# Universidad de Cuenca


**Facultad de Ingeniería - Escuela de Sistemas**

# Aplicación de Tecnologías Semánticas y Minería de Textos para la Comparación de Sílabos entre Distintas Universidades

Proyecto de titulación para obtener el grado de

**Ingeniería de Sistemas**

presenta:

**ESTEBAN SEBASTIÁN ESPINOZA ABRIL - CI: 0106547243**

**NOEMI ELIZABETH SARI UGUÑA - CI: 0105345128**

director:
**Ing. Victor Hugo Saquicela, PhD - CI: 0103599577**

Cuenca - Ecuador. Abril 2018

# Resumen

Actualmente cada una de las Instituciones de Educación Superior (IES) del Ecuador gestionan el contenido académico de las asignaturas mediante un sistema informático de sílabos. En este sistema se tiene diferentes niveles de heterogeneidad, por ejemplo; datos estructurados, en diferentes formatos, entre otros. Esta heterogeneidad genera problemas a la hora de llevar a cabo la movilidad estudiantil, los cuales dificultan la homologación de asignaturas, pues ésta se realiza de forma totalmente manual de modo que puede ocasionar errores en el proceso de homologación.

Para solucionar el problema antes mencionado, en este trabajo se propone la utilización de tecnologías de la Web Semántica para la creación de una ontología de sílabos, la ontología será poblada con los datos que residen en los sistemas de información de la Universidad de Cuenca y se guardará en un repositorio. Posteriormente, sobre este repositorio se aplicará técnicas de minería de textos con el fin de identificar similitudes en el contenido académico de diferentes sílabos, de manera que se automatice el proceso de comparación de sílabos para la movilidad estudiantil y minimizar los errores en el proceso.

**Palabras Clave:** Educación Superior, Web Semántica, Ontología, Minería de Textos.

# Abstract

Nowadays, the IES (by its acronym in spanish "Instituciones de Educación Superior") from Ecuador manage the academic content information of the college subjects, by using a syllabus computing system. In this system they have with different heterogeneity levels of data (e.g., not-structured data, differents formats). This heterogeneity produces troubles at the time of carrying out the undergraduate mobility, which difficult the college subjects homologation, because it is done in totally manual form, so that it can cause mistakes in the homologation process.

To solve the problem above mentioned, in this work we propose the utilization of Web Semantic technologies for the creation of a syllabus ontology, the ontology will be instantiated of data that reside on the University of Cuenca information systems and saved in a repository. Subsequently, on this repository we will apply text mining techniques with the target of identify similarities on the academic content of several syllabus, so that we automate the process of syllabus comparison for the undergraduate mobility and minimizing mistakes in in the process.

**Keywords:** College Education, Semantic Web, Ontology, Text Mining.

# Índice general

# Índice de Figuras

# Índice de Tablas

## Cláusula de Propiedad Intelectual

Esteban Sebastián Espinoza Abril, autor del trabajo de titulación "Aplicación de Tecnologías Semánticas y Minería de Textos para la Comparación de Sílabos entre Distintas Universidades", certifico que todas las ideas, opiniones y contenidos expuestos en la presente investigación son de exclusiva responsabilidad de su autor.

Cuenca, 9 de Mayo 2018

Esteban Sebastián Espinoza Abril

C.I: 0106547243

## Cláusula de licencia y autorización para publicación en el Repositorio Institucional

Esteban Sebastián Espinoza Abril en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de titulación "Aplicación de Tecnologías Semánticas y Minería de Textos para la Comparación de Sílabos entre Distintas Universidades", de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN reconozco a favor de la Universidad de Cuenca una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos.

Asimismo, autorizo a la Universidad de Cuenca para que realice la publicación de este trabajo de titulación en el repositorio institucional, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Cuenca, 9 de Mayo 2018

Esteban Sebastián Espinoza Abril

C.I: 0106547243

## Cláusula de Propiedad Intelectual

Noemi Elizabeth Sari Uguña, autor del trabajo de titulación "Aplicación de Tecnologías Semánticas y Minería de Textos para la Comparación de Sílabos entre Distintas Universidades", certifico que todas las ideas, opiniones y contenidos expuestos en la presente investigación son de exclusiva responsabilidad de su autor.

Cuenca, 9 de Mayo 2018

Noemi Elizabeth Sari Uguña

C.I: 0105345128

## Cláusula de licencia y autorización para publicación en el Repositorio Institucional

Noemi Elizabeth Sari Uguña en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de titulación "Aplicación de Tecnologías Semánticas y Minería de Textos para la Comparación de Sílabos entre Distintas Universidades", de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN reconozco a favor de la Universidad de Cuenca una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos.

Asimismo, autorizo a la Universidad de Cuenca para que realice la publicación de este trabajo de titulación en el repositorio institucional, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Cuenca, 9 de Mayo 2018

Noemi Elizabeth Sari Uguña

C.I: 0105345128

# Dedicatoria

En primer lugar agradezco a toda mi familia, en especial a mi madre y mi abuela, quienes me han apoyado y me motivado a ser mejor cada día. Continuo con mis compañeros y todas las amistades que he formado a lo largo de mi vida universitaria, pues con cada uno de ellos he compartido muchos buenos momentos y establecido un gran sentimiento de estima. Finalmente a mis profesores, por transmitirme su conocimiento y motivación durante el estudio de la carrera.

Esteban Sebastián

# Dedicatoria

A Dios, por haberme dado la salud para cumplir mis objetivos.

A mis padres, Eliseo y María que cada momento de mi vida me brindan su apoyo incondicional, con su ejemplo y dedicación me han inspirado a ser cada día mejor. A mis hermanos con quienes puedo contar en cualquier circunstancia que se presente en la vida.

A mis compañeros y amigos con quienes he compartido experiencias y momentos a lo largo de nuestra vida universitaria, con quienes nos hemos apoyado mutuamente en nuestra formación profesional.

Finalmente, a mis profesores, aquellos que marcaron cada etapa de nuestro camino universitario, por impulsar el desarrollo de nuestra formación profesional.

Noemi Elizabeth

# Agradecimientos

El presente proyecto de titulación no hubiera sido posible sin la ayuda de nuestro director, Ing. Victor Saquicela y al ayudante de investigación de la Universidad de Cuenca, Ing. Fernando Baculima. A quienes hacemos un especial agradecimiento por su apoyo y ayuda incondicional a lo largo del desarrollo del presente trabajo.

Agradecemos también a nuestros amigos y compañeros que nos acompañaron durante toda la carrera universitaria, dado que cada uno de ellos ha brindado un gran apoyo para culminar este objetivo con éxito.

Finalmente, agradecemos a nuestras familias y todas las personas que directa o indirectamente nos han apoyado en la realización del presente trabajo de titulación y la culminación de nuestra carrera universitaria.

Esteban Espinoza
Noemi Sari

# Capítulo 1

# Introducción

Durante los últimos años se han llevado a cabo varias investigaciones con propuestas de iniciativas tecnológicas en el campo de la educación con el objetivo de mejorar los procedimientos la enseñanza y aprendizaje en las Instituciones de Educación Superior(IES). En Ecuador, las IES están regidas por el Reglamento del Régimen Académico (RRA), el cual especifica que cada IES se componen por unidades académicas, cada unidad académica por carreras, a su vez cada carrera está compuesta por asignaturas y cada asignatura se define en diferentes niveles de heterogeneidad, por ejemplo, datos estructurados, en diferentes formatos, entre otros.

Esta heterogeneidad hace que cada IES mantenga su propio formato y modelo de datos de los contenidos de las asignaturas. Además, existe una falta de herramientas, métodos o procedimientos que permitan identificar similitudes entre contenidos académicos de las diferentes carreras, por este motivo hace que la movilidad estudiantil se convierta en una tarea compleja debido a las dificultades del reconocimiento de créditos.

En este trabajo se propone el uso de aplicaciones de tecnologías semánticas y minería de texto en conjunto, permitiendo la creación de la ontología para sílabos y aplicación de algoritmos de minería ofreciendo soluciones en la detección de similitudes de contenidos que permiten tratar.

## 1.1. Identificación del problema

La Ley Orgánica de Educación Superior, señala que los estudiantes tienen derecho a acceder, movilizarse, egresar y titularse conforme a sus méritos académicos. En el caso particular de la movilidad estudiantil, esta misma ley establece que las IES podrán reconocer créditos aprobadas en otras instituciones del sistema de educación superior, ajustándose al cumplimiento de los requisitos establecidos por la entidad de educación elegida.

El proceso de reconocimiento de créditos dentro de la Universidad de Cuenca (cambio de carrera), como también de estudiantes que provienen de otras universidades nacionales o extranjeras; es actualmente llevado a cabo de manera manual mediante un protocolo que involucra al Profesor Fiscal de la Facultad, el Director de Carrera receptora y en algunos casos docentes, quienes de forma individual o colectiva revisan los programas o sílabos de las asignaturas aprobadas por el estudiante, para compararlos con los sílabos de la carrera receptora y emitir un informe en el cual se encuentre los resultados de las comparaciones realizadas.

El método descrito anteriormente, posee ciertos problemas, entre ellos se encuentra el tiempo requerido para comparar cada uno de los sílabos de las asignaturas aprobadas por el estudiante con los sílabos de la carrera receptora. Este hecho ralentiza el proceso de movilidad estudiantil, en especial cuando se tiene una cantidad considerable de estudiantes que deseen realizarla. También, existe una gran probabilidad que el Profesor Fiscal de la Facultad se equivoque al momento de realizar la comparación entre dos sílabos por el hecho que es un proceso manual. A su vez existe el problema de la limitación tecnológica que ayude al Profesor Fiscal de la Facultad y al Director de Carrera receptora a completar el proceso eficientemente.

## 1.2. Justificación

Para dar solución al problema mencionado se propone crear una ontología para el dominio de sílabos que permitirá representar el contenido académico de cualquier asignatura, dentro del plan de estudios de una carrera perteneciente a una IES del Ecuador.

Por otra parte, también se propone el uso de técnicas de minería de textos para descubrir patrones de comportamiento en los contenidos académicos de carrera, de tal manera que se pueda automatizar la comparación entre sílabos

para mejorar el proceso de movilidad estudiantil.

La creación de la ontología y la aplicación de técnicas de minería de textos permitirá minimizar el tiempo de revisión o comparación entre sílabos y los resultados en cuanto a la semejanza obtenida satisfagan un nivel acorde a los requisitos establecidos relacionados con la movilidad estudiantil por la Universidad de Cuenca.

La principal razón por la que se propone crear una ontología y aplicar minería de textos sobre ella, es porque al realizar revisiones sobre papers o tesis actuales, no se encontró soluciones tecnológicas específicas para resolver el problema antes mencionado.

## 1.3. Alcance

En el presente trabajo se utilizará información estructurada del contenido académico de las carreras de la Universidad de Cuenca, referentes a las asignaturas impartidas en ella. Esta información se encuentra en dos fuentes de información como son; bases de datos y archivos PDF. El primer paso es trabajar con las fuentes de datos aplicando un proceso ETL([1]) con el fin de limpiarlos y filtrarlos para obtener datos listos para su uso.

Como siguiente paso es crear una ontología que permita representar la jerarquía de conceptos que se manejan en los sílabos y utilizar la herramienta desarrollada por la Universidad de Cuenca llamada LOD-GF que permite vincular los datos con la ontología creada mediante el proceso de RDF-Ization. Finalmente se aplicarán algoritmos de minería de textos que permitan descubrir patrones de comportamiento en las instancias de la ontología de sílabos, de manera que se pueda automatizar su comparación y obtener similitudes entre ellos.

## 1.4. Objetivos

### 1.4.1. Objetivo general

Mejorar el proceso de comparación y homologación de asignaturas en base a los sílabos de las carreras, con el fin mejorar el proceso de movilidad estudiantil

[1] Siglas en inglés de los términos *Extract, Transform and Load*, en español Extracción, Transformación y Carga.

dentro de la Universidad de Cuenca y de estudiantes que provienen de otras universidades nacionales o extranjeras.

### 1.4.2. Objetivos específicos

- Crear o extender una ontología para el dominio de los sílabos universitarios basada en la investigación de otras similares.
- Usar las técnicas de minería de textos sobre la ontología de sílabos para comparar la semejanza del contenido académico impartido por dos asignaturas similares.
- Validar los resultados de la comparación para determinar el nivel de éxito de la ontología y la minería de textos aplicada.
- Mejorar la ontología y el proceso de minería de textos hasta que los resultados cumplan con un nivel de éxito adecuado al contexto universitario.
- Crear un prototipo que permita la automatización del proceso de verificación.

## 1.5. Antecedentes

La web semántica como marco de referencia para la compartición de datos ha ganado terreno en el campo del *e-learning* en los últimos años. La creación de ontologías dentro del dominio educativo para representar no solo la estructura organizacional, sino también el contenido que se imparte en las instituciones educativas proporciona nuevas formas de vincular la información. Este hecho ha tenido varios objetivos educacionales, uno de ellos es el descubrimiento de nuevo conocimiento que se encuentre implícito en la información educacional. A continuación, se describen algunos trabajos relacionados con el campo de la web semántica especialmente en el dominio de sílabos universitarios.

El trabajo realizado por Chung y Kim [7] presenta un modelo ontológico de curriculums y sílabos en la educación superior, el cual está diseñado para integrar estas estructuras de conocimiento conjuntamente con los sujetos de aprendizaje y materiales. La estructura de clases de esta ontología es bastante completa pues representa bien la estructura organizacional y de contenido. Sin embargo, no contempla el contenido de aprendizaje como lo es un capítulo, subcapítulo o estrategias. Dado que el contenido académico es muy importante para los sílabos de la Universidad de Cuenca, la ontología que se creará tendrá en cuenta no solo la estructura organizacional (facultades, carreras, profesores,

coordinadores) sino también la académica (objetivos, logros, capítulos, subcapítulos, recursos, bibliografía).

Por otra parte el trabajo realizado por Eremin [8] describe el proceso de creación de una ontología que permita estructurar material y temas para cursos educativos. El principal problema que se desea resolver en este trabajo es la rápida renovación de conocimientos que existe en las disciplinas informáticas y los cambios constantes que provoca sobre los sílabos de estas disciplinas. Por lo cual, la construcción de una ontología es una solución informática que permite realizar cambios rápidos y oportunos en el contenido y materiales de una disciplina. A pesar de que este trabajo tiene mucha relación con el presente proyecto, la mayor diferencia es que éste no se centra en los materiales y contenido educativo. Al tener un ámbito más general la ontología de sílabos de la Universidad de Cuenca favorece no solo el cambio de contenido de acuerdo a los avances académicos, también facilita la ejecución algoritmos de minería de textos con el fin de obtener conocimiento implícito dentro de los sílabos.

En contraste el trabajo de Chicaiza et al. [9] tiene como principal finalidad optimizar las tareas de los profesores y el uso de material educativo por medio de un vocabulario ontológico abierto, que habilite el procesamiento inteligente de datos sobre cursos tipo *Open CourseWare* (por sus siglas en inglés OCW), siendo OCW una iniciativa de aprendizaje para estudiantes de educación superior. Este vocabulario se basa y extiende ontologías como FOAF, BIBO, AIISO, TEACH, LOCWD entre otras; lo cual permite describir conceptos de la academia como sílabos, contenidos, criterios de evaluación, bibliografía, etc. Como trabajo de investigación es el que tiene mayor similitud al presente proyecto de titulación en cuanto a las ontologías utilizadas y el nivel de detalle de conceptos que se describen. Al igual que con el trabajo expuesto anteriormente la mayor diferencia se encuentra en el uso de la ontología, pues nuestra propuesta requiere la ejecución de algoritmos semánticos sobre las instancias.

## 1.6. Estructura del documento

A continuación, se detalla la estructura del contenido del presente proyecto que se divide en 5 capítulos.

- **Capítulo 1.- Introducción:** Presentación del proyecto de titulación incluyendo la problemática, justificación, alcance, objetivos y antecedentes.
- **Capítulo 2.- Marco Teórico:** Explicación de los conceptos, funda-

mentos y herramientas involucradas en la web semántica y la minería de textos.

- **Capítulo 3.- Proceso de integración de datos, creación de la ontología de sílabos y minería de textos aplicada:** Integración de las fuentes de datos y ejecución procesos ETL. Creación de la ontología de sílabos de la Universidad de Cuenca. Selección y aplicación de algoritmos de minería de textos sobre las instancias de la ontología.
- **Capítulo 4.- Evaluación y resultados:** Evaluación y pruebas sobre los algoritmos de minería de textos sobre la ontología creada.
- **Capítulo 5.- Conclusiones y trabajos futuros:** Presentación de conclusiones generales y futuros trabajos de investigación.

# Capítulo 2

# Marco teórico

En esta sección se presentan los conceptos, fundamentos, herramientas y metodologías en los que se basa el tema de investigación de este trabajo.

## 2.1. Web semántica conceptos, herramientas y metodologías

La Web Semántica es un proyecto del *World Wide Web Consortium* también conocido por sus siglas como W3C, el cual es un organismo creado en el año de 1994 por Sir Tim Berners-Lee y tiene como principal función regular aspectos de la Web como también crear estándares que la normalicen [10].

El W3C describe a la Web Semántica como: "Un marco de referencia común que permite que los datos sean compartidos y reusados a través de aplicaciones, empresas y fronteras comunitarias. Es un esfuerzo colaborativo liderado por la W3C con la participación de un gran número de investigadores y socios industriales."[11]

En base a la definición de la W3C, el estudio previo realizado por Codina et al. [10] la divide en dos visiones separadas que a la vez se pueden complementar. La primera de ellas tiene relación con la inteligencia artificial, por lo cual es ella que está centrado nuestro tema de investigación. Esta definición indica lo siguiente: "La Web semántica es un conjunto de iniciativas destinadas a promover una futura Web cuyas páginas estén organizadas, estructuradas y codificadas de tal manera que los ordenadores sean capaces de efectuar infe-

rencias y razonar a partir de sus contenidos."

Además de proveer una definición para la Web Semántica, el W3C también proporciona una arquitectura estándar que está representada mediante la Figura 2.1. En las subsecciones a continuación se describen algunas las capas de esta arquitectura.

