el perfil de aplicación de *TV-Anytime* como la extensión del IEEE RCD; obteniendo como producto final el esquema de metadatos propuesto.

Cada fase del proceso se explica a continuación:

Fase A: determinar el alcance del esquema. Esta fase consiste en determinar la información que se busca describir con el esquema.

Fase B: Identificar. En esta fase se realizan tres subprocesos así: en la Fase B.1 (*Cotejar información a describir y el esquema original*) se seleccionan los elementos

del esquema original, ya sea *TV-Anytime* o *IEEE RCD*, que describen la *información que se busca describir* (*Elementos seleccionados del esquema original*). En la Fase B.2 (*Modificar definición de los elementos seleccionados*) se establece si la definición de los elementos seleccionados debe ser modificada o no (*elementos seleccionados con modificaciones*), en cuanto a su significado y función. En la Fase B.3 (*Determinar nuevos elementos a integrar*) se determinan los nuevos elementos que deben ser integrados tanto para definir la *información que se busca describir* que no está abarcada en el esquema original como para vincular las descripciones entre contenidos, competencias y mapas (*Nuevos elementos*). Esto último garantiza que el esquema resultante provea un modelo descriptivo integral.

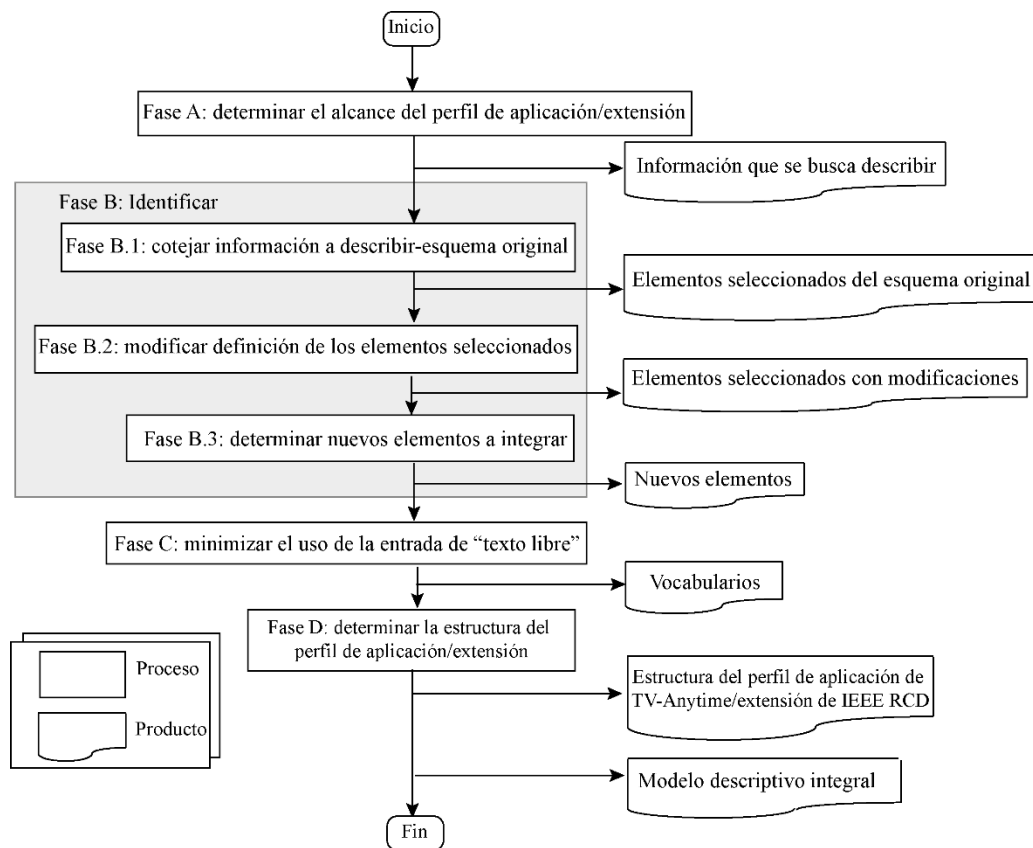


Figura 3.1. Proceso desarrollado para la generación del esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias

Fase C: minimizar el uso de la entrada de “texto libre”. En esta fase se busca identificar, en lo posible, vocabularios a través de los cuales se asignen valores predeterminados para los elementos en lugar del uso de “texto libre”. Según [86], el “texto libre” puede ocasionar problemas como la inutilización de la información por parte de sistemas automáticos, la variación ortográfica y el incremento de costos operacionales producidos por la recuperación ineficiente de la información.

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

Fase D: determinar la estructura del perfil de aplicación/extensión. Esta fase busca determinar la estructura del *perfil de aplicación de TV-Anytime* y de la *extensión del IEEE RCD*. Esto implica la integración de los nuevos elementos en la estructura del esquema original y la definición de todos los elementos en cuanto a su significado y función, tanto los tomados de *TV-Anytime* e *IEEE RCD* (redefinidos y no redefinidos) como los nuevos.

Como resultado de todo el desarrollo de las fases se obtiene el *equema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas*.

En cuanto a las contribuciones al conocimiento provistas en este capítulo se tiene:

1. Un perfil de aplicación para la descripción de contenidos de VoD generado a partir de información justificadamente seleccionada del esquema de metadatos *TV-Anytime*.
2. Una extensión del *IEEE RCD* que integra nueva información relacionada con el modelado de atributos de las competencias.
3. Un esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias, producto de la integración tanto del *perfil de aplicación de TV-Anytime* como la *extensión del IEEE RCD*.

Con respecto al *IEEE RCD*, se realizó una evaluación enmarcada en la Calidad de los Metadatos (CM) [55] con el propósito de validar la nueva información incluida y realizar mejoras sobre una versión previa del esquema (Anexo S).

3.1 Perfil aplicación de TV-Anytime para la descripción de contenidos de VoD

TV-Anytime ofrece en su especificación [36] un esquema de metadatos que define funcional y estructuralmente elementos y tipos de datos útiles para describir una gran variedad de aspectos relacionados con los servicios audiovisuales. Por ejemplo, puede ser usado para definir información descriptiva de los contenidos de video necesaria para desplegar una *Guía Electrónica de Programación (GEP)*. También trata aspectos relacionados con: el tipo de servicio ("*broadcast*" y VoD); la producción (contenidos provistos por plataformas web, productoras de televisión o distribuidoras de contenidos); las colecciones de contenidos (series, programas relacionados, información complementaria); entre otros. Aunque toda esta información está organizada en las especificaciones de *TV-Anytime*, su uso puede resultar confuso. Además, que mucha de esa información puede ser irrelevante en servicios puntuales (por ejemplo, los horarios de transmisión son más importantes en el *broadcast* que en

el VoD). Por tal razón, se ha decidido extraer de él un perfil de aplicación que se erige como un esquema de metadatos que identifica elementos asociados específicamente al servicio de VoD.

Un esquema con éstas características en esta investigación se justifica porque: 1) por medio del esquema de metadatos se apoya el despliegue de las GEP, las cuales muestran a los usuarios información descriptiva de los contenidos en cualquier servicio de VoD en televisión o web; o en una aplicación de un SR que inserte contenidos de VoD propios o de otras fuentes; y 2) el esquema permite identificar vínculos lógicos entre los contenidos y las demás entidades involucradas en este trabajo: las competencias y los mapas de competencias.

Siguiendo la metodología mostrada anteriormente en la Figura 3.1, se obtuvieron los siguientes resultados.

Fase A: determinar el alcance del perfil de aplicación/extensión

Inicialmente se identificó justificadamente la información manejada comúnmente en los servicios de VoD actuales, para lo cual se analizaron las plataformas Youtube y WatchKnowLearn teniendo en cuenta las características de la investigación:

- *Identificación (T1)*: información centrada en la identificación del contenido de VoD. Dada su naturaleza unívoca, la identificación permite vincular de forma precisa los contenidos de video (y como se verá más adelante, los segmentos de video) con las competencias y los mapas de competencia. También permite hacer referencias a los contenidos desde las GEP para extraer metadatos de interés para el usuario como el *título*, la *sinopsis*, la *duración*, entre otros.
- *Descripción (T2)*: información que expone de forma breve y general los puntos claves del contenido de VoD. Esta información es útil para construir las GEP, ya sea en una aplicación del SR propuesto, o en un servicio de VoD que lo integre como herramienta de asistencia al usuario. Para esto, se busca describir los siguientes aspectos: el *título*, *sinopsis*, *palabras clave*, *lenguaje*, *lugar de la producción*, *fecha de publicación*, *duración*, una *referencia a la descripción de una o más competencias* (hechas por medio de la extensión del IEEE RCD, la cual se propone en la siguiente sección) desarrolladas en el contenido.
- *Ubicación (T3)*: información sobre la ubicación del contenido con respecto a la plataforma proveedora del VoD. Esta información es especialmente útil si el servicio de VoD toma los contenidos de otras fuentes, por ejemplo, al insertar contenidos de Youtube en plataformas de noticias, entretenimiento, etc.; para lo cual, se usa la URL del contenido. Esta práctica, habitual hoy en día, permite ahorrar recursos de almacenamiento para quien los usa, al tiempo que se da mayor visibilidad a los

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

contenidos y sus creadores. En caso de generar una aplicación del SR propuesto con su propia GEP, los videos recomendados podrían ser reproducidos insertándolos por medio de su URL. Específicamente, se busca con el esquema describir los siguientes aspectos con relación a la ubicación del contenido: el *identificador del contenido* al cual la ubicación hace referencia, la *URL* del contenido.

- *Segmentos (T4)*: por medio de esta información, los segmentos de video pueden ser considerados como entidades vinculadas, pero independientes, a los contenidos a los cuales pertenecen. Así, se abre la posibilidad de formular estrategias de recomendación centradas en sugerir los segmentos relevantes para cada estudiante o los contenidos que los contienen. Con el esquema se busca describir los siguientes aspectos de los segmentos: *identificador*, *título*, *sinopsis*, *palabras clave*, una *referencia al contenido* al cual pertenece, *ubicación del segmento dentro de la línea de tiempo* del contenido, *duración*, *identificador*, una *referencia a la descripción de una o más competencias desarrolladas*.

Ejemplo 3.1. Analizando un contenido educativo de Youtube titulado *Teorema de Pitágoras Súper fácil*, se ha evidenciado que éste está descrito por los elementos de metadatos *título*, una *descripción* en lenguaje natural introducida por texto libre, la *duración* y *categoría*. Aunque el contenido trata aspectos asociados a las competencias, ninguno de estos elementos fue usado por el creador del contenido para indicarlos, lo cual dificulta que éste pueda ser considerado en un proceso de recomendación como el buscado en la presente investigación. Con el ánimo de abordar este problema, en el ejercicio de aplicar los cuatro tipos de información (T1, T2, T3 y T4) se obtuvo el modelo descriptivo mostrado en la Figura 3.2. En él se identifica dos segmentos: el *Segmento 1*, titulado *El triángulo rectángulo* y el *Segmento 2*, titulado *El Teorema de Pitágoras*. Para indicar a qué contenido pertenecen éstos, en su descripción se ha definido el identificador del contenido (marcado con azul), estableciendo un vínculo lógico entre éstas entidades.

Por otra parte, dentro de la descripción del contenido y los segmentos se hace una referencia (marcada en rojo) a las descripciones de las competencias *Reconozco las características de los triángulos rectángulos* (identificada como *id1*) y *Reconozco las operaciones involucradas en el Teorema de Pitágoras* (identificada como *id2*). De esta forma, se establecen vínculos lógicos entre los contenidos y las competencias. Al respecto, se puede notar que, aunque en la descripción del contenido no se hace referencia a la competencia identificada como *id2*, el vínculo entre estas entidades se establece indirectamente debido a que el *Segmento 2* hace referencia a tal competencia. Las descripciones de competencias a las que se hace alusión en la Figura 3.2 corresponden a la extensión del IEEE RCD propuesto más adelante en este documento. Finalmente, el modelo busca describir información de los contenidos de VoD (tales como el título, la sinopsis, la duración, entre otros) y los segmentos (por ejemplo, el título, las

palabras clave, entre otros) que puede ser usada para construir las GEP tanto para servicios de VoD como para los SR.

En conclusión, un modelo de descripción como el mostrado permite formalizar tanto vínculos lógicos entre entidades como información descriptiva de éstas, siendo estos aspectos mapeables en ontologías y, por lo tanto, aprovechables por un SR como el propuesto en esta investigación.

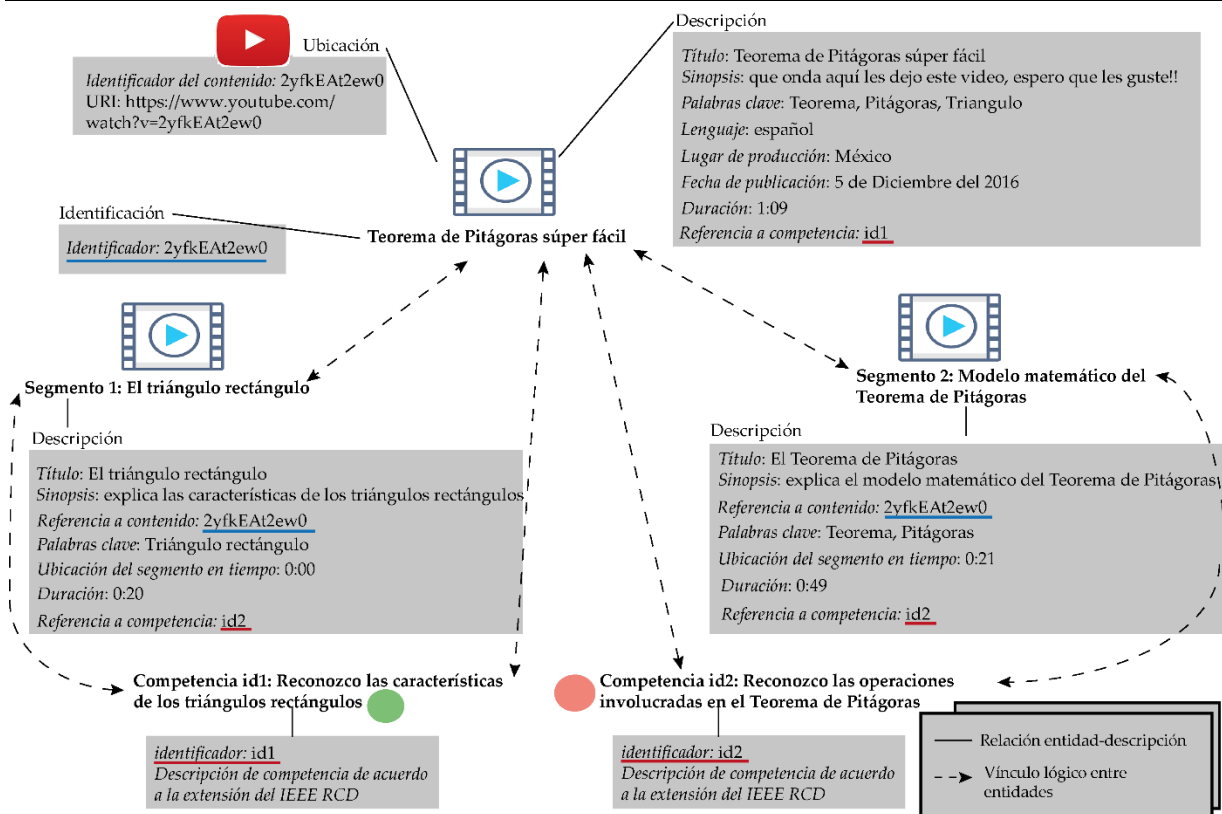


Figura 3.2 Ejemplo del modelo descriptivo de los contenidos desde el punto de vista del servicio de VoD. Fuente: propia.

Fase B.1: cotejar información a describir-esquema original

En esta fase se identificaron y seleccionaron de *TV-Anytime* los tipos de datos y elementos de datos mínimos necesarios para describir los cuatro tipos de información establecidos anteriormente (T1, T2, T3, T4). Los resultados se presentan en la Tabla 3.1, en la cual se relacionan el tipo de información que se desea describir y el elemento de datos de *TV-Anytime* correspondiente.

Fase B.2: modificar definición de elementos seleccionados

En esta fase se realizaron modificaciones en los elementos de *TV-Anytime* elegidos en cuanto a significado y función (sólo en caso de ser necesario). Esto se hizo teniendo en cuenta que los resultados expuestos en la investigación están estrechamente

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

relacionados con el servicio de VoD y las recomendaciones, lo cual no se evidencia en las definiciones originales de *TV-Anytime*. Estas adaptaciones se señalan en la *estructura del perfil de aplicación de TV-Anytime*, en la Fase C del proceso.

Fase B.3: determinar nuevos elementos a integrar

Como pudo observarse en la Fase B.1, la mayoría de la información que se busca describir a través del esquema de metadatos puede definirse a través de los elementos de *TV-Anytime*. Sin embargo, como era de esperar, las relaciones entre los contenidos y segmentos con las competencias no están soportadas. Para abordar este inconveniente, se propone incluir los siguientes dos nuevos elementos:

- **CompRef**: una referencia a la competencia o competencias desarrolladas en el contenido que permite el establecimiento de vínculos entre éste y tales competencias.
- **SegmentCompRef**: una referencia a la competencia o competencias desarrolladas en el segmento que facilita la creación de vínculos entre éste y las tales competencias.

Por otra parte, no fue posible definir vocabularios para los elementos que permitan minimizar el uso de texto libre. Por ende, la fase C no fue considerada.

Tipo de información a describir	Aspecto específico a describir	Elemento de <i>TV-Anytime</i>
Identificación (T1)	Identificador del contenido	programId
Descripción (T2)	El título del contenido	Title
	La sinopsis	Synopsis
	Las palabras clave	Keyword
	El lenguaje del contenido	Language
	El lugar y fecha de producción del contenido	ReleaseInformation
	La duración del contenido	Duration
	Una referencia a la competencia o competencias desarrolladas en el contenido	No soportado
Ubicación (T3)	El identificador del contenido al cual la ubicación hace referencia	Program
	La dirección URI del contenido	ProgramURL
Segmentos (T4)	El identificador del segmento	segmentId
	El título del segmento	Title
	La sinopsis del segmento	Synopsis
	Las palabras clave del segmento	Keyword
	Una referencia al contenido al cual pertenece el segmento	ProgramRef
	La ubicación del segmento dentro de la línea de tiempo del contenido	MediaRelTimePoint

	La duración del segmento	MediaDuration
	El identificador del segmento	segmentId
	Una referencia a la competencia o competencias desarrolladas en el segmento	No soportado

Tabla 3.1. Relación entre la información de los contenidos que se busca describir y los elementos de datos de *TV-Anytime*. Fuente: propia.

Fase D: determinar la estructura del perfil de aplicación/extensión

En esta fase se integraron los elementos del esquema original (los modificados y los no modificados) y los nuevos, manteniendo la estructura de *TV-Anytime*, en la cual los metadatos están asociados a los diferentes tipos de datos.

Como resultado final, se obtuvo la estructura mostrada en la Figura 3.3, en la cual se destacan los dos tipos de aportes realizados: *elementos de TV-Anytime adaptados en cuanto a definición* señalados en cursiva; y *nuevos elementos integrados* indicados en negrita. Los demás elementos son tomados tal como están definidos originalmente en *TV-Anytime*.

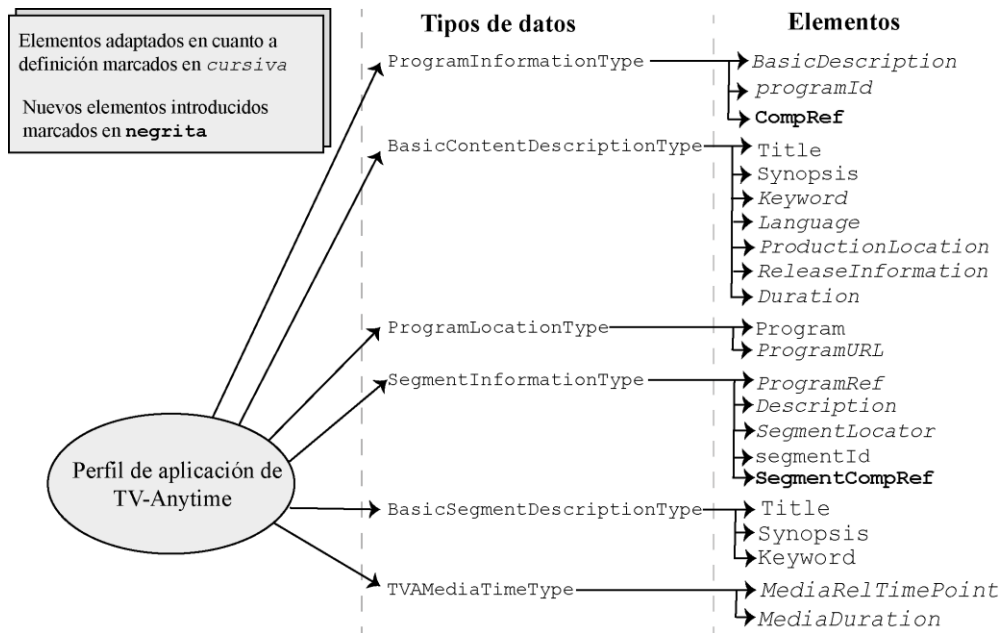


Figura 3.3 Estructura del perfil de aplicación de *TV-Anytime* enfocado en el VoD

Como parte del desarrollo de esta fase, también se realizó la integración de elementos, desde el punto de los tipos de datos, siguiendo la nomenclatura de *TV-Anytime*. Además, se realizó una definición formal de los elementos que conforman el perfil de aplicación propuesto, haciendo énfasis en su significado, al tiempo que se indica en

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

negrita la intención o importancia de cada elemento con respecto a los SR. Algunos se presentan a continuación:

- **Title:** Elemento obligatorio que define el título del contenido. Un contenido puede tener múltiples títulos, por ejemplo, en diferentes lenguajes.
- **Keyword:** Elemento obligatorio que contiene una lista de palabras clave del contenido. Una *keyword* puede ser una sola palabra o una frase compuesta por múltiples palabras. **Las palabras clave pueden ser usadas para establecer estrategias de recomendación basadas en similitudes sintácticas y semánticas.**

La integración de los elementos siguiendo la nomenclatura de *TV-Anytime* y la definición formal de éstos se puede consultar completamente en el Anexo E.

3.2 Extensión del IEEE RCD para la descripción de competencias

De acuerdo al análisis del estado del arte desarrollado en la Sección 2.2.2, el estándar IEEE RCD ofrece un esquema de metadatos para la descripción de competencias con fortalezas como las siguientes: 1) permite el uso de esquemas educativos formales con información externa para mejorar la comprensión de la competencia; 2) facilita la creación de extensiones, definiendo elementos de datos y tipos de datos específicos para tal propósito; 3) cuenta con el esquema SRCM para la formalización de mapas de competencias, facilitando la reutilización de sus descripciones; cuarto, es un estándar aceptado. Por estas fortalezas, el IEEE RCD ha sido seleccionado como el esquema base para construir una extensión, la cual es una decisión apoyada en el enfoque de selección basado en fortalezas y debilidades realizado en trabajos como [87] y [88]. Esto por encima de otros esquemas como *eCOTOOL*, el MCOS, RDCEO, y LOD. Dado que éstos tienen sus propias fortalezas identificadas en el estado del arte (ver Sección 2.2.2), tales como: 1) definen elementos de datos para la descripción de atributos específicos de competencias como alternativa al lenguaje natural, lo cual facilita la creación de conceptos comunes entre las personas; y 2) incluyen elementos adaptados de LOM, lo cual permite establecer vínculos lógicos entre competencias y OA; se ha decidido sumarlas a través de la extensión del IEEE RCD. Éste tipo de integraciones de diferentes esquemas para construir extensiones han sido realizadas en [89].

Para generar tal extensión, se desarrolló la metodología mostrada al inicio del capítulo. Cada fase se detalla a continuación.

Fase A: determinar el alcance del perfil de aplicación/extensión

En esta fase se definió inicialmente la información de las competencias que se desea describir. Ésta se obtuvo de las siguientes fuentes:

- Esquemas de metadatos para la descripción de competencias existentes, de los cuales se extrajeron los atributos específicos definidos como alternativa al lenguaje natural.
- Esquema de metadatos LOM, adaptando de él elementos que pueden traerse del ámbito de los OA al de las competencias.
- De los resultados de una evaluación de una versión previa del esquema enmarcada en la CM, en la que, de acuerdo al criterio de un grupo de expertos, se realizaron modificaciones en la definición de la información, y se añadieron nuevos elementos para obtener la versión aquí presentada. Esta información, y los detalles de la evaluación se extienden en la Sección 3.4.

De estas fuentes, se extrajeron once tipos de información (C1, C2, ..., C11) que pueden ser incluidos en el modelado de competencias. En su explicación, se señalan aspectos a considerar como su función, su posible uso dentro de una estrategia de recomendación y las posibilidades que ofrece en cuanto a la definición de asociaciones semánticas complejas al ser formalizadas en una ontología. Esto último teniendo en cuenta que el esquema de metadatos será mapeado en una ontología, la cual será usada por el SR para inferir conocimiento sobre los vínculos entre las entidades para mantener el carácter personalizado. Los once tipos de información se presentan a continuación (por su influencia en el proceso de recomendación, tres están explicados a continuación; todos se explican en el Anexo F, Sección F.1).

- *La categoría de las competencias (C1):* es información útil para posicionar las competencias dentro de una de las categorías *conocimiento, habilidad y capacidad*. Ésta información se extrajo del esquema eCOTOOL, el cual denomina a este grupo de categorías como KSC (“*Knowledge*”, “*Skill*”, “*Competence*”). Una clasificación y asignación formal de una categoría KSC asegura que las competencias puedan ser comparadas entre diversos sistemas educativos. Por otra parte, de acuerdo con la *Organización para la Cooperación y el Desarrollo Económico* (OCDE), las competencias se relacionan con actitudes y valores. Al respecto, países como Colombia y México han definido competencias ciudadanas para mejorar la convivencia desde el entorno escolar [90]. La *categoría de las competencias* ha sido seleccionada en esta investigación ya que a partir de ella es posible enfocar estrategias de recomendación en el apoyo conocimientos, habilidades, capacidades, actitudes y valores en las que los estudiantes tengan necesidades. Además, esta

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

información permite establecer vínculos lógicos entre competencias. También, tiene el suficiente significado para establecer asociaciones semánticas complejas al ser formalizada como una propiedad en una ontología (Anexo A, Sección A.3).

- *El verbo de acción involucrado en el desarrollo de la competencia (C2)*: según el *Diccionario Oxford de Inglés* [48], un *verbo de acción* expresa acción física o mental. Por lo tanto, es útil para describir la acción inherente a las competencias, independientemente del contexto de uso. La definición del verbo de acción como un atributo de las competencias se extrajo del esquema eCOTOOL. Esta información ha sido considerada porque permite enfocar las estrategias de recomendación al apoyo de verbos en los que los estudiantes tengan dificultades. Además, permite establecer vínculos lógicos entre competencias. También, tiene el suficiente significado para establecer asociaciones semánticas complejas al ser formalizada como una propiedad en una ontología.
- *El esquema de cualificación de la competencia (C3)*.
- *Rol de usuario al que interesa la competencia (C4)*.
- *La edad mínima del estudiante para la cual típicamente éste tiene la capacidad para desarrollar las competencias (C5)*.
- *Dificultad de aprendizaje de la competencia (C6)*.
- *Tiempo típico de aprendizaje de la competencia (C7)*.
- *Temas asociados a las competencias (C8)*.
- *Grado educativo en el que se aplica la competencia (C9)*: información que se refiere a grados escolares tales como *primero de primaria, segundo de primaria, décimo de secundaria*, etc. Ésta fue sugerida por el grupo de evaluadores como resultado de la evaluación mostrada en la Sección 3.4.
- *Palabras clave (C10)*.
- *Lenguaje de la descripción (C11)*.

Ejemplo 3.2. De acuerdo con el Ejemplo 3.1, se identificaron dos competencias en el contenido de Youtube *Teorema de Pitágoras súper fácil*. La primera, denominada *Reconozco las características de los triángulos rectángulos* (identificada como *id1*) y la segunda, *Reconozco las operaciones involucradas en el Teorema de Pitágoras* (identificada como *id2*). Es evidente que éstas tienen vínculos que vale la pena identificar en las descripciones. Aplicando los once tipos de información (C1, C2, ..., C11) en las competencias se obtiene el modelo mostrado en la Figura 3.4. Como puede observarse, en rojo se han indicado valores de descripción comunes entre las dos descripciones. Considerando que éstos tienen el

suficiente significado para vincular entidades, se puede concluir que el modelo permite establecer las relaciones lógicas entre las dos competencias. Precisamente, al formalizar estas descripciones en ontologías, las relaciones establecidas a través de los atributos toman la forma de *asociaciones semánticas complejas* (definidas en la Sección A.3, Anexo A), las cuales son explotadas por el SR propuesto en el presente trabajo para obtener las sugerencias. En el modelo también puede observarse que, aunque el *Esquema de cualificación* tiene el mismo valor para las dos descripciones, éste no tiene el suficiente significado para considerarlo como un atributo asociador. Algo similar ocurre con la *Dificultad de aprendizaje*, el *Tiempo de aprendizaje*, el *Rol de usuario* y la *Edad típica*. Sin embargo, estos tipos de información pueden ser usados en las estrategias de recomendación para enfocar las sugerencias en ellos, según sea el interés.

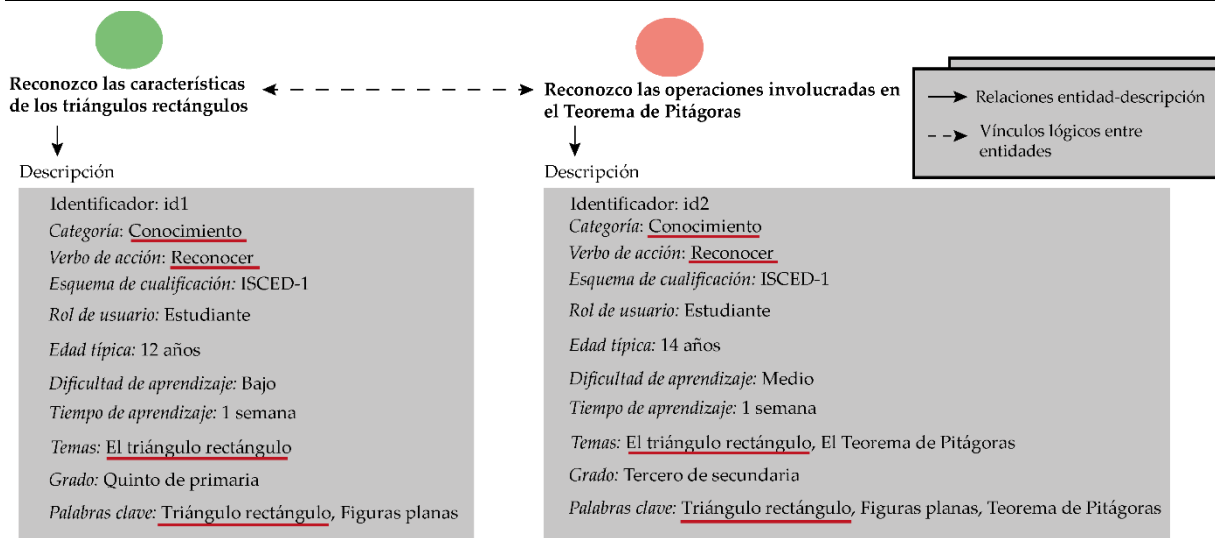


Figura 3.4 Modelo descriptivo de las competencias educativas. Fuente: propia.

Fase B.1: cotejar información a describir-esquema original

Después de un análisis del IEEE RCD se pudo verificar que éste no tiene los elementos necesarios para describir ninguno de los once tipos de información (C1, C2, ..., C11) definidos en la Fase A. Sin embargo, se pudo constatar que algunos de éstos están cubiertos por otros esquemas, los cuales fueron seleccionados para ser integrados. La relación entre los tipos de información, los elementos de datos y el esquema de origen se muestra en la Tabla 3.2.

Tipo de información a describir	Elemento de datos	Esquema origen
La categoría de las competencias (C1)	KSC_category	eCOTOOL
El verbo de acción involucrado en el desarrollo de la competencia (C2)	action_verb	eCOTOOL
El esquema de cualificación de la competencia (C3)	level_attributions, el cual se compone de otros	eCOTOOL

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

	elementos: scheme y level	
Rol de usuario al que interesa la competencia (C4)	intended_end_user_role	LOM
La edad mínima del estudiante para la cual típicamente éste tiene la capacidad para desarrollar las competencias (C5)	typical_age	LOM
Dificultad de aprendizaje de la competencia (C6)	difficulty	LOM
Tiempo típico de aprendizaje de la competencia (C7)	typical_learning_time	LOM
Temas asociados a las competencias (C8)	topic	-
Grado educativo en el que se aplica la competencia (C9)	grade	-
Palabras clave (C10)	keyword	-
Lenguaje de la descripción (C11)	language	-

Tabla 3.2. Relación entre la información de las competencias que se busca describir y los elementos de datos correspondientes. Fuente: propia.

En cuanto a la definición de estos elementos, no se realizó ninguna modificación con respecto a la ofrecida en los esquemas origen; por lo cual, no fue desarrollada la Fase B.2 de la metodología.

Fase B.3: determinar nuevos elementos a integrar

Como pudo observarse en la Fase B.1, algunos tipos de información no están definidos por los esquemas de metadatos involucrados, por lo cual, fue necesario especificar un grupo de nuevos elementos para integrar en la extensión del IEEE RCD:

- `topic`: uno o más temas relacionados con la competencia descrita.
- `grade`: grado escolar de la competencia.
- `keyword`: palabras clave relacionadas con la competencia.
- `Language`: lenguaje de la descripción.

Fase C: minimizar el uso de la entrada de texto libre

En esta fase se definió un grupo de vocabularios controlados para algunos de los elementos, lo cual posibilita que, en una aplicación del esquema, el usuario deba seleccionar un(as) opción(es) para asignar un valor al elemento correspondiente. Los vocabularios propuestos se pueden consultar en la Tabla 3.3, los cuales fueron concebidos después de analizar: primero, los vocabularios de otros esquemas (LOM, eCOTOOL); segundo, consultar los niveles de cualificación tanto de la CINE [91] como del EQF [51]; tercero, el grupo de verbos mutuamente excluyentes propuestos en el perfil de aplicación de LOM para España llamado LOM-ES [92]. Otros valores se introducen a través de texto libre de acuerdo con el estándar iso 10646 [93].

Establecer vocabularios para algunos elementos de datos es un hecho no menor. Retomando el ejemplo sobre el modelo descriptivo de las competencias mostrado en la Figura 3.4, la definición de valores predefinidos evita problemas inherentes a la variabilidad léxica de las personas, lo cual potencia la creación de vínculos lógicos entre competencias sin errores. Por ejemplo, una persona va a describir la competencia *Identifico las partes del círculo*, y otra va a describir la competencia *Reconozco las partes del círculo*. Si la información acerca del verbo fuera definida en texto libre, posiblemente la primera persona establezca el verbo *Identificar* y la segunda el verbo *Reconocer*, lo cual dificultaría la creación de relaciones lógicas entre las competencias (sin algún tipo de procesamiento léxico previo), aun cuando su concepto es el mismo dado que son sinónimos. Este problema es resuelto con el uso de un vocabulario como el definido para el elemento `action_verb`, el cual obliga a definir el verbo *Reconocer* en ambos casos.

Elementos	Vocabularios
scheme	Isced-unesco, EQF
level	<ul style="list-style-type: none"> • Para <i>isced-unesco</i>, los niveles pueden ser consultados en [91] (por ejemplo, 0, 1, etc.) • Para EQF, los niveles pueden ser consultados en [94] (por ejemplo, k1, k2, s1, etc.)
KSC_category	<i>Knowledge, skill, competence, attitude, value</i>
action_verb	38 verbos tomados de LOM-ES, los pueden ser consultados en [92] (por ejemplo, <i>analyzing, comparing, creating, recognizing, etc.</i>)
intended_user_role	<i>Student, teacher, manager y author</i>
difficulty	<i>Difficult, Easy, Medium difficulty, Very difficult, Very easy</i>
language	Códigos de dos letras para lenguajes, iso 639-1 (por ejemplo, es, en, it, etc.)

Tabla 3.3. Vocabularios para los elementos del esquema de metadatos para competencias. Fuente: propia.

Fase D: determinar la estructura del perfil de aplicación/extensión

En esta fase se integran los elementos seleccionados de los esquemas origen y los nuevos en la estructura del IEEE RCD. Como resultado final, se obtuvo la estructura mostrada en la Figura 3.5, en la cual se destacan los dos tipos de aportes realizados: elementos integrados de otros esquemas señalados en cursiva; y los nuevos elementos indicados en negrita. Como puede observarse, todos los elementos han sido integrados dentro del tipo de dato `extensión_type`. En la estructura, los elementos y tipos de datos originales de IEEE RCD han sido omitidos, pero se asume que siguen siendo considerados (pueden ser consultados en [37]).

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

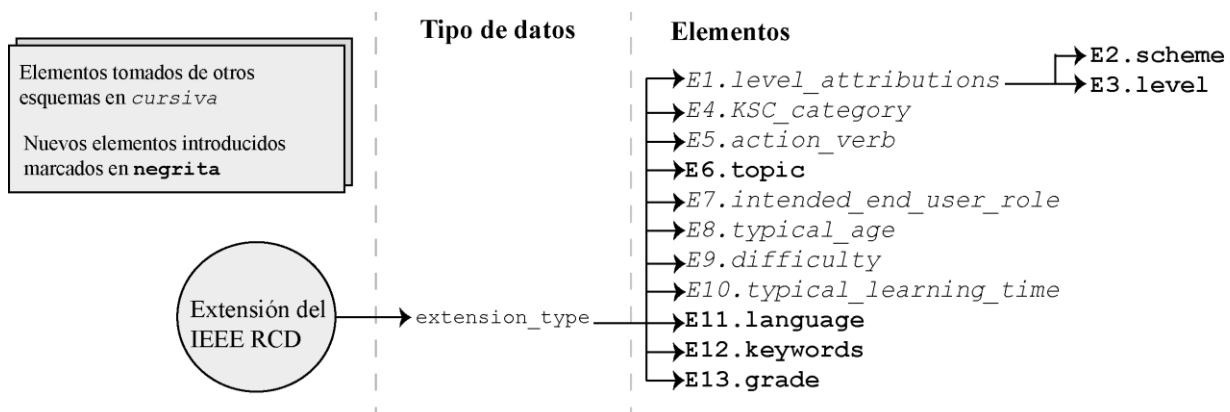


Figura 3.5 Estructura de la extensión del IEEE RCD

Como parte del desarrollo de esta fase, también se realizó la integración de elementos, desde el punto de vista de los tipos de datos, siguiendo la nomenclatura del IEEE RCD. Además, se realizó una definición formal de los elementos en cuanto a su significado. Por ejemplo:

KSC_category: elemento obligatorio usado para definir la categoría KSC de la competencia descrita. Sus valores se pueden aplicar a través del vocabulario correspondiente definido en la Tabla 3.3. Por ejemplo, si se identifican dificultades para desarrollar *habilidades*, entonces conviene sugerir contenidos vinculados a las competencias que hacen parte de tal categoría.

La integración de elementos siguiendo la nomenclatura del IEEE RCD y la definición formal de éstos se puede consultar de forma completa en el Anexo F, Sección F.2.

3.3 Modelo descriptivo integral ofrecido por el esquema

Tal como se introdujo en el presente capítulo, las competencias y los mapas pueden ser vinculados en [38] por medio de un elemento específicamente definido en SRCM para tal fin. Aprovechando esto, y considerando que en el perfil de aplicación *TV-Anytime* propuesto fueron definidos dos elementos (*CompRef* y *egmentCompRef*) cuya función es vincular contenidos y competencias, al final del desarrollo de la metodología se obtuvo en conjunto el esquema de metadatos para la descripción de contenido de VoD de acuerdo a las competencias. Este esquema ofrece un modelo descriptivo integral como el ejemplificado en la Figura 3.6, el cual se basa en los ejemplos 3.1 y 3.2.

Usando el SRCM se ha formalizado el *Mapa de competencias 1*, el cual está compuesto por tres nodos, los cuales referencian las descripciones de las competencias por medio de sus identificadores (marcado en verde). Como puede observarse, el *Nodo2* es un subnodo del *Nodo1*. Dado que el *Nodo1* referencia a la competencia con el identificador *id2*, y el *Nodo2* lo hace con la competencia identificada con el identificador *id1*, se puede inferir que existe un vínculo lógico entre las dos competencias (marcado en verde), indicando que la segunda es subcompetencia de la primera.

Retomando lo dicho en las secciones anteriores, con referencias a identificadores entre entidades, también se establecen vínculos entre contenidos de VoD y sus segmentos (marcados en azul), entre contenidos de VoD y competencias (marcados en rojo) y entre competencias por medio de sus atributos (marcados en amarillo).

Una vez esta información sea formalizada en la ontología, estos vínculos, en forma de *asociaciones semánticas complejas* (Anexo A, Sección A.3), pueden ser inferidos y aprovechados por el SR para la obtención de recomendaciones a través de la aplicación de técnicas de la web semántica.

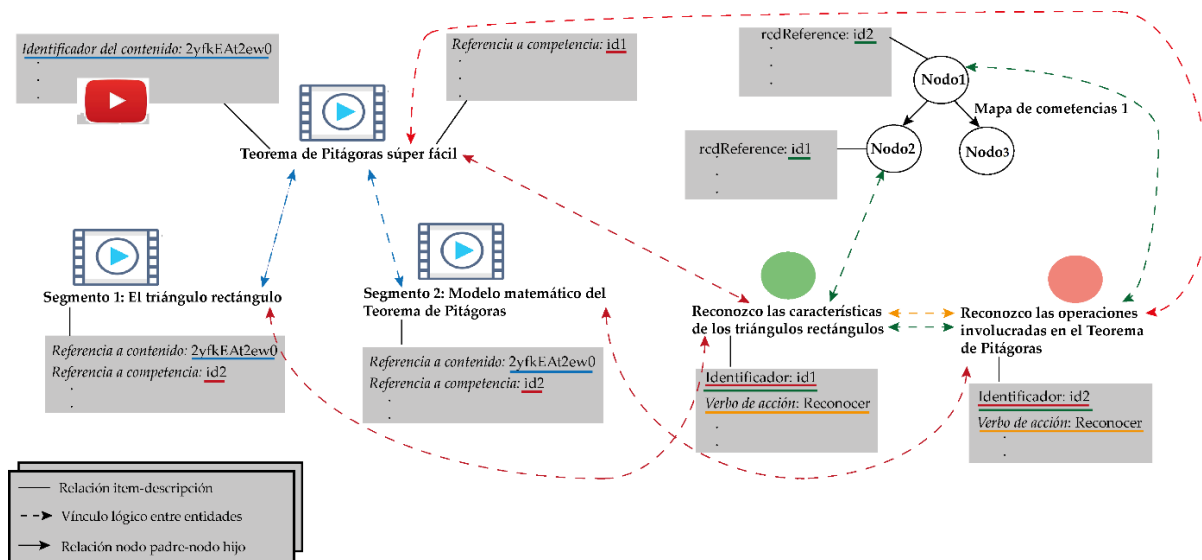


Figura 3.6 Modelo descriptivo integral. Fuente: propia.

3.4 Evaluación de la extensión del IEEE RCD para la descripción de competencias

La evaluación del esquema de metadatos para el modelado de competencias fue desarrollada para validar la nueva información incluida en el IEEE RCD, esto a través de la verificación del cumplimiento de algunas características de calidad [55] según el

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

juicio de un grupo de evaluadores de Colombia, Argentina, México y España; y si es el caso, realizar una serie de modificaciones soportadas en los resultados y enfocadas en la mejora de la propuesta. Así, se busca que el esquema sea el resultado del consenso dentro de una comunidad de uso. La evaluación se realizó sobre una versión previa del esquema (Anexo S), siendo la extensión del IEEE RCD propuesta en este capítulo el resultado final después de las modificaciones. Debido a que la CM se propuso en la literatura para evaluar registros y repositorios de metadatos ([55], [59], [95]) en esta investigación se realizó una adaptación de éstos de tal forma que pudieran ser aplicables a los esquemas de metadatos en su etapa de concepción. Como resultado de tal adaptación, se construyeron dos instrumentos de evaluación web útiles para medir el cumplimiento de las características de calidad y capturar las sugerencias de los evaluadores.

Cada uno de estos aspectos es explicado brevemente a continuación. Una descripción detallada de la evaluación se presenta en el Anexo C.

Teniendo en cuenta que las características de calidad de los metadatos han sido definidas en principio para evaluar repositorios de metadatos (en búsqueda de problemas ocasionados durante las tareas de marcación), fue necesario determinar las únicas que, después de una interpretación, pudiesen ser aplicadas a los esquemas de metadatos. La idea de hacerlo de esta forma nace de la necesidad de reducir las posibilidades de cometer errores durante la descripción de competencias por parte de las personas. Como resultado se seleccionaron de [55] las siguientes características de calidad: *completitud* (los registros de metadatos describen completamente los ítems), *conformidad con las expectativas* (los usuarios encuentran en los esquemas lo que esperan encontrar) y *accesibilidad* (el esquema es entendible por los usuarios). Otras características como *procedencia* (confiabilidad de la entidad que realiza la marcación de los ítems), *consistencia lógica y coherencia* (heterogeneidad de registros de metadatos, colecciones de metadatos federadas) y *precisión* (edición de alta calidad durante la marcación) no fueron consideradas porque su evaluación sólo tiene sentido sobre repositorios de metadatos.

Después de interpretar las características seleccionadas, se establecieron un grupo de preguntas guía que fueron útiles para diseñar los instrumentos de evaluación y, por lo tanto, para verificar si el esquema propuesto cumple las características de calidad. Esta información se resume en la Tabla 3.4.

Característica	Interpretación	Preguntas guía
Completitud	Los elementos de datos describen completamente las competencias	P1. ¿Los elementos de datos son suficientes para describir competencias? P2. ¿Qué elementos de datos son innecesarios?

Accesibilidad	Los elementos de datos son apropiados para la comunidad de usuarios	P3. ¿Los elementos de datos son confusos?
Conformidad con las expectativas	El esquema de metadatos propuesto funciona para lo que fue creado	P4. ¿El esquema cumple las expectativas? P5. ¿Los elementos de datos son apropiados para cumplir funciones específicas?
	Los vocabularios propuestos son adaptados a las características del usuario y a la forma en cómo se comprenden las competencias	P6. ¿Es apropiado elegir una opción de entre un grupo de éstas para asignar un valor a un determinado elemento de datos? P7. ¿Los vocabularios propuestos son apropiados para los elementos de datos?

Tabla 3.4 Características de calidad, interpretación y preguntas guía adaptadas a los esquemas de metadatos. Fuente: propia.

Los siguientes perfiles fueron definidos para seleccionar los evaluadores del esquema de metadatos:

- *Experto*: docente o no, con títulos o conocimientos en pedagogía y experiencia en tecnologías de aprendizaje en línea, *Objetos de Aprendizaje* o similares.
- *Usuario*: docente con experiencia en educación primaria, secundaria o superior.

En total, participaron once evaluadores con perfil experto y ocho con perfil usuario.

Por otra parte, se diseñaron y aplicaron dos instrumentos de evaluación web, uno para el perfil experto³ (enfocado en aspectos descriptivos y técnicos) y otro para el perfil usuario⁴ (enfocado sólo en aspectos descriptivos). Ambos involucran las siguientes partes: 1) contextualización; 2) selección de una competencia de entre un grupo predefinido de ellas; 3) aplicación del esquema para modelar la competencia seleccionada; 4) presentación de preguntas guía (ver Tabla 3.4). Después de la aplicación de los instrumentos de evaluación, la información coleccionada fue organizada de acuerdo a las características de calidad evaluadas. Los cambios y modificaciones sugeridos por los evaluadores ya se reflejan en el esquema propuesto.

Con respecto a los resultados, en cuanto a la *completitud*, pregunta guía P1, algunos expertos sugirieron la inclusión de aspectos relacionados con el currículo escolar, específicamente, el grado escolar. Por lo tanto, el elemento de datos *grade* fue incluido en el esquema de metadatos.

La Figura 3.7a presenta los resultados para la pregunta guía P2. Aproximadamente el 25% de los evaluadores (expertos y usuarios) consideran que *typical_learning_time*, *KSC_category* y *language* son innecesarios. Sin

³ Instrumento de evaluación para expertos <https://goo.gl/gvFp1x>. Los resultados pueden ser consultados en: https://mega.nz/#!rnAhyIKI!1h2ee3iYDci3gdwjlL_MdYtNMszeXpztJBopgzZtwo

⁴ Instrumento de evaluación para usuarios <https://goo.gl/X44i2w>. Los resultados pueden ser descargados de: <https://mega.nz/#!zjB3RCwJ!rV70qJP50TkaZslsANnORU7fB-cfUamz7Fv9tdkmjgg>

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

embargo, como tales valores son bajos y, además, han sido definidos en otros esquemas, los elementos no han sido modificados. Específicamente, en el caso de `language`, aunque un grupo de evaluadores consideran que no es relevante especificar el lenguaje de la descripción, para la mayoría sí lo es, teniendo en cuenta que actualmente éste contribuye en el posicionamiento de las descripciones en la web, por ejemplo, para realizar búsquedas.

Dados los porcentajes presentados, se concluye que el esquema de metadatos propuesto cumple la característica de completitud.

Con respecto a la *accesibilidad*, la Figura 3.7b presenta los resultados para la pregunta guía P3. Para el 25% de los evaluadores y el 18% de expertos consideraron que `level_attributions`, `scheme` y `level` son confusos. `KSC_category` resulta confuso para el 12% de usuarios y el 9% de expertos. Adicionalmente, `topic`, `intended_user_role` y `typical_age` son confusos para el 12% de los expertos. Con el propósito de mejorar la propuesta, se han redefinido las explicaciones de estos elementos añadiendo información útil para construir una propuesta claramente comprensible. Con esto se presume una mejora del esquema en cuanto a accesibilidad.

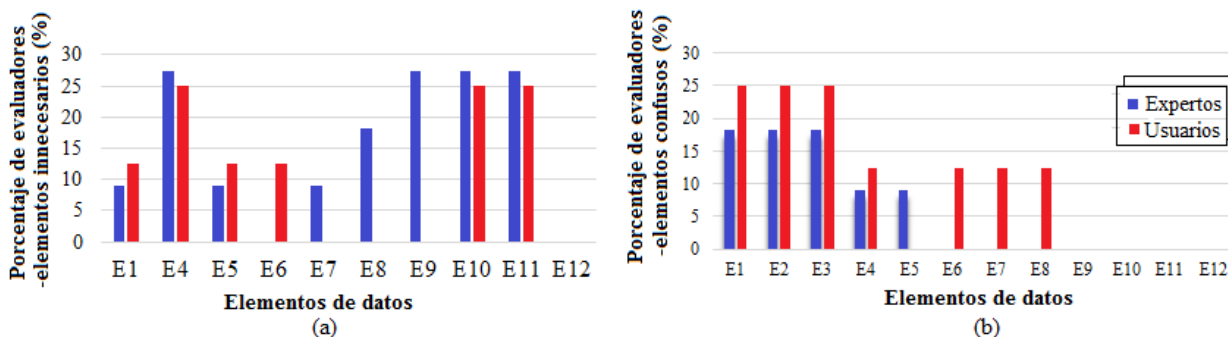


Figura 3.7 Resultados para las preguntas guía: (a) P2, basados en el porcentaje de evaluadores que consideran que los elementos son innecesarios; y (b) P3, basados en el porcentaje de evaluadores que consideran que los elementos son confusos. Fuente: propia.

Dados los porcentajes presentados en la evaluación, se concluye que el esquema propuesto cumple la característica de accesibilidad.

En cuanto a la *conformidad con las expectativas*, la Figura 3.8a muestra los resultados obtenidos para la pregunta P4. En promedio, los evaluadores consideran que el esquema cumple sus expectativas. Los resultados de la pregunta P5 se muestran en la Figura 3.8b. En general, todos los elementos fueron considerados apropiados para cumplir sus funciones específicas. Específicamente, `typical_learning_time` fue considerado inapropiado por el 12% de usuarios, y `language` fue considerado

inapropiado para el 9% de expertos y el 25 % de usuarios. Dado que estos valores son bajos, los elementos mencionados no fueron modificados.

La Figura 3.8c muestra los resultados de la pregunta guía P6. En general, la mayoría de evaluadores están de acuerdo con el uso de vocabularios para definir los valores de los elementos de datos a los que están asociados. En concreto, el 12% de los usuarios no estuvieron de acuerdo con el uso de vocabularios para los elementos `action_verb`, `typical_age`, `difficulty`. Además, el 9% de los expertos no estuvieron de acuerdo con el uso de vocabularios para `difficulty` y `language`. De acuerdo con los comentarios de los evaluadores, los valores para `action_verb` podrían ser asignados por medio de texto libre. Debido a que la mayoría de evaluadores están de acuerdo con usar vocabularios para `action_verb`, en el esquema de metadatos se incluyó el elemento `other` por medio del cual el usuario puede definir un verbo no incluido en el vocabulario. Debido a que los evaluadores no hicieron sugerencia acerca de los demás elementos, no se establecieron alternativas al uso de vocabularios para ellos.

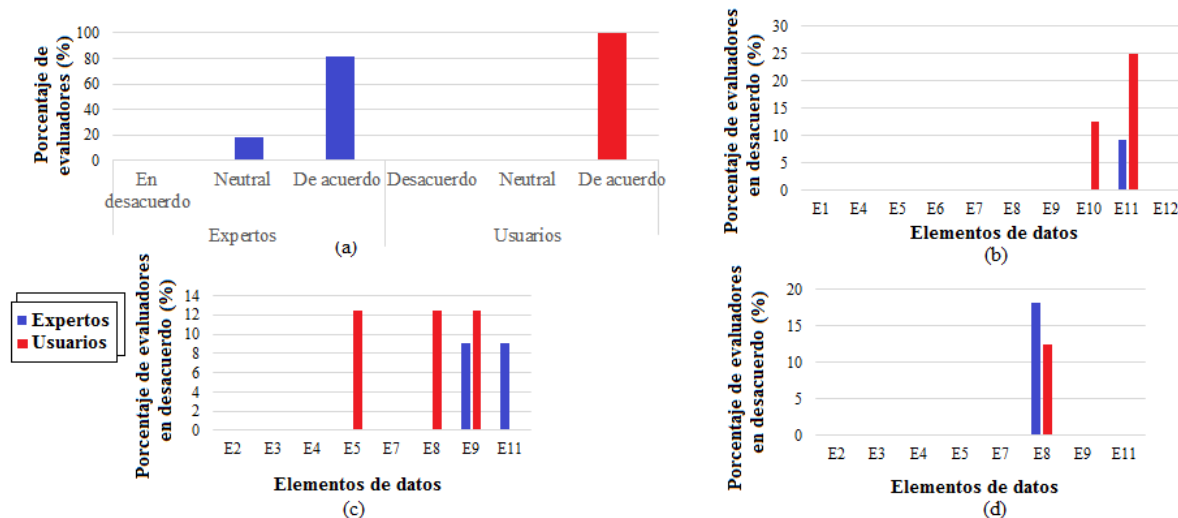


Figura 3.8 Resultados para las preguntas guía: (a) P4, basados en el porcentaje de evaluadores que están de acuerdo con que el esquema cumple sus expectativas; (b) P5, de acuerdo con que los elementos cumplen sus funciones; (c) P6, basados en el porcentaje de evaluadores que no están de acuerdo con que los valores deben ser asignados por medio de vocabularios; y (d) P7, de acuerdo con el porcentaje de evaluadores que no están de acuerdo con los vocabularios propuestos. Fuente: propia.

Los resultados de la pregunta P7 se presentan en la Figura 3.8d. En general, la mayoría de evaluadores están de acuerdo con los vocabularios definidos en el esquema. Sin embargo, el 18% de los expertos y el 12% de los usuarios no están de acuerdo con el vocabulario de `typical_age`. Dado que inicialmente se estableció un

3. Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas

vocabulario compuesto por rangos de edades, se decidió modificar esto definiendo un valor entero de forma manual. Esta decisión se basa en que como los rangos inicialmente definidos eran pequeños (de dos años por rango) y que la posibilidad de ocurrencia de errores tipográficos al escribir números de máximo dos cifras puede ser baja, introduciendo un valor entero se presume que se puede satisfacer a todos los usuarios en cuanto a la definición de una edad típica para el aprendizaje de la competencia. Dados los porcentajes presentados, se concluyó que el esquema propuesto cumple la característica de conformidad con las expectativas.

En resumen, la evaluación del esquema de metadatos para competencias permitió recoger aportes para la mejora del mismo. En concreto: 1) los expertos sugirieron la inclusión del grado escolar en el esquema, lo cual se hizo definiendo el elemento `grade`; 2) los expertos encontraron confusos los elementos `topic`, `intended_user_role` y `typical_age`, por lo cual se redefinieron; 3) los expertos encontraron no adecuado expresar el valor de `typical_age` como un rango de valores enteros. En consecuencia, se permitió la expresión de tal valor a través de un entero con texto libre; 4) entre los comentarios de los evaluadores (expertos y usuarios) se encontró que prefieren introducir en texto libre los valores de `action_verb`, por lo cual, se incluyó tal posibilidad adicionalmente a la definición por medio del vocabulario definido.

Capítulo 4

Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

En el estado del arte (Sección 2.2.1) se estudió la importancia de las ontologías en los *Sistemas de Recomendaciones* (SR) educativos [11], [8], [21], [22], la cual radica en la posibilidad de intercambiar, construir y reutilizar conocimiento del contexto educativo que puede ser relevante en el proceso de personalización. Específicamente, a diferencia de otras estructuras como las taxonomías y los tesauros, las ontologías posibilitan la inferencia de información y ofrecen la posibilidad de formalizar conocimiento más enriquecido. Esto, gracias a que pueden ser codificadas en un formato procesable por software, por lo cual la inferencia puede ser realizada por máquinas [96].

Por otra parte, las ontologías juegan un papel preponderante en la construcción de ambientes de *Datos Abiertos Enlazados* (DAE), en los cuales se publican datos estructurados en la web [97], posibilitando la formalización de recursos y fomentando el descubrimiento de información útil en la web.

En síntesis, las ontologías ofrecen la posibilidad de introducir más conocimiento a los procesos de recuperación de información, incluidos los propios de los SR. Para ejemplificar el aporte de las ontologías en la recomendación, se considera un currículo de un curso en el que se registra el mapa de competencias consignado en la Figura 2.1 (Sección 2.1.4). Suponiendo que un estudiante tiene necesidades educativas en la competencia *Reconozco y comparo las propiedades geométricas usadas en la demostración de teoremas básicos*, un SR inteligente no debería tener problemas en considerar la competencia *Defino el teorema de Pitágoras*, así en la primera competencia no aparezca en su nombre el término *Pitágoras*. En caso de requerir contenidos de apoyo para el estudiante, no sólo podrían aprovecharse los contenidos vinculados con la competencia en las que hay necesidades educativas, sino también los vinculados con las competencias que apoyan el aprendizaje de ésta. En este caso,

se han explotado las relaciones de subsunción de los mapas para inferir conocimiento útil, lo cual puede realizarse automáticamente con la formalización en una ontología de todo el conocimiento involucrado. Por otra parte, en un proceso soportado por técnicas sintácticas (descriptores basados en palabras clave o etiquetas), la recomendación se basaría en competencias o contenidos para los que en sus descripciones aparezcan palabras como *teorema* o *propiedad geométrica*, para lo cual el resultado sería problemático: lleno de *falsos positivos* (contenidos recomendados no relevantes) y *falsos negativos* (contenidos relevantes no recomendados). Por ejemplo, sería incapaz de inferir que un contenido sobre el *Teorema de Pitágoras* podría ser útil para el estudiante, ignorando el hecho de que tal teorema hace parte de los *teoremas básicos*.

Dado que todas las ventajas mencionadas anteriormente pretenden ser aprovechadas en la presente investigación, la generación de una ontología enmarcada en el dominio de los contenidos de VoD y competencias es una tarea clave que posibilita el establecimiento de relaciones entre tales entidades, las cuales pueden ser usadas para extraer conocimiento útil en el desarrollo de estrategias de recomendación.

Así las cosas, este capítulo presenta una *ontología sobre el dominio de los contenidos de VoD y las competencias educativas*, la cual es el producto de la adaptación de la información descrita en el esquema de metadatos del Capítulo 3 a elementos ontológicos (clases, entidades y propiedades). Lo anterior teniendo en cuenta que: 1) el esquema define elementos descriptivos de, y vínculos entre contenidos, competencias y mapas de competencias que pueden ser aprovechados por la ontología; 2) es una práctica común construir ontologías a partir de esquemas de metadatos para permitir la inferencia de conocimiento sobre las descripciones [98], [99].

La concepción de la ontología aquí propuesta pasó por el desarrollo de una adaptación de la metodología descrita en [100] para la creación de ontologías y el “*framework*” *Methontology* [101], el cual es útil para organizar y estructurar el conocimiento en un modelo completo y consistente, usando representaciones externas independientes de lenguajes y entornos de implementación denominadas *Representaciones Intermedias* (RI). La metodología se ilustra en la Figura 4.1, en la cual se pueden ver las fases involucradas y los productos obtenidos en cada una de ellas.

Fase A: analizar ontologías existentes. Fase en la que se hace una revisión de las ontologías existentes con el fin de seleccionar una(s) que pueda(n) ser reutilizada(s), adaptándola(s) a los requerimientos del trabajo. Aquí es importante indicar que en el estado del arte (Sección 2.2.4) se realizó una evaluación de las ontologías existentes en el dominio del VoD, los *Objetos de Aprendizaje* (OA) y las competencias; por lo cual

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

esta fase se da por realizada y no es considerada en el presente capítulo. De tal evaluación se seleccionaron las siguientes ontologías para desarrollar la ontología propuesta en esta investigación.

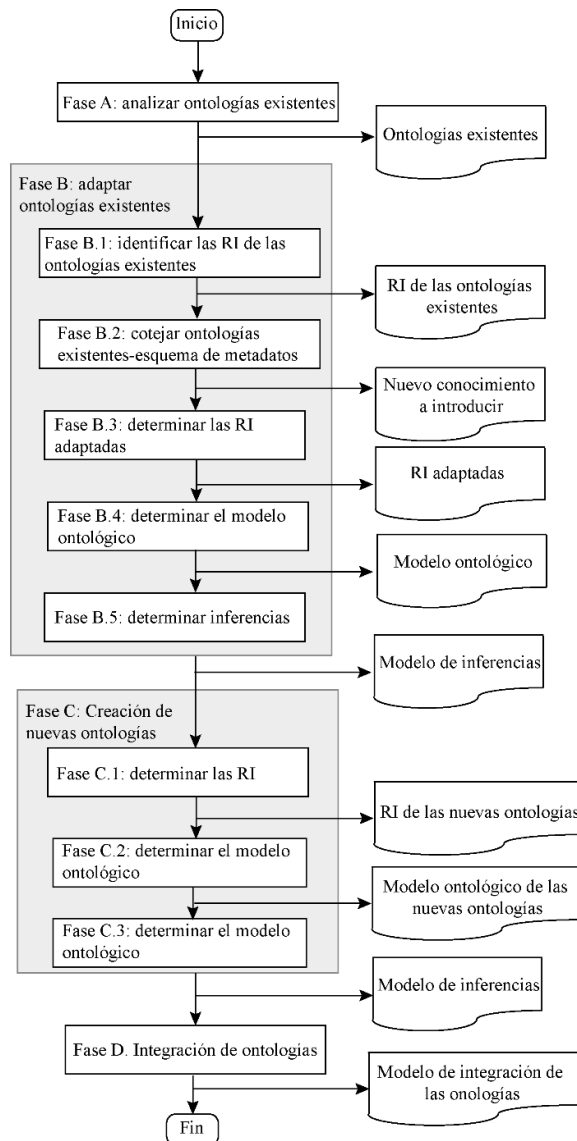


Figura 4.1 Proceso desarrollado para el diseño de la ontología del dominio de los contenidos de VoD y las competencias

- “*Learning Object Metadata Ontology*”: una ontología que modela los conceptos en torno al esquema LOM. La necesidad de adaptar esta ontología radica en la oportunidad de proveer conocimiento al SR acerca de las características del contenido de VoD propias de los OA.
- “*Ontology for Media Resources*”: ontología que modela todos los conceptos en torno a los recursos multimedia. La adaptación de esta ontología se realiza para permitir al

SR descubrir relaciones entre contenidos y segmentos de video, así como de contenidos y competencias. Esto considerando que en el perfil de aplicación de *TV-Anytime* propuesto en el Capítulo 3 se definieron tales relaciones a través de metadatos específicos para tal fin.

- “*Competency model ontology*”: es una ontología definida para el modelado de competencias. Tal como se mostró en el estado del arte (Sección 2.2.4), ésta hace parte de un “*framework*” de ontologías, lo que permite asociar las competencias a otros aspectos de la educación. La necesidad de adaptar esta ontología en el presente trabajo se basa en la posibilidad de proveer al SR conocimiento acerca de los vínculos entre competencias. Y es que en la extensión para la descripción de competencias propuesta en el Capítulo 3 se definió un grupo de metadatos, los cuales, a través de su adaptación en la ontología, posibilitan la inferencia de vínculos entre competencias por medio de propiedades comunes entre éstas.

El proceso de adaptación de las tres ontologías se desarrolla a través de la siguiente fase.

Fase B: adaptar ontologías existentes. En esta fase se adaptan las ontologías existentes seleccionadas en la *fase A* de acuerdo a los requerimientos del trabajo, esto es, adaptarlas de tal forma que permitan formalizar toda la información definida en el esquema de metadatos descrito en el Capítulo 3. Esta fase involucra tres subfases.

Fase B.1: identificar las RI de las ontologías existentes. En esta fase se identifican las RI de las ontologías existentes, incluyendo el *diccionario de conceptos*, *relaciones binarias*, *tabla de atributos de instancias* y *redes semánticas*.

Fase B.2: cotejar ontología existente-esquema de metadatos. Fase en la que se identifica *nuevo conocimiento a introducir* en la ontología existente; lo cual corresponde a la información del *esquema de metadatos para contenidos de VoD de acuerdo a las competencias* (propuesto en el Capítulo 3) que no es posible formalizar a través de la ontología existente en su versión original.

Fase B.3: determinar las RI adaptadas. Fase en la que se introduce a la ontología existente la información del esquema de metadatos que ésta no permite formalizar originalmente. Como resultado se obtienen un grupo de *RI adaptadas*.

Fase B.4: determinar el modelo ontológico. En esta fase se define el *modelo ontológico adaptado* desde el punto de vista de la implementación, esto es, se definen clases, instancias, propiedades (“*data properties*” y “*object properties*”) y tipos de datos a partir de las RI de la *fase B.3*.

Fase B.5: determinar inferencias. Si es posible, se determina un *modelo de inferencias* sobre el modelo ontológico generado en la *fase B.4*.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

Fase C: crear nuevas ontologías. En esta fase se construyen las ontologías que se crean pertinentes. Al respecto, en el estado del arte (Sección 2.2.4) se mencionó que en los trabajos de la literatura revisados no fue posible hallar una ontología para la formalización de mapas de competencias. En consecuencia, hubo la necesidad de crear una a partir del esquema de metadatos “*Simple Reusable Competency Map*” (SRCM) el cual fue justificadamente seleccionado en el estado del arte (Sección 2.2.4). Tal como se sugirió anteriormente, la formalización de mapas a través de la ontología permitirá realizar inferencias sobre las relaciones de subsunción, lo cual dotará al SR de conocimiento útil sobre las competencias que apoyan al aprendizaje de otras. Tal conocimiento podría ser usado para enfocar la recomendación en las competencias que apoyan a las que hacen parte de las necesidades del estudiante.

El proceso de generación de la nueva ontología se desarrolla a través de las siguientes subfases.

Fase C.1: determinar las RI. Fase en la que se determinan las *RI* de las nuevas ontologías.

Fase C.2: determinar el modelo ontológico. En esta fase se define el *modelo ontológico de la nueva ontología* desde el punto de vista de la implementación, esto es, se definen clases, instancias, propiedades (“*data properties*” y “*object properties*”) y tipos de datos a partir de las *RI* de la fase C.1.

Fase C.3: determinar inferencias. Si es posible, se determina un modelo de inferencias sobre el modelo ontológico de las nuevas ontologías generado en la *fase C.2*.

Fase D. Integración de ontologías. En esta fase se realiza un proceso de emparejamiento a nivel técnico para determinar relaciones entre las ontologías, a partir del cual se produce una ontología que, a su vez, ofrece un modelo integrado de ontologías vinculadas.

Como resultado del desarrollo de la metodología se obtiene la *ontología sobre el dominio de los contenidos de VoD y las competencias educativas*.

En cuanto a las contribuciones al conocimiento ofrecidas en este capítulo se tiene:

1. La integración de nuevas entidades y propiedades a las ontologías existentes para la descripción de y vinculación entre contenidos de VoD, competencias y mapas de competencias.
2. Una ontología para la formalización de mapas de competencias denominada aquí *OntoRCM*.
3. Una ontología del dominio de los contenidos de VoD y las competencias producto de la integración de las ontologías mencionadas en 1 y 2.

Cabe mencionar que la ontología fue evaluada según el criterio de *completitud* [39], a partir del cual se mide si se cubre el dominio de interés de forma apropiada; lo cual es un aspecto importante en el modelado de un sistema de recuperación de información.

Todo el despliegue de la metodología y la evaluación se desarrolla en las siguientes secciones.

4.1 Adaptación de ontologías existentes

Dado que cada ontología existente define sus conceptos de forma organizada y estructurada, el proceso de adaptación es presentado por separado para cada una de ellas; facilitando así el desarrollo de las fases de la metodología. Antes de explicar el proceso de adaptación, cabe indicar las *Representaciones Intermedias* (RI) que aporta *Methontology* a la metodología, específicamente, en las fases B.1 y B.3:

- *Diccionario de conceptos*, el cual contiene todos los conceptos del dominio, sus instancias, clases y atributos de las instancias y, opcionalmente, sinónimos y acrónimos de los conceptos.
- *Diagramas de relaciones binarios ad hoc*, con los cuales se establecen relaciones entre conceptos de la misma o de diferentes ontologías. Además, establecerán las directrices para la integración de las ontologías.
- *Tabla de atributos de instancias* para cada atributo de instancia que aparece en el *diccionario de conceptos*. Estos son atributos definidos en los conceptos pero que toman valores en sus instancias.
- *Tablas de instancias* por cada instancia que aparece en el *diccionario de conceptos*, cuyos atributos son conocidos.

Cabe mencionar que el orden anterior de las RI no representa un flujo de trabajo. Desde el punto de vista metodológico, éstas han sido utilizadas para identificar los conceptos, instancias, atributos y relaciones que formaliza cada una de las ontologías existentes.

Teniendo en consideración lo anterior, en las siguientes secciones se presenta el resultado de la adaptación de cada ontología existente, el cual, según la metodología, pasa por identificar sus RI, cotejarla con el esquema de metadatos propuesto en el Capítulo 3 para identificar la nueva información que debe ser incorporada, determinar las RI adaptadas que resultan de incorporar la nueva información, determinar el modelo ontológico resultante desde el punto de vista de la implementación, y determinar, si es posible, un modelo de inferencias.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

4.1.1 Adaptación de “*Learning Object Metadata Ontology*”

Analizando el contenido de Youtube *Teorema de Pitágoras súper fácil* se puede ver que éste, a pesar de no estar definido propiamente como un OA, tiene las características de uno de ellos. Por ejemplo, puede estar dirigido a un rol específico (docente o estudiante), su contenido puede estar asociado a un tiempo y edad típicos de aprendizaje, entre otros. Esta información puede ser útil para recuperar contenidos de acuerdo a aspectos propios de la educación, por lo cual, al introducir atributos propios de los OA a los contenidos de VoD a nivel ontológico, se aumentan las posibilidades para definir estrategias de recomendación basadas en técnicas semánticas. Por ejemplo, si el SR identifica un contenido con las características necesarias para apoyar cierta competencia, el contenido podría ser recomendado o no, tomando como criterio la edad del estudiante (por ejemplo, por la dificultad del lenguaje empleado en el contenido, conviene tener en cuenta la edad). Por otra parte, considerando que la extensión del IEEE RCD para la descripción de competencias propuesta en el Capítulo 3 incorpora elementos tomados de LOM, se pueden establecer vínculos entre las competencias y los OA, por lo cual, una ontología como “*Learning Object Metadata Ontology*” permitiría desarrollar estrategias de recomendación no sólo de contenidos de VoD, sino de otros OA como cursos, textos o páginas web.

Con el fin de aprovechar estas ventajas, en la investigación se justifica el uso de “*Learning Object Metadata Ontology*”, para lo cual, es importante que ésta sea capaz de formalizar toda la información del esquema LOM a través del proceso de adaptación explicado en las siguientes fases de la metodología.

Fase B.1: identificar las RI de las ontologías existentes

Inicialmente, se determinó el conocimiento formalizado en la ontología “*Learning Object Metadata Ontology*”. Para esto, se identificó su *diccionario de conceptos*, el cual se presenta en Tabla 4.1, con elementos que son de interés para este trabajo (esto es, conceptos relacionados con las características del esquema de metadatos propuesto en el Capítulo 3).

Con el propósito de establecer las relaciones entre conceptos, se identificaron las *relaciones binarias* existentes de acuerdo con las relaciones del *diccionario de conceptos anterior*. Tales relaciones se muestran en la Tabla 4.2.

Nombre del concepto	Instancias	Atributos de instancias	Relaciones del concepto
Learning-object	-	-	Has-description
Educational	-	Context Learning-resource-type Interactivity-type Intended-end-user-role Language Difficulty	Is-description-of Has-learning-time
Duration	-	Value	Is-learning-time-of

Tabla 4.1. Diccionario de conceptos de “*Learning Object Metadata Ontology*”. Fuente: propia.

Nombre de la relación	Concepto fuente	Concepto objetivo	Relación inversa
Has-description	Learning-object	Educational	Is-description-of
Has-learning-time	Educational	Duration	Is-learning-time-of

Tabla 4.2. Relaciones binarias de “*Learning Object Metadata Ontology*”. Fuente: propia.

La Tabla 4.3 presenta la *tabla de atributos* de las instancias del *diccionario de conceptos*. Esta ayuda a entender los atributos a través de su tipo de valor, unidad de medida, rango de valores, valor por defecto y cardinalidad.

Nombre del atributo de instancias	Tipo de valor	Unidad de medida	Rango de valores	Valor por defecto	Cardinalidad
Value	Time	Hours, minutes, seconds	-	-	(1,1)
Context	String	-	{ higher education, other, school, training}	-	-
Learning-resource-type	String	-		-	(1,1)
Interactivity-type	String	-	{active, expositive, mixed}	-	(1,1)
Intended-end-user-role	String	-	{ author, learner, manager, teacher}	-	(1,1)
Language	String	-	-	-	(1,1)
Difficulty	String	-	difficult, easy, medium, very difficult, very easy	-	(1,1)

Tabla 4.3. Tabla de atributos de las instancias de “*Learning Object Metadata Ontology*”. Fuente: propia.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

Una representación gráfica de la información mostrada en las anteriores tablas es la *red semántica* de la Figura 4.2, en la cual se observan los conceptos, sus atributos (indicados por medio de la relación *Has*) y relaciones.

Fase B.2: cotejar ontología existente-esquema de metadatos

Aquí se realizó la identificación de los conceptos, instancias, atributos y relaciones que hacen falta para que “*Learning Object Metadata Ontology*” cubra toda la información que describe LOM. Tal ontología define una gran cantidad de propiedades que cubren sintáctica y semánticamente todos los componentes de LOM. Sin embargo, fue conveniente evaluar si todos sus elementos están representados en la ontología. La Tabla 4.4 presenta, parcialmente, cómo está representado LOM, indicando los elementos del esquema de metadatos de interés, qué elementos de la ontología están vinculados a ellos, a qué concepto están relacionados, el tipo de elemento, su dominio y rango. El resultado completo se puede consultar en la Tabla G.2, Anexo G.

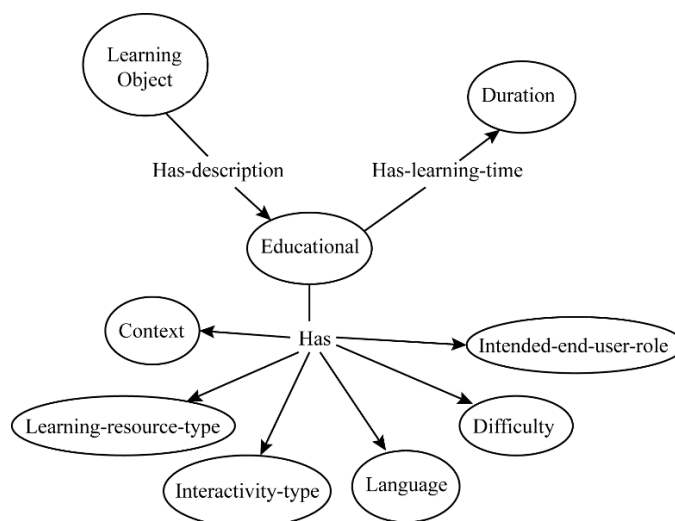


Figura 4.2 Red semántica identificada de “*Learning Object Metadata Ontology*”. Fuente: propia.

Elemento LOM del esquema	Elemento de la ontología	A qué concepto está vinculado el elemento	Tipo de elemento	Dominio	Rango
Context	educationalContext	Educational	Data Property	Educational	<ul style="list-style-type: none"> • xsd:short • {"higher education"^^xsd:string, "other"^^xsd:string, "school"^^xsd:string, "training"^^xsd:string} • rdfs:Literal

Learning Resource Type	-	-	-	-	-
Interactivity Type	educationalInteractivityType	Educational	Data Property	Educational	<ul style="list-style-type: none"> • {"active"^^xsd:string, "expositive"^^xsd:string, "mixed"^^xsd:string}

Tabla 4.4. Fragmento de la relación entre LOM y “*Learning Object Metadata Ontology*”.

Fuente: propia.

Sobre la información presentada en la Tabla 4.4, se realizó posteriormente un análisis para identificar qué elementos de LOM hace falta formalizar. Por ejemplo, el elemento *Learning Resource Type* no está representado en la ontología. Sobre este elemento es importante recordar que LOM define un vocabulario para él. En general, en “*Learning Object Metadata Ontology*” los elementos cuyos valores obedecen a un vocabulario (esto es, los elementos *Context*, *Interactivity Type*, *Intended End User Role*, *Language* y *Difficulty*) están definidos como expresiones de rango para “*Data Properties*”, por lo cual su valor puede ser ingresado en texto libre, dando lugar a errores tipográficos en la práctica. Por ejemplo, teniendo en cuenta que LOM establece a *student* como uno de los posibles valores para el elemento *Intended End User Role*, en la práctica se puede introducir un error al definir *estudent* (la letra *e* está de más). Esto puede perjudicar la recuperación del registro en el cual se introdujo el error.

Por otra parte, en el Capítulo 3 se mencionó que existen unos elementos comunes entre los objetos de aprendizaje y las competencias que establecen relaciones entre ambas entidades. Éstos son: *Language*, *Intended User Role* y *Difficulty*. Además, la ontología (y LOM en realidad) define vocabularios que no concuerdan totalmente con las descripciones de las competencias. Éste es el caso del elemento *Language*. Específicamente, la ontología define un vocabulario éste elemento, mientras que para la descripción de competencias se propuso el uso de los códigos de dos letras del iso 639-1.

Fase B.3: determinar las RI adaptadas

Para abordar las situaciones planteadas en la fase B.2, se crearon conceptos adicionales para representar los elementos del esquema de metadatos que tienen vocabularios, con instancias que representan a cada una de las opciones de éstos. Esta idea es tomada de [62], en donde se expone que así se posibilita el establecimiento de *asociaciones semánticas complejas* (Anexo A, Sección A.3) entre instancias, a partir de las cuales se puede inferir conocimiento usando técnicas de

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

ponderación (Sección A.3.3). De acuerdo a lo anterior, se identificaron nuevos conceptos, instancias, atributos y relaciones, los cuales fueron adaptados al modelo original de “*Learning Object Metadata Ontology*”.

La Tabla 4.5 presenta el nuevo *diccionario de conceptos*, en el cual los atributos de instancias originales del concepto *Educational* han sido reemplazados por los nuevos conceptos y sus instancias, siendo necesaria la determinación de nuevas relaciones (los nuevos elementos ontológicos aportados están indicados en **negrita**).

La Tabla 4.6 presenta las relaciones binarias existentes de acuerdo con las relaciones del anterior *diccionario de conceptos*.

Como resultado del proceso llevado a cabo hasta aquí, se construyó la red semántica de la Figura 4.3, la cual señala las diferentes relaciones entre conceptos del nuevo modelo ontológico adaptado de “*Learning Object Metadata Ontology*”. En gris se señalan los nuevos conceptos y atributos aportados (indicados por la relación *Has*), y en *negrita-cursiva* las nuevas relaciones aportadas entre conceptos.

Nombre del concepto	Instancias	Atributos de instancias	Relaciones del concepto
Learning Object	-	-	Has-description
Educational	-	-	Is-description-of Has-learning-time Has-context Has-resource-type Has-interactivity-type Has-user Has-language Has-difficulty Has-user-age
Duration	-	Value	Is-learning-time-of
Context	Higher education, School, Training, Other	Context-value	Is-context-of
Type	Diagram, Exam, Exercise, Experiment, Figure, Graph, Index, Lecture, Narrative Text, Problem Statement, Questionnaire, Self-Assessment, Simulation, Slide, Table, Video	Type-value	Is-resource-type-of

Interactivity	Active, Expositive, Mixed, Undefined	Interactivity-value	Is-interactivity-type-of
User	Author, Learner, Manager, Teacher	User-value	Is-user-of
Language	Instancias acordes con la descripción de competencias: código de dos letras del ISO 639-1	Language-value	Is-language-of
Difficulty	Difficult, Easy, Medium difficulty, Very difficult, Very easy	Difficulty-value	Is-difficulty-of

Tabla 4.5. Diccionario de conceptos para la adaptación de “*Learning Object Metadata Ontology*”. Fuente: propia.

La Tabla 4.7 corresponde a la tabla de atributos de instancias del *diccionario de conceptos* final.

Nombre de la relación	Concepto fuente	Concepto objetivo	Relación inversa
Has-description	Learning-object	Educational	Is-description-of
Has-learning-time	Educational	Duration	Is-learning-time-of
Has-context	Educational	Context	Is-context-of
Has-type	Educational	Type	Is-resource-type-of
Has-interactivity	Educational	Interactivity	Is-interactivity-type-of
Has-user	Educational	User	Is-user-of
Has-language	Educational	Language	Is-language-of
Has-difficulty	Educational	Difficulty	Is-difficulty-of
Has-age	Educational	Age	Is-user-age-of

Tabla 4.6. Relaciones binarias para la adaptación de “*Learning Object Metadata Ontology*”. Fuente: propia.

La Tabla 4.8 presenta un fragmento de la *tabla de instancias* que aparecen en el *diccionario de conceptos* después de la adaptación. La *tabla de instancias* completa se puede consultar en la Tabla G.1, Anexo G (Sección G.1). Como se puede observar, el valor de los atributos es igual al nombre la instancia, con lo cual, se puede hacer referencia a las instancias consultando su nombre o el valor de sus atributos.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

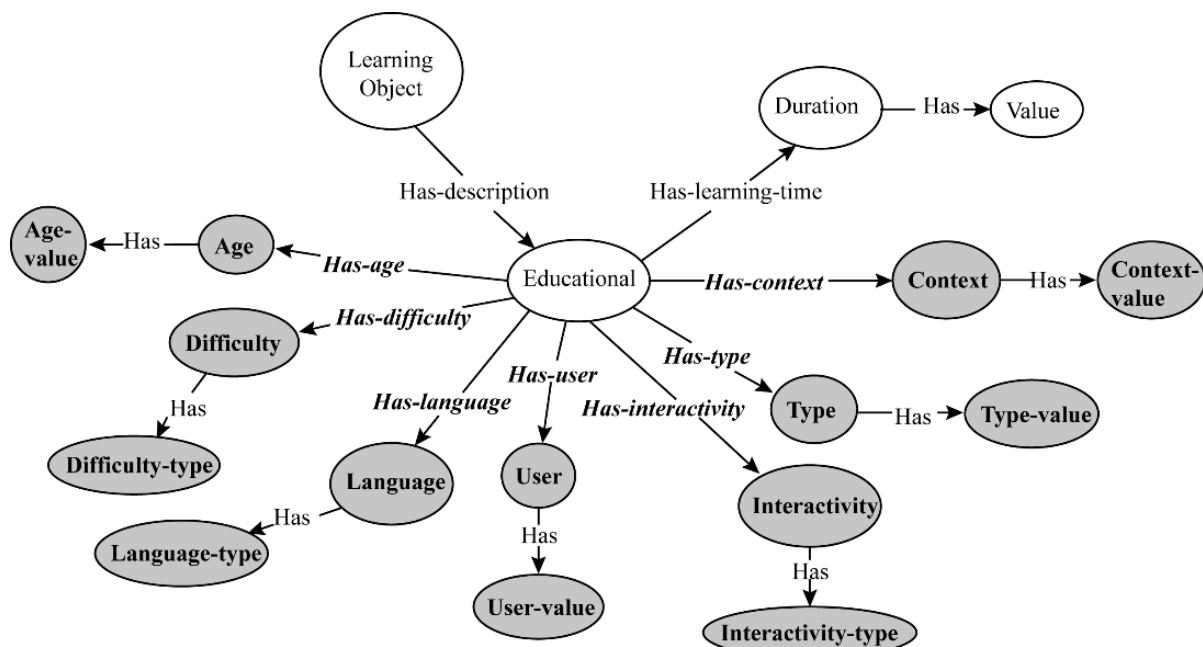


Figura 4.3 Red semántica de la adaptación de "Learning Object Metadata Ontology".

Fuente: propia.

Nombre del atributo de instancias	Tipo de valor	Unidad de medida	Rango de valores	Valor por defecto	Cardinalidad
Value	Decimal	Hours, minutes, seconds	-	-	(1,1)
Context-value	String	-	-	-	
Type-value	String	-	-	-	(1,1)
Interactivity-value	String	-	-	-	(1,1)
User-value	String	-	-	-	(1,1)
Language-value	String	-	-	-	(1,1)
Difficulty-value	String	-	-	-	(1,1)
Age-value	String	-	-	-	(1,1)

Tabla 4.7. Tabla de atributos de instancias para la adaptación de "Learning Object Metadata Ontology". Fuente: propia.

Nombre de la instancia	Atributos con valores conocidos	Valor de los atributos
Higher education	Context-value	Higher education
School	Context-value	School
Training	Context-value	Training
Other	Context-value	Other
Diagram	Type-value	Diagram
Exam	Type-value	Exam
Exercise	Type-value	Exercise
Experiment	Type-value	Experiment
Figure	Type-value	Figure
Graph	Type-value	Graph
Index	Type-value	Index
Lecture	Type-value	Lecture
Narrative	Type-value	Narrative
Text	Type-value	Text
Problem	Type-value	Problem
Statement	Type-value	Statement
Questionnaire	Type-value	Questionnaire
Self-Assessment	Type-value	Self-Assessment
Simulation	Type-value	Simulation
Slide	Type-value	Slide
Table	Type-value	Table

Tabla 4.8. Fragmento de la tabla de instancias para la adaptación de “*Learning Object Metadata Ontology*”. Fuente: propia.

Fase B.4: determinar el modelo ontológico

Finalmente, de acuerdo con los resultados del proceso desarrollado, se estableció el modelo ontológico definitivo adaptado de “*Learning Object Metadata Ontology*” desde el punto de vista de la implementación, el cual se describe parcialmente en la Tabla 4.9. En negrita se señalan los elementos ontológicos aportados a la ontología original. El modelo completo puede consultarse en la Tabla G.3, Anexo G (Sección G.3).

