RNPS No. 2142
ISSN 2072-6287
Versión Digital

SERIE AZUL

CENATAV
Centro de Aplicaciones de Tecnologías de Avanzada
MINISTERIO DE LA INDUSTRIA BÁSICA

REPORTE TÉCNICO
Reconocimiento de Patrones

Alineamiento de ontologías en el dominio geoespacial

Lic. Francisco Vera Voronisky,
Dr. C. Eduardo Garea Llano

RT_010 **Diciembre 2009**

7ma. No. 21812 e/218 y 222,
Rpto. Siboney, Playa;
Ciudad de La Habana.
Cuba. C.P. 12200
www.cenatav.co.cu

RNPS No. 2142
ISSN 2072-6287
Versión Digital

SERIE AZUL

CENATAV
Centro de Aplicaciones de Tecnologías de Avanzada
MINISTERIO DE LA INDUSTRIA BÁSICA

REPORTE TÉCNICO
Reconocimiento de Patrones

Alineamiento de ontologías en el dominio geoespacial

Lic. Francisco Vera Voronisky,
Dr. C. Eduardo Garea Llano

RT_010 **Diciembre 2009**

7ma. No. 21812 e/218 y 222,
Rpto. Siboney, Playa;
Ciudad de La Habana.
Cuba. C.P. 12200
www.cenatav.co.cu

Alineamiento de ontologías en el dominio geoespacial

Lic. Francisco Vera Voronisky, Dr. C. Eduardo Garea Llano

Centro de Aplicaciones de Tecnología de Avanzada, 7a #21812 e/ 218 y 222, Siboney, Playa, Habana, Cuba
fvera@cenatav.co.cu

RT_010 CENATAV
Fecha del camera ready: 28 de mayo de 2009

Resumen. El uso de geo-ontologías en los sistemas de información geográficos ha sido útil para insertar información semántica a los objetos representados. Cuando se pretende utilizar dos o más ontologías, la información de un elemento específico que esté contenido en varias ontologías puede estar representadas de diferentes maneras en cada una de ellas, pudiendo generar problemas al no reconocer estas representaciones como equivalentes. El alineamiento de ontologías es el proceso que se realiza para determinar un conjunto de objetos equivalentes pertenecientes a las ontologías que participan en éste. En este documento se expondrán las técnicas para alinear ontologías, así como las herramientas principales desarrolladas que realizan esta operación. Se mostrará la extensión de este problema aplicado al dominio geoespacial, así como los trabajos principales realizados en este campo.

Palabras claves: ontología, geo-ontología, alineamiento de ontología, comparación de ontología, medidas de similitud

Abstract. The use of geo-ontologies in geographic information systems has been useful to add semantic information to the objects which the geographic object represented. When two or more ontologies are used, the information of a geographic object described in those ontologies may be represented in a different way. This can cause that these representations would not be recognized as equivalent. Ontology alignment is a process that is performed to determine a set of equivalent objects from the ontologies. This work explains the ontology alignment techniques as well as the main frameworks developed to perform this operation. It will explain how this problem is applied to the geospatial domain, and the main works that have been done in this area is mentioned.

Keywords: Ontology, Geo-ontology, Ontology alignment, Ontology matching, Similarity measure

1 Introducción

El uso de ontologías como forma de representación de conocimiento es uno de los campos de investigación que ha surgido recientemente, debido que las ontologías son estructuras que representan contenido semántico rico basado en teorías lógicas.

Los sistemas de referencia semánticos usualmente confían en el uso de ontologías, debido a que las ontologías proveen definiciones formales explícitas de entidades temáticas y sus relaciones y por lo tanto, facilitan las definiciones de métodos para proyectar, trasladar e integrar información geográfica obtenida de diferentes fuentes. Las ontologías son el corazón de los sistemas de información geográfica (GIS, por sus siglas en inglés) que utilizan información semántica, ejemplo de un GIS gobernado por ontologías es ODGIS, el cual fue propuesto por Fonseca et al. [1].

Las ontologías que tratan la temática geoespacial presentan características particulares de este campo de estudio. Debido a su nivel de especialización, a estas ontologías geográficas se les

han denominado geo-ontologías. Las geo-ontologías cumplen con todas la características de una ontología *convencional*, pero tienen además propiedades propias de este dominio, por ejemplo, un par de coordenadas (x,y) que representa la posición geográfica de un objeto. También incluye una serie de relaciones topológicas con una semántica determinada, por

ejemplo, en el siguiente planteamiento: "el *río cruza* el *bosque"*, podemos encontrar los conceptos (objetos) *río* y *bosque*, y también encontramos una relación *cruza* que actúa sobre los objetos río y bosque.