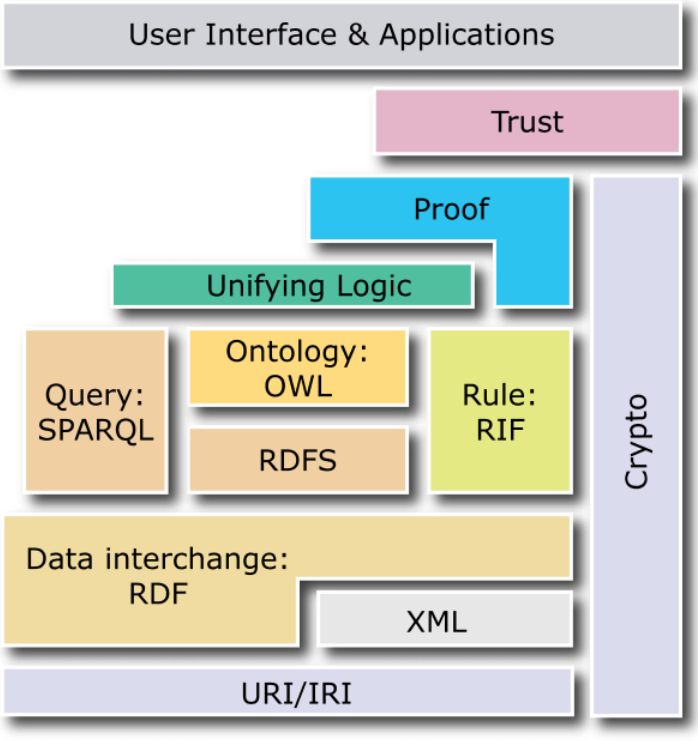



Figura 2.1: Arquitectura en capas de la Web Semántica [1].

#### 2.1.0.1. Unicode y URI

*Unicode* es un estándar que sirve para codificar caracteres de múltiples idiomas sin importar la plataforma o herramienta que lo utilice [10]. *Uniform Resource Identifier* (por sus siglas en inglés URI), es una cadena de caracteres que permite identificar recursos en Internet de manera unívoca. A diferencia del *Uniform Resource Locator* (URL) el URI no necesariamente indica como

acceder al recurso, más bien se dice que un URL es un subconjunto de la URI [10].

#### 2.1.0.2. XML, NS y XML Schema

*eXtended Markup Language* (por sus siglas en inglés XML), es un formato de texto simple y flexible creado a partir de la ISO 8879 SGML (*Standard Generalized Markup Language*) en el año de 1998 por la W3C [12]. Fue diseñado originalmente para la publicación e intercambio de información electrónica a gran escala y actualmente es utilizado conjuntamente con otras tecnologías como XPath, XQuery, XSL, XSD, entre otras [13].

*Name Spaces* (por sus siglas en inglés NS), es un conjunto de nombres recomendado por la W3C. Permite que los elementos de un XML tengan un nombre único dentro de un mismo documento [10].

XML *Schema* es una recomendación de la W3C, que expresa una serie de vocabularios compartidos para especificar tipos de datos, listas de componentes, restricciones, estructura y semántica a un documento XML [10][14].

#### 2.1.0.3. RDF y RDF Schema

*Resource Description Framework* (por sus siglas en inglés RDF), es un modelo estándar para el intercambio de datos en la web [15]. La primera versión de RDF se publicó en el año de 1999 como un lenguaje que permita definir ontologías y metadatos, hoy en día es el estándar más popular y utilizado por la comunidad de la web semántica para representar todo tipo de recursos digitales a lo largo de la web (no se limita solo a sitios web) [16].

La unidad básica para la representación en RDF se conoce como "tripleta", ésta consta de tres elementos: sujeto, predicado y objeto. Tanto el sujeto como el objeto se representan gráficamente mediante nodos, mientras que el predicado se representa mediante un arco dirigido que comienza en el sujeto y termina en el objeto [17]. La Figura 2.2 muestra un ejemplo del diagrama de una tripleta.

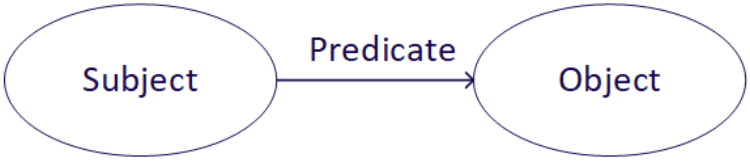



Figura 2.2: Representación gráfica RDF [2].

Además, en RDF existes 3 unidades lógicas, correspondientes a los elementos de una tripleta los cuales son: recursos, propiedades y valores. Los recursos están definidos mediante URI's. Las propiedades son características relevantes que describen a los recursos. Los valores son datos concretos sobre recursos determinados [17]. Con estas 3 unidades lógicas se pueden formar declaraciones como:

- El recurso "X" tiene la propiedad "Y" con un valor igual a "Z".
- El recurso "X" tiene la propiedad "Y" en común con el recurso "W".

No obstante, se debe tener en cuenta casos especiales en los que un recurso no esté definido por una URI, este recurso en una tripleta se lo representa como un nodo en blanco.

Los modelos gráficos en RDF también pueden representarse en líneas de texto, lo que se denomina como serialización. Esta operación utiliza normalmente el formato RDF/XML para definir los recursos, propiedades y valores [10]. En la Figura 2.3 se puede observar un ejemplo del uso de este formato para serialización.

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:si="https://www.w3schools.com/rdf/">