Elemento de la ontología	Tipo de elemento	Instancias	Dominio	Rango	Propiedades inversas
LearningObject	Class	-	-	-	-
Educational	Class	-	-	-	-
Duration	Class	-	-	-	-

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

Context	Class	Higher education, School, Training, Other	-	-	-
UserRole	Class	Author, Learner, Manager, Teacher	-	-	-
Difficulty	Class	Difficult, Easy, Medium difficulty, Very difficult, Very easy	-	-	-
hasEducational	Data Property	-	Duration	xsd:decimal	-
ageValue	Data Property	-	Duration	xs:int	-
hasEducational Context	Object Property	-	Educational	Context	isEducationalContextOf
hasEducational ResourceType	Object Property	-	Educational	ResourceType	isEducationalResourceTypeOf
hasEducational User	Object Property	-	Educational	UserRole	isEducationalUserOf

Tabla 4.9. Fragmento del modelo ontológico final para la adaptación de “*Learning Object Metadata Ontology*”. Fuente: propia.

Ejemplo 4.1. Se busca modelar el contenido de Youtube *Teorema de Pitágoras súper fácil* (tratado en el ejemplo 3.1) desde el punto de vista de los OA para un curso de educación superior. Específicamente, se busca definir que el contenido es: 1) aplicado en un curso de educación superior; 2) de tipo video; 3) de tipo expositivo, ya que no permite la interactividad con los estudiantes; 4) dirigido a los estudiantes; 5) se desarrollan conceptos de dificultad media; y 6) explicado en español. Toda esta información puede ser modelada a través de la adaptación de “*Learning Metadata Ontology*” propuesta. Con su aplicación, se obtiene una base de conocimiento como la mostrada en la Figura 4.4. Se observa cómo el contenido se define como una instancia de la clase *LearnignObject* (propiedad [1]), indicando así que es un OA. También, se ve cómo el contenido es descrito por propiedades de objetos vinculadas a instancias con significado dentro del contexto ([2], [3], ..., [8]). Por ejemplo, el contenido es un OA con interactividad expositiva, para lo cual se hace uso de la propiedad `:hasEducationalInteractivityType`, vinculada a la instancia *Expositive*. Es importante notar que los contenidos son descritos por medio de propiedades de objetos e instancias, lo cual, a la luz de la web semántica, favorece el establecimiento de asociaciones semánticas complejas (ver los ejemplos mostrados en la Sección A.3 en cuanto a la definición de secuencias de propiedades y descubrimiento de asociaciones semánticas).

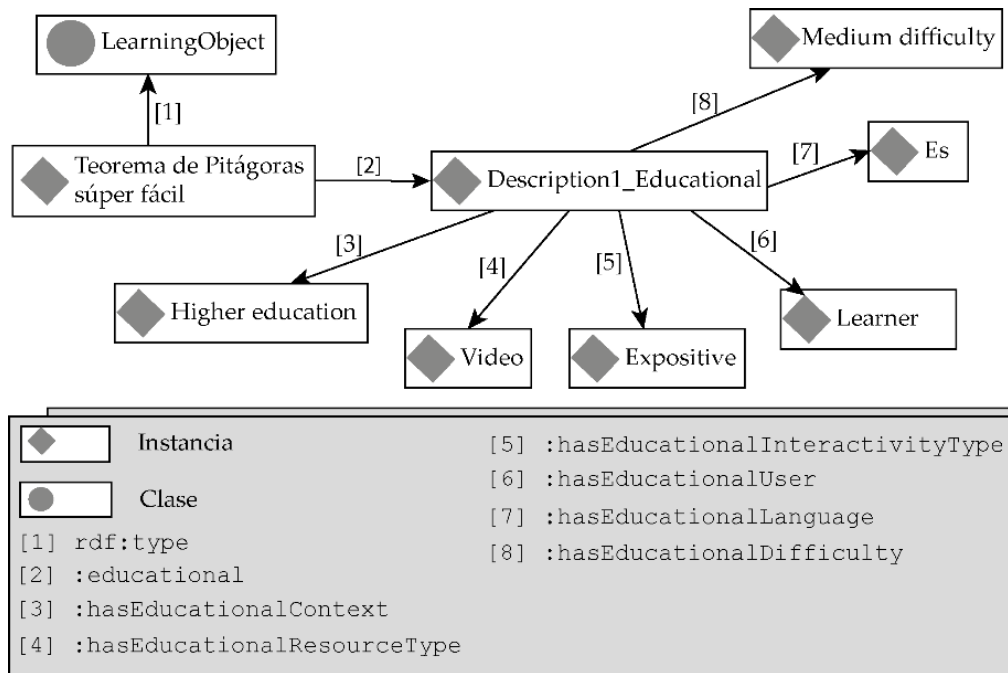


Figura 4.4 Ejemplo de aplicación de la adaptación de “*Learning Object Metadata Ontology*”. Fuente: propia.

Esto implica que si dos o más contenidos comparten propiedades con estas características, es posible descubrir asociaciones entre ellas, lo cual representa conocimiento oculto que puede ser explotado en un sistema de recuperación de información tal como un SR.

4.1.2 Adaptación de “*Ontology for Media Resources*”

En la introducción del capítulo se justificó la necesidad de adaptar “*Ontology for Media Resources*” para inferir conocimiento sobre los vínculos contenido-segmento de video, contenido-competencias y segmento de video-competencias. Esto es conveniente en situaciones en las que hay competencias asociadas a segmentos de video. En caso de que un estudiante tenga necesidades educativas en tales competencias, el SR podría identificar el contenido a recomendar analizando a qué contenido pertenece el segmento de video.

Con respecto a “*Ontology for Media Resources*”, ésta es una ontología que define conceptos comunes entre diferentes tipos de recursos multimedia tales como: video, audio, imagen, texto. De acuerdo con la W3C [102], su intención es vincular las diferentes descripciones de los recursos. Para lograrlo, la especificación describe una ontología genérica (principalmente, las propiedades descriptivas de los recursos) y define un grupo de directrices para realizar su mapeo a esquemas de metadatos que actualmente describen recursos publicados en la web, tales como *Dublin Core*, *MPEG-7*, *TV-Anytime*, entre otros. Precisamente, esto favorece el proceso de adaptación para

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

modelar toda la información definida en el *perfil de aplicación de TV-Anytime* propuesto en el Capítulo 3.

Dado que para la adaptación de “*Ontology for Media Resources*” se realizó el mismo proceso que en el caso de “*Learning Object Metadata Ontology*”, en esta sección únicamente se presenta la *red semántica* y las *inferencias* obtenidas como resultado. Éstos resultados son importantes porque permiten identificar los atributos de los contenidos de VoD y los segmentos de video, así como las relaciones lógicas entre ambas entidades, lo cual juega un papel importante en el desarrollo de la estrategia de recomendación propuesta. El proceso completo de adaptación de la ontología, incluidas todas las fases, se puede consultar en el Anexo H.

La red semántica de la Figura 4.5 señala las diferentes relaciones entre conceptos del nuevo modelo ontológico adaptado de “*Ontology for Media Resources*”. En gris se señalan los aportes a la ontología original en cuanto a conceptos y atributos (señalados por la relación *Has*) y en *negrita-cursiva* en cuanto a relaciones.

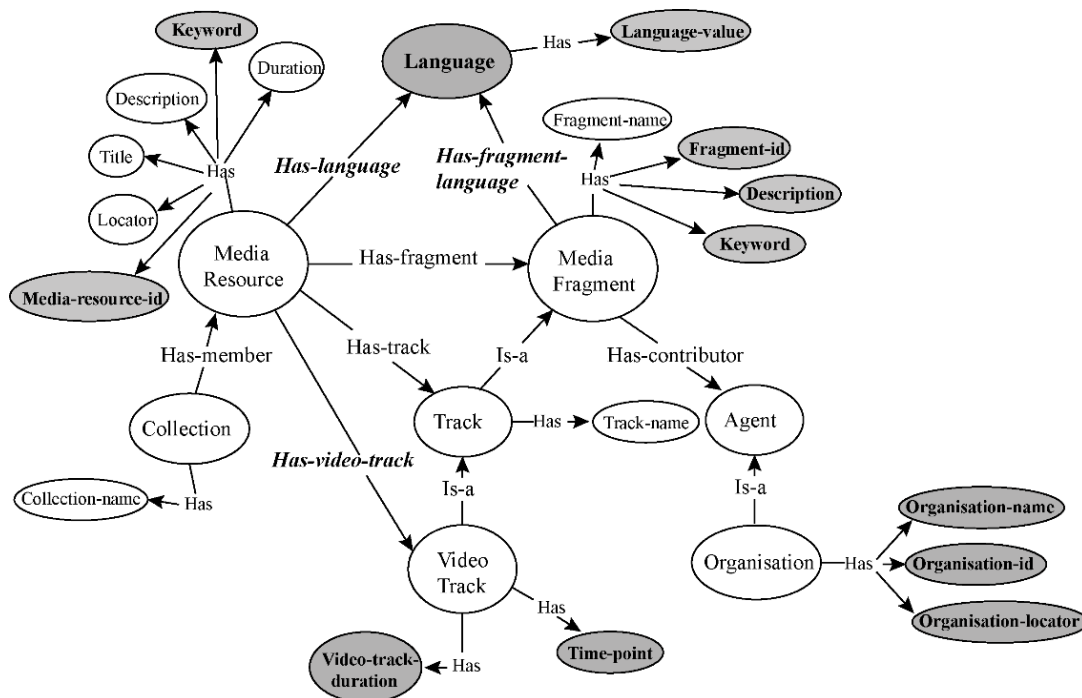


Figura 4.5 Red semántica para la adaptación de “*Ontology for Media Resources*”. Fuente: propia.

Por otra parte, se vio la posibilidad de realizar algunas inferencias con respecto a los segmentos que hacen parte de los contenidos de VoD. Suponiendo que hay una instancia llamada *MR1* de la clase *MediaResource* que representa un contenido, y que está vinculado a la instancia *Track1* de la clase *videoTrack* por medio de la Object Property `:hasVideoTrack`. Entonces se definen las tripletas:

`:MR1` `hasVideoTrack` `:Track1`.

Si se establece que `:hasVideoTrack` es subpropiedad de `:hasTrack`, y éste también lo es de `:hasFragment` de la siguiente forma:

`:hasVideoTrack` `rdfs:subPropertyOf` `:hasTrack`,

`:hasTrack` `rdfs:subPropertyOf` `:hasFragment` (esta tripleta está ya definida en la ontología original). Entonces se infiere que:

`:MR1` `hasFragment` `:Track1`.

Este proceso se ilustra en la Figura 4.6.

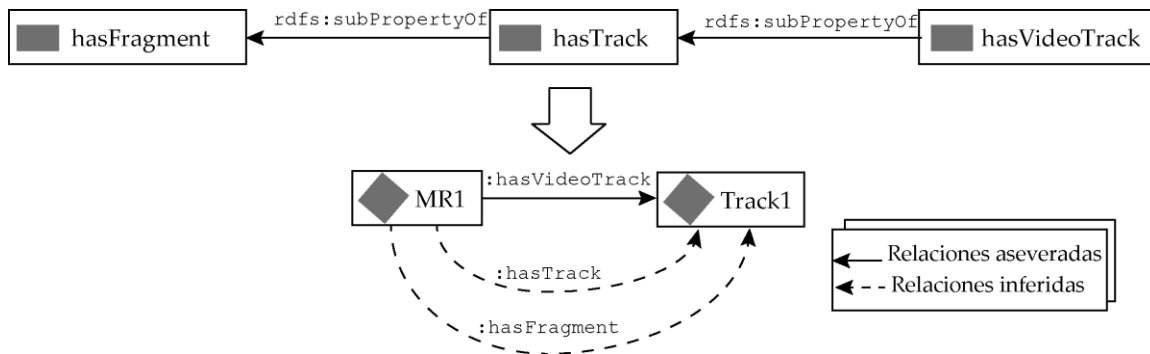


Figura 4.6 Ejemplo de inferencia entre los contenidos de VoD y los segmentos de video.

Fuente: propia.

Ejemplo 4.2. Después de un análisis del contenido de Youtube *Teorema de Pitágoras súper fácil*, se puede verificar que éste se puede fragmentar en dos segmentos de video: *El Teorema de Pitágoras* y *El triángulo rectángulo*. Además, de Youtube se ha extraído información (a partir de los metadatos) del contenido que corresponde a su título, duración, identificador, palabras clave, URL, descripción en texto libre y lenguaje. Esta misma información se extrajo para los segmentos a partir de su análisis.

Al formalizar toda la información, tanto de contenidos como de segmentos, a través de la adaptación de “*Ontology for Media Resources*” se obtiene una base de conocimiento como la mostrada en la Figura 4.7. En ella, la relación entre el contenido y los segmentos queda definida a través de la propiedad `:hasVideoTrack`. La información descriptiva de contenidos y segmentos queda establecida a través de otras propiedades. Por ejemplo, el título de contenido se define a través de la propiedad `ma:title5`; mientras que las palabras clave de los segmentos se definen a través de la propiedad `:fragmentKeyword`.

⁵ Las propiedades con prefijo son definidas por ontologías a las que se les asignan prefijos para su publicación en la web. En este caso, `ma` es un prefijo asignado a la ontología “*Ontology for Media Resources*”.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

Por otra parte, se puede ver que se hace explícito que el contenido es tomado de Youtube, esto haciendo uso de la propiedad `ma:hasContributor`. Finalmente, cabe destacar que al comparar la Figura 4.7 con la Figura 3.2, se observa que se ha logrado una correlación entre el modelo ontológico y el modelo descriptivo del perfil de aplicación de *TV-Anytime* propuesto.

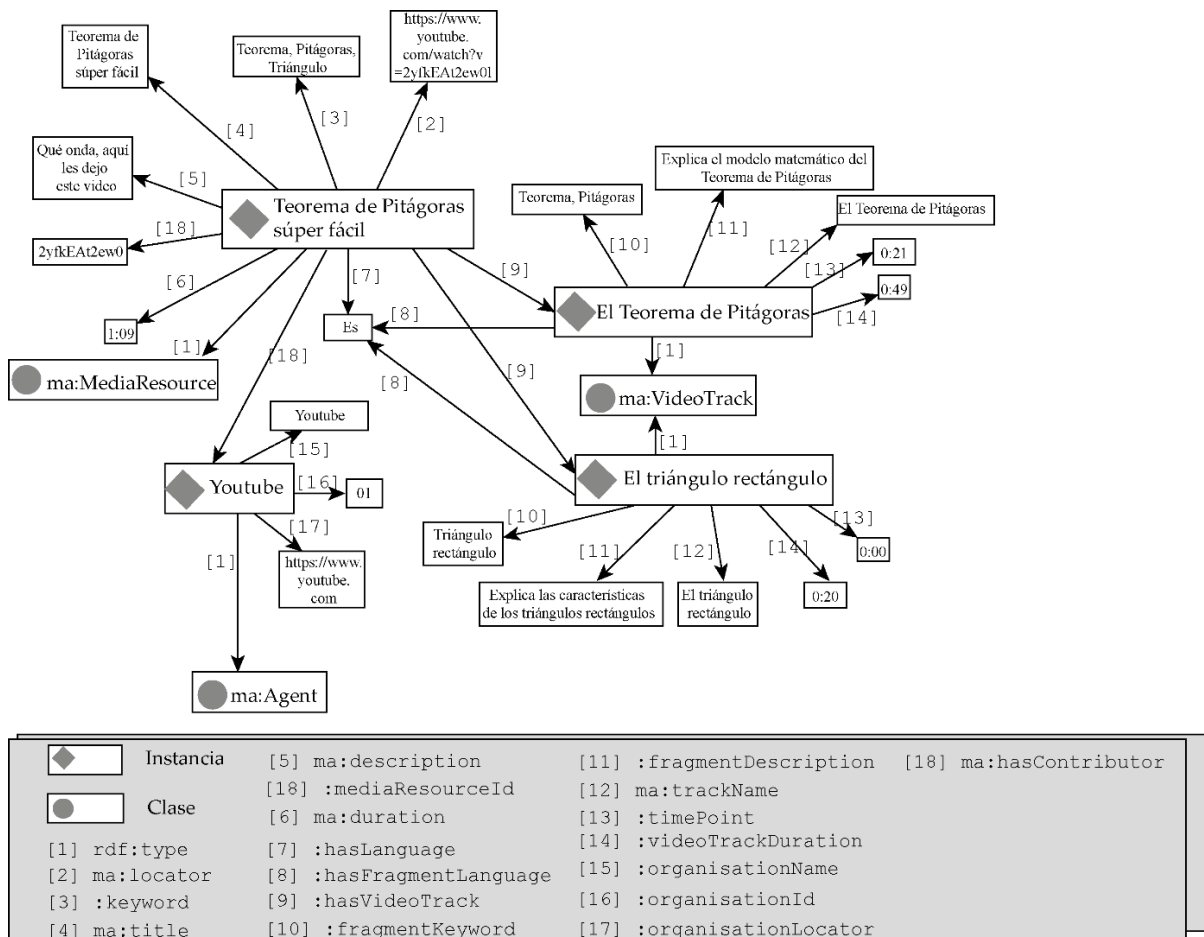


Figura 4.7 Ejemplo de aplicación de “Ontology for Media resources”. Fuente: propia.

4.1.3 Adaptación de “Competency Model Ontology”

La necesidad de adaptar esta ontología en el presente trabajo se basa en la posibilidad de proveer al SR conocimiento acerca de los vínculos entre competencias. Por ejemplo, al considerar dos competencias: *Reconozco las características de los triángulos rectángulos* y *Reconozco las operaciones involucradas en el Teorema de Pitágoras*; se puede ver cómo comparten el mismo verbo *reconocer*. Así, entre las dos competencias existe una *asociación semántica compleja* construida a través de tal atributo vinculador. Según [62], esta asociación puede ser ponderada a través de métricas semánticas y estadísticas (Anexo A, Sección A.3), a partir de lo cual se puede

inferir la relevancia de una competencia con respecto a la otra. En el marco de los SR, esta posibilidad podría ser utilizada para recomendar contenidos asociados a las competencias semánticamente relevantes para otras en las que el estudiante tiene necesidades educativas.

Con el ánimo de aprovechar estas posibilidades, se hace necesaria la adaptación de “*Competency Model Ontology*” de acuerdo con la extensión del IEEE RCD propuesta en el Capítulo 3. Lo anterior teniendo en cuenta que ésta incluye una serie de atributos que, formalizados en la ontología, permiten la identificación de *asociaciones semánticas complejas* entre competencias. Desarrollando el mismo proceso de adaptación ejecutado para las anteriores ontologías, se obtuvo la *red semántica* de la Figura 4.8, la cual señala las diferentes relaciones entre conceptos del nuevo modelo ontológico adaptado de “*Ontology for Media Resources*” así como sus atributos. Éstos son aprovechados por la estrategia de recomendación propuesta más adelante. En gris se señalan los aportes a la ontología original en cuanto a conceptos y atributos (señalados por la relación *Has*) y en negrita-cursiva en cuanto a relaciones.

Ejemplo 4.3. Después de un análisis del contenido de Youtube *Teorema de Pitágoras súper fácil*, se identificaron dos competencias desarrolladas en él: *Reconozco las características de los triángulos rectángulos* y *Reconozco las operaciones involucradas en el Teorema de Pitágoras*. Tal como se indicó en el Ejemplo 3.2 (Sección 3.2), de éstas se extrajo información descriptiva como: su categoría, verbo de acción, rol de usuario, entre otros. Al aplicar la adaptación de “*Competency Model Ontology*” propuesta se obtiene la base de conocimiento mostrada en la Figura 4.9. Como se puede observar, los atributos de las competencias han tomado la forma de “*object properties*” tales como `:hasCompetencyType` (categoría de la competencia), `:hasActionVerb` (verbo de acción) y `:hasIntendedUser` (rol de usuario), cuyos valores son definidos a través de instancias, por ejemplo, *knowledge*, *reconocer*, *estudiante*. Teniendo en cuenta lo establecido en la Sección 2.1.4, estas características favorecen el establecimiento de *asociaciones semánticas complejas*. Específicamente, dado que las dos competencias están vinculadas de diferentes formas, por ejemplo, a través de la propiedad `:hascompetencyType` y la instancia *Knowledge*, se puede concluir que existe una asociación semántica entre ambas. Finalmente, cabe destacar que al comparar la Figura 4.9 con la Figura 3.4, se observa que se ha logrado una correlación entre el modelo ontológico y el modelo descriptivo de la extensión del IEEE RCD propuesto para la descripción de competencias.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

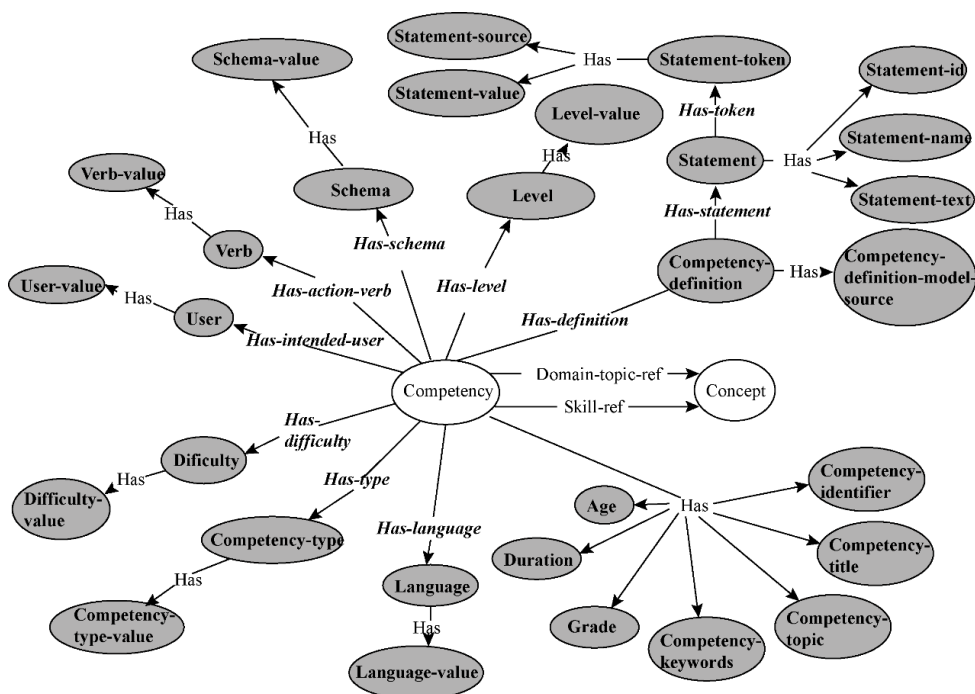


Figura 4.8 Red semántica de la adaptación de “Competency Model Ontology”. Fuente: propia.

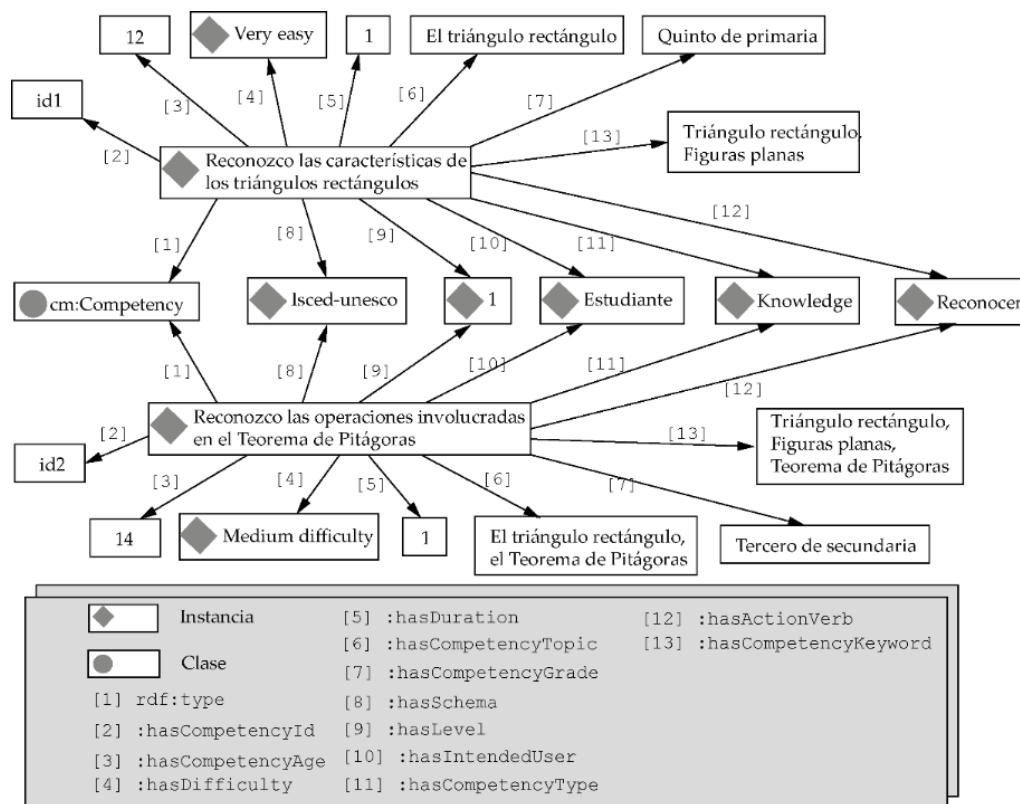


Figura 4.9 Ejemplo de aplicación de la adaptación de “Competency Model Ontology”. Fuente: propia.

4.2 OntoRCM: una ontología para el modelado de mapas de competencias

En los trabajos revisados sobre ontologías existentes (Sección 2.2.4) no fue posible hallar una útil para representar el conocimiento estructurado en los mapas de competencias. Por tal razón, fue necesario construir una ontología que permita introducir el conocimiento involucrado en tales entidades dentro del proceso de recomendación. Como se indicó al inicio del presente capítulo, las relaciones de subsunción existentes en los mapas pueden ser aprovechadas para inferir, por ejemplo, los contenidos asociados a las competencias que apoyan el aprendizaje de otras; lo cual representa conocimiento claramente aprovechable en un proceso de recomendación. Es así como se justifica en el presente trabajo la construcción de una nueva ontología para el modelado de mapas de competencias.

Siguiendo el mismo criterio considerado hasta ahora, se busca construir la ontología a partir de esquemas de metadatos. Para tal fin, se retoma la decisión justificada en el estado del arte (Sección 2.2.2) de usar el esquema SRCM [38].

Como paso inicial en el proceso de creación de la ontología, se presenta a continuación un grupo aspectos resultantes del análisis de los componentes y estructura de los mapas de competencias de SRCM con respecto a las características propias de las ontologías:

- Los elementos más importantes de los mapas de competencias son nodos. Esto, acorde con la estructura de los Grafos Acíclicos Dirigidos (GAD). En consecuencia, su representación en la ontología estaría dada por un solo concepto y sus instancias, eliminando la posibilidad de crear unos conceptos para representar competencias generales, otros para competencias específicas y relacionarlos con alguna propiedad.
- Tal como se plantea en [6], hay una necesidad de incorporar a los mapas las *Actividades de Evaluación* (AE) (por ejemplo, evaluaciones escritas y orales, talleres, mesas redondas, entre otras). Esto con el propósito establecer un método computacional centrado en la evaluación lingüística de competencias a partir de las calificaciones alcanzadas en la AE. Por esta razón, los nodos de los grafos pueden estar vinculados a competencias y actividades de evaluación. Además, los nodos de SRCM hacen referencia a descripciones de competencias en IEEE RCD, pero no hay nada formal en cuanto a los vínculos con las AE. Por esta razón, las AE son conceptualizadas en esta ontología.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

- La estructuración de las relaciones entre competencias generales y específicas no es un proceso rígido (una competencia no es general o específica siempre para las mismas competencias), por el contrario, es un proceso flexible en el que cada competencia puede tener diferentes roles (generales o específicos) en diferentes mapas de acuerdo a las necesidades. Entonces, la construcción de los mapas está marcada por la subjetividad y, por lo tanto, está basada en la representación de conocimiento semiformal.
- En el contexto de cada mapa de competencias, las relaciones generales-específicas son definidas de nodo a nodo y en un solo sentido (A más general que B, o B más específica que A). Sin embargo, hay relaciones inversas y transitivas que pueden ser inferidas en la ontología y útiles en cualquier aplicación.

Con toda la información alrededor de los mapas de competencias y desarrollando la metodología introducida al inicio del capítulo, se construyó la nueva ontología, denominada *OntoRCM*, siendo esta un aporte de la presente investigación. La presente sección muestra resumidamente el proceso llevado a cabo. Una versión extendida de éste se puede consultar en el Anexo J.

Como resultado del proceso se obtuvo la *red semántica* de la Figura 4.10, la cual ilustra los conceptos, sus atributos y relaciones. Éstos son posteriormente aprovechados en la estrategia de recomendación para inferir conocimiento a partir de las relaciones de subsunción de los mapas.

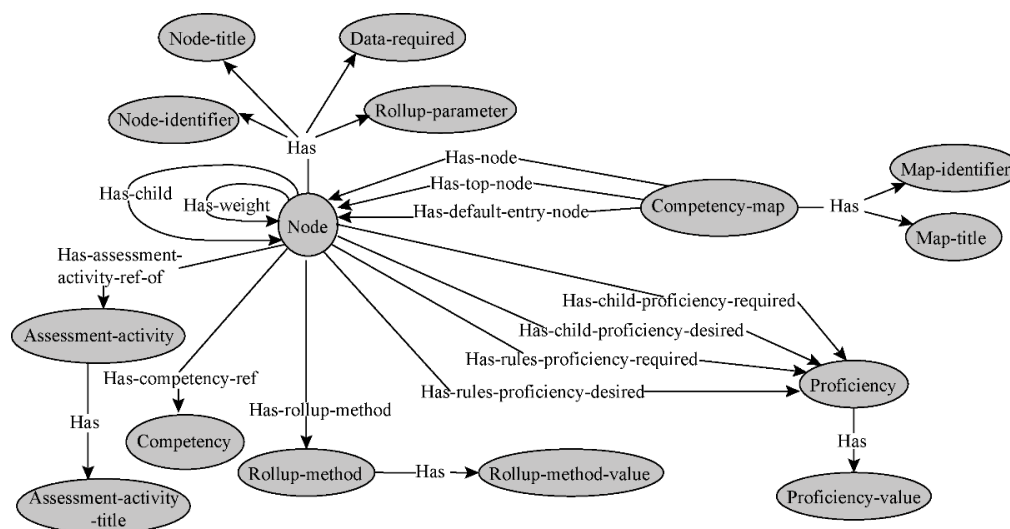


Figura 4.10 Red semántica para *OntoRCM*. Fuente: propia.

Desde el punto de vista técnico, para representar la red semántica anterior se reutilizó del modelo de “*Simple Knowledge Organisation System*” (SKOS) [103], el cual define clases, propiedades y atributos que sintáctica y semánticamente tienen semejanzas

con los conceptos, atributos y relaciones del modelo de los mapas de competencias conseguido. Otra razón por la cual usar SKOS es que las relaciones existentes entre las competencias representan conocimiento semiformal en el marco de los mapas de competencias.

En concreto, SKOS define las propiedades `skos:Broader` y `skos:Narrower`, las cuales son usadas para la representación de vínculo jerárquicos, como la relación entre un todo y sus partes. `skos:Broader` es usada para establecer que un concepto es más amplio (o general) que otro, mientras que `skos:Narrower` sirve para definir que es más angosto (o específico). Con respecto a los mapas, estas propiedades están estrechamente relacionadas con las relaciones generales y específicas de las competencias. Además, define una estructura de relaciones semánticas (que incluye a `skos:Broader` y `skos:Narrower`) que permitiría obtener inferencias sobre las relaciones en los mapas.

Como se ha mencionado anteriormente, los mapas de competencias involucran relaciones de subsunción que vale la pena inferir. Después de un análisis de las posibilidades de inferencia sobre los mapas, se identificaron tres posibles escenarios así: *inferencia de competencias generales y específicas*; *inferencia de competencias de las cuales depende una en particular*; e *inferencia de las competencias posibilitadas por una específica*.

Con respecto al primer escenario, desde el punto de vista educativo, para la construcción de un mapa de competencias se consideran las relaciones entre competencias generales y específicas, las cuales pueden ser interpretadas como de dependencia. Entre otros ejemplos, las competencias específicas facilitan el desarrollo de las generales, o son sus prerrequisitos dentro de un plan de estudios. Entonces, podría ser de gran utilidad consultar en la ontología qué competencia es general o específica para una competencia determinada.

Dado que las relaciones `skos:broader` y `skos:narrower` pueden no ser intuitivas para las personas, cuando se trata de establecer competencias generales y específicas, se han definido dos nuevas “*Object Properties*” denominadas `:isGeneralFor` e `:isSpecificFor`. Cada una, subpropiedad de `skos:narrower` y `skos:broader` respectivamente, con lo cual se busca mantener enlazados los datos con la ontología base de SKOS. Además, al igual que `skos:broader` y `skos:narrower`, `:isGeneralFor` e `:isSpecificFor` también se definen como inversas. La Figura 4.11 presenta la estructura de lo realizado para obtener las inferencias buscadas.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

Para ilustrar la situación, se consideran 3 nodos, representados por las instancias *Node1*, *Node2* y *Node3*. En el mapa de competencias, se ha definido que *Node1* es padre de *Node2*, y éste de *Node3*. Entonces, se definen las tripletas:

```
:Node2      :isSpecificFor   :Node1.
```

```
:Node2      :isGeneralFor   :Node3.
```

Como `:isGeneralFor` e `:isSpecificFor` son subcompetencias de `skos:narrower` y `skos:broader` respectivamente, se infiere:

```
:Node2 skos:broader :Node1.
```

```
:Node2 skos:narrower :Node3.
```

Entonces, como `:isGeneralFor` e `:isSpecificFor` son inversas entre sí, se infiere:

```
:Node1 :isGeneralFor :Node2.
```

```
:Node3 :isSpecificFor :Node2.
```

Finalmente, como `skos:broader` y `skos:narrower` son inversas entre sí, se infiere:

```
:Node1 skos:narrower :Node2.
```

```
:Node3 skos:broader :Node2.
```

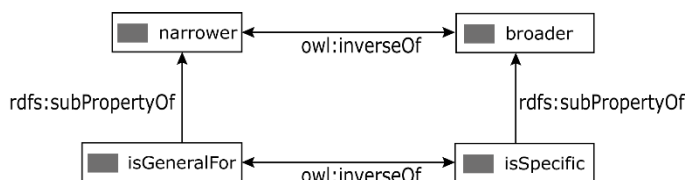


Figura 4.11 Modelo para inferir competencias generales y específicas. Fuente: propia.

Los otros escenarios de inferencia se encuentran descritos en el Anexo J.

Ejemplo 4.4. Se considera el mapa de competencias mostrado en la Figura 4.12. Como se puede observar, el mapa, denominado *Mapa de competencias 1*, está compuesto por cuatro competencias, siendo C1 la más general de todas. Al representar este mapa de acuerdo a la ontología para el modelado de mapas de competencias propuesta (*OntoRCM*) se obtiene el modelo de la Figura 4.13. En éste, el mapa está representado por la instancia *MapaCompetencias1*, la cual es del tipo `skos:ConceptScheme`.

El mapa está compuesto por nodos, los cuales están representados por las instancias *Nodo1*, *Nodo2*, *Nodo3* y *Nodo4*, vinculados al mapa de competencias por la propiedad `:hasNode`. En el ejemplo observa que los nodos están vinculados a las competencias (formalizadas en la adaptación de “*Competency Model Ontology*”) por medio de la propiedad

:hasCompetencyRef. Entonces, cualquier relación entre competencias que quiera consultarse, debe realizarse primero en el ámbito de los nodos y posteriormente en el de las competencias. Tal como se adelantó, esto favorece que en una aplicación real las instancias de las competencias puedan estar asociadas a múltiples mapas a la vez, favoreciendo la reutilización del conocimiento consignado en la base de conocimiento. No así con los nodos, cuyas instancias sólo pueden ser formalizadas como parte de un solo mapa. En cuanto a la estructura de éste, los nodos se inter-relacionan por medio de propiedades como :hasChildVeryHighWeight y :hasChildLowWeight.

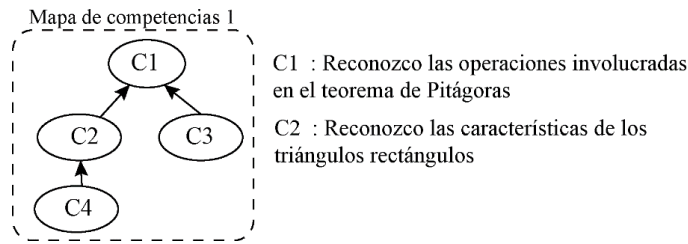
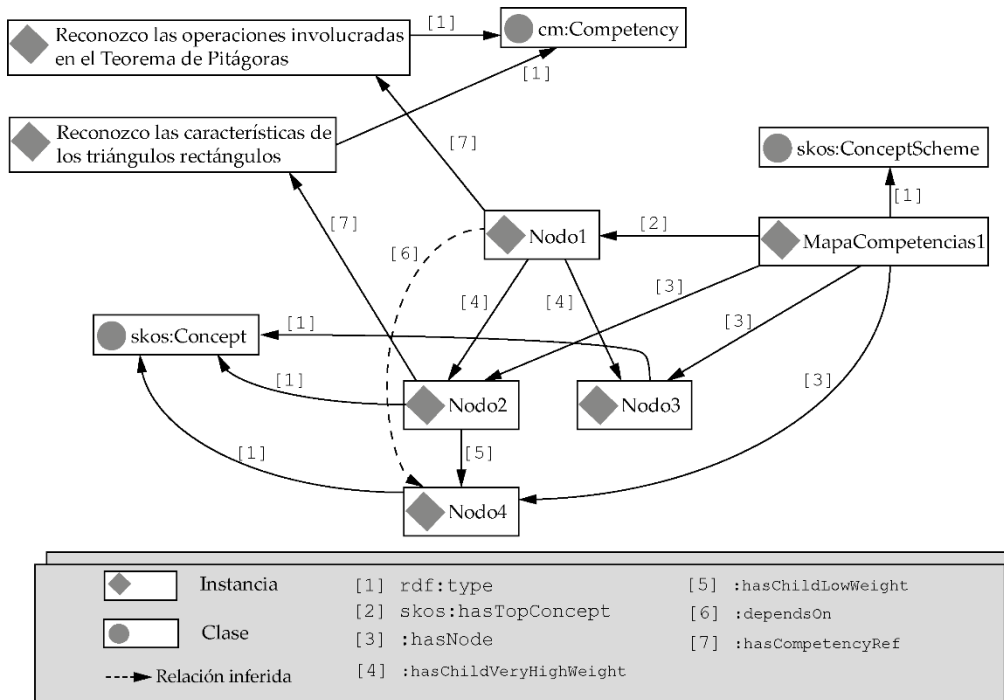


Figura 4.12 Ejemplo de un mapa de competencias

De acuerdo con los modelos de inferencias planteados (ver Figura 4.11 y Figura J.3, Figura J.4 en Anexo J), con solo aseverar las dos anteriores propiedades, se pueden inferir relaciones tales como: *Nodo2* y *Nodo3* son más específicos que *Nodo1*, siendo :isSpecific una propiedad inferida; el *Nodo1* es general para el *Nodo2*, *Nodo3* y *Nodo4*, siendo :isGeneral la propiedad inferida; el *Nodo1* depende del *Nodo2*, *Nodo3* y *Nodo4*, siendo :dependsOn la propiedad inferida.



4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

Figura 4.13 Ejemplo de aplicación de la ontología para el modelado de mapas de competencias. Fuente: propia.

Con respecto al último caso, la dependencia que pueda tener el *Nodo1* del *Nodo4* se obtiene de la transitividad de las relaciones, en este caso, las relaciones *Nodo1-Nodo2* y *Nodo2-Nodo4*. En la práctica, dado que el concepto de mapa de competencias sugiere que las competencias de niveles más bajos apoyan el desarrollo de las de niveles más altos, conviene descubrir este tipo de relaciones en la base de conocimiento.

4.3 Integración de ontologías

La naturaleza los conceptos formalizados señala que las ontologías pueden ser integradas estableciendo relaciones entre ellas. De hecho, esto se ha señalado en el esquema de metadatos del Capítulo 3.

Por ejemplo, los mapas de competencias están vinculados con las competencias a través de los nodos y sus referencias a ellas. De forma similar, los contenidos de VoD son vinculados con las competencias a través de sus segmentos y referencias a ellas. Además, algunos conceptos son idénticos entre varias de las ontologías.

Por ejemplo, el concepto *Language* está formalizado en “*Learning Object Metadata Ontology*”, “*Ontology for Media Resources*” y “*Competency Model Ontology*”. Dado que existen algunas similitudes entre instancias y clases de las diferentes ontologías, lo cual puede ser útil para mejorar la interoperabilidad, se decidió realizar un emparejamiento basado en el alineamiento de ontologías (“*ontology alignment*”) [104], proceso en el cual se descubren similitudes entre conceptos teniendo en cuenta sus propiedades (emparejamiento al nivel de elemento), como el nombre, o comparando la estructura de las ontologías (emparejamiento a nivel de la estructura), por ejemplo, a través de sus jerarquías de conceptos. Otra opción de integración hallada en la literatura es la fusión de ontologías (“*ontology merging*”) [105], la cual provee como resultado una ontología producto de la fusión de dos o más ontologías fuente. Aunque ambos métodos de emparejamiento son útiles para la integración de ontologías, el alineamiento ofrece la información necesaria para que el ingeniero tome las medidas que considere más convenientes.

Para desarrollar el proceso de integración, inicialmente se buscó una ontología que permita integrar todas las ontologías (llamadas *fuentes* en adelante), de tal forma que no sea necesario cambiar sus estructuras. Teniendo en cuenta el análisis de ontologías existentes realizado en el estado del arte (Sección 2.2.4), fue seleccionado el “*framework Learning Object Context Ontologies*” (“*LOCO framework*”) el cual integra ontologías de dominio (incluyendo la aquí adaptada “*Competency Model Ontology*”)

para vincular de forma lógica diferentes conceptos en torno a los OA, tomando como punto integrador la ontología base “*Learning Context Ontology*”. De esta forma, en total son cinco las ontologías que hacen parte de la integración.

Para construir un modelo de alineamiento, fue utilizada la herramienta COMA 3.0, la cual es capaz de encontrar correspondencias entre elementos de dos o más ontologías implementadas a través de un proceso de emparejamiento (“*matching*”). Dichas correspondencias fueron analizadas para establecer si los conceptos de diferentes ontologías pueden ser equivalentes, o ser subconceptos unos de otros; y así mismo con las propiedades, concluyendo en las acciones a ejecutar para el alineamiento de las ontologías.

La Figura 4.14 presenta la interfaz de COMA 3.0 después de realizado el emparejamiento entre “*Learning Object Ontology*” adaptada (a la izquierda del panel) y “*Competency Model Ontology*” adaptada (a la derecha). La herramienta utiliza un algoritmo que estima el *nivel de confianza* de correspondencia entre los elementos, el cual oscila entre cero (correspondencia nula) y uno (correspondencia total).

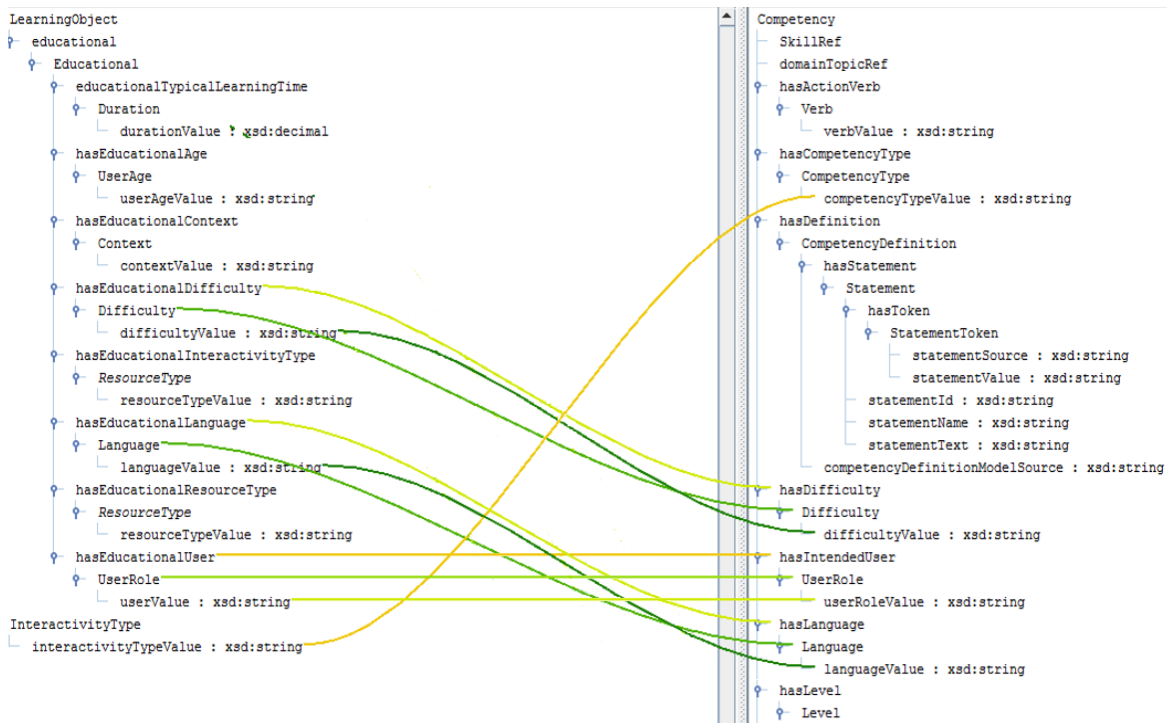


Figura 4.14 Resultado gráfico del “*matching*” de ontologías por medio de COMA. Fuente: propia.

El resultado del emparejamiento puede observarse en la Figura 4.15, en el cual se indican las correspondencias encontradas y su nivel de confianza. En este trabajo, las correspondencias con niveles de confianza superiores a 0.5 fueron analizadas

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

con detalle, en cuanto a significado, instancias, propiedades y jerarquía de clases, para tomar las medidas correspondientes (no porque no existiese, no fue posible hallar en la literatura revisada una escala que determine niveles de confianza bajos y altos). Por ejemplo, se observa un nivel de confianza de 0.85 entre las clases *Language* de una y otra ontología. De esto, sumado a que en sus modelos fueron definidos conceptos y clases idénticas para *Language*, se concluye que ambas clases pueden ser integradas estableciendo una relación de equivalencia. Conclusiones semejantes se obtuvieron con las demás clases y propiedades.

```
- LearningObject.educational.Educational.hasEducationalDifficulty <-> Competency.hasDifficulty: 0.6094048
- LearningObject.educational.Educational.hasEducationalLanguage <-> Competency.hasLanguage: 0.5896367
- LearningObject.educational.Educational.hasEducationalUser <-> Competency.hasIntendedUser: 0.45057648
- LearningObject.educational.Educational.educationalTypicalLearningTime.Duration.durationValue <-> Competency.hasLearningTime.Duration.durationValue: 0.89
- LearningObject.educational.Educational.hasEducationalAge.UserAge <-> Competency.UserAge: 0.6678404
- LearningObject.educational.Educational.hasEducationalAge.UserAge.userAgeValue <-> Competency.ageValue: 0.60240966
- LearningObject.educational.Educational.hasEducationalDifficulty.Difficulty <-> Competency.hasDifficulty.Difficulty: 0.8474026
- LearningObject.educational.Educational.hasEducationalDifficulty.Difficulty.difficultyValue <-> Competency.hasDifficulty.Difficulty.difficultyValue: 0.86494255
- LearningObject.educational.Educational.hasEducationalLanguage.Language <-> Competency.hasLanguage.Language: 0.8390411
- LearningObject.educational.Educational.hasEducationalLanguage.Language.languageValue <-> Competency.hasLanguage.Language.languageValue: 0.8584337
- LearningObject.educational.Educational.hasEducationalUser.UserRole <-> Competency.hasIntendedUser.UserRole: 0.68476146
- LearningObject.educational.Educational.hasEducationalUser.UserRole.userRoleValue <-> Competency.hasIntendedUser.UserRole.userRoleValue: 0.6213813
- InteractivityType.interactivityTypeValue <-> Competency.hasCompetencyType.CompetencyType.competencyTypeValue: 0.42982456
+ Total: 13 correspondences
```

Figura 4.15 Resultado explícito del “*matching*” de ontologías por medio de COMA.

Fuente: propia.

El anterior proceso fue realizado entre pares de las cinco ontologías. Aunque en los trabajos de la literatura revisados no fue posible hallar una guía explícita acerca de las medidas a considerar en el proceso de alineamiento, en la especificación de OWL se definen las siguientes propiedades útiles para establecer emparejamientos entre clases (sólo se mencionan las aplicables a las ontologías propuestas):

- `owl:equivalentClass`. Propiedad OWL usada para indicar que dos clases tienen la misma extensión, es decir, contienen exactamente el mismo conjunto de instancias. Así, puede ser usada para emparejar dos clases que contienen las mismas instancias.
- `rdfs:subClassOf`. Propiedad RDFS usada en caso de que: si la clase C_1 es definida como una subclase de C_2 , entonces el conjunto de instancias de C_1 podría ser un subconjunto del conjunto de instancias de C_2 . En general, si se evidencia que las instancias de una clase C_a de una ontología O_1 pueden ser los de una clase C_b de otra ontología O_2 con un significado más general, las ontologías pueden alinearse definiendo a la primera clase como subclase de la segunda usando `rdfs:subClassOf`.

En cuanto a instancias, se aplicó la propiedad `owl:sameAs`, la cual es usada para indicar que dos instancias tienen la misma identidad, y que por lo tanto se refieren a la misma cosa. Esto es aplicable por ejemplo a las instancias de la clase *Difficulty* de las ontologías “*Object Metadata Ontology*” y “*Competency Model Ontology*”, las cuales

definen un nivel de dificultad como *Very difficult* y *VeryHard*, respectivamente. Aunque sus nombres son diferentes, significan lo mismo, por lo cual se puede establecer la propiedad `owl:sameAs` entre ambas.

Otras propiedades entre clases fueron definidas para vincular las entidades, por ejemplo, entre: competencias y mapas; competencias y segmentos de video; objetos de aprendizaje y competencias.

Después de analizar los resultados entregados por COMA 3.0, verificando posibles errores en la interpretación (por esta razón, las herramientas no hacen la alineación de forma automática) e interpretando los conceptos formalizados en las ontologías hasta aquí, se tomaron las siguientes acciones explícitas para la integración:

1. Definir la clase *LearningObject* de “*Learning Object Metadata Ontology*” adaptada como equivalente a la clase *LearningObject* de “*Learning Context Ontology*” por medio de `owl:equivalentClass`. De esta forma se establece que todas las instancias (los contenidos de VoD) son las mismas para ambas clases.
2. Definir la clase *MediaResource* de “*Ontology for Media Resources*” adaptada como una subclase de *LearningObject* de “*Learning Object Metadata Ontology*” adaptada. De esta forma, las instancias (contenidos de VoD) descritos como Recursos de Video serán un subgrupo de todos los OA.
3. Definir la clase *Competency* de “*Competency Model Ontology*” adaptada como subclase de *LearningContext* de “*Learning Context Ontology*”. De esta forma expresa la complementariedad entre una ontología y otra, siendo la segunda la más general. La segunda clase representa un concepto que engloba cualquier aspecto del contexto que no está formalizado aun, siendo este el caso de las competencias.
4. Definir la clase *CompetencyMap* de *OntoRCM* como subclase de *LearningContext* de “*Learning Context Ontology*”. Tiene las mismas implicaciones del caso anterior.
5. Definir como equivalentes por medio de `owl:equivalentClass`, las clases *Difficulty*, *Language* y *UserRole* de “*Learning Object Metadata Ontology*” adaptada a sus correspondientes de “*Competency Model Ontology*” adaptada. De esta forma se establecen puntos de vinculación entre competencias y OA por medio de algunas de sus características.
6. Definir una “*Object Property*” llamada `:hasLearningObjective` entre la clase *LearningObject* de “*Learning Object Metadata Ontology*” adaptada y la clase *Competency* de “*Competency Model Ontology*” adaptada. Así, se vinculan a las competencias los OA que pueden ser útiles para su adquisición. Esta propiedad

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

representa al elemento `PRCDRef` del esquema de metadatos, el cual cumple la misma función que `:hasLearningObjective` entre descripciones.

7. Definir una “*Object Property*” llamada `:compRef` entre la clase *VideoTrack* de “*Ontology for Media Resources*” adaptada y la clase *Competency* de “*Competency Model Ontology*” adaptada. De esta forma se establecen vínculos entre los segmentos de video y las competencias. Esta propiedad representa al elemento `RCDRef` del esquema de metadatos, el cual cumple la misma función de `:compRef` pero entre descripciones.
8. Definir como iguales a las instancias de *Difficulty*, *Language* y *UserRole* de “*Learning Object Metadata Ontology*” adaptada y a los de sus correspondientes de “*Competency Model Ontology*” adaptada, por medio de `owl:sameAs`. De esta forma se establecen puntos comunes que vinculan a contenidos de VoD y competencias, que pueden ser útiles para las estrategias de filtrado
9. Definir las clases *Competency* de *OntoRCM* y *Competency* de “*Competency Model Ontology*” adaptada como equivalentes por medio de `owl:equivalentClass`. De esta forma se podrán vincular a los mapas las competencias definidas como instancias en una u otra ontología.

La Figura 4.16 y Figura 4.17 en conjunto presentan el modelo general de la integración, señalando cada ontología con un color diferente. Cabe aclarar que en el modelo están únicamente los conceptos y relaciones importantes para la integración, pero es claro que la totalidad de los elementos de las cuatro ontologías propuestas (incluyendo los que son parte de las ontologías originales que no han sido mencionados en el proceso de adaptación) están involucrados en él.

Después de la implementación de la ontología por medio de la herramienta *Protégé* se ha extraído en la Figura 4.18 una sección de ella que permite visualizar algunas relaciones importantes que facilitan la comprensión del modelo obtenido. Se observa que los mapas de competencias *MapaCompetencias1* y *MapaCompetencias2* tienen asociados grupos de nodos por medio de la propiedad `:hasNode`. Los nodos están vinculados entre ellos por medio de la propiedad transitiva inferida `:dependsOn` (inferida a partir de la propiedad `:isGeneralFor`). Adicionalmente, los nodos están relacionados con las competencias por medio de la propiedad `:hasCompetencyRef`. Las competencias están vinculadas con los verbos de acción y el tipo por medio de las propiedades `:hasActionVerb` y `:hasCompetencyType` respectivamente. Además, las competencias están relacionadas con segmentos de video por medio de la propiedad `:CompRef`. Finalmente, los recursos de video se vinculan con segmentos por medio de la propiedad `:hasVideoTrack`. Como se verá más adelante, este tipo

de relaciones entre instancias es explotado por las técnicas para el descubrimiento y ponderación de asociaciones semánticas complejas, las cuales son importantes para el planteamiento de la estrategia de recomendación propuesta en esta investigación.

En el extracto de la base de conocimiento sólo se muestra una parte de los elementos formalizados en el modelo, por lo cual se invita al lector a analizar todas las posibilidades ofrecidas en cuanto a propiedades aseveradas e inferidas en el presente trabajo⁶.

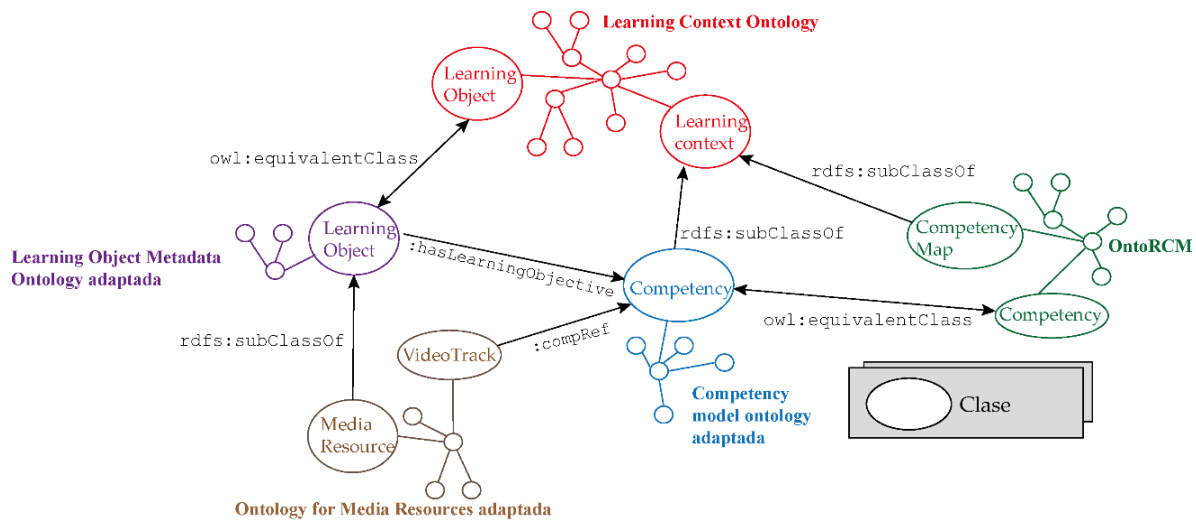


Figura 4.16 Vista 1 del modelo de integración de ontologías. Fuente: propia.

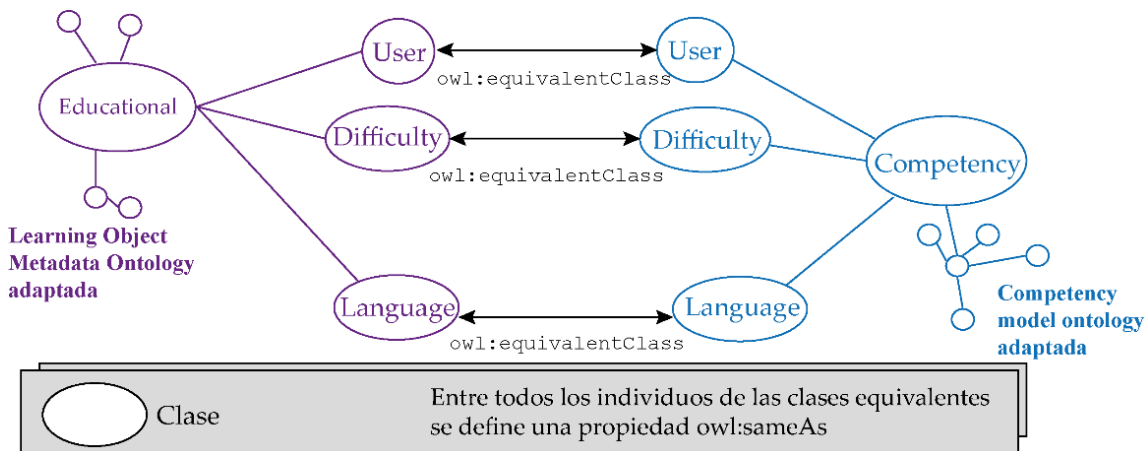


Figura 4.17 Vista 2 del modelo de integración de ontologías. Fuente: propia.

⁶ La base de conocimiento construida puede ser descargada de: https://mega.nz/#F!y3YWVKxZ!dLnxI4fAqjyyU_INXKUHLA

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

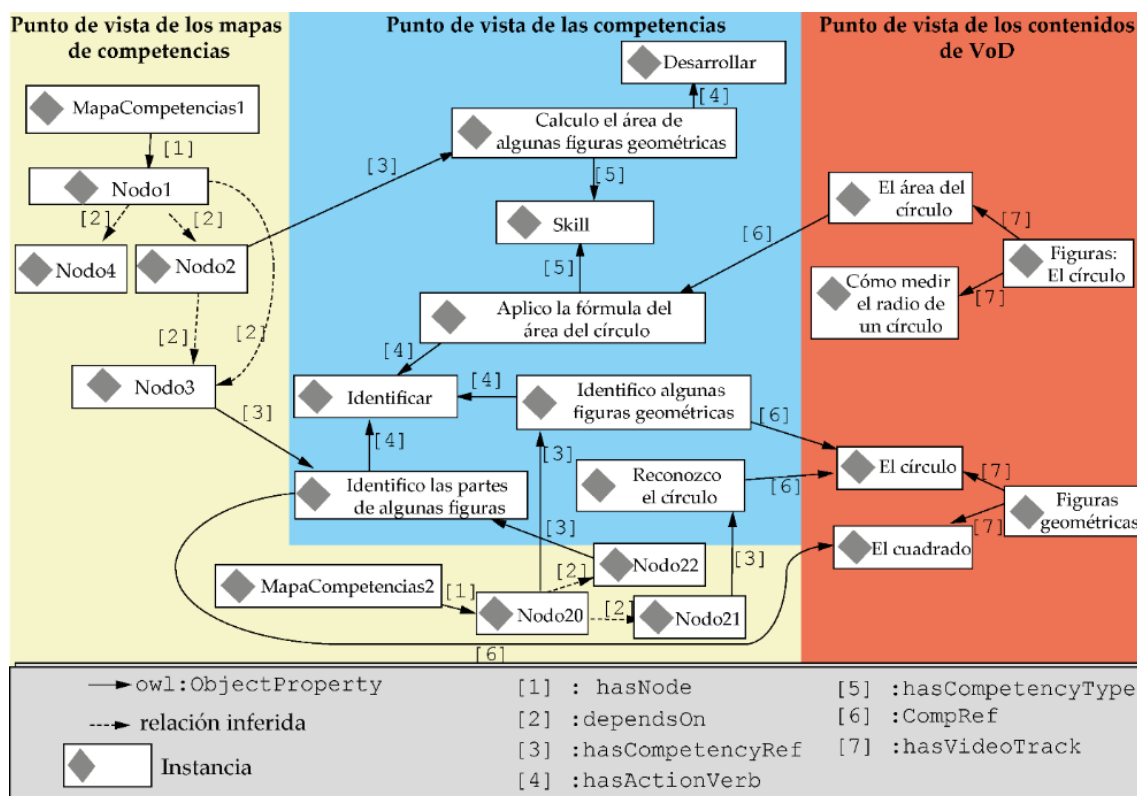


Figura 4.18 Extracto de la base de conocimiento. Fuente: propia.

4.4 Evaluación de la ontología

Una de las ventajas más importantes que ofrecen los SR semánticos es la posibilidad de inferir conocimiento útil para mantener el carácter personalizado de las recomendaciones. Esto, por una parte, es posible gracias a la lógica implícita en la estrategia de recomendación, estableciendo la forma en cómo se realiza y aprovecha la inferencia sobre una estructura de conocimiento; y, por otra parte, la capacidad que tiene dicha estructura para formalizar el conocimiento vinculado al dominio de interés. Precisamente, dado que la ontología propuesta tiene como fin ofrecer los medios para la aplicación de una estrategia de recomendación enfocada en el dominio de los contenidos de VoD y las competencias, se ha realizado una evaluación enfocada en el criterio de *completitud* [39], a partir del cual se mide si se cubre el dominio de interés de forma apropiada.

De acuerdo con [39], la completitud abarca los siguientes aspectos: *declaración de URI, literales y tipos de datos, validación XML, profundidad máxima de la jerarquía de conceptos, preguntas de competencia formalizadas, "hash" (#) vs "slash" (/), etiquetas y comentarios, nodos en blanco y la completitud del lenguaje*. Para la realización de la

evaluación mostrada aquí se tuvieron en cuenta los primeros cinco conceptos porque están ligados a factores importantes dentro de los procesos de recomendación tales como: 1) la forma en cómo se consulta del conocimiento formalizado, para lo cual las URI, los literales y los tipos de datos deben estar bien declarados en un formato de fácil acceso como XML; 2) la clasificación de los conceptos formalizados, a partir de la cual se puede inferir conocimiento, para lo cual la jerarquía de clases de la ontología no debe contener errores; 3) el tipo de conocimiento que debe responder la ontología, para lo cual la ontología debe responder a preguntas de competencia inherentes al dominio.

Otros aspectos tales como el uso del “*hash*” (#) vs “*slash*” (/), *etiquetas y comentarios*, *nodos en blanco* y la *completitud del lenguaje* no fueron tenidos en cuenta en la evaluación porque están enfocados en aclaraciones en lenguaje natural y en distintos idiomas sobre el uso y estilo de la ontología o en la identificación de propiedades o nodos que no han sido considerados en el diseño porque no ha sido necesario; lo cual no influye en el despliegue de estrategias de recomendación.

A continuación, se presentan de forma resumida los resultados de la evaluación. Todo el proceso desarrollado se presenta detalladamente en el Anexo K.

Con respecto a la *declaración de URIs* se obtuvieron los siguientes resultados:

- El 100% de las URI de la ontología tienen declarado su tipo.
- El 100% de las declaraciones son acordes con su uso. Además, no existen declaraciones erróneas de las URI.
- Aunque algunas clases e instancias comparten el mismo nombre entre diferentes ontologías, sus URI se diferencian entre sí gracias a sus espacios de nombres (“*namespaces*”). Esto implica la no necesidad de analizar casos puntuales de “*punning*”.

Con respecto a los *literales y tipos de datos*, los resultados de la evaluación son los siguientes:

- De los 55 “*Data types*” definidos en el modelo ontológico, el 100% están restringidos por tipos de datos en facetas. De ellos, 46 son `xsd:string`, 6 son `xsd:decimal`, 2 son `xsd:anyURI`, 1 es `xsd:integer`. Se observa que los tipos de datos recomendados han sido utilizados de forma justificada. El uso de `xsd:decimal` se justifica porque es el tipo de dato usado para especificar las duraciones de los contenidos y del aprendizaje de las competencias. El uso de `xsd:anyURI` se justifica porque es el mejor para especificar las ubicaciones de los contenidos y de los servicios de VoD.

4. Una ontología sobre el dominio de los contenidos de VoD y las competencias educativas

- Con respecto a la declaración de los literales, el 100% son sintácticamente válidos con los tipos de datos indicados en el ítem anterior. Esto cierra la puerta a posibles inconsistencias. Este resultado ha sido corroborado por medio del uso del razonador *Hermit* en *Protegé*.

Con respecto a la *validación XML*, los resultados son los siguientes:

- La ontología originalmente está serializada en un formato RDF/XML, por lo cual no ha sido necesaria la serialización en otro formato XML.
- Con la ayuda de la herramienta en línea "*XML validator*"⁷, se corroboró que la ontología es válida con respecto a XML. Entonces, ésta puede ser compatible con herramientas XML útiles como convertidores a formatos JSON, HTML, CSV; u otros validadores. A futuro se espera realizar una prueba enfocada en verificar si la ontología puede ser soportada por otras aplicaciones después de un proceso de conversión.

En cuanto a la *profundidad máxima de la jerarquía de conceptos*, se obtuvieron los siguientes resultados:

- La Profundidad Máxima de la Taxonomía de conceptos es TD= 6.
- La Jerarquía de conceptos del modelo no presenta ciclos, lo cual corrobora el valor de TD mencionado y que las relaciones de subsunción son claras y declaradas correctamente.

Con respecto a las *preguntas de competencia formalizadas*, se realizaron consultas SPARQL sobre un fragmento de la base de conocimiento construido a partir de la ontología, para las cuales se realizó la verificación de los resultados esperados. El resultado es que la ontología respondió 100% de forma correcta a las preguntas de competencia. Así, se concluye que la ontología provee el conocimiento (esto es, aseverado e inferido) que se ha establecido como el de mayor relevancia en el contexto del trabajo, que además se presume clave en el proceso de obtención de recomendaciones.

En síntesis, los resultados de la evaluación se deben a que:

1. La ontología se construyó a partir de esquemas de metadatos, lo cual permitió reconocer de antemano el conocimiento del dominio que debía ser formalizado, incluyendo las jerarquías de clases, instancias, propiedades y tipos de datos. Esto

⁷<http://www.xmlvalidation.com/index.php?id=1&L=0&target=/xmlvalidation/start.jsp;jsessionid=E9F659FA35AAB264EDD890D7C2D92CD0>

corroborar las ventajas que ofrece el enfoque de construcción de ontologías a partir de metadatos [83], [57], [58].

2. La ontología fue construida usando la herramienta *Protégé*, la cual asigna URI de forma consistente, incorpora razonadores que identifican problemas de consistencia, y serializa la ontología en un documento XML.

Capítulo 5

Método computacional basado en información cualitativa para la evaluación de competencias

De acuerdo con la hipótesis principal de la presente investigación, se busca mejorar la búsqueda de contenidos de VoD a través de un *Sistema de Recomendaciones* (SR) centrado en las competencias educativas. Esto implica la necesidad de incorporar información del currículo educativo en el proceso con el propósito de orientar las recomendaciones a las necesidades de los estudiantes. Específicamente, la investigación se centra en la oportunidad de diagnosticar el aprendizaje del estudiante a través de la evaluación de competencias, a partir de lo cual se posibilita generar información que puede ser formalizada en el perfil de usuario. Al respecto, la construcción de perfiles es un aspecto importante dentro del proceso de diseño de los SR en cuanto a personalización se refiere.

Con respecto a la dinámica implícita en la evaluación, los docentes califican *Actividades de Evaluación* (AE) (por ejemplo, talleres, mesas redondas, exámenes escritos y orales) asignando *Niveles de Competencia* (NC) de acuerdo al desempeño del estudiante. Esta tarea puede ser realizada usando notas numéricas en el rango [0,10]; sin embargo, según Fung [27], el uso de términos lingüísticos (por ejemplo, *Bajo, Medio, Alto*) puede ser más natural y significativo en la evaluación de algunos tipos de AE y competencias, por ejemplo, si estas involucran habilidades y capacidades observables de los estudiantes en actividades de solución de problemas.

Aunque el uso de términos lingüísticos es útil, la evaluación de competencias puede ser un proceso difícil e impreciso cuando el docente evalúa estructuras en las que se establecen relaciones entre ellas, para lo cual debe considerar una gran cantidad de información. Éste es el caso de los mapas de competencias, los cuales sirven como herramientas para observar los NC alcanzados en el currículo de un curso o programa [7]. De acuerdo con la Figura 2.1 (Sección 2.1.4), los mapas de competencias están compuestos por competencias, relaciones de tipo general-específica entre éstas con etiquetas lingüísticas de pesos (estos ponderan la importancia de una competencia específica en el desarrollo de una general), y los NC lingüísticos asignados en los

procesos de calificación. Con el propósito de aprovechar esta información en la identificación de las necesidades de los estudiantes, se propone un *método computacional para la evaluación de competencias que considera la estructura de los mapas de competencias*.

De acuerdo a lo anterior, el método computa información lingüística, para lo cual, los *modelos computacionales lingüísticos* aportan inevitablemente sus capacidades al desarrollo de la presente investigación. Con el propósito de definir un modelo matemático base, fue necesario analizar en el estado del arte (Sección 2.1.7) modelos como el de *2-tuplas* y el basado en el *principio de extensión* a través de ejemplos de aplicación (ver Ejemplos A.1 y A.2, Anexo A). Los resultados concluyen que el modelo de *2-tuplas* es más preciso en el cómputo términos lingüísticos (área conocida como “*Computing with Words*”), además de que, por su simpleza, sugiere un consumo menor de recursos computacionales. Según [66], la ventaja ofrecida por el modelo de *2-tuplas* en cuanto a precisión también se presenta sobre el *modelo lingüístico simbólico* [106]. Dado que se busca construir perfiles de usuario precisos para el SR, se ha elegido al enfoque de *2-tuplas* como el proveedor del modelo matemático base para la generación del *método computacional basado en información cualitativa para la evaluación de competencias*.

Para esto, inicialmente se presentan aspectos relacionados con la información a computar; específicamente, se trata de las características de la estructura de los mapas de competencias. Posteriormente se presenta el método computacional propuesto, el cual es el producto de una adaptación de un algoritmo de búsqueda en grafos. Precisamente, el método computacional es el aporte al conocimiento ofrecido por el presente capítulo.

Finalmente, cabe destacar que el método fue evaluado según la precisión de los resultados usando como insumo mapas de competencias de procesos educativos reales.

Cada uno de estos aspectos es tratado con detalle en las siguientes secciones. Los resultados de este capítulo se han publicado en [6] y [107].

5.1 Estructura de los mapas de competencias

Tal como se adelantó en la Sección 2.1.4, una buena aproximación a la estructura de un mapa de competencias es ofrecida por el borrador de estándar “*Simple Reusable Competency Map*” (SRCM). Sin embargo, en él no se han considerado las *Actividades de Evaluación* (AE), las cuales son importantes para medir el rendimiento de los estudiantes. En consecuencia, se ha decidido añadir este tipo de entidades, resultando

5. Método computacional basado en información cualitativa para la evaluación de competencias

el modelo mostrado en la Figura 5.1. En ella, se presenta un ejemplo de un mapa de competencias provisto por un docente de matemáticas practicante de los *Estándares Básicos de Competencias de Colombia* para establecer las metas de enseñanza y aprendizaje [53]. Como puede observarse, el nuevo modelo tiene las siguientes características:

- Las competencias tienen subelementos que pueden ser AE, o incluso, otras competencias (notar que una competencia puede ser un subelemento o subcompetencia de otra). Entonces, los mapas pueden estar compuestos por uno o más niveles.
- Las AE diseñadas por los docentes (exámenes orales o escritos, solución de problemas, etc.) han sido incluidas como subelementos de algunas competencias. En la práctica educativa, éstas son usadas por los docentes para realizar un proceso de evaluación de los estudiantes, en el cual son asignados unos NC.
- Según el SRCM, la estructura establece relaciones entre sus elementos, las cuales son etiquetadas con un peso que representa la importancia de una subcompetencia en el desarrollo de una competencia de un nivel superior.
- Los NC han sido asignados por el docente a todos los elementos del mapa.

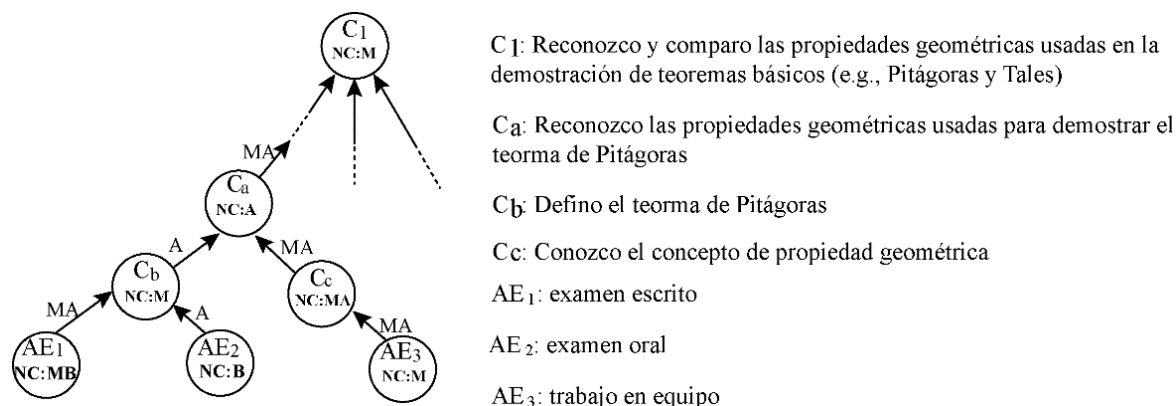


Figura 5.1 Modelo de los mapas de competencias aplicado a través de un ejemplo. Se añadieron las actividades de evaluación al modelo de SRCM. Fuente: propia.

En el ejemplo, tanto pesos como NC fueron asignados de acuerdo a un grupo de siete términos lingüísticos como el siguiente:

$$S = \{N: Nulo, MB: Muy Bajo, B: Bajo, M: Medio, A: Alto, MA: Muy Alto, T: Total\}. \quad (5.1)$$

Cabe mencionar que aunque la misma estructura puede ser usada para varios estudiantes, cada mapa de competencia corresponde a un solo estudiante, ya que registra sus propios NC.

Finalmente es importante destacar que en el Capítulo 4 (Sección 4.2) se propuso la ontología *OntoRCM*, la cual se centra en la descripción de mapas de competencias por medio de las cuatro características del modelo de los mapas anteriormente mencionadas. Esto implica que se mantiene la estructura de los grafos (específicamente son grafos acíclicos dirigidos), por lo cual el método computacional puede estar ligado a este tipo de representaciones simbólicas.

Como comentario final, la descripción de competencias y mapas de competencias puede ser una tarea compleja para los docentes, lo cual puede minar la calidad de las descripciones y por ende, de las recomendaciones. Por un lado, un mapa es susceptible de diferentes interpretaciones, por ejemplo, se puede tomar como la formalización de competencias precondición de otras o de competencias de apoyo para la adquisición de otras. Por otro lado, algunas competencias definidas de manera estandarizada en la educación formal pueden ser llegar a ser complejas en sí mismas porque, por ejemplo, definen más de dos verbos de acción, habilidades y/o tópicos. Para mitigar estos problemas, el *Ministerio de Educación de Colombia* ha definido algunas herramientas pedagógicas tales como los *Derechos Básicos de Aprendizaje* (DBA) y las *Matrices de referencia*. Así las cosas, la calidad de las descripciones de competencias y mapas depende en gran parte de los docentes. Sin embargo, cabe destacar que el esquema de metadatos (Capítulo 3) y la ontología (Capítulo 4) propuestos ofrecen dos posibilidades que pueden reducir la complejidad en la creación de descripciones: 1) generar modelos colaborativos para la creación, reutilización y enriquecimiento de descripciones; y 2) utilizar técnicas de análisis de texto para la creación de descripciones a partir del lenguaje natural. En la práctica, los docentes podrían tomar competencias y mapas preconcebidos en repositorios o en ambientes de datos enlazados en la web para alimentar el SR, reduciendo así la complejidad inherente a la creación de descripciones de calidad. La construcción de herramientas para el despliegue de soluciones como las mencionadas está fuera del alcance del presente trabajo.

5.2 Descripción del método computacional

El proceso involucrado en el método computacional se representa en la Figura 5.2. En ella se definen dos fases así:

Fase A: construcción del perfil de estudiante. Dado que los mapas de competencias aportan información sobre el rendimiento de los estudiantes, resulta propicio extraer de ellos información útil para modelar perfiles de usuario P_u . Precisamente, esta tarea

5. Método computacional basado en información cualitativa para la evaluación de competencias

es realizada en la *Fase A*, además de que se organiza la información de forma que se pueda aplicar un modelo matemático.

Fase B: cómputo de los niveles de competencia. En esta fase se establece la forma en cómo se aplica el enfoque de las *2-tuplas* en la evaluación de competencias. Esto teniendo en cuenta el perfil de usuario P_u y la estructura en grafo de los mapas, incluyendo los tipos de elementos involucrados (competencias y AE), sus relaciones y los pesos de tales relaciones.

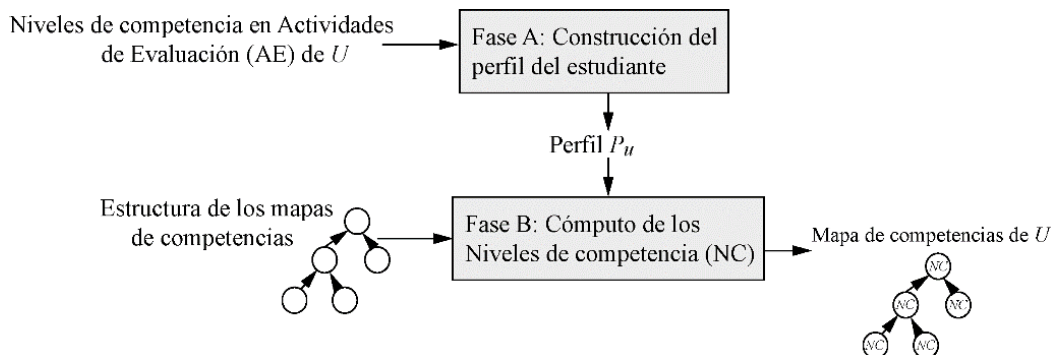


Figura 5.2 Método computacional propuesto. Fuente: propia.

Las próximas secciones describen metodológicamente cómo se ejecutan las fases mencionadas, explicadas con base en el modelo genérico de un grafo etiquetado. Con esto se busca independizar al método del lenguaje de implementación, asegurando que éste pueda ser adaptado a diversos lenguajes y formatos para grafos tales como OWL y el *Formato de Grafos JSON* [108]. Al respecto, la Figura 5.3 presenta la equivalencia entre una representación ontológica usando la ontología para mapas propuesta en el Capítulo 4 (Figura 5.3a) y una genérica de un grafo acíclico etiquetado (Figura 5.3b).

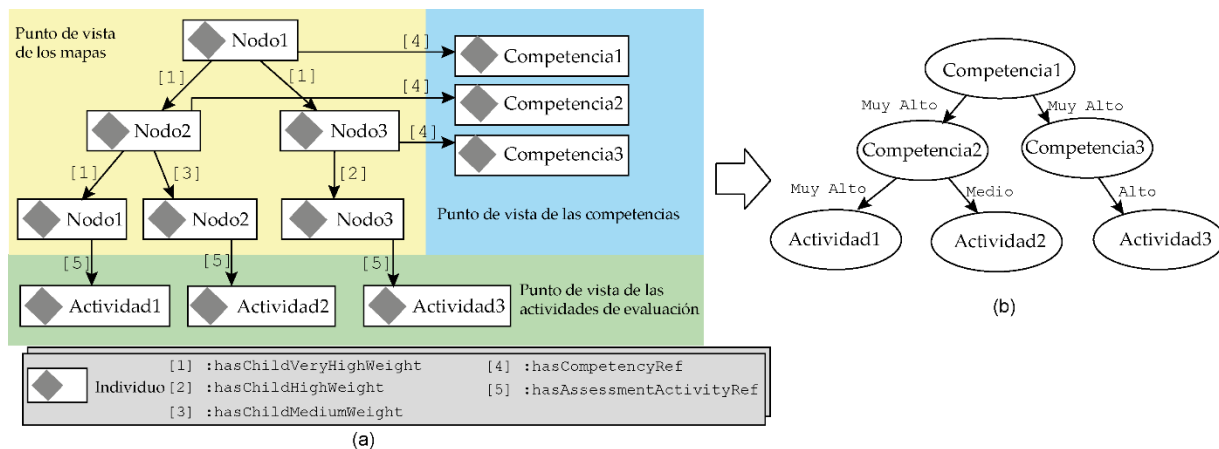


Figura 5.3 Modelo de los mapas de acuerdo con: (a) el modelo ontológico propuesto; (b) el modelo de un grafo etiquetado. Fuente: propia.

5.2.1 Fase A: Construcción del perfil del estudiante

En esta fase se modelan los perfiles de usuario usados para la ejecución del método. En ellos se registran los NC alcanzados por los estudiantes en las AE referenciándolas por medio de los identificadores consignados en la base de conocimiento. Así, se obvia el registro de todos los elementos del mapa en el perfil de usuario. Suponiendo que un usuario U ha alcanzado un conjunto NC_u de *niveles de competencia* en las actividades AE_1, AE_2, \dots, AE_j , entonces tales NC se registran en el perfil P_u , organizándolos en conjuntos de acuerdo al orden de las AE en el grafo, tal como lo define la expresión (5.2).

$$P_u = \{NC_1, NC_2, \dots, NC_j\}. \quad (5.2)$$

Estas características de modelado suponen un bajo consumo de recursos computacionales debido a la simpleza y ordenamiento de los datos.

5.2.2 Fase B: Cómputo de los niveles de competencia

Para definir el proceso de cómputo de los NC, inicialmente se eligió un algoritmo especializado en el recorrido por grafos. Entre los hallados en [109] están el de *Búsqueda Primero en Profundidad* (DFS) y el de *Búsqueda Primero en Anchura* (BFS), los cuales, a la luz de lo requerido en la investigación, ofrecen la misma funcionalidad: recorrer los mapas. Sin embargo, después de una revisión en la web se hallaron implementaciones de DFS en Java que sirven de apoyo para la construcción de prototipos [110]. Así las cosas, éste algoritmo fue elegido para sumarlo a la solución planteada.

Por otra parte, era necesario incorporar un modelo matemático enfocado en tomar los perfiles de usuario P_u y las características estructurales de los mapas para realizar el proceso de evaluación. Para abordar esta situación, el modelo matemático de *2-tuplas* se adaptó al algoritmo DFS para obtener como resultado el algoritmo denominado aquí DFS-CA ("*Depth First Search-Competency Assessment*" por su significado en inglés).

De manera general, DFS-CA recorre los mapas de competencias sistemática y organizadamente al tiempo que ejecuta la matemática necesaria para la evaluación lingüística de las competencias. Para esto, el algoritmo toma como entradas el grafo G que formaliza la estructura del mapa de competencias y el perfil del estudiante P_u que registra los NC alcanzados en las actividades de evaluación. Posteriormente, recorre el grafo extrayendo información sobre las relaciones entre nodos padre y nodos hijo y los pesos de tales relaciones. Finalmente, realiza un proceso de agregación

5. Método computacional basado en información cualitativa para la evaluación de competencias

lingüística (el cual es equivalente a la suma ponderada de números) que permite calcular los NC en todas las competencias del mapa.

De manera específica, DFS-CA funciona de la siguiente manera (el pseudoalgoritmo de DFS-CA se muestra en el Algoritmo M.1, Anexo M):

Sea G el grafo que formaliza la estructura del mapa de competencias, el cual se compone de cada nodo x_i ; P_u el perfil del estudiante U que registra los NC alcanzados por él en las actividades de evaluación; p_i es el conjunto conformado por los pesos de las aristas entre un nodo x_i y sus nodos hijo; y nc_i el conjunto conformado por los NC de los nodos hijo del nodo x_i .

1. Todos los nodos de G se marcan como NO_VISITADOS. Ir al paso 2.
2. Se toma un nodo x_i de G marcado como NO_VISITADO, se marca como VISITADO y se toma como activo. Ir al paso 3.
3. Para el nodo activo, si es del tipo competencia, se crea el conjunto p_i que almacena (durante la ejecución del algoritmo) los pesos de las aristas entre él y sus respectivos nodos hijos x_{ji} ; y se crea el conjunto nc_i que almacena (durante la ejecución del algoritmo) los NC de sus nodos hijos. Ir al paso 4.
4. Se elige un nodo x_i de entre los hijos del nodo activo que esté marcado como NO_VISITADO. Ir al paso 5. Si tal nodo no existe, ir al paso 6.
5. Se marca el nodo x_i como VISITADO y se toma como activo. Si es del tipo *Competencia*, se crea el conjunto p_i que almacena (durante la ejecución del algoritmo) los pesos de las aristas entre él y sus respectivos nodos hijos x_{ji} ; y se crea el conjunto nc_i que almacena (durante la ejecución del algoritmo) los NC de sus nodos hijos. Sea del tipo competencia o no, se consulta: 1) el peso de las aristas entre el nodo activo y su padre en G , el cual se deduce de su etiqueta; y 2) su NC en P_u (sólo si el nodo activo es de tipo *Actividad de evaluación*). Estos valores se añaden a los conjuntos p_i y nc_i de su nodo padre, respectivamente, de acuerdo con $nc_i = (NC_{1,i}, NC_{2,i}, \dots, NC_{j,i})$ y $p_i = (w_{i1}, w_{i2}, \dots, w_{ij})$. Ir al paso 4.
6. Si se han alcanzado todos los nodos de G y se han computado todos los NC de las competencias, entonces se termina el algoritmo y se ejecuta para el siguiente estudiante. En caso contrario, el nodo padre del nodo activo se toma como activo. Si éste tiene nodos hijos marcados como NO_VISITADO se vuelve al paso 4. Si todos sus nodos hijos del están marcados como VISITADOS, se usan sus conjuntos p_i y nc_i para computar su NC por medio de:

$$NC_i = \Delta \left(\frac{\sum_{j=1}^n \beta_{ij} * w_{ij}}{\sum_{j=1}^n w_{ij}} \right), \quad (5.3)$$

Es importante señalar que en la medida que el algoritmo descubre los nodos del grafo, construye los conjuntos necesarios para el cómputo de los *niveles de competencia*. Adicionalmente, el uso de la recursividad permite que se puedan calcular los NC de forma ascendente y ordenada a través de los niveles de los mapas. Con arreglo a la expresión (5.3), ésta corresponde al *Operador de Media Ponderada* (WAO por su significado en inglés). Éste operador ha sido especialmente definido para el enfoque de *2-tuplas* lingüísticas en [66] para agregar términos lingüísticos, o dicho de otra forma, para computar palabras. Por ejemplo, si un estudiante tiene un NC igual a *Bajo* en una actividad de evaluación y otro igual a *Alto* en otra actividad, WAO permite computar estos dos términos para obtener un NC lingüístico total. WAO se compone de una parte numérica, la cual es el argumento de la función Δ , y de la función Δ , la cual convierte la parte numérica en un valor lingüístico. A cada uno de estos valores se les denomina en el presente método como *Valor de Agregación Numérico* (VAN) y *Valor de Agregación Lingüístico* (VAL), respectivamente.