Diversas instituciones se han dedicado a reunir y representar datos de una misma región espacial mediante geo-ontologías, pero en cada una se puede utilizar un nombre diferente para identificar a un mismo objeto. Por ejemplo, la palabra autopista, en California, se le conoce como *freeway*, sin embargo en el Reino Unido se le llama *motorway*. Éste es un ejemplo de heterogeneidad de datos que puede ocurrir al querer integrar la información entre dos geo-ontologías. En general, se pueden encontrar otras heterogeneidades que impiden que se establezca una asociación directa entre los datos. Este tema lo trataremos en la sección 3.2.

Para poder trabajar con la información que nos brindan ambas ontologías, es necesario, establecer un vínculo, una correspondencia entre sus entidades, con el objetivo de compartir información entre ambas.

De aquí es que viene el campo de estudio de alineamiento de ontologías, el cual es el proceso de determinar un conjunto de correspondencias entre los conceptos pertenecientes a ontologías diferentes.

Para tratar el tema de las geo-ontologías, decidimos comenzar el estudio analizando las ontologías convencionales, sus definiciones, las técnicas utilizadas para calcular la similitud entre sus entidades, así como el proceso de alineamiento para, en una segunda parte, hacer una exploración sobre el dominio de las geo-ontologías en sí. Como las geo-ontologías son una especialización de las ontologías, las técnicas que se aplican a las ontologías convencionales pueden ser aplicadas en el dominio geoespacial, aunque con la limitante de que no considerarán sus características específicas, dejando de explotar rasgos que pudieran dar buenos resultados. Se entiende por buenos resultados el hecho de poder encontrar alineamientos que no son detectados por los métodos y poder rechazar falsos alineamientos.

El siguiente documento se dividirá en dos parte fundamentes. La primera parte tratará sobre las ontologías, sin considerar el dominio geoespacial. En esta parte se dará la clasificación de los métodos para la comparación entre entidades de dos ontologías diferentes dados por Euzenat y Shvaiko [2], donde exponen una gran variedad técnicas que pueden servir de base para el desarrollo de algoritmos adaptados para el dominio geoespacial. Además se expone el proceso de alineamiento definido por Ehrig [3] y se mencionan las principales herramientas desarrolladas para realizar el alineamiento.

En la segunda parte nos adentraremos en el tema geoespacial, mostrándose los tipos de heterogeneidades definidos por Hess et al. [4] y se realizará un estado del arte de los trabajos realizados específicamente en este campo.

2 Alineamiento de ontologías

En esta sección, mostraremos varias definiciones del término ontología. Solamente considerará, en esta parte las ontologías sin agregar información geográfica. En la próxima sección se ampliará el alineamiento para ontologías en el dominio geoespacial.

Se expondrán varias definiciones de ontologías dadas por distintos autores. Seguidamente, se mostrarán los métodos para calcular las similitudes entre los elementos de las ontologías, las cuales son fundamentales para determinar la existencia de un alineamiento entre dichas entidades, así como se mostrarán taxonomías y clasificaciones de los métodos para el cálculo de las similitudes según varios autores. Posteriormente, se explicarán los pasos a realizar en el

proceso general de alineamiento de ontologías. Finalmente, se mostrarán las medidas para la evaluación de los alineamientos.

2.1 Definiciones de ontología

En Filosofía, ontología es la teoría de "la naturaleza de las cosas o los tipos de existencia". Los filósofos griegos Sócrates y Aristóteles fueron los primeros en desarrollar los fundamentos de las ontologías. Sócrates introdujo la noción de ideas abstractas, una jerarquía entre ellas, y relaciones clase – instancia. Aristóteles agregó asociaciones lógicas. El resultado es un modelo bien estructurado capaz de describir el mundo real.

Para los matemáticos, las ontologías son percibidas como un grafo complejo que representa el conocimiento acerca del mundo. Este modelo es extendido con axiomas lógicos para permitir inferencias.

En la historia moderna, los primeros artículos que resumen a la ontología como disciplina filosófica fueron publicados alrededor de 1960 [5].