    <rdf:Description rdf:about="https://www.w3schools.com">
      <si:title>W3Schools</si:title>
      <si:author>Jan Egil Refsnes</si:author>
    </rdf:Description>
</rdf:RDF>
```

Figura 2.3: Representación textual mediante el formato RDF/XML.

#### 2.1.0.4. Ontologías, Vocabulario ontológico y OWL

El término ontología proviene de filosofía, sin embargo en el campo de la Inteligencia Artificial la definición de ontología de Castells [16] basada en la de Gruber [18] describe a una ontología como "una jerarquía de conceptos con atributos y relaciones, que define una terminología consensuada para definir redes semánticas de unidades de información interrelacionadas". Las ontologías permiten definir conceptos y relaciones de algún dominio.

Las ontologías son utilizadas en la web semántica para representar una base del conocimiento, los componentes de una ontología según [19] son:

- **Clase:** conjunto de objetos ya sea físicos, tareas, funciones, etc.
- **Relaciones:** Se establecen entre conceptos de una ontología para representar las interacciones.
- **Propiedades:** Los objetos se describen por medio de un conjunto de características o atributos. Estos almacenan diferentes clases de valores. Las especificaciones, rangos y restricciones sobre estos valores se denominan *facets*.
- **Instancias:** Son objetos, miembros de una clase, que no pueden ser divididos sin perder su estructura y características funcionales. Pueden ser agrupados en clases.
- **Axioma:** Elementos que permiten la modelación de verdades que se cumplen siempre en la realidad. Existen tres tipos de axiomas: relacionales, no-relacionales y generales.

Las ontologías se caracterizan porque definen un vocabulario ontológico, el cual contiene términos utilizados para compartir e interpretar conceptos básicos en determinado contexto. De esta manera se posibilita la compartición de información relacionada a un dominio específico a través de la web, convirtiendo a las ontologías en una parte fundamental de la arquitectura de la Web Semántica [20].

*Web Ontology Language* (por sus siglas en inglés OWL) es un lenguaje estándar recomendado por el W3C para la representación de ontologías [10]. Oficialmente el W3C [6] la definió así: "OWL está diseñado para ser usado por aplicaciones que necesiten procesar el contenido de la información en lugar de solo presentar información a los seres humanos. Facilita una mayor interpretación de la máquina del contenido web soportado por XML, RDF, y RDF

*Schema*, proporcionando un vocabulario adicional junto con una semántica formal. OWL tiene tres sublenguajes: OWL *Lite*, OWL DL y OWL *Full*."

OWL es un lenguaje que extiende la descripción de clases y propiedades en un vocabulario ontológico mediante el uso de axiomas que los afecten. Según el nivel de complejidad que se necesite para representar una ontología usando OWL, se debe elegir el sublenguaje más conveniente. Para ello se debe tener en cuenta que mientras más completo sea, mayor cantidad de axiomas posee un sublenguaje [6].

El sublenguaje OWL *Lite* posee los axiomas que se presentan en la Tabla 2.1.

| **RDF Schema** | **Equality** |
|---|---|
| Class (Thing, Nothing)<br>rdfs:subClassOf<br>rdf:Property<br>rdfs:subPropertyOf<br>rdfs:domain<br>rdfs:range<br>Individual | equivalentClass<br>equivalentProperty<br>sameAs<br>differentFrom<br>AllDifferent<br>distinctMembers |
| **Restricted Cardinality** | **Header Information** |
| minCardinality (only 0 or 1)<br>maxCardinality (only 0 or 1)<br>cardinality (only 0 or 1) | Ontology<br>imports |
| **Property Characteristics** | **Property Restrictions** |
| ObjectProperty<br>DatatypeProperty<br>inverseOf<br>TransitiveProperty<br>SymmetricProperty<br>FunctionalProperty<br>InverseFunctionalProperty | Restriction<br>onProperty<br>allValuesFrom<br>someValuesFrom |
| **Annotation Properties** | **Versioning** |
| rdfs:label<br>rdfs:comment<br>rdfs:seeAlso<br>rdfs:isDefinedBy<br>AnnotationProperty<br>OntologyProperty | versionInfo<br>priorVersion<br>backwardCompatibleWthin<br>compatibleWith<br>DeprecatedClass<br>DeprecatedProperty |
| **Class Intersection** | **Datatypes** |
| intersectionOf | xsd datatypes |

Tabla 2.1: Sublenguajes OWL Lite [6].

Los sublenguajes OWL DL y OWL Full poseen los axiomas que se presentan en la Tabla **??**.

| **Class Axioms** | **Boolean Combinations of Class Expressions** |
|---|---|
| oneOf, dataRange<br>disjointWith<br>equivalentClass (applied to class expressions)<br>rdfs:subClassOf (applied to class expressions) | unionOf<br>complementOf<br>intersectionOf |
| **Arbitrary Cardinality** | **Filler Information** |
| minCardinality<br>maxCardinality<br>cardinality | hasValue |

Tabla 2.2: Sublenguajes OWL DL Y OWL Full [6].

#### 2.1.0.5. Lenguaje de consulta SPARQL

SPARQL *Protocol and* RDF *Query Language* (por sus siglas en inglés SPARQL es un acrónimo recursivo), es un estándar definido por la W3C [21] que lo describe como un lenguaje de consultas sobre diversas fuentes de datos, si éstas utilizan un almacenamiento nativo en repositorios semánticos sobre RDF. Además, SPARQL utiliza el formato de serialización *Turtle* para representar su sintaxis. La primera versión de SPARQL fue publicada sobre el año 2008, actualmente se encuentra en la versión 1.1 publicada en el 2013.

SPARQL permite la consulta de patrones obligatorios y opcionales sobre grafos mediante conjunciones o disyunciones respectivas. También soporta agregación, subconsultas, negación y consultas restrictivas un grafo RDF fuente [21].

SPARQL se compone de forma estructural principalmente por los siguientes componentes [22].

- ***PROLOGUE*:** Contiene la declaración de variables, espacios de nombres y abreviaciones usadas en la consulta.

- ***SELECT*:** Selecciona un grupo de variables como resultado de la consulta.
- ***CONSTRUCT*:** Es usado para construir un gráfico RDF usando soluciones obtenidas.
- ***DESCRIBE*:** Es un componente informativo que especifica las URI's referenciadas.
- ***ASK*:** No contiene parámetros y se utiliza en conjunto con la cláusula *WHERE*, retorna *TRUE* si la respuesta de la consulta no es vacía y *FALSE* si es vacía.
- ***FROM*:** Especifica las fuentes o conjuntos de datos requeridos para la consulta.
- ***WHERE*:** Indica los patrones que permiten filtrar la información de las fuentes de datos. Es el componente central de la consulta y permite usar URI's, nodos en blanco, filtros, operaciones de unión, operadores opciones, entre otros.

### 2.1.1. Herramientas

En la evolución de la web semántica se encuentran varios formatos, modelos o lenguajes como XML, RDF y OWL. Para poder trabajar con ellos se puede hacer uso de diversas herramientas que permitan la creación y visualización de los resultados obtenidos. Por esta razón en esta subsección se describen algunas herramientas utilizadas en este proyecto de titulación.

*Protégé*[1], es un editor de código abierto que permite construir ontologías simples o complejas, utilizando el lenguaje OWL para modelarlas. Permite la descarga e instalación de *plugins* para extender el campo de aplicación de una ontología. Esta herramienta se utilizó para la creación de la ontología de sílabos y validación mediante un razonador.

OOPS![2], permite la validación de una ontología a través de un servicio web, su uso es fácil pues solo requiere copiar y pegar el enlace de una ontología o el contenido textual de la misma y pulsar en el botón de escanear, luego se mostrarán los defectos o errores encontrados en dicha ontología. Esta herramienta se utilizó para la validación de la ontología de sílabos mediante iteraciones de

[1]Se puede descargar y encontrar documentación a través de su URL `https://protege.stanford.edu/`.

[2]La URL del servicio web es `http://oops.linkeddata.es/`.

prueba y corrección de errores.

Widoco[3], esta permite la publicación y creación de documentación para una ontología de forma automatizada, para ello utiliza la notación JSON-LD, OOPS! y el framework LODE. Esta herramienta se utilizó para visualizar la ontología de sílabos creada.

*Pentaho Data Integration*[4], es un *software* orientado para el *Business Intelligence* (BI) que permite entre algunas de sus funciones la integración de datos de distintas fuentes, procesos de extracción, transformación y carga de datos también conocido como ETL (por sus significado en inglés *Extract, Transform, Load*) e incluso permite la creación de cubos multidimensionales para la minería de datos. Esta herramienta se utilizó para integrar las fuentes de datos y realizar procesos ETL sobre ellas.

LOD-GF[5], es un *framework* basado en *Pentaho Data Integration* que brinda un entorno gráfico para trabajar con las cinco fases de la metodología de publicación de *Linked Open Data* [3]. La Figura 2.4 muestra los componentes de la herramienta para cada una de las fases. Esta herramienta se utilizó para instanciar la ontología de sílabos.

[3]La URL de descarga y documentación sobre la herramienta es `https://github.com/dgarijo/Widoco`.

[4]La URL de descarga y documentación sobre la herramienta es `http://www.pentaho.com/product/data-integration`.

[5]La URL de descarga y documentación sobre la herramienta es `https://ucuenca.github.io/lodplatform/`.

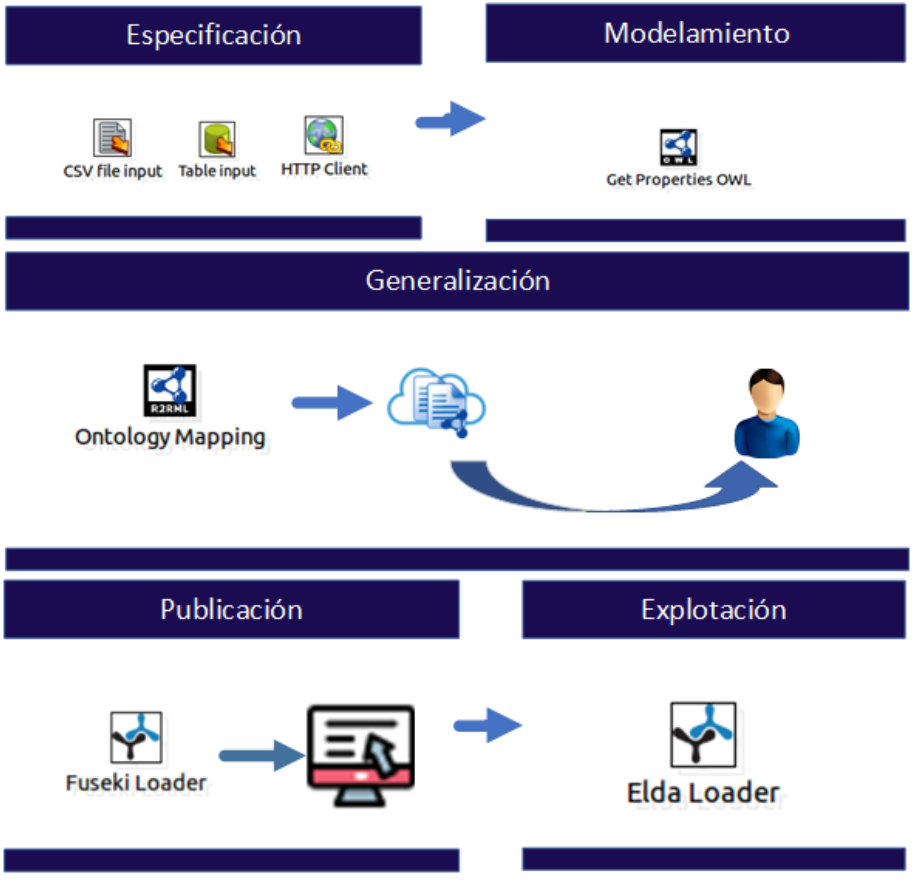



Figura 2.4: Fases de la publicación de datos y componentes de la herramienta [3].

### 2.1.2. Metodologías de desarrollo de ontologías

A continuación se mencionan algunas metodologías utilizadas para la creación de ontologías, las principales o más utilizadas son analizadas y descritas en Luna et al. [23]

#### 2.1.2.1. Metodología CYC:

Fue publicada por Lenat y Guha en el año de 1990, como una de las primeras metodologías. Se divide en dos partes, la primera consiste en extraer

el conocimiento a ser modelado en la ontología de forma manual. El segundo es adquirir nuevo conocimiento utilizando herramientas de procesamiento computacional que lo infieran a partir del implícito [23].

#### 2.1.2.2. Metodología USCHOLD Y KING:

Publicada en el año de 1995 por Uschold y King [24]. Es una metodología que permite crear ontologías sobre un dominio específico mediante cuatro pasos. El primero es identificar el propósito y explicar el por qué la ontología está siendo construida y cules son sus usos [24]. El segundo es construir la ontología mediante la captura de conceptos y relaciones con sus respectiva terminología [23], luego hay que codificar estos conceptos y relaciones en un lenguaje formal determinado. El tercero es la evaluación que permite verificar el funcionamiento de la ontología según su propósito. Por último el cuarto es la documentación la cual propone información sobre el uso de la ontología [24].

#### 2.1.2.3. Metodología GRUNINGER Y FOX:

Fue publicada por Gruninger y Fox [25] al igual que la anterior en 1995. Esta metodología se basa en el uso que se dará a la ontología posteriormente a su construcción. Para ello se define seis pasos: El primero es la descripción de escenarios motivadores con las aplicaciones de la ontología. El segundo es la formulación de preguntas que al responderlas definan los requisitos de la ontología. El tercero es la especificación de terminología en base a los requisitos, que permita definir objetos, sujetos, atributos y relaciones. El cuarto es la formulación de preguntas formales que al responderlas definan axiomas de la ontología. El quinto es la especificación formal de axiomas que definen a las restricciones y predicados para los objetos de la ontología. Por último el sexto es definir teoremas sobre los cuales las soluciones estarán atadas a ciertas condiciones [25].

#### 2.1.2.4. Metodología KACTÜS:

Esta metodología se definió como parte del proyecto *Esprit* KACTUS [26] en 1995 y se compone de cuatro pasos. El primero es especificar el contexto de la aplicación, en el cual se describe el dominio de la ontología en base a las metas que se desea conseguir con la ontología. El segundo es el diseño preliminar basado en categorías ontológicas, lo cual implica buscar ontologías similares para extenderlas o reutilizarlas [26]. Por último, el tercero es el refinamiento y estructuración de la ontología en base a los ontologías similares encontradas [23].

#### 2.1.2.5. Metodología Noy & McGuinness:

Esta metodología se publicó en el año 2001 por Noy y McGuinness [19], utiliza la herramienta *Protégé* para su implementación y se compone de 7 pasos descritos a continuación. El primero es determinar el dominio y alcance de la ontología mediante preguntas como: ¿Cuál es el dominio que la ontología cubrirá?, ¿Para qué se desarrolla la ontología? ó ¿Quién usará la ontología?. El segundo es reutilizar ontologías existentes considerando fuentes de conocimiento para el dominio. El tercero es enumerar términos importantes en la ontología con el fin de obtener una lista de palabras acerca del dominio que puedan ser usados para definir conceptos y propiedades. El cuarto es definir clases y jerarquías de clases por medio de los términos encontrados en el paso anterior para construir una generalización o especificación de clases. El quinto es definir propiedades de las clases describiendo la estructura interna de los conceptos. El sexto es definir las características que describen a las propiedades como los tipos de valores permitidos, la cardinalidad, el dominio y rango. Por último, el séptimo es crear las instancias de las clases y propiedades para comprobár que la ontología sea correcta [27].

#### 2.1.2.6. Metodología NeOn:

Esta metodología se publicó en el año 2010 por Suárez-Figueroa y Gómez-Pérez [3], la metodología propone nueve escenarios para llevar a cabo la construcción de una ontología. Los nueve escenarios son los siguientes:

- **Escenario 1:** Especificación de requisitos, indica que los desarrolladores de la ontología deben crearla desde cero sin reutilizar recursos de conocimiento existentes, para lo cual se especifica los requerimientos que la ontología debe cumplir.
- **Escenario 2:** Reutilización y reingeniería de recursos no ontológicos (NORs[6]), en el cual los desarrolladores deben llevar a cabo un proceso para decidir los NOR a utilizar de acuerdo a los requerimientos.
- **Escenario 3:** Reutilización de recursos ontológicos, donde los desarrolladores deben utilizar distintos recursos ontológicos existentes para crear redes ontológicas, para ellos se puede emplear toda una ontología existente o solo parte de ella.
- **Escenario 4:** Reutilización y reingeniería de recursos ontológicos, donde los desarrolladores deben reutilizar y reorganizar los recursos ontológicos para poder integrarlos con las redes ontológicas creadas.

[6] Siglas en inglés del término NonOntological Resources, en español Recursos No Ontológicos.

- **Escenario 5:** Reutilización y fusión de recursos ontológicos, donde los desarrolladores deben conectar los recursos ontológicos que pertenecen al mismo dominio para fusionarlos en un solo recurso.
- **Escenario 6:** Reutilización, fusión y reingeniería de recursos ontológicos, donde los desarrolladores reutilizan, combinan los recursos ontológicos fusionados previamente.
- **Escenario 7:** Reutilización de patrones de diseño ontológicos, donde los desarrolladores acceden a repositorios[7] de ODPs[8] con el fin de reutilizar patrones sobre las redes ontológicas.
- **Escenario 8:** Reestructuración de recursos ontológicos, donde los desarrolladores modularizan, podan, especializan o extienden los recursos ontológicos para integrarlos en las redes ontológicas.
- **Escenario 9:** Localización de recursos ontológicos, donde los desarrolladores adaptan la ontología a varios lenguajes y comunidades culturales con el fin de que la ontología creada sea multilingüe.

En la Figura 2.5 se muestra un resumen de los nueve escenarios propuestos por la metodología NeOn.

[7] `http://ontologydesignpatterns.org`.

[8] Siglas en inglés del término *Ontology Design Patterns*, en español significa Patrón de Diseño Ontológico.

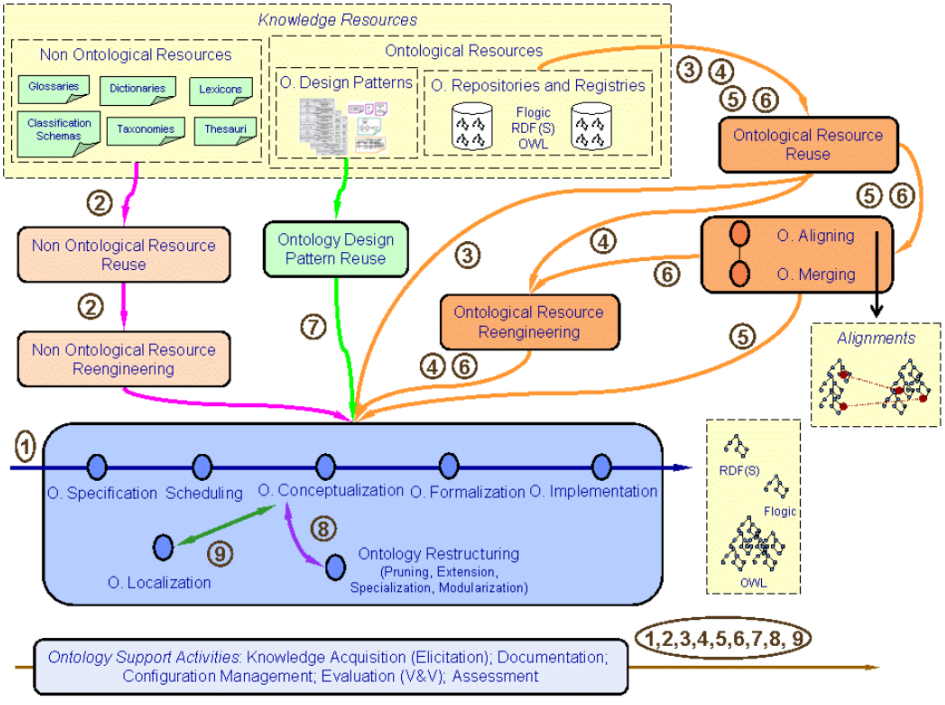



Figura 2.5: Escenarios de la construcción de ontologías de la metodología NeOn [4].

### 2.1.3. RDF-Ization

La RDF-ization es un término utilizado para describir el proceso de transformación de la información proveniente de distintas fuentes en tripletas RDF [28]. Este proceso será utilizado para instanciar la ontología de sílabos una vez que haya sido creada mediante el uso de la herramienta LOD-GF.

## 2.2. Minería de textos, conceptos, fundamentos, algoritmos y herramientas

La Minería de Textos fue definida por Fledman et al. [29] como *Knowledge Discovery from Text* (por sus siglas en inglés KDT), pues refiere al proceso de análisis de texto para descubrir conocimiento implícito y previamente desconocido en documentos con datos estructurados, semiestructurados o no

estructurados.

La minería de textos también puede ser definida como un proceso de conocimiento intensivo en el cual un usuario interactúa con una colección de documentos [30], utilizando tecnologías para el análisis y procesamiento de información textual sea semiestructurada como documentos XML y JSON o no estructurada como documentos de texto [5].

En la actualidad la minería de textos es un campo interdisciplinario pues abarca varias áreas de conocimiento en el ámbito de las Ciencias de la Computación. La Figura 2.6 muestra los campos y áreas que intersectan la minería de textos.

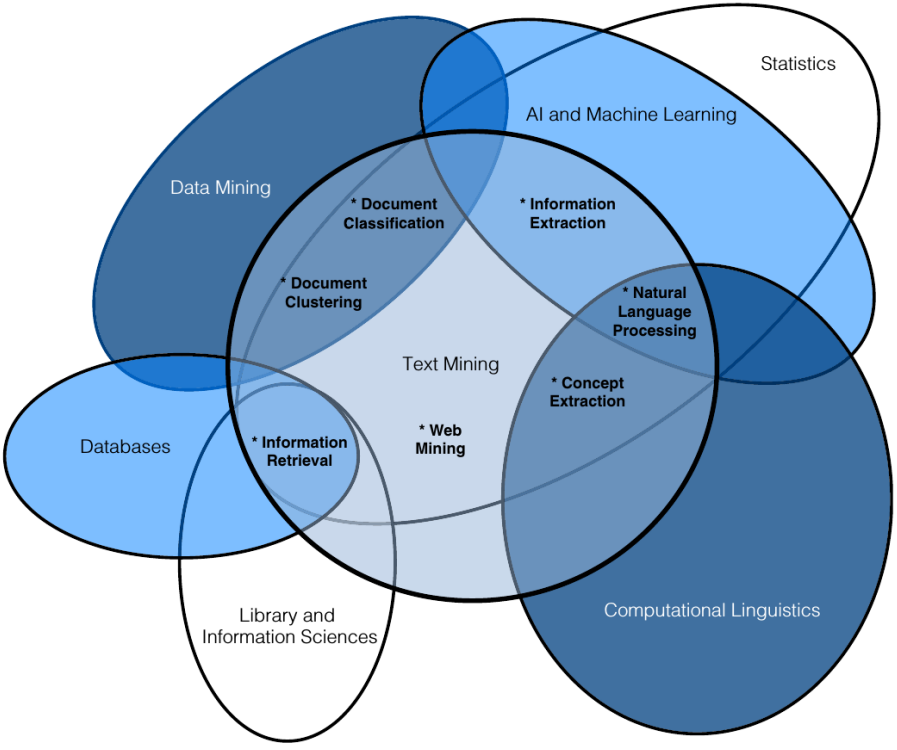



Figura 2.6: Diagrama de Venn de la minería de textos, los seis campos con los que intersecta se muestran como óvalos y las siete áreas que abarca con cada campo [5].

### 2.2.1. Áreas de la minería de textos

La minería de textos abarca 7 áreas prácticas que se encuentran altamente interrelacionadas, sin embargo presentan diferencias según el enfoque del problema que se requiera solucionar [5]. A continuación se indica una breve descripción de cada una.

- ***Information Retrieval* (IR):** La recuperación de información (por sus siglas en inglés IR) permite recuperar y almacenar documentos [5] que se encuentren dentro de colecciones de datos no estructurados, por lo cual esta área está enfocada en facilitar el acceso a la información, mas no en el análisis y el descubrimiento de patrones [31].
- ***Document clustering*:** La agrupación de documentos permite agrupar y categorizar términos, fragmentos, párrafos o documentos completos; utilizando algoritmos de agrupación en el campo de la minería de datos [5].
- ***Document classification*:** La clasificación de documentos permite agrupar y categorizar términos, fragmentos, párrafos o documentos completos; utilizando algoritmos de clasificación en el campo de la minería de datos [5].
- ***Information extraction* (IE):** La extracción de información (por sus siglas en inglés IE) permite identificar y extraer los hechos relevantes y relaciones de un texto no estructurado o semiestructurado [5]. Es considerado como el punto de inicio para la aplicación de otros algoritmos de minería de textos [31].
- ***Natural language processing* (NLP):** Procesamiento de lenguaje natural (por sus siglas en inglés NLP) tiene como objetivo el entendimiento del lenguaje natural, utilizando para ello algoritmos de inteligencia artificial y la ciencia lingüística [31].
- ***Concept extraction*:** La extracción de conceptos permite agrupar palabras y frases dentro de grupos similarmente semánticos [5].
- ***Web mining*:** Es la aplicación de minería de textos y de datos sobre grandes volúmenes de información en la web [5].

### 2.2.2. Preguntas para encontrar el área correcta

Con el fin poder identificar las áreas involucradas en el problema que se requiera solucionar según [5], se debe responder a máximo 4 de las 5 preguntas descritas a continuación.

- **Granularidad:** La primera pregunta permite responder a la granularidad (nivel de detalle de un documento) que se utilizará para aplicar los algoritmos de minería de textos. Por lo cual la pregunta es: ¿El objetivo es agrupar palabras o documentos? [5].
- **Enfoque:** Sin importar que la granularidad sea palabras o documentos lo siguiente es encontrar el enfoque adecuado para la información, es decir si se necesita recuperarla (IR) o extraerla (IE). Para ello se debe responder a la pregunta: ¿Se requiere encontrar palabras y documentos específicos o caracterizar todo el conjunto?. En caso de que se requiera palabras y documentos aislados entonces el problema pertenece al área IE y si se requiere todo el conjunto entonces el problema pertenece al área IR [5].
- **Información disponible:** Si al responder la granularidad o el enfoque se tiene interés en el conjunto de documentos, entonces se debe decidir el tipo de algoritmos a aplicar según la información disponible. Los algoritmos supervisados requieren datos de entrenamiento y de respuesta, mientras que los algoritmos no supervisados pueden utilizar todo tipo de datos. Por lo cual la pregunta es: ¿Se cuenta con documentos categorizados?. En caso de que no se cuente con categorías, entonces se debe utilizar algoritmos no supervisados por lo contrario se debe utilizar algoritmos supervisados [5].
- **Sintaxis o semántica:** Al contrario de la anterior pregunta si al responder la granularidad o el enfoque se tiene interés en palabras, entonces se debe determinar si el problema corresponde a sintaxis o semántica. La sintaxis trata sobre la estructura de las palabras, mientras que la semántica trata sobre su significado en el contexto. Puesto que el lenguaje natural puede conllevar a ambigüedades la semántica se convierte en un problema difícil de resolver. La pregunta en este caso es: ¿Para resolver el problema se debe considerar el significado del texto o su estructura? [5].
- **Texto web o tradicional:** La Internet provee una gran cantidad de información y documentos. La estructura y estilo de los documentos web los convierten en problemas distintos comparados con los documentos tradicionales. Aunque existan algoritmos para procesar tanto texto tradicional como web, es necesario responder a la pregunta ¿Los documentos son independientes o están conectados mediante hiperenlaces? [5].