Los valores de β_{ij} para todos los términos lingüísticos registrados en los conjuntos nc_i y p_i son definidos de acuerdo a dos posibilidades explicadas a continuación, en las cuales, se evidencia la importancia de considerar la diferencia entre los VAN y los VAL.

- La posibilidad *P1* consiste en que al calcular los NC del nivel 2 de la estructura del mapa, se debe tener en cuenta que los NC de las AE son lingüísticos. En consecuencia, para computar el NC de un nodo i que hace parte del nivel 2 del mapa usando el WAO, los valores β_{ij} corresponden a los índices de los NC de sus nodos hijos. Tales índices se obtienen de acuerdo a partir de la escala mostrada en la Figura 5.4. Por ejemplo, para calcular el NC de un nodo i para el cual los NC de sus 2 nodos hijos son M (Medio) y MB (Muy Bajo), se tiene que los valores β_{ij} correspondientes son 3 y 1. Los VAN computados son añadidos a los conjuntos nc_i , mientras que los VAL son los NC entregados como las calificaciones del estudiante en los nodos del nivel 2.

N	MB	B	M	A	MA	T
0	1	2	3	4	5	6

Figura 5.4 Asignación de índices para términos lingüísticos. Fuente: propia.

- La posibilidad *P2*, a diferencia de *P1*, consiste en que los valores β_{ij} corresponden a los VAN computados de los nodos hijos de cada nodo i . Es importante notar que los NC de sus nodos hijos son VAN, los cuales componen, según el algoritmo, a los conjuntos nc_i . Los VAN computados son añadidos a los conjuntos nc_i de sus nodos padre, mientras que los VAL corresponden a los NC entregados como las calificaciones del estudiante en los nodos del nivel 3 y superiores.

5. Método computacional basado en información cualitativa para la evaluación de competencias

Para cualquiera de las dos posibilidades, los VAL se obtienen de la misma manera. De acuerdo con el modelo basado en *2-tuplas*, la función Δ genera un par de *2-tuplas* (S_a, α_a) y (S_b, α_b) que corresponden al término más cercano inferior y superior del VAN. S_a es asignado si $|\alpha_a| < |\alpha_b|$, mientras que S_b es asignado si $|\alpha_b| \leq |\alpha_a|$.

Ejemplo 5.1. Suponiendo que el mapa de competencias mostrado en la Figura 5.1 hace parte de un currículo educativo, el rendimiento de un estudiante ha sido evaluado por el docente usando las actividades de evaluación *examen escrito* (AE1), *examen oral* (AE2) y un *trabajo en equipo* (AE3). En cada una de ellas, el estudiante obtuvo niveles de calificación iguales a *Muy bajo* (A), *Bajo* (B) y *Muy alto* (MA) respectivamente. Con base en toda la información se busca computar los *niveles de competencia* para todas las competencias del mapa (Cb, Cc, Ca y C1). Dado que esta tarea involucra múltiples términos lingüísticos, puede ser engorroso para el docente obtener unos resultados confiables. Entonces, resulta conveniente aplicar el método computacional propuesto para asistirlo en esta labor, tal como se explica a continuación.

De acuerdo con los *niveles de competencia* alcanzados por el estudiante en las AE, se construye su perfil $P_u = \{MB, B, MA\}$. De acuerdo con el algoritmo planteado, para calcular el NC de la competencia Cb, se tienen los conjuntos $p_b = \{A, MA\}$ y $nc_b = \{MB, B\}$. Dado que la competencia Cb hace parte del nivel 2 de la estructura, para la aplicación del WAO se hace uso de la posibilidad P1 así: $NCb = \Delta((1 * 4 + 2 * 5) / (4 + 5)) = \Delta(1.55)$. Por consiguiente, el VAN para la competencia Cb es 1.55. Para obtener el VAL, de la anterior expresión se analizan las *2-tuplas* $(MB, 0.55)$ y $(B, -0.45)$. Dado que $|-0.45| \leq |0.55|$, entonces el VAL asignado es B (Bajo). Con respecto a la competencia Cc, se observa que tiene un solo nodo hijo que es AE3. Por consiguiente, los valores VAL y VAN para la competencia Cc son los mismos que los de la AE3, estos son MA (Muy Alto) y 5 respectivamente. Finalmente, para calcular el NC de la competencia Ca se considera la posibilidad P2, ya que hace parte del nivel 3. En este caso, $p_a = \{MA, A\}$ y $nc_a = \{1.55, 5\}$. Aplicando el WAO, se obtiene: $Nca = \Delta((1.55 * 5 + 5 * 4) / (5 + 4)) = \Delta(3.08)$. Como resultado, el VAN para la competencia A es 3.08, mientras que las dos *2-tuplas* que se derivan del cómputo son $(M, 0.08)$ y $(A, -0.92)$. Debido a que $|0.08| \leq |-0.92|$, el VAL asignado a la competencia Ca es M (Medio). Con el anterior ejemplo se puede notar cómo el método asiste al docente en el cálculo de NC cuando en él intervienen múltiples términos lingüísticos. Además, debido a las características de las *2-tuplas*, el método permite cuantificar qué tan cerca se encuentra un estudiante de un nivel de competencia y otro. Por ejemplo, el estudiante es calificado en la competencia Cb con un NC igual a B (Bajo), pero según las *2-tuplas* derivadas del cómputo, $(MB, 0.55)$ y $(B, -0.45)$, tiene un nivel de pertenencia bastante alto al nivel MB (Muy Bajo). En cambio, para la competencia A, el estudiante tiene un NC igual a M (Medio), y según las *2-tuplas* correspondientes, $(M, 0.08)$ y $(A, -0.92)$, tiene un nivel de pertenencia bajo al nivel A (Alto). En conclusión, el método además de obtener los NC en términos lingüísticos, ofrece la

información necesaria para que los docentes cuantifiquen qué tan cerca está un estudiante a un NC y otro. De cualquier forma, el método contribuye en parte a que los docentes tomen decisiones fundamentadas sobre sus procesos de enseñanza en busca de la mejora del aprendizaje de los estudiantes. Los NC computados son utilizados por el SR propuesto en el presente trabajo para construir perfiles de usuario lo suficientemente precisos para obtener recomendaciones personalizadas.

En síntesis, aquí se propuso un *método computacional basado en información cualitativa para la evaluación de competencias considerando la estructura de los mapas de competencias*, el cual ha sido denominado DFS-CA (*“Depth First Search-Competency Assessment”*). Éste ha sido el resultado de adaptar el modelo matemático de *2-tuplas* en el algoritmo DFS, asegurando así un recorrido sistemático y organizado por los mapas al tiempo que se ejecuta la matemática necesaria para calcular los *niveles de competencia*. Este método utiliza como insumo los mapas formalizados en un currículo educativo y las calificaciones obtenidas por los estudiantes en actividades de evaluación como talleres, exámenes y mesas redondas; por lo cual, es propicio para identificar las necesidades educativas de los estudiantes. Precisamente, esta característica es aprovechada en el SR propuesto para enfocar las recomendaciones en el aprendizaje de las competencias. Este y otros aspectos, son tratados en el siguiente capítulo.

5.3 Evaluación del método computacional

Para evaluar el método computacional se llevaron a cabo una serie de experimentos enfocados en responder las siguientes preguntas de investigación:

1. ¿Cuál es la precisión del método computacional propuesto en comparación con un *Modelo Basado en el Principio de Extensión* (MBPE)?
2. ¿Qué ventajas ofrece el método computacional en términos de tiempo de ejecución en comparación con el MBPE?
3. ¿Qué ventajas ofrece el método computacional propuesto en cuanto a consumo de memoria con respecto al MBPE?

La evaluación se realizó con respecto al MBPE porque es el modelo computacional de referencia usado en la computación lingüística [66]. Para responder las preguntas de investigación planteadas, se construyó una herramienta Java denominada *CMA-Tool*⁸ (*“Competency Map Assessment Tool”*), la cual es un aporte adicional de la presente investigación. *CMA-Tool* ofrece una interfaz gráfica (Figura 5.5) que permite configurar

⁸ El código de CMA-Tool puede ser descargado de: <https://mega.nz/#F!CvhxzShlItRoEGEJBHLBdCVin7y2b8Q>

5. Método computacional basado en información cualitativa para la evaluación de competencias

la estructura de los mapas de competencias usando la especificación *Formato de Grafos JSON* [108] (áreas a y c). La interfaz muestra la estructura del mapa resultante (área b). Para configurar la operación de la herramienta, se introduce el número de estudiantes (área d) y posteriormente, se definen uno a uno los niveles de competencias de los estudiantes en cada una de las actividades de evaluación formalizadas en el mapa (área e).

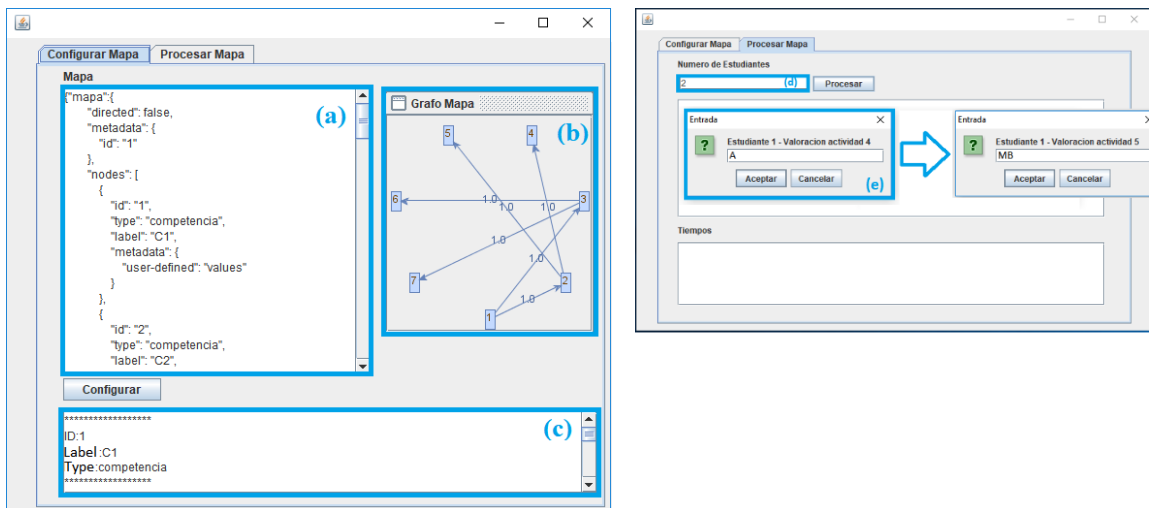


Figura 5.5 Interfaz principal de la herramienta *CMA-Tool*. Fuente: propia.

Finalmente, la herramienta ejecuta el algoritmo DFS-CA (Sección 5.2.2) propuesto para ofrecer al usuario (Figura 5.6) los *niveles de competencia* computados para cada una de las competencias del mapa (área f), usando tanto el método computacional (*modo 1*) como el MBPE (*modo 2*). Además, también para los dos modelos, la herramienta despliega dos gráficos que muestran el tiempo de ejecución (área h) y el consumo de memoria (área i). Los resultados de tiempo también se muestran en texto (área g). Una descripción más detallada de *CMA-Tool*, incluyendo la funcionalidad provista y tecnologías usadas, se presenta en el Anexo L (Sección L.1). Además, una vista funcional, una descripción de los componentes desarrollados y una vista dinámica del prototipo se presenta con mayor detalle en el Anexo O (Sección O.1).

En cuanto a los datos usados, se coleccionaron los de 85 mapas de competencias de procesos educativos reales, los cuales fueron construidos y provistos por un grupo de docentes basándose en estándares educativos de Matemáticas⁹ [53]. A fin de explicar de forma clara los experimentos y resultados alcanzados, vale la pena considerar la diferencia entre las estructuras de los mapas, la cual está constituida por competencias, actividades de evaluación, sus relaciones y pesos; mientras que el

⁹ Las estructuras de los mapas de competencias y los mapas de competencias para experimentación se pueden descargar de: <https://mega.nz/#F!i2xS1BbD!qRv-bhBVUMpu46FRDIdqwg>

mapa es el conjunto estructura-niveles de competencia. En consecuencia, una estructura puede ser aplicada a un grupo de estudiantes, mientras que el mapa de competencias es individual. Así las cosas, los 85 mapas de competencias obedecen a 17 estructuras diferentes. Éstos además involucran 1870 niveles de competencias alcanzados por los estudiantes en las AE. Estos NC son usados en los experimentos para computar los NC en las competencias. Entre los datos coleccionados, están los NC provistos por los docentes, los cuales fueron comparados con los computados por *CMA-Tool* para evaluar la precisión del método computacional propuesto. La Tabla 5.1 resume la estadística de la colección de datos.



Figura 5.6 Interfaces que presentan los resultados de la evaluación de competencias.
Fuente: propia.

Datos	
Mapas de competencias	85
Estructuras de mapas	17
Niveles de competencia de actividades de evaluación	1870
Niveles de competencia en las competencias	810

Tabla 5.1 Estadística de la colección de datos usada para la evaluación del método computacional para la evaluación de competencias. Fuente: propia.

Las descripciones de las estructuras de los mapas fueron coleccionadas por medio de un formato como el mostrado en el ejemplo de la Figura 5.7. En él, las estructuras, las competencias y las actividades de evaluación tienen un identificador. Además, los elementos de una columna, ya sean competencias o actividades, son subelementos de los que están en la columna de la izquierda. Teniendo en cuenta el ejemplo de la Figura 5.7, las competencias SS-PN-061 y SS-PN-062 son subcompetencias de SS-PN-06. Asimismo, las actividades de evaluación AE252, AE253, AE254 y AE255 son subelementos de la competencia SS-PN-061.

5. Método computacional basado en información cualitativa para la evaluación de competencias

Para coleccionar los mapas de competencias de los estudiantes, se utilizó un formato como el presentado en la Figura 5.8. Se observa cómo se hace referencia a estructuras de mapas, competencias, y AE por medio de los identificadores. Se definen tanto los pesos de las relaciones entre elementos y subelementos, como los NC alcanzados.

Id mapa	Competencias	Competencias	Actividades de evaluación
M10-SS	SS-PN-06: Justifico procedimientos aritméticos utilizando las relaciones y propiedades de las operaciones.	SS-PN-061: Aplico relaciones y propiedades de las operaciones aritméticas en ejercicios	AE252
			AE253
			AE254
		SS-PN-062: Recuerdo las relaciones y propiedades de las operaciones aritméticas	AE255
			AE256
			AE257
AE258			

Figura 5.7 Formato usado para la descripción de la estructura de los mapas. Fuente: propia.

Id estructura: M10-SS

Id estudiante: 1-SS

Actividades de Evaluación	Peso	NC	Competencia	Peso	NC	Competencia	NC
AE252	VH	M	SS-PN-061	VH	M	SS-PN-06	M
AE253	VH	M					
AE254	H	M					
AE255	H	M	SS-PN-062	H	M		
AE256	H	M					
AE257	H	M					
AE258	VH	M					

Figura 5.8 Formato usado para la descripción de los mapas de competencias. Fuente: propia.

Para obtener los resultados de la evaluación se desarrollaron una serie de fases, descritas de forma resumida así (una descripción más detallada de las fases de la experimentación se presenta en el Anexo L): inicialmente, las estructuras de los mapas fueron introducidas en *CMA-Tool*. Después, se configuró el número de estudiantes por cada estructura de los mapas. Finalmente se introdujeron los NC de cada estudiante alcanzados en cada AE.

Con el propósito de medir la precisión del método propuesto con respecto al modelo basado en el MBPE, los NC computados por *CMA-Tool* fueron comparados con los provistos por los docentes. Posteriormente, el porcentaje de aciertos fue medido mediante la expresión (5.4).

$$A = \frac{N_{\text{aciertos}}}{T} * 100\%, \quad (5.4)$$

donde N_{aciertos} es el total de aciertos y T es el total de NC evaluados.

La Tabla 5.2 presenta los resultados obtenidos. Del análisis de los datos se concluye que, del total de competencias evaluadas, el método propuesto fue preciso en el 99.4% de los casos, mientras que el MBPE lo fue en el 97.8%.

Estructura	Competencias por estructura	T	Método propuesto		MBPE	
			$N_{aciertos}$	A(%)	$N_{aciertos}$	A(%)
1	8	40	39	97.5	36	90
2	6	30	28	93.33	27	90
3	15	75	72	96	71	94.66
4	7	35	34	97.14	32	91.42
5	11	55	54	98.18	53	96.36
6	3	15	15	100	15	100
7	11	55	55	100	55	100
8	10	50	50	100	50	100
9	24	120	119	99.16	119	99.16
10	18	90	89	98.88	88	97.77
11	3	15	15	100	15	100
12	10	50	49	98	49	98
13	3	15	15	100	15	100
14	9	45	44	97.77	43	95.55
15	11	55	55	100	54	98.18
16	8	40	40	100	39	97.5
17	5	25	25	100	25	100
Total	149	810	798	99.4	787	97.8

Tabla 5.2 Precisión del método para la evaluación de competencias propuesto y del MBPE para cada estructura de los mapas de competencias. Fuente: propia.

Dado que la diferencia en los resultados es aparentemente pequeña, se hizo una prueba de significancia estadística para confirmarlo o refutarlo. Para esto, se considera como hipótesis nula (H_0) *no hay diferencia significativa entre los resultados de precisión de los dos modelos*; y como hipótesis alternativa (H_a) *existe diferencia significativa entre los resultados de precisión de los dos modelos*. Además, se tuvieron en cuenta medidas como las probabilidades promedio entre los enfoques $p = \frac{A_1 + A_2}{2} = 0,986$, donde $A_1 = 0,994$ y $A_2 = 0,978$ son las probabilidades de acierto de cada modelo; un nivel de significancia $f = 0.5$; valores estadísticos críticos en la distribución normal $Z_{\alpha=0.05} = 1.96$, y el error estándar $es = \sqrt{p(1-p)(2/T)}$, donde $T = 810$ es el total de medidas realizadas. Se calcula que $p = 0.986$ y $es = 0.00583$. Para concluir acerca de las hipótesis, se comprueba si $|A_1 - A_2| > se * Z_{\alpha=0.05}$. Al reemplazar los datos, se da que $|A_1 - A_2| = 0.0148$ es mayor que $es * Z_{\alpha=0.05} = 0.0114$, por lo cual se da como cierta la hipótesis alternativa, señalando que la diferencia de precisión entre

5. Método computacional basado en información cualitativa para la evaluación de competencias

el método computacional propuesto y el MBPE es estadísticamente significativa. Esto se debe básicamente a que el tamaño de la muestra (810 niveles de competencia) hace que el error estándar inherente a la experimentación se reduzca. De acuerdo a [111], dado que el error estándar ($se = 0.00583$) es menor que 0,01, entonces se concluye que el método propuesto tiene una precisión significativa sobre el MBPE con una seguridad inherente del 99%.

Estos resultados demuestran que, en un entorno educativo real, tanto el método computacional como el prototipo *CMA-Tool* propuestos pueden ser utilizados de forma confiable por los docentes. Aunque el estudio se centró en competencias de matemáticas, cabe destacar que el método computacional no depende de la sintáctica y semántica de las competencias, sino que se centra exclusivamente en la estructura de los mapas, cuya calidad depende exclusivamente de los docentes. En consecuencia, los resultados y análisis mostrados aquí pueden ser extrapolados a otras áreas diferentes a las matemáticas.

Finalmente, con el propósito de conocer las ventajas del método propuesto en términos de tiempo de ejecución y consumo de memoria en comparación con el MBPE, todos los mapas fueron computados 100 veces para medir el tiempo de ejecución y consumo de memoria promedio de cada estructura y modelo mediante las expresiones (5.5) y (5.6). Cabe mencionar que las medidas fueron tomadas para los métodos Java del prototipo que están directamente relacionados con los procesos propios de cada modelo.

$$t_j = \frac{\sum_{i=1}^{100} t_{ij}}{100}, \quad (5.5)$$

$$MEM_j = \frac{\sum_{i=1}^{100} CPU_{ij}}{100}, \quad (5.6)$$

donde t_{ij} y MEM_{ij} es la i -ésima medida de tiempo y consumo de memoria respectivamente, tomadas para el experimento j .

La Figura 5.9 muestra los resultados de tiempos de ejecución obtenidos (en microsegundos). Se observa que, en comparación con el MBPE, el método propuesto presentó menores tiempos en el 100% de los casos. En promedio, el método propuesto consumió un tiempo de 51,004 μs por mapa; mientras que el MBPE consumió uno de 64,135 μs . Así las cosas, la diferencia en tiempo de ejecución es favorable para la propuesta en 13,13 μs .

Algo similar ocurre con las medidas de memoria (en Bytes) presentadas en la Figura 5.10, siendo las del método propuesto menores que la mitad de las medidas del MBPE en el 100% de los casos. En promedio, el método propuesto consumió 339,294 Bytes

por mapa; mientras que el MBPE consumió 727,176 Bytes. Entonces, el método propuesto consumió en promedio 387,882 Bytes menos que el MBPE. Esto se debe especialmente a la simpleza del enfoque de *2-tuplas* y a que JSON es un formato liviano. Los resultados indican que tanto el método computacional como el prototipo *CMA-Tool* propuestos son viables para ser integrados en plataformas de aprendizaje electrónico, servicios web o aplicaciones de escritorio.

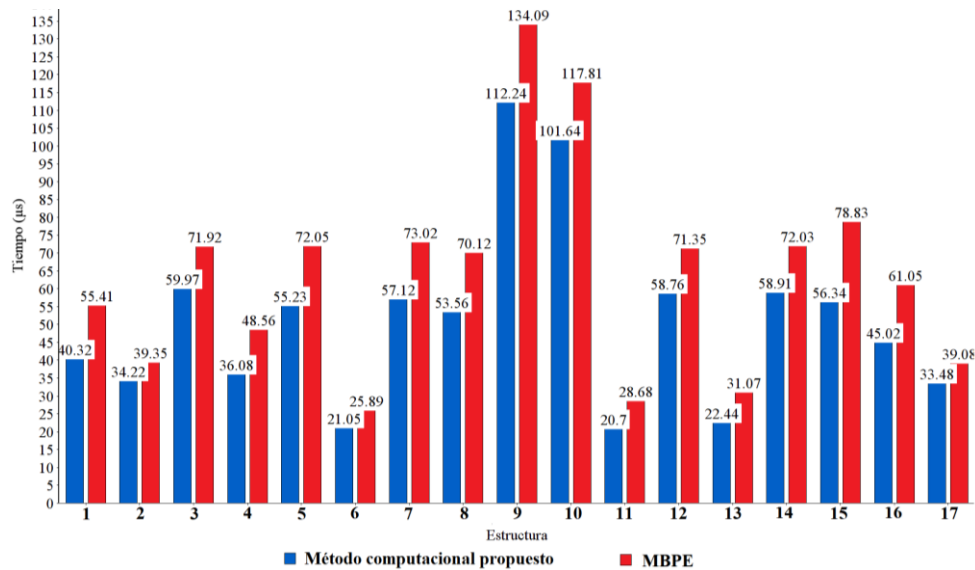


Figura 5.9 Tiempos de ejecución del modelo propuesto y el MBPE para cada una de las estructuras de los mapas. Fuente: propia.

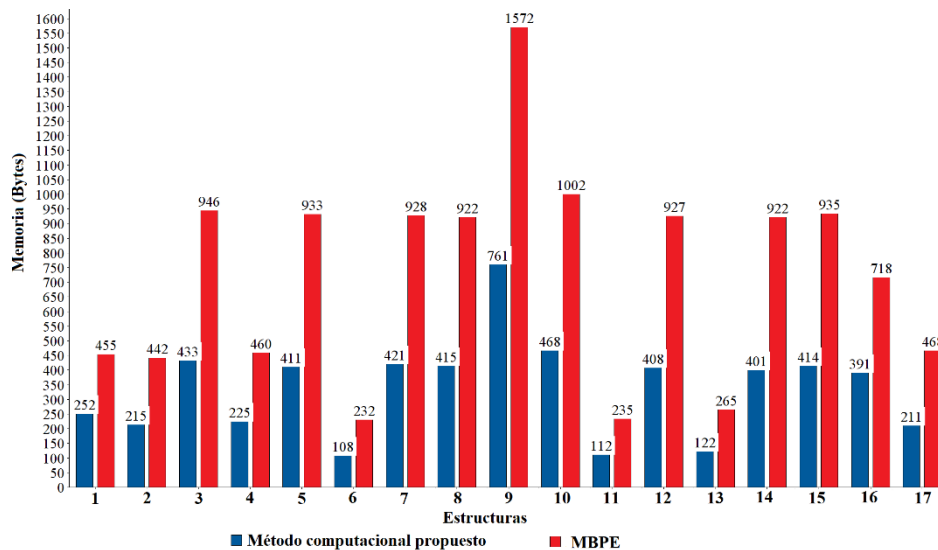


Figura 5.10 Consumo de memoria del modelo propuesto y el MBPE para cada una de las estructuras de los mapas. Fuente: propia.

Capítulo 6

Estrategia de recomendación

En el presente trabajo, el *Sistema de Recomendaciones* (SR) propuesto se apoya en una *estrategia de recomendación* enfocada en apoyar el aprendizaje de competencias por medio de la sugerencia de contenidos de VoD. Es así como, el contexto tiene un papel importante dentro de la investigación, proveyendo información acerca de las necesidades del usuario y del currículo educativo que puede ser aprovechada en la construcción de perfiles y, por ende, en el proceso de recomendación. Precisamente, estos aspectos son abordados por los componentes propuestos en los anteriores capítulos. Por ejemplo, los aspectos asociados al currículo, como las competencias y los mapas de competencias son cubiertos en el esquema de metadatos (Capítulo 3) y la ontología (Capítulo 4). Mientras que las necesidades educativas son abordadas en el método computacional (Capítulo 5). Desde el punto de vista funcional, el SR “orquesta” todos los componentes mencionados para, por medio de la aplicación de técnicas de la web semántica, generar las listas de recomendaciones.

Con respecto al contexto, de acuerdo con la clasificación definida en [23], el SR propuesto se enmarca en las categorías *conocimiento/rendimiento* y *metas de aprendizaje* de la dimensión del usuario.

Como un paso inicial en la definición de la estrategia de recomendación, inicialmente se presenta el principio base de esta:

Recomendar contenidos relacionados con competencias vinculadas semánticamente con las competencias en las cuales los estudiantes tienen cierto grado de necesidad educativa.

Para explicar las implicaciones de una estrategia de recomendación así, se considera una situación en la que un estudiante tiene un NC *Bajo* en una competencia C, por lo cual se entiende que existe cierto grado de necesidad educativa que debe ser abordado. Según el principio base de la estrategia de recomendación planteado, el SR busca inicialmente competencias vinculadas semánticamente con C, es decir, que comparten propiedades tales como el *verbo de acción*, el *esquema de cualificación*, o el *tipo de competencia* (cualquiera de los atributos de las competencias definidos en

la ontología). Estos vínculos, que incluyen tanto competencias como sus propiedades conforman las denominadas *asociaciones semánticas complejas*. Posteriormente tales asociaciones son analizadas usando técnicas de cálculo de intensidad semántica para hallar las competencias más relevantes para C. Finalmente, a partir de tales competencias se descubren los contenidos relacionados con tales asociaciones, los cuales son organizados para entregar la(s) lista(s) de recomendaciones.

El análisis de *relaciones semánticas* formalizadas en una ontología es un tema tratado en la literatura, específicamente en la *búsqueda semántica en la web*. En una revisión del estado del arte se identificaron tres enfoques de interés denominados *OntoLook* [112], *SemDis* [113] y *SemRank* [114]. Dado que en los trabajos de la literatura revisada no fue posible hallar evidencia sobre evaluaciones de rendimiento, fue necesario analizar las prestaciones que ofrecen para seleccionar un enfoque que ofrezca un modelo matemático adaptable a las características del trabajo. En concreto, se analizó la posibilidad de incorporar todas las características del contexto involucradas en la presente investigación: 1) las fortalezas y debilidades de los estudiantes; 2) aspectos del currículo educativo tales como competencias y mapas de competencias. Al respecto, a diferencia de los demás enfoques, *SemDis* provee una variedad de métricas que permiten adaptar tanto aspectos de la perspectiva del usuario (por ejemplo, su contexto) como aspectos ontológicos (por ejemplo, el número y conectividad de entidades y relaciones). Al respecto, según [115] *SemDis* es uno de los mejores enfoques debido a la variedad de sus métricas.

De acuerdo con lo anterior, el presente trabajo explota el concepto de *asociación semántica compleja* propuesto por *SemDis* [62] (más información sobre éste se puede ver en el Anexo A, Sección A.3). En *SemDis*, el enfoque de inferencia consiste en *descubrir todas las asociaciones semánticas entre dos instancias de la base de conocimiento, siendo éstas definidas por el usuario del sistema*. Este enfoque está relacionado exclusivamente con las búsquedas en la web, por lo cual no puede ser usado en la estrategia de recomendación por las siguientes razones:

- El enfoque de *SemDis* requiere la participación del usuario para definir dos instancias de la base de conocimiento, lo cual no es consistente con los niveles de automatización requeridos en los SR.
- En *SemDis* no se consideran los niveles de personalización requeridos en los SR.

Por lo anterior, en lugar de usar el enfoque original de *SemDis*, se ha definido el *flujo de trabajo* mostrado en la Figura 6.1, el cual nace de la necesidad de aprovechar el concepto de *asociación semántica compleja* en el ámbito de las recomendaciones y explotar las posibilidades que ofrecen los resultados alcanzados hasta aquí en la investigación. El flujo de trabajo se explica a continuación:

6. Estrategia de recomendación

1. Los docentes formalizan las descripciones de contenidos de VoD, competencias y mapas de competencias definidos en un currículo educativo usando el *esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias* propuesto en el Capítulo 3. Esto a través de una herramienta de marcación independiente del SR. Además, se introducen las calificaciones alcanzadas por los estudiantes en las actividades de evaluación.
2. Se crean los *registros de metadatos*, los cuales son introducidos al SR a través de una *interfaz de usuario*.
3. El usuario define el *estudiante objetivo*, este es, el estudiante al cual se van a generar las recomendaciones.
4. Los *registros de metadatos* son adaptados a una *base de conocimiento* de acuerdo con la *ontología sobre el dominio de los contenidos de VoD y las competencias* propuesta en el Capítulo 4. Esto posibilita la aplicación de técnicas de la web semántica en la obtención de las recomendaciones.
5. Se realiza una evaluación de competencias de acuerdo al *método computacional basado en información cualitativa para la evaluación de competencias* propuesto en el Capítulo 5.

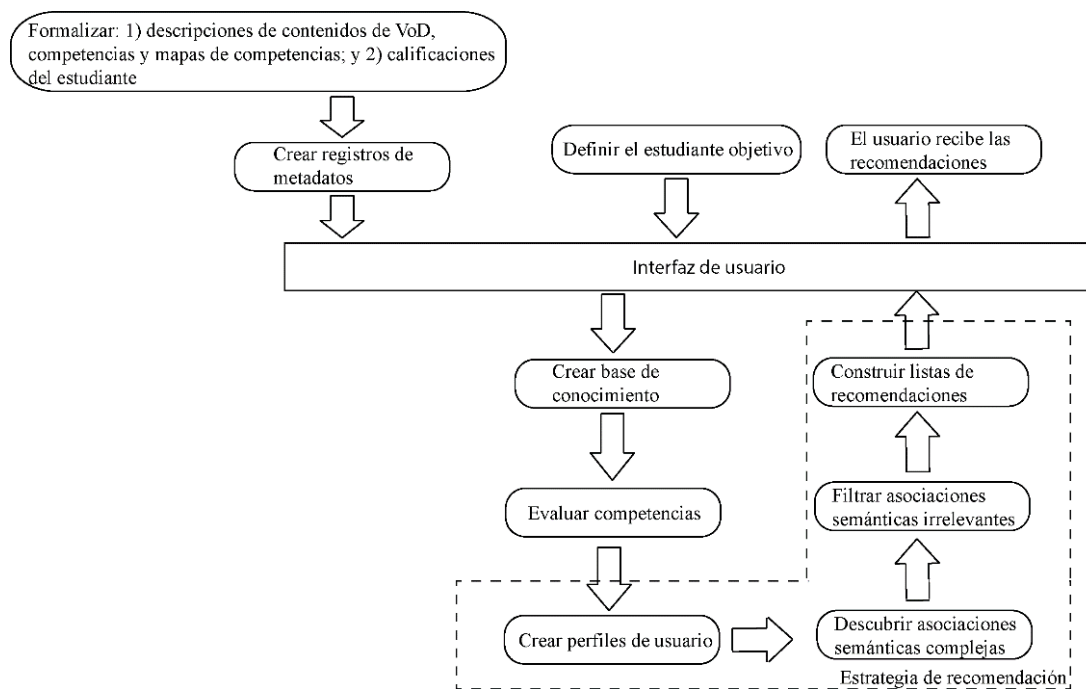


Figura 6.1 Flujo de trabajo del sistema de recomendación propuesto

Los siguientes pasos son propios de la estrategia de recomendación:

6. Se construyen *perfiles de usuario* de acuerdo a los resultados de la evaluación de competencias.

7. Se descubren las *asociaciones semánticas complejas* relacionadas con las competencias en las que hay necesidades educativas.
8. Se filtran las *asociaciones semánticas* irrelevantes con base en un proceso de *ponderación*.
9. Las competencias descubiertas y los resultados de la ponderación son usados para construir las *listas de recomendaciones*, una por cada competencia en la que el estudiante tiene necesidades educativas.
10. Finalmente, el SR provee al usuario las *listas de recomendaciones*.

Teniendo en cuenta que los primeros cinco pasos del flujo se relacionan con los capítulos anteriores, queda por tratar los procesos relacionados directamente con la obtención de las recomendaciones, estos son: *crear perfiles de usuario, descubrir asociaciones semánticas complejas, filtrar asociaciones semánticas irrelevantes y construir las listas de recomendaciones*.

En el presente capítulo, inicialmente se presenta el modelo del SR propuesto. Posteriormente, se describen detalladamente los procesos del flujo de trabajo relacionados con la obtención de las recomendaciones.

Con respecto a los aportes del capítulo al conocimiento, se tiene:

1. Una estrategia de recomendación que adapta el enfoque de búsquedas semánticas de *SemDis* [62] a las recomendaciones en el ámbito educativo de las competencias.
2. Un modelo de perfiles de usuario producto de la adaptación del modelo *perfil-ontología* [116] que formaliza las necesidades de los estudiantes en cuanto a las competencias y sus atributos.
3. Un grupo de métricas para la ponderación de asociaciones semánticas, resultado de la adaptación tanto de las métricas de *SemDis* [62], como de trabajos como [20] y [117]; de forma que representen aspectos ontológicos (longitud de las asociaciones y similitudes semánticas) y del contexto (fortalezas y debilidades educativas e intereses del usuario, relaciones de subsunción en los mapas).

Resultados parciales de este capítulo fueron publicados en [118] y [119].

6.1 Modelo del sistema de recomendaciones

La Figura 6.2 presenta el modelo propuesto para el SR de acuerdo a un diagrama de componentes UML, al cual se han añadido los actores involucrados. Con respecto a la solución del sistema introducida en la motivación del presente trabajo, se muestran los componentes específicos que hacen parte de la estrategia de recomendación.

6. Estrategia de recomendación

En éste, se han integrado un grupo de servicios/herramientas que, de manera informativa, sitúan al SR dentro de un entorno real de uso. Éstos no fueron desarrollados en la investigación, sin embargo, se hace una descripción corta de cada uno de ellos.

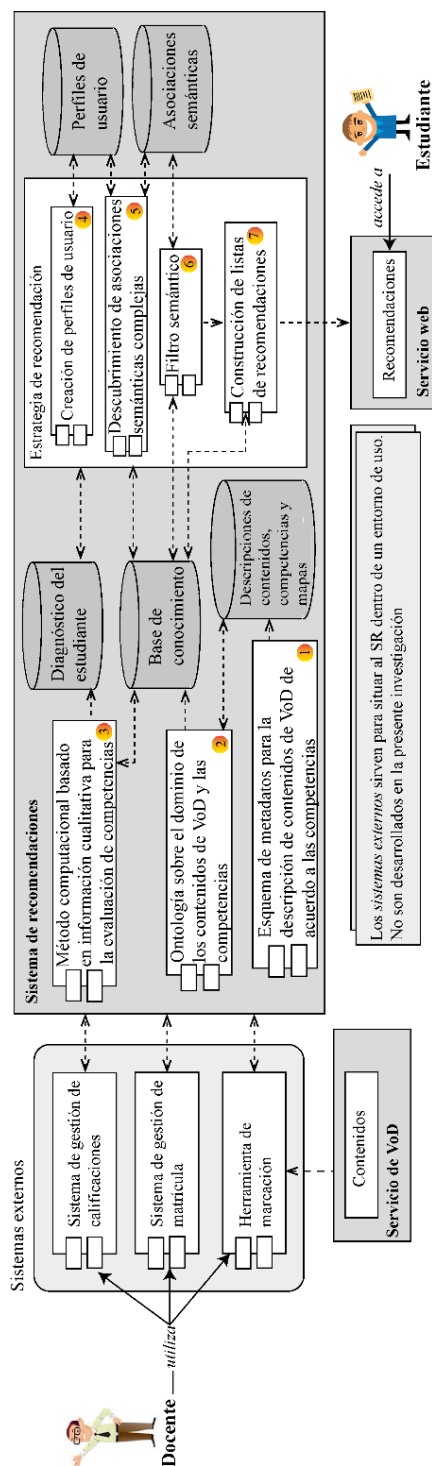


Figura 6.2 Modelo del sistema de recomendaciones propuesto. Fuente: propia.

1. *Sistema de gestión de calificaciones*: son sistemas comúnmente desplegados en instituciones educativas por medio de los cuales los docentes registran las calificaciones alcanzadas por los estudiantes en las *Actividades de Evaluación (AE)*.
2. *Sistema de gestión de matrícula*: son sistemas usados por las instituciones para registrar la información de los usuarios matriculados en un programa educativo o curso.
3. *Herramienta de marcación*: herramienta software usada por los docentes para describir competencias, mapas de competencias y contenidos extraídos de servicios de VoD tales como Youtube. Esta herramienta se basa en el *Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias* para proveer las interfaces y la lógica involucrada.

Toda la información obtenida por servicios como los mencionados anteriormente es aprovechada por el SR mediante los siguientes componentes:

1. *Esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias* (Capítulo 3). Este componente es un esquema de metadatos que puede ser usado por herramientas de marcación para la descripción de contenidos, competencias y mapas. Los registros de metadatos son almacenados en la base de datos *Descripciones de contenidos, competencias y mapas*. Además, es usado como referencia para mapear las descripciones en la *Base de conocimiento*.
2. *Ontología sobre el dominio de los contenidos de VoD y las competencias educativas* (Capítulo 4). Ontología que establece atributos de, y relaciones entre contenidos de VoD, competencias y mapas de competencias. Con base en ella, y el esquema de metadatos, se crea una *Base de conocimiento* que formaliza tales entidades, de forma que la estrategia de recomendación pueda aplicar técnicas de la web semántica. El proceso de mapeo entre registros de metadatos y base de conocimiento está fuera del alcance de la investigación.
3. *Método computacional basado en información cualitativa para la evaluación de competencias* (Capítulo 5). Componente encargado de utilizar los mapas de competencias formalizados en la base de conocimiento y las calificaciones registradas por los docentes para calcular los NC en todas las competencias de los mapas. Los resultados son registrados en una base de datos denominada *Diagnóstico del estudiante*.

Los siguientes componentes son específicos de la estrategia de recomendación.

4. *Creación de perfiles de usuario*. Componente encargado de crear los perfiles de usuario con base en las necesidades educativas. Este proceso se ejecuta sobre los resultados de la evaluación cualitativa de competencias registrados en la base de

6. Estrategia de recomendación

datos *Diagnóstico del estudiante*. Los perfiles son registrados en la base de datos *Perfiles de usuario*.

5. *Descubrimiento de asociaciones semánticas complejas*. Componente encargado de descubrir las asociaciones semánticas entre competencias formalizadas en la *Base de conocimiento*. Las asociaciones descubiertas son registradas en la base de datos *Asociaciones semánticas*.
6. *Filtro semántico*. Componente encargado de seleccionar asociaciones semánticas relacionadas con las necesidades de los estudiantes. Para esto, se hace uso de una adaptación de las técnicas de ponderación de *SemDis* [62]. Se infiere que las asociaciones mejor ponderadas son las más relacionadas con las necesidades de cada estudiante, por lo cual son las más relevantes dentro del proceso de recomendación.
7. *Construcción de listas de recomendaciones*. Se encarga de descubrir y ponderar contenidos de video considerando las asociaciones semánticas más relevantes para cada usuario. Con base en los resultados, se encarga finalmente de organizar las listas de recomendaciones.

Ahora, para explicar cómo se relacionan los componentes durante la ejecución de la estrategia de recomendación se presenta el diagrama de secuencias de la Figura 6.3. Todo el proceso se explica a continuación.

1. *Creación de perfiles de usuario* consulta los NC alcanzados por los estudiantes en las competencias, los cuales están registrado en la base de datos *Diagnóstico del estudiante*.
2. *Creación de perfiles de usuario* consulta la *base de conocimiento*.
3. *Creación de perfiles de usuario* calcula los grados de necesidad de los estudiantes por las competencias y sus atributos, para lo cual se asocian los NC consultados en (1) con la estructura de la ontología consultada en (2). De acuerdo con [116], este modelo es nombrado como perfil-ontología. Todo el proceso se explica con detalle en la Sección 6.2.
4. *Creación de perfiles de usuario* registra los perfiles en la base de datos *Perfiles de usuario*.
5. *Descubrimiento de asociaciones semánticas complejas* consulta la *base de conocimiento*.
6. *Descubrimiento de asociaciones semánticas complejas* descubre de la *base de conocimiento* las asociaciones semánticas entre competencias. Este proceso se explica en la Sección 6.3.

7. *Filtro semántico* identifica las asociaciones semánticas relacionadas con competencias en las que el estudiante objetivo tiene un grado de necesidad alto. Información al respecto se encuentra en la Sección 6.4.
8. *Filtro semántico* filtra las asociaciones semánticas menos relevantes para el usuario objetivo. Para esto, realiza un proceso de ponderación de las asociaciones semánticas obtenidas en (7) de acuerdo a un grupo de métricas asociadas a aspectos ontológicos (longitud de las asociaciones y similitudes semánticas) y del contexto (fortalezas y debilidades educativas e intereses del usuario, y relaciones de subsunción en los mapas). Posteriormente, aplica un criterio sobre las ponderaciones de las asociaciones para excluir las menos relevantes. Este proceso se explica detalladamente en la Sección 6.4.1.

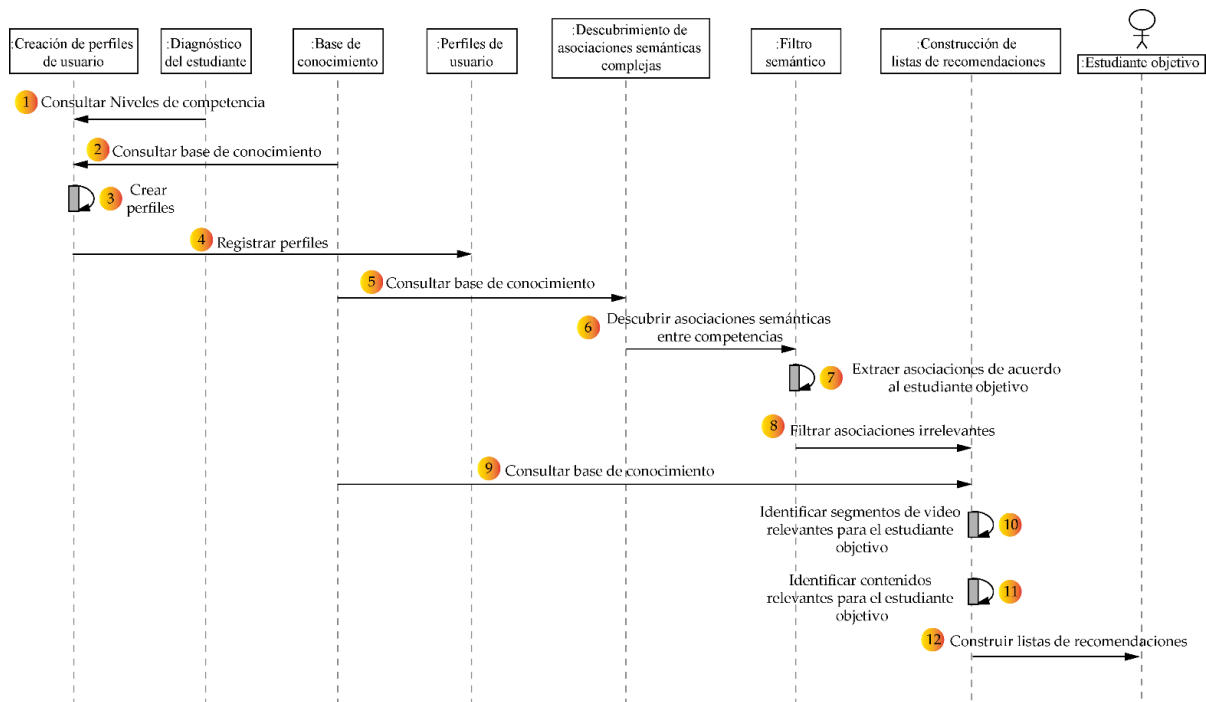


Figura 6.3 Diagrama de secuencias que explica la inter-relación de los componentes durante la ejecución de la estrategia de recomendación. Fuente: propia.

9. *Construcción de listas de recomendaciones* extrae de las asociaciones semánticas relevantes identificadas en (8) las competencias más relevantes para el usuario objetivo. Este proceso se explica en la Sección 6.5.
10. *Construcción de listas de recomendaciones* extrae de la *base de conocimiento* los segmentos de video asociados a las competencias relevantes identificadas en (9). Después, los pondera de acuerdo a los resultados de ponderación de las asociaciones semánticas obtenidos en (8), obteniendo así los segmentos de video más relevantes para el usuario objetivo. Este proceso se explica en la Sección 6.5.

11. *Construcción de listas de recomendaciones* extrae de la *base de conocimiento* los contenidos de video asociados a los segmentos de video más relevantes identificados en (11). Después, los pondera de acuerdo a las ponderaciones calculadas en (11). Este proceso se explica con detalle en la Sección 6.5.
12. *Construcción de listas de recomendaciones* construye las listas de recomendaciones, dando mayor relevancia a los contenidos mejor ponderados en (12). Éstas son entregadas al *estudiante objetivo*.

Todos los componentes de la estrategia de recomendación y los procesos desarrollados se presentan con detalle en las siguientes secciones.

6.2 Construcción de perfiles de usuario

La personalización es una característica inherente a los sistemas de recomendación, pues asegura que los contenidos sugeridos sean acordes con las necesidades y/o preferencias de las personas. En búsqueda de tal meta, el desarrollo de la estrategia de recomendación requiere que se defina una representación formal de las necesidades de los estudiantes que provea conocimiento a utilizar en la obtención de las recomendaciones.

Esta característica imposibilita el uso de enfoques tales como *historiales de consumo* [120], [121], *vectores de características* [122], *matrices de clasificaciones* (o *valoraciones*) [123] y *características demográficas* [124] en un sistema con los requerimientos expuestos en la presente investigación; ya que definen perfiles basados en información no relevante en el contexto educativo o son estructuras muy simples con poca expresividad de información. Por ejemplo, las palabras clave, los historiales de consumo y las clasificaciones podrían no ser un buen criterio para inferir necesidades educativas.

En cambio, un modelo como el *perfil-ontología*¹⁰ propuesto en [34] para el modelado de perfiles a partir de redes semánticas, permite representar más conocimiento sobre el usuario, identificando tanto los nodos (clases o instancias formalizados en la ontología), sus relaciones (propiedades formalizadas en la ontología) y, en este caso, los grados de necesidad en el aprendizaje que tienen los estudiantes por los nodos.

En concreto, adaptando el modelo *perfil-ontología* en la presente investigación, los perfiles del SR propuesto registran los grados de necesidad que tienen los estudiantes por el aprendizaje tanto de las competencias como de sus atributos (por ejemplo, el *verbo de acción*, el *esquema de cualificación*, el *tipo de competencia*, entre otros).

¹⁰ Modelo para la construcción de perfiles de acuerdo a una ontología

Para esto, en primera instancia define un *perfil inicial de usuario* conformado por los NC provistos por el *método computacional basado en información cualitativa para la evaluación de competencias educativas* (Capítulo 5). Posteriormente, se construyen los perfiles-ontologías. Estas tareas se explican en las siguientes secciones.

6.2.1 El perfil inicial de usuario

Dado el contexto educativo del SR, el usuario debe estar asociado al currículo educativo de un curso, lo cual implica que el perfil haga referencias a los identificadores de las competencias correspondientes. Además, considerando que el propósito del sistema es sugerir contenidos ajustados a las necesidades en cuanto a competencias se refiere, los perfiles iniciales de usuario contienen los NC alcanzados por los usuarios en el marco del currículo al cual están asociados, esto es, los NC provistos por el método computacional para la evaluación de competencias descrito en el Capítulo 5.

Esto supone que el usuario no tiene que gastar tiempo entregando información sobre sus preferencias y/o necesidades, situación que es crítica en otros SR que basan su funcionamiento en la valoración de los ítems o en información demográfica.

Una vez definidos los perfiles iniciales de usuario, se construyen los perfiles-ontología, proceso explicado en la siguiente sección.

6.2.2 Construcción de perfiles-ontología

Tal como se mencionó antes, en este trabajo se adopta el modelo perfil-ontología para crear perfiles. En concreto, permite crear perfiles con base en las características de las ontologías [34].

A continuación, se listan las características requeridas en los perfiles-ontología en el contexto del trabajo:

1. Deben formalizar el grado de necesidad educativa por las competencias y sus propiedades semánticas. Esto representa la definición de un valor numérico que pueda ser usado a través de la matemática asociada a la web semántica. Aquí, el grado de necesidad se ve como una adaptación al contexto educativo de las tradicionales clasificaciones (clasificaciones, valoraciones, “likes”, “dislikes”).
2. Muestra las instancias de competencias y propiedades semánticas, así como las relaciones entre éstas. Ésta es la base del modelo de perfiles basado en *redes semánticas* en el que los nodos están relacionados entre sí [34], por lo que se constituye en un requerimiento para construir los perfiles-ontología.
3. Considera el apoyo en el aprendizaje de las competencias a través de competencias. Aquí es importante tener en cuenta que si existe una necesidad educativa en una competencia porque el estudiante tiene un NC bajo (y, por lo tanto,

6. Estrategia de recomendación

un grado de necesidad alto), las subcompetencias pueden servir de apoyo para abordar las necesidades educativas. Esta información puede ser aprovechada en la estrategia de recomendación para enfocar las sugerencias en el apoyo de competencias vinculadas a necesidades educativas.

4. Se adaptan de forma dinámica a los cambios en las necesidades educativas. Este es un requerimiento formulado en cualquier SR para ofrecer recomendaciones pertinentes [116].

A partir de las anteriores características, se define un grupo de aspectos que influyen en la construcción de los perfiles-ontología:

1. En los trabajos de la literatura revisados no fue posible hallar una métrica que permita calcular numéricamente los grados de necesidad educativa. En consecuencia, fue necesario definir una justificadamente.
2. La *ontología del dominio de los contenidos de VoD y las competencias educativas* propuesta en el Capítulo 4 define tanto competencias y sus propiedades a través de instancias, como sus relaciones a través de propiedades. Esto, junto a los grados de necesidad educativa son los insumos necesarios para la formalización de perfiles-ontología.
3. La *ontología del dominio de los contenidos de VoD y las competencias educativas* formaliza las relaciones de subsunción propias de los mapas de competencias. Esto puede ser aprovechado para apoyar el aprendizaje de una competencia con un grado de necesidad alto a través de sus subcompetencias.
4. La adaptación dinámica de perfiles educativos en los SR es un tema que no fue posible hallar en los trabajos de la literatura revisados. En consecuencia, es necesario sentar las bases para modificar los perfiles de tal forma que las recomendaciones sean pertinentes con el paso del tiempo.

Estos aspectos son tratados con detalle en las secciones subsecuentes.

6.2.2.1 El grado de necesidad

Los perfiles-ontología se construyen con base en una variable creada en la presente investigación denominada *Grado de Necesidad (GN)*, el cual es la *ponderación del interés del usuario por las competencias y las propiedades semánticas con respecto al interés en su aprendizaje*, siendo representado por un valor $\in [0,1]$. Dado el principio base de la estrategia de recomendación, *GN* depende exclusivamente de los niveles de cumplimiento definidos en el perfil inicial de cada estudiante.

El modelo de construcción de los perfiles se muestra en la Figura 6.4, en el cual se indican dos perfiles de usuario P_u compuestos por un perfil inicial, constituido por

competencias (CI_y) y sus respectivos niveles de competencia (NC_{uy}). Éstos últimos pasan por un proceso de cálculo de los GN para las competencias (GNC_{uy}) y sus propiedades semánticas (GNP_{uP_j}). Para esto se tienen en cuenta las relaciones entre las competencias y sus relaciones (P_j).

Sea u un estudiante y x una competencia, el GN de u por x se define por medio de:

$$GNC_{ux} = \sigma\left(1 - \frac{\beta}{6}\right) \quad (6.1)$$

donde β es el índice que corresponde al NC alcanzado por u en la competencia x de acuerdo con la Figura 5.4 (esto teniendo en cuenta que los NC son términos lingüístico) y σ es un *factor de olvido* cuyo uso se explica adelante en la Sección 6.2.3. Como resultado de (6.1), el sistema pondera mejor a las competencias en las que el usuario ha obtenido NC bajos y peor a las competencias con NC altos.

Una vez se ha establecido a GN como parámetro principal en la construcción de perfiles-ontología, a continuación se explica su cálculo tanto para las competencias como para las propiedades semánticas.

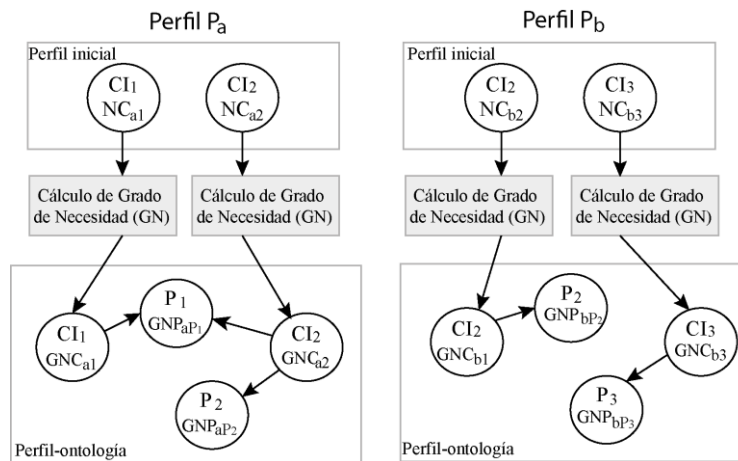


Figura 6.4 Modelo para la construcción de perfiles. Fuente: propia.

6.2.2.2 Grados de necesidad para las competencias de los mapas y las propiedades semánticas

Con respecto a las competencias de los mapas, hay que recordar que el método computacional provee los NC de todas las competencias de los mapas usando 2-*tuplas*, por lo cual, el cálculo de los GN para las competencias se basa en la aplicación de la expresión (6.1).

Por otra parte, con respecto al GN de las propiedades semánticas, debido a que el perfil inicial almacena los identificadores de las competencias, es posible utilizarlos para identificar las correspondientes instancias en la base de conocimiento y realizar

6. Estrategia de recomendación

la consulta de sus propiedades semánticas usando SPARQL. Una vez consultadas dichas propiedades, se procede a calcular sus GN.

Para definir un proceso de cálculo estándar, se tiene en cuenta que en la representación de conocimiento se han definido vocabularios controlados para algunas clases. Dado que estas clases están vinculadas con la clase *Competency* por medio de Propiedades de Datos (*Data properties*), se asume que sus instancias son propiedades semánticas de las competencias. Esta característica la poseen las instancias de las siguientes clases: *Verb*, *UserRole*, *Age*, *Difficulty*, *CompetencyType*, *Schema*, *Level* y *Language*. También se da que dichas propiedades pueden estar asociadas a dos o más competencias a la vez.

De forma general, el cálculo del grado de necesidad GN_{xj} de un usuario u por una propiedad semántica j se calcula por medio de la expresión (6.2). Ésta señala que si la propiedad en cuestión está asociada a n competencias a la vez, su GN es el promedio de los GN de dichas competencias para el usuario x .

$$GNP_{uj} = \frac{\sum_{i=1}^n GNC_{ui}}{n}. \quad (6.2)$$

6.2.3 Adaptación de los perfiles de usuario frente a cambios en las necesidades de los estudiantes

Los objetivos de aprendizaje en los cursos pueden variar por diferentes motivos, ante lo cual pueden cambiar las necesidades de aprendizaje de los estudiantes. Frente a esta situación, la presente estrategia de recomendación incluye el factor de olvido σ en la expresión (6.1), el cual modifica los valores del GN.

La Tabla 6.1 define cada una de las situaciones posibles y su correspondiente valor de σ . Éstos valores son extraídos del grupo de términos lingüísticos y su semántica [63] (ver Figura 2.2) el cual define para cada término su representación numérica.

Se puede notar que la mayor prioridad ($\sigma = 1$) la tienen las competencias que están siendo desarrolladas en el currículo del período académico actual (semestre, trimestre, año, etc.). También, es posible que las competencias tratadas en el período académico anterior sean subcompetencias de las desarrolladas actualmente. Ante esta situación, σ depende del peso consignado entre competencia y subcompetencia en el mapa. Esto se basa en que si un estudiante tiene deficiencias en el aprendizaje de ciertas competencias, se puede afectar el aprendizaje de otras más generales. En consecuencia, conviene seguirlas considerando, aunque varíen las metas de aprendizaje de un período académico a otro. Un análisis similar se hace para las competencias del período académico actual, pero que no son subcompetencias de las actuales. En este caso, $\sigma = 0.5$. Finalmente, la prioridad es nula para las competencias

que se han desarrollado antes de dos períodos académicos atrás, entendiendo que éstas son de poco interés. Cabe mencionar que estos valores fueron definidos bajo el criterio de un grupo de docentes, por lo cual pueden ser modificados en otras aplicaciones. El aporte más importante aquí es la inclusión del factor de olvido como una variable que permite actualizar los GN de acuerdo a características del currículo tales como las relaciones entre las competencias definidas en los mapas y las competencias definidas en los períodos escolares.

Situación	Valor de σ
Las competencias están siendo desarrolladas en el período actual	1
Las competencias fueron desarrolladas en el período académico inmediatamente anterior y son subcompetencias de las del período actual	Depende del peso de la relación entre competencia y subcompetencia en el mapa de la siguiente forma [63]: <ul style="list-style-type: none"> • $\sigma = 0.16$ si peso es <i>Very Low</i> • $\sigma = 0.33$ si peso es <i>Low</i> • $\sigma = 0.5$ si peso es <i>Medium</i> • $\sigma = 0.66$ si peso es <i>High</i> • $\sigma = 0.83$ si peso es <i>Very High</i> • $\sigma = 1$ si peso es <i>Total</i>
Las competencias fueron desarrolladas en el período académico inmediatamente anterior y no son subcompetencias de las del período actual	0.5
Las competencias fueron desarrolladas antes de dos períodos académicos atrás del actual	0

Tabla 6.1 Definición de los valores del factor de olvido. Fuente: propia.

Ejemplo 6.1. Considerando el extracto de la base de conocimiento de la Figura 4.18, un estudiante u registra en su perfil inicial un NC igual a B en la competencia *Calculo el área de algunas figuras geométricas*, un NC igual a M en *Identifico las partes de algunas figuras geométricas*, y un NC igual a MB en *Identifico algunas figuras geométricas*. Las dos primeras competencias hacen parte de un mapa de competencias *MapaCompetencias1* que está siendo trabajado en un curso actual, mientras que la tercera competencia hace parte de otro mapa *MapaCompetencias2*, el cual fue desarrollado en el período académico anterior. Con base en la anterior información, se explica el proceso desarrollado para crear su perfil-ontología. Para hacer la conversión de los NC a su representación numérica β , se hace uso de la Figura 2.2; por lo tanto, para cada NC se tiene: $\beta = 2$, $\beta = 3$ y $\beta = 1$, respectivamente. De acuerdo con los valores de σ consignados en la Tabla 6.1, como *MapaCompetencias1* es del curso actual, entonces $\sigma = 1$; así mismo, como *MapaCompetencias2* es del período académico anterior, entonces $\sigma = 0,5$. Ahora, el sistema crea un perfil-ontología de u calculando el GN para las

6. Estrategia de recomendación

competencias usando la expresión (6.1) y para las propiedades *Desarrollar*, *Skill* e *Identificar* por medio de la expresión (6.2). La forma en cómo se aplican éstas expresiones para competencias y propiedades semánticas se presentan en la Tabla 6.2 y la Tabla 6.3, respectivamente. Con base en los resultados, se obtiene el perfil-ontología mostrado en la Figura 6.5, en el cual se observa que *u* tiene necesidades de aprendizaje en el desarrollo de *habilidades* y para *identificar conceptos matemáticos*. Este tipo de conocimiento es usado por el sistema para descubrir recomendaciones adaptadas a las necesidades de los estudiantes.

Competencias	Grado de necesidad
Calculo el área de algunas figuras geométricas	$GNC_{u1} = 1 * (1 - \frac{2}{6}) = 0,66$
Identifico las partes del algunas figuras	$GNC_{u2} = 1 * (1 - \frac{3}{6}) = 0,5$
Identifico algunas figuras geométricas	$GNC_{u2} = 0.5 * (1 - \frac{1}{6}) = 0,41$

Tabla 6.2 Ejemplo del cálculo de grados de necesidad por competencias. Fuente: propia.

Propiedades semánticas	Grado de necesidad
Desarrollar	$GNP_{u1} = GNC_{u1} = \frac{0,66}{1} = 0.66$
Skill	$GNP_{u2} = GNC_{u2} = 0.9$
Identificar	$GNP_{u3} = \frac{\sum_{i=1}^2 GNC_{ui}}{2} = \frac{0,5 + 0,41}{2} = 0,45$

Tabla 6.3 Ejemplo del cálculo de grados de necesidad por propiedades semánticas.

Fuente: propia.

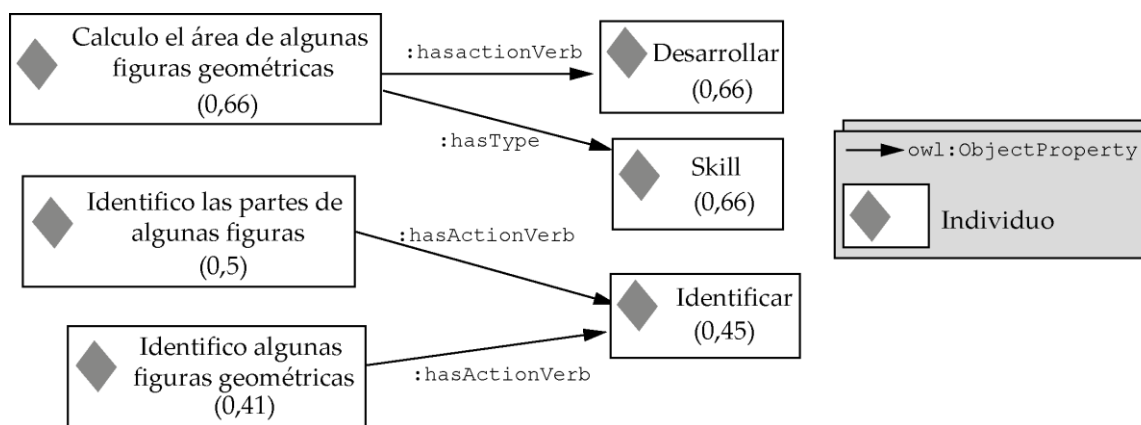


Figura 6.5 Ejemplo de un perfil-ontología. Fuente: propia.

6.3 Descubrimiento de asociaciones semánticas y secuencias de nodos

En este proceso se descubren las asociaciones semánticas complejas y las secuencias de nodos entre las competencias formalizadas en la base de conocimiento. Justamente, a la luz del principio base de la estrategia de recomendación introducido al inicio del capítulo, el descubrimiento de asociaciones semánticas permite hallar relaciones semánticas entre competencias

De acuerdo con [62], se han identificado 4 tipos de asociaciones semánticas en las ontologías, las cuales pueden ser consultadas en el Anexo A (Sección A.3.2): ρ – *pathAssociated*, ρ – *joinAssociated*, ρ – *cpAssociated*, ρ – *IsoAssociated*. Se dice que dos instancias x y y están asociadas semánticamente si se cumple entre ellas cualquiera de tales asociaciones.

Con el propósito de hacer eficiente el proceso de filtrado, se ha realizado un análisis de la ontología propuesta en el Capítulo 4 para identificar tanto los tipos de asociaciones semánticas que pueden ser relevantes como las que no. Para ilustrar las conclusiones explicadas a continuación, el lector puede apoyarse en el fragmento de ontología mostrado en la Figura 4.18.

- Las asociaciones ρ – *pathAssociated* son relevantes en este estudio porque son la esencia del establecimiento de vínculos entre las competencias, permitiendo relacionarlas a través de sus propiedades semánticas. Por ejemplo, en la Figura 4.18 las instancias (de la clase *Competency*) *Calculo el área de algunas figuras geométricas* y *Aplico la fórmula del área del círculo* están vinculadas por medio de la secuencia de propiedades $PS = hasActionVerb.isActionVerbOf$. Es evidente que a partir de estas asociaciones pueden inferirse vínculos entre ambas competencias.
- Debido a que la jerarquía de clases de la ontología es de una profundidad muy reducida, no hay clases con el suficiente significado e importancia como para descubrir asociaciones semánticas con conocimiento relevante. Para ilustrar esta situación, considerar el fragmento de la ontología presentado en la Figura 4.18. Sea $PS_1 = hasCompetencyType.isCompetencyTypeOf.hasActionVerb$ y $PS_2 = hasActionVerb$, se tiene que $NodesOfPS(PS_1) \cap NodesOfPS(PS_2) \neq \emptyset$. Si ps_1 y ps_2 son instancias de PS_1 y PS_2 respectivamente, se tiene $ps_1.PSNodesSequence() = [Aplico la fórmula del área del círculo, Skill, Calculo el área de algunas figuras geométricas, Desarrollar]$ y $ps_2.PSNodesSequence() = [Identifico algunas figuras geométricas, Identificar]$. Aunque se puede observar que hay una asociación ρ – *joinAssociated* entre ps_1 y ps_2 debido a que *Desarrollar* e *Identificar* pertenecen a

6. Estrategia de recomendación

la misma clase de unión *Verb*, se considera que ésta no tiene el significado suficiente para concluir que las competencias *Aplico la fórmula del área del círculo* e *Identifico algunas figuras geométricas* se vinculen sólo porque sus propiedades pertenezcan a la misma clase. En la práctica, esto significa que no es posible asegurar que existe un vínculo entre ambas competencias porque el usuario tiene un especial interés en el verbo como concepto, más bien lo tiene en sus instancias. En consecuencia, una asociación $\rho - joinAssociated$ tendría más importancia si la clase *Verb* tuviese una jerarquía especializada con clases que representen conocimiento más específico, y por lo tanto sean útiles para formar uniones entre asociaciones. Así las cosas, tiene más significado que dos competencias compartan el mismo verbo, traduciéndose esto en el descubrimiento de una asociación $\rho - pathAssociated$. En general, esto ocurre con todas las clases de la ontología, por lo cual se ha decidido no descubrir asociaciones $\rho - joinAssociated$ en el proceso de filtrado.

- Las asociaciones $\rho - IsoAssociated$ no fueron consideradas en el filtrado debido a que no son tenidas en cuenta las asociaciones $\rho - joinAssociated$, siendo la existencia de éstas últimas un requisito para las primeras.
- Las asociaciones $\rho - IsoAssociated$ no fueron tenidas en cuenta ya que en la ontología no hay relaciones entre propiedades de tipo *subPropertyOf* en torno a la descripción de competencias.

Habiendo establecido que el filtrado se centra en descubrir asociaciones $\rho - pathAssociated$, se define a continuación el proceso de descubrimiento de secuencias de nodos y asociaciones semánticas.

Inicialmente, conviene definir un número limitado de secuencias de propiedades para garantizar la viabilidad computacional del SR. Para esto, se identificaron las propiedades de las competencias más significativas (de acuerdo con [72]): *CompetencyType* y *ActionVerb*. Entonces, las secuencias de propiedades definidas para el descubrimiento de asociaciones semánticas son:

1. $PS_1 = [hasCompetencyType, hasCompetencyType^*]$,
2. $PS_2 = [hasActionVerb, hasActionVerb^*]$,
3. $PS_3 = [hasCompetencyType, hasCompetencyType^*, hasActionVerb, hasActionVerb^*]$,
4. $PS_4 = [hasActionVerb, hasActionVerb^*, hasCompetencyType, hasCompetencyType^*]$,

donde P_i^* es la propiedad inversa de P_i . Cabe mencionar que otras secuencias de propiedades pueden ser definidas en otras aplicaciones asegurándose que, al

aplicarlas, tanto el nodo origen como el término de cada secuencia de nodos y asociación semántica sean competencias.

Para el descubrimiento de las secuencias de nodos, como aporte de la investigación se ha construido el algoritmo NSD (“*Node Sequence Discovery*” por su significado en inglés). En términos de *SemDis*, es la aplicación de la función *ps.PSNodesSequence()* sobre las secuencias de propiedades ya definidas. Sea G la base de conocimiento, $GC = \{C_1, C_2, \dots, C_i, \dots, C_n\}$ el conjunto de instancias de competencias formalizadas en G y $PS = \{PS_1, PS_2, \dots, PS_j, \dots, PS_4\}$ el conjunto de secuencias de propiedades y $PS_j = \{P_1^j, \dots, P_i^j, P_n^j\}$ una secuencia de propiedades, NSD se explica a continuación (el pseudoalgoritmo de NSD se muestra en Algoritmo M.2, Anexo M):

1. Crear un conjunto CNS , el cual contendrá las secuencias de nodos descubiertas. Ir al paso 2.
2. Tomar una competencia $C_i \in GC$ a la cual no se le hayan descubierto sus secuencias de nodos, crear un conjunto de secuencias de nodos PS_i , el cual contendrá las secuencias de nodos para tal competencia e ir al paso 3. Si no hay una competencia así, se termina el algoritmo.
3. Tomar una secuencia de propiedades $PS_j \in PS$ que no haya sido evaluada e ir al paso 4. Si no hay una secuencia de propiedades así, se regresa al paso 2.
4. Si el tamaño de la secuencia de propiedades es igual a 3, ir al paso 5. Si no, ir al paso 11.
5. Consultar en G el conjunto N_1 de instancias x_i que cumplen la tripleta (C_i, p_1^j, x_i) . Ir al paso 6.
6. Tomar una instancia $x_i \in N_1$ que no haya sido evaluado e ir al paso 8. Si tal instancia no existe volver al paso 3.
7. Consultar en G el conjunto N_2 de instancias y_i que cumplen la tripleta (x_i, p_2^j, y_i) . Ir al paso 8.
8. Tomar una instancia y_i que no haya sido evaluado e ir al paso 9. Si tal instancia no existe, volver al paso 6.
9. Crear una secuencia de nodos ns_i e introducir C_i, x_i, y_i . Ir al paso 10.
10. Introducir en NS_i la secuencia de nodos ps_i . Volver al paso 8.
11. Si el tamaño de la secuencia de propiedades es igual a 4, ir al paso 12. Si no, ir al paso 2.
12. Consultar en G el conjunto N_1 de instancias x_i que cumplen la tripleta (C_i, p_1^j, x_i) . Ir al paso 13.
13. Tomar una instancia $x_i \in N_1$ que no haya sido evaluado e ir al paso 14. Si tal instancia no existe, volver al paso 3.

6. Estrategia de recomendación

14. Consultar en G el conjunto N_2 de instancias y_i que cumplen la tripleta (x_i, p_2^j, y_i) . Ir al paso 15.
15. Tomar una instancia y_i que no haya sido evaluado e ir al paso 16. Si tal instancia no existe, volver al paso 13.
16. Consultar en G el conjunto N_3 de instancias w_i que cumplen la tripleta (y_i, p_3^j, w_i) . Ir al paso 17.
17. Tomar una instancia $w_i \in N_3$ que no haya sido evaluado e ir al paso 18. Si tal instancia no existe, volver al paso 15.
18. Consultar en G el conjunto N_4 de instancias v_i que cumplen la tripleta (w_i, p_4^j, v) . Ir al paso 19.
19. Tomar una instancia $v_i \in N_3$ que no haya sido evaluado e ir al paso 20. Si tal instancia no existe, volver al paso 17.
20. Crear una secuencia de nodos ns_i e introducir C_i, x_i, y_i, w_i y v_i . Ir al paso 21.
21. Introducir en NS_i la secuencia de nodos ps_i . Volver al paso 19.

Como resultado, el algoritmo obtiene secuencias de nodos NS_i para cada competencia C_i formalizada en la base de conocimiento. Por lógica, se asume que un nodo origen C_i y nodo término y_i o v_i de cada secuencia de nodos, según sea el caso, constituyen una asociación de la forma $\rho - pathAssociated(C_o, C_t)$, donde C_o es el origen y C_t es el término de la asociación. De acuerdo con el modelo del SR mostrado en la Figura 6.2, tanto las secuencias de nodos como las asociaciones semánticas son almacenadas en la base de datos *Asociaciones semánticas y secuencias de nodos*, y posteriormente consultadas en el proceso de filtrado.

A diferencia del enfoque propuesto en [116], en el cual las asociaciones se construyen añadiendo nodo por nodo después del cálculo de intensidades semánticas, en la presente propuesta se descubren las asociaciones semánticas de la base de conocimiento antes de considerar la personalización en el proceso de obtención de recomendaciones. La ventaja de hacerlo de esta manera está en que todas las asociaciones son válidas en principio para todos los usuarios, y por lo tanto, el descubrimiento de secuencias de nodos y asociaciones semánticas puede ser programado (por ejemplo, siguiendo una estrategia *OnLine* u *OffLine*, según convenga) independientemente de los cambios en los intereses de los usuarios, mientras que en el enfoque de [116] un cambio en las preferencias o la inclusión de nuevos ítems requiere un recómputo de asociaciones e intensidades semánticas durante el proceso de obtención de recomendaciones, lo cual puede estropear la experiencia de usuario debido a los tiempos de ejecución producidos.

Tal como se adelantó, todas las asociaciones son válidas en principio para todos los usuarios. En pos de la personalización, a continuación, se explica un proceso en el

cual se descubren las asociaciones más relevantes, esto es, las que más se ajustan a los perfiles de los estudiantes.

Ejemplo 6.2. Teniendo en cuenta el extracto de la base de conocimiento mostrado en la Figura 4.18, se busca descubrir todas las asociaciones semánticas $\rho - pathAssociated$ entre las competencias. Para esto, se consideran las cuatro secuencias de propiedades (PS_1, PS_2, PS_3, PS_4) definidas en la Sección 6.3. Por ejemplo, al aplicar la secuencia de propiedades $PS_1 = [hasCompetencyType, hasCompetencyType^*]$ tomando como origen la competencia *Calculo el área de algunas figuras geométricas*, inicialmente se busca en la base de conocimiento las instancias vinculadas a tal competencia por medio de la propiedad `:hasCompetencyType`. Como se puede ver en la Figura 4.18, la única instancia que cumple este criterio es *Habilidad*. Posteriormente, se buscan las instancias vinculadas a *Habilidad*, esta vez, por medio de la propiedad `:hasCompetencyType^*`. La única instancia que cumple este criterio es la competencia *Aplico la fórmula del área del círculo*. El resultado de este proceso es una asociación $\rho - pathAssociated$ entre las competencias *Calculo el área de algunas figuras geométricas* y *Aplico la fórmula del área del círculo*. Entonces, la secuencia de nodos de tal asociación es $ps_1.PSNodesSequence() = [Calculo el área de algunas figuras geométricas, Habilidad, Aplico la fórmula del área del círculo]$. Ésta permite extraer todos los nodos que hacen parte de la asociación. Siguiendo un proceso similar, se aplican las otras tres secuencias de propiedades sobre la base de conocimiento. Todas las secuencias de nodos y las asociaciones descubiertas se muestran en la Tabla 6.4. En el proceso subsecuente, el cual corresponde al *Filtro semántico*, de éstas secuencias se determinan las más relevantes para el estudiante objetivo.

Secuencias de propiedades	Secuencias de nodos descubiertas	Asociaciones descubiertas
$PS_1 = [hasCompetencyType, hasCompetencyType^*]$	$ps_1.PSNodesSequence() = [Calculo el área de algunas figuras geométricas, Habilidad, Aplico la fórmula del área del círculo]$.	$\rho - pathAssociated(Calculo el área de algunas figuras geométricas, Aplico la fórmula del área del círculo)$.
$PS_2 = [hasActionVerb, hasActionVerb^*]$	<ul style="list-style-type: none"> • $ps_2.PSNodesSequence() = [Aplico la fórmula del área del círculo, Identificar, Identifico las partes de algunas figuras]$. • $ps_3.PSNodesSequence() = [Aplico la fórmula del área del círculo, Identificar, Identifico algunas figuras geométricas]$. • $ps_4.PSNodesSequence() = [Identifico las partes de algunas figuras, Identificar, Aplico la fórmula del área del círculo]$. • $ps_5.PSNodesSequence() = [Identifico algunas figuras geométricas, Identificar, Aplico la fórmula del área del círculo]$. • $ps_6.PSNodesSequence() = [Identifico algunas figuras geométricas, Identificar, Identifico las partes de algunas figuras]$. 	<ul style="list-style-type: none"> • $\rho - pathAssociated(Aplico la fórmula del área del círculo, Identifico las partes de algunas figuras)$. • $\rho - pathAssociated(Aplico la fórmula del área del círculo, Identifico algunas figuras geométricas)$. • $\rho - pathAssociated(Identifico las partes de algunas figuras, Aplico la fórmula del área del círculo)$. • $\rho - pathAssociated(Identifico algunas figuras geométricas, Aplico la fórmula del área del círculo)$. • $\rho - pathAssociated(Identifico algunas figuras geométricas, Identifico las partes de algunas figuras)$.

6. Estrategia de recomendación

	<ul style="list-style-type: none"> • $ps_7.PSNodesSequence()$ =[Identifico algunas figuras geométricas, Identificar, Aplico la fórmula del área del círculo]. 	<ul style="list-style-type: none"> • $\rho - pathAssociated$(Identifico algunas figuras geométricas, Aplico la fórmula del área del círculo).
$PS_3 = [hasCompetencyType, hasCompetencyType^*, hasActionVerb, hasActionVerb^*]$	<ul style="list-style-type: none"> • $ps_8.PSNodesSequence()$ =[Calculo el área de algunas figuras geométricas, Habilidad, Aplico la fórmula del área del círculo, Identificar, Identifico las partes de algunas figuras]. • $ps_9.PSNodesSequence()$ =[Calculo el área de algunas figuras geométricas, Habilidad, Aplico la fórmula del área del círculo, Identificar, Identifico algunas figuras geométricas]. 	<ul style="list-style-type: none"> • $\rho - pathAssociated$(Calculo el área de algunas figuras geométricas, Identifico las partes de algunas figuras). • $\rho - pathAssociated$(Calculo el área de algunas figuras geométricas, Identifico algunas figuras geométricas).
$PS_4 = [hasActionVerb, hasActionVerb^*, hasCompetencyType, hasCompetencyType^*]$	<ul style="list-style-type: none"> • $ps_{10}.PSNodesSequence()$ =[Identifico algunas figuras geométricas, Identificar, Aplico la fórmula del área del círculo, Habilidad, Calculo el área de algunas figuras geométricas]. 	<ul style="list-style-type: none"> • $\rho - pathAssociated$(Identifico algunas figuras geométricas, Calculo el área de algunas figuras geométricas).

Tabla 6.4 Ejemplo de descubrimiento de asociaciones semánticas complejas. Fuente: propia.

6.4 Filtro semántico

En los procesos desarrollados por los SR se involucra una gran carga de información, mucha de la cual puede ser irrelevante. Por ejemplo, en el enfoque *colaborativo*, se busca enfocar la recomendación en ítems que han sido del gusto de otros usuarios con gustos similares al del usuario objetivo [25]. Dado que en un SR puede haber un gran volumen de usuarios registrados, conviene realizar un proceso de filtrado que permita identificar usuarios relevantes para la recomendación. En general, es menester de los SR identificar la información que puede contribuir de mejor manera en la obtención de recomendaciones precisas.

Esta situación no es ajena al SR propuesto en esta investigación. En concreto, entre las asociaciones semánticas complejas descubiertas en la base de conocimiento puede haber unas más útiles que otras para desarrollar la tarea de personalización. Entonces, es pertinente establecer un mecanismo de filtrado que permita inferir tales asociaciones.

Al respecto, tal como se mencionó en la introducción del capítulo, en este trabajo se ha adoptado el enfoque de *SemDis* porque ofrece una gran variedad de métricas que permiten ponderar las asociaciones semánticas. Precisamente, esta posibilidad ha querido ser aprovechada para inferir las asociaciones más relevantes para el estudiante activo.

Con el propósito asegurar la personalización en el proceso de filtrado, se propone el proceso presentado a continuación:

1. Se consulta en la base de datos *Perfiles-ontología* las competencias en las que U tiene cierto grado de necesidad.
2. De las asociaciones semánticas descubiertas en la base de conocimiento se extraen las que tienen como nodo origen las competencias en las que el estudiante objetivo tiene cierto grado de necesidad, al tiempo que se filtran las que no cumplen tal criterio. Este es el paso inicial hacia la inferencia de asociaciones relacionadas con las competencias en las que el estudiante objetivo tiene dificultades educativas.
3. Las asociaciones extraídas en (2) son ponderadas de acuerdo a un grupo de métricas adaptadas de *SemDis* [62] que permiten medir aspectos ontológicos tales como la longitud de las asociaciones; y del contexto tales como fortalezas y debilidades educativas e intereses del usuario, y las relaciones de subsunción en los mapas. Además, se ha tomado prestado en este proceso el cálculo de similitudes semánticas propuesto en [20] y [117].
4. Las N asociaciones mejor ponderadas son seleccionadas para continuar la ejecución de la estrategia de recomendación, a la vez que se filtran las que no cumplen tal criterio.

Los detalles del proceso de filtrado de asociaciones semánticas se explican en la siguiente sección.

6.4.1 Ponderación y filtrado de asociaciones semánticas

Tal como se mencionó anteriormente, únicamente son ponderadas las asociaciones semánticas cuyos nodos origen hace parte del grupo de competencias en las cuales el estudiante tiene cierto grado de necesidad. De manera formal, son competencias c_i para las cuales $DoNC_{ui} > GN_o$, siendo GN_o un umbral configurable por el usuario del SR que separa un GN alto de uno bajo. De acuerdo con un análisis básico, se ha establecido $GN_o = 0,5$ (valor medio). Estas competencias constituyen el conjunto $LC_U = \{c_1, c_2, \dots, c_i, \dots, c_n\}$ y son las competencias para las cuales se obtendrán las listas de recomendaciones. Lo anterior constituye el primer paso en la personalización de las recomendaciones. Lo que sigue, es la ponderación de las asociaciones, proceso determinado por una serie de pesos que se explican más adelante.

De acuerdo a un análisis del contexto, por medio de la ponderación de las asociaciones se busca medir el grado de relación de éstas con:

- Las necesidades de los estudiantes, con respecto a las competencias y a las propiedades semánticas.
- Las propiedades semánticas, competencias y mapas de competencias de interés para los usuarios.

6. Estrategia de recomendación

- Las relaciones de subsunción con respecto a las competencias en las que el estudiante objetivo cierto grado de necesidad.
- La similitud entre las competencias que las componen.

A continuación, se definen las métricas de ponderación usadas.

Peso de la popularidad personalizada (Pop_{UA}). Esta métrica es una adaptación de la métrica *Popularidad* establecida por *SemDis* (Sección A.3.3). Originalmente, la popularidad indica que la relevancia de una asociación es proporcional al número de relaciones entrantes y salientes de sus nodos en una base de conocimiento. Este concepto se ha adaptado para indicar que la relevancia de una asociación es proporcional al número de relaciones entrantes y salientes que tienen sus nodos con otras entidades en las que el estudiante tiene cierto grado de necesidad. De manera formal: La popularidad personalizada de una asociación semántica A para un usuario U con perfil P_U se define en términos de la popularidad de sus nodos para tal usuario. La popularidad p_{Ui} del i -ésimo nodo e_i en una asociación A está definida para el usuario U por la expresión:

$$p_{Ui} = \frac{|pop_{e_i}^U|}{\max(|pop_{e_j}^U|)}, \quad (6.3)$$

donde $typeOf(e_i) = typeOf(e_j)$, tal que $1 \leq j \leq n$, donde n es el número total de instancias formalizadas en la base de conocimiento. Para el cómputo de la popularidad, se proponen dos posibilidades así:

Pos1: $pop_{e_i}^U$ es el conjunto de relaciones entrantes y salientes de e_i vinculadas con instancias (de competencias o sus propiedades, por ejemplo *CompetencyType*, *Verb*, *Age*, entre otros) registrados en P_U cuyo GN es mayor que GN_o ; y $\max(|pop_{e_j}^U|)$ representa el tamaño del conjunto más grande de relaciones entrantes y salientes vinculadas con instancias que hacen parte del perfil P_U cuyo GN es mayor GN_o . En consecuencia, la popularidad de las instancias depende de las competencias y/o propiedades en las que el usuario tiene necesidades educativas.

Pops2: $pop_{e_i}^U$ es el conjunto de relaciones entrantes y salientes de e_i vinculadas con instancias registradas en P_U cuyo GN es menor que GN_o ; y $\max(|pop_{e_j}^U|)$ representa el tamaño del conjunto más grande de relaciones entrantes y salientes vinculadas con instancias registradas en P_U cuyo GN es menor que GN_o . Por lo tanto, la popularidad de las entidades depende de las competencias y/o propiedades en las que U no tiene necesidades educativas.

Ahora bien, de acuerdo a *SemDis*, el peso total de la popularidad Pop de una asociación A para un usuario U es la suma de las popularidades de sus nodos así:

$$Pop_{UA} = \frac{1}{n} \sum_{i=1}^n p_{ui} \quad (6.4)$$

Con respecto a la expresión (6.4), la popularidad personalizada de una asociación es más grande en la medida que sus componentes se relacionen con otros en los cuales U tiene un alto o bajo grado de necesidad, dependiendo de la elección del usuario por $Pos1$ o $Pos2$, respectivamente. En términos prácticos, el usuario del SR deberá decidir si se premian las asociaciones compuestas por competencias y propiedades en las que el estudiante no tiene necesidades, enfocando la recomendación en el aprovechamiento de sus fortalezas ($Pos2$) o, por el contrario, se premian las asociaciones compuestas por competencias y propiedades en las que el usuario tiene necesidades, enfocando la recomendación en el aprovechamiento de sus debilidades ($Pos1$).

Se destaca que a diferencia de la popularidad definida en [62], en la presente propuesta sus valores dependen del rendimiento educativo cada estudiante, de ahí que la popularidad tome el adjetivo de personalizada. Finalmente, una característica que introduce la popularidad personalizada al SR es que se pondera la importancia de un nodo con base en sus vínculos con instancias registradas en los perfiles, lo cual es conveniente si el nodo en cuestión representa una competencia o propiedad no registrada.

Peso del contexto educativo (C_A) y Peso del contexto de los mapas (MC_A). Amén de ejecutar de la mejor manera la estrategia de recomendación, es importante que el usuario identifique los componentes de la base de conocimiento que considere más importantes, tarea conocida en *SemDis* como *identificación del contexto* (Sección A.3.3). Este concepto se ha incluido en este trabajo de tal forma que las recomendaciones sean acordes con el contexto en el que se enmarca el proceso de enseñanza y aprendizaje. En la presente estrategia, el contexto se mide por medio de dos pesos denominados *Peso del contexto del contexto educativo* y *Peso del contexto de los mapas*, explicados a continuación.