En la rama de Inteligencia Artificial y los investigadores web se ha adoptado el término "*ontología*" para sus necesidades. Actualmente, hay diferentes definiciones en la literatura de que cosa debe ser una ontología. Algunos de estas son discutidas por Guarino [6], en donde destaca la definición de Gruber [7] "*Una ontología es una especificación explícita de una conceptualización*". Una conceptualización se refiere a un modelo abstracto de algún fenómeno del mundo identificando los conceptos relevantes de ese fenómeno [8]. Explícita significa que los tipos de conceptos usados y las restricciones para su uso son definidos explícitamente. Esta definición es usualmente extendida con tres condiciones adicionales: "*Una ontología es una especificación* formal, explícita de una *conceptualización* compartida de un dominio de interés*"*, donde formal se refiere al hecho que una ontología pueda ser leída por una computadora (lo cual excluye al lenguaje natural). Compartida refleja la noción que una ontología captura conocimiento consensual, es decir, que no es privado para sólo un individuo. Compartida no necesariamente implica compartida globalmente, puede ser sólo a un grupo. Dominio de interés indica que, para un dominio de ontologías, una no es interesante para modelar el mundo entero pero en cambio, modelar algunas partes de un dominio es relevante.

Guarino [9] define a las ontologías como "*Una teoría lógica justificando el significado deseado de un vocabulario formal, es decir, su propósito ontológico para una conceptualización particular de mundo*". En este contexto, una ontología puede solamente especificar una conceptualización de una manera débil. Guarino [9] afirma que una ontología O consigna a una conceptualización C si O ha sido diseñada con el propósito a caracterizar a C y O aproxima a C. Esto permite a ontologías diferentes consignarse a la misma conceptualización de diferentes maneras. De esta forma, una ontología puede estar cercana a la conceptualización como otra ontología. Una ontología se acercará a la conceptualización agregando más axiomas o agregando más conceptos y conceptualizaciones. Como resultado, hace distinción entre ontologías sin refinar y ontologías refinadas. Típicamente, las ontologías refinadas (con más detalles) serán usadas como referencias mientras que las ontologías sin refinar (más genéricas) podrán ser compartidas. De acuerdo al nivel de generalización, Guarino distinguió cuatro tipos de ontologías: ontología de alto nivel (*top-level*, en inglés), ontología de dominio, ontología de tarea y ontología de aplicación. En la **Fig. 1** se muestran los cuatro tipos de ontologías.

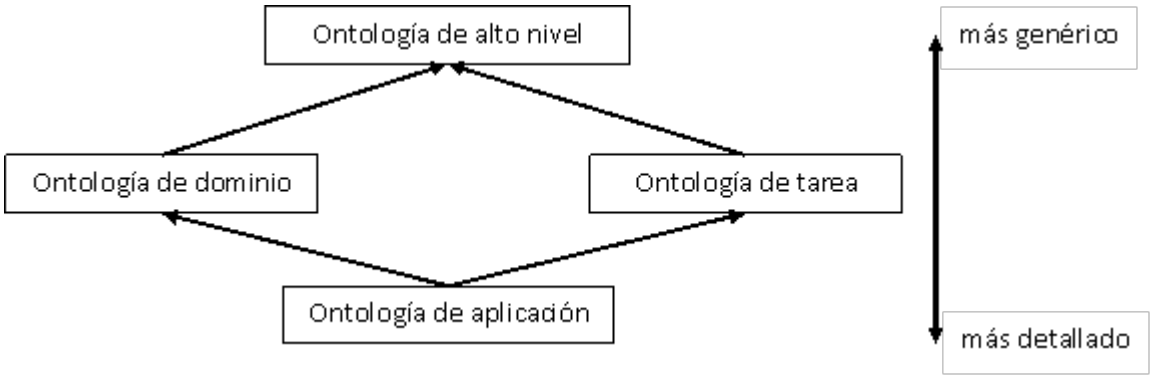

Fig. 1. Los cuatro tipos de ontologías según Guarino [9]

- Las *ontologías de alto nivel* describen los conceptos generales como el espacio, tiempo, materia, objeto, evento, acción, los cuales son independientes de un problema o dominio en particular.
- Las *ontologías de dominio* y de *tareas* describen, respectivamente, el vocabulario relacionado a un dominio genérico, por ejemplo, medicina, o una tarea o actividad genérica, como diagnosis, especializando los términos introducidos en la ontología de alto nivel.
- Las *ontologías de aplicación* describen conceptos dependiendo de un dominio y de una tarea en particular, la cual es una especialización de ambas ontologías relacionadas (ontología de dominio y ontología de tarea). Estos conceptos usualmente corresponden a un papel jugado por las entidades de dominio mientras realizan una actividad.

Algo común en estas definiciones es su alto nivel de generalización, que es lejano a una expresión matemática. La razón de esto es que la definición debería abrazar los diferentes tipos de ontologías, y no debería ser relativa a un método particular de representación [10].

2.1.1 Definición de ontología propuesta por Ehrig

Ehrig [3], utilizó la definición que ha sido desarrollada por el grupo de manejo de conocimientos del Instituto AIFB de la Universidad de Karlsruhe. Esta definición se adhiere al Modelo de Ontología de Karlsruhe expresado en Stumme et al. [11].