La Figura 2.7 muestra un árbol de decisión con las preguntas anteriormente descritas, según las respuestas a dichas preguntas se puede conocer el o las áreas involucradas en la minería de textos que se deben utilizar para resolver

el problema. Además, por cada área se cuenta con tutoriales del libro [5] que permiten conocer ejemplos de resolución de problemas referentes al área de la minería respectiva.

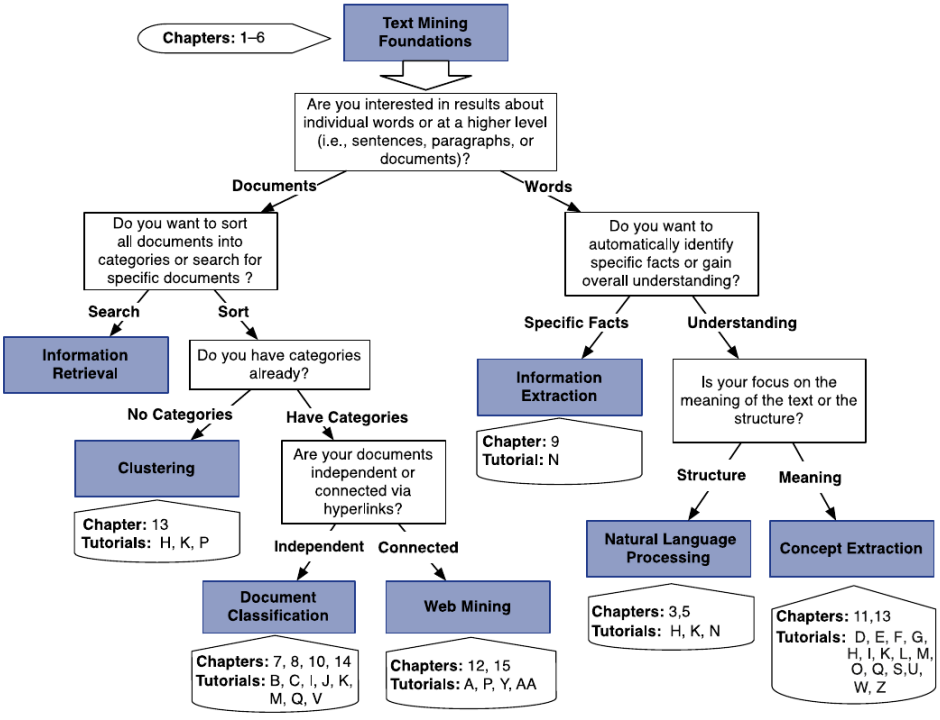



Figura 2.7: Árbol de decisión para encontrar el área de la minería de textos correcta con el fin de resolver el problema [5].

### 2.2.3. Algoritmos de agrupación (*clustering*)

*Clusters* [32] también conocido como segmentación de textos, consiste en la división de los datos en grupos de objetos similares que tienen como finalidad agrupar una colección de objetos en subconjuntos o *clusters* con características similares. Para medir la similitud entre los datos se puede aplicar varias técnicas entre ellas la distancia Euclidiana, de Manhatan, de Mahalanobis, etc. Los algoritmos de agrupamiento pueden dividirse en tres grupos fundamentales: jerarquicos, particionales y basados en densidad [33].

- **Clustering Jerárquico:** Los métodos jerárquicos consiguen la categori-

zación final mediante la separación o la unión de grupos de documentos. Así, estos métodos generan una estructura en forma de árbol en la que cada nivel representa una posible categorización de los documentos. Donde cada vértice de un árbol es un grupo de elementos. En los niveles intermedios cada nodo del nivel n es dividido para formar sus hijos del nivel n+1. Los algoritmos de agrupamiento jerárquicos fueron uno de los primeros enfoques para los problemas de agrupación de documentos.

- **Clustering Particionado:** Son los métodos no jerárquicos también llamados particionales, o de optimización, no producen una serie de grupos anidados, si no que llegan a una única categorización que optimiza el criterio predefinido o función objetivo [33].
- **Algoritmo basados en densidad:** Estos algoritmos usan diversas técnicas que permiten determinar los grupos que pueden ser grafos, se basa en histogramas, kernels, aplica la regla de k-NN, este algoritmo emplea los conceptos del punto central, borde o ruido [33].

Existe una gran variedad de algoritmos de clustering a continuación se describe algunos algoritmos.

#### 2.2.3.1. COBWEB

Es un algoritmo de clustering jerárquico, fue inventado por el profesor Douglas H. Fisher. El algoritmo COBWEB [34] produce un dendrograma de agrupamiento llamado árbol de clasificación donde las hojas representan los segmentos y el nodo raíz engloba por completo el conjunto de datos de entrada. El árbol al inicio contiene un único nodo raíz, las instancias se van agregando una a la vez, y la actualización del árbol se da a cada paso. La actualización en el árbol de clasificación consiste en encontrar el mejor lugar para añadir la nueva instancia, este proceso puede necesitar una reestructuración de todo el árbol. El algoritmo es muy sensible a otros dos parámetros:

- **Acuity:** Este parámetro es muy importante, dado que la utilidad de categoría se basa en una estimación de la media y la desviación estándar de un valor de un atributo para un nodo en particular [32].
- **Cut-off:** Este valor es utilizado para controlar el crecimiento del número de segmentos, indica el grado de mejoría que se debe producir en la utilidad de categoría para que la instancia se tome en cuenta de una manera individual [32].

Sin embargo, COBWEB pertenece a los métodos de aprendizaje conceptual o basados en modelos, al algoritmo no se debe indicar el número exacto de

clusters que necesitamos, dado que gracias a los parámetros antes mencionados encuentra el número óptimo [32].

#### 2.2.3.2. EM (Expectation-Minimization)

Pertenece a la familia de modelos que se conocen como Finite Mixture Models [32], es un algoritmo basado en la mezcla que encuentra estimados de probabilidad máxima de los parámetros en modelos probabilísticos [35]. EM se trata de obtener la FDP (Función de densidad de probabilidad) a la que pertenece un conjunto de datos.

#### 2.2.3.3. K-MEANS

El algoritmo *K-means* fue creado por MacQueen en 1997, es un algoritmo de aprendizaje no supervisado de *clustering*. Sigue un procedimiento simple de clasificación de un conjunto de objetos en un determinado número K de clústeres específicos. El algoritmo tiene como objetivo encontrar grupos en los datos no etiquetados o sin categorías, los datos se agrupan según la similitud de las características que se proporcionan. El nombre "K-means" viene representada por cada uno de los clusters por la media (o media ponderada) de sus puntos es decir, por su centroide [36]. El algoritmo de K-means se realiza en 4 etapas:

- **Etapa 1.** Inicialización: En la primera etapa se define el conjunto de objetos a particionar, además el número de grupos y un centroide para cada grupo [37].
- **Etapa 2.** Clasificación: Para cada objeto de los datos, se calcula la distancia a cada centroide, se determina el centroide más cercano, y el objeto es incorporado al grupo relacionado con ese centroide [37].
- **Etapa 3.** Cálculo del centroide: Para cada grupo generado en el paso anterior se vuelve a calcular su centroide [37].
- **Etapa 4.** Condición de la convergencia: Existen varias formas de convergencia como puede ser: converger cuando alcanza un número de iteraciones dadas; converger cuando no existe un número el intercambio de objetos entre los grupos; o converger cuando la diferencia entre los centroides de dos iteraciones consecutivas es más pequeño que un umbral dado. Si no se satisface la condición de convergencia, se procede a repetir los pasos anteriores [37].

### 2.2.4. Procesamiento del Lenguaje Natural (NLP)

*Natural language processing* (por sus siglas en inglés NLP) es un área de la minería de textos que se define como el procesamiento automático o semiautomático del lenguaje humano [38]. NLP es multidisciplinario pues concierne otras áreas como *Natural Language Generation* (por sus siglas en inglés NLG) y *Natural Language Understanding* (por sus siglas en inglés NLU). NLG asegura que el texto generado se encuentre gramaticalmente correcto y fluido. Por su parte NLU consiste en al menos uno de estos pre-procesamientos: tokenización, análisis léxico, análisis sintáctico y análisis semántico [39].

La terminología utilizada en NLP normalmente corresponde a sub-áreas, entre ellas se encuentran las siguientes.

- **Morfología:** Estudia la estructura de las palabras. Las palabras están compuestas por morfemas, los cuales son la unidad más pequeña que tiene significado léxico o gramatical. Por ejemplo la palabra "asignatura" es un morfema, pues al fusionarla con sufijos o prefijos se convierte en la raíz de otras, por ejemplo al añadir una "s" al final se obtiene "asignaturas" [38].
- **Sintaxis:** Estudia la forma en la que las palabras son usadas para formar frases [38].
- **Semántica:** Estudia el significado de las palabras en base a su sintaxis [38].
- **Pragmática:** Estudia el significado de las palabras según su contexto [38].

#### 2.2.4.1. Técnicas de preprocesamiento de textos

En NLP se debe tener en cuenta conceptos fundamentales sobre el preprocesamiento de textos. Estos conceptos describen técnicas a seguir para preparar los textos de manera que se pueda aplicar adecuadamente diversos algoritmos [5].
Entre las técnicas de preprocesamiento se encuentran las siguientes.

- ***Tokenization*:** La tokenización es un proceso que permite dividir el texto en palabras individuales llamadas también por el término en inglés "*tokens*". Este proceso puede ser realizado de diversas formas dependiendo del idioma que se esté analizando. Una forma de realizarlo es usar delimitadores de texto como espacios en blanco y signos de puntuación [5].

- ***Lemmatization*:** La lematización es un proceso lingüístico y morfológico que permite determinar la forma básica raíz también conocida como lema, de cada palabra dentro de un texto que aparezca en forma flexiva o derivada. Este proceso tiene como fin reducir el el tamaño del vocabulario obtenido a partir del texto [40].

- ***Stemming*:** Es un término que engloba los algoritmos de radicación, los cuales permiten representar de un mismo modo las distintas variantes de un término con el objetivo de mejorar la capacidad de procesamiento de un texto. Este proceso tiene gran relación con la lematización, pues esta se lleva a cabo por medio de estos algoritmos [40].

- ***Normalization*:** La normalización es un proceso que consiste en homogeneizar un texto. La normalización permite controlar cantidades numéricas, fechas, abreviaturas, acrónimos e incluso diferenciación entre mayúsculas y minúsculas. Por ejemplo un texto puede tener palabras en mayúsculas y minúsculas, por lo que al aplicar la normalización se puede convertir todo el texto a una de estas dos formas [40].

- ***Stopping*:** Es un proceso que permite remover las palabras comunes en un texto, con la finalidad de ahorrar espacio en disco y tiempo de procesamiento. Estas palabras se las conoce con el término en inglés "stopwords". Este proceso no implica pérdida de información pues las palabras que se eliminan tienen pequeño impacto sobre el texto completo [5].

- **TF-IDF (*Term Frecuency, Inverse Document Frecuency*):** Es un método que permite estimar la importancia de una palabra en el texto expresándola mediante la ponderación de dicha palabra. Para estimar la ponderación este método utiliza la frecuencia de aparición de una palabra en un texto y en el conjunto de textos de la colección [40].

#### 2.2.4.2. Índices de similitud

Los índices de similitud son algoritmos que parte del área de *Information Retrieval* (IR), los cuales brindan diferentes enfoques para obtener la similitud de textos utilizando vectores de palabras en especial ponderados mediante TF-IDF [41].
A continuación, se describen los índices de similitud utilizados en este proyecto. Las fórmulas para cada uno de los índices utilizan la notación "Q" y "D" para representar vectores de palabras. Siendo "Q" el vector de palabras del primero texto a comparar y "D" el vector de palabras del segundo texto a comparar.

- **Índice de Jaccard:** También conocido como coeficiente de similitud de Jaccard es una medida para calcular la similitud entre textos. En esta medida, el índice comienza con un valor mínimo de 0 (textos completamente diferentes) hasta a un valor máximo de 1 (textos completamente similares). El índice se define matemáticamente como el tamaño de la intersección dividido por el tamaño de la unión de los vectores de palabras correspondientes a los textos [41]. La fórmula del índice de Jaccard se presenta a continuación.

$$Jaccard(Q, D) = \frac{\mid Q \cap D \mid}{\mid Q \mid + \mid D \mid - \mid Q \cap D \mid}$$

- **Similitud coseno:** La similitud coseno es una medida para calcular el valor de la similitud existente entre dos vectores. Se basa en el principio que, si dos vectores se encuentran próximos, el ángulo formado entre ellos será pequeño y al contrario si los vectores se encuentran distantes el ángulo formado entre ellos será grande. Si el valor del coseno es igual a uno significa que los textos son similares, por el contrario, si el valor es igual a cero significa que los textos son completamente distintos. Se puede utilizar el producto interior para evaluar el valor del coseno del ángulo comprendido entre ellos [41]. La fórmula de la similitud coseno se presenta a continuación.

$$Coseno(Q, D) = \frac{Q * D}{\| Q \| \| D \|} = \frac{\sum_{i=1}^{n} Q_i D_i}{\sqrt{\sum_{i=1}^{n} {Q_i}^2} \sqrt{\sum_{i=1}^{n} {Q_i}^2}}$$

- **Coeficiente de Sorensen-Dice:** El coeficiente ó índice de Sorensen-Dice es un estadístico utilizado para comparar la similitud entre dos textos. Este índice fue desarrollado y publicado independientemente por Thorvald Sorensen y Lee Raymond Dice en 1948 y 1945 respectivamente. Para evaluar la similitud entre los textos se utiliza un fórmula matemática semejante al índice de Jaccard [42]. La fórmula del coeficiente de Sorensen-Dice se presenta a continuación.

$$Sorensen - Dice(Q, D) = \frac{2 \mid Q \cap D \mid}{\mid Q \mid + \mid D \mid}$$

- **Overlap coefficient:** El coeficiente de superposición también llamado coeficiente Szymkiewicz-Simpson es una medida para obtener el índice de similitud entre dos textos y se encuentra relacionado con el índice de Jaccard. Con el propósito de encontrar el valor de similitud, éste coeficiente realiza una superposición entre dos vectores de palabras dividiendo el tamaño de la intersección entre ellas por el menor del tamaño de los dos vectores [43]. La fórmula del coeficiente de superposición se presenta a continuación.

$$Overlap(Q, D) = \frac{| \ Q \cap D \ |}{min(| \ Q \ |, | \ D \ |)}$$

### 2.2.5. Herramientas

En la actualidad, los algoritmos de minería de textos se pueden utilizar mediante diversas librerías y herramientas, lo cual facilita el proceso de desarrollo de sistemas o aplicaciones que procesan diferentes tipos de textos con un fin determinado.

Por esta razón, a continuación, se describen algunas herramientas utilizadas en el presente proyecto de titulación, con respecto a los algoritmos relacionados a las técnicas de agrupación y de procesamiento del lenguaje natural.

Python[9], es un lenguaje de programación de código abierto multiparadigma que soporta la programación orientación a objetos, imperativa y funcional. Es compatible con sistemas operativos como Linux, Windows, Mac OS. También permite el uso de bibliotecas externas o librerías que implementan funcionalidades específicas. Esta herramienta se utilizó para la implementación y desarrollo de algoritmos que involucren minería de textos con la finalidad de obtener la similitud de las instancias de los sílabos.

NLTK[10] (por sus siglas en inglés de *Natural Language Toolkit*), es una herramienta para desarrollar aplicaciones en Python con el fin trabajar con datos de lenguaje humano. Proporciona un conjunto de interfaces para el procesamiento de texto, clasificación, tokenización, stemming, etiquetado, análisis y razonamiento semántico. Esta herramienta se utilizó para implementar los algoritmos correspondientes a las técnicas de preprocesamiento de textos.

[9] La URL de documentación e instalación de la herramienta es `https://www.python.org/`.
[10] La URL de documentación e instalación de la herramienta es `http://www.nltk.org/`.

Scikit-learn[11], es una herramienta que proporciona librerías para el aprendizaje de máquina en Python. Implementa varios algoritmos de clasificación, regresión y agrupación, entre los cuales se encuentra las SVM (por sus siglas en de *Support Vector Machines*), *random forest*, k-means o DBSCAN. También se encuentra diseñada para interoperar con otras librerías numéricas y científicas como NumPy y SciPy. Esta herramienta se utilizó para implementar el algoritmo de la técnica TF-IDF.

Weka[12], es una herramienta que proporciona algoritmos de aprendizaje automático para tareas de minería de datos, dichos algoritmos están implementados en código Java. Entre ellos se encuentran librerías para preprocesamiento de datos, clasificación, regresión, *clustering* o reglas de asociación. Esta herramienta se utilizó para aplicar las técnicas de *clustering* sobre las instancias de los sílabos.

[11] La URL de documentación e instalación de la herramienta es `http://scikit-learn.org/stable/`.

[12] La URL de descarga y documentación sobre la herramienta es `https://www.cs.waikato.ac.nz/ml/weka/`.

# Capítulo 3

# Proceso de integración de datos, creación de la ontología de sílabos y minería de textos aplicada

En esta sección se describe las fuentes de datos sobre sílabos universitarios con su respectivo preprocesamiento. Posteriormente se describe el proceso de creación de una ontología para representar el vocabulario ontológico requerido en el dominio de los sílabos utilizados en la Universidad de Cuenca utilizando una metodología en concreto.
Finalmente, se detalla el proceso de minería de textos, explicando los algoritmos implementados para comparar la similitud del contenido académico utilizando instancias reales de sílabos de la Universidad de Cuenca.

## 3.1. Fuentes de datos de los sílabos

Los datos utilizados para instanciar la ontología de sílabos provienen de diversas fuentes de datos tales como: bases de datos, servicios Web, documentos PDF, DOCX y XLSX. Por lo cual, la integración de los datos desde todas las fuentes anteriormente mencionadas necesitan la ejecución de técnicas de preprocesamiento y limpieza de datos, con la finalidad de que las instancias sean correctas y consistentes. La Figura 3.1 muestra gráficamente los pasos

que se deben seguir para obtener datos íntegros y correctos.

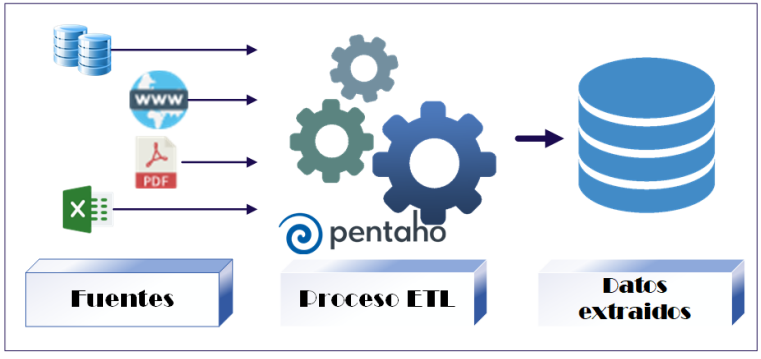



Figura 3.1: Proceso de integración, limpieza y extracción de datos.

## 3.2. Preprocesamiento y limpieza de los datos de sílabos

El preprocesamiento y limpieza de datos se realizó utilizando la herramienta *Pentaho Data Integration*, la cual proporcionó medios para la conexión con bases de datos y servicios web necesarios.
El preprocesamiento consiste en integrar todas las fuentes necesarias para obtener los datos completos de los sílabos. Mientras que la limpieza consiste en la corrección y eliminación de datos erróneos obtenidos de las fuentes.
La Figura 3.2 presenta el proceso de conexión y ETL realizado para obtener la información de las asignaturas a partir de sus sílabos correspondientes desde una base de datos y un servicio *Web REST*.

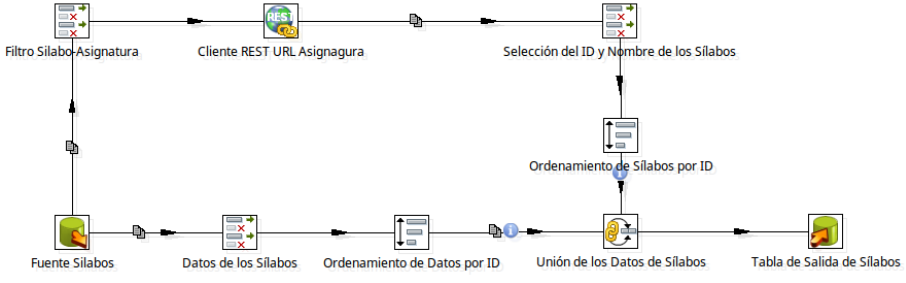



Figura 3.2: ETL de las asignaturas y sílabos

## 3.3. Ontología para sílabos universitarios

En esta sección se presenta el proceso de creación de una red ontológica para describir datos de sílabos de la Universidad de Cuenca.

### 3.3.1. Vocabulario ontológico para representar sílabos universitarios

La ontología que se presenta a continuación fue creada utilizando la metodología *NeOn*, puesto que al realizar la investigación sobre las metodologías para la creación de ontologías se obtuvo como resultado que *NeOn* es la más adecuada para el dominio de sílabos universitarios. Se llegó a este resultado, puesto que *NeOn* es una metodología flexible y permite utilizar solo los escenarios convenientes a criterio de quien vaya a crear la ontología. Por lo cual, de los nueve escenarios que propone *NeOn* se utilizaron los cuatro que se consideró convenientes, estos escenarios se encuentran detallados en las siguientes sub-sub-secciones.
La implementación de la ontología se realizó mediante el editor ontológico *Protégé* utilizando RDF/XML como formato de serialización del RDF resultante.

#### 3.3.1.1. Escenario 1: Especificación de requisitos

El primer escenario consta de la especificación de la aplicación, para ello se utilizará la plantilla ORSD[1] propuesta por la metodología *NeOn* con el fin de describir todas las tareas que permitan generar una especificación acorde a las necesidades de la ontología. Este escenario se compone de las tareas enumeradas y descritas en el documento ORSD de la Tabla 3.1 a continuación.

| **1** | **Propósito** |
|---|---|
| El propósito de la creación de una ontología para la representación de sílabos es, proveer un modelo de datos que permita englobar todo el contenido de un sílabo de la Universidad de Cuenca, con el fin de aplicar técnicas de minería de textos para descubrir patrones sobre las instancias de la misma utilizando información de sílabos reales, consiguiendo automatizar la comparación de sílabos y mejorar el proceso de movilidad estudiantil. | |
| **2** | **Alcance** |
| La ontología tiene un alcance determinado al dominio académico, específicamente el de sílabos de la Universidad de Cuenca. | |
| **3** | **Lenguaje de implementación** |

[1] Siglas en inglés del término *Ontology Requirements Specification Document*, en español significa Documento de Especificación de Requerimientos de la Ontología.

| | |
|---|---|
| La ontología es implementada mediante el lenguaje OWL. | |
| **4** | **Usuarios finales previstos** |
| Los usuarios que utilizarán la ontología son:<br>Usuario 1: El Profesor Fiscal de una Facultad, que realizará una comparación de instancias reales de sílabos.<br>Usuario 2: Los docentes de la Universidad de Cuenca, quienes crearán la instancias de sílabos sobre cada asignatura que impartan. | |
| **5** | **Usos previstos** |
| Los usos de la ontología son:<br>Uso 1: Representar el contenido académico de los sílabos pertenecientes a las asignaturas impartidas en la Universidad de Cuenca.<br>Uso 2: Permitir la aplicación de algoritmos de minería de textos para descubrir patrones sobre los sílabos. | |
| **a. Requisitos no funcionales** | |
| Los requisitos no funcionales de la ontología son:<br>RNFN 1: El vocabulario ontológico debe estar escrito en inglés y español. | |
| **b. Requisitos funcionales: Grupos de preguntas de competencia** | |

Las preguntas de competencia con sus respectivas respuestas son:
P1. ¿Cuál es el nombre de la asignatura?
R1. Programación I, Administración II, Bioestadística I, etc.

P2. ¿Cuál es la carrera en la que se imparte la asignatura?
R2. Medicina y Cirugía, Ingeniería de Sistemas, Marketing, etc.

P3. ¿En qué facultad se imparte la asignatura?
R3. Ingeniería, Ciencias Médicas, Ciencias Económicas y Administrativas, etc.

P4. ¿Cuál es el nombre del coordinador de la asignatura?
R4. Marías Soledad Escandón, Jorge Luis García, etc.

P5. ¿Cuál es el nombre de cada uno de los docentes que imparten la asignatura?
R5. María Soledad, José Ricardo Ramirez, etc.

P6. ¿Cuántos y cuáles son los objetivos de la asignatura?
R6. Desarrollar las aptitudes y actitudes en los estudiantes para el uso correcto de la bioestadística inferencial, etc.

P7. ¿Cuáles son los logros de la asignatura, con sus respectivos indicadores y situaciones de evaluación?
R7. Recordar, identificar y utilizar el,método estadístico para el análisis de estudios descriptivos, etc.

P8. ¿Cuáles son los capítulos, temas y estrategias de aprendizaje del contenido académico de la asignatura?
R8. Nivelación respecto a Bioestadística I, Métodos Analíticos, etc.