Peso de contexto educativo (C_A). Este criterio engloba las características educativas consideradas más importantes al momento de generar recomendaciones. El propósito es clasificar mejor las asociaciones semánticas cuyos nodos se relacionen con tales características. A diferencia del enfoque *SemDis*, el cual define el contexto en términos de las clases de la ontología, en esta propuesta C_p se focaliza en las instancias de la base de conocimiento. Ésta define convenientemente instancias que representan características educativas como el *nivel educativo* según el *marco de cualificación*, el *tipo de competencia*, el *verbo de acción* de interés y la *dificultad* del aprendizaje. Sea A una asociación semántica, v un nodo de A , y H un conjunto de instancias de interés para el usuario del SR, se definen los siguientes conjuntos:

6. Estrategia de recomendación

$$Y = \{v | v \in H \wedge v \in A\}, \quad (6.5)$$

$$T = \{v | v \notin H \wedge v \in A\}, \quad (6.6)$$

donde Y es el conjunto de nodos de A que también están definidos en H , y T el conjunto de competencias de A que no están definidas en H .

La métrica para medir el peso del contexto educativo propuesto aquí es una adaptación de la métrica *contexto* propuesta en *SemDis* (Sección A.3.3). Originalmente, el *contexto* indica que la relevancia de una asociación semántica depende de si sus nodos hacen parte de regiones de la base de conocimiento interesantes para el usuario. Este concepto ha sido adaptado para indicar que la relevancia de una asociación depende de si sus nodos corresponden a las características del contexto educativo interesantes para el usuario. De manera formal:

El peso del contexto educativo se define por:

$$C_A = \frac{1}{length(A)} (|Y|) \times \left(1 - \frac{|T|}{length(A)}\right), \quad (6.7)$$

donde $length(A)$ es el número de nodos de A .

Peso del contexto de los mapas (MC_A): este peso involucra en el filtrado la importancia de los mapas de competencias en los procesos de enseñanza y aprendizaje. Esto con base en que para los usuarios puede resultar conveniente centrar las recomendaciones en un mapa completo o en regiones específicas de éste sobre las cuáles haya especial interés. Sea A una asociación semántica, R_i una región de un mapa de competencias formada por instancias de competencias, c una instancia de competencias de A (se cumple la tripleta $(c, rdf:type, Competency)$, n el número de regiones de mapas de competencias indicadas por el usuario, se definen los siguientes conjuntos:

$$X_i = \{c | c \in R_i \wedge c \in A\}, \quad (6.8)$$

$$Z = \{c | (\forall i | 1 \leq i \leq n) c \notin R_i \wedge c \in A\}, \quad (6.9)$$

donde X_i es el conjunto de competencias de A que además están definidas en la región de interés R_i , y Z el conjunto de competencias de A que no pertenecen a ninguna de las regiones especificadas por el usuario. La métrica propuesta para el peso del contexto de los mapas es una adaptación de la métrica *contexto* propuesta en *SemDis* (Sección A.3.3). En este caso, la relevancia de una asociación semántica depende de si sus nodos hacen parte de los mapas de competencias (o regiones) de interés para el estudiante objetivo. De manera formal:

El peso del contexto de los mapas de la asociación A de acuerdo con:

$$MC_A = \frac{1}{length(A)} \left(\sum_{i=1}^n (W_{R_i} \times |X_i|)\right) \times \left(1 - \frac{|Z|}{length(A)}\right), \quad (6.10)$$

donde $length(A)$ es el número de competencias incluidas en A , W_{R_i} es el índice de interés por la región del mapa R_i , que corresponde a un número entre cero y uno de tal forma que $\sum W_{R_i} = 1$.

Peso de la longitud de la asociación (L_A). Por su naturaleza, la longitud de la asociación no puede ser asociada a los grados de necesidad de los estudiantes por las competencias y sus propiedades. Teniendo en cuenta la estrategia de recomendación, la cual busca competencias vinculadas semánticamente con las competencias en las que el usuario tiene cierto GN, se ha decidido medir la longitud de la asociación L_A de acuerdo a la métrica definida por *SemDis* para tal fin (Sección A.3.3):

$$L_A = \frac{1}{length(A)}. \quad (6.11)$$

Ésta pondera mejor las asociaciones más directas, esto es, las asociaciones con menos nodos y, por lo tanto, más cortas.

Peso de la subsunción en los mapas de competencias (SM_A). Las competencias de los niveles más bajos en los mapas de competencias son más específicas que las de los niveles más altos. Entonces, conviene dar relevancia a las asociaciones semánticas cuyos nodos se relacionan entre sí en los mapas de competencias. Esta métrica es un aporte de la presente investigación, la cual se define de manera formal así: sea A una asociación $\rho - pathAssociated(a, b)$, donde a y b son competencias, y M el conjunto de competencias que hacen parte de un mapa de competencias tal que $\{a, b\} \subseteq M$, y $:dependsOn$ una propiedad transitiva definida en la ontología cuyo dominio es una subcompetencia del origen, el peso de la subsunción en los mapas de competencias de A se pondera por medio de:

$$SM_A = \begin{cases} 1 & \text{si } a : dependsOn b \\ 0 & \text{en otro caso} \end{cases} \quad (6.12)$$

La expresión (6.12) indica que una asociación tiene una ponderación más alta si la competencia vinculada al término de la asociación es subcompetencia de la competencia vinculada al nodo origen en por lo menos uno de los mapas de competencias formalizados en la base de conocimiento. Notar que SM_A se nutre del reúso de competencias en diferentes mapas de competencias. Tal como se adelantó en el Capítulo 4, la ontología permite vincular una instancia de competencia a varios a más de un mapa a la vez, aumentando la posibilidad de descubrir relaciones de subsunción que influyan en la ponderación de las asociaciones y, por ende, en la construcción de las listas de recomendaciones.

Similitud semántica (SS_A). Los temas involucrados en las competencias pueden ser relevantes en algunos currículos educativos. Este es el caso de los estándares básicos

6. Estrategia de recomendación

de competencias definidos por el *Ministerio de Educación de Colombia*. De hecho, la ontología propuesta define la propiedad `:competencyTopic` para las competencias. Debido a que no fue posible hallar en la literatura revisada un vocabulario para asignar temas (aunque podría existir), `:competencyTopic` fue definida como “*Data property*”. Con el propósito de medir la semejanza entre competencias con base en sus temas, se ha definido la similitud semántica.

Considerando que los usuarios pueden usar diferentes palabras para referirse a los mismos temas (por ejemplo, *fraccionario* y *fracción*), en el presente trabajo se han adaptado los enfoques definidos en [20] y [117] para identificar *palabras clave* en un texto y calcular la *similitud semántica*, respectivamente, en lugar de usar técnicas tradicionales de emparejamiento sintáctico. En concreto, los temas son preprocesados para eliminar *palabras vacías* (por ejemplo, preposiciones, conjunciones, pronombres y artículos) y *lematizar* términos. Las palabras resultantes son agrupadas en *bolsas de palabras* asociadas a cada competencia. Posteriormente, se calcula la similitud entre competencias comparando sus bolsas de palabras. Sea A una asociación semántica $\rho - pathAssociated(x, y)$, y C_i una competencia asociada a una bolsa de palabras $CS_k = \{c_1^k, c_2^k, \dots, c_n^k\}$. Cada palabra c_i de todas las bolsas tienen asociado un vecindario $N(c_i)$ compuesto por términos semánticamente relacionados, incluyendo c_i . Los vecindarios pueden ser extraídos de bases de datos léxicas como *WordNet* para formar conjuntos de sinónimos o pueden ser definidos en ontologías por comunidades específicas, formalizando relaciones léxicas entre palabras (por ejemplo, por medio de la *hiponimia* y la *meronimia*). Al comparar dos bolsas de palabras de dos competencias C_i y C_j se crea un vector en un espacio n -dimensional, de acuerdo al modelo:

$$V_l = (\langle t_1^l, \omega_1^l \rangle, \dots, \langle t_p^l, \omega_p^l \rangle), \quad (6.13)$$

donde $l \in \{i, j\}$, ω_i representa el peso asociado a la palabra c_i y $p = |CS_i \cup CS_j|$ es el número de palabras distintas en la unión entre CS_i y CS_j . Si la palabra t_i hace parte de la bolsa de palabras CS_j , entonces $\omega_i = 1$, de otra forma, ω_i es computado con base en la *similitud del recinto máxima* (“*maximum enclosure similarity*”) que tiene con otra palabra t_j en su correspondiente vector V_j . Este procedimiento toma en cuenta la vecindad semántica global de cada palabra de la siguiente forma:

$$\omega_i = \begin{cases} 1 & \text{si la frecuencia de } c_i \text{ en } CS_j \text{ es mayor que cero} \\ \max_j (ES(c_i, c_j)) & \text{en otro caso} \end{cases}, \quad (6.14)$$

donde:

$$ES(c_i, c_j) = \frac{|N(c_i) \cap N(c_j)|}{|N(c_i)|}. \quad (6.15)$$

Finalmente, el peso de la similitud semántica de A es igual a similitud entre las bolsas de palabras de la competencia C_i (origen de A) y C_j (termino de A), la cual se computa por medio de:

$$SS_A = \frac{v_i \cdot v_j}{\|v_i\| \cdot \|v_j\|}. \quad (6.16)$$

Como resultado, el SR premia las asociaciones cuyo origen y término tienen similitudes en sus temas. Notar que el cálculo de similitudes propuesto no sólo se basa en el emparejamiento de palabras, sino que también considera sus sinónimos. Esto representa una ventaja en cuanto a eficacia, dado que se abre la posibilidad de hallar similitudes en una hipotética aplicación del esquema de metadatos y la ontología en un entorno real (por ejemplo, usando *Datos Enlazados* y/o “*Social tags*”) en el que se generen, compartan y vinculen diversas descripciones de competencias, aun cuando los usuarios utilicen diferentes expresiones con el mismo significado.

Peso total de una asociación semántica (W_A). Una vez se han establecido todos los criterios usados para ponderar las asociaciones semánticas, el peso total de una asociación A es dado por:

$$W_{UA} = k_1 \times Pop_{UA} + k_2 \times C_A + k_3 \times MC_A + k_4 \times L_A + k_5 \times SM_A + k_6 \times SS_A, \quad (6.17)$$

donde k_1, k_2, k_3, k_4, k_5 and k_6 son factores tales que $\sum_{i=1}^6 k_i = 1$. Aunque el SR establece estos valores por defecto, el usuario puede modificarlos según su criterio. Esto depende del enfoque que quiera darle a la ponderación de las asociaciones. Por ejemplo, para enfocarse en las fortalezas o debilidades del estudiante objetivo, el valor de k_1 debe ser mayor que los demás. Así mismo, para enfocarse en las competencias que apoyan el desarrollo de otras, k_5 debe ser mayor.

Una vez las asociaciones semánticas han sido ponderadas para cada estudiante, el SR selecciona las N más relevantes, es decir, las de mayor ponderación, para cada una de las competencias en las cada estudiante tiene un $GN > GN_o$. Posteriormente, el SR construye una lista de recomendaciones para cada una de tales competencias con base en un análisis de las N asociaciones correspondientes. Este proceso se presenta en la siguiente sección.

Ejemplo 6.3. Considerando los ejemplos 6.1 y 6.2, se busca ponderar las asociaciones semánticas de forma personalizada para el estudiante U . Inicialmente, se extraen las competencias para las cuales el estudiante objetivo tiene un grado de necesidad (GN) que supera el umbral $GN_o = 0.5$. El SR obtendrá recomendaciones para cada una de éstas competencias. Éstas se incorporan al conjunto $LC_U = \{c_1, c_2, \dots, c_i, \dots, c_n\}$. De acuerdo con el perfil-ontología de U (ver la Figura 6.5), la competencia *Calculo el área del algunas figuras geométricas* tiene un GN mayor que 0.5, por lo cual $LC_U = \{\text{Calculo el área del algunas figuras}$

6. Estrategia de recomendación

geométricas}. Ahora, se identifican de Tabla 6.4 las asociaciones y secuencias de nodos para las cuales el origen es $c_1 = \text{Calculo el área de algunas figuras geométricas}$. Éstas son: A_1 , A_8 y A_9 .

Para calcular el peso de A_1 , se calculan las métricas de ponderación propuestas en la Sección 6.4.1 de la siguiente forma:

La Popularidad personalizada de la asociación A_1 . Inicialmente, se calcula la popularidad de cada nodo de $ps_1.PSNodesSequence()$. Para el primer nodo, usando la ecuación (6.3) se tiene: $p_{U\text{Calculo el área de algunas figuras geométricas}} = \frac{2}{2} = 1$.

Notar en el perfil-ontología mostrado en la Figura 6.5, el nodo evaluado tiene vínculos con 2 propiedades cuyo GN es mayor que 0,5, por lo tanto $|pop_{e_i}^U| = 2$. Además, la competencia con el mayor número de vínculos con propiedades para las cuales su GN es mayor que 0,5 corresponde al mismo nodo evaluado. Por lo tanto $\max(|pop_{e_j}^U|) = 2$. Este mismo valor se aplica al calcular la popularidad de los nodos que corresponden a competencias. Siguiendo un proceso similar, $p_{UHabilidad} = 1$ y $p_{U\text{Aplico la fórmula del área del círculo}} = 0,5$. Ahora, es posible calcular el peso de la popularidad de A_1 aplicando la expresión (6.3):

$$Pop_{UA_1} = \frac{1}{3} \sum_{i=1}^3 p_{ui} = \frac{1}{3} (1 + 1 + 0,5) = 0,83$$

El Peso del contexto educativo y el Peso de del contexto de los mapas de A_1 . Inicialmente, se calcula la popularidad de cada nodo de $ps_1.PSNodesSequence()$. Para esto, se hace la siguiente suposición: U está interesado en reforzar tanto las competencias en las que registra NC más bajos que M (medio), como el mapa *MapaCompetencias1*, al cual le ha asignado un peso $W_{R_i} = 1$. Entonces, se definen los siguientes conjuntos:

$H = \{\text{Calculo el área de algunas figuras geométricas}\}$,

$R_i = \{\text{Calculo el área de algunas figuras geométricas, Identifico las partes de algunas figuras}\}$

A partir de éstos, se construyen los siguientes conjuntos:

$Y = \{\text{Calculo el área de algunas figuras geométricas}\}$,

$T = \{\text{Aplico la fórmula del área del círculo}\}$,

$X_i = \{\text{Calculo el área de algunas figuras geométricas}\}$,

$Z = \{\text{Aplico la fórmula del área del círculo}\}$,

El Peso del contexto educativo de A_1 se calcula aplicando la ecuación (6.7) de la siguiente manera:

$$C_{A_1} = \frac{1}{3} (1) \times \left(1 - \frac{2}{3}\right) = 0,11.$$

El Peso del contexto del mapa se calcula aplicando la ecuación (6.10) así:

$$MC_{A_1} = \frac{1}{2} (1 \times 1) \times \left(1 - \frac{1}{2}\right) = 0,25.$$

El Peso de la longitud de la asociación A_1 . Dado que A_1 tiene tres nodos, entonces:

$$L_{A_1} = \frac{1}{3} = 0,33.$$

El Peso de la subsunción en los mapas de competencias de A_1 . En la Figura 4.18 se observa que el término de A_1 Aplico la fórmula del área del círculo no es subcompetencia del origen. Calculo el área de algunas figuras geométricas en ninguno de los mapas. Por consiguiente: $SMA_1 = 0$. Notar que, para calcular este peso, se consideran las relaciones de subsunción en todos los mapas formalizados en la base de conocimiento. Por lo tanto, conviene que, en una aplicación de la ontología propuesta, se haga una reutilización de descripciones de competencias y mapas en ambientes de *Datos Abiertos Enlazados* (DAE).

El Peso de la similitud semántica de A_1 . Suponiendo que la competencia *Aplico la fórmula del área del círculo* tiene definido el tema *El área del círculo*; y para la competencia *Calculo el área de algunas figuras geométricas* se definió el tema *Áreas de figuras planas*. Después de un procesamiento de los temas, se establecen las siguientes bolsas de palabras (con base en *WordNet*):

$$CS_1 = \{\text{Área, Círculo}\},$$

$$CS_2 = \{\text{Área, Figura plana}\}.$$

Cada palabra de las bolsas tiene asociados los siguientes vecindarios (de acuerdo a *WordNet*):

$$N(\text{Área}) = \{\text{Área}\},$$

$$N(\text{Círculo}) = \{\text{Círculo}\},$$

$$N(\text{Figura plana}) = \{\text{Figura plana}\}.$$

Los anteriores vecindarios están compuestos únicamente por la palabra en cuestión, ya que no fue posible hallar sinónimos con el suficiente significado en las matemáticas. Aplicando la expresión (6.16) se calcula:

$$SS_{A_1} = \frac{1}{2} = 0,5.$$

Finalmente, se calcula el peso de la asociación aplicando la ecuación (6.17) así:

$$W_{UA_1} = 0,1 \times 0,83 + 0,1 \times 0,11 + 0,1 \times 0,25 + 0,1 \times 0,33 + 0,3 \times 0 + 0,3 \times 0,5 = 0,302.$$

En un proceso similar al anterior, se calcula el peso total de las asociaciones A_8 y A_9 . Los resultados se presentan en la Tabla 6.5.

De acuerdo con los resultados, la asociación A_8 es la mejor ponderada con $W_{UA_8} = 0,558$. Analizando sus pesos, se observa que los más importantes son: $SM_{A_8} = 1$, con lo cual se infiere que la competencia *Identifico las partes de algunas figuras planas* (término) sirve de apoyo para *Calculo el área de algunas figuras geométricas* (origen), ya que en alguno de los mapas formalizados en la base de conocimiento fue establecida una relación de subsunción entre ambas; $SM_{A_8} = 0,5$, con lo cual se infiere que existe una correspondencia entre ambas competencias basada en los temas de acuerdo a la similitud semántica. En menor medida, se calculó que $MC_{A_8} = 0,44$, a partir de lo cual se infiere que la asociación semántica puede ser relevante para el usuario debido a su vínculo con los mapas de competencias de interés para él. Dado que todo este conocimiento inferido está representado numéricamente por el peso total de la asociación, se concluye que esta

6. Estrategia de recomendación

asociación es relevante para el usuario y, por lo tanto, puede ser usada en el proceso de construcción de las listas de recomendaciones, el cual se describe en las próximas secciones. Por otra parte, la asociación A_1 fue ponderada con un valor bajo $W_{UA_1}=0,302$, por lo cual no convendría tenerla en cuenta en el proceso de recomendación. Sin embargo, al analizar sus pesos, se observa un valor de popularidad $Pop_{UA_8} = 0,83$, del cual se puede inferir que la asociación está relacionada con las competencias y/o propiedades en las cuales el usuario tiene cierto GN. Analizando el ejemplo de esta forma, convendría considerar esta asociación dentro de la obtención de recomendaciones. Al final, la baja ponderación de la asociación se debió a que el usuario estableció un valor bajo para el factor k_1 . Este ejemplo permite concluir que el uso de asociaciones semánticas complejas permite accionar el conocimiento que se desea inferir y, por lo tanto, que se desea usar dentro del proceso ejecutado por el SR.

Asociaciones	Pesos	Peso total
$A_1 = \rho - pathAssociated$ (Calculo el área de algunas figuras geométricas, Aplico la fórmula del área del círculo).	$Pop_{UA_1} = 0,83, C_{A_1} = 0,11, MC_{A_1} = 0,25,$ $MC_{A_1} = 0,33, SM_{A_1} = 0, SS_{A_1} = 0,5$	$W_{UA_1}=0,302$
$A_8 = \rho - pathAssociated$ (Calculo el área de algunas figuras geométricas, Identifico las partes de algunas figuras).	$Pop_{UA_8} = 0,4, C_{A_8} = 0,04, MC_{A_8} = 0,44,$ $MC_{A_8} = 0,2, SM_{A_8} = 1, SS_{A_8} = 0,5$	W_{UA_8} $= 0,558$
$A_9 = \rho - pathAssociated$ (Calculo el área de algunas figuras geométricas, Identifico algunas figuras geométricas).	$Pop_{UA_9} = 0,4, C_{A_9} = 0,04, MC_{A_9} = 0,44,$ $MC_{A_9} = 0,2, SM_{A_9} = 1, SS_{A_9} = 0,5$	$W_{UA_9} = 0,43$

Tabla 6.5 Ejemplo sobre la ponderación de asociaciones semánticas complejas. Fuente: propia.

6.5 Construcción de listas de recomendaciones

En esta fase se identifican los contenidos de VoD a recomendar. De acuerdo con el principio adoptado para la estrategia de recomendación, conviene construir una lista de recomendaciones para las competencias en las que cada estudiante u tiene cierto grado de necesidad (es decir, competencias para las cuales $GNC_{ui} > GN_o$). Para esto, se tiene en cuenta que los contenidos de VoD están vinculados con competencias, directamente o a través de segmentos de video, tal como está definido en el esquema de metadatos y la ontología propuestos. De acuerdo con estas características, se han establecido cuatro procesos así:

1. *Identificación de competencias relevantes*: proceso que analiza las N asociaciones de las competencias en las que $GNC_{ui} > GN_o$ con el propósito de extraer las competencias más relevantes.
2. *Consulta y ponderación de segmentos de video*: proceso que consulta la base de conocimiento y pondera los segmentos de video con base en las competencias relevantes descubiertas en (1).

3. *Consulta y ponderación de contenidos de VoD*: proceso que consulta en la base de conocimiento, pondera y ordena los contenidos de VoD de acuerdo con los segmentos de video ponderados en (2). De aquí se obtiene como resultado el grupo de recomendaciones para cada competencia en las que el estudiante tiene cierto grado de necesidad.
4. *Ordenamiento de contenidos de VoD*: los contenidos ponderados son ordenados para obtener la lista de recomendaciones.

Cada uno de los anteriores procesos se explica con detalle en el Anexo N.

Ejemplo 6.4. En este ejemplo se obtienen las recomendaciones un estudiante U cuyo perfil-ontología se muestra en la Figura 6.5. Inicialmente se identifican las competencias en las que el estudiante tiene dificultades, esto es, las competencias para las cuales el GN es mayor que 0.5. De acuerdo con el perfil-ontología, este criterio se cumple para la competencia $c_1 = \text{Calculo el área de algunas figuras geométricas}$. En el Ejemplo 6.3, se calculó el peso de las asociaciones para las cuales c_1 es el origen (A_1, A_8 y A_9), obteniendo los siguientes resultados: $W_{UA_1}=0,302$, $W_{UA_8}=0,558$ y $W_{UA_9}=0,43$.

De acuerdo con el proceso propuesto para la construcción de las listas de recomendaciones, inicialmente se identifican las competencias relevantes para c_1 . Considerando los pesos de las asociaciones mencionados, se determina el conjunto de N asociaciones $TopNA_{c_1U}$ y sus pesos totales WA_{c_1U} . Suponiendo $N = 3$, se obtiene $TopNA_{c_1U} = \{A_8, A_9, A_1\}$ y $WA_{c_1U} = \{0.558, 0.43, 0.302\}$. De acuerdo con el orden de $TopNA_{c_1U}$, se deduce que los términos de las asociaciones constituyen el conjunto de competencias relevantes T_{c_1U} . Por consiguiente, se obtiene $T_{c_1U} = \{\text{Identifico las partes de algunas figuras, Identifico algunas figuras geométricas, Aplico la fórmula del área del círculo}\}$. Ahora, se realiza la consulta y ponderación de los segmentos de video. Para esto inicialmente se consultan en la ontología los segmentos de video relacionados con las competencias del conjunto T_{c_1U} . De acuerdo con el extracto de la ontología mostrado en la Figura 4.18, se determina el conjunto de segmentos de video $VT_{c_1U} = \{\text{El círculo, El área del círculo}\}$. Para calcular el peso de los segmentos, se consultan las competencias asociadas a cada uno de ellos que hacen parte del perfil-ontología de U . Dado que *El círculo* está asociado únicamente con $c_a = \text{Identifico algunas figuras geométricas}$, la cual tiene un peso $W_1^{c_1U}(c_a) = 0.43$, aplicando la expresión (6.20) se calcula el su peso de la siguiente forma:

$$WVT_1^{c_1U} = \frac{0.43}{1} = 0.43.$$

De igual forma, se calcula el peso para el segmento *El área del círculo*:

$$WVT_2^{c_1U} = \frac{0.302}{1} = 0.302.$$

El siguiente paso es consultar y ponderar los contenidos de video. Para esto se consultan los contenidos vinculados con los segmentos del conjunto VT_{c_1U} . De acuerdo con el extracto de

6. Estrategia de recomendación

la ontología (Figura 4.18), se obtiene $V_{c1U} = \{\text{Figuras geométricas}, \text{Figuras: El círculo}\}$. Dado que el contenido *Figuras geométricas* está asociado únicamente con $vt_1 = \text{El círculo}$, cuyo peso es 0.43, aplicando la expresión (6.21) se calcula su peso de la siguiente forma:

$$WV_1^{c1U} = \frac{0.43}{1}.$$

De igual forma, se calcula el peso para el contenido *Figuras: El círculo* así:

$$WV_2^{c1U} = \frac{0.302}{1}.$$

Finalmente, se ordenan los contenidos de acuerdo con su peso de mayor a menor. Entonces, la lista de recomendaciones para la competencia c_1 se compone de los contenidos: *Figuras geométricas* y *Figuras: El círculo*. La Figura 6.6 muestra el resumen de los resultados, mostrando cómo se vinculan las competencias, los segmentos y los contenidos de VoD por medio de las propiedades de la ontología. Por ejemplo, la ponderación de *Figuras: El círculo* obtuvo una ponderación de 0.302 debido a que su segmento *El área del círculo*, identificado a través de la propiedad `:compRef`, fue ponderado también con 0.302. A su vez, la ponderación de este segmento se debió a su vínculo, a través de la propiedad `:hasVideoTrack`, con la competencia *Aplico la fórmula del área del círculo*, la cual se extrajo de una asociación semántica ponderada también con 0.302.

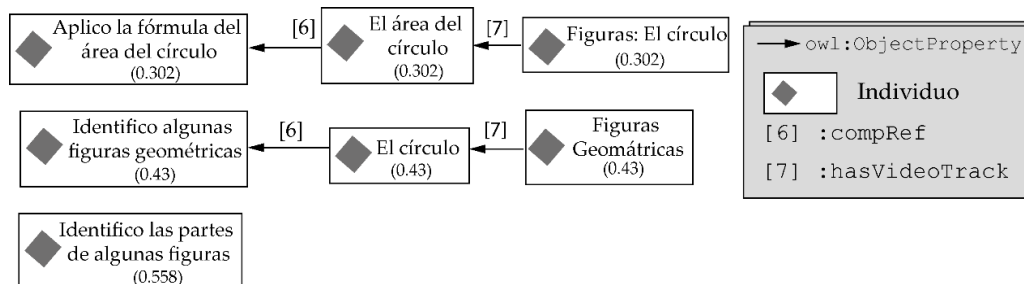


Figura 6.6 Ejemplo de ponderación de segmentos de video y contenidos de VoD. Fuente: propia.

En síntesis, el presente capítulo describe una estrategia de recomendación que asegura el requisito de personalización que ofrecen los sistemas de recomendación. Para obtener recomendaciones de contenidos de VoD enfocadas en las competencias, inicialmente se desarrolla la construcción de perfiles de usuario, la cual es una adaptación del modelo perfil-ontología propuesto en [116]. En la presente investigación, para la construcción de perfiles se toman los *niveles de competencia* computados por el método computacional propuesto (Capítulo 5) para adaptarlos a una métrica propuesta aquí como *grado de necesidad*, la cual es una medida de la necesidad que tiene un estudiante por las competencias o un atributo específico de éstas, por ejemplo, porque tiene bajos *niveles de competencia* en su historial académico. Una vez los perfiles han sido modelados, la estrategia de recomendación filtra asociaciones semánticas, determinadas a partir de ontología propuesta (Capítulo 4), cuyo origen son las competencias en las que el estudiante tiene cierto grado de necesidad; esto a través de un proceso de ponderación de asociaciones cuyas métricas han sido adaptadas de *SemDis* y de trabajos como [20] y [117] para representar aspectos ontológicos (longitud de las

asociaciones, relaciones de subsunción en los mapas y similitudes semánticas) y del contexto (fortalezas y debilidades educativas e intereses del usuario). Esto con el propósito de inferir las asociaciones semánticas más relevantes. Finalmente, a partir de las asociaciones inferidas, se realiza un proceso de ponderación de segmentos de video y contenidos de VoD. Con base en esta ponderación, se organizan los contenidos en listas de recomendaciones, dando mayor prioridad a los mejor ponderados.

Capítulo 7

Evaluación del Sistema de recomendaciones

El presente capítulo describe la evaluación del *Sistema de Recomendaciones* (SR) propuesto, la cual se enfoca en las siguientes preguntas de investigación:

1. ¿Cuál es la precisión del SR propuesto en comparación con un SR sintáctico?
2. ¿Cómo es el comportamiento del SR en comparación con un enfoque sintáctico en términos del tiempo de ejecución?

Dado que el SR propuesto se clasifica dentro del enfoque semántico, se eligió realizar la evaluación en contraste con el enfoque sintáctico. Para esto, se seleccionó el coeficiente Sørensen-Dice porque, según [25], es propicio para hallar similitudes sintácticas entre conjuntos de palabras clave que describen ítems. Éste se define de la siguiente manera [25]:

$$SD = \frac{2|A \cap B|}{|A| + |B|}, \quad (7.1)$$

donde A y B son los conjuntos de términos que describen un ítem.

Para responder las preguntas de investigación, se construyeron dos prototipos Java, los cuales son aportes adicionales de la presente investigación. El primero, que implementa todo lo propuesto en el trabajo, ofrece una interfaz que permite especificar el estudiante objetivo y, de acuerdo a la estrategia de recomendación propuesta (Capítulo 6), ofrece la posibilidad de configurar tanto los coeficientes de las métricas de ponderación como las regiones de los mapas y los atributos de las competencias interesantes para el usuario. El segundo prototipo desarrolla una estrategia de recomendación equivalente a la propuesta aquí, con la diferencia de que las asociaciones entre competencias son de tipo sintáctico, para lo cual se aplica el coeficiente Sørensen-Dice sobre las palabras clave que describen las competencias. Una vez ejecutados los procesos correspondientes, los prototipos entregan las listas de recomendaciones, las cuales fueron usadas para comparar la precisión ofrecida por cada sistema. Aspectos específicos de los prototipos se tratan más adelante en la Sección 7.1.

Con respecto a la evaluación de los SR, es común que ésta se realice a partir de colecciones de datos consolidadas de acceso libre que modelan las preferencias de los usuarios tales como las de *MovieLens* [125], las cuales registran las valoraciones explícitas de los usuarios por películas. Usando un enfoque denominado *experimentos "offline"* [126], el sistema evaluado se alimenta con una parte de la colección (*datos de entrenamiento*) para producir las recomendaciones. Posteriormente, de acuerdo al grupo de datos restantes (*datos de prueba*), se cuantifica la capacidad del sistema ya sea para predecir las valoraciones (a través de métricas de error) o para generar recomendaciones acertadas (usando métricas de precisión). Precisamente, la posibilidad de utilizar libremente colecciones de datos de gran tamaño dispuestas gratuitamente en la web y la facilidad para realizar el número de pruebas necesario para evaluar diferentes algoritmos hace que desarrollar *experimentos "offline"* sea la opción con menores costos de implementación [126].

En el marco de las observaciones anteriores, dado que no se creyó conveniente utilizar las valoraciones de los estudiantes por los contenidos como una fuente confiable de información debido a su posible falta de metacognición (esto es, podrían no tener el criterio para juzgar si un contenido puede o no apoyar su aprendizaje), además de que no se contó con una colección de datos consolidada que representara la efectividad de los contenidos en la adquisición de competencias, y ante la dificultad de crear una con tales características debido a los plazos de la investigación, no fue posible configurar una evaluación del sistema de recomendaciones propuesto enfocada en *experimentos "offline"*. Como alternativa se decidió dar cuenta de la precisión del SR adaptando la métrica *Precision* para cuantificar la tasa de aciertos del sistema de acuerdo con el criterio de docentes expertos, verificando si las recomendaciones son propicias para los estudiantes según su historial de aprendizaje en procesos educativos reales. De esta manera, se busca aprovechar el conocimiento de primera mano que tienen los docentes tanto en las competencias y mapas de competencias evaluados en cada caso como en los procesos de aprendizaje de sus estudiantes, para reconocer si los contenidos de VoD recomendados por el sistema pueden ser útiles en el apoyo en la adquisición de competencias. Entonces, la opinión de los expertos permite concluir de manera confiable acerca del rendimiento del sistema propuesto. En teoría, este enfoque es más costoso de conducir que los *experimentos "offline"*: incorporar a los expertos y solicitarles la evaluación de 3900 recomendaciones requiere de una gran cantidad de tiempo y de dinero si su participación no es voluntaria. Sin embargo, desde el principio de la investigación se contó con el apoyo voluntario de los expertos, lo cual estimuló la opción de evaluar la propuesta de acuerdo a su criterio.

En relación a aspectos específicos de la evaluación desarrollada, un grupo de 10 expertos, docentes de matemáticas, proveyeron concienzudamente y a través de formatos digitales las descripciones de 160 mapas de competencias de 41 estudiantes de básica secundaria, correspondientes a seis períodos académicos (de dos meses y medio cada uno), con base en los estándares básicos de competencias de Colombia en matemáticas [53]. Los estudiantes seleccionados demuestran en su historial tanto bajo como alto rendimiento en el aprendizaje de competencias. Los docentes también proveyeron las descripciones de 304 competencias y 191 contenidos de VoD con base en un grupo de elementos seleccionados del esquema de metadatos propuesto en el Capítulo 3. Estos aspectos, que se relacionan con la colección de los datos, se presentan con más detalle posteriormente en la Sección 7.2.

Con respecto al proceso llevado a cabo en la experimentación, se ejecutaron una serie de fases que involucran: 1) la formalización de los datos coleccionados en bases de datos y bases de conocimiento; 2) la configuración y ejecución de los prototipos, incluyendo la determinación de los factores de las métricas de ponderación (k_1, \dots, k_6); 3) la obtención de las recomendaciones y la medida de los tiempos de ejecución; y 4) el análisis de las recomendaciones por parte de los expertos, el cual se enfocó en verificar la pertinencia de los contenidos de VoD sugeridos para cada estudiante de acuerdo a su perfil. Con esto se buscó verificar si el SR ofrece resultados que, en principio, son útiles para apoyar el aprendizaje. El proceso de experimentación se explica con mayor detalle en la Sección 7.4.

Después de la experimentación, se realizó la organización y análisis de los resultados, a partir de los cuales se obtuvo conclusiones con respecto a las preguntas de investigación planteadas al inicio del capítulo. Esto se presenta detalladamente en la Sección 7.5.

7.1 Prototipos

Con el propósito de realizar un estudio comparativo, se construyeron dos prototipos Java. El primero, denominado *EduCompRS*¹¹, correspondiente al SR propuesto, se compone de:

1. Una base de conocimiento basada en la ontología propuesta, creada por medio del editor *Protége*¹². Específicamente, para este prototipo se usaron las ontologías:

¹¹ El pototipo Java de EduCompRS puede ser descargado de:

<https://mega.nz/#F!H6Qz0IyC!uprn0NVV3qApn0ePdSn4YA>

¹² La base de conocimiento en RDF/XML puede ser descargada de:

https://mega.nz/#F!y3YWVKxZ!dLnxl4fAqjyyU_INXKUHLA

adaptación de “*Ontology for Media Resources*”, adaptación de “*Competency Model Ontology*” y la construida para el modelado de mapas de competencias *OntoRCM*.

2. Una base de datos *MongoDB* que almacena el historial de NC de los estudiantes y los vecindarios de términos¹³.
3. Un grupo de aplicaciones Java que usan tecnologías como *Jena Ontology API*. Tal grupo de aplicaciones se compone de componentes para: 1) descubrir asociaciones semánticas; 2) descubrir relaciones de subsunción en mapas de competencias; 3) computar el peso de la popularidad personalizada; 4) computar el peso de la longitud de las asociaciones; 5) computar los *Grados de Necesidad (GN)* en las competencias y sus propiedades; 6) medir la similitud semántica entre competencias; y 7) desplegar una aplicación de usuario que obtiene las recomendaciones, cuya interfaz se presenta en la Figura 7.1.

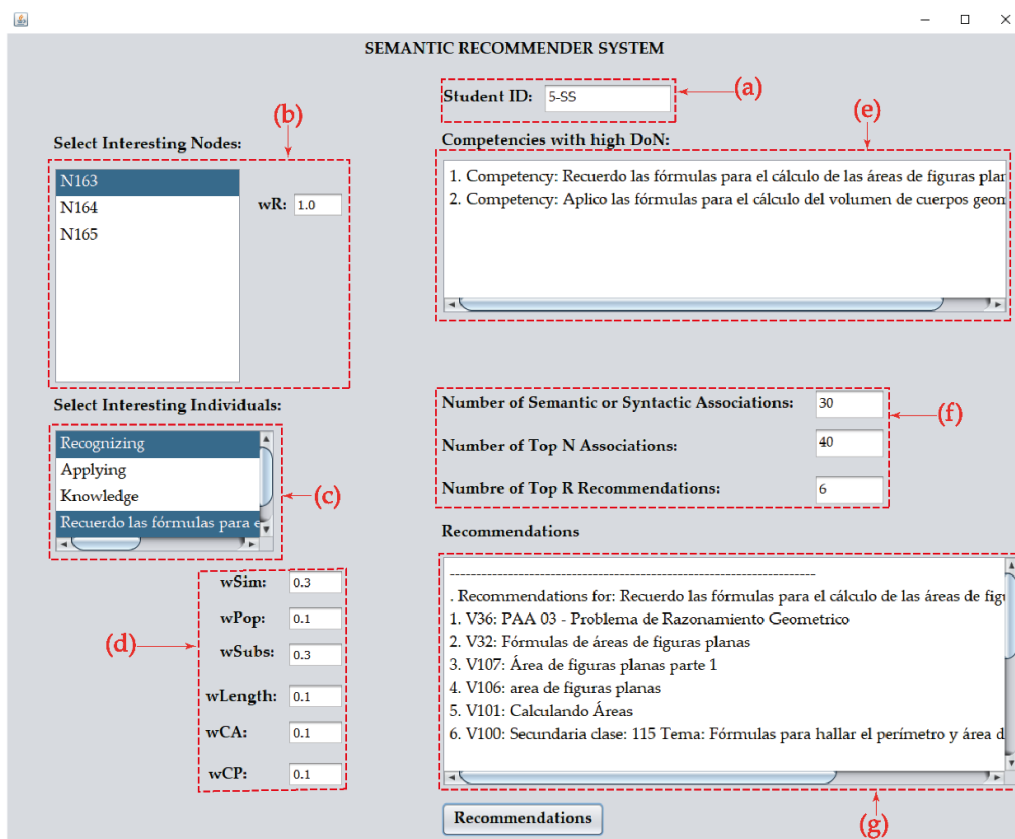


Figura 7.1 Interfaz del prototipo del sistema de recomendaciones propuesto. Fuente: propia.

A través de la aplicación de usuario, se establece el identificador del estudiante objetivo en el área (a), las regiones de los mapas interesantes relacionadas con el

¹³ La base de datos Mongo puede ser descargada de: <https://mega.nz/#F!D6QmXITY!Qr2HWLtlSJcquld7eHi91w>

Peso del contexto de los mapas en el área (b) y las instancias interesantes de la base de conocimiento relacionados con el *Peso del contexto educativo* en el área (c). Adicionalmente, se pueden variar los pesos k_1, \dots, k_6 en el área (d), el número de asociaciones a considerar para obtener las recomendaciones (δ), el número de las N asociaciones y el tamaño de la lista de recomendaciones en el área (f). Al presionar el botón *Recommendations*, la lista de competencias con un GN alto se despliega en el área (e), y se calculan tanto el *Peso del contexto de los mapas* como el *Peso del contexto educativo*. Después, estos resultados y los pesos de los demás componentes son usados para calcular el peso total de las asociaciones. Subsecuentemente, se lleva a cabo el proceso de filtrado. Finalmente, se despliegan las listas de recomendaciones en el área (g) (una lista por cada competencia con un GN alto). Con el propósito de garantizar la viabilidad computacional, los componentes 1, 2, 3, 4, 5 y 6 pueden ser ejecutados sin la intervención del usuario (por ejemplo, con ejecuciones programadas) e independientemente de la ejecución de la aplicación de usuario.

El segundo prototipo¹⁴ (llamado SR sintáctico), aplica el coeficiente Sørensen-Dice para descubrir asociaciones entre competencias con base en las similitudes entre palabras clave calculadas por medio de la expresión (7.1). Dado que tal algoritmo es generalmente usado en el emparejamiento sintáctico, en este trabajo tales asociaciones se denominan *asociaciones sintácticas*. Los valores de similitud entre competencias se usan para pesar los segmentos y contenidos de video. Este prototipo también desarrolla la estrategia de recomendación del SR propuesto. En consecuencia, los dos sistemas difieren únicamente en la forma en que son pesadas las asociaciones semánticas o sintácticas, según sea el caso.

Una vista funcional, una descripción de los componentes desarrollados y una vista dinámica de los prototipos se presenta en el Anexo O (Sección O.1 y Sección O.2).

7.2 Colección de datos

Un grupo de 10 expertos, docentes de matemáticas, proveyeron las descripciones de 160 mapas de competencias de 41 estudiantes, correspondientes a 31 diferentes estructuras de mapas. Las descripciones de estructuras y mapas siguieron el mismo formato usado en la colección de datos para la evaluación del método computacional (ver Figura 5.7 y Figura 5.8, respectivamente). Los mapas corresponden a seis períodos académicos (de cuatro meses cada uno), con base en los estándares básicos

¹⁴ El prototipo del SR sintáctico puede ser descargado de:
<https://mega.nz/#F!PvIRTYiQ!g2YcqIK9IKUoMLKZI4Em3Q>

de competencias de Colombia. Los docentes también proveyeron las descripciones de competencias y contenidos de VoD con base en un grupo de elementos seleccionados del esquema de metadatos propuesto en el Capítulo 3. La selección de elementos se realizó teniendo en cuenta si cada uno está incluido o no en la estrategia de recomendación propuesta. Específicamente, para la descripción de las competencias se utilizó un formato como el mostrado en la Figura 7.2, mientras que para los contenidos se usó el formato mostrado en la Figura 7.3.

Id competencia (identifier)	Título (title)	Esquema de cualificación (scheme)	Nivel del esquema (level)	Categoría (KSC_category)	Verbo (action_verb)	Temas (topic)	Lenguaje (language)	Palabras clave (keyword)
SS-PN-061	Aplico relaciones y propiedades de las operaciones aritméticas en ejercicios	Isced-unesco	2	Skill	Aplicar	Operaciones aritméticas Propiedades de las operaciones aritméticas	es	Aplicar, relaciones, propiedades, operaciones aritméticas, aritmética

Figura 7.2 Formato usado para la descripción de competencias. Fuente: propia.

Id contenido (program Id)	Título (Title - contenido)	URL (ProgramURL)	Id segmento (segment Id)	Título (Title - segmento)	Competencias asociadas (SegmentComp Ref)
V3	Suma y resta de fracciones	https://www.youtube.com/watch?v=Y-j8Y6VUX9Q1	S12	Ejercicio resuelto de suma de fracciones con el mismo denominador	CQ-PN-014
			S13	Representación gráfica de suma de fracciones con el mismo denominador	CQ-PN-018
			S14	Ejercicio resuelto de resta de fracciones con el mismo denominador	CQ-PN-017

Figura 7.3 Formato usado para la descripción de contenidos de VoD y segmentos. Fuente: propia.

Subsecuentemente, se obtuvieron las bolsas de palabras de las competencias a partir de la lematización de sus temas (elemento de datos *topic*). Éstas fueron posteriormente analizadas para establecer un glosario de los términos resultado de la unión de todas las bolsas de palabras. Después, cada término fue estudiado para establecer las vecindades a tener en cuenta durante el proceso de recomendación, específicamente, durante la ponderación de la similitud semántica de las asociaciones semánticas complejas. Cabe mencionar que la lematización, la construcción del glosario de términos y de las vecindades fueron procesos no incluidos dentro del prototipo, siendo éstos realizados de forma manual. Los detalles de este tipo de procesos se pueden hallar en [20].

7. Evaluación del Sistema de recomendaciones

Con toda la información mencionada, comprendida por descripciones de competencias, mapas de competencias, estructuras de mapas, contenidos de VoD y segmentos; y bolsas de palabras y vecindarios de términos, se conformó la colección de datos usada para evaluar el SR propuesto¹⁵. La estadística de ésta se presenta en la Tabla 7.1.

Finalmente, la parte de la colección de datos relacionada con la descripción de entidades (competencias, estructuras de mapas, contenidos de VoD y segmentos) fue formalizada en una base de conocimiento¹⁶, producto de la construcción de instancias de las clases, y las propiedades de datos y objetos de la ontología propuesta en el Capítulo 4; mientras que la información relacionada con los NC de los estudiantes, las bolsas de palabras de las competencias y los vecindarios de términos fue almacenada en una base de datos *MongoDB*¹⁷.

Tanto la base de conocimiento como la base de datos resultantes fueron usadas por el prototipo del SR propuesto para llevar a cabo la evaluación.

Datos	Cantidad
Estudiantes	41
Competencias	322
Estructuras de mapas	31
Mapas de competencias	160
Niveles de competencia (NC)	1340
Contenidos de VoD	181
Segmentos de video	400
Bolsas de palabras de competencias	322
Vecindarios de términos	124

Tabla 7.1 Estadística de la colección de datos para la evaluación del sistema de recomendaciones propuesto. Fuente: propia.

7.3 Métricas

Con el propósito de evaluar la precisión del SR y responder la primera pregunta de investigación, se usó una métrica *Precision* definida por [126]:

$$P = |h|/|recset|, \quad (7.2)$$

¹⁵ La documentación de los datos coleccionados puede ser descargada de:

<https://mega.nz/#F!m2IzGBYB!XUGgtvIku3I9eel0Io49BQ>

¹⁶ La base de conocimiento en formato RDF/XML puede ser descargada de:

https://mega.nz/#F!y3YWVKxZ!dLnxl4fAqjyyU_INXKUHLA

¹⁷ El dataset JSON de niveles de competencia, bolsas de palabras y vecindarios se encuentra en:

<https://mega.nz/#F!PiBQIKwT!qqrAORsoyW44hGpf7utKmg>

donde h es el número de recomendaciones acertadas, y $reset$ es el número total de recomendaciones. El *Error Medio Absoluto* (MAE), al igual que otras métricas tales como el *Error Cuadrático Medio* (RMSE) y el *Error Medio Absoluto Normalizado* (NMAE) no fueron consideradas en la evaluación del sistema propuesto ya que sólo pueden ser aplicadas en sistemas capaces de generar recomendaciones con base en el cálculo de predicciones a partir de las valoraciones por los ítems expresadas por los usuarios [25]. Entonces, el error representa la diferencia entre las valoraciones expresada por los usuarios y las predicciones computadas por el sistema. Éste no es el caso del sistema de recomendaciones propuesto, ya que, por su naturaleza, se basa en el cálculo de la relevancia semántica entre las competencias en las cuales los estudiantes tienen necesidades educativas y otras competencias formalizadas en las ontologías, siendo esta la base para el descubrimiento y ponderación de los contenidos de VoD. Entonces, el sistema busca sugerir contenidos de VoD que pueden ser útiles para los estudiantes, en lugar de predecir sus preferencias. Este enfoque útil para la medición de la precisión, denominado *medida de la predicción de uso* [126], establece unas métricas consolidadas para su uso: *Precision*, *Recall* y *F1*. Precisamente, *Precision* ha sido seleccionada como la métrica para cuantificar la precisión del sistema propuesto por su utilidad en la medición de la tasa de aciertos del sistema. Justamente, de acuerdo con los resultados de la evaluación presentados posteriormente en el Capítulo 7, *Precision* permitió concluir acerca de la capacidad del sistema para identificar los contenidos más relevantes para un estudiante dado, siendo éste un factor clave en la determinación de si lo propuesto realmente aporta en la determinación de recomendaciones útiles para el apoyo personalizado de necesidades en la adquisición de competencias. Por su parte, *Recall* y *F1*, las cuales representan la relación entre el *número de aciertos* y el *número máximo teórico de aciertos en todo el conjunto de prueba* [25], no fueron tenidas en cuenta porque la identificación del máximo de aciertos existente de entre todos los contenidos del sistema para cada estudiante fue difícil de determinar debido a las restricciones de disponibilidad de los expertos docentes participantes en la evaluación.

Por otra parte, para evaluar la eficiencia del SR y responder la segunda pregunta de investigación, se usó una medida del tiempo de ejecución definida por:

$$t = \sum_{i=1}^{100} t_i / 100, \quad (7.3)$$

donde t_i es la i -ésima medida de tiempo. Notar que cada medida se toma 100 veces para reducir la variabilidad de los resultados.

7.4 Fases de la evaluación

Las fases de la evaluación ejecutadas fueron las siguientes:

1. Las descripciones de competencias, mapas y contenidos de VoD fueron usadas para construir una base de conocimiento basada en la ontología propuesta.
2. El historial de NC fue formalizado en una base de datos *MongoDB*.
3. Por medio del prototipo *EduCompRS*, se calcularon: el *Peso de la popularidad personalizada*, el *Peso de la longitud de la asociación*, el *Peso de la subsunción de los mapas de competencias*, y la *Similitud semántica*. Los resultados fueron almacenados en una base de datos *MongoDB*.
4. Mientras se realizaron los cálculos del ítem 3, se realizaron las medidas de tiempo de ejecución de acuerdo con la expresión (7.3).
5. Se configuró *EduCompRS* para cada estudiante objetivo, tomando como regiones de los mapas e instancias interesantes los mapas de competencias, los verbos de acción desarrollados en los cursos y las competencias en las que el estudiante objetivo tiene un GN alto. Los pesos k_1, \dots, k_6 se configuraron de la siguiente forma: $k_1 = 0.1$, $k_2 = 0.1$, $k_3 = 0.1$, $k_4 = 0.1$, $k_5 = 0.3$ and $k_6 = 0.3$. Estos valores se establecieron experimentalmente. Esto se explica más adelante (ver Sección 7.5.3).
6. Se obtuvieron las recomendaciones para cada estudiante objetivo por medio del SR semántico. Al tiempo, se midieron los tiempos de ejecución de los procesos del cálculo del *Peso del contexto de los mapas* y el *Peso del contexto educativo* y de la obtención de las recomendaciones. Es importante tener en cuenta que, a diferencia del cálculo de los otros pesos, los mencionados sólo pueden ser configurados y calculados al momento de obtener las recomendaciones.
7. Se obtuvieron las recomendaciones del SR sintáctico y los tiempos de ejecución.
8. Las recomendaciones de ambos SR fueron evaluadas por el grupo de 10 expertos, verificando su pertinencia para los estudiantes objetivo. De acuerdo con los resultados, se calculó la precisión del sistema propuesto y la del sintáctico por medio de la expresión (7.2).

7.5 Resultados

En las siguientes subsecciones se presentan los resultados de la evaluación.

7.5.1 Resultados de precisión

La Figura 7.4 presenta los resultados de precisión del SR propuesto y el sintáctico de acuerdo con diversos números de asociaciones δ . El total, los expertos evaluaron 3900 recomendaciones (listas de seis recomendaciones, una por cada competencia con GN alto). Se puede observar que para, por ejemplo, 150 asociaciones (semánticas o sintácticas según sea el caso), el SR propuesto tiene una precisión igual a 0,48. Es decir que, por cada 100 recomendaciones, el SR acierta en 48 de ellas. En el caso del SR sintáctico, éste tiene una precisión igual a 0,2. Entonces por cada 100 recomendaciones, éste acierta en 20 de ellas. En general, las curvas tienen una tendencia lineal que describe una correlación directa entre la precisión y el número de asociaciones. Analizando Figura 7.4, es evidente que el SR propuesto tiene un mejor rendimiento, superando en 38% al SR sintáctico.

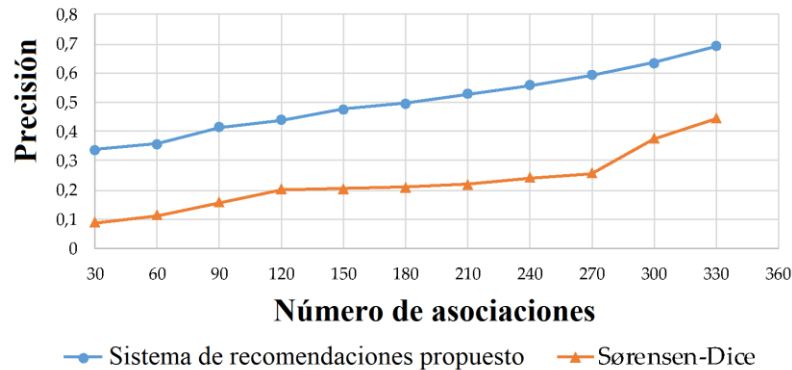


Figura 7.4 Resultados de precisión: sistema de recomendaciones propuesto vs sistema de recomendaciones sintáctico. Fuente: propia.

Los resultados también permiten observar que la incorporación del *dominio de aprendizaje* (competencias educativas), del *contenido* (enfocado en el VoD) y del *usuario* (centrado en su rendimiento) al modelo funcional y matemático del sistema de recomendaciones propuesto fue determinante para alcanzar precisiones superiores a 0,6 (38% mayores que el enfoque sintáctico, el cual no considera el contexto), demostrando cómo el uso de información contextual en el proceso de recomendación permite compensar la falta de grandes volúmenes de datos (de valoraciones, por ejemplo) usados comúnmente en los enfoques CB y CF. Esto se evidencia en el hecho de que la colección de datos construida y utilizada durante la investigación (Sección 7.2), la cual formaliza el rendimiento de los estudiantes y la descripción de competencias, mapas de competencias y contenidos (del orden de los 2000 datos) es pequeña en comparación a las colecciones de datos sintéticos usadas comúnmente en la *experimentación offline* (para *MovieLens*, colecciones con más de 100000 datos) para evaluar sistemas de los enfoques CB y CF. A efecto de contar con información

7. Evaluación del Sistema de recomendaciones

adicional a la contextual en la recomendación, el uso de técnicas de la web semántica también aportó al proceso el conocimiento aseverado e inferido adyacente a los vínculos lógicos entre competencias, mapas de competencias y contenidos.

Por otra parte, se decidió hacer un estudio enfocado en determinar si las diferencias en los resultados de precisión son significativos. Esto se debe a que, en evaluaciones experimentales comparativas, se pueden obtener resultados aparentemente distantes, pero que no tienen validez porque el tamaño de la muestra no es adecuado. O puede suceder lo contrario, resultados aparentemente cercanos, pero cuya diferencia es representativa porque el tamaño de la muestra es el adecuado.

Para realizar este estudio, se definió como hipótesis nula (H_0) *no hay diferencia significativa entre los resultados de precisión de los dos sistemas de recomendaciones* y como hipótesis alternativa (H_a) *existe diferencia significativa entre los resultados de precisión de los dos sistemas de recomendaciones*. La probabilidad promedio de acierto entre los dos sistemas es $p = (A_1 + A_2)/2 = 0.36527624$ para 300 asociaciones, donde A_1 es la probabilidad de acierto del sistema propuesto ($A_1 = 0,65$) y A_2 es la probabilidad de acierto del sistema sintáctico ($A_2 = 0,38$). Considerando una significancia $\alpha = 0,05$, el valor crítico de la distribución normal es $Z_{\alpha=0.05} = 1.96$. El error estándar es $se = 0,008$. Para determinar si las diferencias entre los resultados son estadísticamente significativas, es decir, para determinar si H_a es probablemente cierta, se debe cumplir que $|A_1 - A_2| > se * Z_{\alpha=0.05}$. Al reemplazar los datos, se da que $|A_1 - A_2| = 0,27$ es mayor que $se * Z_{\alpha=0.05} = 0,1568$. Por lo tanto, se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alternativa como probablemente cierta (H_a). Entonces, se puede concluir que la diferencia en los resultados es estadísticamente significativa. Esto se debe básicamente a que el tamaño de la muestra (3900 recomendaciones) hace que el error estándar inherente a la experimentación se reduzca. De acuerdo a [111], dado que el error estándar ($se = 0.008$) es menor que 0,01, entonces se concluye que el SR propuesto tiene una precisión significativa sobre el SR sintáctico con una seguridad inherente del 99%.

Cabe mencionar que para que el sistema de recomendaciones propuesto impacte positivamente en el proceso de aprendizaje, se han identificado una serie de aspectos a tener en cuenta, los cuales corresponden a *variables de la investigación en educación* definidas en [127] tales como la *actitud y aptitud ante el aprendizaje* (de los estudiantes ante el consumo del video), el *aprovechamiento académico* (de los videos y herramientas software como herramientas de apoyo al aprendizaje), el *desarrollo intelectual y cognitivo* (con respecto al conocimiento en todo lo concerniente a las competencias). Otras variables como la *pertenencia a un género, grupo social o grupo étnico*, la *motivación profesional*, la *edad* y el *sexo* no son consideradas porque no

inciden en el impacto que pueden tener las recomendaciones de video en la adquisición de competencias. Los aspectos identificados a tener en cuenta se listan a continuación:

- *Trastornos cognitivos y limitaciones visuales y/o auditivas*: es deseable que los estudiantes no tengan ningún tipo de trastorno cognitivo y limitación visual y/o auditiva dadas las características de los contenidos de VoD.
- *Calidad de los contenidos*: se espera los videos deben ser de calidad en cuanto a contenido y edición, lo cual puede aumentar el interés de los estudiantes.
- *Conocimiento en torno a las competencias y el VoD*: se aspira a que los docentes conozcan y practiquen la educación centrada en competencias y sean habituales consumidores de VoD en la web para el apoyo de su enseñanza.
- *Calidad de las descripciones*: relacionada con la variable anterior, las descripciones de competencias, mapas de competencias y contenidos deben ser de calidad. Lo anterior teniendo en cuenta que las descripciones constituyen el conocimiento de entrada del SR.
- *Herramientas de apoyo*: es deseable que los docentes cuenten con herramientas tecnológicas que faciliten la creación de descripciones. La idea es que se concentren sólo en lo netamente pedagógico y no en lo tecnológico.
- *Estilo de aprendizaje*: se desea que los estudiantes tengan los estilos de aprendizaje visual o auditivo, dadas las características de los contenidos de VoD.

7.5.2 Resultados de tiempo de ejecución

Dado que hay una correlación directa entre la precisión del SR propuesto y el número de asociaciones usado, no resulta conveniente usar un número muy grande de éstas si la viabilidad computacional no puede ser garantizada, especialmente durante el uso de la aplicación de usuario. Por esta razón, se realizó un estudio basado en el tiempo de ejecución.

La Figura 7.5 muestra el tiempo de ejecución consumido al desarrollar procesos importantes de los dos sistemas de recomendaciones. La Figura 7.5a muestra el tiempo del sistema construyendo una lista de seis recomendaciones (incluyendo la consulta y ponderación de segmentos de video y de los contenidos de VoD) para una sola competencia, sin incluir el proceso de filtrado. Con respecto al sistema propuesto, las medidas tomadas son independientes del número de asociaciones semánticas y más bajas que las del sistema sintáctico.

7. Evaluación del Sistema de recomendaciones

Por otra parte, la aplicación de usuario del SR semántico ejecuta procesos adicionales al sintáctico, específicamente, procesos relacionados el cómputo del *Peso del contexto de los mapas*, el *Peso del contexto educativo*, y la *Construcción de listas de recomendaciones*. Entonces, fue conveniente analizar los tiempos de ejecución promedio (obteniendo recomendaciones para un estudiante) relacionados con tales procesos de forma conjunta, los cuales se presentan en la Figura 7.5b. Se observa que hay una correlación directa entre el tiempo de ejecución y el número de asociaciones semánticas. Además, las medidas son bastante altas, del orden de los minutos. Sin embargo, es importante tener en cuenta que estos procesos se realizan una sola vez para todo un curso, suponiendo que las regiones de los mapas y los verbos de acción son interesantes para todos los estudiantes. Estos resultados también permiten concluir que la ejecución de la estrategia de recomendación en el SR propuesto conlleva más tiempo que el SR sintáctico, aunque, tal como se indicó anteriormente, ofrece mayor precisión.

Dados los tiempos altos inherentes al SR semántico propuesto, fue conveniente realizar un análisis dirigido a determinar un número de asociaciones semánticas por defecto para el cual se alcance un balance entre la precisión y el tiempo de ejecución. Tal análisis y la determinación de otros parámetros se discute en la Sección 7.5.3.

Los tiempos de ejecución de los procesos involucrados en la *Construcción de perfiles-ontología*, el cálculo del *Peso de la popularidad personalizada*, *Peso de la longitud de la asociación*, *Peso de la subsunción de los mapas de competencias*, y la *Similitud semántica* pueden ser consultados en el Anexo P.

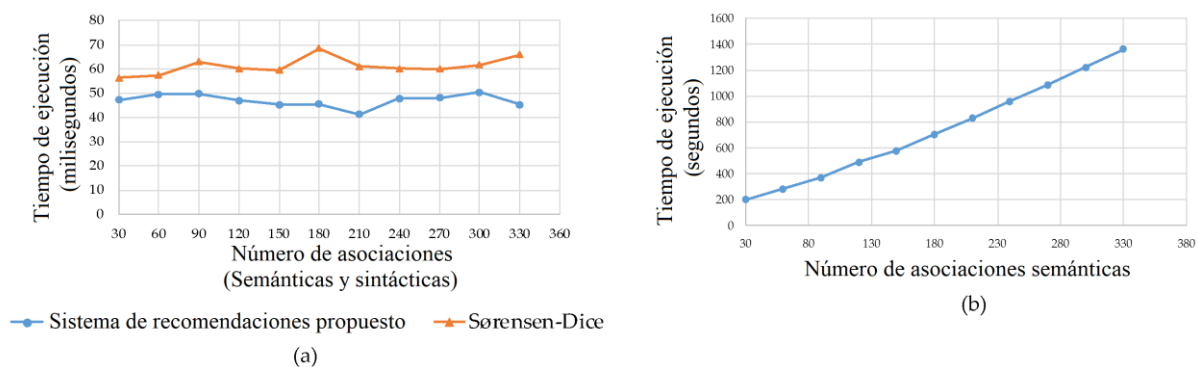


Figura 7.5 Resultados de tiempo de ejecución para: (a) obtener la lista de recomendaciones para una competencia; (b) ejecutar procesos propios del sistema propuesto. Fuente: propia.

7.5.3 Determinación de parámetros

Los valores por defecto de los pesos k_1, \dots, k_6 fueron establecidos experimentalmente con base en el juicio de los expertos. Esto fue hecho antes de obtener los resultados

de precisión y tiempos de ejecución mostrados en la Sección 7.5.1 y Sección 7.5.2, por medio de una adaptación del *método de ascenso coordinado* [3] de la siguiente forma:

1. Los pesos k_1 , k_2 , k_3 , k_4 , k_5 y k_6 se inicializaron con valores equivalentes (0.16). Esto teniendo en cuenta que, según [62], la suma de los factores de ponderación debe ser tal que $\sum_{i=1}^6 k_i = 1$.
2. Se obtuvo una lista de recomendaciones, la cual fue analizada por los expertos.
3. Los valores de los pesos se corrigieron de acuerdo al juicio de los expertos hasta alcanzar la máxima precisión posible.

Como resultado, los valores por defecto definidos son $k_1 = 0.1$, $k_2 = 0.1$, $k_3 = 0.1$, $k_4 = 0.1$, $k_5 = 0.3$ y $k_6 = 0.3$. Por consiguiente, se infiere que las relaciones de subsunción de los mapas y las similitudes semánticas de los temas son aspectos importantes en el cálculo de recomendaciones.

Con el propósito de determinar un número de asociaciones semánticas por defecto (δ), inicialmente se seleccionó un rango de precisión entre 0.6 y 0.7. Entonces, se asegura un mínimo de cuatro recomendaciones relevantes de seis. De acuerdo a la Figura 7.4, este rango corresponde al rango de δ entre 300 y 330. Notar que en la Figura 7.5b, los tiempos de ejecución que corresponden a tal rango son 1210 segundos (20.1 minutos) y 1370 segundos (22.8 minutos). Finalmente, los expertos fueron consultados acerca de la viabilidad de tales tiempos, concluyendo que estarían dispuestos a esperar 20 minutos para obtener recomendaciones para todo un curso. En conclusión, el número de asociaciones semánticas establecido es 300.

Se espera que los valores por defecto determinados en el estudio sean útiles para reducir la interacción entre el usuario y la aplicación del SR. Sin embargo, pueden ser cambiados a consideración.

Capítulo 8

Conclusiones y trabajos futuros

En el presente capítulo inicialmente se presentan las conclusiones de la investigación. Después, se esbozan los trabajos futuros. De forma complementaria, el Anexo Q presenta una discusión relacionada con el uso esperado de lo aquí propuesto, los resultados de precisión del Sistema de Recomendaciones y sus limitaciones; mientras que el Anexo R presenta un resumen de los productos derivados de la investigación.

8.1 Conclusiones

A continuación, se enumeran las conclusiones del presente trabajo.

1. La investigación realiza aportes en cuanto a: primero, modelos descriptivos y/o ontológicos que aborden las debilidades de los esquemas de metadatos y ontologías actuales para modelar: competencias, mapas de competencias, y competencias, mapas de competencias y contenidos de VoD de forma integral; segundo, modelos de evaluación basado en información lingüística computacionalmente aplicable a la estructura de los mapas de competencias; y tercero, estrategias de recomendación centradas en competencias por medio de l VoD. Específicamente, se propuso inicialmente: una extensión del IEEE RCD que define propiedades, vocabularios y tipos de datos específicos para describir competencias y relaciones entre éstas (Sección 3.2); un perfil de aplicación de *TV-Anytime* para describir contenidos de VoD (Sección 3.1); un *modelo descriptivo integral* que relaciona competencias, mapas de competencias y contenidos de VoD (Sección 3.3). Con base en el modelo descriptivo basado en metadatos, se propuso una *ontología sobre el dominio de los contenidos de VoD y las competencias* (Capítulo 4), propicia para desarrollar una *estrategia de recomendación* basada en razonamiento semántico. También, se propuso un *método computacional* basado en el modelo de *2-tuplas* para la evaluación lingüística de competencias capaz de calcular niveles de competencia precisos para ser incluidos en los perfiles de usuario del *sistema de recomendaciones*. Finalmente, se propuso una *estrategia de recomendación* que aporta la lógica para la obtención de las recomendaciones.

2. El *sistema de recomendaciones* propuesto aporta en la selección personalizada de contenidos de VoD, tomando como base el historial académico de los estudiantes y aspectos del currículo escolar como las competencias y los mapas de competencias.
3. La *estrategia de recomendación* propuesta sugiere contenidos de VoD relacionados con las necesidades de los estudiantes con base en el descubrimiento de vínculos lógicos entre los contenidos disponibles en el sistema, las competencias y los mapas de competencias a partir de las descripciones semánticas de éstos. Gracias a este proceso de razonamiento semántico, el sistema cuenta con el conocimiento para relacionar las necesidades educativas con los contenidos disponibles, favoreciendo así la personalización (Capítulo 6).
4. El *sistema de recomendaciones* propuesto es capaz de proveer una alta precisión en sus sugerencias, entendiéndose ésta como una medida de la pertinencia de los contenidos de VoD para apoyar el aprendizaje de los estudiantes (Sección 7.5.1) según el criterio de los expertos. En la configuración por defecto, el sistema alcanza una precisión del 65%, y aumenta directamente proporcional a la cantidad de asociaciones semánticas descubiertas. En concreto, para 300 asociaciones el sistema propuesto supera en un 38% el rendimiento de un sistema basado en emparejamiento sintáctico que aplica el coeficiente Sørensen-Dice. Esta diferencia es estadísticamente significativa, de acuerdo a un estudio de significancia estadística. Los resultados se obtuvieron a partir de una evaluación en la que 10 expertos analizaron si las recomendaciones de contenidos de VoD podrían apoyar el aprendizaje de 41 estudiantes.
5. La precisión y el tiempo de ejecución del *sistema de recomendaciones* propuesto tienen una correlación directa con el número de asociaciones semánticas usadas en el proceso de filtrado (Sección 7.5.2). Dicho de otra forma, una mayor precisión implica un mayor consumo de tiempo de ejecución. Dado que la segunda variable representa un aspecto crítico desde el punto de vista de la viabilidad, se buscó experimentalmente una configuración por defecto para el sistema de tal forma que se provea un balance entre ambas (Sección 7.5.3).
6. El prototipo generado del sistema de recomendaciones *EduCompRS* provee listas de recomendaciones de contenidos de VoD asociadas a las competencias en las que los estudiantes tienen cierto grado de necesidad. Para esto, ofrece una interfaz que facilita la configuración de la estrategia de recomendación según el interés del usuario en: las debilidades de los estudiantes, instancias de la base de conocimiento y mapas de competencias específicos, la longitud de las asociaciones, las relaciones de subsunción entre las competencias y las similitudes semánticas entre

8. Conclusiones y trabajos futuros

competencias. El prototipo puede ser usado en entornos reales para apoyar a estudiantes y docentes en los procesos educativos a través de sugerencias personalizadas.

7. La *extensión del IEEE RCD para la descripción de competencias educativas* propuesto (Sección 3.2): 1) suma las fortalezas de diversos esquemas hallados en la literatura y aborda sus debilidades (Sección 2.2.2); y 2) es el resultado del consenso de una comunidad de uso, integrando elementos de datos de su interés identificados a partir de una evaluación de *calidad de los metadatos* (Sección 3.4).
8. El *esquema de metadatos propuesto para la descripción de competencias* posee las características de *completitud*, *conformidad con las expectativas* y *accesibilidad*. Esto se concluyó a partir de los resultados obtenidos en una evaluación de *calidad de los metadatos*, para lo cual se aplicaron dos instrumentos web de evaluación con la participación de 19 evaluadores de países como Colombia, Argentina, México y España (Sección 3.4).
9. La *ontología sobre el dominio de los contenidos de VoD y las competencias educativas* propuesta ofrece un modelo para la formalización de mapas de competencias de acuerdo con el borrador de estándar “*Simple Reusable Competency Map*” (SRCM) (Sección 4.2). Después de un análisis basado en la identificación de representaciones intermedias se construyó un modelo ontológico basado en el “*Simple Knowledge Organization System*” (SKOS), el cual es pertinente para formalizar conocimiento semiformal. La ontología provee un modelo de inferencias sobre la estructura de los mapas que permite inferir relaciones de subsunción entre las competencias. Estas relaciones son aprovechadas dentro del proceso de filtrado de asociaciones semánticas (Sección 6.4).
10. La *ontología sobre el dominio de los contenidos de VoD y las competencias educativas* propuesta describe semánticamente las competencias a través de atributos de descripción específicos y las relaciona con otros conceptos educativos a través de su integración con el conocido “*framework*” “*Learning Object Context Ontology*” (LOCO) (Sección 4.3).
11. La *ontología sobre el dominio de los contenidos de VoD y las competencias educativas* propuesta cumple la característica de *completitud*, por lo cual cubre todo el dominio de interés de la investigación, siendo esto apropiado para recuperar información aprovechable por el sistema de recomendaciones. Para esto, se consideraron los siguientes aspectos: *declaración de URIs*, *literales* y *tipos de datos*, *validación XML*, *profundidad máxima de la jerarquía de conceptos* y *preguntas de competencias formalizadas* (Sección 4.4).

12. El *método computacional basado en información cualitativa para la evaluación de competencias* es altamente efectivo, ofreciendo un porcentaje de aciertos del 99,4% y más alto en un 1.6% que el del enfoque basado en el principio de extensión. Estos resultados demuestran que en un entorno educativo real el método puede ser utilizado de forma confiable por los docentes. La diferencia de resultados entre el método propuesto y el enfoque basado en el principio de extensión es estadísticamente significativa, de acuerdo a un estudio de significancia estadística. Además, provee valores bajos en *tiempo de ejecución* y *consumo de memoria*, por lo cual es una fuente confiable de datos para formalizar perfiles de usuario para el sistema de recomendaciones y es viable para ser integrado a plataformas de e-learning o servicios web para apoyar procesos educativos reales. Esto parte de los resultados obtenidos en el marco de una evaluación en la que se usó una colección de datos de 85 mapas de competencias reales (Sección 5.3), incluyendo sus estructuras e historiales de niveles de competencia.
13. El prototipo generado del *método computacional basado en información cualitativa para la evaluación de competencias CMA-Tool* facilita la evaluación de mapas de competencias a partir de los *niveles de competencia* alcanzados por los estudiantes en las actividades de evaluación. Para esto, permite configurar gráficamente las estructuras de los mapas, indicando relaciones de subsunción entre competencias y actividades de evaluación. El prototipo puede ser usado en entornos reales para apoyar a docentes en los procesos de evaluación cualitativos.

8.2 Trabajos futuros

Existen diferentes caminos a seguir para evaluar y mejorar lo propuesto en el presente trabajo:

1. Realizar una iteración más de la evaluación del esquema de metadatos para la descripción de competencias. Ésta estaría enfocada en verificar si las modificaciones mencionadas en la Sección 3.4 representa mejoras en cuanto al cumplimiento de las características de la calidad de los metadatos.
2. Verificar si la ontología puede ser soportada por otras aplicaciones después de un proceso de conversión a formatos como JSON, HTML y CSV. Los resultados pueden ser usados, por ejemplo, para obtener modelos de conversión automáticos para la generación de registros y repositorios descriptivos a partir una base de conocimiento y viceversa.

8. Conclusiones y trabajos futuros

3. Evaluar cualitativamente el sistema de recomendaciones con miras a fortalecer la estrategia de recomendación propuesta en el Capítulo 6, por ejemplo, verificando si las propiedades usadas para descubrir las asociaciones fueron las más adecuadas (Sección 6.3), si los pesos para la ponderación de las asociaciones pueden ser interpretados para medir otros aspectos educativos de tal forma que se infiera nuevo conocimiento (Sección 6.4.1), o si el factor de olvido, usado en la construcción de perfiles-ontología, puede tener otras interpretaciones (Sección 6.2.3). Esto con base en el criterio de expertos del ámbito de la educación.
4. Incorporar técnicas basadas en confianza (tales como las propuestas en [3]) a la estrategia de recomendación propuesta. Específicamente, consiste en crear redes de confianza en las cuales se establezcan relaciones ponderadas entre estudiantes y creadores de contenido. Por ejemplo, en una plataforma web como Youtube, se pueden tener en cuenta variables como el número de “likes”, vistas y suscriptores para medir la confianza. Este aspecto podría ser incluido en el proceso de personalización como una nueva etapa de filtrado, por ejemplo, para dar mayor prioridad a los contenidos de los creadores con mayor confianza al generar las listas de recomendaciones, o incluyendo un nuevo componente de ponderación de asociaciones semánticas; otra forma de incluir la confianza es estableciendo una estrategia de recomendación que combine, siguiendo un enfoque híbrido, las recomendaciones del sistema propuesto y las basadas en confianza.
5. Extender la recomendación a programas educativos de televisión, para lo cual convendría inicialmente extender el esquema de metadatos y ontología de acuerdo a sus propiedades específicas y a las del servicio. Al respecto, *TV-Anytime* establece la información necesaria sobre tal tipo de contenidos, la cual puede ser formalizada. Ejemplos de esta son: los canales y horarios de presentación, el lenguaje y lugar de edición de los programas. Al considerar esta información, el sistema podría adaptarse dinámicamente, modificando la lista de recomendaciones ofrecida por la estrategia propuesta de acuerdo a: 1) la franja horaria en la cual el estudiante tiene disponibilidad o prefiere ver televisión; 2) los horarios de repetición de los programas; 3) los lugares en los que se centra el programa, en caso de estar vinculados con las temáticas involucradas en las competencias. Estos son unos ejemplos de cómo se podrían adaptar la estrategia de recomendación a las características de los contenidos y servicio de televisión. Sin embargo, a futuro sería propicio analizar a fondo *TV-Anytime* para establecer las variables y reglas que permitan dirigir la construcción de los perfiles y el proceso de filtrado.
6. En la Sección 5.1 se discutió la dificultad inherente a la creación de descripciones de competencias y mapas de competencias. Con el propósito de abordar este

problema y asistir a los docentes en la creación de descripciones de calidad, se plantea como trabajo futuro la creación de herramientas que se basen en el esquema de metadatos y la ontología propuestos para: 1) generar esquemas colaborativos para la creación, reutilización y enriquecimiento de descripciones; y 2) utilizar técnicas de análisis de texto para la creación automática de descripciones a partir del lenguaje natural [117], [20]. Las herramientas podrían ofrecer una interfaz que facilite a los docentes la generación de descripciones de competencias y la construcción gráfica de mapas para, posteriormente, crear vínculos entre estas entidades. Una vez finalizado el proceso de descripción, la herramienta podría generar archivos XML con un formato previamente establecido para la creación de repositorios y/o crear una base de conocimiento en RDF/XML que puede ser publicada en la web para conformar un ambiente de *Datos Abiertos Enlazados* (DAE). Entonces, los docentes podrían tomar y/o extender descripciones de competencias y mapas preconcebidos según sus intereses.

7. Añadir al SR propuesto una etapa colaborativa. Siguiendo un enfoque híbrido, el SR deberá elegir inteligentemente si ejecutar el enfoque propuesto (consciente del contexto) por ejemplo, cuando se desee enfocar las recomendaciones en algún aspecto específico de entre los propuestos (en las deficiencias del estudiante, los mapas de competencias, la similitud semántica, etc.); o ejecutar el enfoque colaborativo por ejemplo, cuando se tenga un conjunto de datos consolidado acerca de información sobre el interés de los usuarios por los contenidos (valoraciones o el impacto de las recomendaciones en los procesos educativos). Para concebir el enfoque colaborativo, será necesario determinar un modelado de usuarios, de los datos, y unos algoritmos, ya sean para el cálculo de la predicción o ponderación de los contenidos de VoD, o de *machine learning* para su clasificación. En este trabajo futuro se podría evaluar si el enfoque colaborativo aborda el problema de los altos tiempos de ejecución medidos para el SR propuesto en la presente investigación.
8. Proponer herramientas que se basen en el esquema de metadatos y la ontología propuestos para:
 - Generar modelos colaborativos para la creación, reutilización y enriquecimiento de descripciones. A través de interfaces gráficas, los docentes podrían crear descripciones de competencias, mapas de competencias y contenidos de VoD. Otros docentes podrían tomar tales descripciones y realizar las mejoras que crean convenientes, lo cual supondría un aumento paulatino de su calidad. La herramienta podría incorporar un sistema de *ranking* que registre explícitamente las valoraciones dadas por los docentes, de manera que se facilite la recuperación de las mejores descripciones.

8. Conclusiones y trabajos futuros

- Utilizar técnicas de análisis y minería de texto para la creación automática de descripciones a partir del lenguaje natural. Usando una interfaz gráfica, los docentes podrían ingresar las descripciones de las competencias utilizando lenguaje natural. A través de técnicas como la propuesta en [20], del texto se podrían eliminar *palabras vacías* (por ejemplo, preposiciones, conjunciones, pronombres y artículos) y *lematizar* términos. Después, retomando las ideas de [128], con algoritmos de *machine learning* (por ejemplo, *árboles de decisión* o *Naïve Bayes*) se podrían clasificar los términos resultantes en los elementos del esquema de metadatos propuesto. Dado que tales elementos son conocidos (porque están definidos en el esquema), los algoritmos se podrían aplicar de manera supervisada, alimentándolos con el histórico de las descripciones de mayor calidad. Por otra parte, debido a la variabilidad y ambigüedad léxica, retomando las ideas de [129] se podría crear una ontología que formalice todo el lenguaje en torno a las competencias y contenidos de video y una lógica que la utilice para desambiguar términos.

Para una u otra solución, las herramientas podrían generar archivos XML con un formato previamente establecido para la crear o nutrir repositorios de metadatos y/o crear o enriquecer una base de conocimiento en RDF/XML que puede ser publicada en la web para conformar un ambiente de *Datos Abiertos Enlazados*. Entonces, los docentes podrían tomar y/o extender descripciones preconcebidas de competencias, mapas y contenidos que han sido creadas por otros docentes, según sus intereses para alimentar el sistema de recomendaciones.

9. Extender la recomendación a otros tipos de *Objetos de Aprendizaje* (OA) (por ejemplo, páginas web, cursos en línea, textos y presentaciones). Se puede realizar vinculando el esquema de metadatos propuesto para la descripción de contenidos de VoD (Capítulo 3) a esquemas más generales como el estándar *Learning Object Metadata* (LOM) o el esquema *Objetos de Aprendizaje Abiertos* (OBAA) desarrollado por la *Universidade Federal do Rio Grande do Sul* de Brasil [75] primero a nivel de metadatos y después a nivel de ontologías. La manera más natural de hacerlo sería creando inicialmente perfiles de aplicación de LOM u OBAA, adaptando a sus metadatos los elementos y vocabularios propuestos, por ejemplo, considerando que hay algunos elementos que son semejantes. Éste es el caso del *identificador*, *título*, *lenguaje*, *duración*, *rol del usuario*, *descripción*, *palabras clave*, y demás. Otros elementos como la URI, las referencias a competencias y todo lo relacionado con los segmentos de vídeo, deberán ser incluidos a LOM u OBAA como nuevos metadatos. Considerando que en la ontología propuesta se definió a la clase *Media resources* (cuyas instancias representan los contenidos de VoD) de “*Ontology for media resources*” como una subclase de la clase *Learning Object* de

“*Learning Object Metadata Ontology*” (ontología para LOM), entonces LOM y el esquema para VoD propuesto ya están adaptados a nivel de ontologías. Esta decisión permitió reconocer a los contenidos de VoD como un tipo de objeto de aprendizaje. Para lograr un resultado similar con OBAA, se puede realizar un análisis de la ontología propuesta por Thompsen, Rolim y Vicari [130]. Con la intención de conocerla y adaptarla, será importante ejecutar una metodología como la utilizada en el presente trabajo (Capítulo 4) para inicialmente determinar sus *Representaciones Intermedias* (RI) y posteriormente realizar su integración con “*Ontology for media resources*”. Con respecto a la estrategia de recomendación, ésta deberá ajustarse para que el descubrimiento y ponderación de los OA obedezcan a las características de la ontología resultante, especialmente, en cuanto a la definición de las consultas SPARQL.

10. Enriquecer la ontología con base en la conceptualización ontológica propuesta por Ovalle, Salazar y Duque [131], en la cual los *objetivos educativos* y sus *prerrequisitos* se definen como conceptos, además de que se definen las relaciones binarias entre éstos. La idea es adaptar tal conceptualización a la ontología propuesta en el Capítulo 4, primero estableciendo a las competencias como objetivos educativos, segundo, definiendo a las competencias como prerrequisito en el aprendizaje de otras en el marco de los mapas de competencias, y finalmente, definiendo las propiedades que permiten vincular estos conceptos. Esto supone la posibilidad de establecer un orden en el aprendizaje para, por ejemplo, adaptar cursos virtuales al progreso en el aprendizaje de manera personalizada.

Referencias bibliográficas

- [1] D. Levy, «Information overload,» de *The Handbook of Information and Computer Ethics*, Hoboken, Wiley, 2008, pp. 497-515.
- [2] E. Papaleontiou, *Metacognition and Theory of Mind*, Newcastle: Cambridge Scholars Publishing, 2008.
- [3] X. Zheng, C. Chen, J. Hung, W. He, F. Hong y Z. Lin, «A Hybrid Trust-Based Recommender System for Online Communities of Practice,» *IEEE Transactions on Learning Technologies*, vol. 8, nº 4, pp. 345-356, 2015.
- [4] A. Obaya, Y. Vargas y G. Delgadillo, «Aspectos relevantes de la educación basada en competencias para la formación profesional,» *Educación química*, vol. 22, nº 1, pp. 63-68, 2011.
- [5] UNESCO, «General education quality analysis/Diagnosis framework,» 2012. [En línea]. Available: <https://goo.gl/fdDnFP>. [Último acceso: 26 01 2018].
- [6] D. F. Duran, G. E. Chanchí y J. L. Arciniegas, «A Fuzzy Approach for Assessing Educational Competencies based on Competency Maps,» *Communications in Computer and Information Science*, vol. 735, pp. 372-386, 2017.
- [7] M. García y J. Gairín, «Los Mapas de Competencias: una Herramienta para mejorar la Calidad de la Formación Universitaria,» *Revista Iberoamericana sobre Calidad, Eficacia y Cambio en Educación*, vol. 9, nº 1, pp. 84-102, 2011.
- [8] J. Tarus, Z. y A. Yousif, «A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining,» *Future Generation Computer Systems*, vol. 72, pp. 37-48, 2017.
- [9] M. Dascalu, C. Bodea, A. Moldoveanu, A. Mohora, M. Lytras y P. Ordoñez, «A recommender agent based on learning styles for better virtual collaborative learning experiences,» *Computers in Human Behavior*, vol. 45, pp. 243-253, 2015.
- [10] J. Bobadilla, F. Serradilla y A. Hernando, «Collaborative filtering adapted to recommender systems of e-learning,» *Knowledge-Based Systems*, vol. 22, nº 4, pp. 261-265, 2009.

- [11] S. Fraihat y Q. Shambour, «A Framework of Semantic Recommender System for e-Learning,» *Journal of Software*, vol. 10, nº 3, pp. 317-330, 2015.
- [12] C. Porcel, A. G. Lopez-Herrera y E. Herrera-Viedma, «A recommender system for research resources based on fuzzy linguistic modeling,» *Expert Systems with Applications*, vol. 36, nº 3, p. 5173–5183, 2009.
- [13] Z. Sevarac, V. Devedzic y J. Jovanovic, «Adaptive neuro-fuzzy pedagogical recommender,» *Expert Systems with Applications*, vol. 39, nº 10, p. 9797–9806, 2012.
- [14] M. Salehi y I. Kmalabadi, «A Hybrid Attribute–based Recommender System for E–learning Material Recommendation,» en *International Conference on Future Computer Supported Education*, Seoul, 2012.
- [15] M. Hsu, «Proposing an ESL recommender teaching and learning system,» *Expert Systems with Applications*, vol. 34, nº 3, p. 2102–2110, 2008.
- [16] J. Serrano, F. Romero y J. Olivas, «Hiperion: A fuzzy approach for recommending educational activities based on the acquisition of competences,» *Information Sciences*, vol. 248, pp. 114-129, 2013.
- [17] M. Holenko y N. Hoic-Bozic, «Recommender system for Web 2.0 supported eLearning,» en *Global Engineering Education Conference (EDUCON)*, Istanbul, 2014.
- [18] A. Anaya, M. Luque y T. García-Saiz, «Recommender system in collaborative learning environment using an influence diagram,» *Expert Systems with Applications*, vol. 40, nº 18, p. 7193–7202, 2013.
- [19] M. Hsu, «A personalized English learning recommender system for ESL students,» *Expert System with Applications*, vol. 34, nº 1, pp. 683-688, 2008.
- [20] P. Montuschi, F. Lamberti, V. Gatteschi y C. Demartini, «A Semantic Recommender System for Adaptive Learning,» *IT Professional*, vol. 17, nº 5, pp. 50-58, 2015.
- [21] B. Vesin y A. Klasnja-Milicevic, «Applying Recommender Systems and Adaptive Hypermedia for e-learning Personalization,» *Computing and Informatics*, vol. 32, nº 3, pp. 629-659, 2013.
- [22] B. Vesin, M. Ivanovic, A. Klasnja-Milicevic y Z. Budimac, «Protus 2.0: Ontology-based semantic recommendation in programming tutoring system,» *Expert Systems with Applications*, vol. 39, nº 15, pp. 12229-12246, 2012.

- [23] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, y E. Duval, «Context-Aware Recommender Systems for Learning: A Survey and Future Challenges,» *IEEE Transactions on Learning Technologies*, vol. 5, nº 4, pp. 318-335, 2012.
- [24] H. Thanh, H. Nguyen y N. Thai. , «A Semantic Approach in Recommender Systems,» en *International Conference on Future Data and Security Engineering*, Can Tho City, 2016.
- [25] Y. Blanco, Propuesta metodológica para el razonamiento en sistemas de recomendación personalizada y automática, Vigo: Universidad de Vigo, 2007.
- [26] D. Jannach, M. Zanker, A. Felfernig y G. Friedrich, Recommender systems: an introduction, Cambridge: Cambridge University Press, 2011.
- [27] C. Desrosiers y G. Karypis, «A Comprehensive Survey of Neighborhood-based Recommendation Methods,» de *Recommender systems handbook*, New York, Springer, 2011, pp. 107-144.
- [28] K. Fung, «A Fuzzy Qualitative Evaluation System: A multi-granular aggregation approach using fuzzy compound linguistic variable,» *Journal of Intelligent & Fuzzy Systems*, vol. 24, pp. 61-78, 2013.
- [29] J. Montero, F. Alias, D. Badía, D. Fonseca y L. Vicent, «A method for designing automatic assessment systems based on teachers reasoning for evaluating subjective engineering student's competences,» *Information Systems and Technologies (CISTI), 2014 9th Iberian Conference on*, 2014.
- [30] Z. Zhaojun, Q. Xiao y L. Xiaoqing, «Competency Assessment Study Based on IPCA-Fuzzy Comprehensive Evaluation,» *Management and Service Science (MASS), 2010 International Conference on*, pp. 1-9, 2010.
- [31] I. Berdrow y F. Evers, «Bases of competence: an instrument for self and institutional assessment,» *Assessment & Evaluation in Higher Education*,, vol. 35, nº 4, pp. 419-434, 2010.
- [32] M. J. García, «Diseño y validación de un modelo de evaluación por competencias en la universidad,» *Tesis doctoral. Universidad autónoma de Barcelona*, 2010.
- [33] N. Tho y L. Hoang, «HIFCF: An effective hybrid model between picture fuzzy clustering and intuitionistic fuzzy recommender systems for medical diagnosis,» *Experts Systems with Applications*, vol. 42, nº 7, pp. 3682-3701, 2015.

- [34] M. Ghavipour y M. Reza, «An adaptive fuzzy recommender system based on learning automata,» *Electronic Commerce Research and Applications*, vol. 20, pp. 105-115, 2016.
- [35] D. Allemang y J. Hendler, *Semantic Web for the Working Ontologist*, Waltham MA: Morgan Kaufmann, 2011.
- [36] ETSI, « Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"), ETSI TS 102 822-3-1,» Sophia Antipolis, 2011.
- [37] IEEE Computer Society, «IEEE Standard for Learning Technology—Data Model for Reusable Competency Definitions - 1484.20.1-2007,» New York, 2007.
- [38] Ostyn Consulting, «Proposed Draft Standard for Learning Technology-Simple Reusable Competency Map,» 22 Feb 2006. [En línea]. Available: <https://bit.ly/2Ty0rVY>
- [39] D. Vrandečić, «Ontology evaluation,» Universidad en Karlsruhe, Karlsruhe, 2010.
- [40] F. Ricci, L. Rokach y B. Shapira, «Introduction to Recommender Systems,» *Recommender systems handbook*, New York: Springer, 2015, pp. 1-38.
- [41] N. Villegas, C. Sánchez, J. Díaz-Cely y G. Tamura, «Characterizing Context-Aware Recommender Systems: A Systematic Literature Review,» *Knowledge-Based Systems*, vol. 140, pp. 173-200, 2018.
- [42] D. Duran, R. Cerón y J. Arciniegas, «Architecture for the Support of the Video on Demand Service for Virtual Academic Communities on IPTV,» en *6 Colombian Computing Congress*, pp. 1-6, 2012.
- [43] S. Chang, D. Anastassiou, A. Eleftheriadis y J. Pavlik. *Video on Demand Systems: Technology, Interoperability and Trials*. Massachusetts: Springer, 2007.
- [44] M. Ramírez, «Posibilidades del uso educativo de Youtube,» *Ra Ximhai*, vol. 12, nº 6, pp. 537-546, 2016.
- [45] C. Snelson y P. Elison, «Using YouTube videos to engage the affective domain in e-learning,» *Research, Reflections and Innovations in Integrating ICT in Education*, vol. 3, pp. 1481-1485, 2009.

- [46] R. Berk, «Multimedia Teaching with Video Clips: TV, Movies, YouTube, and mtvU in the College Classroom,» *International Journal of Technology in Teaching and Learning*, vol. 5, nº 1, pp. 1-21, 2009.
- [47] C. Bonk, «YouTube Anchors and Enders: The Use of Shared Online Video Content as a Macrocontext for Learning,» *Asia-Pacific Collaborative education Journal*, vol. 7, nº 1, pp. 13-24, 2011.
- [48] Oxford University Press, «English Oxford Living Dictionaries,» [En línea]. Available: <https://en.oxforddictionaries.com/definition/competence>.
- [49] D. Burgos, E. Herder y D. Olmedilla, «TENCompetence: Construyendo la Red Europea para el Desarrollo Continuo de Competencias,» *Revista Iberoamericana de Inteligencia Artificial*, vol. 11, nº 33, pp. 79-84, 2007.
- [50] MedBiquitous Consortium, «Competency Object Specification,» [En línea]. Available: <https://bit.ly/2VCFruT>.
- [51] European Qualifications Framework, «El Marco Europeo de Cualificaciones para el aprendizaje permanente (EQF-MEC),» 2009. [En línea]. Available: <https://bit.ly/2RWwjTC>.
- [52] IMS Global Learning Consortium, «Reusable Definition of Competency or Educational Objective,» 2002. [En línea]. Available: <https://bit.ly/2XBf8XN>.
- [53] Ministerio de Educación de Colombia, «Estándares básicos de competencias,» Enero 2006. [En línea]. Available: <https://bit.ly/2bTRTB5>.
- [54] M. Baca. Introducción a los metadatos: vías a la información digital. USA: Getty publications, 1999.
- [55] W. Moen, E. Stewart y C. McClure, «Assessing metadata quality: Findings and methodological considerations from an evaluation of the US Government information locator service,» *Advances in Digital Libraries Conference*, pp. 246-255, 1998.
- [56] IEEE Working Group, «IEEE Standard for Learning Object Metadata, IEEE 1484.12.1,» 2002.
- [57] Linked Open Vocabularies, «Learning Object Metadata Ontology.» [En línea]. Available: <http://lov.okfn.org/dataset/lov/vocabs/lom>.
- [58] W3C, «Ontology for media resources.» [En línea]. Available: <https://www.w3.org/TR/mediaont-10/>.

- [59] L. Gasser y S. Besiki, «A new framework for information quality,» ISRN UIUCLIS–2001/1+AMAS, 2001.
- [60] T. Gruber, «A translation approach to portable ontologies,» *Knowledge Acquisition*, vol. 5, nº 2, pp. 199-220, 1993.
- [61] H. Weigand, «A multilingual ontology-based lexicon for news filtering - the TREVI project,» en XV International Joint Conference on Artificial Intelligence, pp. 160-165, 1997.
- [62] K. Anyanwu y A. Sheth, «p-Queries: Enabling Querying for Semantic Associations on the Semantic Web,» en *12th international conference on World Wide Web*, pp. 690-699, 2003.
- [63] L. Martínez, R. Rodríguez y F. Herrera. *The 2-tuple Linguistic Model: Computing with Words in Decision Making*. New York: Springer, 2015.
- [64] F. Herrera, E. Herrera-Viedma y L. Martínez, «A Fuzzy Linguistic Methodology to Deal With Unbalanced Linguistic Term Sets,» *IEEE Transactions on Fuzzy Systems*, vol. 16, nº 2, pp. 354-370, 2008.
- [65] I. Saleh y S. Kim, «A fuzzy system for evaluating students' learning achievement,» *Expert Systems with Applications* 36, vol. 36, pp. 6236–6243, 2009.
- [66] F. Herrera y L. Martínez, «A 2-Tuple Fuzzy Linguistic Representation Model for Computing with Words,» *IEEE Transactions on Fuzzy Systems*, vol. 8, nº 6, pp. 746-752, 2000.
- [67] T. Ross. *Fuzzy Arithmetic and the Extension Principle*. New Jersey: John Wiley & Sons, 2011.
- [68] McGuinness, y N. F. Noy, «Desarrollo de Ontologías-101: Guía Para Crear Tu Primera Ontología,» Stanford University, Stanford, 2005.
- [69] B. Kitchenham, R. Pretorius y D. Budgen, «Systematic literature reviews in software engineering – A tertiary study,» *Information and Software Technology*, vol. 52, pp. 792–805, 2010.
- [70] P. Dwivedi y K. Bharadwaj, «e-Learning recommender system for a group of learners,» *Expert Systems*, vol. 32, nº 2, pp. 264–276, 2013.
- [71] A. Kaklauskas, E. Zavadskas, M. Seniut, V. Stankevicius, J. Raistenskis, C. Simkevicius, T. Stankevicius, A. Matuliauskaite, L. Bartkiene, L. Zemeckyte, R. Paliskiene, R. Cerkauskiene y V. Gribniak, «Recommender System to Analyze

- Student's Academic Performance,» *Expert Systems with Applications*, vol. 40, nº 15, pp. 6150–6165, 2013.
- [72] eCOTOOL, «eCOTOOL Competence Model.» [En línea]. Available: http://www.simongrant.org/pubs/eCOTOOL/eCOTOOL_CM.html.
- [73] ICOPER, «The ICOPER Reference Model for Outcome-based Higher Education,» [En línea]. Available: http://nm.wu.ac.at/nm/resources/icoper/ICOPER_D7.3b.pdf.
- [74] MedBiquitous Consortium, «Competency Framework Specifications,» 2012. [En línea]. Available: <https://bit.ly/2C4LKzO>.
- [75] J. Gluz y R. Vicari, «MILOS: Infraestructura de agentes para,» en *XXI Simpósio Brasileiro de Informática na Educação*, 2010.
- [76] Gobierno de España, «Perfil de aplicación LOM-ES V1.0,» [En línea]. Available: http://www.lom-es.es/norma_une.htm. [Último acceso: 26 02 2019].
- [77] A. Vargas, S. Baldassarri y J. Arciniegas, «Study and Comparison of Metadata Schemas for the description of Multimedia Resources,» *Communications in Computer and Information Science*, vol. 605, pp. 59-73, 2015.
- [78] E. Paslaru-Bontas. *A Contextual Approach to Ontology Reuse: Methodology, Methods and Tools for the Semantic Web*. Berlin: Freie Universitat Berlin, 2010.
- [79] Z. Pengshun y H. Jianguo, «Application of Fuzzy Hierarchy Evaluation in the Occupational Competency Assessment of the Vocational Colleges,» en *2012 International Symposium on Information Technology in Medicine and Education*, pp. 182-185, 2012.
- [80] J. Serrano, F. Romero y J. Olivas, «Hiperion: A fuzzy approach for recommending educational activities based on the acquisition of competences,» *Information Sciences*, vol. 248, pp. 114-129, 2013.
- [81] Z. Jia y L. Zhao, «Evaluation of Enterprise Core Competence Based on the Entropy and Triangular Fuzzy Number,» en *2010 International Conference on E-Business and E-Government*, pp. 2045-2048, 2010.
- [82] H. Zhang, «The multiattribute group decision making method based on aggregation operators with interval-valued 2-tuple linguistic information,» *Mathematical and Computer Modelling*, vol. 56, nº 1, pp. 27-35, 2012.

- [83] Association for Information Science and Technology, «Dublin Core Metadata Initiative,» [En línea]. Available: <http://dublincore.org/>.
- [84] F. Martínez y M. Amaya, «El papel de los metadatos en la Web Semántica,» *Biblioteca universitaria*, vol. 20, nº 1, pp. 3-10, 2017.
- [85] K. Mekhabunchakij, «Towards Modeling Linked Open Data for Decision Support: An example application of Thailand Tourism Linked Data Visualization,» en *The 2016 Management and Innovation Technology International Conference*, pp. 88-92, 2016.
- [86] Asociación española de normalización, «Orientación sobre la elaboración de un esquema de metadatos-Norma UNE-ISO 23081,» 2010.
- [87] M. Marzal , J. Calzada y A. Cuevas, «Desarrollo de un esquema de metadatos para la descripción de recursos educativos: el perfil de aplicación MIMETA,» *Revista española de documentación científica*, vol. 29, nº 4, pp. 551-571, 2006.
- [88] S. Agostinho , S. Bennett y B. Harper, «Developing a learning object metadata application profile based on LOM suitable for the Australian higher education context,» *Australasian Journal of Educational Technology*, vol. 20, nº 2, pp. 191-208, 2004.
- [89] D. Pons, J. Hilera, L. Fernández y C. Pagés, «A Learning Quality Metadata approach: Automatic quality assessment of virtual training from metadata,» *Computr Standards & Interfaces*, vol. 45, pp. 45-61, 2016.
- [90] J. Patti y A. Cepeda, «Citizenship competencies in Colombia: Learning from policy and practice,» *Conflict Resolution Quarterly*, vol. 25, nº 1, pp. 109-125, 2007.
- [91] UNESCO, «International Standard Classification of Education ISCED 2011,» 2012. [En línea]. Available: <https://bit.ly/2rFmZVK>.
- [92] Gobierno de España, «Anexo IV de la Guía para la aplicación del Perfil de Aplicación LOM-ES V1.0 en la Educación,» 2009. [En línea]. Available: <https://bit.ly/2H9vf9b>.
- [93] International Organization for Standardization, «ISO/IEC 10646:2012- Information technology - Universal Coded Character Set (UCS),» 2012.
- [94] European Qualifications Framework, «The european qualifications framework for lifelong learning,» [En línea]. Available: https://ec.europa.eu/ploteus/sites/eac-eqf/files/leaflet_en.pdf.

- [95] N. Palavitsinis , N. Manouselis y S. Sanchez, «Metadata quality in learning object repositories: A case study,» *The Electronic Library*, vol. 32, nº 1, pp. 62-82, 2014.
- [96] S. Miguel-angel. Handbook Of Metadata, Semantics And Ontologies. Singapur: World Scientific Publishing , 2013.
- [97] E. Méndez, «Cultura abierta: conocimiento compartido,» en *Anuario ThinkEPI 2015: Análisis de tendencias en información y documentación*, Barcelona, Editorial UOC, 2015, pp. 126-131.
- [98] Linked Open Vocabularies, «Learning Object Metadata Ontology,» [En línea]. Available: <https://lov.linkeddata.es/dataset/lov/vocabs/lom>.
- [99] World Wide Web Consortium, «Ontology for Media Resources,» [En línea]. Available: <https://bit.ly/2SGmtkS>.
- [100] N. Noy y D. McGuinness, «Desarrollo de Ontologías-101: Guía Para Crear Tu Primera Ontología,» Stanford University, Stanford, 2005.
- [101] M. Fernández, A. Gómez-Pérez y N. Juristo, «METHONTOLOGY: From Ontological Art Towards Ontological Engineering,» *Spring symposium series on ontological engineering*, pp. 33-40, 1997.
- [102] W3C, «Ontology for Media Resources 1.0.» [En línea]. Available: <https://www.w3.org/TR/mediaont-10/>.
- [103] W3C, «SKOS Simple Knowledge Organization System.» [En línea]. Available: <https://www.w3.org/2004/02/skos/>.
- [104] M. Ehrig. Ontology alignment: Bridging the Semantic Gap. New York: Springer Science & Business Media, 2006.
- [105] J. de Bruijn, M. Ehrig, C. Feier, F. Martíns, F. Scharffe y M. Weiten, «Ontology Mediation, Merging, and Aligning,» en *Semantic Web Technologies: Trends and Research in Ontology-based Systems*. Chichester: John Wiley & Sons, 2006, pp. 95-114.
- [106] M. Delgado, J. Verdegay y A. Vila, «On aggregation operations of linguistic labels,» *International journal of intelligent systems*, vol. 8, nº 3, pp. 351-370, 1993.
- [107] D. Duran, G. Chanchí y J. L. Arciniegas, «Evaluación de mapas de competencias educativas: una propuesta difusa basada en 2-tuplas,» *Revista Ibérica de Sistemas y Tecnologías de la Información*, nº 24, pp. 22-38, 2017.

- [108] A. Bargnesi, A. DiFabio, W. Hayes, G. Shibaev, C. Benz y H. Pyle, «JSON Graph Format.» [En línea]. Available: <http://jsongraphformat.info/>.
- [109] T. Cormen, C. Leiserson, R. Rivest y C. Stein. *Introduction To Algorithms*. Cambridge: MIT press, 2001.
- [110] GeeksforGeeks, «Depth First Search or DFS for a Graph.» [En línea]. Available: <https://www.geeksforgeeks.org/depth-first-search-or-dfs-for-a-graph/>.
- [111] P. Fernández y P. Díaz, «Significancia estadística y relevancia clínica,» 19 09 2001. [En línea]. Available: https://www.fisterra.com/mbe/investiga/signi_estadi/signi_estadisti2.pdf.
- [112] Y. Li, Y. Wang y X. Huang, «A Relation-Based Search Engine in Semantic Web,» *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, nº 2, p. 273–282, 2007.
- [113] B. Aleman-Meza, C. Halaschek-Wiener, I. Budak Arpinar, C. Ramakrishnan y A. Sheth, «Ranking Complex Relationships on the Semantic Web,» *IEEE Internet Computing*, vol. 9, nº 3, pp. 37-44, 2005.
- [114] K. Anyanwu, A. Maduko y A. Sheth, «SemRank: Ranking Complex Relationship Search Results on the Semantic Web,» en *14th international conference on World Wide Web*, pp. 117-127, 2005.
- [115] V. Jindal, S. Bawa y S. Batra, «A review of ranking approaches for semantic search on Web,» *Information Processing and Management*, vol. 50, pp. 416-425, 2014.
- [116] Y. Blanco, J. Pazos, A. Gil, M. Ramos, M. López, J. García, A. Fernández, R. Díaz y J. Bermejo, «A flexible semantic inference methodology to reason about user preferences in knowledge-based recommender systems,» *Knowledge-Based Systems*, vol. 21, p. 305–320, 2008.
- [117] W. IJntema, F. Goossen, F. Frasinca y F. Hogenboom, «Ontology-Based News Recommendation,» *Extending Database Technology*, pp. 1-6, 2010.
- [118] D. Duran, G. Chanchí, J. L. Arciniegas y S. Baldassarri, «A Semantic Recommender System for iDTV based on Educational Competencies,» *Communications in Computer and Information Science*, vol. 689, pp. 47-61, 2016.
- [119] D. Duran, G. Chanchí, J. L. Arciniegas y S. Baldassarri, «Sistema de Recomendación Semántico de Programas Educativos de TVdi: una propuesta

Basada en Competencias,» en *V Jornadas Iberoamericanas sobre Aplicaciones y Usabilidad de la Televisión Digital Interactiva*, La Habana, 2016.

[120] «Amazon,» [En línea]. Available: <https://www.amazon.com>.

[121] «eBay,» [En línea]. Available: <https://www.ebay.com/>.

[122] B. Miller, I. Albert, S. Lam, J. Konstan y J. Riedl, «Movielens unplugged: experiences with an occasionally connected recommender system,» *ACM International conference on Intelligent user interfaces*, pp. 263-266, 2003.

[123] S. Bhat, «Item-based Hybrid Recommender System for newly marketed pharmaceutical drugs,» en *International Conference on Advances in Computing, Communications and Informatics*, pp. 2107-2111, 2013.

[124] V. Zhao, M. Moh y T. Moh, «Contextual-Aware Hybrid Recommender System for Mixed Cold-Start Problems in Privacy Protection,» en *IEEE International Conference on Intelligent Data and Security*, pp. 400-405, 2016.

[125] G. Shani y A. Gunawardana, «Evaluating Recommendation Systems» de *Recommender systems handbook*, New York, Springer, 2011, pp. 256-257.

[126] MovieLens, «MovieLens Latest Datasets,» [En línea]. Available: <https://grouplens.org/datasets/movielens/latest/>.

[127] M. Carballo y E. Guelmes, «Algunas consideraciones acerca de las variables en las investigaciones que se desarrollan en educación,» *Universidad y Sociedad*, vol. 8, nº 1, pp. 140-150, 2016.

[128] C. Amrit , T. Paauw , R. Aly y M. Lavric, «Identifying child abuse through text mining and machine learning,» *Expert systems with Applications*, vol. Volume 88, nº 1, pp. 402-418, 2017.

[129] M. Wijewickrema, «Impact of an ontology for automatic text classification,» *Annals of Library and Information Studies*, vol. 61, pp. 263-272, 2014 .

[130] T. Thompsen, A. Rolim y R. Vicari, «Una Ontología alternativa para describir Objetos de Aprendizaje,» de *Objetos de Aprendizaje, Repositorios y Federaciones... Conocimiento para Todos*, Manizales, Universidad Nacional de Colombia, sede Manizales, 2014, pp. 149-159.

[131] D. Ovalle, O. Salazar y N. Duque, «Modelo de Recomendación Personalizada en Cursos Virtuales basado en Computación Ubicua y Agentes Inteligentes,» *Información Tecnológica*, vol. 25, nº 6, pp. 131-142, 2014.

- [132] ISO/IEC 15938-5 (2003), *Information technology - Multimedia content description interface - Part 5: Multimedia description schemes*, 2003.
- [133] Ministerio de Educación de la República Argentina, «Competencias de Educación Digital.» [En línea]. Available: <https://bit.ly/2EJxMFu>.
- [134] Ministerio de Educación de Perú, «Rutas del aprendizaje-¿Qué y cómo aprenden nuestros estudiantes?.» [En línea]. Available: <https://bit.ly/2TtuUoe>.
- [135] Github, «jgraph/jgraphx.» [En línea]. Available: <https://github.com/jgraph/jgraphx>.
- [136] Github, «json-path/JsonPath.» [En línea]. Available: <https://github.com/json-path/JsonPath>.
- [137] Github, «json-simple.» [En línea]. Available: <https://github.com/fangyidong/json-simple>.
- [138] W3C, «Linked Data.» [En línea]. Available: <https://www.w3.org/standards/semanticweb/data>.
- [139] D. Ovalle, O. Salazar y N. Duque, «Modelo de Recomendación Personalizada en Cursos Virtuales basado en Computación Ubicua y Agentes Inteligentes,» *Información Tecnológica*, vol. 25, nº 6, pp. 131-142, 2014.

Anexo A

Aspectos complementarios de la base conceptual

El presente anexo muestra aspectos complementarios a la base conceptual (Sección 2.1): características de calidad de los metadatos, ejemplos de aplicación de los modelos computacionales lingüísticos y asociaciones semánticas complejas.

A.1 Características de calidad de los metadatos

A continuación se hace una descripción resumida de las características de calidad propuestas en [55], las cuales son tenidas en cuenta en la presente investigación.

- *Compleitud*, la cual se enfoca en verificar que los elementos de metadatos puedan describir los objetos tan completamente como sea posible;
- *Precisión*, encargada de evaluar que los archivos de descripción sean correctos y factuales;
- *Procedencia*, la cual permite evaluar si el esquema de metadatos provee la información necesaria para facilitar los juicios de calidad realizados por expertos;
- *Conformidad con las expectativas*, por medio de la cual se evalúa si el esquema contiene los elementos que la comunidad espera encontrar;
- *Consistencia y coherencia lógica*, usada para verificar si los metadatos son consistentes con definiciones y conceptos de estándares usados para el objeto o en un dominio relacionado y entregado al usuario en una forma consistente;
- *Atemporalidad*, la cual es útil para evaluar si los metadatos han cambiado de acuerdo con el cambio de las características del objeto descrito;
- *Accesibilidad*, usada para verificar que los metadatos puedan ser leídos y entendidos por los usuarios.

A.2 Ejemplos de aplicación de modelos computacionales lingüísticos

Ejemplo A.1. Un ejemplo, tomado de [63], indica que una compañía de distribución necesita renovar su sistema de cómputo, entonces contrata a una consultora para llevar a cabo una revisión de las diferentes posibilidades existentes en el mercado para decidir cuál es la mejor opción para cubrir sus necesidades. Las alternativas son las siguientes:

1. x_1 : UNIX.
2. x_3 : AS/400.
3. x_2 : WINDOWS.
4. x_4 : VMS

La consultora tiene un grupo de cuatro departamentos de consultoría así:

5. p_1 : Análisis de costo
6. p_3 : Análisis de riesgo.
7. p_2 : Análisis de sistema.
8. p_4 : Análisis de tecnología.

Cada departamento provee un vector de rendimiento expresando sus resultados de evaluación para cada alternativa. Estos resultados se presentan en la Tabla A.1, los cuales son expresados de acuerdo al conjunto de términos $S = \{N, MB, B, M, A, MA, T\}$.

		Alternativas			
		x_1	x_2	x_3	x_4
Expertos	p_1	MB	M	M	L
	p_2	M	L	MB	A
	p_3	A	MB	M	M
	p_4	A	A	B	B

Tabla A.1. Resultados de la evaluación de expertos para el ejemplo. Fuente: [63].

Con el propósito de agregar (fusionar) todos los términos lingüísticos para obtener un resultado de evaluación para cada una de las alternativas, inicialmente se transforman los valores lingüísticos de la Tabla A.1, obteniendo la Tabla A.2.

		Alternativas			
		x_1	x_2	x_3	x_4
Expertos	p_1	(MB,0)	(M,0)	(M,0)	(L,0)
	p_2	(M,0)	(L,0)	(MB,0)	(A,0)
	p_3	(A,0)	(MB,0)	(M,0)	(M,0)
	p_4	(A,0)	(A,0)	(B,0)	(B,0)

Tabla A.2. Resultados de la evaluación de expertos convertidos a 2-tuplas para el ejemplo. Fuente: [63].

A. Aspectos complementarios de la base conceptual

Aplicando la Media Aritmética para x_1 , definida por medio de:

$$x_i = \Delta \left(\frac{1}{n} \sum_{i=1}^n \beta_i \right), \quad (\text{A.1})$$

Se obtiene:

$$x_1 = \Delta \left(\frac{1}{4} (1 + 3 + 4 + 4) \right) = \Delta(0.5).$$

A partir de este resultado, se obtiene la 2-tupla (M,0). Siguiendo un proceso similar, se da que el resultado para x_2 es (M,-0.5), para x_3 es (L,-0.25) y para x_4 es (M,-0.25). Al tomar una decisión de entre las alternativas, se puede ver que x_1 , x_2 y x_4 están empatadas porque fueron evaluadas con un valor M. Sin embargo, al analizar la el valor de traslación simbólica ofrecida por las 2-tuplas, se puede ver que el de x_1 es 0, mientras que el de x_2 es 0.5 y el de x_4 es 0.25. En conclusión, la mejor opción es x_1 .

Ejemplo A.2. Al igual que en el Ejemplo A.1, se busca agregar los términos lingüísticos para obtener un resultado final para la alternativa x_1 . Teniendo en cuenta las funciones triangulares mostradas en la Figura 2.2 (Sección 2.1.7), las cuales tienen la forma (a_{ij}, b_{ij}, c_{ij}) , cada término se representa de la siguiente forma: N=(0,0,0.17), MB=(0,0.17,0.33), B=(0.17,0.33,0.5), M=(0.33,0.5,0.67), A=(0.5,0.67,0.83), MA=(0.67,0.83,1) y T=(0.83,1,1).

Aplicando el vector de rendimiento colectivo para x_1 , definido por:

$$x_j = \left(\frac{1}{m} \sum_{i=1}^m a_{ij}, \frac{1}{m} \sum_{i=1}^m b_{ij}, \frac{1}{m} \sum_{i=1}^m c_{ij} \right), \quad (\text{A.2})$$

se tiene:

$$x_1 = \left(\frac{1}{4} (0 + 0.33 + 0.5 + 0.5), \frac{1}{4} (0.17 + 0.5 + 0.67 + 0.67), \frac{1}{4} (0.33 + 0.67 + 0.83 + 0.83) \right) = (0.3325, 0.5025, 0.665).$$

Dado que el resultado no coincide exactamente con ninguno de los términos lingüísticos en S , entonces se aplica un proceso de aproximación lingüístico basado en la distancia Euclidiana entre el resultado colectivo $x_1 = (a_j, b_j, c_j)$ y cada término $s_l = (a_l, b_l, c_l) \in S$, la cual se define por medio de:

$$d(s_i, x_i) = \sqrt{P_1(a_l - a_j)^2 + P_2(b_l - b_j)^2 + P_3(c_l - c)^2}, \quad (\text{A.3})$$

donde P_1 , P_2 y P_3 son pesos que miden la representatividad de los parámetros a , b y c de la función de pertenencia del conjunto difuso S . Generalmente se tiene que $P_1 = 0.2$, $P_2 = 0.6$ y $P_3 = 0.2$, indicando que el parámetro b_{ij} es el más representativo de la función de pertenencia y que a_{ij} y c_{ij} son equitativamente representativos.

Por ejemplo, analizando el resultado colectivo $x_1 = (0.3325, 0.5025, 0.665)$ se puede observar que está entre los términos M=(0.33,0.5,0.67) y A=(0.5,0.67,0.83). Entonces,

en condiciones normales sería necesario medir la distancia Euclidiana entre x_1 y M y A. Sin embargo, en este ejemplo específicamente, es evidente que x_1 está más cercano a M que a A, por lo cual el resultado de la evaluación es M. Para calcular la distancia entre x_1 y $s_l = M = (0.33, 0.5, 0.67)$ se aplica la expresión (A.3) de la siguiente manera:

$$d(M, x_1) = \sqrt{0.2(0.3325 - 0.33)^2 + 0.6(0.5025 - 0.5)^2 + 0.2(0.665 - 0.67)^2} = 0.14.$$

Este resultado apoya lo que se supuso inicialmente: x_1 está muy cerca a M, por lo cual el resultado lingüístico de la evaluación para x_1 es M.

Realizando un proceso similar para las demás alternativas, se obtiene que el resultado de x_2 es M, el de x_3 es L y el de x_4 es M. Al tomar una decisión acerca de la selección de una de las alternativas, se puede ver que hay un empate entre x_1 , x_2 y x_4 , los cuales fueron evaluados con un valor M.

A.3 Asociaciones semánticas complejas

Asociación semántica compleja es un concepto definido en el marco del proyecto de investigación *SemDis* (“*Semantic Discovery*”) [62] relacionado con la capacidad para consultar y descubrir relaciones semánticas ocultas (o indirectas) en una base de conocimiento “*Resource Description Framework*” (RDF). Conscientes de que los lenguajes de consulta semántica como SPARQL exigen que el usuario tenga un conocimiento detallado de la base de conocimiento para extraer todas las entidades relacionadas con otras por medio de sus relaciones, los integrantes de *SemDis* desarrollaron un sistema que ofrece capacidades para consultar y descubrir relaciones semánticas complejas. El objetivo de tal sistema, es consultar todas las posibles relaciones entre dos entidades.

Las asociaciones semánticas complejas son esencialmente secuencias de propiedades de la forma $\rho(e_1, e_n)$ que enlazan dos entidades en la consulta. Las dos entidades e_1 y e_n están semánticamente asociadas si existe una o más secuencias de propiedades de la forma $e_1, p_1, e_2, p_2, e_3, \dots, e_{n-1}, p_{n-1}, e_n$ en un grafo RDF donde e_i , $1 \leq i \leq n$, son entidades, y cada p_j , $1 \leq j \leq n$, es una relación (propiedad) entre entidades e_j y e_{j+1} . Por notación, en este trabajo las entidades y relaciones involucradas en una asociación semántica compleja son llamadas de forma genérica como nodos de la asociación. Las asociaciones semánticas son relaciones complejas que abarcan esquemas heterogéneos y, en consecuencia, propiedades y entidades heterogéneas. Por ejemplo, en el extracto de una base de conocimiento mostrado en la Figura A.1, expuesto en [62] existe una asociación semántica entre las instancias *&r1 (Pablo Picasso, de la clase Painter)* e *&r3 (Reina Sofia, de la clase Museum)*,

A. Aspectos complementarios de la base conceptual

debido a que existen unos atributos como *paints* y *exhibited*, y una instancia como *&r2* (*oil on canvas*, de la clase *Painting*) que juntos constituyen un camino entre ambas. Este tipo de conocimiento accionable posibilita encontrar conexiones entre diferentes personas, lugares y eventos. Precisamente, esta característica es aprovechada en el presente trabajo para vincular competencias semánticamente y e inferir conocimiento útil para la generación de recomendaciones, características que otras técnicas como las basadas en *emparejamiento sintáctico* no proveen.

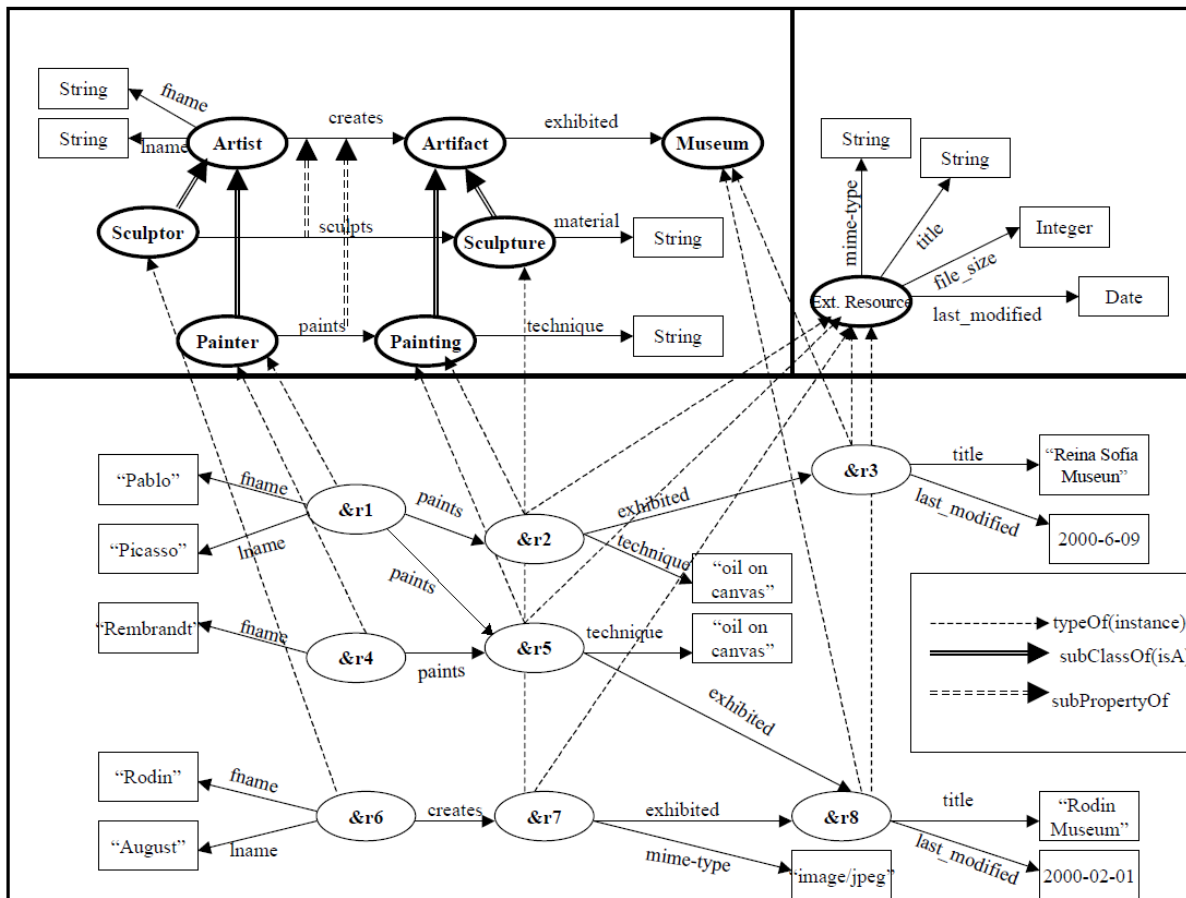


Figura A.1 Extracto de una base de conocimiento de un portal cultural. Fuente: [62].

Con el propósito de ofrecer flexibilidad para el usuario en cuanto al descubrimiento de vínculos entre entidades con base en sus preferencias, iniciativas como *SemDis* [62] y *SemRank* [114] han propuesto diferentes enfoques que buscan ponderar las asociaciones semánticas. En el área de los motores de búsqueda, esto se traduce en la presentación prioritaria de los resultados más significativos de forma personalizada, por encima de los menos importantes.

En las siguientes secciones, se repasa una serie de conceptos inherentes a las asociaciones semánticas complejas, los cuales son explotados por el SR propuesto para proveer las sugerencias.

A.3.1 Secuencias de propiedades

Para descubrir las asociaciones semánticas complejas es necesario analizar las propiedades formalizadas en la ontología considerada [5]. Mediante la exploración de tales propiedades, es posible identificar dos estructuras básicas:

- *Secuencia de propiedades (PS)*: es una secuencia de propiedades finita $[P_1, P_2, \dots, P_n]$ donde P_i es una propiedad definida en un esquema RDF. Por ejemplo, de la Figura A.1, se puede extraer la secuencia $PS_1 = [paints, technique]$, la cual vincula a algunas clases de la base de conocimiento (y por lo tanto de la ontología).
- *Instancia de una secuencia de propiedades (ps)*: es una secuencia de propiedades que vincula a las instancias de las clases vinculadas a través de una PS . Por ejemplo, de la Figura A.1 se puede extraer la instancia de una secuencia $ps_1 = [paints, technique]$. Como puede observarse, esta es igual a PS_1 . Sin embargo, es importante tener en cuenta que ps vincula a instancias de las clases, mientras que PS vincula las clases.

Para acceder a los nodos incluidos en las anteriores secuencias y en sus instancias, se definen las funciones $PS.NodesOfPS()$ y $ps.PSNodesSequence()$, respectivamente.

Por ejemplo, considerando los ejemplos de las secuencias PS_1 y ps_1 , se tiene que $PS_1.NodesOfPS()=[Painter,Painting,String]$ y $ps_1.PSNodesSequence()=[\&r1,\&r2,oil on canvas]$, Además, la primera instancia entregada por $ps.PSNodesSequence()$ o $PS.NodesOfPS()$ recibe el nombre de *origen*, mientras que la última se denomina *término*.

A.3.2 Tipos de asociaciones semánticas complejas

Sean ps_1 y ps_2 dos instancias de las secuencias de propiedades PS_1 y PS_2 , y x y y dos instancias definidas en la base de conocimiento, *SemDis* define cuatro tipos de asociaciones semánticas complejas, citadas a continuación

1. Asociación $\rho - pathAssociated$: una asociación $\rho - pathAssociated(x, y)$ se cumple si existe una secuencia de propiedades cuyo origen es x y cuyo término es y , oviceversa. De acuerdo con los ejemplos anteriores, dado que PS_1 es una secuencia de propiedades, entonces existe una asociación $\rho - pathAssociated$ entre $\&r1$ y "oil on canvas". Esta asociación se define como $ps_1 = \rho - pathAssociated(\&r1, oil on canvas)$.

A. Aspectos complementarios de la base conceptual

2. Asociación $\rho - join$: una asociación $\rho - join(x, y)$ se cumple si es posible encontrar dos secuencias PS_1 y PS_2 unidas ($PS_1 \bowtie_{\rho} PS_2$) y dos instancias de éstas ps_1 y ps_2 en las que se cumple una de las dos siguientes condiciones:

- x es el origen de ps_1 y y es el origen de ps_2 .
- x es el término de ps_1 y y es el término de ps_2 .

Considerando la Figura A.1, se puede observar que no existen dos secuencias de propiedades unidas por medio de una clase común, por lo tanto, no es posible establecer asociaciones $\rho - join$.

3. Asociación $\rho - cp$: una asociación $\rho - cp(x, y)$ se cumple si existen dos secuencias de propiedades unidas ($PS_1 \bowtie_{\rho} PS_2$) y dos instancias de éstas ps_1 y ps_2 en las que se cumple la asociación $\rho - join(x, y)$. Además, las propiedades de PS_1 y PS_2 deben ser de la forma `rdf:type (rdfs:subClassOf)`. Es decir, ambas secuencias tienen una primera propiedad `rdf:type` que permite identificar la clase a la que pertenece x (o y para la otra secuencia), y el resto de propiedades de la secuencia (de tipo `rdfs:subClassOf`) identifican relaciones de herencia entre dicha clase y otras definidas en la jerarquía de la ontología, hasta alcanzar el nodo de unión en el que terminan ambas secuencias y que provoca la asociación de tipo $\rho - join$. Con el propósito de evitar el descubrimiento de asociaciones poco significativas, se establecen algunas restricciones:

- Es posible especificar la clase más general de la ontología que puede actuar como nodo de unión, denominada por esta razón *techo* de la asociación $\rho - cp$. En el contexto de *SemDis*, la especificación del techo es una tarea encomendada al usuario del sistema.
- Si se considera que las secuencias unidas PS_1 y PS_2 originan una nueva secuencia, cuya longitud se corresponde con la suma de las longitudes individuales de cada una de ellas, el enfoque de *SemDis* también permite poner una cota superior a este parámetro, asegurando con ello que las instancias x y y no estén demasiado distantes en la ontología. Esto último es importante, ya que una longitud elevada se traduce en el descubrimiento de asociaciones poco relevantes para la consulta del usuario.

De acuerdo con la Figura A.1, al no haber asociaciones $\rho - join$, no es posible establecer asociaciones $\rho - cp$.

4. Asociación $\rho - iso$: una asociación $\rho - iso(x,y)$ se cumple si existen dos secuencias isomorfas ($PS_1 \cong_{\rho} PS_2$) y dos instancias de éstas ps_1 y ps_2 , tales que x es el origen de ps_1 y y es el origen de ps_2 .

Por ejemplo en la Figura A.1, $PS_1 = [paints, exhibited]$ y $PS_2 = [creates, exhibited]$ son isomorfas dado que *paints* y *created* son consideradas similares porque *paints* es subpropiedad de *created*. A partir de estas asociaciones, se tiene que $ps_1.PSNodesSequence()=[\&r1,\&r8]$ y $ps_2.PSNodesSequence()=[\&r6,\&r3]$. Como resultado, se establece una asociación $\rho - iso(\&r1,\&r6)$.

Como se pudo ver en los ejemplos, no todos los tipos de asociaciones pudieron ser identificados en la Figura A.1. En general, el tipo de asociaciones existentes en una base de conocimiento dependen de las características de ésta. Teniendo en cuenta que en el presente trabajo se explota el descubrimiento de asociaciones semánticas en el SR, fue necesario realizar un análisis sobre la estructura de la ontología aquí propuesta para determinar el tipo de asociaciones a utilizar. Esto puede ser consultado en la Sección 6.3, donde se discute el rol de las asociaciones dentro del proceso de filtrado del SR.

A.3.3 Ponderación de asociaciones semánticas complejas

Una vez se definieron los tipos de asociaciones semánticas complejas, diferentes iniciativas como *SemDis* [62] y *SemRank* [114], desarrollaron mecanismos para ponderar su relevancia con respecto a la consulta formulada. En la presente investigación se acogen las métricas propuestas por *SemDis*, debido a que pueden ser interpretadas y adaptadas a las características del contexto, lo cual es conveniente para inferir conocimiento con significado en el contexto durante la ejecución de una estrategia de recomendación personalizada como la que se propone en este trabajo. Mientras que las métricas de *SemRank* dependen de las características de la ontología en cuanto a su estructura.

Para *SemDis*, en una consulta realizada a un motor de búsqueda semántico, el usuario especifica dos entidades formalizadas en la base de conocimiento (esto implica que el usuario debe conocer la base de conocimiento). Posteriormente, el sistema recupera las asociaciones establecidas entre ambas entidades que aportan conocimiento útil en el dominio de aplicación del sistema. Para esto, se computa un valor de calificación para cada una de las asociaciones inferidas, el cual es más alto para las asociaciones más relevantes. Tal valor depende de diferentes criterios, siendo estos organizados en dos grupos de métricas: métricas semánticas y métricas estadísticas.

Métricas semánticas

A. Aspectos complementarios de la base conceptual

Las métricas semánticas consideran aspectos relacionados con las instancias y relaciones establecidas en la ontología. En este grupo, se encuentran tres métricas denominadas: *contexto*, *inclusión* y *confianza*.

1. Contexto

Al formular una consulta en *SemDis*, el usuario debe identificar el contexto de la misma. Para esto, debe indicar una o varias regiones dentro de la ontología (subgrafos), correspondientes a sus dominios de interés. Para cada una de tales regiones, se debe indicar un peso a modo de nivel de interés, de forma que los pesos de todas las regiones sumen uno. El objetivo, es que las asociaciones que involucran las clases y/o propiedades que hacen parte de las regiones indicadas por el usuario, sean mejor calificadas y, por lo tanto, más relevantes.

Sea A una asociación semántica, esto es, una secuencia compuesta por nodos (entidades) y aristas (relaciones) que conectan dos entidades. Sea $length(A)$ el número de relaciones y entidades de A . R_i representa la región i , esto es, el conjunto de clases y relaciones que capturan el dominio de interés del usuario. Dado que ambas entidades y relaciones contribuyen en la calificación, c es un componente de A (una entidad o una relación). Se definen convenientemente los siguientes conjuntos, usando la notación $c \in R_i$ para representar si el tipo (`rdf:type`) de c pertenece a la región R_i :

$$X_i = \{c | c \in R_i \wedge c \in A\}, \quad (A.4)$$

$$Z = \{c | (\forall i | 1 \leq i \leq n) c \notin R_i \wedge c \in A\}, \quad (A.5)$$

donde n es el número de regiones en el contexto de la consulta. Entonces, X_i es el conjunto de componentes de A en la i -ésima región y Z es el conjunto de componentes de A que no están en ninguna región contextual. Ahora, se define el peso del contexto de una asociación A , C_A , como:

$$C_A = \frac{1}{length(A)} \left(\sum_{i=1}^n (W_{R_i} \times |X_i|) \right) \times \left(1 - \frac{|Z|}{length(A)} \right), \quad (A.6)$$

donde n es el número de regiones, W_{R_i} es el peso de la región i -ésima. Por ejemplo, sea $ps_1.PSNodesSequence()=[\&r1,\&r2]$ los nodos de una asociación $A = \rho - pathAssociated(\&r1, \&r2)$, se busca ponderar el contexto de ps_1 . Suponiendo que el usuario ha expresado tener un interés especial en la región $R_1 = \{\&r2,\&r8\}$, con un peso $W_{R_1} = 1$. A partir de esta información se generan los siguientes conjuntos:

$X_1 = \{\&r2\}$, dado que $\&r2$ pertenece tanto a la asociación como a la región de interés, y $Z = \{\&r1\}$, dado que $\&r1$ pertenece a la asociación pero no a la región de interés. Considerando que $length(A) = 2$ (dado que A está compuesta por dos nodos), se calcula el contexto de A aplicando la expresión (A.6) de la siguiente manera:

$$C_A = \frac{1}{2}(1 \times 1) \times \left(1 - \frac{1}{2}\right) = 0.25.$$

2. Subsunción

Las clases en una ontología más bajas en la jerarquía pueden ser consideradas como instancias más especializadas que las de arriba, esto es, transportan información más detallada y tienen un significado más específico. Se define el peso de subsunción csw del i –ésimo componente, c_i , en una asociación A como:

$$csw_i = \frac{H_{c_i}}{H_{depth}}, \quad (A.7)$$

donde H_{c_i} es la posición de c_i en la jerarquía H (la clase más alta tiene un valor de 1) y H_{depth} es la altura total de la jerarquía de clases/relaciones de la rama común. Ahora, el peso total de la subsunción de una asociación se define como:

$$S_A = \prod_{i=1}^{length(A)} csw_i. \quad (A.8)$$

Por ejemplo, se quiere calcular la subsunción de la asociación $A = \rho - pathAssociated(\&r1,\&r2)$. Inicialmente se calcula la subsunción de cada componente de la asociación, para lo cual se tiene en cuenta la taxonomía de clases de la Figura A.1. Dado que a partir de la Figura A.1 no es posible conocer la altura de la taxonomía, se asume que esta es $H_{depth} = 4$ y que la posición de las clases *Painter* y *Painting* es $H_{c_1} = H_{c_2} = 2$, aplicando la expresión (A.7) para cada nodo de la asociación se tiene:

$csw_1 = csw_2 = \frac{2}{4} = 0.5$. Entonces, para calcular la subsunción de la asociación se aplica (2.6) de la siguiente manera:

$$S_A = 0.5 \times 0.5 = 0.25.$$

3. Confianza

Diferentes entidades y sus relaciones en una asociación semántica son originadas por diferentes fuentes. Algunas de éstas pueden ser más confiables que otras (por ejemplo, Reuters puede ser considerada como una fuente más confiable en noticias internacionales que otras organizaciones). Entonces, los valores de confianza necesitan ser asignados en los metadatos extraídos dependiendo de su fuente. Al computar los pesos de confianza de una asociación semántica, se sigue la intuición: la fortaleza de una asociación es tan fuerte como su enlace más débil [8]. t_{c_i} representa el valor de confianza asignado de un componente c_i . Se define el peso total de confianza de una asociación A como:

$$T_A = \min(t_{c_i}). \quad (A.9)$$

Suponiendo que la instancia r_1 pertenece a una fuente que tiene asignado un valor de confianza $t_{c_1} = 0.5$ y r_2 proviene de una fuente tal que $t_{c_2} = 0.6$, entonces la confianza de la asociación $A = \rho - pathAssociated(r_1, r_2)$ se calcula aplicando la expresión (A.9) de la siguiente manera:

$$T_A = \min(t_{c_1}, t_{c_2}) = 0.5.$$

Métricas estadísticas

Estas métricas consideran aspectos estadísticos de la ontología, basados principalmente en el número de instancias pertenecientes a las clases definidas en la misma. Son tres las métricas de este grupo: rareza, popularidad y longitud de la asociación.

1. Rareza

Dado el tamaño de los actuales bancos de pruebas de la web semántica, (por ejemplo, SWETO, TAP KB), existen muchas relaciones y entidades del mismo tipo. En *SemDis* consideran que, en algunas consultas, las entidades y relaciones que ocurren rara vez pueden ser consideradas más interesantes. Sin embargo, también puede darse lo contrario. Por ejemplo, en el contexto de lavado de dinero, a menudo los individuos se involucran en transacciones de casos comunes para evadir la detección. En este caso, las transacciones comunes (no raras) son usadas para lavar dineros de tal forma que los movimientos financieros pasen por alto [10]. Entonces el usuario podría determinar, dependiendo de la consulta, qué peso de rareza prefiere.

Se define la rareza de una asociación A , en términos de la rareza de sus componentes. Sea K la base de conocimiento (todas las entidades y relaciones), se define la rareza rar del i -ésimo componente, c_i , como:

$$rar_i = \frac{|M| - |N|}{|M|}, \quad (A.10)$$

donde,

$$M = \{res \mid res \in K\}, \quad (A.11)$$

$$N = \{res_j \mid res_j \in K \wedge \text{typeOf}(res_j) = \text{typeOf}(c_i)\}, \quad (A.12)$$

con la restricción de que si res_j y c_i son ambas del tipo `rdf:Property`, el sujeto y objeto de res_j y c_i deben tener el mismo `rdf:type`. Entonces, rar_i captura la frecuencia de ocurrencia del componente c_i , con respecto a toda la base de conocimiento. Ahora, se define el peso total de rareza R de una asociación A , como una función de todos sus componentes de la forma:

$$R_A = \frac{1}{\text{length}(A)} \times \sum_{i=1}^{\text{length}(A)} rar_i, \quad (A.13)$$

o:

$$R_A = 1 - \frac{1}{\text{length}(A)} \times \sum_{i=1}^{\text{length}(A)} rar_i, \quad (\text{A.14})$$

donde $\text{length}(A)$ es el número de nodos de A . Si un usuario quiere favorecer a las asociaciones raras, entonces se usa la expresión (A.13), por el contrario, si un usuario quiere favorecer a las asociaciones más comunes, entonces se usa la expresión (A.14). Por ejemplo, para calcular la rareza de la asociación $A = \rho - \text{pathAssociated}(\&r1, \&r2)$ al observar el extracto de la base de conocimiento de la Figura A.1 se contabiliza un total de ocho instancias ($\&r1$, $\&r2$, hasta $\&r8$) y seis relaciones entre éstas, por lo cual $|M| = 14$. Para calcular la rareza de A , inicialmente se calcula la rareza de cada uno de sus nodos. Para calcular la rareza de $\&r1$, es necesario contar las instancias del mismo tipo que $\&r1$. Dado que en el extracto hay dos entidades del mismo tipo de $\&r1$ (son $\&r1$ y $\&r2$, los cuales hacen parte de la competencia *Painter*), entonces $|N_1| = 2$. Así mismo, se puede observar que hay dos entidades del mismo tipo de $\&r2$ (son $\&r2$ y $\&r5$, los cuales hacen parte de la clase *Painting*), por lo cual $|N_2| = 2$. Aplicando la expresión (A.10) para ambas entidades se tiene:

$$rar_1 = rar_2 = \frac{18-2}{18} = 0.88.$$

Entonces, si se desea favorecer las asociaciones raras, la rareza de A se calcula aplicando la expresión (A.13) de la siguiente manera:

$R_A = \frac{1}{2} \times (0.88 + 0.88) = 0.88$. Este resultado sugiere que la asociación está compuesta por instancias raras dentro de la base de conocimiento. Además, de que es coherente con el hecho de que la fracción de entidades con el mismo tipo de $\&r1$ e $\&r2$ con respecto al total de entidades y relaciones de la base de conocimiento es baja. Por otra parte, si se desea favorecer las asociaciones más comunes, la rareza de A se calcula aplicando la expresión (A.14) de la siguiente manera:

$$R_A = 1 - 0.88 = 0.12.$$

2. Popularidad

Cuando se analizan las entidades en una asociación, se observa que algunas entidades tienen más relaciones entrantes y salientes que otras. Algo similar con el algoritmo de calificación de la página web de Kleinberg [11], así como el algoritmo *PageRank* usado por Google [12]. En el enfoque *SemDis*, las asociaciones con entidades que tienen una alta popularidad pueden ser más relevantes.

A. Aspectos complementarios de la base conceptual

Se define la popularidad de una asociación en términos de la popularidad de sus entidades. La popularidad p de la i –ésima entidad e_i en la asociación A se define como:

$$p_i = \frac{|pop_{e_i}|}{\max_{0 \leq j \leq n} (|pop_{e_j}|)}, \quad (\text{A.15})$$

donde $\text{type}(e_i) = \text{type}(e_j)$, n es el número total de entidades en la base de conocimiento. Entonces, pop_{e_i} es el conjunto de relaciones entrantes y salientes de e_i , y $\max_{0 \leq j \leq n} (|pop_{e_j}|)$ representa el tamaño del conjunto más grande entre todas las entidades en la base de conocimiento de la misma clase que e_i . Entonces, p_i captura la popularidad de e_i con respecto a todas las otras entidades de su mismo tipo en la base de conocimiento. Ahora, se define el peso total de la popularidad, P , de una asociación, A , como:

$$P_A = \frac{1}{n} \times \sum_{i=1}^n p_i, \quad (\text{A.16})$$

o:

$$P_A = 1 - \frac{1}{n} \times \sum_{i=1}^n p_i, \quad (\text{A.17})$$

donde n es el número de entidades (nodos) en A y p_i es la popularidad de la i –ésima entidad en A . Si un usuario quiere favorecer asociaciones populares, entonces se usa la expresión (A.16); por el contrario, si un usuario quiere favorecer las asociaciones menos populares, es usada la expresión (A.17). Por ejemplo, teniendo en cuenta la Figura A.1, al computar la popularidad de $A = \rho - \text{pathAssociated}(\&r1, \&r2)$, primero debe calcularse la popularidad de $\&r1$ e $\&r2$. Tal como se observa en el extracto de la base de conocimiento, se cuenta una relación saliente de $\&r1$ y ninguna entrante, por lo cual $|pop_{e_1}| = 1$. Así mismo, una relación entra a y sale de $\&r2$, por lo cual $|pop_{e_2}| = 2$. Ahora, se busca la entidad en la base de conocimiento con la mayor cantidad de relaciones entrantes y salientes que tienen el mismo tipo de $\&r1$. Dado que para todas las entidades de la clase *Painter* se da que $|pop_{e_i}| = 1$, entonces $\max_{0 \leq j \leq n} (|pop_{e_j}|) = 1$. Algo similar pasa con las entidades del mismo tipo de $\&r2$, dándose que $\max_{0 \leq j \leq n} (|pop_{e_j}|) = 2$. Para calcular la popularidad de $\&r1$ e $\&r2$ se aplica la expresión (A.15) de la siguiente manera:

$$p_1 = \frac{1}{1} = 1,$$

$$p_2 = \frac{2}{2} = 1.$$

Finalmente, si se quiere favorecer las asociaciones populares, para calcular la popularidad de A se aplica la expresión (A.16) de la siguiente forma:

$P_A = \frac{1}{2} \times (1 + 1) = 1$, lo cual sugiere que la asociación está compuesta por instancias populares. Por otro lado, si se quiere favorecer las asociaciones menos populares, se aplica la expresión (A.17) de la siguiente manera:

$$P_A = 1 - 1 = 0.$$

3. Longitud de la asociación

En algunas consultas, un usuario puede estar interesado en asociaciones más directas (esto es, asociaciones más cortas). En otros casos un usuario puede desear encontrar asociaciones indirectas (asociaciones más largas).

Se define el peso de la longitud de la asociación L , de una asociación A . Si un usuario quiere favorecer las asociaciones más cortas, se usa la expresión (A.18), de otra forma, se usa (A.19).

$$L_A = \frac{1}{length(A)}, \quad (A.18)$$

$$L_A = 1 - \frac{1}{length(A)}. \quad (A.19)$$

Para calcular la longitud de la asociación $A = \rho - pathAssociated(\&r1,\&r2)$, primero es importante saber que $length(A) = 2$, ya que A está compuesta por dos instancias. Entonces, si se busca favorecer las asociaciones más cortas, para calcular la longitud de la asociación de A se aplica la expresión (A.18) de la siguiente forma:

$L_A = \frac{1}{2} = 0.5$. Este resultado indica que hay una relación directa entre el origen y el término de A .

4. Criterio total de calificación

La calificación total de una asociación A suma las métricas antes expuestas de la siguiente forma:

$$W_A = k_1 \times C_A + k_2 \times S_A + k_3 \times T_A + k_4 \times R_A + k_5 \times P_A + k_6 \times L_A, \quad (A.20)$$

de tal forma que la suma de k_i ($1 \leq i \leq 6$) es 1, posibilitando el ajuste de los criterios de calificación (por ejemplo, la popularidad puede tener más peso que la longitud de la asociación). Esto provee un enfoque de calificación flexible dependiente de la consulta para evaluar la relevancia total de las asociaciones.

Como se puede observar, todas las métricas de ponderación dependen exclusivamente de las características de la ontología y de algunas condiciones generales del contexto. Con el propósito de vincular a éstas las características de las

A. Aspectos complementarios de la base conceptual

competencias y los mapas de competencias, en el presente trabajo se han seleccionado justificadamente las que pueden aportar conocimiento significativo dentro del contexto educativo, para ser adaptadas posteriormente tanto en su concepto como en su modelo matemático. Esto se trata con detalle en la explicación de la estrategia de recomendación en el Capítulo 6.

Anexo B

Análisis de esquemas de metadatos del ámbito de las competencias

Este anexo presenta una versión extendida del análisis de esquemas de metadatos relacionados con competencias mostrado en el estado del arte (Sección 2.2.2). A partir de éste se seleccionaron los esquemas base para la construcción del *esquema de metadatos para la descripción de contenidos de VoD de acuerdo a las competencias educativas* propuesto en el Capítulo 3.

Inicialmente, con respecto a las competencias y mapas de competencias, se halló un conjunto de esquemas y relaciones entre ellos, los cuales se muestran en la Figura B.1. Éstos han sido clasificados en tres grupos de la siguiente manera:

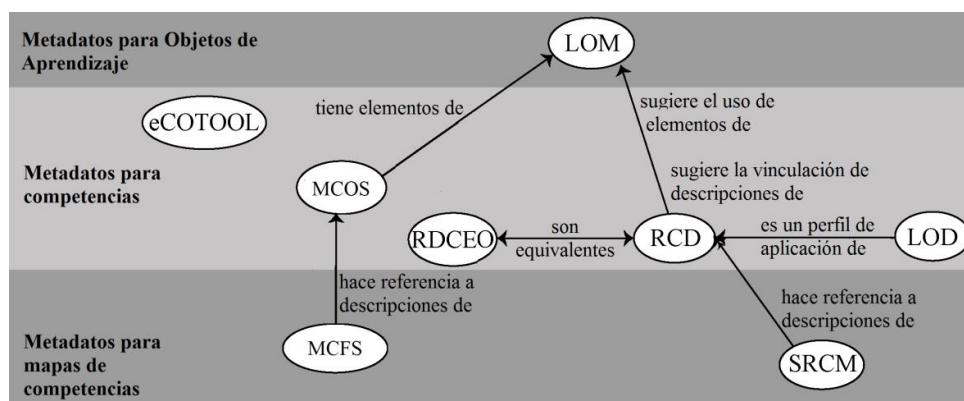


Figura B.1 Esquemas de metadatos relacionados con competencias. Fuente: propia.

1. *Esquemas de metadatos para Objetos de Aprendizaje.* Este tipo de esquemas han sido considerados porque proveen elementos que pueden ser adaptados para describir atributos específicos de las competencias y los contenidos de VoD. Esto considerando que los contenidos de VoD deben ser vistos como OA. En este grupo se encuentra el “*Learning Object Metadata*” (LOM) [56], el cual es el esquema más importante actualmente para la descripción de OA.
2. *Esquemas para competencias,* los cuales proveen elementos de datos para describir atributos de competencias. En este grupo se encuentran: “eCOTOOL

Competence Model” (eCOTOOL) [72], “*MedBiquitous Competency Object Specification*” (MCOS) [50], “*Reusable Definition of Competency or Educational Objective*” (RDCEO) [52], “*IEEE Reusable Competency Definition*” (IEEE RCD) [37] y “*Learning Outcome Definition*” (LOD) [73].

3. *Esquemas de metadatos para mapas de competencias*, los cuales ofrecen elementos de datos para definir relaciones entre descripciones de competencias realizadas por medio de esquemas del segundo grupo. En este grupo se encuentran: “*Simple Reusable Competency Map*” (SRCM) [38] y “*MedBiquitous Competency Framework Specification*” (MCFS) [74].

En la primera parte del análisis, se encontraron algunas relaciones entre los esquemas de metadatos. Por ejemplo, MCOS incluye elementos de datos de LOM directamente en su modelo, de tal forma que se establecen atributos comunes entre competencias y OA. Ésta es una práctica sugerida por IEEE RCD. También, se encontró que IEEE RCD sugiere vincular competencias y OA por medio de identificadores. Sin embargo, los esquemas no definen elementos de datos para tal fin. Adicionalmente, se encontró una equivalencia semántica entre RDCEO y IEEE RCD, definiendo elementos de datos sintácticamente diferentes, pero con el mismo significado. SRCM posibilita la construcción de mapas de competencias a través de la referencia de descripciones de competencias construidas por medio de IEEE RCD. Algo similar ocurre entre MCFS y MCOS.

Hasta aquí, se concluye que IEEE RCD ofrece la posibilidad de establecer un “*framework*” para vincular OA, competencias y mapas de competencias.

Debido a la variedad de las iniciativas mostradas, en la segunda fase del análisis se hizo una comparación de los esquemas para metadatos para competencias con base en la información que describen. Los resultados, incluyendo observaciones importantes, se muestran en la Tabla B.1. En resumen, los esquemas describen la siguiente información de forma complementaria:

- *Identificación* (I1): involucra elementos de datos para identificar y ubicar descripciones, facilitando su intercambio y reutilización. Generalmente, se utilizan elementos como *Identifier* y *Title*.
- *Descripción* (I2): es una descripción en lenguaje natural (legible sólo por humanos) hecha por su creador.
- *Conceptos relacionados* (I3): son afirmaciones en lenguaje natural relacionadas con la competencia descrita, las cuales son tomadas de un modelo formal.
- *Elementos adicionales* (I4): reúne elementos definidos para crear extensiones del esquema.

B. Análisis de esquemas de metadatos del ámbito de las competencias

- *Autor* (I5): es información acerca del creador de la extensión.
- *Descripción corta* (I6): es una descripción con el suficiente significado para entender el concepto de la competencia descrita.
- *Categorización KSC* (“*Knowledge*”, “*Skill*”, “*Competence*”) (I7): permite la categorización de la competencia descrita según tres grupos: *conocimientos*, *habilidades* y *capacidades*.
- *Niveles de cualificación* (I8): son usados para clasificar competencias con base en un marco de cualificación formal como el “*European Qualifications Framework*” (EQF).
- *Relaciones entre descripciones* (I9): permiten la formalización de relaciones entre competencias, por ejemplo, entre una competencia general y una específica. Algunos esquemas hacen uso del “*Simple Organization System*” (SKOS) [35] o de esquemas externos como el SRCM.
- *Referencias a literatura externa* (I10): son referencias bibliográficas a literatura relacionada con la competencia descrita.

De la Tabla B.1, se extrajo una taxonomía de fortalezas (S_i) y debilidades (W_i). Éstas son las siguientes:

- | | |
|---|--|
| <ul style="list-style-type: none"> • S_1: permite el uso de esquemas educativos formales. Así, se puede utilizar información externa para mejorar la comprensión de la competencia. | <ul style="list-style-type: none"> • W_1: sus descripciones dependen del lenguaje natural. Esto puede ocasionar errores tipográficos y en dificultades para crear conceptos comunes. |
| <ul style="list-style-type: none"> • S_2: facilita la creación de extensiones, definiendo elementos de datos y tipos de datos especiales para este propósito. | <ul style="list-style-type: none"> • W_2: no define vocabularios controlados para elementos específicos. Esto puede ocasionar errores tipográficos. |
| <ul style="list-style-type: none"> • S_3: cuenta con esquemas externos para formalizar relaciones entre competencias (por ejemplo, SRCM y MCFS). Esto facilita la reutilización de descripciones de competencias. | <ul style="list-style-type: none"> • W_3: las relaciones entre competencias se especifican en las descripciones de competencias. Por lo tanto, su reutilización se dificulta. |
| <ul style="list-style-type: none"> • S_4: es un estándar aceptado. | |

- S_5 : define elementos específicos (por ejemplo, el verbo de acción y el marco de cualificación) como una alternativa al lenguaje natural. Esto facilita la creación de conceptos comunes de competencias.
- S_6 : usa elementos de datos de LOM, estableciendo enlaces lógicos entre competencias y OA.

La Tabla B.2 analiza los esquemas desde el punto de vista de sus fortalezas y debilidades. En ésta, se observa que IEEE RCD es el esquema con más fortalezas y menos debilidades. Es un estándar aceptado, lo cual es apropiado para la proliferación, intercambio y reutilización de descripciones. Además, puede ser extendido sin afectar su estructura y cuenta con SRCM para describir relaciones entre competencias a través de mapas de competencias manteniendo la dependencia entre entidades y facilitando así la reutilización de descripciones. Al igual que IEEE RCD, RDCEO depende del lenguaje natural, lo cual puede dificultar la construcción de conceptos comunes debido a la variabilidad sintáctica y léxica entre las personas. Al respecto, eCOTOOL, MCOS y LOD dan un paso adelante en la definición de elementos específicos (por ejemplo, un verbo de acción, categoría *KSC* y niveles de cualificación) como una alternativa al lenguaje natural.

Considerando estos resultados, el presente trabajo provee como contribución una *extensión de IEEE RCD*, la cual suma las fortalezas de los esquemas estudiados y aborda sus dificultades añadiendo nuevos elementos de datos, vocabularios y tipos de datos para la descripción de competencias, algunos definidos como resultado de una revisión de literatura asociada y otros a partir de los resultados de un proceso de evaluación llevado a cabo por expertos y usuarios con base en características de la CM.

Por otra parte, considerando la posibilidad de crear relaciones entre competencias descritas por IEEE RCD, se adopta el SRCM como el esquema para el modelado de mapas de competencias. Al respecto, SRCM es el esquema más expresivo hallado en la literatura.

B. Análisis de esquemas de metadatos del ámbito de las competencias

	IEEE RCD	RDCEO	eCOTOOL	MCOS	LOD
I1	Si, con un título legible por humanos e identificación conforme a una URI	Si, con un título legible por humanos e identificación conforme a una URI	Si. Identificación conforme a una URI. No define un título	Si. Los elementos de título e Identificación son adaptados de LOM	Si. Título legible por humanos e identificación conforme a una URI
I2	Si, en lenguaje natural	Si, en lenguaje natural	Si, en lenguaje natural	Si, en lenguaje natural. Esta información es adaptada de LOM	Si, en lenguaje natural
I3	Si, son asertos en lenguaje natural tomados de una fuente formal.	Si, son asertos en lenguaje natural tomados de una fuente formal	Si, son conceptos enmarcados por la competencia, no necesariamente tomados de una fuente formal	No	No
I4	Si, permitiendo extensiones. El estándar no obliga al uso de un esquema o elementos determinados, aunque sugiere el uso de LOM	Si, permitiendo extensiones. Es posible integrar elementos nuevos o de otros esquemas de descripción	No	No	No
I5	No	No	Si, definiendo elementos que identifican al autor de la descripción	No	No
I6	No	No	Si, por medio de un verbo de acción y una expresión complementaria en lenguaje natural	No	No
I7	No	No	Si, con base en la categorización KSC	No	No
I8	No	No	Si, de acuerdo al EQF	No	Si, de acuerdo al EQF
I9	Si, por medio del uso externo del SRCM	No	No	Si, por medio del uso externo del MCFS o SKOS	No
I10	No	No	No	Si, permitiendo la definición subelementos o enlaces a información de soporte	No

Tabla B.1. Resultados del análisis comparativo sobre la información descrita por los esquemas de metadatos. Fuente: propia.

Esquemas	S_i	W_i
IEEE RCD	S_1, S_2, S_3, S_4	W_1
RDCEO	S_1, S_2, S_3	W_1
eCOTOOL	S_4	W_1, W_2, W_3
COSM	S_5	W_1, W_2
LOD	S_5, S_6	W_1, W_2, W_3

Tabla B.2. Fortalezas y debilidades de los esquemas de metadatos. Fuente: propia.

Anexo C

Versión extendida de la evaluación de la extensión del IEEE RCD

Este anexo presenta, de manera extendida, la evaluación de la extensión del IEEE RCD a la que se hace referencia en la Sección 3.4.

C.1 Características de calidad evaluadas

Teniendo en cuenta que las características de calidad de los metadatos han sido definidas en principio para evaluar repositorios de metadatos (en búsqueda de problemas ocasionados durante las tareas de marcación), fue necesario determinar las únicas que, después de una interpretación, pudiesen ser aplicadas a los esquemas de metadatos. La idea de hacerlo de esta forma nace de la necesidad de reducir las posibilidades de cometer errores durante la descripción de competencias por parte de las personas.

Como resultado se seleccionaron de [55] las siguientes características de calidad: *completitud* (los registros de metadatos describen completamente los ítems), *conformidad con las expectativas* (los usuarios encuentran en los esquemas lo que esperan encontrar) y *accesibilidad* (el esquema es entendible por los usuarios). Otras características como *procedencia* (confiabilidad de la entidad que realiza la marcación de los ítems), *consistencia lógica y coherencia* (heterogeneidad de registros de metadatos, colecciones de metadatos federadas) y *precisión* (edición de alta calidad durante la marcación) no fueron consideradas porque su evaluación sólo tiene sentido sobre repositorios de metadatos.

Después de interpretar las características seleccionadas, se establecieron un grupo de preguntas guía que fueron útiles para diseñar los instrumentos de evaluación y, por lo tanto, para verificar si el esquema propuesto cumple las características de calidad. Esta información se resume en la Tabla C.1.

C.2 Evaluadores e instrumentos de evaluación

Los siguientes perfiles fueron definidos para seleccionar los evaluadores del esquema de metadatos:

- Experto: docente o no, con títulos o conocimientos en pedagogía y experiencia en tecnologías de aprendizaje en línea, Objetos de Aprendizaje o similares.
- Usuario: docente con experiencia en educación primaria, secundaria o superior.

En total, participaron once evaluadores con perfil experto y ocho con perfil usuario.

Características de calidad	Interpretación	Preguntas guía
Compleitud	Los elementos de datos describen completamente las competencias	P1. ¿Los elementos de datos son suficientes para describir competencias? P2. ¿Qué elementos de datos son innecesarios?
Accesibilidad	Los elementos de datos son apropiados para la comunidad de usuarios	P3. ¿Los elementos de datos son confusos?
Conformidad con las expectativas	El esquema de metadatos propuesto funciona para lo que fue creado	P4. ¿El esquema cumple las expectativas? P5. ¿Los elementos de datos son apropiados para cumplir funciones específicas?
	Los vocabularios propuestos son adaptados a las características del usuario y a la forma en cómo se comprenden las competencias	P6. ¿Es apropiado elegir una opción de entre un grupo de éstas para asignar un valor a un determinado elemento de datos? P7. ¿Los vocabularios propuestos son apropiados para los elementos de datos?

Tabla C.1 Características de calidad, interpretación y preguntas guía adaptadas a los esquemas de metadatos. Fuente: propia.

Por otra parte, se diseñaron y aplicaron dos instrumentos de evaluación web, uno para cada perfil de evaluación. El instrumento para el perfil experto¹⁸ está enfocado tanto en las funciones de descripción como las relacionadas con la web de los elementos (por ejemplo, URL y tipos de datos como el iso 10646). Mientras que el instrumento para el perfil usuario¹⁹ se centra únicamente en las funciones de descripción de los elementos. Sin embargo, los dos instrumentos involucran las siguientes partes: 1) contextualización; 2) selección de una competencia de entre un grupo predefinido de ellas; 3) aplicación del esquema para modelar la competencia seleccionada; 4) presentación de preguntas guía (ver Tabla C.1).

¹⁸ El instrumento de evaluación para expertos puede ser accedido a través del enlace <https://goo.gl/gvFp1x>. Los resultados pueden ser consultados en:

https://mega.nz/#!rnAhyIKI!1h2ee3iYDci3gdwjl_MdYtNMszeXpztJXBopgzZtwo

¹⁹ El instrumento de evaluación para usuarios puede ser accedido a través del enlace <https://goo.gl/X44i2w>. Los resultados pueden ser descargados de: <https://mega.nz/#!zjB3RCwJ!rV70qJP50TkaZsIsANnORU7fB-cfUamz7Fv9tdkmjgg>

1. Primero, se presenta una introducción que contextualiza al evaluador acerca de aspectos relacionados con la investigación. Se explica el objetivo del esquema de metadatos, incluyendo la importancia de crear y compartir descripciones de competencias.
2. Segundo, se presenta un conjunto de competencias pre-definidas (en áreas como matemáticas, biología, tecnología, entre otras). Se solicita al evaluador que selecciona la opción de su preferencia.
3. Teniendo en cuenta la competencia seleccionada, se propone al evaluador la construcción de descripciones de acuerdo con los elementos de datos del esquema. Cada elemento es explicado con detalle, facilitando la buena aplicación de los instrumentos.
4. Finalmente, se presentan una serie de preguntas basadas en las preguntas guía de la Tabla C.1. También, se solicita la explicación de algunas de las respuestas.

C.3 Resultados

Después de la aplicación de los instrumentos de evaluación, la información coleccionada fue organizada de acuerdo a las características de calidad evaluadas. Los cambios y modificaciones sugeridos por los evaluadores ya se reflejan en el esquema propuesto.

C.3.1 Completitud

Con respecto a la pregunta guía P1, algunos expertos sugirieron la inclusión de aspectos relacionados con el currículo escolar, específicamente, el grado escolar. Por lo tanto, el elemento de datos `grade` fue incluido en el esquema de metadatos. Debido a que los grados escolares pueden variar entre sistemas educativos, no fue posible definir un vocabulario. Entonces, su valor se especifica de forma libre con una cadena de caracteres.

La Figura C.1a presenta los resultados para la pregunta guía P2. Aproximadamente el 25% de los evaluadores (expertos y usuarios) consideran que `typical_learning_time`, `KSC_category` y `language` son innecesarios. Sin embargo, como tales valores son bajos y, además, han sido definidos en otros esquemas, los elementos no han sido modificados. Específicamente, en el caso de `language`, aunque un grupo de evaluadores consideran que no es relevante especificar el lenguaje de la descripción, para la mayoría sí lo es, teniendo en cuenta

que actualmente éste contribuye en el posicionamiento de las descripciones en la web, por ejemplo, para realizar búsquedas.

Dados los porcentajes presentados, se concluye que el esquema de metadatos propuesto cumple la característica de completitud.

C.3.2 Accesibilidad

La Figura C.1b presenta los resultados para la pregunta guía P3. Para el 25% de los evaluadores y el 18% de expertos consideraron que *level_attributions*, *scheme* y *level* son confusos. *KSC_category* resulta confuso para el 12% de usuarios y el 9% de expertos. Adicionalmente, *topic*, *intended_user_role* y *typical_age* son confusos para el 12% de los expertos. Con el propósito de mejorar la propuesta, se han redefinido las explicaciones de estos elementos añadiendo información útil para construir una propuesta claramente comprensible. Con esto se presume una mejora del esquema en cuanto a accesibilidad.

Dados los porcentajes presentados en la evaluación, se concluye que el esquema propuesto cumple la característica de accesibilidad. Sin embargo, se espera a futuro realizar una nueva iteración de la evaluación enfocada en validar si las modificaciones realizadas mejoran los resultados.

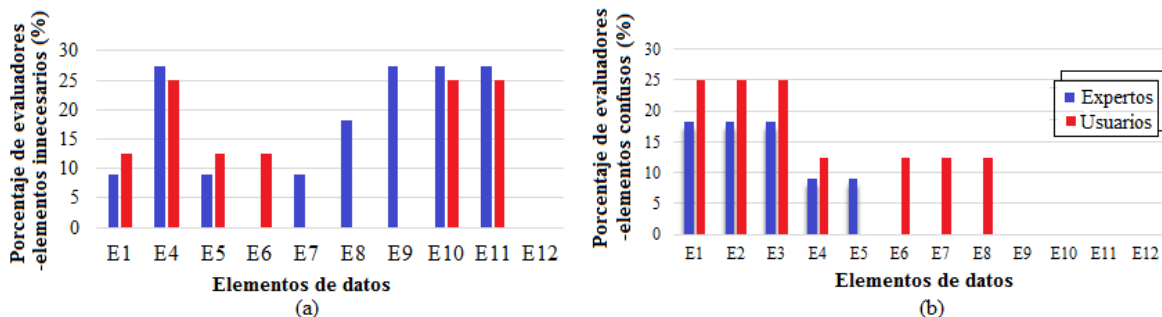


Figura C.1 Resultados para las preguntas guía: (a) P2, basados en el porcentaje de evaluadores que consideran que los elementos son innecesarios; y (b) P3, basados en el porcentaje de evaluadores que consideran que los elementos son confusos. Fuente: propia.

C.3.3 Conformidad con las expectativas

La Figura C.2a muestra los resultados obtenidos para la pregunta P4. En promedio, los evaluadores consideran que el esquema cumple sus expectativas.

Los resultados de la pregunta P5 se muestran en la Figura C.2b. En general, todos los elementos fueron considerados apropiados para cumplir sus funciones específicas. Específicamente, *typical_learning_time* fue considerado inapropiado por el 12% de usuarios, y *language* fue considerado inapropiado para el 9% de expertos y

C. Versión extendida de la evaluación de la extensión del IEEE RCD

el 25 % de usuarios. Dado que estos valores son bajos, los elementos mencionados no fueron modificados.

La Figura C.2c muestra los resultados de la pregunta guía P6. En general, la mayoría de evaluadores están de acuerdo con el uso de vocabularios para definir los valores de los elementos de datos a los que están asociados. Específicamente, el 12% de los usuarios no estuvieron de acuerdo con el uso de vocabularios para los elementos `action_verb`, `typical_age`, `difficulty`. Además, el 9% de los expertos no estuvieron de acuerdo con el uso de vocabularios para `difficulty` y `language`. De acuerdo con los comentarios de los evaluadores, los valores para `action_verb` podrían ser asignados por medio de texto libre. Debido a que la mayoría de evaluadores están de acuerdo con usar vocabularios para `action_verb`, en el esquema de metadatos se incluyó el elemento `other` por medio del cual el usuario puede definir un verbo no incluido en el vocabulario originalmente. Debido a que los evaluadores no hicieron sugerencia acerca de los demás elementos, no se establecieron alternativas al uso de vocabularios para ellos.

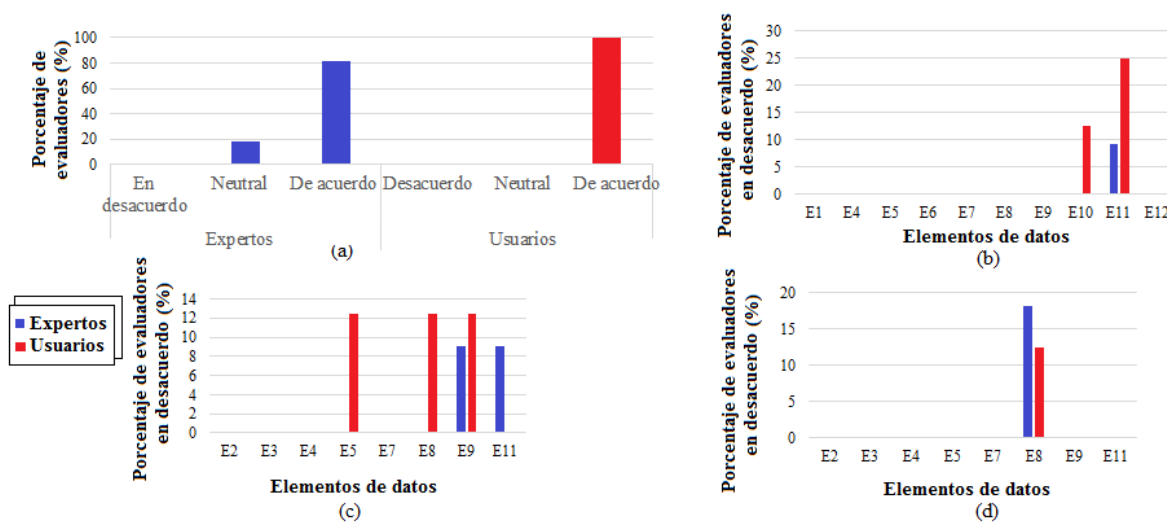


Figura C.2 Resultados para las preguntas guía: (a) P4, basados en el porcentaje de evaluadores que están de acuerdo con que el esquema cumple sus expectativas; (b) P5, de acuerdo con que los elementos cumplen sus funciones; (c) P6, basados en el porcentaje de evaluadores que no están de acuerdo con que los valores deben ser asignados por medio de vocabularios; y (d) P7, de acuerdo con el porcentaje de evaluadores que no están de acuerdo con los vocabularios propuestos. Fuente: propia.

Los resultados de la pregunta P7 se presentan en la Figura C.2d. En general, la mayoría de evaluadores están de acuerdo con los vocabularios definidos en el esquema. Sin embargo, el 18% de los expertos y el 12% de los usuarios no están de acuerdo con el vocabulario de `typical_age`. Dado que inicialmente se estableció un vocabulario compuesto por rangos de edades, se decidió modificar esto definiendo un

valor entero de forma manual. Esta decisión se basa en que como los rangos inicialmente definidos eran pequeños (de dos años por rango) y que la posibilidad de ocurrencia de errores tipográficos al escribir números de máximo dos cifras puede ser baja, introduciendo un valor entero se presume que se puede satisfacer a todos los usuarios en cuanto a la definición de una edad típica para el aprendizaje de la competencia.

Dados los porcentajes presentados, se concluyó que el esquema propuesto cumple la característica de conformidad con las expectativas. Sin embargo, se espera a futuro desarrollar otra iteración de la evaluación para validar si las modificaciones realizadas mejoran los resultados alcanzados.

En resumen, la evaluación del esquema de metadatos para competencias permitió recoger aportes para la mejora del mismo. En concreto: 1) los expertos sugirieron la inclusión del grado escolar en el esquema, lo cual se hizo a través de la definición del elemento `grade`; los expertos encontraron confusa la explicación de los elementos `topic`, `intended_user_role` y `typical_age`, por lo cual se realizó una redefinición de éstos; 3) los expertos encontraron no adecuado expresar el valor de `typical_age` como un rango de valores enteros. En consecuencia, se permitió la expresión de tal valor a través de un entero con texto libre; 4) entre los comentarios de los evaluadores (expertos y usuarios) se encontró que prefieren introducir en texto libre los valores de `action_verb`, por lo cual, se incluyó tal posibilidad adicionalmente a la definición por medio del vocabulario.

Anexo D

Versión extendida del estudio de ontologías existentes del dominio del VoD y las competencias

El presente anexo muestra de forma extendida el estudio de ontologías existentes al cual se hace referencia en el estado del arte (Sección 2.2.4), llevado a cabo para determinar las ontologías a reutilizar en la presente investigación. El estudio constó de dos fases así: *descubrimiento de ontologías* y *evaluación de ontologías existentes*; las cuales se explica a continuación.

D.1 Fase 1: descubrimiento de ontologías

Inicialmente se definió un grupo de palabras clave para realizar la búsqueda en buscadores convencionales y repositorios de ontologías. Una lista de éstas, organizada alfabéticamente se presenta a continuación, las cuales fueron combinadas con palabras relacionadas con la noción de ontología como *ontology*, *classification*, *taxonomy* y *thesaurus*:

- | | | |
|--------------------------|-----------------------------|---------------------|
| 1. Competency | 6. Learning Object Metadata | 11. TVA |
| 2. Competency map | 7. LOM | 12. Video on Demand |
| 3. Educational framework | 8. Television | 13. VoD |
| 4. Learning framework | 9. TV-Anytime | |
| 5. Learning Object | 10. TV | |

Como resultado de la búsqueda, se realizó una preselección de ontologías existentes, las cuales se presentan en la Tabla D.1 junto a una breve descripción de cada una.

D.2 Fase 2: evaluación de ontologías existentes

Con el propósito de evaluar las ontologías preseleccionadas, se tuvieron en cuenta las siguientes dimensiones tomadas de [78]:

1. *Evaluación de contenido*: evaluación de la información contenida en la ontología con respecto a su relevancia en el dominio de interés de la aplicación.
2. *Evaluación de la representación de conocimiento*: es la evaluación de la representación ontológica del contenido con respecto a la calidad del modelado. Como herramientas de evaluación de esta dimensión se usan razonadores como *Hermit*, *Pellet* y *jcel*.

Ontología	Lugar de publicación	Descripción	URL
"LOM-OWL ontology"	"Semantic Learning Objects Repository" (SLOR)	Define registros de OA basados en LOM	http://slor.sourceforge.net/ontology/lom.owl
"Learning Object Metadata Ontology"	"Linked Open Vocabularies" (LOV)	Ontología y vocabulario para interconectar metadatos educativos basados en LOM dentro del área de los Datos Abiertos Enlazados ²⁰ (DAE)	http://lov.okfn.org/dataset/lov/vocabs/lom
"RDF Schemas of LOM Metadata"	"Knowledge Management Research Group" (KMR)	Marco de esquemas RDF que constituyen un "binding" para el esquema LOM	http://kmr.nada.kth.se/static/ims/metad_ata.html
"Ontology for Media Resources"	"World Wide Web Consortium" (W3C)	Recomendación W3C. Ontología que provee un vocabulario para enlazar diferentes descripciones de recursos multimedia, junto a un conjunto de propiedades descriptivas de ellos	https://www.w3.org/TR/mediaont-10/
"Content Ontology for the TVAnytime Content CS"	Rhizomik	Ontología para clasificar programas de televisión de acuerdo a su género	http://rhizomik.net/html/ontologies/mpeg7ontos/
"Learning Object Context Ontologies" (The LOCO framework)	jelenajovanovic.net	Marco de ontologías que ofrece una descripción formal del contexto de los OA y sus principales aspectos: actividades de aprendizaje, roles de usuario,	http://jelenajovanovic.net/LOCO-Analyst/loco.html

²⁰ Parte de la web semántica resultado de la integración a gran escala de, y razonamiento sobre, los datos en la web [138].

D. Versión extendida del estudio de ontologías existentes del dominio del VoD y las competencias

		y el objetivo de aprendizaje. En el grupo de ontologías se encuentra una para modelar competencias	
--	--	--	--

Tabla D.1. Ontologías pre-seleccionadas para su reutilización. Fuente: propia.

3. *Evaluación técnica*: evaluación de la usabilidad de la ontología con respecto al contexto técnico de la aplicación.
4. *Evaluación de la aplicación*: se refiere a la usabilidad de la ontología requerida por la aplicación en la cual la ontología final será embebida y las tareas que serían cumplidas por la ontología.
5. *Evaluación de disponibilidad*: se refiere a la disponibilidad de restricciones impuestas por una ontología y su institución de proveniencia.