P9. ¿Qué referencias bibliográficas tienen los capítulo y temas del contenido académico de la asignatura?
R9. Dawson-Saunders, B.; Trapp, R., (1997), Bioestadística Médica, etc.

10. ¿Cuáles son lo prerrequisitos y correquisitos de la asignatura?
R10. Informática básica, Administración I, etc.

**7 Pre-glosario de términos**

**a. Términos de las preguntas de competencia**

Los términos de las preguntas de competencia son

| | Termino | Frecuencia |
|---|---|---|
| | asignatura | 10 |
| | capítulos | 1 |
| | carrera | 1 |
| | contenido académico | 2 |
| | coordinador | 1 |
| | correquisitos | 1 |
| | docente | 1 |
| | estrategias de aprendizaje | 1 |
| | facultad | 1 |
| | indicadores | 1 |
| | logros | 1 |
| | nombre | 3 |
| | objetivos | 1 |
| | prerrequisitos | 1 |
| | referencias bibliográficas | 1 |
| | situaciones de evaluación | 1 |
| | temas | 2 |
| **c. Objetivos** | | |
| Los objetivos que responden a las preguntas de competencia son: Administración I, Administración II, Programación I, Ingeniería, Ciencias Médicas, Marías Soledad Escandón, Jorge Luis Garcia, etc. | | |

Tabla 3.1: Descripción de los requerimientos de la ontología.

#### 3.3.1.2. Escenario 2: Reutilización y reingeniería de recursos no ontológicos (NORs)

El segundo escenario es la reutilización y reingeniería de recursos no ontológicos, para este escenario se utiliza un glosario de términos que se encuentran en las fuentes de datos descritas previamente, sobre los sílabos de las asignaturas que se imparten en la Universidad de Cuenca. Este escenario se compone de dos partes, cada una con sus actividades se detallan a continuación.

La primera parte detalla los recursos no ontológicos que se usarán para crear la ontología de sílabos de la Universidad de Cuenca. En la Tabla 3.2 se describen las actividades correspondientes a esta parte.

| 1 | **Buscar recursos no ontológicos** | | |
|---|---|---|---|
| Los recursos no ontológicos que tienen relevancia para la ontología son documentos en formato PDF que detallen el contenido de un sílabo sobre una asignatura específica y las bases de datos de sílabos que se detalló previamente como fuente de información. Ambos recursos no ontológicos contienen términos que pueden o no ser relevantes para la ontología. | | | |
| **2** | **Validar el conjunto de recursos no ontológicos candidatos** | | |
| Los términos de los recursos no ontológicos son enumerados y descritos, para validar su relevancia dentro del dominio de la ontología. En caso de que un término sea relevante, se utilizará el mismo nombre dentro del vocabulario ontológico para la ontología de sílabos. | | | |
| | **Término** | **Descripción** | **Relevante** |
| | Asignatura | Refiere a una asignatura que se imparta dentro de una carrera perteneciente a una facultad. | Si |
| | Bibliografía | Es una referencia hacia una obra literaria de la cual se haya obtenido parte del contenido de la asignatura. | Si |
| | Capítulo | Es el nombre de una determinada sección del contenido. | Si |
| | Carrera | Nombre de la carrera en la que se dicta la asignatura. | Si |
| | Contenido Académico | Describe un conjunto de capítulos y temas que debe el profesor debe impartir para cubrir la asignatura en su totalidad. | Si |
| | Coordinador | Persona encargada de la asignatura. | Si |
| | Correquisito | Indica la asignatura que se debe de tomar en en conjunto con la asignatura descrita por el sílabo. | Si |
| | Denominación | Es el nombre de la asignatura. | Si |
| | Eje de Formación | Indica el tipo de formación profesional de la asignatura. | No |
| | Estrategia de Aprendizaje | Es la descripción de una estrategia de aprendizaje que el profesor utiliza para enseñar un determinado capítulo. | Si |
| | Facultad | Nombre de la facultad a la que pertenece una carrera. | Si |

| | | | |
|---|---|---|---|
| | Horas Teóricas | Indica el número de horas teóricas que cubre la asignatura semanalmente. | No |
| | Horas Prácticas | Indica el número de horas prácticas que cubre la asignatura semanalmente. | No |
| | Horas Teórico Prácticas | Indica el número de horas teórico-prácticas que cubre la asignatura semanalmente. | No |
| | Indicador | Es la descripción de una cualidad visible y medible que los estudiantes deben manifestar al alcanzar un logro de aprendizaje. | Si |
| | Logro de Aprendizaje | Es la descripción de un logro de aprendizaje que los estudiantes deben alcanzar al terminar la asignatura. | Si |
| | Modalidad | Indica la modalidad de enseñanza de la asignatura. | Si |
| | Objetivo | Indica una meta que los estudiantes deben alcanzar al final de la asignatura. | Si |
| | Período | Es el período académico en el que se imparte la asignatura. | Si |
| | Prerrequisito | Indica la asignatura previa que debe ser tomada para poder cursar la asignatura descrita por el sílabo. | Si |
| | Profesor | Es la persona que imparte la asignatura. | Si |
| | Recurso | Describe a un medio de aprendizaje material que el profesor utiliza para impartir la asignatura. | Si |
| | Situación de Evaluación | Es la descripción de una actividad, tarea o instrumento que permite evaluar un logro de aprendizaje. | Si |
| | Sub-capítulo | Es una parte de un capítulo que se debe enseñar a los estudiantes. | Si |
| | Universidad | Nombre de una institución universitaria. | Si |
| **3** | **Seleccionar los recursos no ontológicos más apropiados** | | |

| |
|---|
| Ambos recursos no ontológicos son apropiados, puesto que los documentos PDF indican la terminología y la base de datos permitirá instanciar la ontología una vez creada. |

Tabla 3.2: NOR sobre el glosario de términos utilizados en sílabos de la Universidad de Cuenca.

La segunda parte corresponde a la reingeniería de los recursos no ontológicos. En la Tabla 3.3 se describen las actividades correspondientes a esta parte.

| | |
|---|---|
| **1** | **Ingeniería inversa sobre los recursos no ontológicos** |
| Al aplicar la ingeniería inversa se obtuvo dos listados, el primero indica los términos de los recursos no ontológicos que se convertirán en clases dentro de la ontología de sílabos y el segundo indica los términos que se convertirán en propiedades de datos. A continuación se muestran ambos listados. | |
| | Los términos que serán clases en la ontología son los siguientes.<br>* Asignatura, Bibliografía, Capítulo, Carrera, Contenido Académico, Coordinador, Correquisito, Estrategia de Aprendizaje, Facultad, Indicador, Logro de Aprendizaje, Modalidad, Objetivo, Período, Prerrequisito, Profesor, Recurso, Situación de Evaluación, Sub-capítulo, Universidad. |
| | Los términos que serán propiedades de datos en la ontología son los siguientes.<br>* Créditos, Criterios, Código, Denominación, Descripción. |
| **2** | **Transformación de los recursos no ontológicos** |
| La transformación de recursos no ontológicos se realiza utilizando el patrón ABox. Esta transformación utiliza el esquema de datos de los recursos no ontológicos para crear la jerarquía de conceptos de la ontología y los datos de los recursos no ontológicos para crear las instancias de la ontología. | |
| **3** | **Ingeniería hacia la ontología** |
| La Figura 3.1 muestra un ejemplo del análisis realizado al conjunto de datos de los recursos no ontológicos. | |

Tabla 3.3: Ingeniería inversa y transformación de recursos no ontológicos.

#### 3.3.1.3. Escenario 6: Reutilización, la fusión y reingeniería de los recursos ontológicos

El sexto escenario comprende la reutilización, fusión y reingeniería de recursos ontológicos (ontologías) generales necesarios para la creación de una ontología de sílabos de la Universidad de Cuenca. Este escenario se compone de las tareas y figuras descritas a en el Anexo A.1.

#### 3.3.1.4. Escenario 9: Localización de recursos ontológicos

El noveno escenario comprende la localización de recursos ontológicos, en la cual se presenta los lenguajes que se utilizaron para la creación de una ontología de sílabos de la Universidad de Cuenca.
Este escenario se compone de las tareas enumeradas y descritas a continuación en la Tabla 3.4.

| | | |
|---|---|---|
| **1** | **Seleccionar los activos lingüísticos más apropiados** | |
| Los activos lingüísticos más apropiados son el Inglés y el Español. | | |
| **2** | **Seleccionar las etiquetas de la ontología a ser localizadas** | |
| La ontología tiene etiquetas en todas las clases y propiedades. | | |
| **3** | **Obtener la traducción de las etiquetas de la ontología** | |
| Algunas etiquetas de clases y propiedades se presentan a continuación en ambos activos lingüísticos. | | |
| | **Inglés** | **Español** |
| | Faculty | Facultad |
| | Professor | Profesor |
| | Subject | Asignatura |
| | owner | autor |
| | responsibility of | responsabilidad de |
| | edition | edición |
| | first name | nombres |
| | page start | página de inicio |
| **4** | **Evaluar la traducción de las etiquetas** | |
| Las etiquetas en inglés tienen concordancia con su traducción al español y viceversa. | | |
| **5** | **Actualizar la ontología** | |
| La ontología se encuentra actualizada con las etiquetas en ambos activos lingüísticos. | | |

Tabla 3.4: Localización de recursos no ontológicos y elección de lenguajes para la ontología de sílabos de la Universidad de Cuenca.

Una vez creada la ontología de sílabos, se puede decir que los cuatro escenarios utilizados de la metodología *NeOn* fueron de gran utilidad para integrar los recursos ontológicos con los no ontológicos. Además, ayudaron en la investigación y la reutilización de ontologías relacionadas al contexto educativo universitario. Finalmente, la ontología cumple el propósito principal para el que fue creada, permitiendo aplicar técnicas de minería de textos sobre sus instancias.

### 3.3.2. Visualización de la ontología

Con el propósito de brindar una forma de visualización agradable y que proporcione información sobre la ontología, se creó una página web utilizando la herramienta Widoco. La información sobre la ontología se puede visualizar en las Figuras 3.3 y 3.4.


Figura 3.3: Cabecera y contenido de la página web creada para la visualización de la ontología.

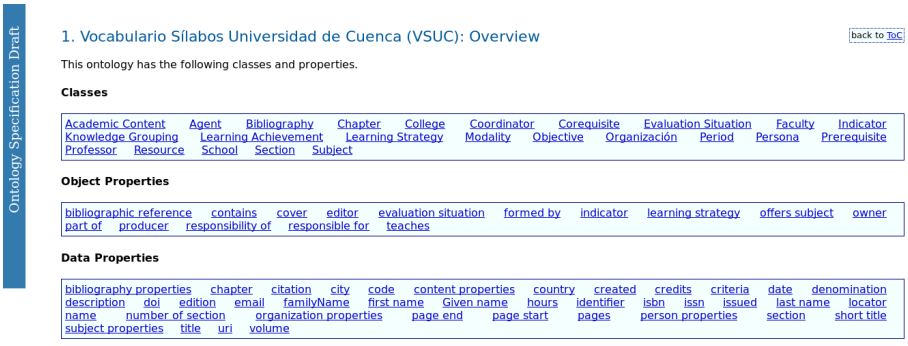



Figura 3.4: Enumeración de las clases, propiedades de datos y de objetos desde la página web.

El modelo ontológico correspondiente a las clases de la ontología se pueden observar a manera de grafo en la Figura 3.5. Este modelo se encuentra en la página web creada al utilizar Widoco.

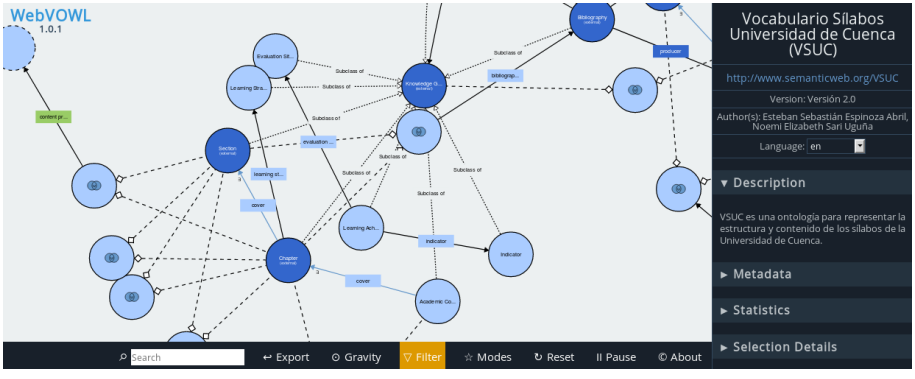



Figura 3.5: Grafo parcial de las clases y, propiedades de la ontología de sílabos.

### 3.3.3. Proceso de RDF-Ization aplicado sobre la ontología

El proceso de RDF-Ization se realizó mediante el uso de las herramientas *Pentaho Data Integration* y LOD-GF una vez que los datos pasaron por un proceso de integración y limpieza. En la Figura 15 se puede observar parte del proceso realizado para generar las instancias de los sílabos a partir de la ontología creada y de los datos previamente procesados.

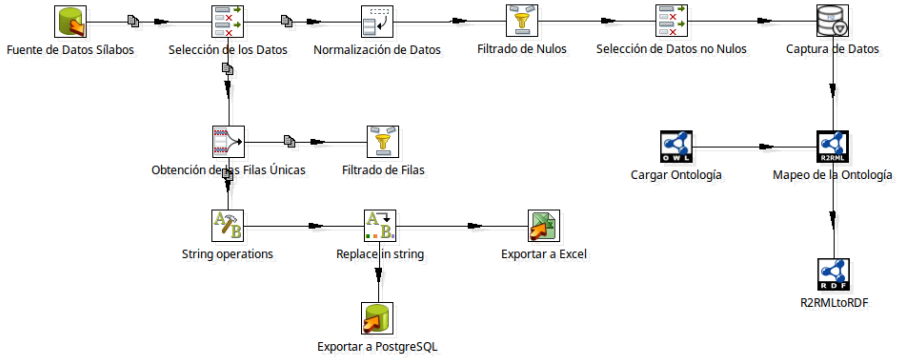



Figura 3.6: Proceso de RDF-Ization para obtener instancias de sílabos.

### 3.3.4. Publicación de la ontología

Con la finalidad de concluir con el proceso de creación de la ontología, se describen a continuación algunas direcciones web para encontrar información sobre la ontología y sus instancias. Estas direcciones web se encuentran por el momento accesibles únicamente desde una computadora conectada a la red de la Universidad de Cuenca.

- La ontología se encuentra disponible desde la dirección web: *http://201.159.223.25:8081/syllabusEc/ontology.xml*.
- La documentación de la ontología se encuentra accesible desde la dirección web: *http://201.159.223.25:8081/syllabusEc/index-en.html*.
- El repositorio que contiene las instancias de los sílabos, se encuentra accesible desde la siguiente dirección web: *http://201.159.223.25:10035/*.

## 3.4. Minería de textos aplicada a la ontología

En esta sección se presenta la aplicación de algoritmos de minería de textos sobre la red de ontologías de sílabos utilizados en la Universidad de Cuenca.

### 3.4.1. Selección de las áreas de la minería de textos involucradas

Una vez creada e instanciada la ontología de sílabos, se debe aplicar algoritmos de minería de textos sobre ella para poder comparar los sílabos y determinar la similitud de los mismos. Para ello se debe seleccionar las áreas involucradas con el fin de determinar el tipo de algoritmos que se puede utilizar.

Lo primero es responder a mínimo 2 de las 5 preguntas que permiten conocer el o las áreas involucradas en la resolución de la problemática del presente trabajo. Las preguntas y sus respuestas se detallan a continuación.

- **¿El objetivo es agrupar palabras o documentos?:** El objetivo es agrupar documentos como un todo y poder compararlos para determinar su similitud. Por lo cual el nivel de granularidad es alto.
- **¿Se requiere encontrar palabras y documentos específicos o caracterizar todo el conjunto?:** Esta pregunta no tiene una respuesta exacta, puesto que para comparar la similitud de los sílabos se debe tanto recuperar como extraer información de documentos de manera individual y colectiva. Por lo tanto, no se puede dar un enfoque específico.
- **¿Se cuenta con documentos categorizados?:** Los documentos en este caso son las instancias de sílabos por lo que no son categorizados. Esto implica que se debe usar algoritmos no supervisados.
- **¿Para resolver el problema se debe considerar el significado del texto o su estructura?:** La estructura de los documentos está bien definida por la ontología de sílabos, por lo cual se debe considerar el significado de las palabras u oraciones dentro del contenido de las instancias.
- **¿Los documentos son independientes o están conectados mediante hiperenlaces?:** Los documentos en la web semántica utilizan URI's por lo que cuentan con hiperenlaces. Sin embargo, los documentos que utilizaremos para aplicar algoritmos no supervisados son independientes.

Una vez respondidas la preguntas y utilizando el árbol de decisión de la Figura 2.7 se puede determinar las áreas de la minería de textos que están involucradas en el problema. Estas áreas se muestran a continuación en la Tabla 3.5.

| | |
|---|---|
| *Natural Language Processing* | El área de procesamiento de lenguaje natural contempla algoritmos para obtener la similitud de textos. Por lo cual se la utilizará para comparar las instancias de los sílabos y determinar su similitud. |
| *Web mining* | Con respecto al área de *web mining* esta permite filtrar algoritmos de minería de textos relevantes para la ontología de sílabos en el contexto de la web. Por lo que sería un área contemplada para trabajos posteriores. |

Tabla 3.5: Áreas de la minería de textos involucradas en la problemática del proyecto.

De las dos áreas presentadas anteriormente se determinó que sólo se aplicará el *Natural Language Processing*, con el fin de comparar los contenidos de los sílabos y obtener un porcentaje de similitud. Mientras que *Web mining* es un área que se puede aplicar para trabajos futuros.

### 3.4.2. Uso y aplicación de algoritmos para NLP

Los algoritmos utilizados para el área de procesamiento de lenguaje natural fueron implementados o desarrollados en lenguaje Python.

Con respecto a las técnicas de preprocesamiento de textos se implementaron los siguientes: normalización, tokenización, stopping, stemming, TF-IDF. La Figura del Anexo A.2 muestra parte del código utilizado para implementar las técnicas de preprocesamiento. En cuanto a los índices de similitud se desarrollaron los siguientes algoritmos: similitud coseno, índice de Jaccard, coeficiente de Sorensen-Dice y coeficiente de superposición (Overlap). La Figura del Anexo A.3 muestra parte del código utilizado para implementar los índices de similitud.

El funcionamiento de la comparación entre los sílabos de las asignaturas se divide en varios pasos. Primero se compara las descripciones de cada sílabo y el resultado representa un porcentaje del 20 % de la similitud total. Posteriormente, cada capítulo del primer sílabo se compara con todos los capítulos del segundo sílabo y de ellos se escoge el que mayor porcentaje de similitud

obtenga como resultado.

Finalmente, los temas de cada capítulo del primer sílabo son comparados con los temas del capítulo escogido del segundo sílabo; la comparación de los capítulos y temas obtiene el 80 % restante de la similitud total. En la Figura del AnexA.4 muestra parte del código en Python sobre el funcionamiento de la comparación entre los sílabo y la Figura 3.8 describe de forma gráfica el proceso.

La razón por la que los capítulos y temas tienen un mayor porcentaje en la similitud total, es porque detallan el contenido académico que será visto por los estudiantes de dichas asignaturas; esto significa que brindan información fundamental en la comparación de sílabos. El valor de los porcentajes se determinó en base a la experiencia, de forma empírica al probar varias combinaciones.

Para una mejor compresión del funcionamiento de los algoritmos. La Figura 3.7 detalla un ejemplo de la comparación entre los sílabo; Fundamentos de Inteligencia Artificial y Redes Nueronales.

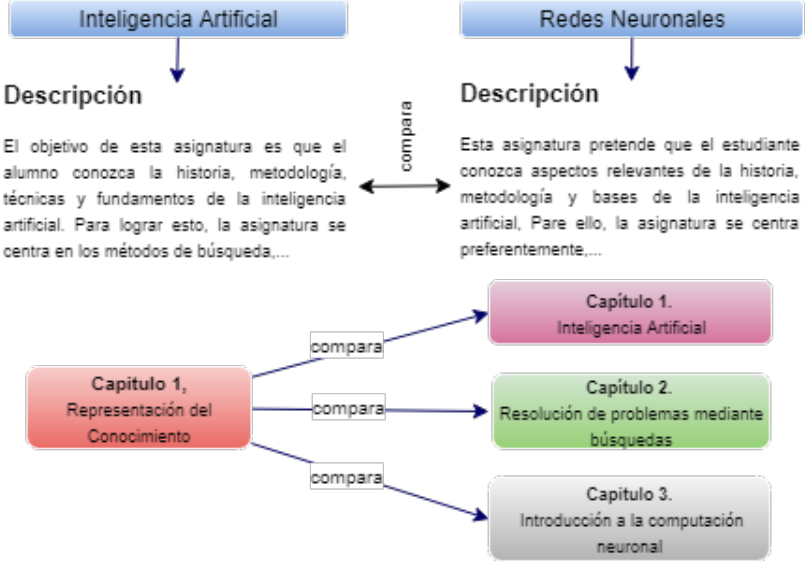



Figura 3.7: Comparación entre los sílabos, Inteligencia Artificial y Redes Neuronales.

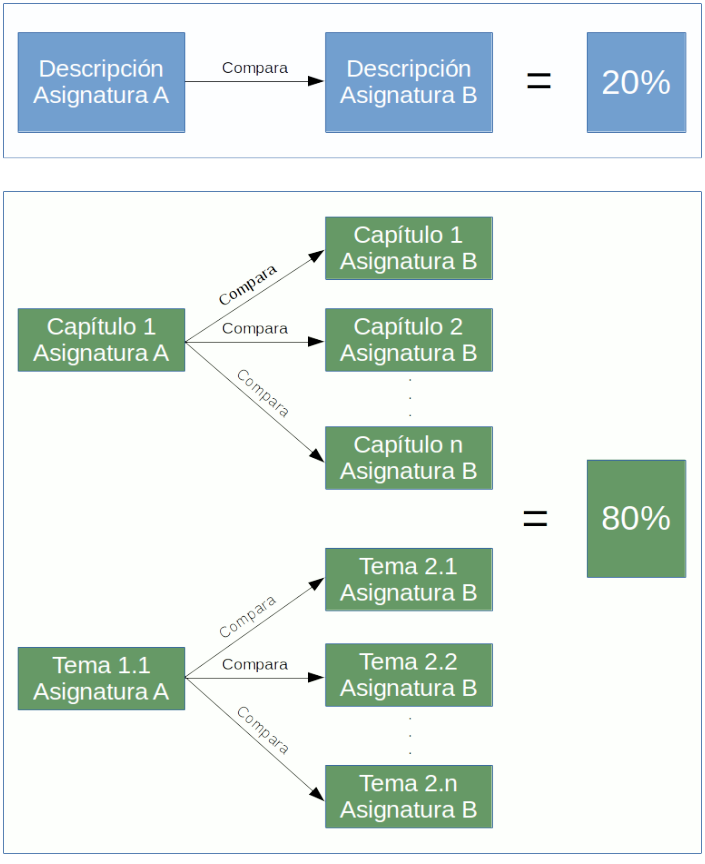



Figura 3.8: Descripción gráfica de la comparación entre los sílabos.

# Capítulo 4

# Evaluación y resultados

En esta sección se describe las técnicas de evaluación utilizadas para validar tanto la ontología de sílabos de la Universidad de Cuenca como la ejecución de los algoritmos de minería de textos.

## 4.1. Evaluación de la ontología

La evaluación de la ontología de sílabos se ha realizado utilizando dos herramientas, la primera es una prueba con el razonador de Protégé, el cual es una herramienta que permite evaluar las relaciones existentes entre las clases y propiedades de una ontología. La Figura 4.1 muestra que el razonador no ha encontrado errores en la ontología.

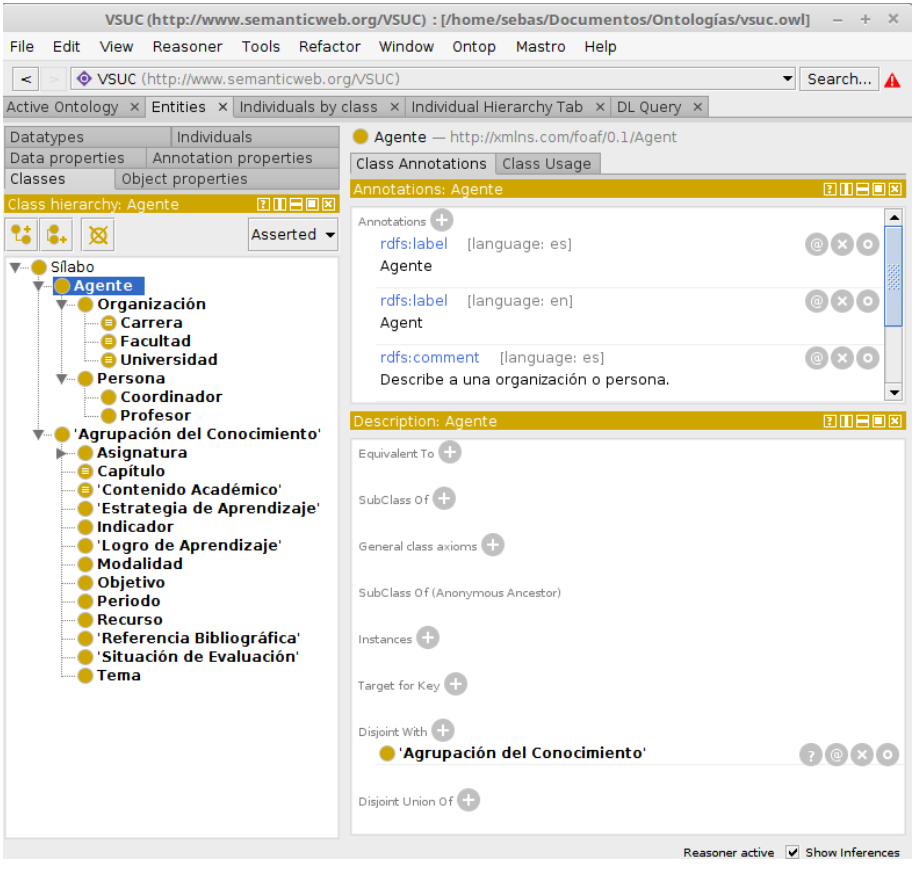



Figura 4.1: Evaluación de la ontología utilizando el razonador de la herramienta Protégé.

En la segunda prueba se utilizó la herramienta OOPS! como se puede ver en la Figura 4.2 se encontraron algunos errores leves, importantes y graves, de los cuales se corrigió la mayor parte dejando solo algunos leves e importantes.

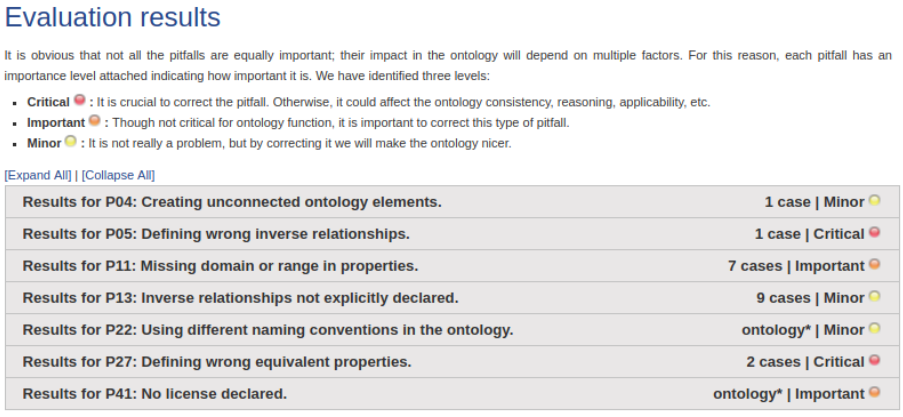



Figura 4.2: Primera evaluación de la ontología y visualización de errores utilizando la herramienta OOPS!.

La Figura 4.3 muestra la segunda prueba realiza en la herramienta OPPS!, en ella se puede observar que se han reducido los errores a solo leves e importantes.

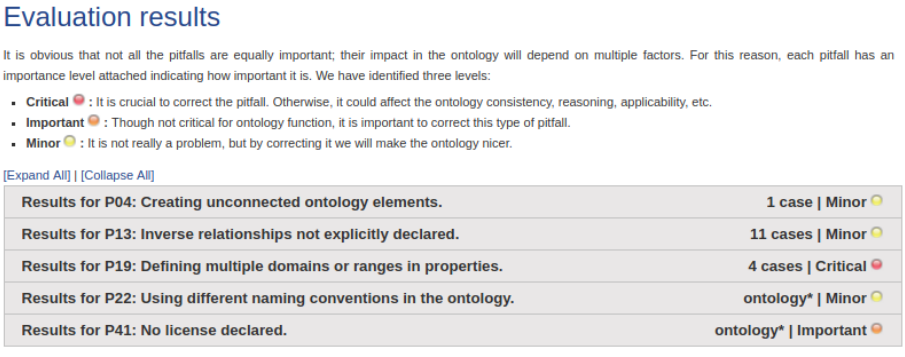



Figura 4.3: Segundo evaluación de la ontología y visualización de errores utilizando la herramienta OOPS!.

Una vez realizada la evaluación de la ontología se determinó que esta lista

para su instanciación y uso en la minería de textos.

## 4.2. Evaluación de los algoritmos de minería de textos

La evaluación de los algoritmos de minería de textos se realizó utilizando datos de las instancias de los sílabos de cinco asignaturas, las cuales son: Redes neuronales, Fundamentos de la inteligencia artificial, Diseño geométrico de carreteras, Cálculo integral y Teoría electromagnética II.

Las dos primeras pertenecen a las carreras de Ingeniería electrónica e Ingeniería de sistemas respectivamente, además sus sílabos son similares por lo que al analizar su similitud ésta debe dar un resultado alto. Sin embargo, las asignaturas tercera, cuarta y quinta pertenecen a diferentes carreras, por lo tanto, el resultado del análisis de similitud para ellas debe ser bajo o muy bajo. La razón por la que se escogió estas asignaturas es para demostrar que las dos primeras obtendrán mejor porcentaje de similitud que el resto.

### 4.2.1. Comparación de los algoritmos sobre índices de similitud y determinación del más apropiado

El primer paso de la evaluación es establecer el mejor algoritmo de NLP sobre índices de similitud de entre todas las opciones explicadas en el marco teórico. Para ello se realizó un análisis de similitud entre las asignaturas explicadas anteriormente y se procedió a elegir el mejor de todos los algoritmos. Este algoritmo será utilizado para el resto de pruebas de evaluación.

La Tabla 4.1 presenta información sobre las asignaturas que se compararon mediante los algoritmos de índice de similitud.

| Código Asignatura | Nombre |
|---|---|
| 417 | REDES NEURONALES (ESCUELA DE ELECTRONICA Y TELECOMUNICACIONES)_ 4 |
| 496 | INTELIGENCIA ARTIFICIAL (ESCUELA DE INFORMATICA)_ 4 |
| 410 | DISEÑO GEOMETRICO DE CARRETERAS (MALLA MODERNA)_ 4 |
| 1264 | CALCULO INTEGRAL (MALLA MODERNA)_ 4 |
| 461 | TEORIA ELECTROMAGNETICA II (ELECTRICA MALLA 2013)_ 6 |

Tabla 4.1: Código y nombre de las asignaturas.

La Tabla 4.2 describe el porcentaje de similitud entre las asignaturas, utilizando los distintos algoritmos sobre índices de similitud. Por su parte, la Figura 4.4 muestra el resultado de comparación entre las descripciones de las asignaturas con códigos 417 y 496 respectivamente.

| Código Asignaturas | Índice de Jaccard | Similitud Coseno | Coeficiente de Sorensen- Dice | Coeficiente de Superposición |
|---|---|---|---|---|
| 417 - 496 | 15.69 % | 59.91 % | 27.12 % | 44.44 % |
| 417 - 410 | 4.82 % | 51.38 % | 9.2 % | 8.7 % |
| 417 - 1264 | 6.21 % | 60.48 % | 11.69 % | 7.96 % |
| 1264 - 417 | 6.21 % | 60.48 % | 11.69 % | 21.95 % |
| 417 - 461 | 8.33 % | 58.42 % | 15.38 % | 12.7 % |
| 496 - 410 | 3.23 % | 41.98 % | 6.25 % | 4.35 % |
| 496 - 1264 | 3.15 % | 51.98 % | 6.11 % | 3.54 % |
| 496 - 461 | 3.85 % | 45.25 % | 7.41 % | 4.76 % |
| 410 - 1264 | 5.3 % | 59.69 % | 10.06 % | 7.08 % |
| 1264 - 410 | 5.3 % | 59.69 % | 10.06 % | 17.39 % |
| 410 - 461 | 7.92 % | 52.48 % | 14.68 % | 12.7 % |
| 1264 - 461 | 6.67 % | 61.15 % | 12.5 % | 17.46 % |
| 461 - 1264 | 6.67 % | 61.15 % | 12.5 % | 9.73 % |

Tabla 4.2: Resultados de la comparación entre los distintos algoritmos de índice de similitud en base a la descripción de las asignaturas.