Después del análisis de las ontologías pre-seleccionadas, se identificó una taxonomía de sus fortalezas (X_i) y debilidades (Y_i), las cuales se muestran a continuación:

- X_1 : los razonadores no mostraron problemas para el razonamiento sobre la ontología
- X_2 : ontología que puede ser gestionada por *Protégé*
- X_3 : ontología soportada por APIS Java de Apache Jena, por lo cual puede ser implementada en aplicaciones
- X_4 : ontología pequeña, por lo que puede ser fácilmente integrables a aplicaciones o con otras ontologías
- X_5 : ontología indexada de acuerdo con esquemas de metadatos, lo cual facilita su reutilización
- Y_1 : los razonadores evidenciaron problemas para razonar sobre la ontología, lo cual indica la presencia de inconsistencias
- Y_2 : ontología no indexada de acuerdo con esquemas de metadatos, lo cual dificulta su reutilización
- Y_3 : no se puede realizar un seguimiento a las actualizaciones de la ontología o ésta no ha sido actualizada
- Y_4 : ontología con un gran número de axiomas, lo cual podría dificultar su integración en aplicaciones o con otras ontologías

- X_6 : la ontología está bien comentariada, lo cual facilita su reutilización
- X_7 : la ontología respeta la estructura de un grafo, haciéndola correcta sintáctica y semánticamente
- X_8 : la ontología es de acceso libre
- X_9 : la ontología ha sido actualizada

La Tabla D.2 resume la evaluación de las ontologías preseleccionadas de acuerdo a las cinco dimensiones de evaluación y la taxonomía de fortalezas y debilidades. De ella se puede concluir que existen ontologías útiles para el desarrollo del presente trabajo, que deben pasar por un proceso de adaptación e integración para proveer las condiciones que imponen los esquemas de metadatos expuestos en las anteriores secciones.

A continuación, se listan las ontologías que son reutilizadas y la respectiva justificación de forma resumida:

- “*Learning Object Metadata Ontology*”: modela todos los componentes de LOM. Es de fácil comprensión, lo cual facilita su integración con otras ontologías y la inclusión de nuevas clases y propiedades.
- “*Ontology for Media Resource*”: ontología que permite adaptar los conceptos representados a los elementos de *TV-Anytime*. En su documentación se encuentra explícitamente cómo realizar el mapeo respectivo usando propiedades OWL. Además, ofrece una gran cantidad de comentarios que facilitan su comprensión.
- “*Learning Object Context Ontologies*” (“*The LOCO framework*”): ofrece un marco modular de ontologías pequeñas y fácilmente comprensibles en torno a los OA. De entre el grupo de ontologías que define, hay dos que serán reutilizadas:
 - “*Competency Model Ontology*”: ontología útil para modelar las competencias.
 - “*Learning Context Ontology*”: ontología que integra las competencias con otros aspectos del contexto de los OA.

Tal como se adelantó en la introducción del trabajo (Sección 1.1), en la presente investigación se busca proveer una ontología para la formalización de relaciones entre competencias, contenidos de VoD y mapas de competencias. Para esto, las ontologías seleccionadas son integradas y usadas para mapear todos los elementos y relaciones que se definen en el *esquema de metadatos para la descripción de contenidos de VoD*

D. Versión extendida del estudio de ontologías existentes del dominio del VoD y las competencias

de acuerdo a las competencias educativas (Capítulo 3). Cabe señalar, que en los trabajos de la literatura revisados no fue posible hallar durante una ontología para modelar mapas de competencias. Entonces, fue necesario diseñar una nueva ontología a partir del esquema de metadatos SRCM, el cual fue seleccionado por su expresividad. La descripción completa de la ontología se presenta en el Capítulo 4.

Ontologías	Dimensiones de evaluación				
	Contenido	Representación de conocimiento	Técnica	Aplicación	Disponibilidad
“ <i>LOM-OWL ontology</i> ”	Formaliza conceptos de LOM. No formaliza las características pedagógicas de los OA, las cuales son importantes en este trabajo de acuerdo al esquema de metadatos propuesto	Y_1	X_2, X_3	Y_2, Y_4	X_8, Y_3
“ <i>Learning Object Metadata Ontology</i> ”	Basada completamente en el esquema LOM. Las características pedagógicas de los OA, están totalmente formalizadas. Refleja los principios de los DAE	X_1	X_2, X_3	X_4, X_5 (de acuerdo con LOM), X_6, X_7	X_8, X_9
“ <i>RDF Schemas of LOM Metadata</i> ”	Basada en LOM. Cada categoría de LOM está representada por una ontología. Todas se integran a través de una ontología base. De entre las ontologías, hay una que es del interés para el desarrollo de este trabajo: lom-educational; la cual representa las propiedades educativas de los OA. Todos los vocabularios que define LOM para algunos los elementos de la categoría <i>educational</i> están representados por instancias	X_1	X_2, X_3	X_4, X_5 (de acuerdo con LOM), X_7	X_8, Y_3
“ <i>Ontology for Media Resources</i> ”	Define conceptos genéricos de cualquier tipo de recurso multimedia que pueden ser mapeados en elementos de diferentes esquemas, incluyendo <i>TV-Anytime</i>	X_1	X_2, X_3	X_4, Y_4, X_6, X_7	X_8, Y_3
“ <i>Ontology for the TVAnytime Content CS</i> ”	Formaliza el esquema de clasificación por géneros de los programas de televisión. No formaliza la descripción de los programas, grupos de estos, ni los servicios fuente	X_1	X_2, X_3	X_4, X_5 (de acuerdo con <i>TV-Anytime</i>), X_7	X_8, Y_3

D. Versión extendida del estudio de ontologías existentes del dominio del VoD y las competencias

<p><i>“Learning Object Context Ontologies” (The LOCO framework)</i></p>	<p>Marco de ontologías propicias para representar los OA. Éstas se integran por medio de una ontología central.</p> <p>De entre las ontologías de se destacan tres debido a las características de este trabajo:</p> <ol style="list-style-type: none"> 1. <i>“ALOCOM Core Ontology”</i>, la cual modela OA y sus atributos básicos, así como sus relaciones. 2. <i>“Competency model ontology”</i> es propicia para modelar competencias. Define una subpropiedad de la propiedad <i>description</i> de <i>Dublin Core</i> para realizar descripciones en lenguaje natural. Además, utiliza el modelo SKOS [35] para asociar conceptos de competencias a su descripción. Esta ontología no considera la mayoría de propiedades definidas por IEEE RCD, ni ofrece un medio para formalizar relaciones entre competencias. 3. <i>“Learning Context Ontology”</i> funciona como integradora de <i>“ALOCOM Core Ontology”</i> y <i>“Competency model ontology”</i>. A través de ésta se establecen relaciones entre los OA y el contexto de aprendizaje, y entre los usuarios y las competencias. Sin embargo, no define relaciones entre los OA y las competencias. 	<p>X_1</p>	<p>X_2, X_3</p>	<p>X_4, X_5 (de acuerdo con el modelo <i>Abstract Learning Object Content Model</i>), X_6, X_7</p>	<p>X_8, Y_3</p>
---	--	-------------------------	------------------------------	--	------------------------------

Tabla D.2. Resultado de evaluación de las ontologías pre-seleccionadas. Fuente: propia

Anexo E

Definición formal del perfil de aplicación de TV-Anytime

Este anexo muestra el *perfil de aplicación de TV-Anytime* propuesto para la descripción de los contenidos de VoD; esto desde el punto de vista de los tipos de datos (siguiendo la nomenclatura de *TV-Anytime*) y la definición de los elementos de metadatos, en cuanto a su significado. Todo, de forma complementaria a lo presentado en la Sección 3.1. Cabe aclarar que las representaciones de los tipos de datos y los elementos mostradas a continuación son únicamente informativas, por lo cual pueden ser adaptadas a cualquier “*binding*”. Más información acerca de otros elementos de *TV-Anytime* puede ser consultada en [36]. Los aportes realizados a *TV-Anytime* se señalan en negrita.

Inicialmente, se abarca la descripción de los contenidos. De acuerdo a *TV-Anytime*, ésta información se define a través del tipo de datos `ProgramInformationType`, el cual define *TV-Anytime* como sigue (se mantiene el formato para la definición de tipos de datos simples y complejos de *TV-Anytime*, aclarando que no es un “*binding*”):

```
<complexType name="ProgramInformationType">
  <sequence>
    <element name="BasicDescription"
      type="tva:BasicContentDescriptionType"/>
  </sequence>
  <attribute name="programId" type="tva:CRIDType" use="required"/>
  <element name="CompRef" type="characterString- iso 10646"
    minOccurs="0" maxOccurs="unbounded"/> //Nuevo elemento para vincular
    el contenido con una competencia descrita en IEEE RCD
</complexType>
```

Se observa que dentro éste tipo de datos se abarca la identificación del contenido (`programId`), el nuevo elemento añadido (`CompRef`, indicado en negrita), el cual vincula al contenido con una o más descripciones de competencias hechas en IEEE RCD, y el tipo de datos complejo llamado `BasicContentDescriptionType`, el cual es definido por *TV-Anytime* para agrupar la información del contenido que se puede presentar en la GEP. Según el esquema, la asignación en texto libre de valores para los elementos de datos debe realizarse de acuerdo al iso 10646 [93].

El tipo de datos `BasicContentDescriptionType` se define como sigue:

```
<complexType name="BasicContentDescriptionType">
  <sequence>
    <element name="Title" type="mpeg7:TitleType" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="Synopsis" type="tva:SynopsisType" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="Keyword" type="tva:KeywordType" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="Language" type="mpeg7:ExtendedLanguageType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="ProductionLocation" type="mpeg7:regionCode"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="ReleaseInformation"
      type="tva:ReleaseInformationType" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="Duration" type="duration" minOccurs="0"/>
  </sequence>
</complexType>
```

Se observa que con éste se describe información del contenido como el título (`Title`), la sinopsis (`Synopsis`), las palabras clave (`Keyword`), el lenguaje del contenido (`Language`), la ubicación del contenido (`ProductionLocation`), la información de publicación (`ReleaseInformation`) y la duración (`Duration`).

Con respecto a la ubicación de los contenidos, *TV-Anytime* establece un tipo de datos llamado `ProgramLocationType`, el cual se define de la siguiente manera:

```
<complexType name="ProgramLocationType" abstract="true">
  <sequence>
    <element name="Program" type="tva:CRIDRefType"/>
    <element name="ProgramURL" type="anyURI" minOccurs="0"/>
  </sequence>
</complexType>
```

Éste incluye elementos para definir el identificador del contenido descrito (`Program`) y su URL (`ProgramURL`).

Por otra parte, ante la posibilidad de describir segmentos de los contenidos de VoD, *TV-Anytime* define el tipo de datos `SegmentInformationType` para tal fin, el cual se define de la siguiente forma:

```
<complexType name="SegmentInformationType">
  <sequence>
    <element name="ProgramRef" type="tva:CRIDRefType"
      minOccurs="0"/>
    <element name="Description"
      type="tva:BasicSegmentDescriptionType" minOccurs="0"/>
    <element name="SegmentLocator" type="tva:TVAMediaTimeType"
      minOccurs="0"/>
    <attribute name="segmentId" type="tva:TVAIDType"
      use="required"/>
  </sequence>
```

E. Definición formal del perfil de aplicación de TV-Anytime

```

    <element name="segmentCompRef" type="characterString- iso 10646"
    minOccurs="0" maxOccurs="unbounded"/> //Nuevo elemento usado
    para vincular segmentos y descripciones de competencias en IEEE
    RCD
  </sequence>
  <attribute name="segmentId" type="tva:TVAIDType" use="required"/>
</complexType>

```

Como puede observarse, éste tipo de datos define los elementos para referenciar el contenido al cual los segmentos pertenecen (*ProgramRef*), uno de los nuevos elementos (*segmentCompRef*, señalado en negrita) por medio del cual se establecen vínculos entre los segmentos de video y una o más descripciones de competencias construidas por medio del IEEE RCD, una descripción para ser mostrada en las GEP (*Description*, el cual se define por medio del tipo de datos *BasicSegmentDescriptionType*), y la ubicación de los segmentos en la línea de tiempo del contenido al cual pertenecen (*SegmentLocator*, el cual se define por medio del tipo de datos *TVAMediaTimeType*). Específicamente, *BasicSegmentDescriptionType* es definido por *TV-Anytime* de la siguiente forma:

```

<complexType name="BasicSegmentDescriptionType">
  <sequence>
    <element name="Title" type="mpeg7:TitleType" minOccurs="0"
    maxOccurs="unbounded"/>
    <element name="Synopsis" type="tva:SynopsisType" minOccurs="0"
    maxOccurs="unbounded"/>
    <element name="Keyword" type="tva:KeywordType" minOccurs="0"
    maxOccurs="unbounded"/>
  </sequence>
</complexType>

```

Éste elemento define información como el título (*Title*), la sinopsis (*Synopsis*) y las palabras clave (*Keyword*) de los segmentos.

Por su parte, el tipo de datos *TVAMediaTimeType* define la información usada para establecer la ubicación del segmento dentro de la línea de tiempo de un contenido (*MediaRelTimePoint*) y su duración (*MediaDuration*), de la siguiente forma:

```

<complexType name="TVAMediaTimeType">
  <sequence>
    <choice>
      <element name="MediaRelTimePoint"
      type="mpeg7:MediaRelTimePointType"/>
    </choice>
    <choice minOccurs="0">
      <element name="MediaDuration"
      type="mpeg7:mediaDurationType"/>
    </choice>
  </sequence>
</complexType>

```

La definición formal de cada tipo de datos y elementos se resume en la Tabla E.1, en la cual se detallan sus funciones a nivel descriptivo. Éstos tienen identificadores del tipo DT.n y E.n, respectivamente. Información adicional sobre el esquema *TV-Anytime* puede ser consultada en [36]. Las descripciones son tomadas, en principio, de *TV-Anytime*, y se les ha añadido una parte, indicada en negrita, que señala la intención o importancia de cada elemento en este trabajo.

Identificador	Elemento o tipo de datos	Descripción
DT.1	ProgramInformationType	Un tipo de datos complejo que describe un contenido. Éste se compone de los elementos E.1, E.2 y E.3
E.1	BasicDescription	Es la descripción de un contenido. Es usado para definir información para el despliegue de GEP de Sistemas de Recomendaciones (SR) o servicios de VoD. Se define por medio del tipo de datos BasicContentDescriptionType (ver identificador DT.2)
E.2	ProgramId	Es el " <i>Content Reference Identifier</i> " (CRID) para el contenido. Es un identificador único para referenciar el contenido descrito para extraer su información para el despliegue de las GEP y para referenciarlos desde los segmentos de video que le pertenecen, estableciendo así vínculos lógicos entre estas entidades
E.3	CompRef	Elemento que relaciona el contenido descrito con una competencia descrita por medio de la extensión de IEEE RCD (o de IEEE RCD original). Así, se establecen vínculos lógicos entre los contenidos y las competencias
DT.2	BasicContentDescriptionType	Tipo de datos complejo que define los elementos de descripción estándar de los programas. Éste engloba los elementos E.4, E.5, E.6, E.7, E.8, E.9 y E.10. De forma general, son útiles para construir las GEP de los Sistemas de Recomendaciones y los servicios de VoD
E.4	Title	Elemento obligatorio que define el título del contenido. Un contenido puede tener múltiples títulos, por ejemplo, en diferentes lenguajes

E. Definición formal del perfil de aplicación de TV-Anytime

E.5	Synopsis	Elemento obligatorio que define una descripción textual del contenido de VoD
E.6	Keyword	Elemento obligatorio que contiene una lista de palabras clave del contenido. Una <i>keyword</i> puede ser una sola palabra o una frase compuesta por múltiples palabras. Las palabras clave pueden ser usadas para establecer estrategias de recomendación basadas en similitudes sintácticas y semánticas
E.7	Language	Elemento obligatorio que describe el lenguaje hablado en el contenido. Puede haber más de un lenguaje hablado en el contenido. Este elemento puede ser usado para filtrar las recomendaciones de acuerdo al lenguaje del usuario
E.8	ProductionLocation	El país en el cual fue producido el contenido. Se define por medio de un código de región (ver clausula 5.6.4 de iso/iec 15938-5 [132] para una especificación detallada). Éste elemento puede ser usado para filtrar las recomendaciones de acuerdo al país del usuario
E.9	ReleaseInformation	Información sobre el lugar y fecha de publicación del contenido. Ésta información es útil para alimentar las GEP
E.10	Duration	Elemento obligatorio que indica la duración aproximada del contenido. Ésta información es útil para alimentar las GEP
DT.3	ProgramLocationType	Tipo de datos abstracto que representa la ubicación de un contenido simple
E.11	Program	Una referencia al CRID del contenido descrito
E.12	ProgramURL	Elemento que especifica la ubicación lógica de un contenido. Es útil para insertar los contenidos desde fuentes externas como Youtube en GEP de Sistemas de Recomendaciones o servicios de VoD
DT.4	SegmentInformationType	Define la información de un segmento de un contenido. Éste tipo de datos está

		constituido por los elementos de datos E.13, E.14, E.15, E.16, E.17
E.13	ProgramRef	Elemento obligatorio que referencia el contenido al que este segmento pertenece por medio de su identificador. Así, se establecen vínculos lógicos entre el segmento descrito y el contenido al cual pertenece
E.14	Description	Elemento obligatorio con la información básica del segmento descrito. Se define por medio del tipo de datos BasicSegmentDescriptionType (ver identificador DT.5). Este elemento es útil para generar las GEP
E.15	SegmentLocator	Elemento obligatorio, que permite ubicar el segmento dentro de un contenido de VoD en términos de tiempo de inicio y duración. Se define por medio del tipo de datos TVAMediaTimeType (ver identificador DT.6). Es útil para formular estrategias de recomendación enfocadas en los segmentos de video, para lo cual es importante identificar su ubicación dentro la reproducción de un contenido
E.16	segmentId	Identificador obligatorio único para el segmento descrito. Este elemento es útil para
E.17	SegmentCompRef	Elemento que relaciona el segmento descrito con una competencia descrita por medio de la extensión del IEEE RCD (o del IEEE RCD original). Corresponde al elemento identifier de RCD. Es útil para definir vínculos lógicos entre los segmentos y las competencias. Además, con el uso complementario de ProgramRef, se pueden definir vínculos indirectos entre los contenidos y las competencias, sirviendo los segmentos como eslabones dentro de tales vínculos
DT.5	BasicSegmentDescriptionType	Define la descripción básica de un segmento. Éste se compone de los elementos de datos E.18, E.19 y E.20. Éstos son usados para extraer

E. Definición formal del perfil de aplicación de TV-Anytime

		información para el despliegue de las GEP de Sistemas de Recomendaciones o servicios de VoD
E.18	Title	Elemento opcional que define un título para el segmento
E.19	Synopsis	Elemento opcional con una descripción breve del segmento
E.20	Keyword	Elemento obligatorio que define palabras clave sobre el segmento
DT.6	TVAMediaTimeType	Tipo de datos complejo usado para definir un punto en una línea de tiempo y una duración para el segmento. Éste se compone de los elementos de datos E.21 y E.22
E.21	MediaRelTimePoint	Elemento obligatorio para señalar un punto de tiempo usando fecha y hora gregoriana que corresponde al momento aproximado en que inicia el segmento durante la reproducción del contenido. Es importante para el establecimiento de estrategias de recomendación enfocadas en la sugerencia de segmentos de video, para lo cual es importante ubicar el segmento recomendado dentro de la reproducción de un contenido
E.22	MediaDuration	Elemento obligatorio usado para definir los días, oras, segundos y fragmentos de segundo que dura el segmento. Por defecto, se asume que el segmento dura hasta el final de la reproducción del contenido de VoD. Es útil para establecer estrategias de recomendación enfocadas en los segmentos de video, para lo cual es importante conocer la duración del segmento

Tabla E.1. Definición de elementos y tipos de datos para la descripción de contenidos de VoD de acuerdo a *TV-Anytime*. Fuente: propia.

Anexo F

Información complementaria de la extensión del IEEE RCD

Este anexo presenta información complementaria de la extensión del IEEE RCD para el modelado de competencias mostrado en la Sección 3.2. Se abarcan aspectos como los tipos de información integrados a dicha extensión y su definición formal, en cuanto a nomenclatura y descripción de los elementos.

F.1 Tipos de información integrados al IEEE RCD

Esta sección es complementaria a la Sección 3.2, en la cual se exponen once tipos de información que se integran al IEEE RCD para crear una nueva extensión de él.

1. *La categoría de las competencias (C1)*: es información útil para posicionar las competencias dentro de una de las categorías *conocimiento, habilidad y capacidad*. Ésta información se extrajo del esquema *eCOTOOL*, el cual denomina a este grupo de categorías como *KSC* (“*Knowledge*”, “*Skill*”, “*Competence*”). Una clasificación y asignación formal de una categoría *KSC* asegura que las competencias puedan ser comparadas entre diversos sistemas educativos. Por otra parte, de acuerdo con la *Organización para la Cooperación y el Desarrollo Económico* (OCDE), las competencias se relacionan con actitudes y valores. Al respecto, países como Colombia y México han definido competencias ciudadanas para mejorar la convivencia desde el entorno escolar [90]. La *categoría de las competencias* ha sido seleccionada en esta investigación ya que a partir de ella es posible enfocar estrategias de recomendación en el apoyo conocimientos, habilidades, capacidades, actitudes y valores en las que los estudiantes tengan necesidades. Además, esta información permite establecer vínculos lógicos entre competencias. También, tiene el suficiente significado para establecer asociaciones semánticas complejas al ser formalizada como una propiedad en una ontología (Sección A.3).

2. *El verbo de acción involucrado en el desarrollo de la competencia (C2)*: según el *Diccionario Oxford* de Inglés [48], un *verbo de acción* expresa acción física o mental. Por lo tanto, es útil para describir la acción inherente a las competencias, independientemente del contexto de uso. La definición del verbo de acción como un atributo de las competencias se extrajo del esquema eCOTOOL. Esta información ha sido considerada porque permite enfocar las estrategias de recomendación al apoyo de verbos en los que los estudiantes tengan dificultades. Además, permite establecer vínculos lógicos entre competencias. También, tiene el suficiente significado para establecer asociaciones semánticas complejas al ser formalizada como una propiedad en una ontología.
3. *El esquema de cualificación de la competencia (C3)*: esta información es usada para clasificar una competencia de acuerdo a esquemas formales internacionales de cualificación. Por ejemplo, eCOTOOL establece al “*European Qualifications Framework*” (EQF) [51] como el esquema para clasificar competencias en ocho niveles, abarcando desde el conocimiento más básico hasta la habilidad más compleja. Otro esquema que se halló en la literatura es la *Clasificación Internacional Normalizada de la Educación* (CINE) [91], la cual fue definida por UNESCO para clasificar cualquier aspecto relacionado con la educación de acuerdo a nueve niveles, abarcando desde la educación preescolar hasta los doctorados. Esta información permite comparar y homologar competencias entre diferentes esquemas de cualificación. Además, posibilita la creación de estrategias de recomendación enfocadas en apoyar esquemas de cualificación específicos. También permite establecer vínculos lógicos entre competencias. Sin embargo, podría no tener el significado suficiente para establecer asociaciones semánticas al ser formalizada como una propiedad en una ontología. Ésta es una decisión que depende del contexto de la aplicación en la que se implemente la ontología.
4. *Rol de usuario al que interesa la competencia (C4)*: esta información es adaptada de LOM, e indica el rol de la persona a quien puede interesar la competencia. Puede ser utilizada en una estrategia de recomendación para filtrar los contenidos dependiendo del rol usuario, por ejemplo, si es un estudiante o si es un docente. Esto teniendo en cuenta que pueden existir contenidos dirigidos a roles específicos. También, permite establecer vínculos lógicos entre competencias. Sin embargo, podría no tener el significado suficiente para establecer asociaciones semánticas al ser formalizada como una propiedad en una ontología. Ésta es una decisión que depende del contexto de la aplicación en la que se implemente la ontología.

5. *La edad mínima del estudiante para la cual típicamente éste tiene la capacidad para desarrollar las competencias (C5)*: teniendo en cuenta las categorías de las competencias, éstas pueden involucrar tanto conocimientos básicos como capacidades complejas, lo cual implica una serie de facultades que se van desarrollando con la edad de las personas. Ésta información, adaptada de LOM, es tenida en cuenta en este trabajo porque, además de ser un atributo propicio para establecer vínculos entre competencias, puede ser usado en una estrategia de recomendación para filtrar las sugerencias de acuerdo a la edad del usuario. Sin embargo, podría no tener el significado suficiente para establecer asociaciones semánticas al ser formalizada como una propiedad en una ontología. Ésta es una decisión que depende del contexto de la aplicación en la que se implemente la ontología.
6. *Dificultad de aprendizaje de la competencia (C6)*: esta información es adaptada de LOM para indicar la dificultad inherente al aprendizaje de la competencia. Puede ser útil para definir una estrategia de recomendación que apoye diferentes niveles de dificultad, de acuerdo a una clase de aprendizaje gradual de éstas. También, es información útil que facilita el establecimiento de vínculos entre competencias. Sin embargo, podría no tener el significado suficiente para establecer asociaciones semánticas al ser formalizada como una propiedad en una ontología. Ésta es una decisión que depende del contexto de la aplicación en la que se implemente la ontología.
7. *Tiempo típico de aprendizaje de la competencia (C7)*: se refiere al tiempo en semanas usado normalmente en el aprendizaje de la competencia. Esta información es adaptada de LOM y puede ser útil, por ejemplo, para definir una estrategia de recomendación que siga un cronograma y períodos de estudio de las competencias establecidos en un currículo escolar. La posibilidad de definir esta información en semanas fue recomendada por algunos de los evaluadores en el marco de la evaluación mostrada en la Sección 3.4. El tiempo típico permite el establecimiento de vínculos lógicos entre competencias. Sin embargo, podría no tener el significado suficiente para establecer asociaciones semánticas al ser formalizada como una propiedad en una ontología. Ésta es una decisión que depende del contexto de la aplicación en la que se implemente la ontología.
8. *Temas asociados a las competencias (C8)*: a partir de un análisis de las competencias definidas en países como Colombia [53], Argentina [133] y Perú [134], se estableció que las competencias pueden estar asociadas a temas como *El Teorema de Pitágoras*, *Las figuras geométricas*, *Los números reales*, etc. Esta información puede ser útil para enfocar las recomendaciones en las competencias

asociadas a temas específicos establecidos en el currículo escolar. Además, permite establecer vínculos lógicos entre competencias. También, tiene el suficiente significado para establecer asociaciones semánticas complejas al ser formalizada como una propiedad en una ontología en el caso de que hubiese un vocabulario controlado para los temas. En caso contrario, esta información puede ser usada para enfocar estrategias de recomendación hacia el fortalecimiento de los temas en los que se enmarcan las competencias en las que los estudiantes tienen necesidades educativas.

9. *Grado educativo en el que se aplica la competencia (C9)*: información que se refiere a grados escolares tales como *primero de primaria, segundo de primaria, décimo de secundaria*, etc. Ésta fue sugerida por el grupo de evaluadores como resultado de la evaluación mostrada en la Sección 3.4. Dado que la organización de los grados escolares depende del sistema educativo de cada país, esta información no puede ser usada para establecer equivalencias entre distintos sistemas educativos. Para esto, es conveniente usar información como el esquema de cualificación de la competencia. El grado educativo puede ser útil para enfocar las recomendaciones en las competencias asociadas a un grado específico del currículo escolar. Además, permite establecer vínculos lógicos entre competencias. También, tiene el suficiente significado para establecer asociaciones semánticas complejas al ser formalizada como una propiedad en una ontología, toda vez que sea posible definir un vocabulario.
10. *Palabras clave (C10)*: es información comúnmente usada por cualquier esquema de metadatos actual que busca posicionar las descripciones en la web. Esta puede ser usada para establecer estrategias de recomendación basadas en las similitudes sintácticas y/o semánticas entre descripciones de competencias, por ejemplo, para buscar contenidos vinculados a competencias similares a las que hacen parte de las necesidades de los estudiantes. Esta información permite establecer vínculos lógicos entre competencias. También, tiene el suficiente significado para establecer asociaciones semánticas complejas al ser formalizada como una propiedad en una ontología en caso de existir un vocabulario controlado para definir las palabras clave.
11. *Lenguaje de la descripción (C11)*: es información que define el lenguaje de la descripción de la competencia. Es usada en los sistemas web actuales para filtrar los contenidos de acuerdo al lenguaje de los usuarios. Por lo tanto, el lenguaje puede ser un factor importante para el filtrado de las competencias usadas en cualquier proceso de recomendación. Ésta permite el establecimiento de vínculos lógicos entre competencias. Sin embargo, podría no tener el significado suficiente

para establecer asociaciones semánticas al ser formalizada como una propiedad en una ontología. Ésta es una decisión que depende del contexto de la aplicación en la que se implemente la ontología.

F.2 Definición formal de la extensión del IEEE RCD

Esta sección presenta la integración de elementos, desde el punto de vista de los tipos de datos (según la nomenclatura del IEEE RCD) y la definición de los elementos en cuenta a su significado. Cabe aclarar que las representaciones de los tipos de datos y los elementos mostradas a continuación son únicamente informativas, por lo cual pueden ser adaptadas a cualquier “*binding*” (como el definido en el esquema RDCEO [52]). En negrita se indican los aportes realizados por este trabajo al IEEE RCD.

De acuerdo al estándar [37], las extensiones del IEEE RCD deben realizarse por medio del elemento `additional_metadata`, el cual a su vez hace parte del tipo de datos `metadata_type` de la siguiente forma:

```
metadata_type = record
(
  rcd_schema :
  rcd_schema_version :
  additional_metadata :
    extension_type, // el Nuevo tipo de datos
)
```

En este tipo de datos, se observa que se ha incluido como aporte un nuevo tipo de datos denominado `extension_type`, el cual sirve para incluir los nuevos elementos de datos de la Tabla 3.2 (Sección 3.2) en el elemento `additional_metadata`.

De esta forma, el nuevo tipo de datos `extensión_type` se ha definido de la siguiente forma:

```
extension_type = record
(
  level_attributions :
    level_attributions_type, // obligatorio_ocurre una o más veces
  KSC_category: // obligatorio_ocurre una vez
    state (knowledge, skill, competence, attitude, value),
  action_verb : // obligatorio_ocurre una o más veces
    state(valores definidos en la Tabla 3.3, other),
  other :
    characterstring (iso 10646)
  topic : // opcional_ocurre una o más veces
    bag of characterstring (iso 10646),
  grade : // opcional_ocurre una o más veces
    bag of characterstring (iso 10646),
  intended_end_user_role : // obligatorio_ocurre una o más veces
    state (student, teacher, manager, author),
```

```

typical_age : // obligatorio_ocurre una vez
  integer (iso 10646),
difficulty : // obligatorio_ocurre una vez
  state (very low, low, medium, high, very high),
typical_learning_time : // obligatorio_ocurre una vez
  integer (iso 10646),
language : // obligatorio_ocurre una vez
  state (valores definidos en la Tabla 3.3),
keyword : // obligatorio_ocurre una o más veces
  bag of characterstring (iso 10646),
)

```

Por otra parte, el tipo de datos `level_attributions_type` es un nuevo tipo de datos para el elemento `level_attributions`, en el cual se han definido dos nuevos elementos de la siguiente forma:

```

level_attributions_type = record
(
  scheme : // obligatorio_ocurre una vez
    state (isced-unesco, EQF, other),
  level : // obligatorio_ocurre una vez
    choice (state) of (
      isced-unesco :
        state (valores definidos en la Tabla 3.3),
      EQF :
        state (valores definidos en la Tabla 3.3),
      other :
        characterstring (iso 10646),
    )
)

```

En una aplicación del esquema, se debe seleccionar un valor para `scheme`, y luego, uno para `level`. Notar que los valores para `level` dependen de los valores para `scheme`. Por ejemplo, si en la aplicación se selecciona el valor *isced-unesco* para el elemento `scheme`, entonces los posibles valores para `level` corresponden a los niveles del esquema *isced-unesco*. En caso de seleccionar el valor *other* para el elemento `scheme`, entonces el usuario tiene la posibilidad de escribir un valor para `level` de forma libre de acuerdo al iso 10646 [93]. Todos los nuevos elementos de datos que se insertan al IEEE RCD se definen formalmente en la Tabla F.1, en la cual se detalla tanto sus funciones descriptivas como su aplicación dentro de posibles estrategias de recomendación.

Identificador	Elemento o tipo de datos	Descripción
DT.9	<code>extension_type</code>	Tipo de datos usado para describir la competencia con elementos de datos de extensión. Éste define los elementos de datos E.1, E.2, E.3, E.4, E.5, E.6, E.7, E.8, E.9, E.10, E.11
E1	<code>level_attributions</code>	Elemento obligatorio para clasificar la competencia descrita de acuerdo a uno o más esquemas de

F. Información complementaria de la extensión del IEEE RCD

		<p>cualificación. Se define por medio del tipo de datos <code>level_attributions_type</code> (ver identificador DT.10). Puede ser usado para enfocar las recomendaciones en el apoyo de competencias vinculadas a cierto nivel de un esquema de cualificaciones, tales como el k1 del EQF o el 1 del <i>isced-unesco</i></p>
E4	<code>KSC_category</code>	<p>Elemento obligatorio usado para definir la categoría KSC de la competencia descrita. Su vocabulario se presenta en la Tabla 3.3. Puede ser usado para enfocar las recomendaciones en las categorías en las que el estudiante tiene dificultades. Por ejemplo, si se identifican dificultades para desarrollar habilidades, entonces conviene sugerir contenidos vinculados a las competencias que hacen parte del tal categoría</p>
E5	<code>action_verb</code>	<p>Elemento obligatorio usado para definir uno o más verbos relacionados con la competencia descrita. Su vocabulario se presenta en la Tabla 3.3. Puede ser útil para dirigir las recomendaciones hacia el apoyo de cierto verbo en el que el estudiante tiene dificultades. Por ejemplo, si se identifican falencias en el verbo “Reconocer”, las sugerencias podrían ser contenidos vinculados a competencias cuyo verbo también es “Reconocer”</p>
E6	<code>Topic</code>	<p>Elemento opcional usado para definir uno o más temas relacionados con la competencia descrita. Su valor se define por medio de cadenas de caracteres de acuerdo al iso 10646 [93]. Puede ser útil para dirigir las recomendaciones hacia el apoyo de ciertos temas en los que hayan dificultades o que se ven durante un ciclo escolar</p>
E7	<code>intended_user_role</code>	<p>Elemento obligatorio usado a definir el rol de usuario relacionado competencia descrita. Su vocabulario se presenta en la Tabla 3.3. Teniendo en cuenta que pueden existir contenidos de VoD tanto para estudiantes como para docentes, conviene usar este elemento para filtrar las recomendaciones para uno u otro rol</p>
E8	<code>typical_age</code>	<p>Elemento obligatorio usado para indicar la edad a para la cual el estudiante tiene la capacidad para aprender la competencia. Su valor se define por medio de un entero basado en el iso 10646 [93]. Es útil para filtrar las recomendaciones dependiendo de la edad del estudiante. Por ejemplo, puede usarse para evitar recomendar un contenido asociado a una competencia que no puede ser entendida por el estudiante debido a que éste aún no tiene la capacidad de hacerlo por su edad</p>

E9	Difficulty	Elemento obligatorio que define la dificultad asociada al aprendizaje de la competencia. Su vocabulario se presenta en la Tabla 3.3. Este elemento puede ser usado para filtrar las recomendaciones de acuerdo con la dificultad de las competencias involucradas
E10	typical_learning_time	Elemento obligatorio usado para indicar el número de semanas estimadas para aprender la competencia descrita. Se define por medio de un entero de acuerdo con el iso 10646 [93]. Este elemento puede ser útil para enfocar las recomendaciones en el apoyo de competencias que se ajustan a cierto período de tiempo dentro del currículo escolar
E11	Language	Elemento obligatorio usado para indicar el lenguaje de la descripción. Su vocabulario es referenciado en la Tabla 3.3. Puede ser útil para enfocar las recomendaciones en el apoyo de competencias descritas en el lenguaje del estudiante. También es útil para gestionar los registros o repositorios de las descripciones de competencias
E12	Keywords	Elemento obligatorio usado para establecer un grupo de palabras relacionadas con la competencia descrita. Se define por medio de una bolsa de cadenas de caracteres basadas en el iso 10646 [93]. Es útil para establecer estrategias de recomendación enfocadas en similitudes sintácticas y semánticas
E13	grade	Elemento obligatorio útil para definir el grado escolar de la competencia, por ejemplo, <i>quinto de educación secundaria</i> . Se define por medio de una bolsa de cadenas de caracteres de acuerdo al iso 10646 [93]. Dado que su valor puede variar entre sistemas educativos, no puede ser usado para comparar competencias en diferentes sistemas educativos. En caso de comparación, se debe usar <code>level_attributions</code> . Este elemento es útil para enfocar las recomendaciones en el apoyo a competencias que hacen parte de determinado grado escolar
DT.10	level_attributions_type	Tipo de datos para describir el elemento <code>level_attributions</code>
E2	Scheme	Subelemento obligatorio de <code>level_attributions</code> usado para definir un esquema de cualificación. Su vocabulario se muestra en la Tabla 3.3. Es útil para enfocar las recomendaciones en el apoyo de cierto esquema de cualificaciones
E13	Level	Subelemento obligatorio de <code>level_attributions</code> usado para definir el nivel del esquema de cualificación seleccionado en <code>scheme</code> . Los vocabularios para

F. Información complementaria de la extensión del IEEE RCD

		<p><i>iscid-unesco</i> y <i>EQF</i> se presentan en la Tabla 3.3. Para <i>other</i>, el valor de <code>level</code> se introduce usando una cadena de caracteres con base en el iso 10646 [93]. Una sola instancia de <code>level</code> se define para cada instancia de <code>level_attributions</code>. Este elemento es útil para enfocar las recomendaciones en el apoyo de competencias asociadas a cierto nivel del esquema de calificaciones. Por ejemplo, se determinan falencias educativas en el nivel k5 del EQF, entonces conviene sugerir contenidos asociados a competencias en las que se requiera un nivel de conocimientos alto</p>
--	--	---

Tabla F.1. Definición de elementos y tipos de datos para la descripción de competencias.

Fuente: propia.

En síntesis, los elementos de datos y los vocabularios propuestos ofrecen una guía común y específica para describir aspectos educativos de las competencias, posibilitando la construcción de conceptos comunes entre las personas. De esta forma, se han abordado las debilidades del IEEE RCD y, de hecho, las debilidades de todos los esquemas para competencias asociadas a la variabilidad léxica del lenguaje natural, la cual puede dificultar la creación de conceptos comunes entre las personas; y la falta de atributos y vocabularios específicos, lo cual puede ocasionar errores tipográficos que afecten la calidad de los registros y repositorios de metadatos en el futuro. Además, el esquema ofrece todas las posibilidades para establecer vínculos lógicos entre las competencias y para definir diferentes estrategias de recomendación según se requiera.

Anexo G

Aspectos complementarios de la adaptación de “Learning Object Metadata Ontology”

El presente anexo muestra una serie de aspectos complementarios relacionados con la adaptación de “Learning Object Metadata Ontology” desarrollada en la Sección 4.1.1.

G.1 Representaciones intermedias después de la adaptación

La Tabla G.1 presenta la *tabla de instancias* después de la adaptación de “Learning Object Metadata Ontology”.

Nombre de la instancia	Atributos con valores conocidos	Valor de los atributos
Higher education	Context-value	Higher education
School	Context-value	School
Training	Context-value	Training
Other	Context-value	Other
Diagram	Type-value	Diagram
Exam	Type-value	Exam
Exercise	Type-value	Exercise
Experiment	Type-value	Experiment
Figure	Type-value	Figure
Graph	Type-value	Graph
Index	Type-value	Index
Lecture	Type-value	Lecture
Narrative	Type-value	Narrative
Text	Type-value	Text
Problem	Type-value	Problem
Statement	Type-value	Statement
Questionnaire	Type-value	Questionnaire
Self-Assessment	Type-value	Self-Assessment
Simulation	Type-value	Simulation
Slide	Type-value	Slide
Table	Type-value	Table

Active	Interactivity-value	Active
Expositive	Interactivity-value	Expositive
Mixed	Interactivity-value	Mixed
Undefined	Interactivity-value	Undefined
Author	User-value	Author
Learner	User-value	Learner
Manager	User-value	Manager
Teacher	User-value	Teacher
Código de dos letras del iso 639-1	Language-value	Código de dos letras del iso 639-1
Difficult	Difficulty-value	Difficult
Easy	Difficulty-value	Easy
Medium difficulty	Difficulty-value	Medium difficulty
Very difficult	Difficulty-value	Very difficult
Very easy	Difficulty-value	Very easy

Tabla G.1. Tabla de instancias para la adaptación de “*Learning Object Metadata Ontology*”. Fuente: propia.

G.2 Cotejamiento ontología-esquema LOM

La Tabla G.2 presenta el resultado de cotejar el esquema LOM y la ontología.

Elemento LOM del esquema	Elemento de la ontología	A qué concepto está vinculado el elemento	Tipo de elemento	Dominio	Rango
Context	educationalContext	Educational	Data Property	Educational	<ul style="list-style-type: none"> xsd:short {“higher education”^^xsd:string, “other”^^xsd:string, “school”^^xsd:string, “training”^^xsd:string} rdfs:Literal
Learning Resource Type	-	-	-	-	-
Interactivity Type	educationalInteractivityType	Educational	Data Property	Educational	<ul style="list-style-type: none"> {“active”^^xsd:string, “expositive”^^xsd:string, “mixed”^^xsd:string}
Intended End User Role	educationalIntendedUserRole	Educational	DataProperty	Educational	<ul style="list-style-type: none"> {“author”^^xsd:string, “learner”^^xsd:string, “manager”^^xsd:string,

G. Aspectos complementarios de la adaptación de “Learning Object Metadata Ontology”

					“teacher”^^xsd:string}
Typical Age Range	educationalTypicalAgeRange	Educational	Object Property	Educational	-
Typical Learning Time	educationalTypicalLearningTime	Educational-Duration	ObjectProperty	Educational	Duration
Language	educationalLanguage	Educational	Data Property	Learning Object	<ul style="list-style-type: none"> • {“ba”^^xsd:string, “ca”^^xsd:string, “es”^^xsd:string, “ga”^^xsd:string, “none”^^xsd:string }
Difficulty	educationalDifficulty	Educational	Data Property	Educational	<ul style="list-style-type: none"> • {“difficult”^^xsd:string, “easy”^^xsd:string, “medium”^^xsd:string, “very difficult”^^xsd:string, “very easy”^^xsd:string }

Tabla G.2. Relación entre LOM y “Learning Object Metadata Ontology”. Fuente: propia.

G.3 Modelo ontológico final

La Tabla G.3 muestra el modelo ontológico resultado de la adaptación de “Learning Object Metadata Ontology”.

Elemento de la ontología	Tipo de elemento	Instancias	Dominio	Rango	Propiedades inversas
LearningObject	Class	-	-	-	-
Educational	Class	-	-	-	-
Duration	Class	-	-	-	-
Context	Class	Higher education, School, Training, Other	-	-	-
ResourceType	Class	Diagram, Exam, Exercise, Experiment, Figure, Graph, Index, Lecture, Narrative Text, Problem Statement, Questionnaire, Self-Assessment, Simulation, Slide, Table, Video	-	-	-
InteractivityType	Class	Active, Expositive, Mixed, Undefined	-	-	-

UserRole	Class	Author, Learner, Manager, Teacher	-	-	-
Language	Class	Instancias acordes con la descripción de competencias: código de dos letras del iso 639-1	-	-	-
Difficulty	Class	Difficult, Easy, Medium difficulty, Very difficult, Very easy	-	-	-
UserAge	Class	-	-	-	-
hasEducational	Data Property	-	Duration	xsd:decimal	-
educational	Object Property	-	LearningObject	Educational	educational
educationalTypicalLearningTime	Object Property	-	Educational	Duration	TypicalLearningTime
ageValue	Data Property	-	Duration	xs:int	-
hasEducationalContext	Object Property	-	Educational	Context	isEducationalContextOf
hasEducationalResourceType	Object Property	-	Educational	ResourceType	isEducationalResourceTypeOf
hasEducationalInteractivityType	Object Property	-	Educational	InteractivityType	isEducationalInteractivityTypeOf
hasEducationalUser	Object Property	-	Educational	UserRole	isEducationalUserOf
hasEducationalLanguage	Object Property	-	Educational	Language	isEducationalLanguageOf
HasEducationalDifficulty	Object Property	-	Educational	Difficulty	isEducationalDifficultyOf
hasEducationalAge	Object Property	-	Educational	UserAge	isEducationalAgeOf
contextValue	Data Property	-	Context	xsd:string	-
resourceTypeValue	Data Property	-	ResourceType	xsd:string	-
interactivityTypeValue	Data Property	-	InteractivityType	xsd:string	-
userValue	Data Property	-	User	xsd:string	-
languageValue	Data Property	-	Language	xsd:string	-
difficultyValue	Data Property	-	Difficulty	xsd:string	-
userAgeValue	Data Property	-	Age	xsd:string	-

Tabla G.3. Modelo ontológico final para la adaptación de “Learning Object Metadata Ontology”. Fuente: propia.

Anexo H

Versión extendida de la adaptación de “*Ontology for Media Resources*”

El presente anexo muestra de forma extendida el proceso desarrollado para adaptar la ontología “*Ontology for Media Resources*” mostrado en la Sección 4.1.2. Todo este proceso se explica en las siguientes fases.

Fase B.1: identificar las RI de las ontologías existentes

Iniciando con la interpretación de la “*Ontology for Media Resources*”, se identificó su *diccionario de conceptos*, el cual se muestra en la Tabla H.1.

Nombre del concepto	Instancias	Atributos de instancias	Relaciones del concepto
Media Resource	-	Title Description Locator Duration	Has-fragment Has-contributor Is-member-of Has-track
Media Fragment	-	Fragment-name	Is-fragment-of
Track	-	Track-name	Is-a Is-track-of
Video Track	-	-	Is-a
Collection	-	Collection-name	Has-member
Agent	-	-	Has-contributed-to
Organisation	-	-	Is-a

Tabla H.1. Diccionario de conceptos de “*Ontology for Media Resources*”. Fuente: propia.

Con el propósito de establecer las relaciones entre conceptos, se identificaron las relaciones binarias existentes de acuerdo con las relaciones del *diccionario de conceptos* anterior, las cuales se presentan en la Tabla H.2.

Nombre de la relación	Concepto fuente	Concepto objetivo	Relación inversa
Has-fragment	Media Resource	Media Fragment	Is-fragment-of
Has-member	Collection	Media Resource	Is-member-of
Has-contributor	Media Resource	Agent	Has-contributed-to
Has-track	Media-resource	Track	Is-track-of
Is-a	Track	Media Fragment	-

Is-a	Video Track	Track	-
Is-a	Organisation	Agent	-

Tabla H.2. Relaciones binarias de “Ontology for Media Resources”. Fuente: propia.

Considerando el *diccionario de conceptos* (Tabla H.1) y la tabla de *relaciones binarias* anterior (Tabla H.2), se construyó la *red semántica* de la Figura H.1, la cual resume los conceptos, sus atributos (señalados por la relación *Has*) y relaciones entre ellos.

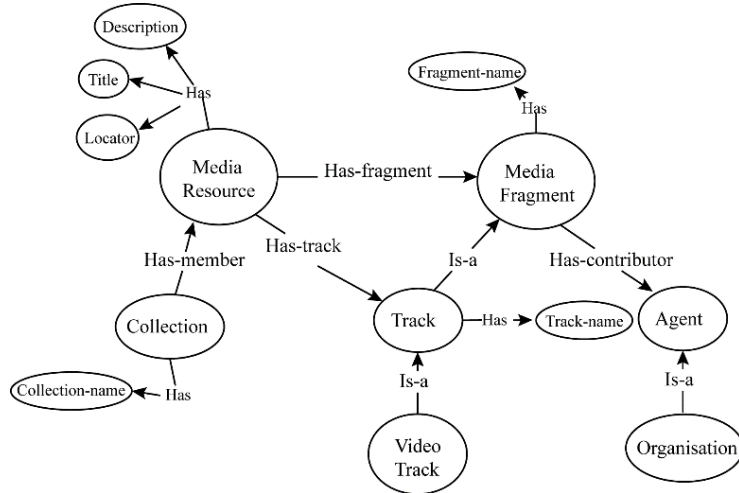


Figura H.1 Red semántica de “Ontology for Media Resources”. Fuente: propia.

La Tabla H.3 representa la *tabla de atributos* de las instancias del *diccionario de conceptos*. Ésta ayuda a entender los atributos a través de su tipo de valor, unidad de medida, rango de valores, valor por defecto y cardinalidad.

Nombre del atributo de instancias	Tipo de valor	Unidad de medida	Rango de valores	Valor por defecto	Cardinalidad
Locator	URI	NA	NA	NA	(1,1)
Title	ND	ND	ND	ND	(1,1)
Duration	Decimal	Hours, minutes, seconds	ND	ND	(1,1)
Fragment-name	ND	ND	ND	ND	(1,1)
Track-name	ND	ND	ND	ND	(1,1)
Collection-name	ND	ND	ND	ND	(1,1)
Description	ND	ND	ND	ND	(1,1)

Tabla H.3. Tabla de atributos de instancias de “Ontology for Media Resources”. Fuente: propia.

Fase B.2: cotejar ontología existente-esquema de metadatos

A pesar de la utilidad que ofrece “Ontology for Media Resources”, es importante evaluar si ésta es capaz de describir toda la información descrita por el perfil de aplicación de *TV-Anytime* para contenidos de VoD propuesto en la Sección 3.1, la cual

se lista a continuación (esta información fue explicada con mayor detalle en la Sección 3.1):

1. *Identificación (T1)*: información centrada en la identificación del contenido de VoD.
2. *Descripción (T2)*: información que expone de forma breve y general los puntos claves del contenido de VoD. Específicamente, la descripción involucra los siguientes aspectos:
 - El título del contenido.
 - La sinopsis.
 - Las palabras clave.
 - El lenguaje del contenido.
 - El lugar de la producción del contenido.
 - La fecha de publicación del contenido.
 - La duración del contenido.
 - Una referencia a la descripción de una o más competencias (hechas por medio de la extensión del IEEE RCD) desarrolladas en el contenido.
3. *Ubicación (T3)*: información sobre la ubicación del contenido con respecto a la plataforma proveedora del VoD. Específicamente, se busca con el esquema describir los siguientes aspectos con relación a la ubicación del contenido:
 - El identificador del contenido al cual la ubicación hace referencia.
 - La dirección URI del contenido.
4. *Segmentos (T4)*: por medio de esta información, los segmentos de video pueden ser considerados como entidades vinculadas, pero independientes, a los contenidos a los cuales pertenecen. Específicamente, este tipo de información involucra los siguientes aspectos:
 - El identificador del segmento.
 - El título del segmento.
 - La sinopsis del segmento.
 - Las palabras clave del segmento
 - Una referencia al contenido al cual pertenece el segmento.
 - La ubicación del segmento dentro de la línea de tiempo del contenido
 - La duración del segmento.
 - El identificador del segmento.

- Una referencia a la descripción de una o más competencias (construidas por medio de la extensión del IEEE RCD, propuesta en la siguiente sección) desarrolladas en el segmento.

Dado que los tipos de información están representados por los elementos de datos del perfil de aplicación de *TV-Anytime*, se analizó si éstos están cubiertos por los elementos ontológicos de “*Ontology for Media Resources*”. Además, se verificó si los dominios y rangos están debidamente definidos. Así, fue posible vislumbrar las falencias que posee y cómo abordarlas. Como resultado se obtuvo la Tabla H.4, la cual presenta los elementos del perfil de aplicación de *TV-Anytime*, qué elementos de “*Ontology for Media Resources*” los representan, a qué conceptos se vinculan, su tipo, dominio y rango.

Elemento TV-Anytime del esquema	Elemento de “ <i>Ontology for Media Resources</i> ”	A qué concepto está vinculado el elemento de la ontología	Tipo de elemento	Dominio	Rango
VoD service	Organisation	Organisation	Class	-	-
Name	-	-	-	-	-
ServiceId	-	-	-	-	-
serviceURL	-	-	-	-	-
Program	MediaResource	Media Resource	Class	-	-
Title	title	Media Resource	Data Property	MediaResource	-
Synopsis	description	Media Resource	Data Property	MediaResource	-
Duration	Duration	Media Resource	Data Property	MediaResource	xsd:decimal
Keyword	description	Media Resource	Data Property	MediaResource	-
	isMemberOf	Media Resource	Object Property	MediaResource	Collection
programId	-	-	-	-	-
Language	hasLanguage	Media Resource	Object Property	MediaResource	-
ProgramURL	locator	Media Resource	Data Property	MediaResource	Xsd:anyURI
	hasCaptioning	Media Resource	ObjectProperty	MediaResource	MediaFragmen t
Segment	MediaFragment	Media Fragment	Class	-	-
ProgramRef	isFragmentOf	Media Fragment	ObjectProperty	MediaFragment	MediaResourc e
Title	fragmentName	Media Fragment	Data Property	MediaFragment	-

H. Versión extendida de la adaptación de “Ontology for Media Resources”

segmentId	-	-	-	-	-
Synopsis	-	-	-	-	-
Keyword	description		Data Property	MediaResource	-
MediaDuration	duration	Media Resource	Data Property	MediaResource (no aplicable a imágenes)	xsd:decimal
MediaStartTimePoint	-	-	-	-	-

Tabla H.4. Relación entre elementos *TV-Anytime* del esquema y “*Ontology for Media Resources*”. Fuente: propia.

Se observa que varias propiedades de objetos (“*Object Properties*”) y propiedades de datos (“*Data Properties*”) no tienen definidos rangos. Se intuye que esto se debe a que la ontología ofrece características genéricas, ya sea para facilitar su expansión o refinamiento, o para su mapeo en diferentes esquemas de metadatos sin generar inconsistencias. En el caso específico de `:hasLanguage`, el rango no está vinculado a ninguna clase, y de hecho no hay una clase apropiada en la ontología para representar la información de los lenguajes. De la Figura H.1 se observa que no hay una relación entre *Media-resource* y *Video-track*, por lo cual no se pueden vincular los contenidos a sus segmentos de video. En Tabla H.4 también se observa que algunos elementos del perfil de aplicación de *TV-Anytime* propuesto no están representados en la ontología, por ejemplo, el lenguaje. En consecuencia, se concluye que “*Ontology for Media Resources*” no cubre toda la información que se busca describir para los contenidos de VoD (*T1*, *T2*, *T3* y *T4*). Las acciones realizadas para abordar estas falencias se muestran en la siguiente subfase de la metodología.

Fase B.3: determinar las RI adaptadas

Las falencias mencionadas en la fase anterior fueron abordadas realizando las siguientes acciones sobre la ontología:

1. Se definió un nuevo concepto que representa los lenguajes. Esto implica la creación de nuevas relaciones, instancias y atributos.
2. Para solucionar la falta de un rango de las *Object Property*, se establecieron los *conceptos objetivo* de las relaciones entre conceptos a través de una *tabla de relaciones binarias*, de acuerdo al esquema de metadatos. Así mismo, para el caso de las *Data Property*, se definieron los tipos de datos a través de la *tabla de atributos de instancias*, para posteriormente establecer sus *rangos* en la implementación.
3. Se crearon nuevas *relaciones* y *atributos* que representan los elementos del esquema que “*Ontology for Media Resources*” no cubre.

4. Se definió una relación llamada `:hasTrack`, la cual permite vincular los conceptos *Media-resource* y *Video-track*. Así, se establecen vínculos entre los contenidos y los segmentos de video.

Específicamente, se crearon nuevos elementos ontológicos aportados por el presente trabajo (señalados en negrita), los cuales se presentan en el diccionario de conceptos mostrado en la Tabla H.5.

Nombre del concepto	Instancias	Atributos de instancias	Relaciones del concepto
Media Resource	-	Title Description Locator Duration Media-resource-id Language Keyword	Has-fragment Has-contributor Is-member-of Has-track
Media Fragment	-	Fragment-name Fragment-id	Is-fragment-of
Track	-	Track-name	Is-a
Video Track	-	Time-point Video-track-duration	Is-a
Agent	-	-	
Organisation	-	Organisation-name Organisation-id Organisation-locator	Is-a Has-contributed-to
Language	Código de dos letras del iso 639-1	Language-value	Is-language-of

Tabla H.5. Diccionario de conceptos para la adaptación de “Ontology for Media Resources”. Fuente: propia.

La Tabla H.6 presenta todas las *relaciones binarias* existentes de acuerdo con las relaciones del anterior *diccionario de conceptos*.

Nombre de la relación	Concepto fuente	Concepto objetivo	Relación inversa
Has-fragment	Media Resource	Media Fragment	Is-fragment-of
Has-member	Collection	Media Resource	Is-member-of
Has-contributor	Media Resource	Agent	Has-contributed-to
Has-track	Media Resource	VideoTrack	Is-video-track-of
Is-a	Track	Media Fragment	-
Is-a	Video Track	Track	-
Is-a	Organisation	Agent	-
Has-language	Media Resource	Language	Is-language-of
Has-segment-language	Media Fragment	Language	Is-segment-language-of

H. Versión extendida de la adaptación de “Ontology for Media Resources”

Tabla H.6. Relaciones binarias para la adaptación de “*Ontology for Media Resources*”.
Fuente: propia.

La Figura H.2 presenta la red semántica identificada que señala las diferentes relaciones entre conceptos del nuevo modelo ontológico adaptado de “*Ontology for Media Resources*”. En gris se señalan los aportes a la ontología original en cuanto a conceptos y atributos (señalados por la relación *Has*) y en **negrita-cursiva** en cuanto a relaciones.

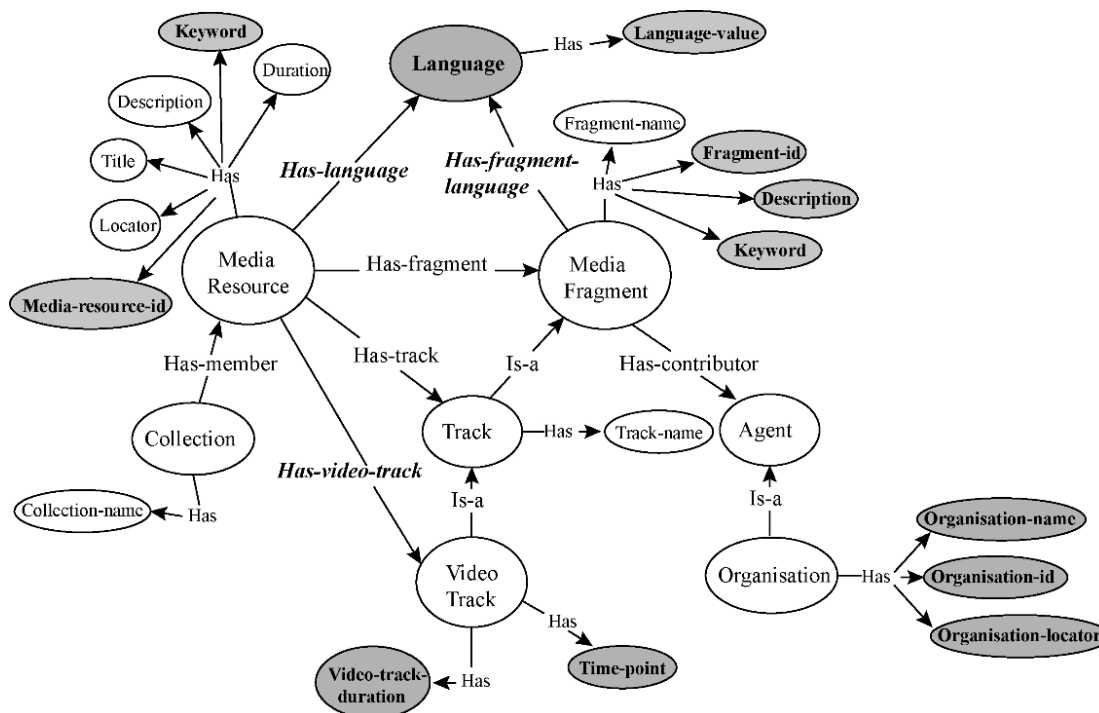


Figura H.2 Red semántica para la adaptación de “*Ontology for Media Resources*”. Fuente: propia.

La Tabla H.7, corresponde a la *tabla de atributos de instancias* del *diccionario de conceptos* final.

Nombre del atributo de instancias	Tipo de valor	Unidad de medida	Rango de valores	Valor por defecto	Cardinalidad
Title	String	-	-	-	(1,1)
Keyword	String	-	-	-	(1,n)
Fragment-name	String	-	-	-	(1,1)
Track-name	String	-	-	-	(1,1)
Collection-name	String	-	-	-	(1,1)
Description	String	-	-	-	(1,1)
Locator	URI	-	-	-	(1,1)
Media-resource-id	String	-	-	-	(1,1)
Fragment-id	String	-	-	-	(1,1)

Language-value	String	-	-	-	(1,1)
Collection-id	String	-	-	-	(1,1)
Number-of-items	Integer	Items	Z=[1, 2, 3, 4 ... ∞]	-	(1,1)
Collection-description	String	-	-	-	(1,1)
Time-point	Time	Hours, minutes, seconds	-	-	(1,1)
Video-track-duration	Time	Hours, minutes, seconds	-	-	(1,1)
Organisation-name	String	-	-	-	(1,1)
Organisation-id	String	-	-	-	(1,1)
Organisation-locator	URI	-	-	-	(1,1)
Duration-value	String	-	-	-	(1,1)

Tabla H.7. Tabla de atributos de instancias para la adaptación de “Ontology for Media Resources”. Fuente: propia.

La Tabla H.8, presenta la tabla de instancias que aparecen en el diccionario de conceptos adaptado (ver Tabla H.5). Por cada código del iso 639-1 hay una instancia, cada una con un atributo (*Language-value*) cuyo valor es el código correspondiente.

Fase B.4: determinar el modelo ontológico

De acuerdo con el *diccionario de conceptos, relaciones binarias, y tablas de atributos de instancias*, se establece la representación de la ontología adaptada desde el punto de vista de la implementación. Este se detalla en la Tabla H.9, en la cual se señalan en negrita los elementos aportados a la ontología original.

Nombre de la instancia	Atributos con valores conocidos	Valor de los atributos
Código de dos letras del iso 639-1	Language-value	Código de dos letras del iso 639-1

Tabla H.8. Tabla de instancias de la adaptación de “Ontology for Media Resources”. Fuente: propia.

Elemento de la ontología	Tipo de elemento	Instancias	Dominio	Rango	Propiedades inversas
MediaResource	Class	-	-	-	-
MediaFragment	Class	-	-	-	-
Track	Class (subclase de <i>MediaFragment</i>)	-	-	-	-
VideoTrack	Class (subclase de <i>Track</i>)	-	-	-	-

H. Versión extendida de la adaptación de “Ontology for Media Resources”

Collection	Class	-	-	-	-
Agent	Class	-	-	-	-
Organisation	Class	-	-	-	-
Language	Class	Código de dos letras del iso 639-1	-	-	-
title	Data Property	-	MediaResource	xsd:string	-
description	Data Property	-	MediaResource	xsd:string	-
locator	Data Property	-	MediaResource	xsd:anyURI	-
duration	Data Property	-	MediaResource	xsd:decimal	-
keyword	Data Property	-	MediaResource	xsd:string	-
mediaResourceId	Data Property	-	MediaResource	xsd:string	-
fragmentName	Data Property	-	MediaFragment	xsd:string	-
fragmentId	Data Property	-	MediaFragment	xsd:string	-
fragmentDescription	Data Property	-	MediaFragment	xsd:string	-
fragmentKeyword	Data Property	-	MediaFragment	xsd:string	-
languageValue	Data Property	-	Language	xsd:string	-
collectionName	Data Property	-	Collection	xsd:string	-
trackName	Data Property	-	Track	xsd:string	-
timePoint	Data Property	-	VideoTrack	xsd:decimal	-
videoTrackDuration	Data Property	-	VideoTrack	xsd:decimal	-
organisationName	Data Property	-	Organisation	xsd:string	-
organisationId	Data Property	-	Organisation	xsd:string	-
organisationLocator	Data Property	-	Organisation	xsd:anyURI	-
hasFragment	Object Property	-	MediaResource	MediaFragment	isFragmentOf
hasContributor	Object Property	-	MediaResource	Agent	hasContributedTo
hasVideoTrack	Object Property	-	MediaResource	VideoTrack	isVideoTrackOf
hasLanguage	Object Property	-	MediaResource	Language	isLanguageOf
hasFragmentLanguage	Object Property	-	MediaFragment	Language	isFragmentLanguageOf

Tabla H.9. Modelo ontológico final para la adaptación de “*Ontology for Media Resources*”.
Fuente: propia.

Fase B.5: determinar inferencias

Para finalizar, se vio la posibilidad de realizar algunas inferencias con respecto a los segmentos que hacen parte de los contenidos de VoD. Suponiendo que hay una instancia llamado *MR1* de la clase *MediaResource* que representa un contenido, y que está vinculado a la instancia *Track1* de la clase *videoTrack* por medio de la Object Property *:hasVideoTrack*. Entonces se definen las tripletas:

```
:MR1      :hasvideoTrack   :Track1.
```

Si se establece que *:hasVideoTrack* es subpropiedad de *:hasTrack*, y éste también lo es de *:hasFragment* de la siguiente forma:

```
:hasVideoTrack rdfs:subPropertyOf :hasTrack,
```

```
:hasTrack rdfs:subPropertyOf :hasFragment (esta tripleta está ya definida en la ontología original). Entonces se infiere que:
```

```
:MR1 :hasFragment :Track1.
```

Este proceso se ilustra en la Figura H.3.

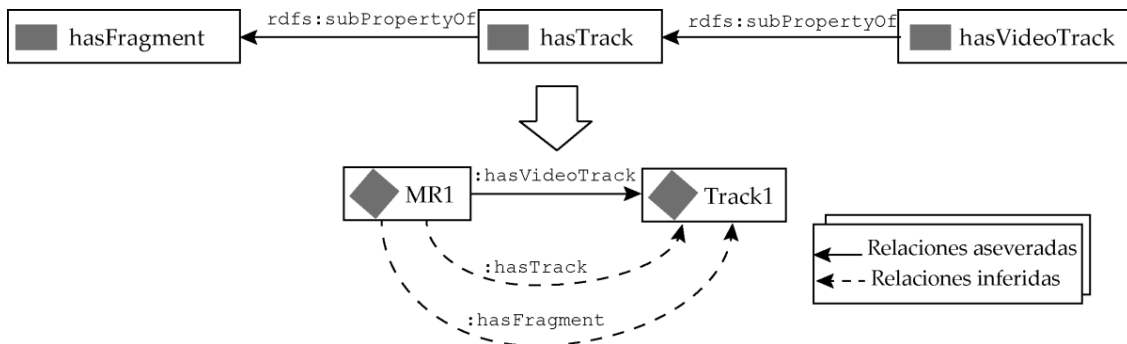


Figura H.3 Ejemplo de inferencia entre los contenidos de VoD y los segmentos de video.
Fuente: propia.

Anexo I

Versión extendida de la adaptación de “Competency Model Ontology”

El presente anexo muestra el proceso completo desarrollado para adaptar la ontología “Competency Model Ontology” mostrado en la Sección 4.1.3, de acuerdo a las fases mostradas en la Figura 4.1 (Capítulo 4). Todo este proceso se explica a continuación.

Fase B.1: identificar las RI de las ontologías existentes

Siguiendo la dinámica de las secciones anteriores, inicialmente se identificó el *diccionario de conceptos* de “Competency Model Ontology”, el cual se presenta en la Tabla I.1. En la primera columna están los nombres de los conceptos, en la segunda las instancias del concepto si las hubiere, en la tercera columna los atributos de las instancias, y en la cuarta, las relaciones asociadas a los conceptos.

Nombre del concepto	Instancias	Atributos de instancias	Relaciones del concepto
Competency	-	-	Domain-topic-ref Skill-ref
Concept	-	-	-

Tabla I.1. Diccionario de conceptos de “Competency Model Ontology”. Fuente: propia.

Con el propósito de establecer las relaciones entre conceptos, se identificaron las *relaciones binarias* existentes de acuerdo con las relaciones del *diccionario de conceptos* anterior, las cuales se presentan en la Tabla I.2.

Nombre de la relación	Concepto fuente	Concepto objetivo	Relación inversa
Domain-topic-ref	Competency	Concept	ND
Skill-ref	Competency	Concept	ND

Tabla I.2. Relaciones binarias de “Competency Model Ontology”. Fuente: propia.

El modelo ontológico se basa en definir conceptos asociados a las competencias (“*Domain topics*”) y las habilidades base de las mismas dentro de un mismo concepto. Esta es una forma de representar conocimiento de manera semiformal, en el que los atributos y/o relaciones no están bien establecidos; y, por lo tanto, cualquier concepto puede ser vinculado con otro a través de cualquier tipo de relación.

Considerando el *diccionario de conceptos* y la *tabla de relaciones binarias* anterior, se ha construido la *red semántica* de la Figura I.1, la cual resume los conceptos y las relaciones entre ellos.

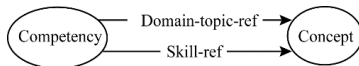


Figura I.1 Red semántica de “Competency Model Ontology”. Fuente: propia.

Fase B.2: cotejar ontología existente-esquema de metadatos

Aquí se evaluó si “Competency Model Ontology” permite describir toda la información descrita por la extensión del IEEE RCD para competencias propuesta en el Capítulo 3. El objetivo de esto es conseguir una correspondencia entre el modelo descriptivo provisto por la extensión (por ejemplo, el mostrado en la Figura 3.4, Sección 3.2) y el modelo ontológico conseguido con la adaptación de “Competency Model Ontology”, de tal forma que se puedan aplicar técnicas de la web semántica sobre él.

De forma resumida, la información descrita por la extensión se lista a continuación (una explicación con detalle sobre la naturaleza de la información, se presentó en la Sección 3.2), indicando su naturaleza y las posibilidades que ofrece en cuanto a la aplicación de asociaciones semánticas complejas:

1. *La categoría de las competencias (C1)*: es información útil para posicionar las competencias dentro de una de las categorías *conocimiento, habilidad y capacidad, actitud, valor*. Dado que esta información puede especificarse a través de una de las cinco mencionadas opciones, lo cual está en concordancia con el vocabulario para el elemento `KSC_category` de la extensión del IEEE RCD, se posibilita el establecimiento de vínculos lógicos entre competencias, lo cual favorece la formalización de asociaciones semánticas complejas entre éstas.
2. *El verbo de acción involucrado en el desarrollo de la competencia (C2)*: información que expresa acción física o mental. De acuerdo con la extensión del IEEE RCD propuesta, los valores para el *verbo de acción*, representado en el esquema de metadatos a través del elemento `action_verb`, se pueden definir por medio de un vocabulario (Tabla 3.3, Sección 3.2). En consecuencia, al adaptar esta información como un elemento ontológico, es posible establecer asociaciones semánticas complejas entre competencias.
3. *El esquema de cualificación de la competencia (C3)*: esta información es usada para clasificar una competencia de acuerdo a esquemas formales internacionales de cualificación, por ejemplo, el EQF [51] y la CINE [91]. Dado que esta información está representada en la extensión del IEEE RCD para competencias por medio de los elementos *scheme* y *level*, para los cuales se ha definido un vocabulario en la

I. Versión extendida de la adaptación de “Competency Model Ontology”

Tabla 3.3, esta puede ser usada para definir asociaciones semánticas complejas entre competencias.

4. *Rol de usuario al que interesa la competencia (C4)*: esta información es adaptada de LOM, e indica el rol de la persona a quien puede interesar la competencia. Esta información se formaliza en la extensión del IEEE RCD por medio del elemento `intended_user_role`, para el cual se definió en la Tabla 3.2 un vocabulario. Entonces, al formalizar esta información en la ontología, se puede favorecer la definición de asociaciones semánticas complejas entre competencias.
5. *La edad mínima del estudiante para la cual típicamente éste tiene la capacidad para desarrollar las competencias (C5)*: esta información define la edad mínima que puede tener un estudiante para aprender la competencia. En el IEEE RCD se representa esta información por medio del elemento `typical_age`, para el cual no fue posible definir un vocabulario. Por esta razón, esta información, al ser formalizada en la ontología, no puede ser usada para definir asociaciones semánticas complejas. Sin embargo, puede ser incluida en una estrategia de recomendación, por ejemplo, para filtrar los contenidos de acuerdo a la edad del estudiante.
6. *Dificultad de aprendizaje de la competencia (C6)*: esta información es adaptada de LOM para indicar la dificultad inherente al aprendizaje de la competencia. En la extensión del IEEE RCD se representa por medio del elemento `difficulty`, para el cual se definió un vocabulario en la Tabla 3.3. En principio, esto favorecería la definición de asociaciones semánticas complejas. Sin embargo, tal como se adelantó en la Sección 3.2, esta información podría no tener el significado suficiente para considerarla como una propiedad asociadora entre competencias. Ésta es una decisión que depende del contexto de la aplicación en la que se implemente la ontología.
7. *Tiempo típico de aprendizaje de la competencia (C7)*: se refiere al tiempo en semanas usado normalmente en el aprendizaje de la competencia. En el IEEE RCD se representa esta información por medio del elemento `typical_learning_time`, para el cual no fue posible definir un vocabulario. Por esta razón, esta información, al ser formalizada en la ontología, no puede ser usada para definir asociaciones semánticas complejas. Sin embargo, puede ser incluida en una estrategia de recomendación, por ejemplo, basada en un cronograma y períodos de estudio de las competencias establecidos en un currículo escolar.
8. *Temas asociados a las competencias (C8)*: información que establece temas en los que se enmarcan las competencias (por ejemplo, *El Teorema de Pitágoras*,

Las figuras geométricas y Los números reales). En el IEEE RCD se representa esta información por medio del elemento `topic`, para el cual no fue posible definir un vocabulario. Por esta razón, esta información, al ser formalizada en la ontología, no puede ser usada para definir asociaciones semánticas complejas. Sin embargo, puede ser usada, por ejemplo, para enfocar las estrategias de recomendación hacia los temas relacionados con las competencias en las que los estudiantes tengan necesidades educativas.

9. *Grado educativo en el que se aplica la competencia (C9)*: información que se refiere a grados escolares tales como *primero de primaria, segundo de primaria, décimo de secundaria*, etc. En el IEEE RCD se representa esta información por medio del elemento `grade`, para el cual no fue posible definir un vocabulario. Por esta razón, esta información, al ser formalizada en la ontología, no puede ser usada para definir asociaciones semánticas complejas. Sin embargo, puede ser usada en la estrategia de recomendación, por ejemplo, para filtrar las recomendaciones de acuerdo al grado educativo del estudiante objetivo.
10. *Palabras clave (C10)*: es información comúnmente usada por cualquier esquema de metadatos actual que busca posicionar las descripciones en la web. En el IEEE RCD se representa esta información por medio del elemento `keywords`, para el cual no fue posible definir un vocabulario. Por esta razón, esta información, al ser formalizada en la ontología, no puede ser usada para definir asociaciones semánticas complejas. Sin embargo, puede ser usada en la estrategia de recomendación, por ejemplo, para obtener recomendaciones de acuerdo a similitudes sintácticas o semánticas entre competencias en las que el estudiante objetivo tenga necesidades educativas.
11. *Lenguaje de la descripción (C11)*: es información que define el lenguaje de la descripción de la competencia. En el IEEE RCD se representa esta información por medio del elemento `language`, para el cual se definió un vocabulario en la Tabla 3.3. Por esta razón, esta información, al ser formalizada en la ontología, puede ser usada para definir asociaciones semánticas complejas. Sin embargo, podría no tener el significado suficiente para ser considerada como propiedad vinculante entre competencias. Ésta es una decisión que depende del contexto de la aplicación en la que se implemente la ontología.

Fase B.3: determinar las RI adaptadas

Después de constatar que “*Competency Model Ontology*” no cubre ninguno de los once tipos de información (*C1, C2, ..., C11*), se inició el proceso de adaptación incluyendo nuevos conceptos, atributos y relaciones, teniendo en cuenta los elementos de datos definidos en la extensión del IEEE RCD. Manteniendo los criterios de

I. Versión extendida de la adaptación de “Competency Model Ontology”

adaptación considerados hasta ahora, los elementos de la extensión para los cuales se han definido vocabularios se representan en la ontología como conceptos; los posibles valores de los vocabularios se definen como instancias de tales conceptos, lo cual favorece la definición de asociaciones semánticas, los elementos que no poseen vocabularios se definen en la ontología como atributos.

Esto se puede observar en el nuevo *diccionario de conceptos* adaptado de “*Competency Model Ontology*” presentado en la Tabla I.3. Los nuevos elementos, aportados por el presente trabajo, se indican en negrita.

Nombre del concepto	Instancias	Atributos de instancias	Relaciones del concepto
Competency	-	Competency-Identifier Competency-Title Competency-topic Competency-Keywords Competency-grade	Has-definition Has-level-schema Has-schema-level Has-type Has-action-verb Has-intended-user Has-user-age Has-difficulty Has-language Has-learning-time Domain-topic-ref Skill-ref
Competency-definition	-	Competency-definition-model-source	Has-statement Is-definition-of
Statement	-	Statement-Id Statement-name Statement-text	Has-token Is-statement-of
Statement-token	-	Statement-source Statement-value	Is-statement-token-of
Schema	Isced-unesco, EQF	Schema-value	Is-schema-of
Level	<ul style="list-style-type: none"> • Para isced-unesco, los niveles pueden ser consultados en [91] (por ejemplo, 0, 1, 2, etc.) • Para EQF, los niveles pueden ser consultados en [94] (por ejemplo, k1, k2, s1, c2, etc.) 	Level-value	Is-level-of

Competency-type	Knowledge, Skill, Competency, attitude, value	Competency-type-value	Is-type-of
Verb	38 verbos tomados de LOM-ES, los pueden ser consultados en [92] (por ejemplo, analyzing, comparing, creating, recognizing, etc.)	Verb-value	Is-action-verb-of
User	Author, Learner, Manager, Teacher	User-value	Is-intended-user-of
Difficulty	Difficult, Easy, Medium difficulty, Very difficult, Very easy	Difficulty-value	Is-difficulty-of
Language	Vocabulario acorde con la descripción de competencias: código de dos letras del iso 639-1	Language-value	Is-language-of
Concept	-	-	-

Tabla I.3. Diccionario de conceptos de la adaptación de “Competency Model Ontology”.
Fuente: propia.

Cabe destacar que el nuevo modelo adaptado de “Competency Model Ontology” comparte algunos conceptos, atributos y relaciones con el modelo adaptado de “Learning Object Metadata Ontology”. Éstos son *User*, *Difficulty* y *Language*; lo cual vincula las dos ontologías y, por lo tanto, debe considerarse en el proceso de integración posterior.

La Tabla I.4 presenta todas las *relaciones binarias* existentes de acuerdo con las relaciones del anterior *diccionario de conceptos*.

Nombre de la relación	Concepto fuente	Concepto objetivo	Relación inversa
Has-definition	Competency	Definition	Is-definition-of
Has-level	Competency	Level-schema	Is-level-schema-of
Has-schema	Competency	Schema-level	Is-schema-level-of
Has-type	Competency	Competency-type	Is-type-of
Has-action-verb	Competency	Verb	Is-action-verb-of
Has-intended-user	Competency	User	Is-intended-user-of

I. Versión extendida de la adaptación de “Competency Model Ontology”

Has-difficulty	Competency	Difficulty	Is-difficulty-of
Has-language	Competency	Language	Is-language-of
Has-statement	Competency-definition	Statement	Is-statement-of
Has-token	Statement	Statement-token	Is-token-of
Domain-topic-ref	Competency	Concept	ND
Skill-ref	Competency	Concept	ND

Tabla I.4. Relaciones binarias para la adaptación de “Competency Model Ontology”.

Fuente: propia.

A partir del *diccionario de conceptos* y las *relaciones binarias* anteriores, se ha construido la *red semántica* de la Figura I.2, la cual señala las diferentes relaciones entre conceptos del nuevo modelo ontológico adaptado de “*Ontology for Media Resources*”. En gris se señalan los aportes a la ontología original en cuanto a conceptos y atributos (señalados por la relación *Has*) y en *negrita-cursiva* en cuanto a relaciones.

La Tabla I.5 corresponde a la *tabla de atributos de instancias* del *diccionario de conceptos* final.

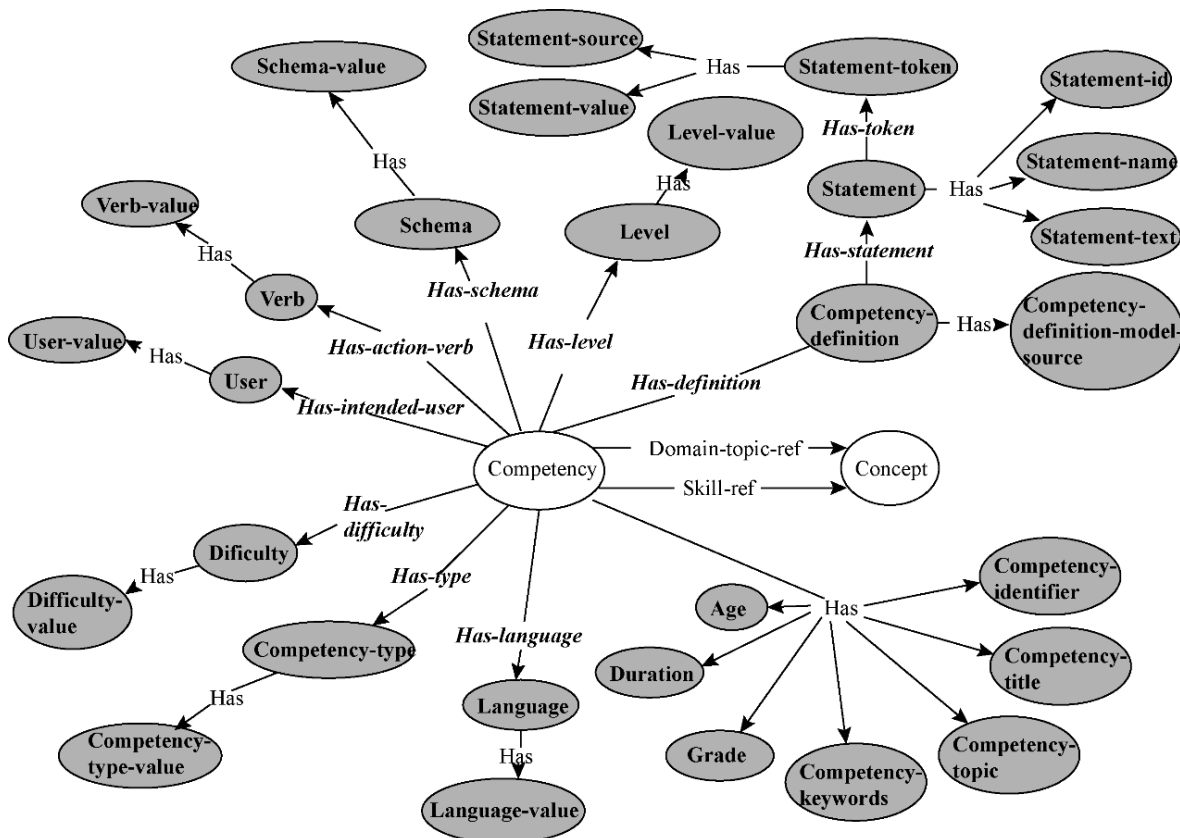


Figura I.2 Red semántica de la adaptación de “Competency Model Ontology”. Fuente: propia.

Nombre del atributo de instancias	Tipo de valor	Unidad de medida	Rango de valores	Valor por defecto	Cardinalidad
Competency-Identifier	String	-	-	-	(1,1)
Competency-Title	String	-	-	-	(1,1)
Competency-topic	String	-	-	-	(1,n)
Competency-Keywords	String	-	-	-	(1,1)
Competency-definition-model-source	String	-	-	-	(1,1)
Statement-Id	String	-	-	-	(1,1)
Statement-name	String	-	-	-	(1,1)
Statement-text	String	-	-	-	(1,1)
Statement-source	String	-	-	-	(1,1)
Statement-value	String	-	-	-	(1,1)
Level-value	String	-	-	-	(1,1)
Schema-value	String	-	-	-	(1,1)
Competency-type-value	String	-	-	-	(1,1)
Verb-value	String	-	-	-	(1,1)
User-value	String	-	-	-	(1,1)
Age	String	-	-	-	(1,1)
Difficulty-value	String	-	-	-	(1,1)
Language-value	String	-	-	-	(1,1)
Duration	Decimal	-	-	-	(1,1)

Tabla I.5. Tabla de atributos de instancias de la adaptación de “Competency Model Ontology”. Fuente: propia.

La Tabla I.6 presenta la *tabla de instancias* que aparecen en el *diccionario de conceptos* adaptado.

Nombre de la instancia	Atributos con valores conocidos	Valor de los atributos
Isced-unesco	Scheme-value	Isced-unesco
EQF	Scheme-value	EQF
Los niveles del isced-unesco [91]	Level-value	Correspondiente a cada nivel
Los niveles de EQF [94]	Level-value	Correspondiente a cada nivel
Knowledge	Competency-type-value	Knowledge
Skill	Competency-type-value	Skill
Competency	Competency-type-value	Competency
Los 38 verbos definidos en [92]	Verb-value	Correspondiente a cada verbo
Author	User-value	Author
Learner	User-value	Learner
Manager	User-value	Manager
Teacher	User-value	Teacher
Difficult	Difficulty-value	Difficult
Easy	Difficulty-value	Easy
Medium difficulty	Difficulty-value	Medium difficulty
Very difficult	Difficulty-value	Very difficult
Very easy	Difficulty-value	Very easy
Los códigos de dos letras del iso 639-1	Language-value	Correspondiente a cada código

I. Versión extendida de la adaptación de “Competency Model Ontology”

Tabla I.6. Instancias definidas en la adaptación de “*Competency Model Ontology*”.
Fuente: propia.

Fase B.4: determinar el modelo ontológico

De acuerdo con los resultados de las fases anteriores, se definió el modelo ontológico definitivo para la adaptación de “*Learning Object Metadata Ontology*” desde el punto de vista de la implementación, el cual se muestra en la Tabla I.7. En negrita se indican los elementos ontológicos que son aporte del trabajo.

Elemento de la ontología	Tipo de elemento	Instancias	Dominio	Rango	Propiedades inversas
Competency	Class	-	-	-	-
CompetencyDefinition	Class	-	-	-	-
Statement	Class	-	-	-	-
StatementToken	Class	-	-	-	-
Level	Class	Los niveles del isced-unesco [91] y los de EQF [94]	-	-	-
Schema	Class	Isced-unesco, EQF	-	-	-
CompetencyType	Class	Knowledge, Skill, Competency, attitude, value	-	-	-
Verb	Class	Los 38 verbos definidos en [92]	-	-	-
UserRole	Class	Author, Learner, Manager, Teacher	-	-	-
Difficulty	Class	Difficult, Easy, Medium difficulty, Very difficult, Very easy	-	-	-
Language	Class	Código de dos letras del iso 639-1	-	-	-
Concept	Class	-	-	-	-
hasDefinition	Object Property	-	Competency	Competency Definition	isDefinitionOf
hasLevel	Object Property	-	Competency	Level	isLevelOf
hasSchema	Object Property	-	Competency	Schema	isSchemaOf
hasCompetencyType	Object Property	-	Competency	Competency Type	isTypeOf
hasActionVerb	Object Property	-	Competency	Verb	isActionVerbOf
hasIntendedUser	Object Property	-	Competency	UserRole	isIntendedUserOf
hasDifficulty	Object Property	-	Competency	Difficulty	isDifficultyOf

hasLanguage	Object Property	-	Competency	Language	isLanguageOf
DomainTopicRef	Object Property	-	Competency	Concept	ND
SkillRef	Object Property	-	Competency	Concept	ND
hasStatement	Object Property	-	CompetencyDefinition	Statement	isStatementOf
hasToken	Object Property	-	Statement	StatementToken	isTokenOf
hasCompetencyId	Data Property	-	Competency	xsd:string	-
hasCompetencyTitle	Data Property	-	Competency	xsd:string	-
hasCompetencyTopic	Data Property	-	Competency	xsd:string	-
hasCompetencyGrade	Data Property	-	Competency	Xsd:string	-
hasCompetencyKeyword	Data Property	-	Competency	xsd:string	-
competencyDefinitionModelSource	Data Property	-	CompetencyDefinition	xsd:string	-
statementId	Data Property	-	Statement	xsd:string	-
statementName	Data Property	-	Statement	xsd:string	-
statementText	Data Property	-	Statement	xsd:string	-
statementSource	Data Property	-	StatementToken	xsd:string	-
statementValue	Data Property	-	StatementToken	xsd:string	-
levelSchemaValue	Data Property	-	LevelSchema	xsd:string	-
schemaLevelValue	Data Property	-	SchemaLevel	xsd:string	-
competencyTypeValue	Data Property	-	CompetencyType	xsd:string	-
verbValue	Data Property	-	Verb	xsd:string	-
userRoleValue	Data Property	-	UserRole	xsd:string	-
hasUserAge	Data Property	-	Competency	xsd:string	-
difficultyValue	Data Property	-	Difficulty	xsd:string	-

I. Versión extendida de la adaptación de “Competency Model Ontology”

languageValue	Data Property	-	Language	xsd:string	-
hasDuration	Data Property	-	Competency	xsd:decimal	-

Tabla I.7. Modelo ontológico para la adaptación de “*Competency Model Ontology*”.