```
La descripción de la Asignatura 'REDES NEURONALES (ESCUELA DE ELECTRONICA Y TELECOMUNICACIONES)_ 4' es: Est
a asignatura pretende que el estudiante conozca aspectos relevantes de la historia, metodología y bases de
la inteligencia artificial. Para ello, la asignatura se centra preferentemente en el estudio de técnicas am
pliamente utilizadas, como son las redes neuronales y la lógica difusa. Desde un enfoque de la clasificació
n y reconocimiento de patrones, se hace uso de diversos modelos neuronales de espacios multidimensionales e
n aplicaciones prácticas que permiten comprender el funcionamiento y determinación de los diferentes domini
os en los que pueden implementarse.

La descripción de la Asignatura 'FUNDAMENTOS DE INTELIGENCIA ARTIFICIAL (SISTEMAS MALLA 2013)_ 4' es: El ob
jetivo de esta asignatura es que el alumno conozca la historia, metodología, técnicas y fundamentos de la i
nteligencia artificial. Para lograr esto, la asignatura se centra en los métodos de búsqueda, representació
n del conocimiento y el aprendizaje máquina.

El porcentaje de similitud de las descripciones según los 4 algoritmos se presenta a continuación:
Índice de Jaccard: 15.69%
Similitud Coseno: 59.91%
Coeficiente de Sorensen-Dice: 27.12%
Coeficiente de Superposición: 44.44%
```

Figura 4.4: Resultados de la ejecución de los algoritmos de similitud entre el sílabo de Inteligencia Artificial y el de Redes Neuronales.

El análisis de los resultados de la Tabla 4.1, sobre los distintos algoritmos de índices de similitud indica que el mejor es el: coeficiente de Sorensen-Dice (Sørensen–Dice coefficient). Esta decisión se fundamenta en que el índice de Jaccard proporciona porcentajes de similitud bajos para sílabos similares. Por el contrario, la similitud coseno proporciona porcentajes de similitud altos para sílabos diferentes. Mientras que el coeficiente de superposición, proporciona porcentajes de similitud distintos si se altera el orden de comparación de los sílabos. Debido a estas razones, la comparación completa entre las asignaturas se realizará utilizando el algoritmo de Sorensen-Dice.

### 4.2.2. Comparación completa de las asignaturas y resultados de similitud

Utilizando el algoritmo coeficiente de Sorensen-Dice se procedió a realizar una comparación completa entre las cinco asignaturas con el fin de encontrar su porcentaje de similitud. La Tabla 4.3 describe el porcentaje de similitud entre las asignaturas, comparando su descripción, capítulos y temas. Por su parte la Figura 4.5 muestra el resultado de comparación entre dos asignaturas.

| Código Asignaturas | Similitud Descripción | Similitud Capítulos | Similitud Temas | Similitud Total |
|---|---|---|---|---|
| 417 - 496 | 27.12 % | 34.17 % | 9.95 % | 23.07 % |
| 417 - 410 | 9.2 % | 22.31 % | 0.0 % | 10.77 % |
| 417 - 1264 | 11.69 % | 0.0 % | 0.0 % | 2.34 % |
| 417 - 461 | 15.38 % | 0.0 % | 0.0 % | 3.08 % |
| 496 - 410 | 6.25 % | 0.0 % | 0.0 % | 1.25 % |
| 496 - 1264 | 6.11 % | 0.0 % | 0.0 % | 1.22 % |
| 496 - 461 | 7.41 % | 0.0 % | 0.0 % | 1.48 % |
| 410 - 1264 | 10.06 % | 0.0 % | 0.0 % | 2.01 % |
| 410 - 461 | 14.68 % | 0.0 % | 0.0 % | 2.94 % |
| 1264 - 461 | 12.5 % | 0.0 % | 0.0 % | 2.5 % |

Tabla 4.3: Comparación de la similitud entre las asignaturas utilizando el algoritmo de Sorensen-Dice.

```
La descripción de la Asignatura 'REDES NEURONALES (ESCUELA DE ELECTRONICA Y TELECOMUNICACIONES)_ 4' es:
Esta asignatura pretende que el estudiante conozca aspectos relevantes de la historia, metodología y bas
es de la inteligencia artificial. Para ello, la asignatura se centra preferentemente en el estudio de té
cnicas ampliamente utilizadas, como son las redes neuronales y la lógica difusa. Desde un enfoque de la
clasificación y reconocimiento de patrones, se hace uso de diversos modelos neuronales de espacios multi
dimensionales en aplicaciones prácticas que permiten comprender el funcionamiento y determinación de los
 diferentes dominios en los que pueden implementarse.

La descripción de la Asignatura 'INTELIGENCIA ARTIFICIAL (ESCUELA DE INFORMATICA)_ 4' es: El objetivo de
 esta asignatura es que el alumno conozca la historia, metodología, técnicas y fundamentos de la intelig
encia artificial. Para lograr esto, la asignatura se centra en los métodos de búsqueda, representación d
el conocimiento y el aprendizaje máquina.

La similitud de las asignaturas se presenta a continuación:
El porcentaje similitud de las descripciones es: 27.12%
El porcentaje similitud de los capítulos es: 34.17%
El porcentaje similitud de los temas es: 9.95%
El porcentaje similitud total de las asignaturas es: 23.07%
```

Figura 4.5: Resultados de la ejecución del algoritmo de similitud coeficiente de Sorensen-Dice entre el sílabo de Inteligencia Artificial y el de Redes Neuronales.

# Capítulo 5

# Conclusiones y trabajos futuros

## 5.1. Conclusiones

El esfuerzo realizado en este trabajo de titulación con el fin de reducir el proceso que se lleva a cabo en la movilidad estudiantil en la Universidad de Cuenca. Con el fin de lograr los objetivos planteados se realizó varias investigaciones sobre tecnologías semánticas, las cuales ayudaron a proponer una ontología para el sistema de sílabos. Así mismo se realizó investigaciones minuciosas de minería de textos, con el propósito de seleccionar los mejores algoritmos para la identificación de las similitudes entre los diferentes sílabos.

### 5.1.1. Objetivos alcanzados

Como se describió en el capítulo 1.4, el presente trabajo de titulación ha sido desarrollado mediante 5 objetivos específicos.

1. Crear o extender una ontología para el dominio de los sílabos universitarios basada en la investigación de otras similares.
   En el capítulo 3.3 se divide en 3 actividades específicas que ayudan a proponer el vocabulario ontológico para la representación de sílabos universitarios.

2. Usar las técnicas de minería de textos sobre la ontología de sílabos para comparar la semejanza del contenido académico impartido por dos asignaturas similares.

En el capítulo 3.4 se describe sobre la aplicación de la minería de texto a la ontología, Así mismo se especifica los respectivos algoritmos de Procesamiento de Lenguaje Natural aplicadas para las comparaciones de similitud entre sílabos.

3. Validar los resultados de la comparación para determinar el nivel de éxito de la ontología y la minería de textos aplicada.

4. Mejorar la ontología y el proceso de minería de textos hasta que los resultados cumplan con un nivel de éxito adecuado al contexto universitario.

5. Crear un prototipo que permita la automatización del proceso de verificación.

En el capítulo 4 de Evaluación y resultados se puede ver los objetivos 3, 4 y 5 cumplidos, las evaluaciones que se realizó sobre la ontología permitió validar y verificar que cumpla los requisitos específicos para la cual fue propuesta. Con respeto a la minería de texto, se realizó una revisión y pruebas minuciosas sobre los algoritmos seleccionados. las diferentes pruebas aplicadas permitió verificar el porcentaje de similitudes que existen entre los sílabos seleccionados.

## 5.2. Trabajos Futuros

Debido a que la implantación de la ontología para el sistema de sílabos de la Universidad de Cuenca aún se encuentra en proceso, como trabajo futuro se propone la puesta en ejecución para otros IES.

Como se menciona en el capítulo 3 en la sección de minería de texto para este proyecto se realizaron las pruebas sobre ciertos algoritmos. Se plantea la investigación de otros algoritmos que tengan relación al proceso de movilidad estudiantil.

También, se puede utilizar como datos complementarios a la descripción y contenidos académicos, el número de cŕeditos y horas totales de una asignatura. Estos podrían ser factores que mejoren la exactitud al momento de obtener un porcentaje de similitud entre dos asignaturas.

Con el fin de enriquecer las instancias de la ontología, se pueden utilizar fuentes externas de datos, tales fuentes permitirán obtener el significado de la

terminología utilizada en los sílabos a comparar.

Para mejorar la efectividad de los algoritmos de similitud, se puede utilizar la bibliografía citada en el contenido académico de los sílabos; de esta manera el proceso de comparación incluiría la información de fuentes base.

Finalmente, se propone realizar un análisis de similitud de documentos completos que contengan todo el contenido de un sílabo.

# Apéndice A

# Anexos

## A.1. Ontologías

### A.1.1. Comparación de Ontologías

| | | |
|---|---|---|
| **1** | **Identificación del tipo de ontologías generales que se va a utilizar** | |
| Las ontologías generales a utilizar deben ser relevantes en el dominio universitario, bibliográfico, estructura documental y sobre personas. | | |
| **2** | **Identificación de definiciones y axiomas más significativos que caracterizan la teoría** | |
| Las definiciones más significativas obtenidas en el escenario 2 son las siguientes: Asignatura, Facultad, Carrera, Contenido Académico, Capítulo y Tema. | | |
| **3** | **Búsqueda de ontologías generales soportadas por la teoría** | |
| Las ontologías generales soportadas por la teoría son: FOAF, DoCO, AIISO y BIBO. | | |
| **4** | **Realización de un estudio comparativo** | |
| Cada ontología seleccionada es descrita para proporcionar información sobre ella. | | |
| | **Nombre** | **Descripción** |

| | | |
|---|---|---|
| | FOAF | *Friend of a Friend vocabulary* (foaf): En Kalemi [44] se define esta ontología como un proyecto aplicado en las tecnologías de la web semántica para las redes sociales. Este proyecto describe un vocabulario ontológico en el dominio de las personas y sus conexiones contemplando tanto propiedades de una persona como de las relaciones que tiene con otras. La URI del espacio de nombres del vocabulario es `http://xmlns.com/foaf/0.1` y cuenta con 13 clases y 62 propiedades. |
| | AIISO | *Academic Institution Internal Structure Ontology* (AIISO): En Mouromtsev et al. [45], se la define como una ontología que proporciona clases y propiedades para describir la estructura organizacional interna de una institución académica. Está diseñada para trabajar en asociaciones con participación, (`http://purl.org/vocab/participation/schema`), FOAF (`http://xmlns.com/foaf/0.1/`) y aiiso-roles (`http://purl.org/vocab/aiiso-roles/schema`),para describir los roles que las personas juegan dentro de una institución [46]. El URI para este vocabulario es `http://purl.org/vocab/aiiso/schema#`, cuenta con 15 clases y 10 propiedades entre las cuales se encuentran: Center, College, Course, Department, Division, Faculty, Institute, Institution, Knowledge Grouping, Module, Programme, Research Group, School. |
| | DoCO | *Document Components Ontology* (DoCO): En Constantin et al. [47] se la define como una ontología que caracteriza la forma estructural de un documento con todos los elementos que puede contener. Por lo cual provee un vocabulario en el dominio retórico connotativo sobre los componentes pertenecientes a documentos. La URI del espacio de nombres del vocabulario es `http://purl.org/spar/doco` y cuenta con 53 clases que posee la ontología. |

| | | |
|---|---|---|
| | BIBO | *The Bibliographic Ontology (bibo): En Surla* [48] se la define como una ontología que provee un gran número de conceptos y propiedades para describir citaciones y referencias bibliográficas. Está diseñada para trabajar en asociación con la ontología Dublin Core (`http://purl.org/dc/terms/`) en la descripción de fechas y autoría de recursos.La URI del espacio de nombres del vocabulario es `http://purl.org/ontology/bibo/` y cuenta con 58 clases y 67 propiedades. El conjunto de clases permite identificar el tipo de documento que se quiere referenciar (libros, artículos, revistas, páginas web) para lo cual incluye algunos conceptos de la ontología Dublin Core y de la FOAF [49]. |
| | Bowlogna | Bowlogna: En Demartini et al. [8] se la define como una ontología que describe el entorno universitario definido por la reforma Bologna de 1999. La ontología cuenta con 66 clases que describen conceptos como estudiantes, profesores, evaluaciones, unidades de enseñanza, certificados, programas de estudio, entre otros. Los términos léxicos están en idioma Inglés, Francés, Alemán e Italiano. |

Tabla A.1: Tipos de ontologías.

### A.1.2. Ontologías Selecionadas

| **5** | **Selección de las ontología generales** | |
|---|---|---|
| Cada ontología seleccionada es descrita para especificar la razón por la que fue seleccionada. | | |
| | **Nombre** | **Descripción** |
| | FOAF | Esta ontología es una de las más utilizadas en el dominio de las personas, lo cual implica ventajas en cuanto a compatibilidad con otras ontologías. Por esta razón se la usará para definir a las personas (profesores o coordinadores) dentro del vocabulario ontológico para la representación de sílabos. |

| | | |
|---|---|---|
| | AIISO | Esta ontología abarca de manera sencilla y amplía la estructura organizacional de la academia, además permite representar y englobar cualquier tipo de conocimiento dentro de ella. Es por esta razón que se la puede utilizar para describir el contenido de un sílabo; como la asignatura con sus respectivos objetivos, logros, capítulos, temas e incluso la facultad y carrera a la que pertenece. |
| | DoCO | Esta ontología permite representar la estructura de cualquier obra literaria. Por esta razón se la va a utilizar para describir la estructura del contenido académico, específicamente a los capítulos y temas. |
| | BIBO | Esta ontología permite describir citas bibliográficas. Por esta razón se la puede utilizar para representar las referencias bibliográficas correspondientes al contenido académico dentro de la ontología de sílabos. |

Tabla A.2: Ontologías Reutilizadas

### A.1.3. Propiedades y Clases

| **6** | **Personalización de las ontologías generales seleccionadas** | |
|---|---|---|
| Se describe que Clases y Propiedades de partir de las ontologías seleccionadas se utilizarán dentro de la ontología de sílabos. | | |
| Las clases y propiedades utilizadas de la ontología FOAF son las siguientes. | | |
| **Clase o Propiedad** | **IRI** | **Descripción** |
| Agent | http://xmlns.com/foaf/0.1/Agent | La etiqueta en español de la clase es: Agente. |
| Organization | http://xmlns.com/foaf/0.1/Organization | La etiqueta en español de la clase es: Organización. |
| Person | http://xmlns.com/foaf/0.1/Person | La etiqueta en español de la clase es: Persona. |

| | | |
|---|---|---|
| familyName | http://xmlns.com/foaf/0.1/familyName | La etiqueta en español de la propiedad es: apellido autor. |
| firstName | http://xmlns.com/foaf/0.1/firstName | La etiqueta en español de la propiedad es: nombre autor. |
| givenName | http://xmlns.com/foaf/0.1/givenName | La etiqueta en español de la propiedad es: nombres. |
| lastName | http://xmlns.com/foaf/0.1/lastName | La etiqueta en español de la propiedad es: apellidos. |
| Las clases y propiedades utilizadas de la ontología AIISO son la siguientes. | | |
| **Clase o Propiedad** | **IRI** | **Descripción** |
| Collage | http://purl.org/vocab/aiiso/schema #College | La etiqueta en español de la clase es: Universidad. |
| Knowledge Grouping | http://purl.org/vocab/aiiso/schema #KnowledgeGrouping | La etiqueta en español de la clase es: Conocimiento del grupo. |
| Faculty | http://purl.org/vocab/aiiso/schema #Faculty | La etiqueta en español de la clase es: Facultad. |
| School | http://purl.org/vocab/aiiso/schema #School | La etiqueta en español de la clase es: Escuela. |

| | | |
|---|---|---|
| Subject | http://purl.org/vocab/aiiso/schema #Subject | La etiqueta en español de la clase es: Materia. |
| Description | http://purl.org/vocab/aiiso/schema #description | La etiqueta en español de la clase es: Descripción. |
| Code | http://purl.org/vocab/aiiso/schema #code | La etiqueta en español de la clase es: Código. |
| responsability Of | http://purl.org/vocab/aiiso/schema #responsibilityOf | La etiqueta en español de la propiedad es: responsabilidad de. |
| responsability For | http://purl.org/vocab/aiiso/schema #responsibleFor | La etiqueta en español de la propiedad es: responsable de. |
| teaches | http://purl.org/vocab/aiiso/schema #teaches | La etiqueta en español de la propiedad es: enseña. |
| part_of | http://purl.org/vocab/aiiso/schema #part_of | La etiqueta en español de la propiedad es: parte de. |
| Las clases y propiedades utilizadas de la ontología DoCO son la siguientes. | | |
| **Clase o Propiedad** | **IRI** | **Descripción** |
| Bibliography | http://purl.org/spar/doco/Bibliography | La etiqueta en español de la clase es: Bibliografía. |

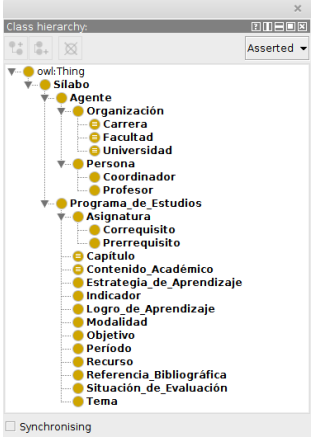
| | | |
|---|---|---|
| Chapter | http://purl.org/spar/doco/Chapter | La etiqueta en español de la clase es: Capítulo. |
| Section | http://purl.org/spar/doco/Section | La etiqueta en español de la clase es: Sección. |
| Las clases y propiedades utilizadas de la ontología BIBO son la siguientes. | | |
| **Clase o Propiedad** | **IRI** | **Descripción** |
| chapter | http://purl.org/dc/terms/chapter | La etiqueta en español de la propiedad es: número de capítulo. |
| created | http://purl.org/dc/terms/created | La etiqueta en español de la propiedad es: fecha de creación. |
| edition | http://purl.org/ontology/bibo/edition | La etiqueta en español de la propiedad es: edición. |
| editor | http://purl.org/ontology/bibo/editor | La etiqueta en español de la propiedad es: editor. |
| date | http://purl.org/dc/terms/date | La etiqueta en español de la propiedad es: fecha. |
| doi | http://purl.org/ontology/bibo/doi | La etiqueta en español de la propiedad es: doi. |

| | | |
|---|---|---|
| familyName | http://purl.org/ontology/bibo/familyName | La etiqueta en español de la propiedad es: apellido autor. |
| givenName | http://purl.org/ontology/bibo/givenName | La etiqueta en español de la propiedad es: nombre autor. |
| isbn | http://purl.org/ontology/bibo/isbn | La etiqueta en español de la propiedad es: isbn. |
| issn | http://purl.org/ontology/bibo/issn | La etiqueta en español de la propiedad es: issn. |
| issued | http://purl.org/dc/terms/created | La etiqueta en español de la propiedad es: fecha de publicación. |
| owner | http://purl.org/ontology/bibo/owner | La etiqueta en español de la propiedad es: autor. |
| pageEnd | http://purl.org/ontology/bibo/pageEnd | La etiqueta en español de la propiedad es: página de fin. |
| pageStart | http://purl.org/ontology/bibo/pageStart | La etiqueta en español de la propiedad es: página de inicio. |

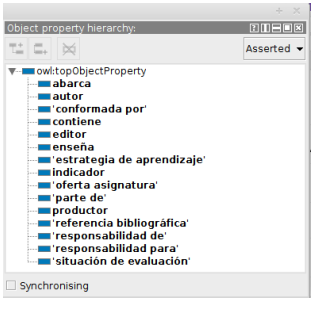
| | | |
|---|---|---|
| pages | http://purl.org/ontology/bibo/pages | La etiqueta en español de la propiedad es: rango de páginas. |
| producer | http://purl.org/ontology/bibo/producer | La etiqueta en español de la propiedad es: productor. |
| section | http://purl.org/ontology/bibo/section | La etiqueta en español de la propiedad es: sección. |
| shortTitle | http://purl.org/ontology/bibo/shortTitle | La etiqueta en español de la propiedad es: título de la obra. |
| uri | http://purl.org/ontology/bibo/uri | La etiqueta en español de la propiedad es: uri. |
| volume | http://purl.org/ontology/bibo/volume | La etiqueta en español de la propiedad es: volumen. |
| **7** | **Integración de las ontologías generales en la ontología que va a ser desarrollada** | |
| Las clases y jerarquías de la ontología para la representación de sílabos se muestran en la Figura A.1a a continuación. En ella se puede observar que se utilizó una definición jerárquica de top-down, por lo en los niveles superiores se encuentran las clases más generales y en los inferiores las más específicas. | | |
| Las propiedades referentes a las clases que describen la estructura interna de los conceptos se muestran en la Figura A.1b. En ella se puede observar que se utilizó una definición jerárquica de top-down, por lo en los niveles superiores se encuentran las propiedades más generales y en los inferiores las más específicas. | | |

| |
|---|
| Las propiedades referentes a las relaciones entre los conceptos se muestran en la Figura A.1c. En ella se puede observar que se no se utilizó una definición jerárquica en específico. |

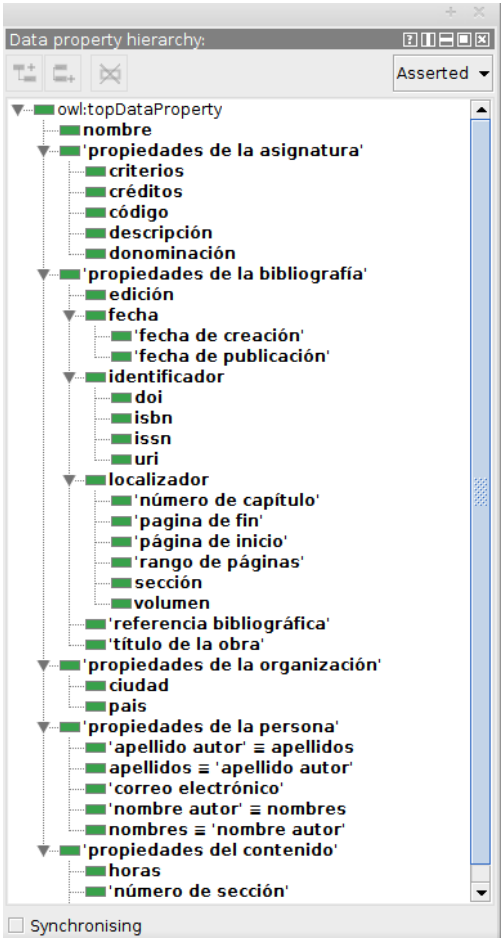
Tabla A.3: Descripción de propiedades y clases.




(a) Jerarquía de clases pertenecientes al vocabulario ontológico para sílabos.




(b) Jerarquía de propiedades de objetos pertenecientes al vocabulario ontológico para sílabos.




(c) Jerarquía de propiedades de datos pertenecientes al vocabulario ontológico para sílabos.

## A.2. Algoritmos