Fuente: propia.

Anexo J

Versión extendida de la construcción de OntoRCM

Este anexo presenta en extendido el proceso de creación de *OntoRCM*, la ontología para el modelado de mapas de competencias propuesta en la Sección 4.2.

Con toda la información alrededor de los mapas de competencias y desarrollando la metodología introducida al inicio del Capítulo 4, se construyó la nueva ontología, siendo esta un aporte de la presente investigación.

Fase C.1: determinar las RI

Después del análisis anterior, se ha construido el *diccionario de conceptos* presentado en la Tabla J.1. En primera instancia fueron identificados cuatro conceptos relacionados con la estructura de los mapas:

1. *Competency-map*: concepto que representa a los mapas de competencias.
2. *Node*: concepto que representa los nodos que constituyen los mapas.
3. *Competency*: concepto que representa las competencias.
4. *Assessment-activity*: concepto que representa las actividades de evaluación.

Ahora bien, siendo consistente con la construcción de las ontologías realizada hasta ahora, se definieron los conceptos a partir de los elementos del esquema de metadatos cuyo valor obedece a un vocabulario. De acuerdo con SRCM, otros conceptos son:

1. *Proficiency*: concepto que representa los niveles de cumplimiento de las competencias. Aunque en SRCM el valor de este elemento se establece por medio de un valor real entre cero y uno, se decidió incluir los valores lingüísticos: muy bajo, bajo, medio, alto, muy alto; esto considerando la naturaleza cualitativa del presente trabajo.
2. *Rollup-method*: concepto que representa los métodos *rollup* usados para calcular el nivel de cumplimiento de las competencias (*Proficiency level* en el esquema SRCM). Al respecto, SRCM define un grupo de métodos “*rollup*” de la siguiente manera:

- *All*: es usado para indicar que una competencia es aprobada por el estudiante si todas sus subcompetencias son aprobadas.
- *Any*: sirve para definir que una competencia es aprobada si cualquiera de sus subcompetencias es aprobada.
- *Fraction*: este método es usado para indicar que una competencia es aprobada si una fracción de sus subcompetencias es aprobada. La fracción se expresa por medio de un número real.
- *Units*: es usado para indicar que una competencia es aprobada si un número de sus subcompetencias es aprobado. Tal número se expresa por medio de un entero.
- *Mean*: este método indica que el NC de la competencia se calcula promediando los NC de sus subcompetencias. Aunque en SRCM los NC considerados en el método *mean* son valores numéricos (los resultados pueden calcularse por medio de una suma ponderada), en una aplicación del esquema debe considerarse que los valores pueden también ser expresiones lingüísticas. De esta forma, un sistema encargado de calcular NC con base en la información definida en registros de metadatos que utilizan SRCM, deben reconocer si las expresiones usadas son numéricas o lingüísticas, para lo cual deberá ejecutar un modelo matemático apropiado para cada caso.

Para la definición de atributos de instancias y relaciones de los conceptos, fueron utilizados los demás elementos del esquema de metadatos SRCM. Esto se ha señalado entre paréntesis dentro del diccionario, indicando el mapeo entre los metadatos y los elementos de la ontología.

Nombre del concepto	Instancias	Atributos de instancias	Relaciones del concepto
Competency-map	-	Map-identifier (identifier) Map-title (title)	Has-node Has-top-node (entryNodes) Has-default-entry-node (defaultEntryNode)
Node	-	Node-identifier (nodeId) Node-title (title) Data-required (dataRequired) Rollup-parameter (rollupParameter)	1. Has-competency-ref (rcdReference) 2. Has-assessment-activity-ref 3. Has-child (nodeRef) 4. Has-child-proficiency-required (proficiencyRequired) 5. Has-child-proficiency-desired (proficiencyDesired) 6. Has-rules-proficiency-required (proficiencyRequired) 7. Has-rules-proficiency-desired (proficiencyDesired) 8. Has-weight (weight) • Has-very-low-weight

J. Versión extendida de la construcción de OntoRCM

			<ul style="list-style-type: none"> • Has-low-weight • Has-medium-weight • Has-high-weight • Has-very-high-weight 9. Is-child-of (parents) 10. Has-rollup-method 11. Is-node-of 12. Is-top-node-of 13. Is-default-entry-node-of 14. Is-child-of
Competency	-	-	Is-competency-ref-of
Assessment-activity	-	Assessment-activity-title	Is-assessment-activity-of
Proficiency	Ver-low, Low, Medium, High, Very-high	Proficiency-value	Is-child-proficiency-required-of Is-child-proficiency-desired-of Is-rules-proficiency-required-of Is-rules-proficiency-desired-of
Rollup-method (rollupMethod)	All, Any, Fraction, Fuzzy, Mean, Units	Rollup-method-value	Is-rollup-method-of

Tabla J.1. Diccionario de conceptos para OntoRCM. Fuente: propia.

Para entender las relaciones entre conceptos, se ha construido la tabla de *relaciones binarias* de la Tabla J.2.

Nombre de la relación	Concepto fuente	Concepto objetivo	Relación inversa
Has-node	Competency-map	Node	IsNodeOf
Has-top-node	Competency-map	Node	Is-top-node-of
Has-default-entry-node	Competency-map	Node	Is-default-entry-node-of
Has-competency-ref	Node	Competency	Is-competency-ref-of
Has-assessment-activity-ref	Node	Assessment-activity	Is-assessment-activity-ref-of
Has-child	Node	Node	Is-child-of
Has-child-proficiency-required	Node	Proficiency	Is-child-proficiency-required-of
Has-child-proficiency-desired	Node	Proficiency	Is-child-proficiency-desired-of
Has-rules-proficiency-required	Node	Proficiency	Is-rules-proficiency-required-of
Has-rules-proficiency-desired	Node	Proficiency	Is-rules-proficiency-desired-of
Has-weight	Node	Node	Is-weight-of
Has-rollup-method	Node	Rollup-method	Is-rollup-method-of

Tabla J.2. Relaciones binarias para OntoRCM. Fuente: propia.

La Figura J.1 presenta la *red semántica* que ilustra los conceptos, sus atributos y relaciones. La Tabla J.3, corresponde a la *tabla de atributos de instancias* del *diccionario de conceptos*.

Con el modelo ontológico ya definido, a continuación, se presenta la ontología para los mapas de competencias desde el punto de vista de la implementación, buscando aprovechar el potencial de las tecnologías disponibles.

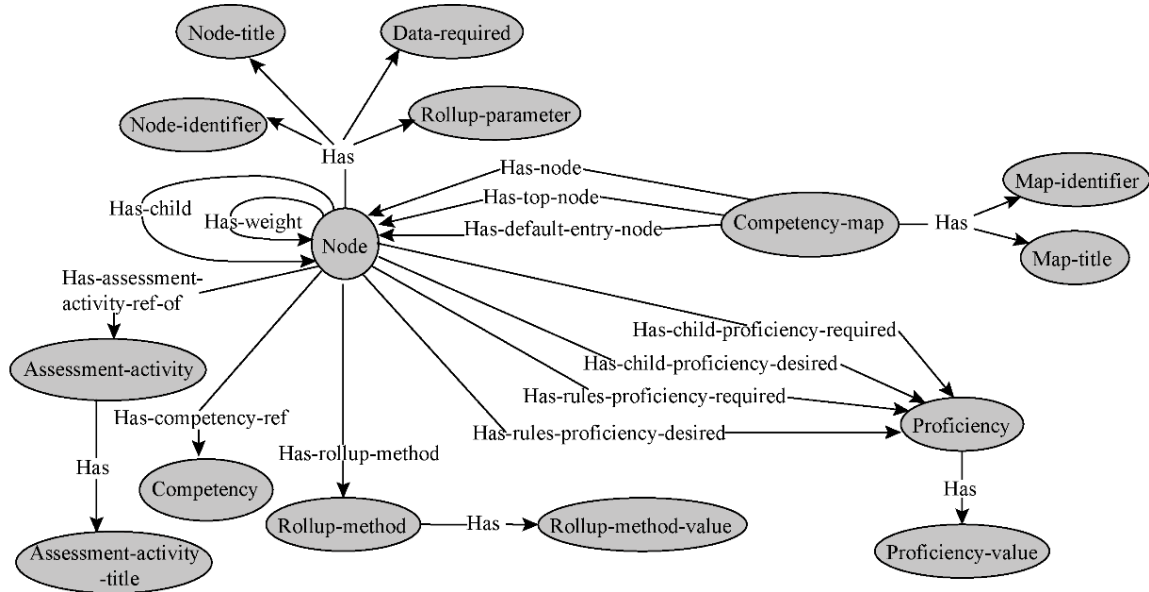


Figura J.1 Red semántica para OntoRCM. Fuente: propia.

Nombre del atributo de instancias	Tipo de valor	Unidad de medida	Rango de valores	Valor por defecto	Cardinalidad
Map-identifier	String	-	-	-	(1,1)
Map-title	String	-	-	-	(1,1)
Node-identifier	String	-	-	-	(1,1)
Node-title	String	-	-	-	(1,1)
Data-required	Boolean	-	{True, False}	True	(1,1)
Rollup-parameter	Decimal	-	$(-\infty, +\infty)$	-	(1,1)
Proficiency-value	String	-	-	-	(1,1)
Rollup-method-value	String	-	-	-	(1,1)

Tabla J.3. Tabla de atributos para *OntoRCM*. Fuente: propia.

La Tabla J.4, presenta la *tabla de instancias* que aparecen en el diccionario de conceptos.

Nombre de la instancia	Atributos con valores conocidos	Valor de los atributos
All	Rollup-method-value	All
Any	Rollup-method-value	Any
Fraction	Rollup-method-value	Fraction
Fuzzy	Rollup-method-value	Fuzzy
Mean	Rollup-method-value	Mean

Units	Rollup-method-value	Units
Very-low	Proficiency-value	Very-low
Low	Proficiency-value	Low
Medium	Proficiency-value	Medium
High	Proficiency-value	High
Very-high	Proficiency-value	Very-high

Tabla J.4. Tabla de instancias para *OntoRCM*. Fuente: propia.

Fase C.2: determinar el modelo ontológico

Las *Representaciones Intermedias* (RI) de la nueva ontología, mostradas en la sección anterior, permitieron identificar los conceptos, sus atributos y relaciones con respecto al conocimiento en torno a los mapas de competencias. Con base en ellos, en esta sección se construye la jerarquía de clases, propiedades e instancias.

Antes de iniciar, aquí es tratada una situación considerada clave para el desarrollo del presente trabajo. Y es que se avizoran dos posibles soluciones en la construcción de la representación de conocimiento: la primera, basada en los mismos principios tenidos en cuenta hasta ahora en la adaptación en las anteriores ontologías, en la que hay correspondencia entre conceptos, atributos y relaciones con nuevas clases, “*Object properties*” y “*Data Properties*”; la segunda, basada en la reutilización del modelo de SKOS (“*Simple Knowledge Organisation System*”) [103], el cual define clases, propiedades y atributos que sintáctica y semánticamente tienen semejanzas con los conceptos, atributos y relaciones del modelo de los mapas de competencias conseguido .

Las dos soluciones resultan ser válidas, y sin ventajas una sobre la otra a la luz del Sistema de Recomendaciones. Sin embargo, el uso de SKOS potenciaría el uso de la ontología por sí misma, pudiendo ser reutilizada y refinada de forma eficiente en otras aplicaciones, manteniendo la interoperabilidad de las mismas (este es uno de los factores claves del éxito de la web semántica). Otra razón por la cual usar SKOS radica en que las relaciones existentes entre las competencias representan conocimiento semiformal en el marco de los mapas de competencias. Dadas las capacidades de reutilización que ofrece el SRCM, permitiendo usar una competencia IEEE RCD en diferentes mapas de competencias, no es posible asegurar que una competencia sea siempre subcompetencia de otra, por lo cual las relaciones jerárquicas se establecen subjetivamente, lo cual representa conocimiento semiformal. Bajo estas condiciones, SKOS ha sido creado exclusivamente para organizar conocimiento semiformal de forma distribuida y enlazable, por lo cual es propicio para representar las relaciones entre competencias establecidas en los mapas de competencias. Cabe indicar que SKOS no puede ser usado en las ontologías descritas hasta ahora porque éste se centra en establecer clasificaciones jerárquicas entre conceptos (del tipo *más amplio*

y *más angosto*), mientras que las ontologías definen jerarquías de clases y describen instancias por medio de atributos, toda vez que éste conocimiento sea considerado como formal.

De acuerdo con [35], entre las características de SKOS destacadas están:

1. Define dos clases básicas para la representación del conocimiento. *Concept* para representar cualquier concepto (sin importar su clase o características) en una sola clase y *ConceptScheme* para estructurar los conceptos de la clase *Concept* por medio de relaciones. De lo anterior, se deduce que los nodos pueden estar representados como instancias de la clase *Concept* y los mapas como instancias de la clase *ConceptScheme*.
2. Define las propiedades `skos:Broader` y `skos:Narrower`, las cuales son usadas para la representación de vínculo jerárquicos, como la relación entre un todo y sus partes. `skos:Broader` es usada para establecer que un concepto es más amplio (o general) que otro, mientras que `skos:Narrower` sirve para definir que es más angosto (o específico). Con respecto a los mapas, están estrechamente relacionadas con las relaciones generales y específicas de las competencias. Además, define una estructura de relaciones semánticas (que incluye a `skos:Broader` y `skos:Narrower`) que permitiría obtener inferencias sobre las relaciones en los mapas.
3. Permite asociar un concepto a diferentes esquemas, lo cual es acorde con la vinculación de una competencia a diferentes mapas.
4. Define algunas propiedades directamente relacionadas con algunos elementos del esquema de metadatos para la descripción de los mapas en cuanto a su función. Por ejemplo, la propiedad `skos:hasTopConcept`, cuyo dominio es la clase *ConceptScheme* y rango *Concept*, es usada para indicar el concepto más general del esquema. Esto es útil para indicar el punto de partida en un proceso de recuperación de información. Por ejemplo, al recorrer un mapa de competencias en búsqueda de las relaciones entre las competencias, la propiedad `skos:hasTopconcept` puede ser usada para identificar las competencia por la cual iniciar el recorrido a través, por ejemplo, de algoritmos como Búsqueda primero en profundidad o Búsqueda primero en anchura.
5. Esta propiedad presenta una estrecha relación con el elemento *entryNodes* del esquema, utilizado para indicar los nodos más generales de los mapas.

De acuerdo a lo anterior, se elige el modelo SKOS para representar componentes de los mapas de competencias en la ontología porque: permite definir las relaciones entre competencias generales y específicas; permite vincular una competencia a diferentes

mapas al tiempo, permite definir cualquier clase de concepto; permite identificar las competencias más generales como las entradas para realizar procesos de búsqueda. Sin embargo, algunos elementos de los mapas no pueden ser representados por SKOS, por lo cual es necesario adaptar nuevos elementos (clases y propiedades) para obtener como resultado el modelo ontológico para mapas de competencias. Este proceso es explicado en lo que sigue.

Construcción de clases

La construcción de la representación de conocimiento consiste en verificar qué partes del modelo SKOS pueden representar los elementos ontológicos definidos en las RI ya definidas.

Para iniciar, se hace énfasis en que SKOS define algunos elementos relacionados con la determinación de las clases:

1. La clase `skos:Concept` está definida en el modelo como la clase estandarizada de todos los conceptos SKOS. En la práctica, los conceptos del conocimiento están representados por instancias de esta clase.
2. La clase `skos:ConceptScheme` está definida en el modelo como la clase estandarizada para integrar uno o más conceptos de forma ordenada. En la práctica, cada estructura de conceptos está representada por una instancia de esta clase.

Al respecto, anteriormente se mencionó que los mapas de competencias pueden ser representados a través de instancias de la clase `skos:ConceptScheme` y sus nodos por los de la clase `skos:Concept`. Esto significa que los conceptos *Competency-map* y *Node* estarán representados por las clases `skos:ConceptScheme` y `skos:Concept` respectivamente. Posteriormente, fue necesario representar los demás conceptos por medio de clases. Estos son: *Competency*, *Rollup-method* y *Proficiency*.

El resumen de las clases de la ontología de los mapas de competencias se presenta en la Tabla J.5 (las nuevas clases incluidas se señalan en negrita), en la cual además se señala el concepto al cual representa.

Concepto	Clase
Competency-map	<code>skos:ConceptScheme</code>
Node	<code>skos:Concept</code>
Competency	Competency
Assessment-activity	AssessmentActivity
Rollup-method	RollupMethod
Proficiency	Proficiency

Tabla J.5. Clases de la ontología para *OntoRCM*. Fuente: propia.

Definición de Data Properties

La definición de “*Data Properties*” pasa por representar los atributos de instancias del *diccionario de conceptos*. El resumen de ellas es presentado en la Tabla J.6, en la cual se relaciona el dominio y rango de tales propiedades.

Atributos de instancias	Data Properties	Dominio	Rango
Map-identifier	mapId	skos:ConceptScheme	xsd:string
Map-title	mapTitle	skos:ConceptScheme	xsd:string
Node-identifier	nodeId	skos:Concept	xsd:string
Node-title	nodeTitle	skos:Concept	xsd:string
Assessment-activity-title	activityTitle	AssessmentActivity	xsd:string
Data-Required	dataRequired	skos:Concept	xsd:boolean
Rollup-parameter	rollupParameter	skos:Concept	Xsd.decimal
Proficiency-value	proficiencyValue	Proficiency	xsd:string
Rollup-method-value	rollupMethodValue	RollupMethod	xsd:string

Tabla J.6. Tabla de atributos de instancias para *OntoRCM*. Fuente: propia.

Definición de Object Properties

La definición de “*Object Properties*” es un proceso de dos etapas: la primera, consiste en formalizar las relaciones entre los conceptos definidas en el diccionario de conceptos. La segunda, consiste en formalizar propiedades que permitan inferir conocimiento útil en torno a los mapas de competencias.

Como parte inicial de la primera etapa, se identificaron propiedades de SKOS que pueden representar las relaciones entre conceptos:

1. `skos:narrower`, `skos:broader`, `skos:narrowerTransitive` y `skos:broaderTransitive` son propiedades usadas para relacionar conceptos más generales y más específicos.
2. `skos:inScheme` es una propiedad útil para indicar que un concepto hace parte de un esquema.
3. `skos:hasTopConcept` es una propiedad usada para indicar el concepto más general de un esquema; su inversa `skos:topConceptOf` es usada para indicar que un concepto es el más general de un esquema.

Las relaciones entre conceptos han sido mapeadas en las propiedades de SKOS. Adicionalmente, han sido creadas otras “*Object Properties*” que cubren las demás relaciones que SKOS por sí solo no puede. Toda la información se resume en la Tabla J.7.

J. Versión extendida de la construcción de OntoRCM

Relaciones entre conceptos	Object Property	Dominio	Rango	Inversa
Has-node	:hasNode	skos:ConceptScheme	skos:Concept	skos:inScheme
Has-top-node	skos:hasTopConcept	skos:ConceptScheme	skos:Concept	skos:topConceptOf
Has-default-entry-node	:hasDefaultEntryNode	skos:ConceptScheme	skos:Concept	:isDefaultEntryNodeOf
Has-competency-ref	:hasCompetencyRef	skos:Concept	Competency	:isCompetencyRefOf
Has-assessment-activity-ref	:hasAssessmentActivityRef	Skos:Concept	AssessmentActivity	:isAssessmentActivityRefOf
Has-child	skos:narrower skos:narrowerTransitive	skos:Concept	skos:Concept	skos:broader skos:broaderTransitive
Has-rollup-method	:hasRollupMethod	skos:Concept	RollupMethod	:isRollupMethodOf
Has-child-proficiency-required	:hasChildProficiencyRequired	skos:Concept	Proficiency	:isChildProficiencyRequiredOf
Has-child-proficiency-desired	:hasChildProficiencyDesired	skos:Concept	Proficiency	:isChildProficiencyDesiredOf
Has-rules-proficiency-required	:hasRulesProficiencyRequired	skos:Concept	Proficiency	:isRulesProficiencyRequiredOf
Has-rules-proficiency-desired	:hasRulesProficiencyDesired	skos:Concept	Proficiency	:isRulesProficiencyDesiredOf
Has-Weight	:hasChildWeight	skos:Concept	skos:Concept	:hasParentWeight
	:hasChildVeryLowWeight	skos:Concept	skos:Concept	:hasParentVeryLowWeight
	:hasChildLowWeight	skos:Concept	skos:Concept	:hasParentLowWeight
	:hasChildMediumWeight	skos:Concept	skos:Concept	:hasParentMediumWeight
	:hasChildHighWeight	skos:Concept	skos:Concept	:hasParentHighWeight
	:hasChildVeryHighWeight	skos:Concept	skos:Concept	:hasParentVeryHighWeight

Tabla J.7. Modelo ontológico de *OntoRCM*. Fuente: propia.

Puede observarse que de la propiedad `:HasChildWeight` se definieron subpropiedades que hacen más explícita la relación entre un nodo padre y su hijo (por ejemplo, *HasChildVeryLowWeight*). De esta forma se ofrece la opción de inferir que hay una relación de importancia entre un nodo y otro (puede ser una relación de importancia que va desde un nivel muy bajo hasta muy alto). Además, para cada una

de ellas, se han definido propiedades inversas que permitirán inferir conocimiento útil acerca de la importancia de cierto nodo para su(s) padre(s).

El modelo definido hasta aquí permite consultar conocimiento tanto aseverado como inferido que puede ser de utilidad en el proceso para obtener recomendaciones. Del primer tipo se destacan:

1. Dado un nodo, conocer sus nodos más amplios inmediatos (los que responden a la propiedad `skos:broader`).
2. Dado un nodo, conocer sus nodos más estrechos inmediatos, que responden a cierto peso (los que responden a las propiedades *VeryLow*, *Low*, *Medium*, etc).
3. Los nodos más generales de un mapa (los que responden a la propiedad `skos:hasTopConcept`).
4. Los nodos que hacen parte de un mapa (los que responden a la propiedad `:hasNode`).

Del segundo tipo se destacan:

1. Dado un nodo, conocer sus nodos más estrechos inmediatos (los que responden a la propiedad `skos:narrower`).
2. Dado un nodo, conocer los nodos más amplios inmediatos que responden a cierto peso (los que responden a las propiedades *VeryLow*, *Low*, *Medium*, etc).
3. De qué mapa(s) hace parte cierto nodo.

Como ya se ha mencionado, la estructura de grafo de los mapas de competencias hace pensar que existe un conocimiento adicional que puede ser inferido a partir de las relaciones de subsunción, siendo esto útil en la recuperación de información. Por ende, la meta inmediata fue definir la forma en cómo usar las diferentes posibilidades que ofrece OWL para inferir conocimiento sobre los mapas. Este proceso de explica a continuación.

Fase C.3: determinar inferencias

Tal como se introdujo anteriormente, los mapas de competencias involucran relaciones de subsunción que vale la pena inferir. Después de un análisis de las posibilidades de inferencia sobre los mapas, se identificaron tres posibles escenarios así: inferencia de competencias generales y específicas; inferencia de competencias de las cuales depende una en particular; e inferencia de las competencias posibilitadas por una específica. Cada uno de estos se desarrolla a continuación.

Inferiendo competencias generales y específicas inmediatas

Desde el punto de vista educativo, la construcción de un mapa de competencias puede tener diversos principios base. Aparte de esto, las relaciones entre competencias

generales y específicas pueden ser interpretadas como de dependencia, de una u otra forma. Entre otros ejemplos, las competencias específicas facilitan el desarrollo de las generales, o son sus pre-requisitos dentro de un plan de estudios. Entonces, podría ser de gran utilidad consultar en la ontología qué competencia es general o específica para una competencia determinada.

Para incorporar esta posibilidad a la ontología, notar que las relaciones `skos:broader` y `skos:narrower` pueden no ser intuitivas para las personas, cuando se trata de establecer competencias generales y específicas. Por esta razón, han sido definidas dos nuevas “*Object Properties*” denominadas `:isGeneralFor` e `:isSpecificFor`. Cada una, subpropiedad de `skos:narrower` y `skos:broader` respectivamente, con lo cual se busca mantener enlazados los datos con la ontología base de SKOS. Además, al igual que `skos:broader` y `skos:narrower`, `:isGeneralFor` e `:isSpecificFor` también se definen como inversas. La Figura J.2 presenta la estructura de lo realizado para obtener las inferencias buscadas.

Para ilustrar la situación, se consideran 3 nodos, representados por las instancias *Node1*, *Node2* y *Node3*. En el mapa de competencias, se ha definido que *Node1* es padre de *Node2*, y éste de *Node3*. Entonces, se definen las tripletas:

```
:Node2      :isSpecificFor   :Node1.
```

```
:Node2      :isGeneralFor   :Node3.
```

Como `:isGeneralFor` e `:isSpecificFor` son subcompetencias de `skos:narrower` y `skos:broader` respectivamente, se infiere:

```
:Node2 skos:broader :Node1.
```

```
:Node2 skos:narrower :Node3.
```

Entonces, como `:isGeneralFor` e `:isSpecificFor` son inversas entre sí, se infiere:

```
:Node1 :isGeneralFor :Node2.
```

```
:Node3 :isSpecificFor :Node2.
```

Finalmente, como `skos:broader` y `skos:narrower` son inversas entre sí, se infiere:

```
:Node1 skos:narrower :Node2.
```

```
:Node3 skos:broader :Node2.
```

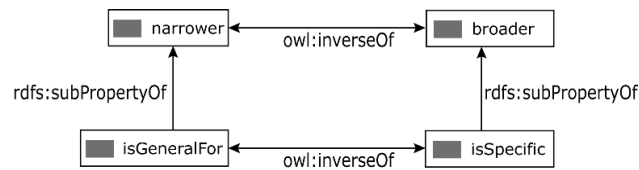


Figura J.2 Modelo para inferir competencias generales y específicas. Fuente: propia.

Infiriendo competencias de las cuales depende una en particular

Considerando que las competencias más específicas apoyan el aprendizaje de las más generales, podría ser de gran utilidad conocer todas las competencias del mapa de las cuales depende una en particular. Por ejemplo, para verificar el historial de niveles de cumplimiento de las competencias que ha tenido un estudiante para llegar a una competencia más general.

Para inferir este conocimiento a partir de lo definido hasta aquí, se ha creado una nueva Object Property llamada `:dependsOn`. Anteriormente, se definió una estructura que permite inferir las competencias específicas inmediatas. Esta, puede ser utilizada para obtener una lista de todas las competencias específicas de una en particular, asignándole la propiedad `owl:TransitiveProperty` y haciendo sus subpropiedades a `skos:narrower`, `:isGeneralFor` y `:hasChildWeight` (y todas sus subpropiedades) tal como lo ilustra la estructura de la Figura J.3.

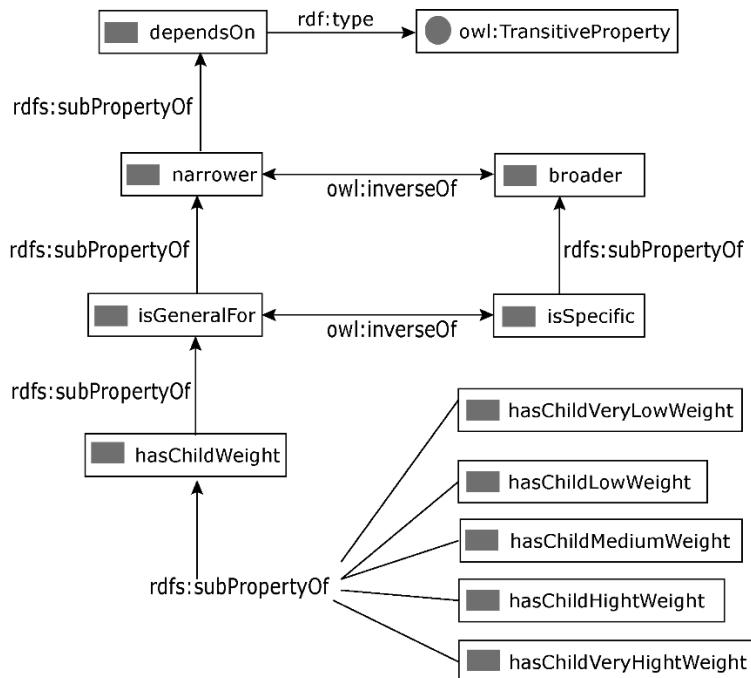


Figura J.3 Modelo para inferir competencias de las cuales depende una en particular. Fuente: propia.

Continuando con el ejemplo de los nodos *Node1*, *Node2* y *Node3*, anteriormente se obtuvieron las siguientes inferencias:

```
:Node1 skos:narrower :Node2.
```

```
:Node2 skos:narrower :Node3.
```

Como `skos:narrower` es subpropiedad de la propiedad transitiva `:dependsOn`, se infiere:

```
:Node1 :dependsOn :Node2.
```

```
:Node1 :dependsOn :Node3.
```

```
:Node2 :dependsOn :Node3.
```

Estas inferencias también se obtienen a partir de tripletas como las siguientes:

```
:Node1 :hasChildHighWeight :Node2.
```

```
:Node2 :hasChildMediumWeight :Node3.
```

Al definir propiedades como `:hasChildHighWeight` y `:hasChildMediumWeight` (y en realidad, cualquier subpropiedad de `:hasWeight`) se infieren tripletas en las que se involucran las propiedades: `:isGeneralFor`, `skos:narrower` y `:dependsOn`.

De esta forma se puede conocer de qué competencias depende otra en todo el mapa.

Infiriendo las competencias posibilitadas por una específica

Siguiendo un análisis similar al caso anterior, podría ser de gran utilidad conocer todas las competencias del mapa que pueden ser posibilitadas por una competencia en particular (por ejemplo, para plantear una carta de navegación por el mapa después de un nivel de cumplimiento satisfactorio por parte de un estudiante).

Para inferir este conocimiento, se ha creado una “*Object Property*” llamada `:enables`, a la cual se le ha dado la propiedad transitiva, y se han hecho subpropiedades de ésta a `skos:broader` e `:isSpecificFor`, tal como lo presenta la estructura de la Figura J.4.

De acuerdo con el ejemplo de los tres nodos que se ha venido desarrollando, anteriormente se obtuvieron las siguientes inferencias:

```
:Node2 skos:broader :Node1.
```

```
:Node3 skos:broader :Node2.
```

La nueva estructura permite inferir:

```
:Node2 :enables :Node1.
```

```
:Node3 :enables :Node2.
```

```
:Node3 :enables :Node1.
```

De esta forma es posible conocer todas las competencias que posibilita una en particular.

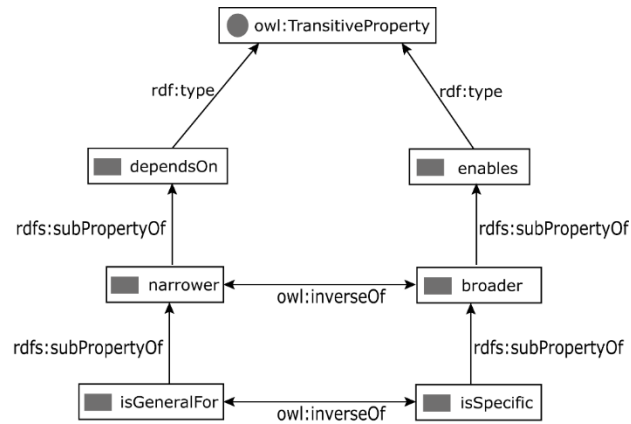


Figura J.4 Modelo para inferir competencias posibilitadas por una específica. Fuente: propia.

Anexo K

Versión extendida de la evaluación de la ontología propuesta

El presente anexo describe de forma detallada la evaluación de la ontología propuesta introducida en el Capítulo 4 (Sección 4.4) según la característica de *completitud*. De acuerdo con [39], ésta abarca los siguientes aspectos: *declaración de URI, literales y tipos de datos, validación XML, profundidad máxima de la jerarquía de conceptos, preguntas de competencia formalizadas, “hash” (#) vs “slash” (/), etiquetas y comentarios, nodos en blanco y la completitud del lenguaje*. Para la realización de la evaluación mostrada aquí se tuvieron en cuenta los primeros cinco conceptos porque están ligados a factores importantes dentro de los procesos de recomendación tales como: 1) la forma en cómo se consulta del conocimiento formalizado, para lo cual las URI, los literales y los tipos de datos deben estar bien declarados en un formato de fácil acceso como XML; 2) la clasificación de los conceptos formalizados, a partir de la cual se puede inferir conocimiento, para lo cual la jerarquía de clases de la ontología no debe contener errores; 3) el tipo de conocimiento que debe responder la ontología, para lo cual la ontología debe responder a preguntas de competencia inherentes al dominio.

Otros aspectos tales como el uso del *“hash” (#) vs “slash” (/), etiquetas y comentarios, nodos en blanco y la completitud del lenguaje*; no fueron tenidos en cuenta en la evaluación, ya sea porque están enfocados en aclaraciones en lenguaje natural y en distintos idiomas sobre el uso y estilo de la ontología, o en la identificación de propiedades o nodos que no han sido considerados en el diseño porque no ha sido necesario; lo cual no influye en el despliegue de estrategias de recomendación.

A continuación, se describe cómo se evaluó cada aspecto, señalando los métodos utilizados y los resultados.

K.1 Declaración de URI

Según Vrandečić [39] las ontologías web no requieren que los nombres sean declarados, por lo tanto, es imposible para los razonadores discernir si `ex:Adress` es una nueva entidad o es solo un error tipográfico de `ex:Address`. Este problema se puede abordar exigiendo la declaración de nombres, lo cual permite chequear que todos los nombres usados estén declarados apropiadamente. Así, también se posibilita un análisis sintáctico (“*parsing*”) más eficiente de la ontología. Las declaraciones son axiomas, estableciendo no solo que un nombre existe sino también su tipo, esto es, que sea declarado como una clase, una instancia, un “*data type*”, “*object*” o “*annotation property*”. A continuación, se describe el método utilizado y los resultados.

K.1.1 Método: Chequear declaraciones de nombres

Con base en lo propuesto en [39], el método está constituido por las siguientes partes:

1. Extraer todas las URIs de la ontología.
2. Verificar si las URIs tienen un tipo declarado como los siguientes: instancias, “*Data type*”, “*Object*” o “*Annotation property*”.
3. Verificar si el tipo declarado es consistente con el uso. Si no es así, es posible la existencia de “*punning*” (la declaración de la misma URI para recursos de diferentes tipos) erróneo. Cabe aclarar que el “*punning*” es permitido en OWL 2, sin embargo algunas veces es introducido erróneamente (sin intención).

Los resultados de la evaluación son los siguientes:

- El 100% de las URIs de la ontología tienen declarado su tipo.
- El 100% de las declaraciones son acordes con su uso. De entre el total de axiomas, no fueron hallados algunos que ocasionaran inconsistencias con las declaraciones. Para apoyar este resultado, también fueron creados un grupo de instancias de las clases, a los cuales se les asignaron literales para sus “*Data Properties*”, y fueron vinculados de acuerdo a las “*Object Properties*”. Con la ayuda del razonador *Hermit* de *Protégé*, fueron analizados todos los elementos de la ontología para la búsqueda de inferencias erróneas que evidenciaran declaraciones erróneas. En conclusión, no existen declaraciones erróneas de las URIs.
- Aunque algunas clases e instancias comparten el mismo nombre entre diferentes ontologías, sus URIs se diferencian entre sí gracias a sus espacios de nombres (“*namespaces*”). Esto implica la no necesidad de analizar casos puntuales de “*punning*”. Todos estos casos se presentan en la Tabla K.1. Se indican los elementos de la ontología y sus correspondientes URIs. Aunque es necesario poner

K. Versión extendida de la evaluación de la ontología propuesta

atención en las URIs al momento de construir aplicaciones para referenciar los elementos correctamente, cabe aclarar que las clases e instancias de ontologías distintas con el mismo nombre han sido relacionados con las propiedades `owl:equivalentClass` y `owl:sameAs` respectivamente. Esto facilita la construcción de instancias y la definición de sus “*Data properties*” y “*Object properties*”, evitando inconsistencias.

Nombre del elemento	Tipo declarado	URI
Difficulty	Class	http://www.semanticweb.org/ontologies/2017/adaptCompModel#Difficulty
Difficulty	Class	http://www.semanticweb.org/ontologies/2017/adaptLOM#Difficulty
Language	Class	http://www.semanticweb.org/ontologies/2017/adaptCompModel#Language
Language	Class	http://www.semanticweb.org/ontologies/2017/adaptLOM#Language
UserRole	Class	http://www.semanticweb.org/ontologies/2017/adaptCompModel#UserRole
UserRole	Class	http://www.semanticweb.org/ontologies/2017/adaptLOM#UserRole
VeryLow	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#VeryLow
Low	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#Low
Medium	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#Medium
High	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#High
VeryHigh	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#VeryHigh
Difficult	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#Difficult
Easy	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#Easy
Medium difficulty	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#MediumDifficulty
Very difficult	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#VeryDifficult
Very easy	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#VeryEasy
Todos los códigos del estándar iso 639-1 para la identificación de lenguajes	Individual	La URI de cada instancias que representa a cada código tiene los siguientes sufijos: http://www.semanticweb.org/ontologies/2017/adaptCompModel# http://www.semanticweb.org/ontologies/2017/adaptLOM#Language
Author	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#author
Learner	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#Learner
Manager	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#Manager
Teacher	Individual	http://www.semanticweb.org/ontologies/2017/adaptCompModel#Teacher
Author	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#Author
Learner	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#Learner
Manager	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#eManager
Teacher	Individual	http://www.semanticweb.org/ontologies/2017/adaptLOM#Teacher

Tabla K.1 Clases e instancias con el mismo nombre entre ontologías. Fuente: propia.

K.2 Literales y tipos de datos

Un literal es un par compuesto por una representación léxica y un tipo de dato. El tipo de datos es dado por una URI que define la interpretación de la representación léxica. El estándar OWL requiere que todas las herramientas soporten `xsd:string` y `xsd:integer` (“XML Schema Definition”). OWL 2 extiende el número de tipos de datos considerablemente, añadiendo tipos de datos para números, texto, valores booleanos, datos binarios, URIs, instantes de tiempo y literales XML. También añade la posibilidad de restringir los literales por medio de facetas. Un error común en la definición de los tipos de datos es usar un prefijo de espacio de nombre (`xs:string` aparece 19927 veces y `xsd:string` 518 veces). Otros errores comunes incluyen errores de ortografía de la URI del tipo de dato (por ejemplo, olvidando el *hash* o escribiendo con mayúscula erróneamente el nombre local) o usando versiones obsoletas del esquema XML.

K.2.1 Método: Chequeo de literales y tipos de datos

El método de evaluación es el siguiente:

1. Verificar si en la ontología predominan los tipos de datos recomendados por la especificación OWL, esto es, `xsd:integer` y `xsd:string`. El uso de otros tipos deberían estar debidamente justificados.
2. Verificar si hay un tipo de dato personalizado. En caso afirmativo, corroborar que su uso está fuertemente soportado.
3. Verificar si todos los literales son sintácticamente válidos con respecto a su tipo de dato.

Los resultados de la evaluación son los siguientes:

- En cuanto a facetas, de los 55 “Data types” definidos en el modelo ontológico, el 100% están restringidos por tipos de datos en facetas. De ellos, 46 son `xsd:string`, 6 son `xsd:decimal`, 2 son `xsd:anyURI`, 1 es `xsd:integer`. Se observa que los tipos de datos recomendados han sido utilizados de forma justificada. El uso de `xsd:decimal` se justifica porque es el tipo de dato usado para especificar las duraciones de los contenidos y del aprendizaje de las competencias. El uso de `xsd:anyURI` se justifica porque es el mejor para especificar las ubicaciones de los contenidos y de los servicios de VoD.
- En cuanto a la declaración de los literales, el 100% son sintácticamente válidos con los tipos de datos indicados en 1. Esto cierra la puerta a posibles inconsistencias.

Este resultado ha sido corroborado por medio del uso del razonador *Hermit* en *Protegé*.

K.3 Validación XML

La web semántica fue creada como un grupo de estándares construidos unos sobre otros, e incorporando estándares existentes ya establecidos tales como las URI para identificadores o XML para la serialización. En un escenario ideal, se busca que, con la reutilización de esos estándares, las herramientas ampliamente desplegadas y la experticia construida no se perderían sino que serían más relevantes y en continuo uso. Sin embargo, esto no se ha dado en su totalidad.

Debido a las numerosas posibilidades de los grafos RDF que pueden ser expresados en XML, crear aplicaciones usando el conjunto de herramientas y experticia es a menudo ineficiente e irrazonablemente difícil. Esto es particularmente verdad para enfoques de evaluación XML típicas tales como la validación de documento. Se propone restringir RDF/XML con STD ("*XML-native Document Type Definitions*"), los documentos resultantes son acordes con archivos RDF, y entonces podrán ser usados por otras herramientas de la web semántica.

K.3.1 Método: Validar con respecto a XML

Los pasos para la validación son:

1. Serializar la ontología a un formato XML en caso de ser necesario.
2. Si fue necesaria la serialización, realizar una comparación entre la ontología original y la serializada en busca de datos perdidos (por ejemplo, instancias).
3. En caso de no ser necesaria la serialización, realizar una validación de acuerdo al estándar XML.

Los resultados son los siguientes:

- La ontología originalmente está serializada en un formato RDF/XML, por lo cual no ha sido necesaria la serialización en otro formato XML.
- Con la ayuda de la herramienta en línea *XML validator*²¹, se corroboró que la ontología es válida con respecto a XML. Entonces, la ontología puede ser compatible con herramientas XML útiles como convertidores a formatos JSON, HTML, CSV; u otros validadores. A futuro se espera realizar una prueba enfocada

²¹<http://www.xmlvalidation.com/index.php?id=1&L=0&target=/xmlvalidation/start.jsp;jsessionid=E9F659FA35AAB264EDD890D7C2D92CD0>

en verificar si la ontología puede ser soportada por otras aplicaciones después de un proceso de conversión.

K.4 Profundidad máxima de la jerarquía de conceptos

La *Profundidad Máxima de la Jerarquía de Conceptos* (TD) es definida como el camino más largo existente siguiendo las relaciones de herencia a través de la taxonomía. Esta definición lleva a un grupo de problemas. Primero, los ciclos en la jerarquía de clases pueden ocasionar una lectura errónea de la profundidad máxima, pudiendo ser inclusive infinita (∞) o cero (0). Claro está, este tipo de resultados no son útiles al momento de construir aplicaciones, por ejemplo, en los casos en los que es importante la similitud semántica entre dos conceptos.

Como solución, se puede usar una métrica llamada *Longitud Máxima de la Ruta de Subsunción* (SL). Para calcularla, primero es necesario obtener la tercera normalización de la ontología original, encontrando así una estructura alternativa que evite los ciclos. Después, medir su profundidad.

A partir de TD y SL se calcula la métrica *Claridad de la jerarquía de subsunción* (ET), definida por:

$$ET = \frac{TD}{SL} \quad (7.1).$$

Según el valor de ET, se deben tomar medidas para reparar la jerarquía. Esto se explica en el siguiente método.

K.4.1 Método: Claridad de la jerarquía de subsunción

El método define los siguientes pasos:

1. Medir la Profundidad Máxima de la Jerarquía de conceptos (TD).
2. Verificar si la ontología presenta ciclos.
3. En caso negativo, la jerarquía de conceptos está bien.
4. En caso positivo, realizar la tercera normalización de la jerarquía y medir su Longitud Máxima de la Ruta de Subsunción (SL).
5. Calcular ET.
6. Con respecto al valor de ET, pueden darse los siguientes casos:
 - Si $ET = 0$, la jerarquía puede estar bien (pueden haber otros problemas que no se pueden detectar con este método).
 - Si $ET < 1$, entonces algunas clases en la ontología han colapsado. En tal caso encontrar las clases colapsadas y reparar la jerarquía explícita.

K. Versión extendida de la evaluación de la ontología propuesta

- Si $ET > 1$, entonces una parte de las clases no ha sido explícita. En tal caso hallar qué parte hace falta y reparar la jerarquía

Para ejecutar el método, se hizo uso de la herramienta *OntoGraf* de *Protégé* para medir TD de forma sencilla y verificar si existen ciclos en la jerarquía del modelo. Una imagen tomada de *Protégé* se observa en la Figura K.1, en la cual todas las subsunciones entre las clases de la ontología (ontología integrada) están marcadas por las líneas azules.

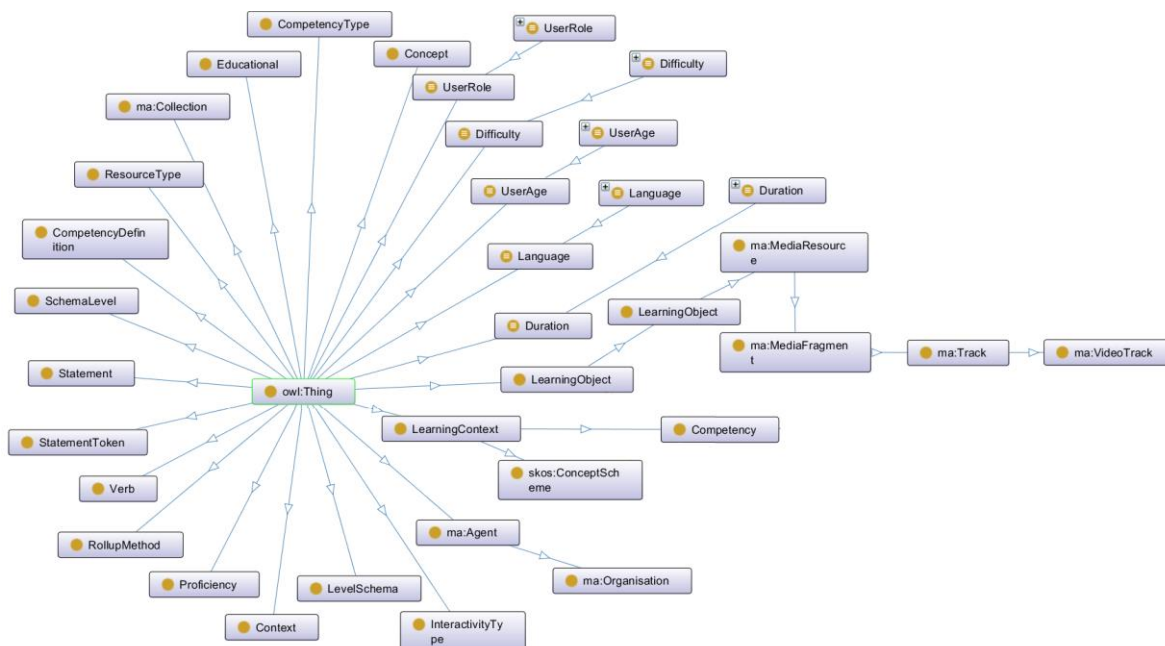


Figura K.1 Jerarquía de la ontología integrada. Fuente: propia.

De la figura se obtienen los siguientes resultados:

- La Profundidad Máxima de la Taxonomía de conceptos es TD= 6.
- La Jerarquía de conceptos del modelo no presenta ciclos, lo cual corrobora el valor de TD mencionado y que las relaciones de subsunción son claras y declaradas correctamente.

K.5 Preguntas de competencia formalizadas

Las preguntas de competencia describen el tipo de conocimiento que se supone debe responder la ontología. Estas preguntas son necesariamente formalizables en un lenguaje de consulta. Formalizar las consultas en lugar de escribirlas en lenguaje natural, y formalizar las respuestas esperadas también permite a un sistema chequear automáticamente si la ontología cumple los requerimientos establecidos con las preguntas de competencia.

K.5.1 Método: Chequear preguntas de competencia con relación a resultados

Este método consiste en:

1. Formalizar las preguntas de competencia como unas consultas SPARQL.
2. Escribir las respuestas esperadas como un resultado de consulta SPARQL, en XML o en JSON.
3. Realizar las consultas SPARQL y obtener los resultados.
4. Comparar el los resultados obtenidos y las respuestas esperados.

Para la evaluación, se tuvieron en cuenta las preguntas consideradas más importantes, esto es, las vinculadas con los aportes del presente trabajo. Específicamente, son preguntas relacionadas con el conocimiento formalizado en los mapas de competencias, en las relaciones entre contenidos y segmentos, en las relaciones entre segmentos y competencias, y en las relaciones entre competencias y mapas. Finalmente, para limitar el alcance de la evaluación, el conocimiento evaluado se centra en el extracto de la base de conocimiento mostrado en la Figura 4.18 (Sección 4.3).

Inicialmente se formalizan las preguntas de competencia en SPARQL, indicando su equivalente en lenguaje natural. Considerar los siguientes prefijos para cada ontología:

- pref1: IRI de la ontología propuesta.
- pref2: IRI de *OntoRCM*.
- pref3: IRI de la ontología de inferencias derivada de la ontología propuesta (esta puede obtenerse usando Protege).
- pref4: IRI de “*Ontology for Media Resources*” adaptada.
- ma: IRI de “*Ontology for Media Resources*”.

1. ¿Qué nodos hacen parte del mapa de competencias *MapaCompetencias1*?

```
SELECT ?Nodes
WHERE { pref1:MapaCompetencias1 pref2:hasNode ?Nodes }
```

Los resultados esperados son:

Nodes
Nodo1
Nodo2
Nodo3
Nodo4

2. ¿Dado un nodo (*Nodo3* en este caso), qué otros nodos posibilita? Esta consulta se realiza en la ontología de inferencias derivada de la ontología propuesta.

K. Versión extendida de la evaluación de la ontología propuesta

```
SELECT ?Nodes
WHERE { pref1:Nodo3 pref2:enables ?Nodes }
```

Los resultados esperados son:

Nodes
Nodo2
Nodo1

3. ¿Dado un nodo (*Nodo1* en este caso), de qué otros nodos depende? Esta consulta también se realiza en la ontología de inferencia derivada de ontología propuesta.

```
SELECT ?Nodes
WHERE { pref1:Nodo1 pref2:dependsOn ?Nodes }
```

Los resultados esperados son:

Nodes
Nodo2
Nodo3
Nodo4

4. ¿Dado un nodo (*Nodo2*), qué competencia(s) están relacionadas con él?

```
SELECT ?Competencias
WHERE { pref1:Nodo2 pref1:hasCompetencyRef ?Competencias }
```

Los resultados esperados son:

Competencias
Calculo el área de algunas figuras geométricas

5. ¿Dada una competencia (competencia *Calculo el área de algunas figuras geométricas* en este caso), con qué nodos se relaciona?

```
SELECT ?Nodes
WHERE { pref1: Calculo el área del algunas figuras geométricas
pref1:isCompetencyRefOf ?Competencias }
```

Los resultados esperados son:

Nodes
Nodo1

6. ¿Qué segmentos hacen parte de un contenido (*Figuras geométricas*, en este caso)?

```
SELECT ?Segments
WHERE { pref1:Figuras geométricas pref4:hasVideoTrack ?Segments }
```

Esta consulta también se puede realizar en la ontología de inferencia de la siguiente forma:

```
SELECT ?Segments
WHERE { pref1: Figuras geométricas pref5:hasFragment ?Segments }
```

7. En ambos casos el resultado esperado es:

Segments
El círculo
El cuadrado

8. ¿Dado un segmento (L18-V1-S2 en este caso), a qué competencias hace referencia?

```
SELECT ?Competencias
WHERE { pref1:El círculo pref1:CompRef ?Competencias }
```

El resultado esperado es:

Competencias
Identifico algunas figuras geométricas
Reconozco el círculo

9. ¿Dada una competencia (*Aplico la fórmula del área del círculo* en este caso), qué segmentos están vinculados a ella?

Esta consulta es realizada en la ontología de inferencias (ya que `:CompRef` es simétrica).

```
SELECT ?Segments
WHERE { pref1: Aplico la fórmula del área del círculo en este caso
pref3:CompRef ?Segments }
```

El resultado esperado es:

Segments
El área del círculo

Con base en las consultas SPARQL definidas, se realizó la verificación de los resultados esperados. El resultado es que la ontología respondió 100% de forma correcta a las preguntas de competencia formalizadas. Esto se debe principalmente a la definición justificada y al refinamiento previo del esquema de metadatos para la descripción de competencias y contenidos de VoD, por lo cual no fue necesaria la evaluación y mejora en más iteraciones que la presentada aquí.

Así, se concluye que la ontología provee el conocimiento (esto es, aseverado e inferido) que se ha establecido como el de mayor relevancia en el contexto del trabajo, que además se presume clave en el proceso de obtención de recomendaciones. Esto se consiguió porque:

- La ontología se construyó a partir de esquemas de metadatos, lo cual permitió reconocer la información del dominio que debía ser formalizada, incluyendo las jerarquías de clases, instancias, propiedades y tipos de datos.

K. Versión extendida de la evaluación de la ontología propuesta

- La ontología fue construida usando la herramienta *Protégé*, la cual asigna URI de forma consistente, incorpora razonadores que identifican problemas de consistencia, y serializa la ontología en un documento XML.

Anexo L

Versión extendida de la evaluación del método computacional propuesto

Este anexo muestra una versión extendida de la evaluación del método computacional introducida en la Sección 5.3.

Para evaluar el método computacional para la evaluación de competencias se llevaron a cabo una serie de experimentos enfocados en responder las siguientes preguntas de investigación:

1. ¿Cuál es la precisión del método computacional propuesto en comparación con un *Modelo Basado en el Principio de Extensión* (MBPE)?
2. ¿Qué ventajas ofrece el método computacional en términos de tiempo de ejecución en comparación con el MBPE?
3. ¿Qué ventajas ofrece el método computacional propuesto en cuanto a consumo de memoria con respecto al MBPE?

La evaluación se realizó con respecto al MBPE porque es el modelo computacional de referencia usado en la computación lingüística [66].

Para responder las preguntas de investigación planteadas, se construyó una herramienta Java denominada *CMA-Tool*²² (“*Competency Map Assessment Tool*”), la cual es un aporte adicional de la presente investigación. La herramienta fue construida a partir de la necesidad de medir las ventajas o desventajas provistas por el método computacional con respecto a un modelo computacional lingüístico de referencia como el MBPE. *CMA-Tool* ofrece una interfaz gráfica que permite configurar la estructura de los mapas de competencias usando la especificación *Formato de Grafos* JSON [108]. Para configurar la operación de la herramienta, se introduce el número de estudiantes y posteriormente, se definen uno a uno los niveles de competencias de los estudiantes en cada una de las actividades de evaluación formalizadas en el mapa. Finalmente, la herramienta ejecuta el algoritmo DFS-CA propuesto (Sección 5.2.2) propuesto para ofrecer al usuario los niveles de competencia computados para cada una de las

²² El código de CMA-Tool puede ser descargado de: <https://mega.nz/#F!CvhxzSh!tRoEGEJBHLBdCVin7y2b8Q>

competencias del mapa, usando tanto el método computacional propuesto como el MBPE. Además, también para los dos modelos, la herramienta despliega dos gráficos que muestran el tiempo de ejecución y el consumo de memoria.

L.1 Prototipo del método computacional

CMA-Tool ofrece dos modos de operación para computar los NC de las competencias: el *Modo 1* está basado en el método computacional propuesto y el *Modo 2* está basado en el MBPE (Sección 2.1.7). Su creación tiene dos fines principalmente: 1) facilitar la evaluación de mapas de competencias usando términos lingüísticos; 2) proveer los tiempos de ejecución y consumo de memoria usados durante la evaluación de los mapas.

La herramienta utiliza la especificación *Formato de Grafos JSON* para formalización de los mapas de competencias [108]. La Figura L.1 presenta la interfaz principal de la herramienta. Al seleccionar la opción *Configurar Mapa* del menú superior, se visualizan tres áreas principales. El área (a) presenta un formato JSON por defecto regido por el *Formato de Grafos JSON* para la formalización de los mapas de competencias. La razón de usar tal formato está en que se posibilita, con el uso de librerías Java tales como *Jgraph* [135], *JsonPath* [136] y *JsonSimple* [137], el recorrido organizado y sistemático de los grafos implícito en el algoritmo propuesto DFS-CA, el cual define la forma en cómo opera el método computacional para la evaluación de competencias propuesto. Al respecto, no fue posible hallar librerías que permitan hacer este tipo de procesos con grafos RDF (esto es, el formato de los grafos formalizados en una ontología). Sin embargo, es importante señalar que el uso del *Formato de Grafos JSON* fue una solución válida para ejecutar la evaluación de precisión aquí presentada porque permite describir el mismo conocimiento inherente a los mapas de competencias que, de hecho, fue formalizado en la ontología para el modelado de competencias propuesto en el Capítulo 4 (Sección 4.1.3). Entonces, queda por desarrollar un proceso capaz de adaptar el formato RDF al del *Formato de Grafos JSON* o que directamente pueda ejecutar el recorrido de los grafos RDF. Cualquiera de los dos casos está fuera del alcance del presente trabajo.

En el área (a) se presenta un mapa por defecto, el cual sirve de modelo para el usuario para que pueda hacer las adaptaciones necesarias para construir la estructura del mapa de competencias deseado, esto es, copiar y pegar fragmentos, crear nuevos nodos y aristas, configurar pesos, identificadores, tipos de nodos (competencia o actividad de evaluación) y etiquetas. Seleccionando el botón *Configurar* se presenta el grafo resultante en el área (b), de tal forma que el usuario pueda visualizar si el resultado corresponde al grafo esperado. El área (c) presenta un texto que lista los

identificadores, etiquetas y tipos de nodos creados, así como el origen y objetivo de las aristas definidas.

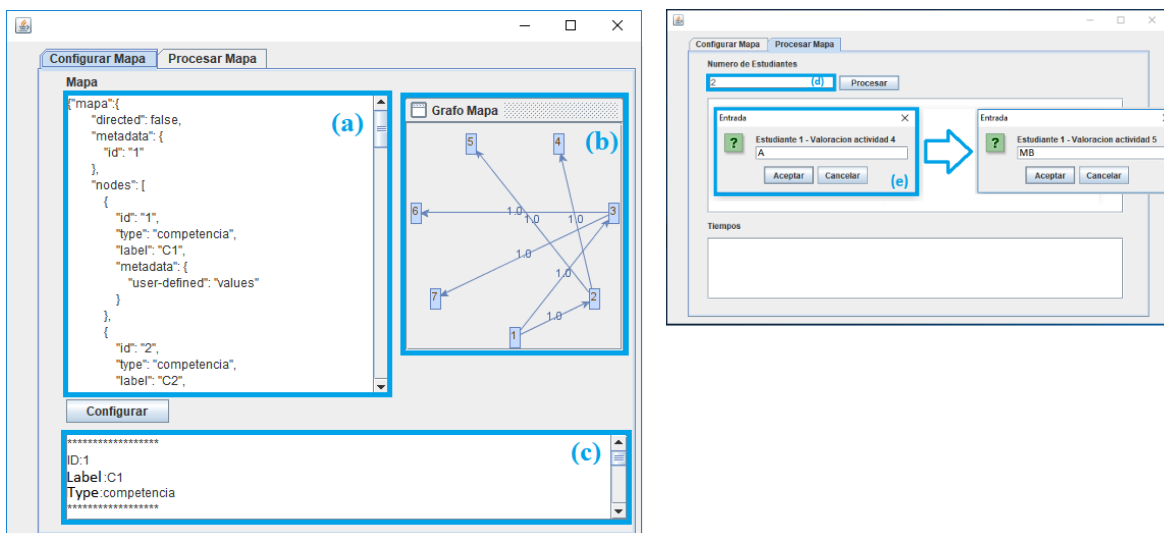


Figura L.1 Interfaz principal de la herramienta para la evaluación de competencias.

Fuente: propia.

Una vez la estructura del mapa ha sido establecida, mediante la opción *Procesar Mapa* del menú superior, el usuario puede introducir en el área (d) el número de estudiantes a los cuales les serán calculados sus NC. Al seleccionar el botón *Procesar* se despliega una ventana, marcada como área (e), en la cual se solicita el ingreso de los NC alcanzados por el estudiante en cada una de las AE. Las solicitudes se van realizando por cada estudiante y AE. Al finalizar el ingreso de toda la información, la herramienta despliega las ventanas mostradas en la Figura L.2. El área (f) muestra los resultados de la evaluación de acuerdo con el método propuesto, denominado en la herramienta como *Modo 1* y de acuerdo con el MBPE, denominado *Modo 2*; detallando el identificador del nodo, el tipo de nodo, el VAN, el par de *2-tuplas* resultantes y el NC lingüístico para cada uno de los estudiantes. Adicionalmente, se despliegan dos gráficas en las áreas (h) e (i) que detallan los tiempos de ejecución en microsegundos y el consumo de memoria computacional en Mega Bytes respectivamente, para cada uno de los estudiantes y cada uno de los modos. Los tiempos y consumo de memoria son presentados en texto en el área (g), junto a la razón entre los tiempos del *Modo 2* y el *Modo 1*. En conclusión, la herramienta no sólo facilita el cómputo de los NC de las competencias del mapa, sino que ofrece la información necesaria para evaluar variables como el tiempo de ejecución y el consumo de memoria.

Una vista funcional, una descripción de los componentes desarrollados y una vista dinámica del prototipo se presenta con mayor detalle en el Anexo O (Sección O.1).

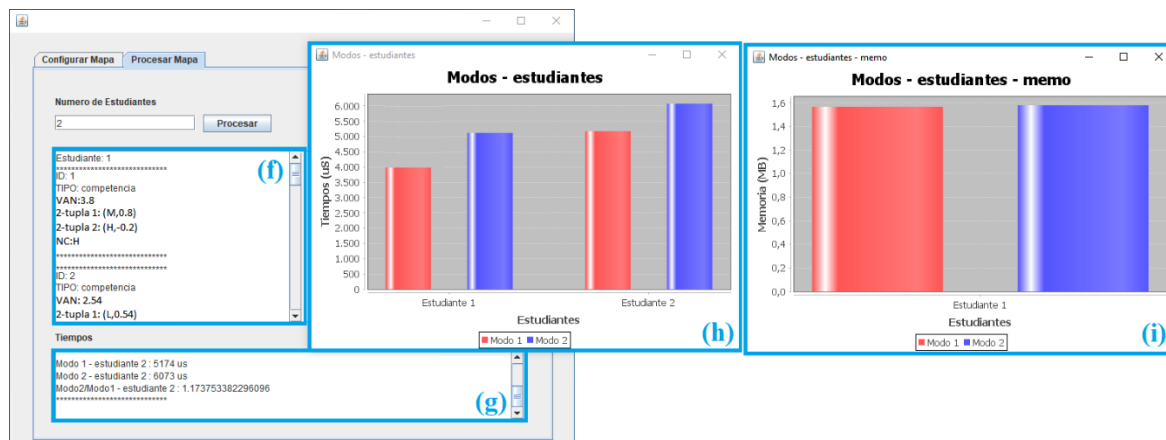


Figura L.2 Interfaces que presentan los resultados de la evaluación de competencias.
Fuente: propia.

L.2 Colección de datos

Para la evaluación se coleccionaron los datos de 85 mapas de competencias de procesos de aprendizaje reales, los cuales fueron construidos y provistos por un grupo de docentes basándose en estándares educativos de Matemáticas²³ [53]. A fin de explicar de forma clara los experimentos y resultados alcanzados, vale la pena considerar la diferencia entre las estructuras de los mapas, la cual está constituida por competencias, actividades de evaluación, sus relaciones y pesos; mientras que el mapa es el conjunto estructura-niveles de competencia. En consecuencia, una estructura puede ser aplicada a un grupo de estudiantes, mientras que el mapa de competencias es individual. Así las cosas, los 85 mapas de competencias obedecen a 17 estructuras diferentes. Además, los 85 mapas involucran 1870 niveles de competencias alcanzados por los estudiantes en las actividades. Estos NC son usados en los experimentos para computar los NC en las competencias. Entre los datos coleccionados, están los NC provistos por los docentes, los cuales fueron comparados con los computados por el prototipo para evaluar la efectividad del método computacional propuesto. La Tabla L.1 resume la estadística de la colección de datos.

Las descripciones de las estructuras de los mapas fueron coleccionadas por medio de un formato como el mostrado en el ejemplo de la Figura L.3. En él, las estructuras, las competencias y las actividades de evaluación tienen un identificador. Además, los elementos de una columna, ya sean competencias o actividades, son subelementos de los que están en la columna de la izquierda. Teniendo en cuenta el ejemplo de la Figura L.3, las competencias SS-PN-061 y SS-PN-062 son subcompetencias de SS-

²³ Las estructuras de los mapas de competencias y los mapas de competencias para experimentación se pueden descargar de: <https://mega.nz/#F!i2xS1BbD!qRv-bhBVUMpu46FRDIdqwg>

L. Versión extendida de la evaluación del método computacional propuesto

PN-06. Asimismo, las actividades de evaluación AE252, AE253, AE254 y AE255 son subelementos de la competencia SS-PN-061.

Datos	
Mapas de competencias	85
Estructuras de mapas	17
Niveles de competencia de actividades de evaluación	1870
Niveles de competencia en las competencias	810

Tabla L.1 Estadística de la colección de datos usada para la evaluación del método computacional para la evaluación de competencias. Fuente: propia.

Id mapa	Competencias	Competencias	Actividades de evaluación
M10-SS	SS-PN-06: Justifico procedimientos aritméticos utilizando las relaciones y propiedades de las operaciones.	SS-PN-061: Aplico relaciones y propiedades de las operaciones aritméticas en ejercicios	AE252
			AE253
			AE254
			AE255
		SS-PN-062: Recuerdo las relaciones y propiedades de las operaciones aritméticas	AE256
			AE257
			AE258

Figura L.3 Formato usado para la descripción de la estructura de los mapas. Fuente: propia.

Para coleccionar los mapas de competencias de los estudiantes, se utilizó un formato como el presentado en la Figura L.4. Se observa cómo se hace referencia a estructuras de mapas, competencias actividades de evaluación por medio de los identificadores. Se definen tanto los pesos de las relaciones entre elementos y subelementos, como los NC alcanzados.

Id estructura: M10-SS

Id estudiante: 1-SS

Actividades de Evaluación	Peso	NC	Competencia	Peso	NC	Competencia	NC
AE252	VH	M	SS-PN-061	VH	M	SS-PN-06	M
AE253	VH	M					
AE254	H	M					
AE255	H	M					
AE256	H	M	SS-PN-062	H	M		
AE257	H	M					
AE258	VH	M					

Figura L.4 Formato usado para la descripción de los mapas de competencias. Fuente: propia.

L.3 Métricas

Para dar respuesta a la primera pregunta de investigación fue usada la métrica definida en la expresión (L.1), la cual se define como el porcentaje de NC en las competencias computadas por el prototipo que son iguales a las provistas por los docentes. También, puede ser considerada como el porcentaje de aciertos provistos por el método.

$$A = \frac{N_{\text{aciertos}}}{T} * 100\%, \quad (\text{L.1})$$

donde N_{aciertos} es el total de aciertos y T es el total de NC evaluados. Con respecto a la segunda y tercera pregunta de investigación, el tiempo de ejecución y el consumo de memoria son variables que dependen de varios factores, tales como los recursos propios del computador o las tareas paralelas realizadas por él al momento de realizar los experimentos. Con el propósito de reducir la variabilidad de las medidas, se ha calculado un valor promedio t_j y MEM_j en cien medidas para cada experimento j , de acuerdo con:

$$t_j = \frac{\sum_{i=1}^{100} t_{ij}}{100}, \quad (\text{L.2})$$

$$MEM_j = \frac{\sum_{i=1}^{100} CPU_{ij}}{100}, \quad (\text{L.3})$$

donde t_{ij} y MEM_{ij} es la i -ésima medida de tiempo y consumo de memoria respectivamente, tomadas para el experimento j .

L.4 Fases de la evaluación

Las fases de la evaluación ejecutadas fueron las siguientes:

1. Cada estructura de los mapas fue adaptada a la especificación Formato de Grafos JSON y registrada en el prototipo.
2. El número de estudiantes a evaluar fue especificado en el prototipo.
3. Los NC alcanzados por los estudiantes en las AE fueron introducidos en la herramienta.
4. Se computaron los NC de las 810 competencias por medio de la herramienta usando los modos 1 y 2.
5. Con el propósito de medir la efectividad de la propuesta y el modelo basado en el MBPE, los NC computados por la aplicación en la etapa 4 fueron comparados con los provistos por los docentes. Posteriormente, el porcentaje de aciertos fue medido mediante la expresión (L.1).
6. Con el propósito de conocer las ventajas del método propuesto en términos de tiempo de ejecución y consumo de memoria en comparación con el MBPE, todos los mapas fueron computados 100 veces para medir el tiempo de ejecución y consumo de memoria promedio de cada estructura y modelo mediante las expresiones (L.2) y (L.3). Cabe mencionar que las medidas fueron tomadas para los métodos de del prototipo que están directamente relacionados con los procesos propios de cada modelo.

7. Se realizó un estudio para conocer si los resultados de efectividad del método propuesto son estadísticamente significativos con respecto a los obtenidos por el MBPE.

L.5 Resultados

A continuación, se presentan los resultados de efectividad, tiempo de ejecución, consumo de memoria y de significancia estadística alcanzados en la evaluación.

L.5.1 Resultados de precisión

La Tabla L.2 presenta los resultados de precisión obtenidos en la etapa 5 de los experimentos. Del análisis de los datos se concluye que, del total de competencias evaluadas, el método propuesto fue preciso en el 99.4% de los casos, mientras que el MBPE lo fue en el 97.8%. Dado que la diferencia en los resultados es aparentemente pequeña, se hizo una prueba de significancia estadística para concluir si es así o no. Para esto, se considera como hipótesis nula (H_0) *no hay diferencia significativa entre los resultados de precisión de los dos modelos*; y como hipótesis alternativa (H_a) *existe diferencia significativa entre los resultados de precisión de los dos modelos*. Además, se tuvieron en cuenta medidas como las probabilidades promedio entre los enfoques $p = \frac{A_1 + A_2}{2} = 0,986$, donde $A_1 = 0,994$ y $A_2 = 0,978$ son las probabilidades de acierto de cada modelo; un nivel de significancia $f = 0.5$; valores estadísticos críticos en la distribución normal $Z_{\alpha=0.05} = 1.96$, y el error estándar $es = \sqrt{(p(1-p)(2/T))}$, donde $T = 810$ es el total de medidas realizadas. Se calcula que $p = 0.986$ y $es = 0.00583$. Para concluir acerca de las hipótesis, se comprueba si $|A_1 - A_2| > se * Z_{\alpha=0.05}$. Al reemplazar los datos, se da que $|A_1 - A_2| = 0.0148$ es mayor que $es * Z_{\alpha=0.05} = 0.0114$, por lo cual se da como cierta la hipótesis alternativa, señalando que la diferencia de precisión entre el método computacional propuesto y el MBPE es estadísticamente significativa. Esto se debe básicamente a que el tamaño de la muestra (810 niveles de competencia) hace que el error estándar inherente a la experimentación se reduzca. De acuerdo a [111], dado que el error estándar ($se = 0.00583$) es menor que 0,01, entonces se concluye que el método propuesto tiene una precisión significativa sobre el MBPE con una seguridad inherente del 99%.

Estos resultados demuestran que, en un entorno educativo real, tanto el método computacional como el prototipo *CMA-Tool* propuestos pueden ser utilizados de forma confiable por los docentes. Aunque el estudio se centró en competencias en matemáticas, cabe destacar que el método computacional no depende de la sintáctica y semántica de las competencias, sino que se centra exclusivamente en la estructura

de los mapas, cuya calidad depende exclusivamente de los docentes. En consecuencia, los resultados y análisis mostrados aquí pueden ser extrapolados a otras áreas diferentes a las matemáticas.

Estructura	Competencias por estructura	T	Método propuesto		MBPE	
			$N_{aciertos}$	$A(\%)$	$N_{aciertos}$	$A(\%)$
1	8	40	39	97.5	36	90
2	6	30	28	93.33	27	90
3	15	75	72	96	71	94.66
4	7	35	34	97.14	32	91.42
5	11	55	54	98.18	53	96.36
6	3	15	15	100	15	100
7	11	55	55	100	55	100
8	10	50	50	100	50	100
9	24	120	119	99.16	119	99.16
10	18	90	89	98.88	88	97.77
11	3	15	15	100	15	100
12	10	50	49	98	49	98
13	3	15	15	100	15	100
14	9	45	44	97.77	43	95.55
15	11	55	55	100	54	98.18
16	8	40	40	100	39	97.5
17	5	25	25	100	25	100
Total	149	810	798	99.4	787	97.8

Tabla L.2 Precisión del método para la evaluación de competencias propuesto y del MBPE para cada estructura de los mapas de competencias. Fuente: propia.

L.5.2 Resultados de tiempo de ejecución y consumo de memoria

Dado que el método computacional propuesto ha sido definido a través del algoritmo propuesto (DFS-CA), conviene realizar un análisis de los recursos computacionales usados. Esto para verificar si su implementación software es viable en casos reales. Con respecto a los tiempos de ejecución, la Figura L.5 muestra los resultados obtenidos (en microsegundos) en la etapa 6 de los experimentos. Se observa que, en comparación con el MBPE, el método propuesto presentó menores tiempos de ejecución en el 100% de los casos. En promedio, el método propuesto consumió un tiempo de $51,004 \mu s$ por mapa; mientras que el MBPE consumió uno de $64,135 \mu s$. Así las cosas, la diferencia en tiempo de ejecución es favorable para la propuesta en $13,13 \mu s$.

L. Versión extendida de la evaluación del método computacional propuesto

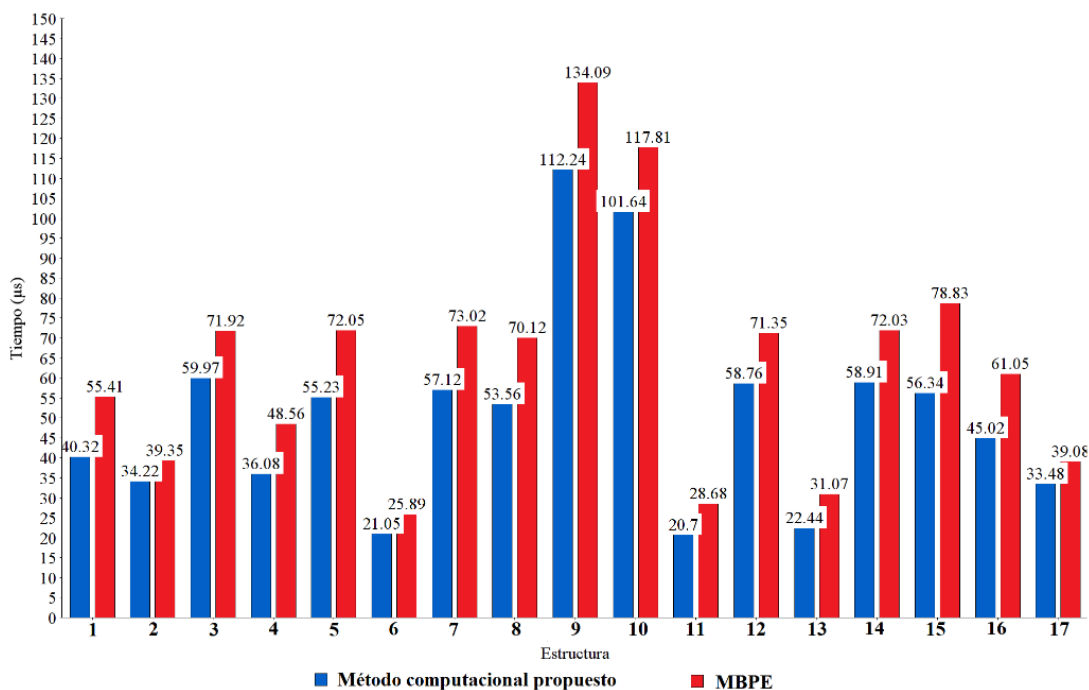


Figura L.5 Tiempos de ejecución del modelo propuesto y el MBPE para cada una de las estructuras de los mapas. Fuente: propia.

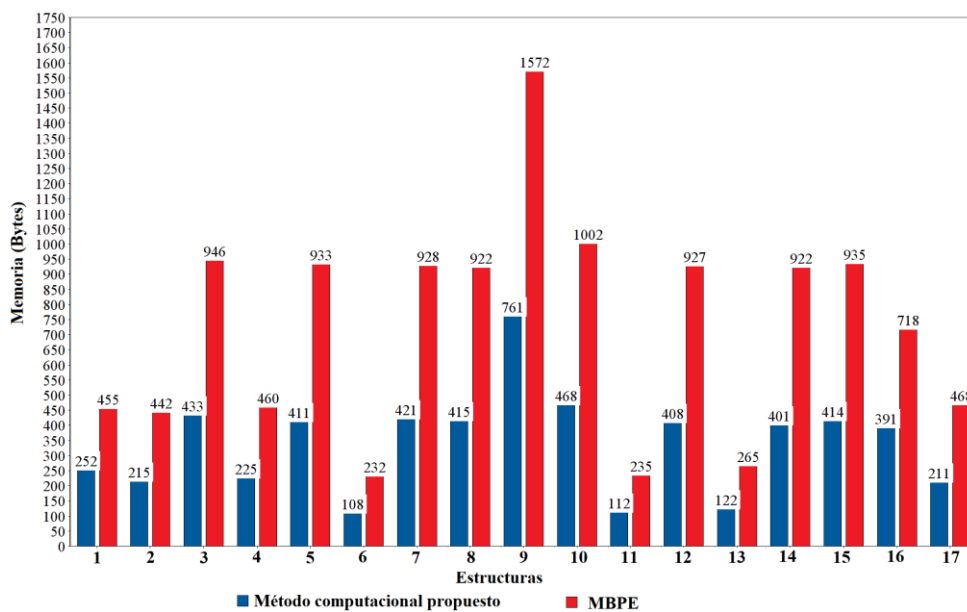


Figura L.6 Consumo de memoria del modelo propuesto y el MBPE para cada una de las estructuras de los mapas. Fuente: propia.

Algo similar ocurre con las medidas de memoria (en Bytes) presentadas en la Figura L.6, siendo las del método propuesto menores que la mitad de las medidas del MBPE en el 100% de los casos. En promedio, el método propuesto consumió 339,294 Bytes por mapa; mientras que el MBPE consumió 727,176 Bytes. Entonces, el método

propuesto consumió en promedio 387,882 Bytes menos que el MBPE. Esto se debe especialmente a la simpleza del enfoque de *2-tuplas* y a que JSON es un formato liviano. Los resultados indican que tanto el método computacional como el prototipo *CMA-Tool* propuestos son viables para ser integrados en plataformas de aprendizaje electrónico, servicios web o aplicaciones de escritorio.

Anexo M

Pseudoalgoritmos propuestos

Este anexo presenta los pseudoalgoritmos de los algoritmos propuestos en el trabajo usando la nomenclatura usada en [34].

M.1 DFS-CA: “*Depth First Search-Competency Assessment*”

Sea G el grafo que formaliza la estructura del mapa de competencias, el cual se compone de cada nodo x_i ; P_u el perfil del estudiante U que registra los NC alcanzados por él en las actividades de evaluación; p_i es el conjunto conformado por los pesos de las aristas entre un nodo x_i y sus nodos hijo; y nc_i el conjunto conformado por los NC de los nodos hijo del nodo x_i ; DFS-CA (“*Depth First Search-Competency Assessment*”), el cual es un aporte de la investigación, se ejecuta según el pseudoalgoritmo mostrado en el Algoritmo M.1.

Algoritmo M.1. Depth First Search-Competency Assessment. Fuente: propia.

*Entradas: $G \rightarrow$ grafo que formaliza la estructura del mapa de competencias

$P_u \rightarrow$ Perfil de usuario de U

*Salida: $CNC_u \rightarrow$ Conjunto de niveles de competencia computados por el método

```
DFS-CA( $G, P_u$ ) {
  Crear  $CNC_u$ ;
  Para cada vértice  $x_i \in G$  {
    estado[ $x_i$ ]  $\leftarrow$  NO_VISITADO;
  }
  DFS-CA_visitar(raíz( $G$ ),  $CNC_u$ );
}
```

```

DFS-CA_visitar( $x_i, CNC_u$ );
  estado[ $x_i$ ] ← VISITADO;
  Si (no existe  $nc_i$  y  $p_i$ ) Y (tipo[ $x_i$ ]=competencia) Entonces
    Crear  $nc_i$ ;
    Crear  $p_i$ ;
  Para cada  $x_{ji}$  adyacente  $x_i$  {
    Si (estado[ $x_{ji}$ ]=NO_VISITADO) Y (tipo[ $x_{ji}$ ]=actividad de
    evaluación) Entonces
       $P_u$ .consultar( $NC_{ji}$ );
       $G$ .consultar( $w_{ij}$ );
       $nc_i$ .añadir( $NC_{ji}$ );
       $p_i$ .añadir( $w_{ij}$ );
    Si (estado[ $x_{ji}$ ]=NO_VISITADO) Y (tipo[ $x_{ji}$ ]=competencia) Entonces
      DFS-CA_visitar( $x_{ji}$ );
       $P_u$ .consultar( $NC_{ji}$ );
       $G$ .consultar( $w_{ij}$ );
       $nc_i$ .añadir( $NC_{ji}$ );
       $p_i$ .añadir( $w_{ij}$ );
  }
   $NC_i = \Delta \left( \frac{\sum_{j=1}^n \beta_{ij} * w_{ij}}{\sum_{j=1}^n w_{ij}} \right)$ ;
   $CNC_u$ .añadir( $NC_i$ );
   $nc_{padre[x_i]}$ .añadir( $NC_i$ );
}

```

M.2 NSD: “Node Sequence Discovery”

Sea G la base de conocimiento, $GC = \{C_1, C_2, \dots, C_i, \dots, C_n\}$ el conjunto de instancias de competencias formalizadas en G y $PS = \{PS_1, PS_2, \dots, PS_j, \dots, PS_4\}$ el conjunto de secuencias de propiedades y $PS_j = \{P_1^j, \dots, P_i^j, P_n^j\}$ una secuencia de propiedades, el algoritmo para el descubrimiento de secuencias de nodos NSD (“Node Sequence Discovery” por su significado en inglés), el cual es un aporte de la investigación, se ejecuta de acuerdo al pseudoalgoritmo mostrado en el Algoritmo M.2.

Algoritmo M.2. Node Sequence Discovery. Fuente: propia.

* Entradas: $G \rightarrow$ Base de conocimiento

M. Pseudoalgoritmos propuestos

$GC \rightarrow$ Conjunto de todas las competencias formalizadas en la base de conocimiento

$PS \rightarrow$ Conjunto de secuencias de propiedades

* Salida: $NS_i \rightarrow$ Conjunto de secuencias de nodos para la competencia C_i

```

NSD(G,GC,PS) {
  Para cada  $C_i \in GC$  {
    Para cada  $PS_j \in PS$  {
      Si  $|PS_j| = 2$  Entonces
        Consultar en G el conjunto  $N_1$  de instancias  $x_i$  tal que
           $(C_i, p_1^j, x_i)$ ;
          Para cada  $x_i \in N_1$  {
            Consultar en G el conjunto  $N_2$  de instancias  $y_i$  tal
              que  $(x_i, p_2^j, y_i)$ ;
              Para cada  $y_i \in N_2$  {
                 $ns_i$ .añadir  $(C_i, x_i, y_i)$  ;
                 $CNS$ .añadir  $(ps_i)$  ;
              }
            }
          }
        Si  $|PS_j| = 4$  Entonces
          Consultar en G el conjunto  $N_1$  de instancias  $x_i$  tal que
             $(C_i, p_1^j, x_i)$ ;
            Para cada  $x_i \in N_1$  {
              Consultar en G el conjunto  $N_2$  de instancias  $y_i$  tal
                que  $(x_i, p_2^j, y_i)$ ;
                Para cada  $y_i \in N_2$  {
                  Consultar en G el conjunto  $N_3$  de instancias
                     $w_i$  tal que  $(y_i, p_3^j, w_i)$ ;
                    Para cada  $w_i \in N_3$  {
                      Consultar en G el conjunto  $N_4$  de
                        instancias  $v_i$  tal que  $(w_i, p_4^j, v_i)$ ;
                      Para cada  $v_i \in N_4$  {
                         $ns_i$ .añadir  $(C_i, x_i, y_i, w_i, v_i)$  ;
                         $NS_i$ .añadir  $(ps_i)$  ;
                      }
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}

```

M.3 RCS: “Relevant Competencies Search”

Sea U un estudiante con perfil P_U , $LC_U = \{c_1, c_2, \dots, c_i, \dots, c_n\}$ es el conjunto de competencias para las que U tiene $DoNC_{ui} > GN_o$, se definen los siguientes conjuntos: 1) el conjunto de N asociaciones para la competencia c_i después del proceso de filtrado se define como $TopNA_{CiU} = \{TopA_1^{CiU}, TopA_2^{CiU}, \dots, TopA_n^{CiU} \dots, TopA_N^{CiU}\}$, donde $TopA_n^{CiU}$ es una asociación semántica tal que $TopA_n^{CiU} = p - patAssociated(c_i, y_u^{Ci})$; 2) el conjunto de pesos totales de las N asociaciones de la competencia c_i es $WA_{CiU} = \{W_1^{CiU}, W_2^{CiU}, \dots, W_n^{CiU}, \dots, W_N^{CiU}\}$, donde W_n^{CiU} es el peso de la n -ésima asociación de la competencia c_i . Con el propósito de obtener una lista de recomendaciones para la competencia c_i , el SR extrae las competencias y_u^{Ci} de las N asociaciones cuyo origen corresponde a c_i para cada estudiante U , de acuerdo con el pseudo-algoritmo definido en el Algoritmo M.3, denominado RCS (“Relevant Competencies Search” por su significado en inglés), el cual es un aporte de la investigación.

Algoritmo M.3. Relevant Competencies Search. Fuente: propia.

* Entradas: $LC_U \rightarrow$ conjunto de competencias para las cuales U tiene un $GNC_{ui} \geq GN_o$

$TopNA_{CiU} \rightarrow$ conjunto de las N asociaciones para la competencia c_i

* Salida: $T_{CiU} \rightarrow$ conjunto de competencias relevantes para la c_i para el usuario U

```
RCS( $LC_U, TopNA_{CiU}$ ) {
  Para cada  $c_i \in LC_U$  {
    Para cada  $TopA_n^{CiU} \in TopNA_{CiU}$  {
      Extraer  $y_u^{Ci}$ ;
       $T_{CiU} \cdot \text{añadir}(y_u^{Ci})$ ;
    }
  }
}
```

M.4 VTR: “Video Tracks Ranking”

Considerando los conjuntos $LC_U = \{c_1, c_2, \dots, c_i, \dots, c_n\}$, $T_{CiU} = \{y_1^{Ci}, y_2^{Ci}, \dots, y_n^{Ci}, \dots, y_N^{Ci}\}$ y $WA_{CiU} = \{W_1^{CiU}, W_2^{CiU}, \dots, W_n^{CiU}, \dots, W_N^{CiU}\}$ definidos en el algoritmo RCS (Algoritmo

M. Pseudoalgoritmos propuestos

M.3), y sea $VT_{CiU} = \{VT_1^{CiU}, VT_2^{CiU}, \dots, VT_n^{CiU}, \dots, VT_N^{CiU}\}$ el conjunto de segmentos relacionados con las competencias relevantes descubiertas para la competencia c_i tal que $DoNC_{ui} > GN_o$ para el estudiante U , y $CVT_n = \{c_1, c_2, \dots, c_m, \dots, c_M\}$ es el conjunto de competencias que pertenecen a T_{CiU} vinculadas en la base de conocimiento con el segmento de video VT_n^{CiU} , la ponderación del n-ésimo segmento de video es computada por:

$$WVT_n^{CiU} = \frac{\sum_{m=1}^M W_n^{CiU}(c_m)}{M},$$

donde $W_n^{CiU}(c_m)$ es el peso de la competencia relevante $c_m \in CVT_n$ correspondiente en el conjunto VT_{CiU} y $M = |CVT_n|$. El pseudo-algoritmo mostrado en el Algoritmo M.4, describe el funcionamiento de VTR (“*Video Tracks Ranking*”), el cual es un aporte de la investigación.