```
#------------------ Importa las librerias ------------------
import numpy as np
import nltk, string
from nltk.stem import *
from stop_words import get_stop_words
from sklearn.feature_extraction.text import TfidfVectorizer

#------------ Definicion de Variables Globales ------------
tweet_tokenizer = TweetTokenizer()
snowball_stemmer = SpanishStemmer()
lancaster_stemmer = LancasterStemmer()
porter_stemmer = PorterStemmer()
remove_punctuation_map = dict((ord(char), None) for char in
    string.punctuation)

#----------------- Definicion de Funciones ----------------
def normalization(text):
    return text.lower().translate(remove_punctuation_map)
def tweet_tokenization(text):
    return tweet_tokenizer.tokenize(text)
def word_tokenization(text):
    return word_tokenize(text)
def stopping(tokens):
    return sorted(set([token for token in tokens if token not in
     get_stop_words('spanish')]))
def snowball_stemming(stopwords):
    return set([snowball_stemmer.stem(word) for word in
    stopwords])
def lancaster_stemming(stopwords):
    return [lancaster_stemmer.stem(word) for word in stopwords]
def porter_stemming(stopwords):
    return [porter_stemmer.stem(word) for word in stopwords]
```

Figura A.2: Algoritmos utilizados para implementar las técnicas de preprocesamiento de textos.

```python
#------------------ Importa las librerias ------------------
import numpy as np
import nltk, string
from nltk.stem import *
from nltk import word_tokenize
from stop_words import get_stop_words
from nltk.metrics import masi_distance
from nltk.tokenize import TweetTokenizer
from nltk.stem.porter import PorterStemmer
from nltk.stem.snowball import SpanishStemmer
from nltk.stem.lancaster import LancasterStemmer
from sklearn.feature_extraction.text import TfidfVectorizer

#----------------- Definicion de Funciones ----------------
def cosine_similarity(text1, text2):
    vectorizer = TfidfVectorizer(tokenizer=word_tokenization)
    tf_idf = vectorizer.fit_transform([text1, text2])
    return ((tf_idf * tf_idf.T).A)[0,1]

def jaccard_similarity(a, b):
    c = a.intersection(b)
    return float(len(c)) / (len(a) + len(b) - len(c))

def dices_similarity(a, b):
    c = a.intersection(b)
    return float(2*len(c)) / (len(a) + len(b))

def overlap_similarity(a, b):
    c = a.intersection(b)
    a = np.array(a)
    b = np.array(b)
    minValue = np.min((a.min(), b.min()))
    return float(len(c)) / float(len(minValue))
```

Figura A.3: Algoritmos utilizados para implementar los índices de similitud entre textos.

```
#------------------ Importa las librerias ------------------
import psycopg2, psycopg2.extras
import Nltk_Similarity, Gensim_Similarity
from Nltk_Similarity import *

#----------------- Definicion de Funciones ----------------
def calculo_similitud_capitulos(asignatura1, asignatura2):
    similitud_capitulos = {}
    porcentaje_similitud = 0
    longitud = len(asignatura1)
    for cap_asignatura1 in asignatura1:
        for cap_asignatura2 in asignatura2:
            pro_asignatura1 = preprocesamiento(cap_asignatura1)
            pro_asignatura2 = preprocesamiento(cap_asignatura2)
            similitud = dices_similarity(pro_asignatura1,
    pro_asignatura2)
            if cap_asignatura1 in similitud_capitulos:
                temp = similitud_capitulos[cap_asignatura1][0]
                if similitud >= temp:
                    similitud_capitulos[cap_asignatura1] = [
    similitud, cap_asignatura2]
            else:
                similitud_capitulos[cap_asignatura1] = [
    similitud, cap_asignatura2]
        porcentaje_similitud += similitud_capitulos[
    cap_asignatura1][0]
    return [porcentaje_similitud/longitud, similitud_capitulos]

#-------------- Ejecucion de los Algoritmos --------------
porcentaje_similitud_descripciones = dices_similarity(
    texto1_stemming,texto2_stemming)
[porcentaje_similitud_capitulos, similitud_capitulos] =
calculo_similitud_capitulos(capitulos_asignatura1.keys(),
    capitulos_asignatura2.keys())
porcentaje_similitud_total = porcentaje_similitud_descripciones)
    *float(1/5) + (porcentaje_similitud_capitulos +
    porcentaje_similitud_subcapitulos)*float(2/5)
```

Figura A.4: Funcionamiento de la comparación entre sílabos utilizando los algoritmos de NLP.

# Bibliografía


[1] W3C. (2007) Latest layercake diagram. [Accessed 12 Nov. 2017]. [Online]. Available: https://www.w3.org/2007/03/layerCake.png

[2] G. Klyne and J. J. Carrol. (2004) Resource description framework (rdf):concepts and abstract syntax. [Accessed 12 Nov. 2017]. [Online]. Available: https://www.w3.org/TR/2004/REC-rdf-concepts-20040210/

[3] A. Gómez-Pérez and M. C. Suárez-Figueroa, "Neon methodology for building ontology networks: a scenario-based methodology," 2009.

[4] O. E. G. (OEG). (2015) La metodología neon. [Accessed 29 Dec. 2017]. [Online]. Available: http://mayor2.dia.fi.upm.es/oeg-upm/index.php/es/methodologies/59-neon-methodology/index.html

[5] G. Miner, J. Elder IV, and T. Hill, *Practical text mining and statistical analysis for non-structured text data applications*. Academic Press, 2012.

[6] D. McGuinness and F. Harmelen. (2009) Owl web ontology language overview. [Accessed 13 Dec. 2017]. [Online]. Available: https://www.w3.org/TR/owl-features/

[7] H. Chung and J. Kim, "An ontological approach for semantic modeling of curriculum and syllabus in higher education," *International Journal of Information and Education Technology*, vol. 6, no. 5, p. 365, 2016.

[8] E. Eremin, "About ontology application to the description of syllabus," 2008.

[9] J. Chicaiza, N. Piedra, and J. López-Vargas, "Diseño de un vocabulario para conectar e interoperar los syllabus de cursos ocw mediante tecnologías de la web semántica."

[10] L. Codina and C. Rovira, "La web semántica," in *Tendencias en documentación digital*. Trea, 2006.

[11] S. Hawke, I. Herman, and E. Prud'hommeaux. (2013) W3c semantic web activity homepage. [Accessed 23 Dec. 2017]. [Online]. Available: https://www.w3.org/2001/sw/

[12] L. Quin. (2013) Extensible markup languaje (xml). [Accessed 27 Oct. 2017]. [Online]. Available: https://www.w3.org/XML/

[13] W3C. (1999) Guía breve de tecnologías xml. [Accessed 27 Oct. 2017]. [Online]. Available: https://www.w3c.es/Divulgacion/GuiasBreves/TecnologiasXML

[14] C. Sperberg-McQueen and H. Thompson. (2000) W3c xml schema. [Accessed 27 Oct. 2017]. [Online]. Available: https://www.w3.org/XML/Schema

[15] et al. (2014) Rdf - semantic web standards. [Accessed 27 Oct. 2017]. [Online]. Available: https://www.w3.org/RDF/

[16] P. Castells, "La web semántica," *Sistemas interactivos y colaborativos en la web*, pp. 195–212, 2003.

[17] R. Pedraza-Jiménez, L. Codina, and C. Rovira, "Web semántica y ontologías en el procesamiento de la información documental," *El profesional de la información*, vol. 16, no. 6, pp. 569–578, 2007.

[18] T. Gruber, "What is an ontology," *WWW Site http://www-ksl. stanford. edu/kst/whatis-an-ontology. html (accessed on 07-09-2004)*, 1993.

[19] N. F. Noy, D. L. McGuinness *et al.*, "Ontology development 101: A guide to creating your first ontology," 2001.

[20] M. Barrera, H. Núñez, and E. Ramos, *Ingeniería Ontológica*. Caracas - Venezuela: Centro de Ingeniería de Software y Sistemas (ISYS), Laboratorio de Inteligencia Artificial (LIA), 2012, vol. 1.

[21] W. W. W. Consortium. (2008) Sparql lenguaje de consulta para rdf. [Accessed 12 Nov. 2017]. [Online]. Available: http://skos.um.es/TR/rdf-sparql-query/

[22] M. Arenas, C. Gutierrez, and J. Pérez, "On the semantics of sparql," in *Semantic Web Information Management*. Springer, 2010, pp. 281–307.

[23] J. A. G. Luna, M. L. Bonilla, and I. D. Torres, "Metodologías y métodos para la construcción de ontologías," *Scientia et technica*, vol. 2, no. 50, pp. 133–140, 2012.

[24] M. Uschold and M. King, *Towards a methodology for building ontologies.* Artificial Intelligence Applications Institute, University of Edinburgh Edinburgh, 1995.

[25] M. Grüninger and M. S. Fox, "Methodology for the design and evaluation of ontologies," 1995.

[26] G. Schreiber, B. Wielinga, W. Jansweijer *et al.*, "The kactus view on the 'o'word," in *IJCAI workshop on basic ontological issues in knowledge sharing*, 1995, pp. 159–168.

[27] A. Silva-Sprock, V. Miguel, M. G. López, L. Ramos, N. Montaño, and O. Villarroel, "Hacia un proceso de ingenieria del conocimiento en la creacion de la ontologia de ambar," in *Universidad 2008. Congreso Internacional de la Educacion Superior,(5to. Congreso: 2008: Palacio de las Convenciones, Cuba)*, no. 378 378. e-libro, Corp., 2008.

[28] K. Uyi Idehen. (2008) What is linked data oriented rdf-ization? [Accessed 15 Feb. 2018]. [Online]. Available: https://www.openlinksw.com/blog/~kidehen/index.vspx?page=&id=1453&cmf=1

[29] R. Feldman and I. Dagan, "Knowledge discovery in textual databases (kdt)." in *KDD*, vol. 95, 1995, pp. 112–117.

[30] R. Feldman and J. Sanger, *The text mining handbook: advanced approaches in analyzing unstructured data.* Cambridge university press, 2007.

[31] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D. Trippe, J. B. Gutierrez, and K. Kochut, "A brief survey of text mining: Classification, clustering and extraction techniques," *arXiv preprint arXiv:1707.02919*, 2017.

[32] M. Garre, J. J. Cuadrado, M. A. Sicilia, D. Rodríguez, and R. Rejas, "Comparación de diferentes algoritmos de clustering en la estimación de coste en el desarrollo de software," *REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software*, vol. 3, no. 1, 2007.

[33] D. Pascual, F. Pla, and S. Sánchez, "Algoritmos de agrupamiento," *Método Informáticos Avanzados*, pp. 164–174, 2007.

[34] N. Sharma, A. Bajpai, and M. R. Litoriya, "Comparison the various clustering algorithms of weka tools," *facilities*, vol. 4, no. 7, 2012.

[35] M. Singh, H. Nagar, and A. Sant, "K-mean and em clustering algorithm using attendance performance improvement primary school student," 2016.

[36] C. G. Cambronero and I. G. Moreno, "Algoritmos de aprendizaje: knn & kmeans," *Intelgencia en Redes de Comunicación, Universidad Carlos III de Madrid*, 2006.

[37] J. Pérez, M. Henriques, R. Pazos, L. Cruz, G. Reyes, J. Salinas, and A. Mexicano, "Mejora al algoritmo de agrupamiento k-means mediante un nuevo criterio de convergencia y su aplicación a bases de datos poblacionales de cáncer," *Liver-2do Taller Latino Iberoamericano de Investigacion de Operaciones "la IO aplicada a la solución de problemas regionales*, pp. 1–7, 2007.

[38] A. Copestake, "Natural languaje processing," 2004.

[39] K. Sumathy and M. Chidambaram, "Text mining: concepts, applications, tools and issues-an overview," *International Journal of Computer Applications*, vol. 80, no. 4, 2013.

[40] M. Vallez and R. Pedraza, "El procesamiento del lenguaje natural en la recuperación de información textual y áreas afines," *Hipertext. net*, 2007.

[41] A. Jain, A. Jain, N. Chauhan, V. Singh, and N. Thakur, "Information retrieval using cosine and jaccard similarity measures in vector space model," *International Journal of Computer Applications*, vol. 164, no. 6, 2017.

[42] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[43] M. Vijaymeena and K. Kavitha, "A survey on similarity measures in text mining," *Machine Learning and Applications: An International Journal*, vol. 3, no. 2, pp. 19–28, 2016.

[44] E. Kalemi, "Foaf and its application," *ICT Innovations*, vol. 2009, 2009.

[45] D. Mouromtsev, F. Kozlov, O. Parkhimovich, and M. Zelenina, "Development of an ontology-based e-learning system," in *International Conference on Knowledge Engineering and the Semantic Web*. Springer, 2013, pp. 273–280.

[46] R. Styles and N. Shabir. (2008) Academic institution internal structure ontology (aiiso). [Accessed 1 Dec. 2017]. [Online]. Available: http://vocab.org/aiiso/schema

[47] A. Constantin, S. Peroni, S. Pettifer, D. Shotton, and F. Vitali, "The document components ontology (doco)," *Semantic Web*, vol. 7, no. 2, pp. 167–181, 2016.

[48] B. Dimić Surla, M. Segedinac, and D. Ivanović, "A bibo ontology extension for evaluation of scientific research results," in *Proceedings of the Fifth Balkan Conference in Informatics*. ACM, 2012, pp. 275–278.

[49] A. Ruiz-Iniesta and O. Corcho, "A review of ontologies for describing scholarly and scientific documents." in *SePublica*, 2014.

[50] T. M. Mitchell, *Machine Learning*, 1997.

[51] L. Codina, "¿ web 2.0, web 3.0 o web semántica?: El impacto en los sistemas de información de la web," in *Congreso Internacional de Ciberperiodismo y Web*, vol. 2, no. 1, 2009.

[52] M. Montes and G. L. de Lenguaje Natural, "Minería de texto: Un nuevo reto computacional," *Obtenido de https://ccc. inaoep. mx/˜ mmontesg/publicaciones/2001/MineriaTexto-md01. pdf*, 2001.

[53] D. Brickley and L. Miller. (2014) Foaf vocabulary specification. [Accessed 24 Nov. 2017]. [Online]. Available: http://xmlns.com/foaf/spec/

[54] D. Shotton and S. Peroni. (2015) Doco, the document components ontology. [Accessed 24 Nov. 2017]. [Online]. Available: http://purl.org/spar/doco

[55] E. Prud'hommeaux and A. Seaborne. (2008) Sparql query language for rdf. [Accessed 26 Nov. 2017]. [Online]. Available: https://www.w3.org/TR/rdf-sparql-query/

[56] M. Kifer and H. Boley. (2013) Rif overview (second edition). [Accessed 13 Dec. 2017]. [Online]. Available: https://www.w3.org/TR/rif-overview/