Algoritmo M.4. Video Tracks Ranking. Fuente: propia.

* Entradas: $G \rightarrow$ base de conocimiento

$LC_U \rightarrow$ Conjunto de competencias c_i en las que U registra $DoNC_{ui} > GN_o$

$T_{CiU} \rightarrow$ Conjunto de competencias relevantes para U con relación a c_i

$WA_{CiU} \rightarrow$ Conjunto de pesos para las competencias de T_{CiU}

* Salidas: $VT_{CiU} \rightarrow$ Conjunto de segmentos de video seleccionados para obtener las recomendaciones para c_i

$WVT_{CiU} \rightarrow$ Conjunto de los pesos de los segmentos de video de VT_{CiU}

VTR($G, LC_U, T_{CiU}, WA_{CiU}$) {

Para cada $c_i \in LC_U$ {

Para cada $y_n^{c_i} \in T_{CiU}$ {

Consultar en G el conjunto VT_{CiU} de instancias VT_n^{CiU} **tal que** $(VT_n^{CiU}, compRef, y_n^{c_i});$

Si VT_{CiU} no contiene a VT_n^{CiU} **entonces**

$VT_{CiU}.añadir(VT_n^{CiU});$

Para cada $VT_n^{CiU} \in VT_{CiU}$ {

Consultar en G el conjunto CVT_n de competencias c_m **tal**

que $(VT_n^{CiU}, compRef, c_m)$ **y** $c_m \in T_{CiU};$

$CVT_n.añadir(c_m);$

$WVT_n^{CiU} = \frac{\sum_{m=1}^M W_n^{CiU}(c_m)}{M};$

$WVT_{CiU}.añadir(WVT_n^{CiU});$

 }

 }

```

    }
  }
}

```

M.5 VR: “Video Ranking”

Teniendo en cuenta los conjuntos $VT_{CiU} = \{VT_1^{CiU}, VT_2^{CiU}, \dots, VT_n^{CiU}, \dots, VT_N^{CiU}\}$ y $WVT_{CiU} = \{WVT_1^{CiU}, WVT_2^{CiU}, \dots, WVT_n^{CiU}, \dots, WVT_N^{CiU}\}$, definidos en el algoritmo VTR (Algoritmo M.4), los cuales contienen los segmentos de video y sus pesos relacionados con la competencia c_i , para la cual el estudiante U tiene que $DoNC_{ui} > GN_o$; sea $V_{CiU} = \{V_1^{CiU}, V_2^{CiU}, \dots, V_n^{CiU}, \dots, V_N^{CiU}\}$ el conjunto de contenidos de video relacionados con los segmentos pertenecientes a VT_{CiU} , y $VTV_n = \{vt_1, vt_2, \dots, vt_m, \dots, vt_M\}$ es el conjunto de segmentos de video que pertenecen a VTV_n vinculadas en la base de conocimiento con el contenido de video V_n^{CiU} por medio de la propiedad `:hasVideoTrack`, la ponderación del n -ésimo contenido de video es computada por:

$$WV_n^{CiU} = \frac{\sum_{m=1}^M WVT_n^{CiU}(vt_m)}{M},$$

donde $WVT_n^{CiU}(vt_m)$ es el peso del segmento de video $vt_m \in VTV_n$ correspondiente en el conjunto WVT_{CiU} y $M = |VTV_n|$. Con el propósito de construir la lista de recomendaciones para la competencia c_i en la que el estudiante U registra un $DoNC_{ui} > GN_o$, se define como aporte de la investigación el algoritmo denominado VR (“Video Ranking” por su significado en inglés), el cual explica a través del pseudoalgoritmo mostrado en Algoritmo M.5.

Algoritmo M.5. Video Ranking. Fuente: propia.

* Entradas: $G \rightarrow$ base de conocimiento

$LC_U \rightarrow$ Conjunto de competencias c_i en las que U registra $DoNC_{ui} > GN_o$

$VT_{CiU} \rightarrow$ Conjunto de segmentos de video relevantes para U con relación a c_i

$WVT_{CiU} \rightarrow$ Conjunto de pesos para los segmentos de VT_{CiU}

* Salidas: $V_{CiU} \rightarrow$ Conjunto de contenidos de video seleccionados para obtener las recomendaciones para c_i

$WV_{CiU} \rightarrow$ Conjunto de los pesos de los contenidos de video de V_{CiU}

$VR(G, LC_U, VT_{CiU}, WVT_{CiU}) \{$

M. Pseudoalgoritmos propuestos

```

Para cada  $c_i \in LC_U$  {
  Para cada  $VT_n^{ciU} \in V_{ciU}$  {
    Consultar en  $G$  el conjunto  $V_{ciU}$  de instancias  $V_n^{ciU}$  tal que
       $(V_n^{ciU}, hasVideoTrack, VT_n^{ciU})$ ;
    Si  $V_{ciU}$  no contiene a  $V_n^{ciU}$  entonces
       $V_{ciU}.añadir(V_n^{ciU})$ ;
    Para cada  $V_n^{ciU} \in VT_{ciU}$  {
      Consultar en  $G$  el conjunto  $VTV_n$  de competencias  $v_m$  tal que
         $(V_n^{ciU}, hasVideoTrack, v_m)$  Y  $v_m \in VT_{ciU}$ ;
       $CV_n.añadir(c_m)$ ;
       $WV_n^{ciU} = \frac{\sum_{m=1}^M WV_n^{ciU}(v_m)}{M}$ ;
       $WV_{ciU}.añadir(WV_n^{ciU})$ ;
    }
  }
}

```

Anexo N

Construcción de listas de recomendaciones para la estrategia de recomendación

Este anexo presenta en extenso la construcción de las listas de recomendaciones introducida en la Sección 6.5.

De acuerdo con el principio adoptado para la estrategia de recomendación (Capítulo 6), conviene construir una lista de recomendaciones para las competencias en las que cada estudiante u tiene cierto grado de necesidad (es decir, competencias para las cuales $GNC_{ui} > GN_o$). Para esto, se tiene en cuenta que los contenidos de VoD están vinculados con competencias, directamente o a través de segmentos de video, tal como está definido en el esquema de metadatos y la ontología propuestos. De acuerdo con estas características, se han establecido cuatro procesos así:

1. *Identificación de competencias relevantes*: proceso que analiza las N asociaciones de las competencias en las que $GNC_{ui} > GN_o$ con el propósito de extraer las competencias más relevantes.
2. *Consulta y ponderación de segmentos de video*: proceso que consulta la base de conocimiento y pondera los segmentos de video con base en las competencias relevantes descubiertas en el proceso 1.
3. *Consulta y ponderación de contenidos de VoD*: proceso que consulta en la base de conocimiento, pondera y ordena los contenidos de VoD de acuerdo con los segmentos de video ponderados en el proceso 2. De aquí se obtiene como resultado el grupo de recomendaciones para cada competencia en las que el estudiante tiene cierto grado de necesidad.
4. *Ordenamiento de contenidos de VoD*: los contenidos ponderados son ordenados para obtener la lista de recomendaciones.

Cada uno de los anteriores procesos se explica con detalle a continuación

N.1 Identificación de competencias relevantes

Sea U un estudiante con perfil P_U , $LC_U = \{c_1, c_2, \dots, c_i, \dots, c_n\}$ es el conjunto de competencias para las que U tiene $DoNC_{ui} > GN_o$, se definen los siguientes conjuntos: 1) el conjunto de N asociaciones para la competencia c_i después del proceso de filtrado se define como $TopNA_{ciU} = \{TopA_1^{ciU}, TopA_2^{ciU}, \dots, TopA_n^{ciU}, \dots, TopA_N^{ciU}\}$, donde $TopA_n^{ciU}$ es una asociación semántica tal que $TopA_n^{ciU} = p - patAssociated(c_i, y_u^{ci})$; 2) el conjunto de pesos totales de las N asociaciones de la competencia c_i es $WA_{ciU} = \{W_1^{ciU}, W_2^{ciU}, \dots, W_n^{ciU}, \dots, W_N^{ciU}\}$, donde W_n^{ciU} es el peso de la n -ésima asociación de la competencia c_i . Con el propósito de obtener una lista de recomendaciones para la competencia c_i , el SR extrae las competencias y_u^{ci} de las N asociaciones cuyo origen corresponde a c_i para cada estudiante U , de acuerdo con el algoritmo, aporte de la investigación, denominado RCS (“*Relevant Competencies Search*” por su significado en inglés). De manera general, RCS explora las N asociaciones semánticas inferidas por el *Filtro semántico* con el fin de extraer de ellas las competencias más importantes. Los pasos que constituyen la ejecución de este algoritmo se explican a continuación (el pseudoalgoritmo de RCS se presenta en el Algoritmo M.3, Sección M.3):

1. Se elige una competencia $c_i \in LC_U$ que no haya sido evaluada. Se crea un conjunto T_{ciU} que contendrá las competencias relevantes para la competencia c_i para la cual U tiene un $DoNC_{ui} > GN_o$. Ir al paso 2. Si no existe tal competencia, se termina el algoritmo y se reinicia para otro usuario.
2. Se toma una asociación semántica $TopA_n^{ciU} \in TopNA_{ciU}$ que no haya sido evaluada. Ir al paso 3. Si tal asociación no existe, volver al paso 1.
3. Se extrae la competencia y_u^{ci} que corresponde al término de la asociación semántica evaluada y se introduce al conjunto T_{ciU} . Volver al paso 2.

Como resultado se obtiene un conjunto T_{ciU} que contiene las N competencias más relevantes para cada competencia c_i de cada estudiante U (notar que $|T_{ciU}| = |TopNA_{ciU}| = N$). Con el propósito de ponderar segmentos y contenidos de video en los próximos procesos, se ha asignado a cada competencia relevante el peso de la asociación de la que fue extraída. En consecuencia, hay una correspondencia en cuanto a posición, de cada elemento de los conjuntos T_{ciU} y WA_{ciU} .

N.2 Consulta y ponderación de segmentos de video

Con el propósito de construir la lista de recomendaciones para la competencia c_i en la que el estudiante U registra un $DoNC_{ui} > GN_o$, el SR consulta inicialmente en la base

de conocimiento los segmentos de video relacionados con las competencias relevantes de c_i por medio de la propiedad $:compRef$. Teniendo en cuenta que un segmento de video puede estar vinculado con una o más competencias relevantes, la ponderación de éste es el resultado de promediar las ponderaciones de las éstas. Considerando los conjuntos $LC_U = \{c_1, c_2, \dots, c_i, \dots, c_n\}$, $T_{CiU} = \{y_1^{c_i}, y_2^{c_i}, \dots, y_n^{c_i}, \dots, y_N^{c_i}\}$ y $WA_{CiU} = \{W_1^{c_iU}, W_2^{c_iU}, \dots, W_n^{c_iU}, \dots, W_N^{c_iU}\}$ definidos en la sección anterior, y sea $VT_{CiU} = \{VT_1^{c_iU}, VT_2^{c_iU}, \dots, VT_n^{c_iU}, \dots, VT_N^{c_iU}\}$ el conjunto de segmentos relacionados con las competencias relevantes descubiertas para la competencia c_i tal que $DoNC_{ui} > GN_o$ para el estudiante U , y $CVT_n = \{c_1, c_2, \dots, c_m, \dots, c_M\}$ es el conjunto de competencias que pertenecen a T_{CiU} vinculadas en la base de conocimiento con el segmento de video $VT_n^{c_iU}$, la ponderación del n-ésimo segmento de video es computada por:

$$WVT_n^{c_iU} = \frac{\sum_{m=1}^M W_n^{c_iU}(c_m)}{M}, \quad (N.1)$$

donde $W_n^{c_iU}(c_m)$ es el peso de la competencia relevante $c_m \in CVT_n$ correspondiente en el conjunto VT_{CiU} y $M = |CVT_n|$. El algoritmo, denominado VTR (“*Video Tracks Ranking*” por su significado en inglés) explica este proceso, el cual es un aporte de la investigación (el pseudoalgoritmo de VTR se presenta en el Algoritmo M.4, Sección M.4):

1. Se toma una competencia $c_i \in LC_U$, tal que $DoNC_{ui} > GN_o$, que no haya sido evaluada. Se crean dos conjuntos $VT_{CiU} = \{VT_1^{c_iU}, VT_2^{c_iU}, \dots, VT_n^{c_iU}, \dots, VT_N^{c_iU}\}$ y $WVT_{CiU} = \{WVT_1^{c_iU}, WVT_2^{c_iU}, \dots, WVT_n^{c_iU}, \dots, WVT_N^{c_iU}\}$, los cuales son el conjunto de segmentos de video vinculados con las competencias relevantes y el conjunto de los pesos de tales segmentos, respectivamente. Ir al paso 2. Si no existe tal competencia, se termina el algoritmo para el estudiante U y se inicia para otro estudiante.
2. Se toma una competencia relevante $y_n^{c_i} \in T_{CiU}$ para la competencia c_i , que no haya sido evaluada. Ir al paso 3. Si no existe tal competencia, ir al paso 5.
3. Se consultan en la base de conocimiento G los segmentos de video vinculados a $y_n^{c_i}$, de acuerdo con la tripleta $(VT_n^{c_iU}, compRef, y_n^{c_i})$. Ir al paso 4.
4. Los segmentos consultados se añaden al conjunto VT_{CiU} , cuidando que no haya repeticiones de éstos. Ir al paso 5.
5. Se toma un segmento de video $VT_n^{c_iU} \in VT_{CiU}$ que no haya sido evaluado. Se crea un conjunto CVT_n que contendrá las competencias asociadas a $VT_n^{c_iU}$. Ir al paso 6. Si no existe tal segmento de video, volver al paso 2.
6. Consultar en G las competencias c_m asociadas al segmento de video $VT_n^{c_iU}$ de acuerdo la tripleta $(VT_n^{c_iU}, compRef, c_m)$. Ir al paso 7.

7. Si las competencias consultadas también hacen parte del conjunto T_{CiU} , éstas se añaden a CVT_n . Ir al paso 8.
8. Se calcula el peso WVT_n^{CiU} del segmento de video VT_n^{CiU} por medio de la expresión (N.1). Ir al paso 9.
9. Se añade WVT_n^{CiU} al conjunto WVT_{CiU} . Volver al paso 1.

Como resultado del algoritmo, se generan los vectores VT_{CiU} y WVT_{CiU} , los cuales contienen los segmentos de video correspondientes a la competencia c_i y sus pesos, respectivamente. Es importante señalar que hay una correspondencia en cuanto a la posición, entre cada segmento y peso. Estos vectores son importantes para ponderar los contenidos de video en la construcción de la lista de recomendaciones para cada competencia c_i . Este proceso se explica en la siguiente sección.

N.3 Consulta y ponderación de contenidos de VoD

Este proceso mantiene el enfoque de la consulta y ponderación de los segmentos de video tratado en la sección anterior. Dado que se ha calculado una lista de segmentos relevantes con su correspondiente ponderación, aquí se consultan los contenidos relacionados a éstos y se realiza su ponderación, permitiendo construir una lista de recomendaciones final para cada competencia en la que hay un nivel alto de necesidad. Teniendo en cuenta los conjuntos $VT_{CiU} = \{VT_1^{CiU}, VT_2^{CiU}, \dots, VT_n^{CiU}, \dots, VT_N^{CiU}\}$ y $WVT_{CiU} = \{WVT_1^{CiU}, WVT_2^{CiU}, \dots, WVT_n^{CiU}, \dots, WVT_N^{CiU}\}$, definidos en la sección anterior, los cuales contienen los segmentos de video y sus pesos relacionados con la competencia c_i , para la cual el estudiante U tiene que $DoNC_{ui} > GN_o$; sea $V_{CiU} = \{V_1^{CiU}, V_2^{CiU}, \dots, V_n^{CiU}, \dots, V_N^{CiU}\}$ el conjunto de contenidos de video relacionados con los segmentos pertenecientes a VT_{CiU} , y $VTV_n = \{vt_1, vt_2, \dots, vt_m, \dots, vt_M\}$ es el conjunto de segmentos de video que pertenecen a VTT_{CiU} vinculadas en la base de conocimiento con el contenido de video V_n^{CiU} por medio de la propiedad `:hasVideoTrack`, la ponderación del n-ésimo contenido de video es computada por:

$$WV_n^{CiU} = \frac{\sum_{m=1}^M WVT_n^{CiU}(vt_m)}{M}, \quad (N.2)$$

donde $WVT_n^{CiU}(vt_m)$ es el peso del segmento de video $vt_m \in VVT_n$ correspondiente en el conjunto WVT_{CiU} y $M = |VVT_n|$. Con el propósito de construir la lista de recomendaciones para la competencia c_i en la que el estudiante U registra un $DoNC_{ui} > GN_o$, se define como aporte de la investigación el algoritmo VR (“Video

Ranking” por su significado en inglés), el cual se explica así (el pseudoalgoritmo de VR se muestra en el Algoritmo M.5, Sección M.5):

1. Se toma una competencia $c_i \in LC_U$, tal que $DoNC_{ui} > GN_o$, que no haya sido evaluada. Se crean dos conjuntos $V_{CiU} = \{V_1^{CiU}, V_2^{CiU}, \dots, V_n^{CiU}, \dots, V_N^{CiU}\}$ y $WV_{CiU} = \{WV_1^{CiU}, WV_2^{CiU}, \dots, WV_n^{CiU}, \dots, WV_N^{CiU}\}$, los cuales son el conjunto de contenidos de video vinculados con las competencias relevantes y el conjunto de los pesos de tales contenidos, respectivamente. Ir al paso 2. Si no existe tal competencia, se termina el algoritmo para el estudiante U y se inicia para otro estudiante.
2. Se toma un segmento de video relevante $VT_n^{CiU} \in VT_{CiU}$ para la competencia c_i , que no haya evaluada. Ir al paso 3. Si no existe tal competencia, ir al paso 5.
3. Se consultan en la base de conocimiento G los contenidos de video vinculados a VT_n^{CiU} , de acuerdo con la tripleta $(V_n^{CiU}, hasVideoTrack, VT_n^{CiU})$. Ir al paso 4.
4. Los contenidos de video consultados se añaden al conjunto V_{CiU} , cuidando que no haya repeticiones. Ir al paso 5.
5. Se toma un contenido de video $V_n^{CiU} \in V_{CiU}$ que no haya sido evaluado. Se crea un conjunto CV_n que contendrá las competencias asociadas a V_n^{CiU} . Ir al paso 6. Si no existe tal segmento de video, volver al paso 2.
6. Consultar en G los segmentos de video v_m asociadas al contenido de video V_n^{CiU} de acuerdo la tripleta $(V_n^{CiU}, hasVideoTrack, VT_n^{CiU})$. Ir al paso 7.
7. Si las competencias consultadas también hacen parte del conjunto VT_{CiU} , éstas se añaden a CV_n . Ir al paso 8.
8. Se calcula el peso WV_n^{CiU} del contenido de video V_n^{CiU} por medio de la expresión (N.2). Ir al paso 9.
9. Se añade WV_n^{CiU} al conjunto WV_{CiU} . Volver al paso 1.

N.4 Ordenamiento de contenidos de VoD

Finalmente, se obtiene una lista de recomendaciones para el estudiante U apropiados para apoyar la competencia c_i para la que $DoNC_{ui} > GN_o$. Para esto, el SR realiza una ordenación de los contenidos de video que hacen parte del conjunto V_{CiU} de acuerdo con sus pesos de mayor a menor almacenados en WV_{CiU} , seleccionando los R mejor ponderados.

Anexo O

Prototipos

Este anexo describe los prototipos Java construidos del método computacional para la evaluación de competencias y del *Sistema de Recomendaciones (SR)* propuesto en el trabajo de investigación. Para cada uno de éstos, se muestra inicialmente una vista funcional que señala los componentes desarrollados para llevar a cabo cada proceso involucrado. Después, se hace una descripción de cada componente desde el punto de vista funcional, haciendo especial énfasis en la estructura de las colecciones Mongo de las bases de datos. Finalmente, se presenta una vista dinámica de cada prototipo en la que se detalla el orden de ejecución de los procesos, al tiempo que se menciona el rol de los componentes del prototipo.

O.1 CMA-Tool: prototipo del método computacional para la evaluación de competencias

Aquí se extiende la información presentada en la Sección 5.3 sobre *CMA-Tool*, el cual es un prototipo Java del método computacional para la evaluación de competencias. La Figura O.1 presenta una vista funcional de éste, por medio de un diagrama de componentes UML, la cual corresponde a una aplicación de escritorio basada en el *Modelo Vista Controlador (MVC)*. En la vista funcional se destacan tres componentes, los cuales se describen a continuación.

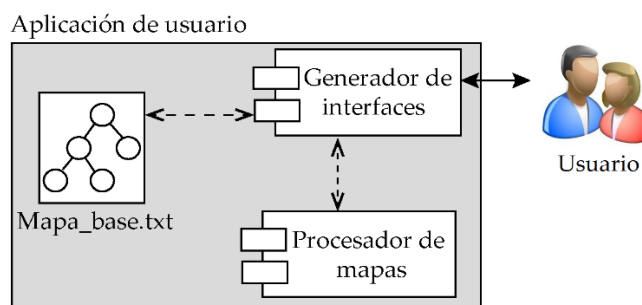


Figura O.1 Vista funcional del prototipo del método computacional para la evaluación de competencias

O.1.1 Generador de interfaces

Este componente es el encargado de generar las interfaces (ventanas) que presentan la información al usuario y permiten su interacción con la aplicación. Tal como se describió en la Sección 5.3, al iniciar la aplicación se despliega una interfaz principal (ver Figura O.2) compuesta por diferentes áreas dispuestas de la siguiente manera: el área (a) presenta un mapa por defecto, el cual sirve de modelo para que el usuario pueda hacer las adaptaciones necesarias para construir la estructura del mapa de competencias deseado. Seleccionando el botón *Configurar*, se presenta el grafo resultante en el área (b), de tal forma que se pueda visualizar si el resultado corresponde al grafo esperado. El área (c) presenta un texto que lista los identificadores, etiquetas y tipos de nodos creados, así como el origen y objetivo de las aristas definidas.

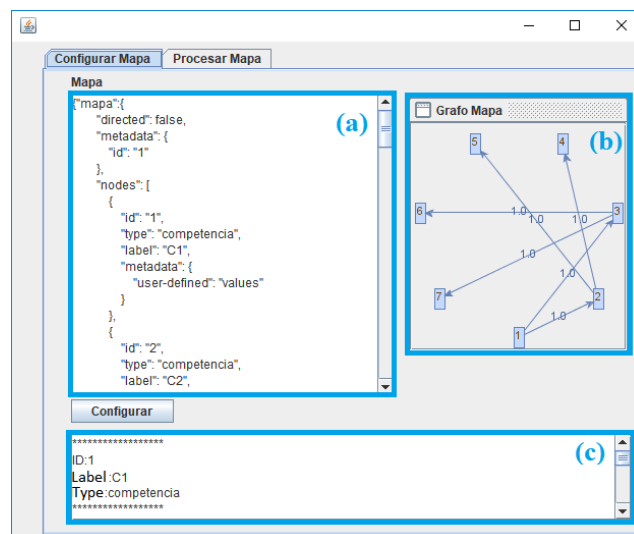


Figura O.2 Interfaz principal del prototipo

Una vez la estructura del mapa ha sido establecida, mediante la opción *Procesar Mapa* del menú superior, el usuario puede introducir en el área (d) (ver Figura O.3) el número de estudiantes a los cuales les serán calculados sus *Niveles de competencia* (NC).

Al seleccionar el botón *Procesar* se despliega una ventana, marcada como área (e), en la cual se solicita el ingreso de los NC alcanzados por el estudiante en cada una de las Actividades de Evaluación (AE). Las solicitudes se van realizando por cada estudiante y AE. Al finalizar el ingreso de toda la información, la herramienta despliega las ventanas mostradas en la Figura O.4. El área (f) muestra los resultados de la evaluación de acuerdo con el modelo propuesto, denominado en la herramienta como *Modo 1* y de acuerdo con el Modelo Basado en el Principio de Extensión (MBPE),

O. Prototipos

denominado *Modo 2*; detallando el identificador del nodo, el tipo de nodo, el Valor de Agregación Numérico (VAN), el par de *2-tuplas* resultantes y el NC lingüístico para cada uno de los estudiantes. Adicionalmente, se despliegan dos gráficas en dos ventanas, en las áreas (h) e (i), las cuales detallan los tiempos de ejecución en microsegundos y el consumo de memoria computacional en Mega Bytes, respectivamente, para cada uno de los estudiantes y cada uno de los modos. Los tiempos y consumo de memoria son presentados en texto en el área (g), junto a la razón entre los tiempos del *Modo 2* y el *Modo 1*.

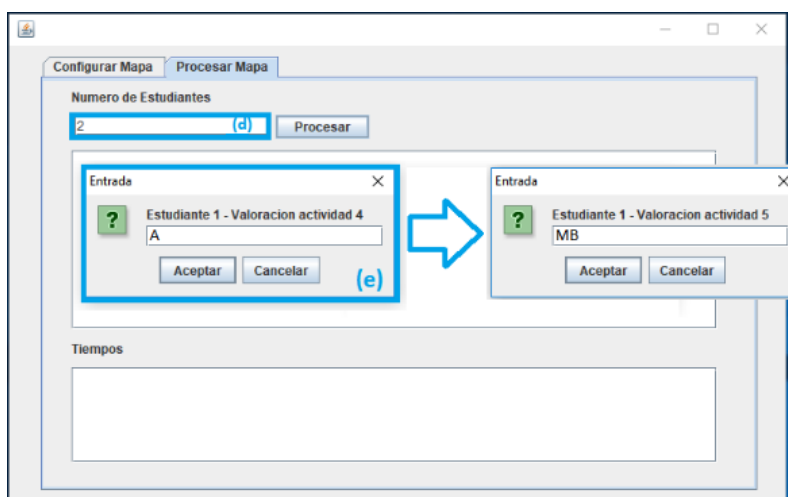


Figura O.3 Interfaz para la introducción de los *Niveles de Competencia*

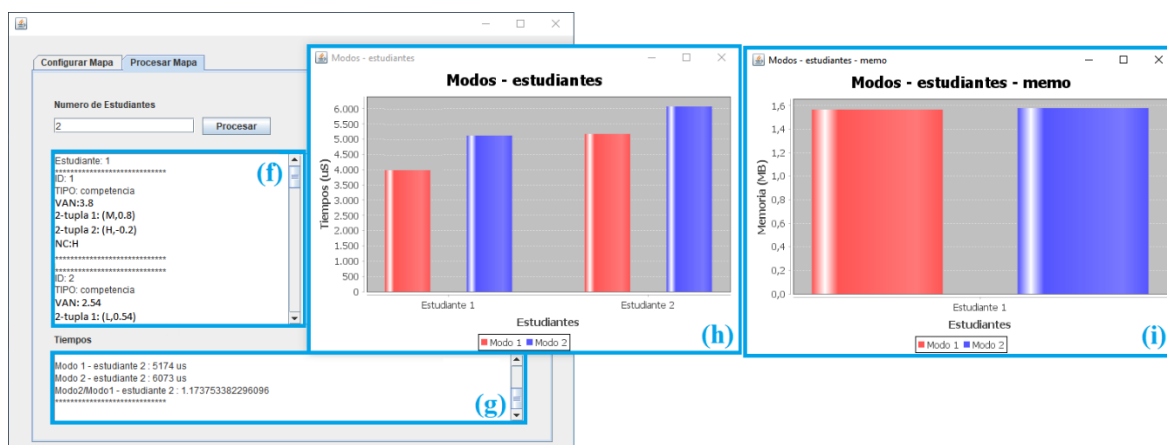


Figura O.4 Interfaces que presentan los resultados de la evaluación de los mapas de competencia, tiempos de ejecución y consumo de memoria

O.1.2 Mapa base

Tal como se presenta en la interfaz principal del prototipo (ver Figura O.2), el área (a) formaliza un mapa de competencias por defecto según el *Formato de Grafos JSON* [108], el cual sirve de modelo para ser modificado de tal forma que se obtenga la

estructura del mapa deseada por el usuario. La razón de usar tal formato está en que se posibilita, con el uso de librerías Java tales como *Jgraph* [135], *JsonPath* [136] y *JsonSimple* [137], el recorrido organizado y sistemático de los grafos implícito en el algoritmo DFS-CA (Sección 5.2.2) en el marco del método computacional para la evaluación de competencias. Al respecto, no fue posible hallar librerías que permitan hacer este tipo de procesos con grafos RDF (esto es, el formato de los grafos formalizados en una ontología). Entonces, queda por desarrollar un proceso capaz de adaptar el formato RDF al del *Formato de Grafos JSON* o que directamente pueda ejecutar el recorrido de los grafos RDF. Cualquiera de los dos casos está fuera del alcance del presente trabajo.

Volviendo a la estructura del mapa presentado en la interfaz principal, ésta se encuentra definida en un archivo *.txt*, el cual es accedido por el componente *Generador de Interfaces* para construir una vista en grafos de la estructura del mapa, la cual se presenta en el área (b) de la interfaz principal. A través del Formato de Grafos JSON se formalizaron los siguientes aspectos:

1. Identificadores (*id*) para los mapas y los nodos.
2. Tipos de nodos (*type*) para los nodos (competencia o actividad).
3. Etiquetas (*label*) para los nodos (por ejemplo, C1 para una competencia o AE1 para una actividad de evaluación).
4. Las aristas, para la cual se usaron los siguientes campos:
 - *source* y *target*, usados para definir el nodo origen y objetivo de la arista
 - *directed*, es un booleano usado para establecer si la relación es dirigida
 - *peso*, es un metadato adicional que se incluyó para establecer el peso de las aristas entre nodos.

Un fragmento del mapa base que ejemplifica la aplicación del *Formato de Grafos JSON* se presenta en la Figura O.5. Se observa cómo inicialmente se definen los nodos del mapa (marcados en verde), los cuales pueden ser *competencia* o *actividad de evaluación* (campo *type*). Después, se definen las aristas (marcadas en rojo), por medio de las cuales se establece la estructura de los mapas.

O.1.3 Procesador de mapas

Este componente se encarga de tomar todos los datos introducidos por el usuario a través de las interfaces para posteriormente calcular los NC ejecutando el método computacional propuesto (Capítulo 5) y el Modelo Basado en el Principio de Extensión (MBPE). Específicamente, este componente ejecuta las siguientes fases:

O. Prototipos

1. *Fase A: construcción del perfil del estudiante*, en la cual se almacenan en vectores los NC alcanzados por los estudiantes en las AE.
2. *Fase B: cómputo de los NC* de acuerdo a dos modos: el *Modo 1*, el cual ejecuta el algoritmo propuesto “*Depth First Search-Competency Assessment*” (Sección 5.2.2), para recorrer los grafos y ejecutar un modelo matemático basado en *2-tuplas* difusas; y el *Modo 2*, el cual ejecuta el MBPE.

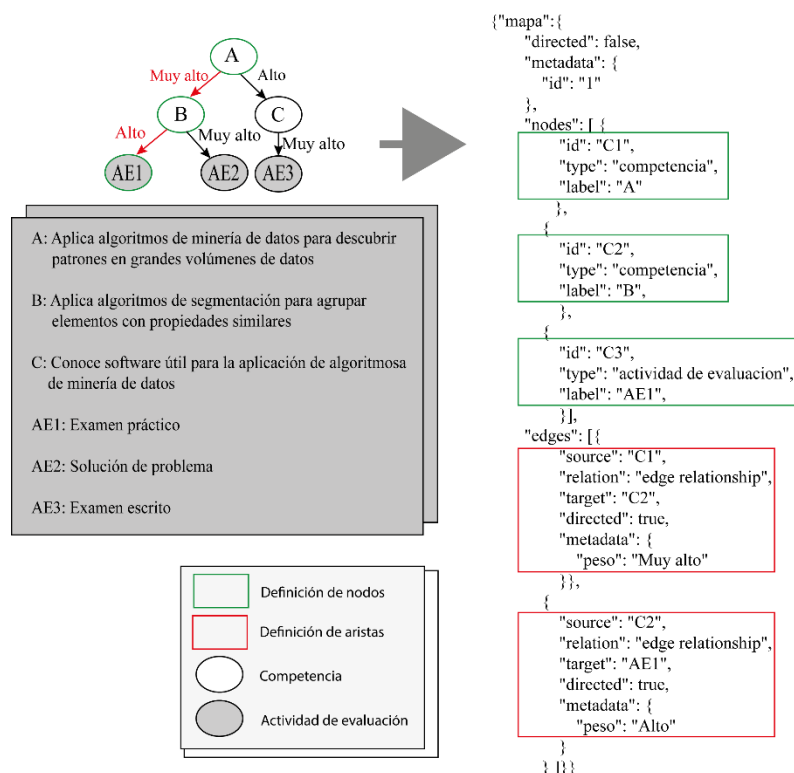


Figura O.5 Ejemplo sobre la representación de la estructura de un mapa usando el Formato de Grafos JSON

O.1.4 Vista dinámica del prototipo

La Figura O.6 presenta, a través de un diagrama de flujo, una vista dinámica de alto nivel del prototipo. Inicialmente, el usuario accede a la aplicación (1). El componente *Generador de Interfaces* realiza la lectura de *Mapa_base.txt* (2). Después, lo procesa (3). Como resultado, se presenta la interfaz principal de la aplicación, incluida la representación del mapa (4). El usuario ingresa los datos de la nueva estructura del mapa de competencias (5). Si la estructura difiere de la estructura por defecto, ésta se procesa y se presenta en la interfaz la nueva representación del mapa (6). En caso contrario, el usuario introduce la cantidad de estudiantes a evaluar y los NC alcanzados en las AE (7). Posteriormente, el componente *Procesador de mapas* realiza el cómputo de los *niveles de competencia* ejecutando el algoritmo propuesto DFS-CA (Sección

5.2.2) (8), mide el tiempo de ejecución (9) y el consumo de memoria (10). Después, el componente *Generador de Interfaces* despliega los resultados del cómputo de NC y genera dos ventanas adicionales en las que presenta gráficamente los resultados de tiempo de ejecución y consumo de memoria (11). Finalmente, el usuario cierra la aplicación (12).

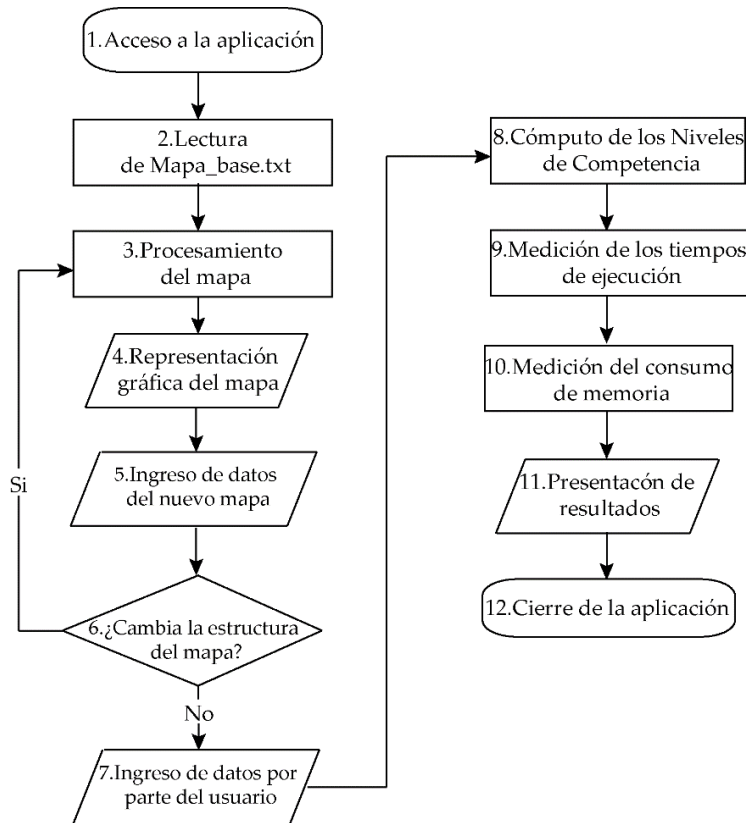


Figura O.6 Vista dinámica de *CMA-Tool*

O.2 Prototipos de sistemas de recomendaciones

En esta sección se presenta el prototipo tanto del SR semántico propuesto como del sintáctico, el cual se basa en el coeficiente Sørensen-Dice. Una vista funcional del mismo se presenta en la Figura O.7, en la cual se delimitan los dos sistemas de recomendaciones en cuestión. En las siguientes secciones se describe cada uno de los prototipos.

O.2.1 EduCompRS: prototipo del sistema de recomendaciones semántico

Tal como se adelantó en la Sección 7.1, el prototipo define una serie de procesos que pueden ejecutarse sin la intervención del usuario (por ejemplo, con ejecuciones programadas) e independientemente de la ejecución de la aplicación de usuario. Otros

O. Prototipos

procesos hacen parte de la aplicación de usuario, ya que dependen de la configuración del interés para éste. Esto podrá observarse en la vista funcional mostrada más adelante, en la que hay una correspondencia entre los procesos y los componentes que las ejecutan.

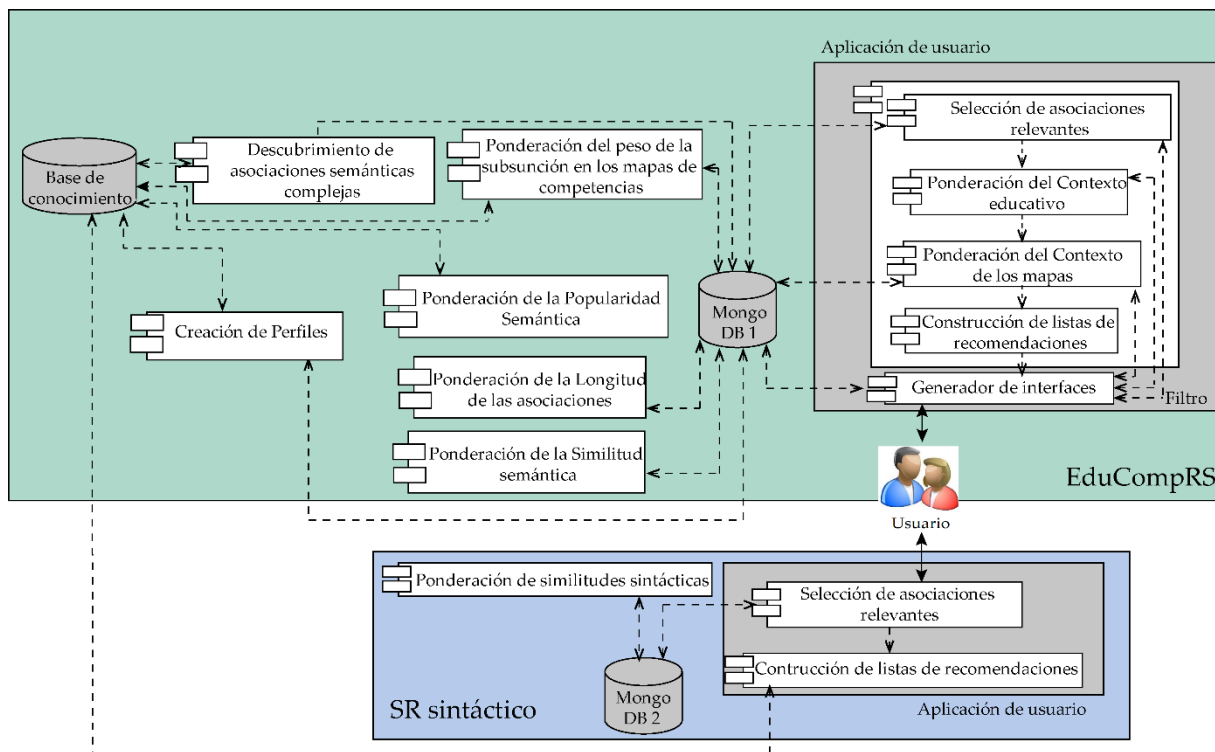


Figura O.7 Vista funcional de los prototipos *EduCompRS* y SR sintáctico

O.2.1.1 Mongo DB 1

Es una base de datos Mongo usada para almacenar información útil durante el proceso de obtención de recomendaciones. Específicamente, define un grupo de colecciones JSON usadas por los demás componentes del SR semántico. Las colecciones serán mencionadas en la explicación de cada componente, incluyendo una descripción de su estructura.

O.2.1.2 Creación de perfiles

Este componente se encarga de consultar los NC de todos los estudiantes, almacenados en la colección *CompetencyUserProfile*, para posteriormente computar sus perfiles- ontología (Sección 6.2), los cuales son almacenados en la colección *UserProfile*. Los documentos de *CompetencyUserProfile* tiene la siguiente estructura:

```
{
  "_id":1,
  "idUser": "1-ON",
  "idInstance": "ON-PE-0211",
```

```

    "class": "Competency",
    "competencyLevel": "M"
  },

```

en la cual se especifica el identificador del usuario (`idUser`), el identificador de la instancia de competencia (`idInstance`), la clase de la instancia (`class`) y el NC del estudiante por la competencia (`competencyLevel`). Los documentos de *UserProfile* tienen la siguiente estructura (es un ejemplo para una propiedad semántica que corresponde al tipo de competencia *Knowledge*):

```

{
  "_id": ObjectId('5a6b5bd283c27f4944939f0a'),
  "idUser": "5-PT",
  "idInstance": "Knowledge",
  "class": "CompetencyType",
  "degreeOfNeed": "0.5"
},

```

en el cual se define el identificador del usuario (`idUser`), el identificador de la instancia (`idInstance`), la clase de la instancia (`class`) y el GN calculado para el usuario (`degreeOfNeed`).

O.2.1.3 Descubrimiento de asociaciones semánticas complejas

Componente encargado de descubrir las asociaciones semánticas formalizadas en la *Base de conocimiento* según el proceso descrito en la Sección 6.3. Las asociaciones descubiertas se almacenan en la colección *Associations*, cuyos documentos tienen la siguiente estructura:

```

{
  "_id": ObjectId('5a41cc8b77363221cdc87e32'),
  "Origin": "CQ-PN-0212",
  "Terminus": "SS-PN-0733",
  "Nodes": [
    "CQ-PN-0212",
    "Solving",
    "SS-PN-0733"
  ]
}.

```

En la estructura se observan definidos los campos que definen el origen de la asociación (`Origin`), el término (`Terminus`) y los nodos que la componen (`Nodes`).

O.2.1.4 Ponderación de la Popularidad personalizada

Componente encargado de calcular el peso de la *Popularidad personalizada* ejecutando (Sección 6.4.1). Para esto, consulta la *Base de conocimiento* y los perfiles-ontología registrado en la colección *UserProfile*. Los valores de popularidad son

guardados en una colección llamada *personalizedPopularities*, cuyos documentos tienen la siguiente estructura (este ejemplo registra la popularidad del verbo *Recognizing* para un usuario determinado):

```
{
  "_id": 2,
  "userId": "5-PT",
  "idInstance": "Recognizing",
  "pop": 1
},
```

en la cual se especifica el identificador del usuario (*userId*), el identificador de la instancia a la cual se le ha calculado su popularidad (*idInstance*) y el valor de la popularidad personalizada (*pop*).

O.2.1.5 Ponderación de la Similitud semántica

Componente encargado de calcular el peso de la *Similitud semántica* entre las competencias con base en sus temas (Sección 6.4.1). Esto lo hace consultando la *Base de conocimiento* para identificar todas las competencias, y una colección llamada *Synsets*, la cual registra los vecindarios de sinónimos entre los términos asociados a los temas de las competencias. A continuación, se muestra la estructura de los documentos de *Synsets* por medio de un ejemplo:

```
{
  "_id": "7",
  "keywords": [
    "Identificar",
    "Reconocer",
    "Fichar",
    "Establecer",
    "Referir",
    "Reseñar",
    "Registrar",
    "Recordar"
  ]
},
```

Los valores de similitud son almacenados en la colección *Similarities*, cuyos documentos tienen la siguiente estructura:

```
{
  "_id": 4,
  "competencyId1": "ON-PE-02",
  "competencyId2": "ON-PE-0212",
  "similarity": 1
},
```

en la cual se definen los identificadores de las dos competencias comparadas (`competencyId1` y `competencyId2`) y el valor de similitud computado (`similarity`).

O.2.1.6 Ponderación de la Longitud de las asociaciones

Este componente se encarga de analizar las asociaciones descubiertas por el *Descubrimiento de asociaciones semánticas complejas*, almacenadas en la colección *Associations*, para calcular su longitud (Sección 6.4.1). Los valores calculados son guardados en una colección llamada *Lenghts*, cuyos documentos tienen la siguiente estructura:

```
{
  "_id":9,
  "Origin": "CQ-PN-0212",
  "Terminus": "A-on-pv-41",
  "Lenght": 5,
  "assoLenght": 0.2
},
```

en la cual se especifican el origen de la asociación (`Origin`), su término (`Terminus`), el número de nodos que la componen (`Lenght`) y el valor del peso de la longitud (`assoLenght`).

O.2.1.7 Ponderación del Peso de la subsunción en los mapas de competencias

Inicialmente, este componente se encarga de explorar todos los mapas de competencias para encontrar relaciones de subsunción entre dos competencias (definidas como *origen* y *término* de las asociaciones descubiertas), para después calcular el peso de la *Subsunción en los mapas de competencias* (Sección 6.4.1). Para esto, consulta la *Base de Conocimiento* y la colección *Associations*. Los resultados son almacenados en una colección llamada *Subsumptions*, cuyos documentos tienen la siguiente estructura (en este caso el componente halló una relación de subsunción entre las dos competencias):

```
{
  "_id":5,
  "Origin": "CQ-PN-091",
  "Terminus": "CQ-PN-0913",
  "mapId": "M24-CQ",
  "Subsumption": 1
},
```

en la cual se especifica el origen de la asociación (`Origin`), su término (`Terminus`), los mapas en los que se halló la relación de subsunción (`mapId`) y el valor del peso de la subsunción (`Subsumption`).

O. Prototipos

En las siguientes secciones se presentan los componentes del prototipo que hacen parte de la aplicación de usuario.

O.2.1.8 Generador de interfaces

Este componente es el encargado de generar las interfaces (ventanas) que presentan la información al usuario y permiten su interacción con la aplicación. Al iniciar la aplicación, se genera la interfaz principal, la cual se muestra en la Figura O.8. En ella, el usuario define el identificador del estudiante objetivo (a), las regiones de los mapas interesantes relacionadas con el *Peso del contexto de los mapas* (b) y las instancias interesantes de la base de conocimiento relacionados con el *Peso del contexto educativo* (c). Adicionalmente, se pueden variar los pesos k_1, \dots, k_6 (d), el número de asociaciones a considerar para obtener las recomendaciones (δ), el número de las N asociaciones y el tamaño de la lista de recomendaciones (f).

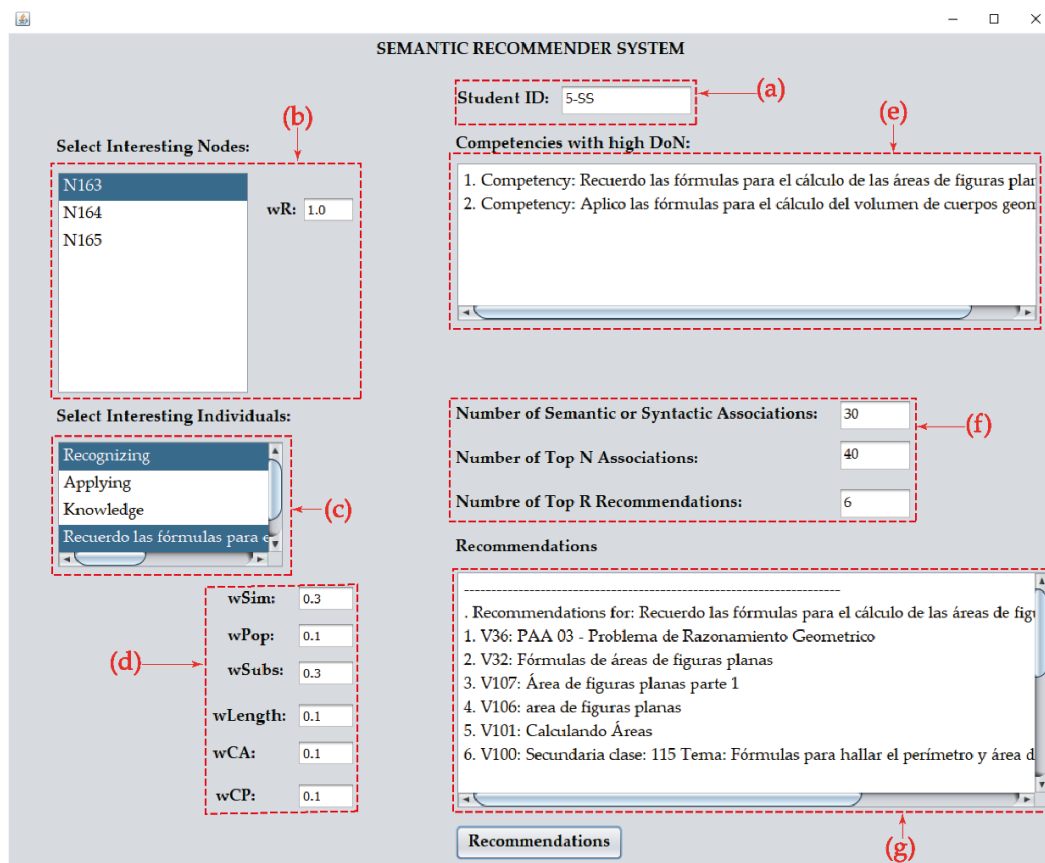


Figura O.8 Interfaz del prototipo del sistema de recomendaciones semántico

Al presionar el botón *Recommendations*, la lista de competencias con un GN alto (un GN mayor que un umbral GN_o) se despliega en pantalla (e), y se calculan tanto el *Peso del contexto de los mapas* como el *Peso del contexto educativo* (Sección 6.4.1). Después, estos resultados y los pesos de los demás componentes son usados para

calcular el peso total de las asociaciones. Subsecuentemente, se lleva a cabo el proceso de filtrado. Finalmente, se despliegan las listas de recomendaciones (g), una lista por cada competencia con un GN alto.

O.2.1.9 Selección de asociaciones relevantes

De acuerdo al identificador del estudiante objetivo, este componente consulta en la colección *UserProfile* las competencias para las cuales el grado de necesidad es mayor que el umbral. A partir de estas competencias, consulta en la colección *Associations* las asociaciones semánticas cuyo origen corresponde a las competencias identificadas. Estas asociaciones son usadas por el sistema para obtener las recomendaciones para el estudiante objetivo.

O.2.1.10 Filtro

Componente encargado de filtrar las asociaciones irrelevantes, de acuerdo al ponderado de las asociaciones seleccionadas por el componente *Selección de asociaciones relevantes*. Inicialmente explora cada asociación, para la cual extrae de las colecciones *personalizedPopularities*, *Similarities*, *Lenghts*, los valores de los pesos de la *Popularidad personalizada*, *Similitud semántica* y *Longitud de las asociaciones*, respectivamente. Posteriormente, de acuerdo a la configuración del sistema definida por el usuario, el *Filtro* solicita calcular el peso del *Contexto educativo* y el peso del *Contexto de los mapas* a los componentes *Ponderación del contexto educativo* y *Ponderación del Contexto de los mapas*. Con estos valores, calcula el peso total de todas las asociaciones. Finalmente, selecciona las N asociaciones mejor ponderadas, las cuales son usadas posteriormente por el componente *Creación de listas de recomendaciones*. Todo el proceso puede consultarse en la Sección 6.4. Éste se ha implementado desarrollando componentes individuales, los cuales se describen a continuación.

O.2.1.10.1 Ponderación del Contexto educativo

Este componente usa las instancias de la ontología del interés del usuario, especificados en la interfaz principal, para calcular el peso del Contexto educativo. Sus resultados son usados por el *Filtro* para calcular el peso de las asociaciones.

O.2.1.10.2 Ponderación del Contexto de los mapas

Este componente usa los nodos de los mapas de interés para el usuario, especificados en la interfaz principal, para calcular el peso del *Contexto de los mapas*. Sus resultados son usados también por el *Filtro* para calcular el peso de las asociaciones.

O.2.1.10.3 Creación de listas de recomendaciones

Este componente utiliza las N asociaciones mejor ponderadas seleccionadas por el *Filtro* para obtener las listas de recomendaciones. Su funcionamiento se puede consultar en la Sección 6.5. Las listas de recomendaciones son presentadas en la interfaz principal.

O.2.1.11 Vista dinámica de la aplicación de usuario del sistema de recomendaciones semántico

La Figura O.9 presenta, a través de un diagrama de flujo, una vista dinámica de alto nivel de la aplicación del SR semántico. El funcionamiento de los demás componentes no ha sido considerado, ya que se ejecutan de forma independiente.

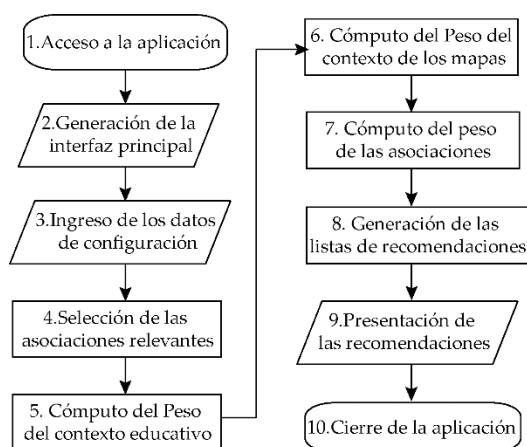


Figura O.9 Vista dinámica de la aplicación de usuario del sistema de recomendaciones semántico

Inicialmente, el usuario accede a la aplicación (1). Después, el *Generador de interfaces* de la aplicación genera la interfaz principal, en la cual se carga la configuración por defecto (2). Ahora, el usuario puede ingresar la configuración del SR de su interés (3) a través de la interfaz principal (Figura O.8). *Selección de asociaciones relevantes* selecciona las asociaciones semánticas relevantes para el usuario objetivo (4). *Ponderación del Contexto educativo* usa las asociaciones semánticas relevantes para calcular el peso del *Contexto educativo* (5). Después, *Ponderación del contexto de los mapas* calcula el peso del *Contexto de los mapas* (6). Estos pesos, junto a los demás (calculados por los componentes que no hacen parte de la aplicación de usuario) almacenados en la *BD Mongo 1* son usados por el *Filtro* para calcular el peso de las asociaciones semánticas y escoger las N mejor ponderadas (7). Con base en estas asociaciones y su ponderación, *Creación de listas de recomendaciones* pondera los segmentos de video relacionados y los contenidos para obtener finalmente las recomendaciones para el estudiante objetivo (8). Las recomendaciones son mostradas

en la interfaz principal de la aplicación (9). Finalmente, el usuario cierra la aplicación (10).

O.2.2 Prototipo del sistema de recomendaciones sintáctico

A continuación, se describen los componentes que hacen parte del SR sintáctico. Cabe mencionar que éste no tiene una interfaz gráfica, por lo cual, los datos se ingresan y los resultados se muestran por consola. Pero antes, se realiza una descripción de la estrategia de recomendación sintáctica, la cual fue usada para comparar la precisión, el tiempo de ejecución y el consumo de memoria del SR semántico propuesto.

O.2.2.1 Estrategia de recomendación sintáctica

Tal como se mencionó en la Sección 7.1, la estrategia de recomendación sintáctica es equivalente a la planteada para el SR semántico. Específicamente, en lugar de ponderar las asociaciones de acuerdo a las métricas de peso semánticas, en la estrategia sintáctica se calculan las similitudes entre las competencias usando el coeficiente Sørensen-Dice (expresión (7.1)) con base en las palabras clave definidas en las descripciones. Los demás procesos son similares:

1. Para cada estudiante objetivo U , se identifican las competencias tales que su Grado de Necesidad (GN) es mayor que GN_o , siendo GN_o un umbral configurable por el usuario que representa un límite entre un GN alto y uno bajo. Por defecto $GN_o=0.5$.
2. Para cada competencia con un GN mayor que GN_o , se identifican las competencias relevantes, las cuales corresponden a las N con los mayores valores de similitud.
3. En la base de conocimiento se consultan los M segmentos de video vinculados a las N competencias relevantes. Cada segmento es ponderado siguiendo el mismo proceso definido para el SR semántico, el cual se explica en la Sección N.2 (incluyendo la aplicación de la expresión (N.1)). Esto, usando los valores de similitud sintáctica calculados para las N competencias relevantes en lugar de los pesos de las asociaciones semánticas.
4. En la base de conocimiento se consultan los N contenidos asociados a los M segmentos de video. Los contenidos son ponderados siguiendo el mismo proceso definido para el SR semántico explicado en la Sección N.3 (incluyendo la aplicación de la expresión (N.2)). Específicamente, la ponderación de los contenidos se calcula con base en los valores de ponderación calculados para los segmentos de video.
5. Finalmente se generan las listas de recomendaciones, organizando los contenidos de mayor a menor según su ponderación. Este proceso es similar al del SR semántico explicado en la Sección N.4.

O.2.2.2 Mongo DB 2

Es una base de datos Mongo que contiene las colecciones JSON usadas en la obtención de recomendaciones del prototipo del SR sintáctico. Las colecciones serán descritas en la descripción de los siguientes componentes, según su uso.

O.2.2.3 Ponderación de similitudes sintácticas

Este componente se encarga de calcular las similitudes entre las competencias usando las palabras clave definidas en las descripciones de las mismas. Para esto, inicialmente consulta en la colección *Associations* las asociaciones cuyo origen son las competencias cuyo grado de necesidad es mayor que un umbral para el estudiante objetivo. Después, calcula la similitud entre el origen y el término de cada asociación, de acuerdo al coeficiente Sørensen-Dice (expresión 7.1). La similitud se mide comparando las palabras clave (*keywords*) de las competencias, las cuales están descritas en la colección *Competency*, cuyos documentos tienen la siguiente estructura:

```
{
  "_id": 1,
  "competencyId": "ON-PE-02",
  "keywords": [
    "Teorema",
    "Pitágoras",
    "Figura geométrica",
    "Relación",
    "Tales",
    "Propiedad",
    "Triángulo",
    "Demostrar",
    "Contrastar"
  ],
  "topicKeywords": [
    "Teorema de Pitágoras"
  ]
},
```

en la cual se especifica el identificador de la competencia (*competencyId*), sus palabras clave (*keywords*) y las palabras clave de los temas tratados en la competencia (*topicKeywords*).

Los resultados se almacenan en la colección *DICESimilarities*, cuyos documentos tienen la siguiente estructura:

```
{
  "_id": ObjectId('5a7c8e47a4266622eecf606b'),
```

```

    "competencyId1": "ON-PE-02",
    "competencyId2": "ON-PE-021",
    "DICESimilarity": 0.75
  },

```

donde se definen los identificadores de las competencias comparadas (`competencyId1` y `competencyId2`) y el valor de la similitud (`DICESimilarity`).

O.2.2.4 Creación de listas de recomendaciones

Este componente consulta los valores de similitud en la colección *DICESimilarities* y selecciona las asociaciones mejor ponderadas para el estudiante objetivo. A partir de estas asociaciones, extrae sus términos, los cuales son las competencias más relevantes. Después, consulta en la *Base de conocimiento* los segmentos de video y contenidos asociados a las competencias más relevantes, para después ponderarlos. Finalmente, construye y presenta las listas de recomendaciones tomando como criterio el valor de la ponderación.

O.2.2.5 Vista dinámica de la aplicación de usuario del sistema de recomendaciones sintáctico

La Figura O.10 presenta, a través de un diagrama de flujo, una vista dinámica de alto nivel de la aplicación del SR sintáctico. El funcionamiento de los demás componentes no ha sido considerado, ya que se ejecutan de forma independiente.

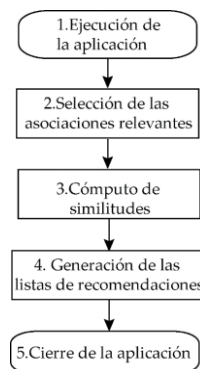


Figura O.10 Vista dinámica del prototipo del sistema de recomendaciones sintáctico

Inicialmente, se ejecuta la aplicación Java (1). Después, *Ponderación de similitudes sintácticas* selecciona de la colección *Associations*, almacenada en la *Mongo DB 2*, las asociaciones relevantes para el usuario objetivo (2). Posteriormente, el mismo componente realiza el cálculo de las similitudes sintácticas entre los orígenes y los términos de las asociaciones (3). Los resultados son almacenados en la colección *DICESimilarities* de la *Mongo DB 2* (4). Después, *Creación de listas de recomendaciones* obtiene las recomendaciones (5) las cuales son presentadas posteriormente en pantalla (6). Finalmente, se cierra la aplicación (7).

ANEXO P

Resultados complementarios de tiempos de ejecución del prototipo del sistema de recomendaciones propuesto

Este anexo presenta, de manera complementaria a la Sección 7.5.2, las medidas de tiempo de ejecución del prototipo del *Sistema de Recomendaciones* (SR) semántico propuesto (*EduCompRS*), correspondientes a procesos que pueden ejecutarse de forma independiente a la aplicación de usuario. Precisamente, este hecho hace que los tiempos usados por tales procesos no sean críticos en cuanto a la experiencia del usuario, por lo cual, fueron omitidos en los resultados del Capítulo 7. Sin embargo, conviene realizar un análisis al respecto para identificar el comportamiento y tendencia de las variables involucradas con miras a desplegar un SR en ambientes reales.

Inicialmente, se presentan en la Figura P.1 los tiempos de ejecución usados en la construcción de los perfiles-ontología. Éstos resultados involucran el cálculo de los *Grados de Necesidad* (GN) de los usuarios por las competencias²⁴ y sus atributos identificados como de importancia para el desarrollo de la estrategia de recomendación: el *verbo de acción* y el *tipo de competencia*, correspondientes a las propiedades ontológicas *hasActionVerb* y *hasCompetencyType*, respectivamente. Se presume que el uso de más atributos, tales como *hasIntendedUser* (el rol de usuario), *hasSchema* (el esquema de cualificación) o *hasLanguage* (el lenguaje), implica el consumo de mayores tiempos de ejecución. Sin embargo, el uso de más propiedades puede comprometer la viabilidad computacional del SR. Con respecto a la Figura P.1, se observa que los tiempos guardan una correlación directa con el número de usuarios. Asumiendo que la curva tiene una tendencia polinómica (curva roja), el comportamiento de los tiempos en milisegundos es aproximadamente igual al de la función $y = -1,353x^2 + 164,53x + 1144,4$, donde x es el número de usuarios registrados en el sistema.

²⁴ Los tiempos de ejecución presentados no incluyen los correspondientes al cálculo de los niveles de competencia. Éstos se pueden consultar en la evaluación del método computacional en la Sección 5.3.

Otro proceso importante dentro de la estrategia de recomendación es el descubrimiento de las asociaciones semánticas complejas en la base de conocimiento implementada. Los tiempos de ejecución medidos para este proceso se presentan en la Figura P.2. Al respecto, cabe recordar que se definió un número por defecto de secuencias de propiedades (Sección 7.5.3) construidas a partir de las propiedades *hasActionVerb*, *hasCompetencyType* y sus propiedades inversas. Entonces, se presume que el uso de más propiedades, tales como *hasIntendedUser* (el rol de usuario), *hasSchema* (el esquema de cualificación) o *hasLanguage* (el lenguaje), implica el consumo de más tiempo de ejecución. Considerando los anteriores aspectos, el descubrimiento de asociaciones semánticas tiene una correlación con el número de competencias formalizadas en la base de conocimiento. Asumiendo que la curva tiene una tendencia polinómica (curva roja), el comportamiento de los tiempos en milisegundos es aproximadamente igual al de la función $y = 2,5007x^2 - 7,4428x + 858,17$, donde x es el número de competencias formalizadas en la base de conocimiento.

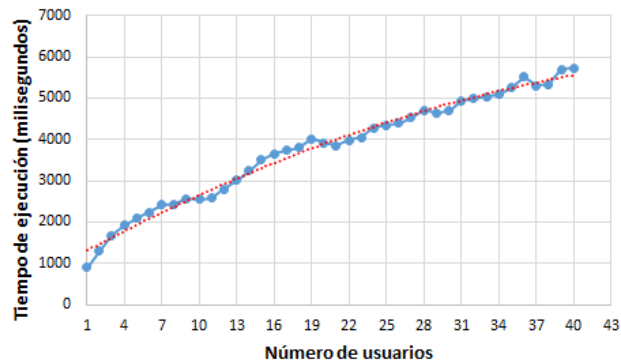


Figura P.1 Resultados de tiempo de ejecución para la construcción de perfiles-ontología

Por otra parte, el cálculo de la popularidad personalizada es un proceso propio de la ponderación de las asociaciones semánticas dentro del filtrado del SR propuesto. Sus tiempos de ejecución, mostrados en la Figura P.3, dependen del número de usuarios registrados en el sistema, aunque, cabe mencionar que para la evaluación se consideraron las dos propiedades ya mencionadas: *hasActionVerb* y *hasCompetencyType*. Entonces, se presume que el uso de más propiedades conlleva el consumo tiempos de ejecución mayores.

P. Resultados complementarios de tiempos de ejecución del prototipo del sistema de recomendaciones propuesto

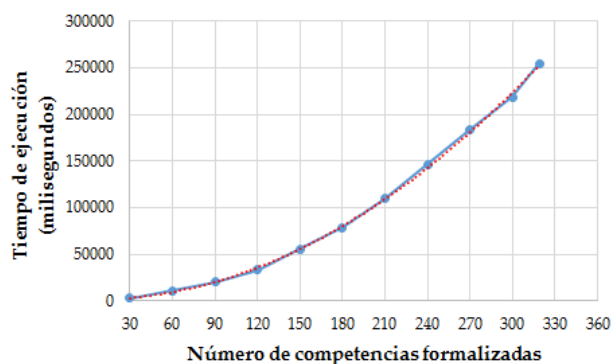


Figura P.2 Resultados de tiempo de ejecución para el descubrimiento de asociaciones semánticas complejas

Asumiendo que la curva tiene una tendencia lineal (curva roja), el comportamiento de los tiempos en milisegundos es aproximadamente igual al de la función $y = 6,2738x + 13,843$, donde x es el número de usuarios del sistema.

Con respecto al cálculo de la similitud semántica, los tiempos de ejecución medidos se presentan en la Figura P.4. Tal como se adelantó en el Capítulo 7, en el prototipo no se consideró la construcción de las bolsas de palabras, producto de la lematización de términos a partir de los temas de las competencias, ni la construcción de vecindarios de términos, haciéndose esto de forma manual. En consecuencia, los tiempos de ejecución medidos corresponden a la ejecución de los algoritmos propios del cálculo de similitud semántica. En la gráfica se observa que éstos tienden a comportarse como una función polinómica (curva roja) $y = 2,5184x^2 - 14,595x + 1792,8$, donde x es el número de competencias formalizadas en la base de conocimiento.

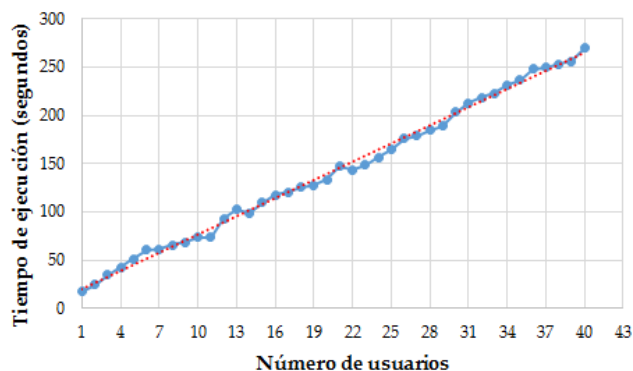


Figura P.3 Resultados de tiempo de ejecución para el cálculo de la popularidad personalizada

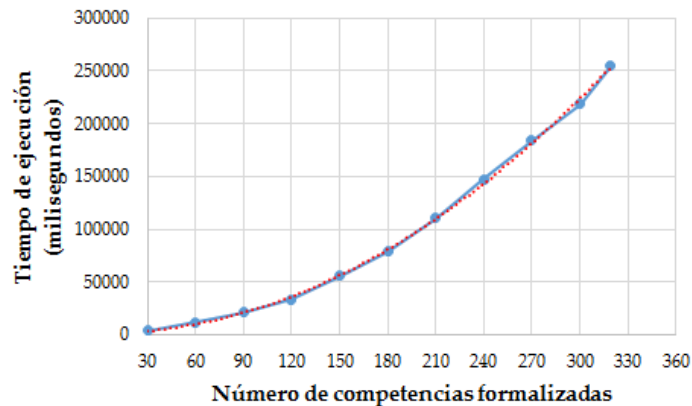


Figura P.4 Resultados de tiempo de ejecución para el cálculo de la similitud semántica

Por otra parte, la Figura P.5 muestra los tiempos de ejecución del cálculo de la longitud de las asociaciones. Como era de esperar, éstos tienen una correlación directa con el número de asociaciones descubiertas en la base de conocimiento. Asumiendo que la curva tiene una tendencia lineal (curva roja), el comportamiento de los tiempos en milisegundos es aproximadamente igual al de la función $y = 38,839x + 333,93$, donde x es el número de asociaciones semánticas descubiertas en la base de conocimiento.

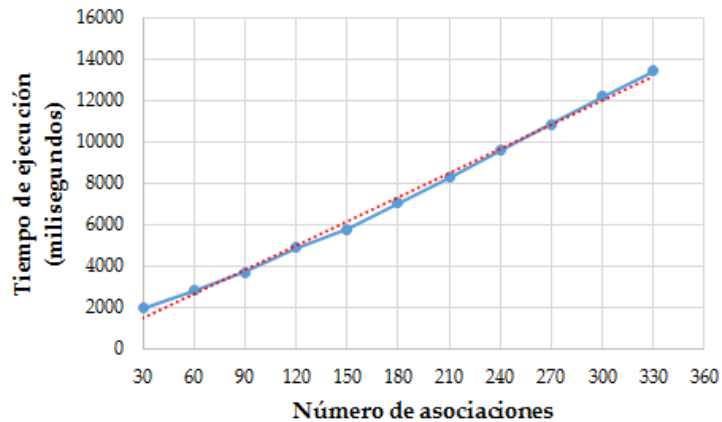


Figura P.5 Resultados de tiempo de ejecución para el cálculo de la longitud de la asociación

Finalmente, fueron medidos los tiempos de ejecución usados en la ponderación del peso de la *Subsunción en los mapas de competencias* formalizadas en la base de conocimiento, los cuales se muestran en la Figura P.6.

P. Resultados complementarios de tiempos de ejecución del prototipo del sistema de recomendaciones propuesto

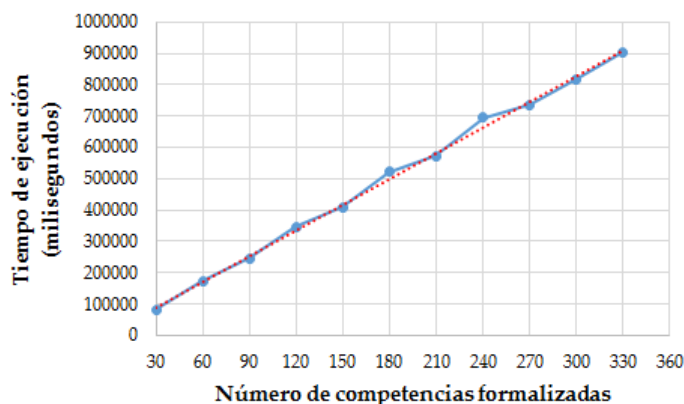


Figura P.6 Resultados de tiempo de ejecución para la ponderación del peso de la subsunción en los mapas

Aquí, los tiempos guardan una correlación directa con el número de competencias formalizadas (en realidad, con los nodos de los mapas que las representan). Asumiendo que la curva tiene una tendencia lineal (curva roja), el comportamiento de los tiempos es aproximadamente igual al de la función $y = 2733,2x + 8498,2$, donde x es el número de competencias formalizadas en la base de conocimiento que están representadas en los mapas de competencias por medio de nodos. Como puede verse en la Figura P.6, el proceso de ponderación del peso de la *Subsunción en los mapas de competencias* es el más crítico porque alcanza tiempos de ejecución altos (por ejemplo, para 330 competencias el tiempo de ejecución es aproximadamente 15 minutos). Una medida para reducir el efecto negativo de los tiempos de ejecución altos podría ser la de limitar el número de competencias a considerar, lo cual implica la implementación de una forma de selección de éstas con base en un criterio bien establecido (por ejemplo, considerando las competencias formalizadas en los mapas de un currículo escolar específico).

Como observación final, los resultados de los tiempos de ejecución indican que éstos tienden a aumentar linealmente en algunos casos, y polinómicamente en otros, manteniendo una correlación con la cantidad de competencias formalizadas en la ontología y la cantidad de usuarios del sistema. Por esta razón se decidió independizar los procesos aquí evaluados de la aplicación de usuario del SR para no comprometer la viabilidad computacional y la experiencia del usuario, pudiendo éstos ser ejecutados de manera programada en cualquier momento.

ANEXO Q

Discusión

Este anexo presenta una discusión acerca del uso esperado de lo propuesto en este trabajo, los resultados de precisión obtenidos por el método computacional para la evaluación de competencias y el sistema de recomendaciones. Finaliza con las limitaciones identificadas.

Q.1 Uso esperado de lo propuesto

Dada la importancia de las competencias en los sistemas educativos actuales, es relevante el desarrollo de herramientas computacionales efectivas para apoyar su enseñanza y aprendizaje.

En primer lugar, el esquema de metadatos provee elementos de descripción específicos. Esto permite que puedan ser aplicadas diversas técnicas de “*machine learning*”, minería de datos y web semántica en el desarrollo de sistemas para: 1) el aprendizaje adaptativo, de tal forma que un grupo de recursos humanos y *Objetos de Aprendizaje* (OA) se asignen de acuerdo al rendimiento de los estudiantes en las competencias; 2) la recuperación información, que sean capaces de buscar y sugerir competencias y OA manual o automáticamente. Esto es posible porque el esquema incluye aspectos pedagógicos útiles para razonar acerca del rendimiento del estudiante, por ejemplo, generando perfiles de usuario directamente relacionados con los verbos, conocimientos, habilidades, temas, edades y competencias en general, tal cual lo hace el SR propuesto. En general, el esquema es apropiado para establecer relaciones lógicas entre competencias y OA. Por lo tanto, con el propósito de sacar ventaja de las posibilidades mencionadas, es ideal que a futuro se generen repositorios y que éstos sean integrados al currículo educativo de los cursos en línea y presenciales, por medio de herramientas de anotación.

Con respecto a la ontología, se espera que ésta sea utilizada en entornos de *Datos Abiertos Enlazados* (DAE). Así, los conceptos en torno a las competencias podrán ser publicados, referenciados e interconectados para que puedan ser explorados y descubiertos por personas o máquinas. Al igual que el SR propuesto, se espera que

se desarrollen otros sistemas centrados en el descubrimiento de conocimiento asociado al currículo educativo.

En tercer lugar, se espera que el método computacional y su prototipo sean integrados a procesos educativos formales, lo que facilitaría el cómputo de niveles de competencia lingüísticos en el marco de cursos en línea y presenciales, especialmente cuando las competencias se vinculan al desarrollo de mapas de competencias. Para esto, se podría integrar a esta idea, herramientas de anotación basadas en el esquema de metadatos o la ontología.

Finalmente, cabe mencionar que Youtube provee una gran cantidad de canales y contenidos de video útiles para apoyar procesos de enseñanza y aprendizaje, por lo cual es una fuente importante de conocimiento para entornos educativos. Sin embargo, esto puede producir problemas relacionados con la *sobrecarga de información* o la falta de *metacognición*, lo cual implica el uso de una gran cantidad de tiempo en la búsqueda de recursos interesantes. Por lo tanto, el SR propuesto es una herramienta necesaria porque es capaz de asistir a docentes y estudiantes, ofreciendo contenidos de video adaptados a sus necesidades y centrados en el currículo educativo. Adicionalmente, el sistema les permite focalizarse en otros aspectos de la educación porque les ahorra tiempo y esfuerzo en la búsqueda de información efectiva.

La alta precisión del SR puede motivar a profesores, instituciones y creadores de contenidos de video a formalizar y reutilizar colaborativamente las descripciones de competencias, mapas y contenidos en aplicaciones basadas en el esquema y la ontología propuestos. Esto podría incrementar la visibilidad de los recursos y sus creadores en plataformas web de video. De hecho, actualmente hay una baja visibilidad de la categoría educativa en comparación con otras (por ejemplo, juegos, tecnología, música, etc.) en Youtube. Esto se apoya en un estudio comparativo que se realizó entre los cinco mejores canales educativos de Youtube y los de otras categorías en cuanto a número de suscriptores y vistas. Los resultados se presentan en la Figura Q.1. Al respecto, el SR podría incrementar la visibilidad de la categoría educativa, alentando a youtubers a crear más contenidos con mayor calidad.

Q. Discusión

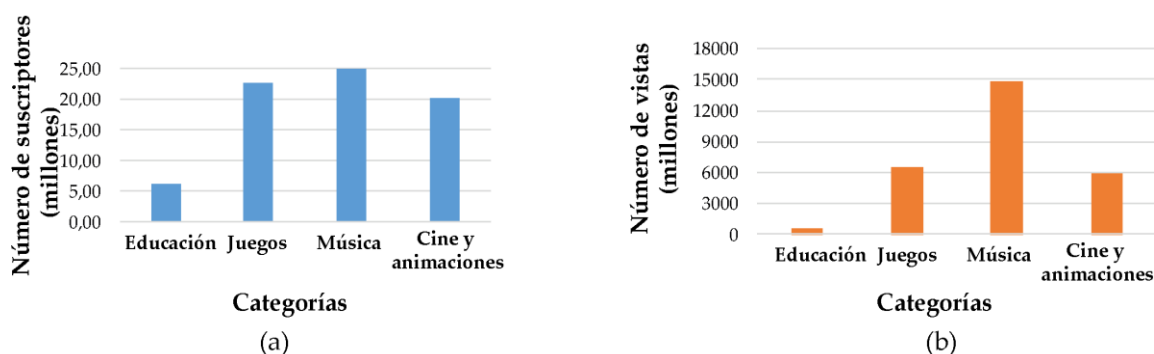


Figura Q.1 Comparación entre categorías de Youtube con base en: (a) el número de suscriptores; (b) el número de vistas. Fuente: propia.

Q.2 Resultados de precisión

El método computacional para la evaluación de competencias presentó unos resultados de precisión altos. Sin embargo, no hubo una concordancia del 100% con relación a las calificaciones provistas por los docentes. Esto se debe a la subjetividad inherente a la evaluación lingüística, la cual es más crítica en casos específicos, por ejemplo, en los que se agregan una gran cantidad de niveles de competencia y pesos de diversos valores. En tales casos, es posible que el juicio de los docentes sea alterado en el proceso de calificación y, en consecuencia, los *Niveles de Competencia* (NC) no sean los adecuados. Por lo tanto, los NC provistos por el método computacional pueden ser más confiables en tales casos, así no sean acordes con el juicio de los docentes.

Por otra parte, el SR propuesto también provee sugerencias con un alto nivel de precisión. Sin embargo, después de analizar la colección de datos, se observó que la precisión fue afectada por la ausencia de relaciones entre algunas competencias y contenidos de video en la base de conocimiento. Por lo tanto, en las listas de recomendaciones se incluyeron contenidos con bajos pesos. Este inconveniente puede ser abordado impulsando la creación de repositorios de descripciones de competencias, mapas y contenidos, tal como se describió en la Sección Q.1.

Q.3 Limitaciones

Todo lo propuesto en el presente trabajo está basado en la definición de competencias, por lo cual no puede ser usado en entornos no enfocados en ellas. Adicionalmente, el funcionamiento del prototipo del SR indica que el tiempo de ejecución puede ser bastante alto en aplicaciones reales. Sin embargo, dado que las recomendaciones pueden ser requeridas en momentos puntuales del ciclo educativo (por ejemplo,

cuando se actualiza el historial de NC de los estudiantes), los problemas asociados al tiempo de ejecución pueden no ser críticos. Una estrategia es obtener recomendaciones automáticamente al detectar tales actualizaciones. Así, las recomendaciones podrán ser obtenidas antes de ser requeridas por los usuarios.

ANEXO R

Resumen de productos derivados de la investigación

La investigación derivó en una serie de productos de diferente tipo: artículos, prototipos, bases de datos y bases de conocimiento.

Con respecto a los artículos, se tienen los siguientes:

1. **D. F. Duran**, G. Chanchí, J. L. Arciniegas, "Evaluación de mapas de competencias educativas: una propuesta difusa basada en 2-tuplas," *Revista Ibérica de Sistemas y Tecnologías de la Información*, no. 24, pp. 22-38, 2017. Reconocimiento como el mejor artículo en el X Seminario Internacional de Ciencias de la Computación (SICC 2017), Medellín, 2017.
2. **D. F. Duran**, G. Chanchí, J. L. Arciniegas, "A Fuzzy Approach for Assessing Educational Competencies based on competency Maps," *Communications in Computer and Information Science*, vol. 735, pp. 372-386, 2017.
3. **D. F. Duran**, G. Chanchí, J. L. Arciniegas, S. Baldassarri , "A Semantic Recommender System for iDTV based on Educational Competencies," *Communications in Computer and Information Science*, vol. 689, pp. 47-61, 2016.
4. **D. F. Duran**, G. Chanchí, J. L. Arciniegas, S. Baldassarri, "Sistema de Recomendación Semántico de Programas Educativos de TVdi: una propuesta Basada en Competencias," en *V Jornadas Iberoamericanas sobre Aplicaciones y Usabilidad de la Televisión Digital Interactiva*, La Habana, 2016.
5. **D. F. Duran**, G. Chanchí, J. L. Arciniegas, "An Ontology-based Recommender System of Web Video Resources Focused on Educational Competencies", artículo en revision en *IEEE Transactions on Learning Technologies*.

En la investigación se desarrollaron los siguientes prototipos:

1. *EduCompRS*: prototipo Java del sistema de recomendaciones propuesto que ofrece una interfaz que permite especificar el estudiante objetivo y, de acuerdo a la estrategia de recomendación propuesta, ofrece la posibilidad de configurar tanto los coeficientes de las métricas de ponderación como las regiones de los mapas y los

atributos de las competencias interesantes para el usuario. Éste provee una lista de recomendaciones de contenidos de VoD por cada competencia en la que el estudiante objetivo tiene cierto grado de necesidad. Una descripción más detallada puede hallarse en la Sección 7.1 y Sección O.2 (Anexo O).

2. Prototipo Java de un sistema de recomendaciones que ejecuta una estrategia de recomendación similar a la propuesta para el sistema semántico, con la diferencia de que la ponderación de las asociaciones se realiza a través de un coeficiente sintáctico como Sørensen-Dice. Este prototipo provee una lista de recomendaciones por cada competencia en la que el estudiante objetivo tiene cierto grado de necesidad.
3. *CMA-Tool*: prototipo Java del método computacional propuesto que ofrece una interfaz gráfica que permite configurar la estructura de los mapas de competencias usando la especificación *Formato de Grafos* JSON. Para configurar la operación del prototipo, se introduce el número de estudiantes y posteriormente, se definen uno a uno los niveles de competencias de los estudiantes en cada una de las actividades de evaluación formalizadas en el mapa. Finalmente, el prototipo ejecuta el algoritmo DFS-CA (Sección 5.2.2) propuesto para ofrecer al usuario los niveles de competencia computados para cada una de las competencias del mapa, usando tanto el método computacional propuesto como el MBPE. Además, también para los dos modelos, la herramienta despliega dos gráficos que muestran el tiempo de ejecución y el consumo de memoria.

En cuanto a bases de conocimiento, se construyó una que formaliza 31 estructuras de mapas de competencias, 322 descripciones de competencias, 400 segmentos de video y 181 contenidos de VoD extraídos de Youtube.

Finalmente, se obtuvo una base de datos Mongo que formaliza los perfiles de usuario de 41 estudiantes, las bolsas de palabras de las 322 competencias y 124 vecindarios de términos extraídos de WordNet. De esta base de datos se extrajeron “*datasets*” JSON para cada de los aspectos mencionados.

Anexo S

Versión previa de la extensión del IEEE RCD

Este anexo presenta las diferencias que tiene la versión del IEEE RCD previa con respecto a la presentada como resultado de la investigación en la Sección 3.2, la cual fue evaluada en el ámbito de la Calidad de los Metadatos (CM) con el fin validar la nueva información modelada y determinar mejoras (Sección 3.4).

De acuerdo con la Figura S.1, la versión previa no tenía incluido el elemento `grade`. Éste fue integrado en la versión definitiva del esquema según las sugerencias de los evaluadores.

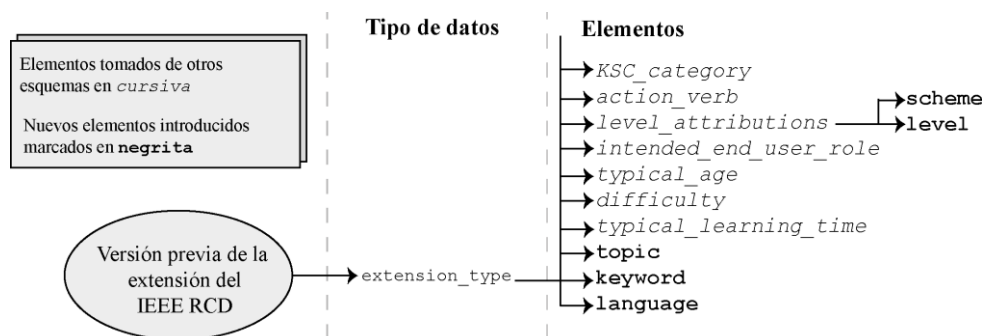


Figura S.1 Versión evaluada de la extensión del IEEE RCD

Por otra parte, tenía definido un vocabulario para el elemento `typical_age` (Tabla S.1) que, por sugerencia de los evaluadores, fue cambiado por el ingreso de valores en texto libre.

Elemento	Vocabulario (años)
<code>typical_age</code>	3, 4, 5, 6-7, 8-9, 10-11, 12-13, 14-16, 17-21, 22-33

Tabla S.1 Vocabulario para el elemento `typical_age`

También, en la evaluación algunos evaluadores manifestaron que tres elementos eran confusos (Tabla S.2). Por tal razón, en la versión propuesta se redefinieron las explicaciones de los elementos `level_attributions`, `scheme` y `level`.

Finalmente, en la evaluación un pequeño grupo de evaluadores (12%) sugirieron introducir el valor para `action_verb` usando texto libre. Como la mayoría de

evaluadores expresaron estar de acuerdo con el vocabulario planteado (78%), entonces se decidió introducir complementariamente un nuevo valor llamado `other`, el cual permite introducir texto libre.

Elemento	Descripción
<code>level_attributions</code>	Elemento con el cual la competencia descrita puede categorizarse en cierto nivel de las escalas reconocidas internacionalmente como la de UNESCO y EQF (European Qualifications Framework).
<code>scheme</code>	Elemento propuesto obligatorio que describe la organización o nombre de la escala que define los niveles en los cuales puede categorizarse la competencia.
<code>level</code>	Elemento propuesto obligatorio que define los niveles en el cual se categoriza la competencia descrita, de acuerdo a <i>LevelScheme</i> .

Tabla S.2 Definiciones de la versión previa del IEEE RCD que fueron modificadas en la versión propuesta

La versión definitiva del esquema propuesto, incluyendo los elementos de datos, su definición, vocabularios y nomenclatura se pueden consultar en la Sección 3.2 y Anexo F de manera complementaria.