Definición 1 (Núcleo de ontología). Un núcleo de ontología es una estructura

$$S = (C, \leq_C, R, \leq_R)$$

consistente en:

- dos conjuntos disjuntos C y R cuyos elementos son llamados identificadores de conceptos e identificadores de relaciones (o simplemente conceptos y relaciones),
- un orden parcial $\leq_C$ en C, llamado jerarquía de conceptos o taxonomía,
- una función $\sigma: R \rightarrow C \times C$ llamada signatura, donde $\sigma(r) = \langle dom(r), ran(r) \rangle$ con $r \epsilon R$, dominio $dom(r)$, y rango $ran(r)$,
- un orden parcial $\leq_R$ en R, llamado jerarquía de relaciones, donde $r_1 \leq_R r_2$ si y solo si $dom(r_1) \leq_C dom(r_2)$ y $ran(r_1) \leq_C ran(r_2)$.

Para simplificar, los tipos de datos como enteros o cadenas de caracteres, estos son tratados como un tipo especial de conceptos, $D \subset C$. Además, diremos que si $c_1 <_C c_2$ para $c_1, c_2 \in C$, entonces c_1 es un subconcepto de c_2, y c_2 es un superconcepto de c_1. Denotamos esto por $c_1 \prec c_2$. Las super relaciones y sub relaciones se definen análogamente. El núcleo de ontología es usualmente referenciada como esquema.

Relaciones entre conceptos y/o relaciones así como restricciones pueden ser expresadas dentro de un lenguaje lógico como la lógica de primer orden o lógica de Horn.

Definición 2 (Axiomas). Sea L un lenguaje lógico. Un sistema *L-axioma* para un núcleo de ontología es un par

$$A = (AI, \alpha)$$

donde

- AI es un conjunto cuyos elementos son llamados identificadores de axiomas y
- $\alpha: AI \to L$ es una asociación.

Los elementos de $A := \alpha(AI)$ son llamados axiomas. S es considerada como parte del lenguaje *L*.

Los núcleos de ontología formalizan los aspectos intencionales de un dominio. Los aspectos extensionales son suministrados por las bases de conocimientos, las cuales contienen aserciones acerca de las instancias de los conceptos y relaciones.

Definición 3 (Base de conocimiento). Una base de conocimiento es una estructura

$$KB = (C, R, I, \iota_C, \iota_R)$$

consistente en:

- dos conjuntos disjuntos C y R definidos anteriormente,
- un conjunto I cuyos elementos son llamados identificadores de instancias (o instancias),
- una función $\iota_C: C \to \mathfrak{P}(I)$ llamados instanciación de conceptos,
- una función $\iota_R: R \to \mathfrak{P}(I^2)$ con $\iota_R(r) \subseteq \iota_C\big(dom(r)\big) \times \iota_C\big(ran(r)\big)$, para todo $r \in R$. La función ι_R es llamada instanciación de relaciones.

Tal como los tipos de datos son tratados como conceptos en el núcleo de ontología, los valores concretos serán análogamente tratados como instancias $V \subset I$.

A los conceptos (y relaciones), generalmente le asignamos un nombre. En lugar del nombre, podemos usar *sign* para permitir mayor generalidad.

Definición 4 (Lexicón): Un lexicón, para una ontología, es una estructura

$$Lex := (G_C, G_r, G_I, Ref_C, Ref_R, Ref_I)$$

consistente en

- tres conjuntos G_C, G_R, y G_I cuyos elementos son llamados *signs* para conceptos, relaciones e instancias, respectivamente.
- una relación $Ref_C \subseteq G_C \times C$ llamado referencia léxica para conceptos, Ref_R y Ref_I análogamente.

Para Ehrig, una ontología consiste en un núcleo de ontología, axiomas, datos instanciados en la base de conocimientos y un lexicón correspondiente.

Definición 5 (Ontología). Una ontología O es definida a través de la siguiente tupla:

$$= (S, A, KB, Lex)$$

consistente en

- un núcleo de ontología S,

- un sistema *L-axioma A*,
- una base de conocimiento KB y
- un lexicón Lex.

En este documento, posteriormente haremos referencia a un conjunto de entidades E. Una entidad $e \in E$ interpretada en una ontología O es un concepto, una relación, o una instancia, es decir, $e_{|O} \in C \cup R \cup I$. Usualmente, escribimos $e_{|O}$ como e cuando la ontología O está clara en el contexto.

Ejemplo de representación de una ontología según Ehrig

La ontología mostrada en la **Fig. 2** describe el dominio de automóviles como un comerciante de carros que ha modelado su inventario y pueden haber relaciones de clientes.

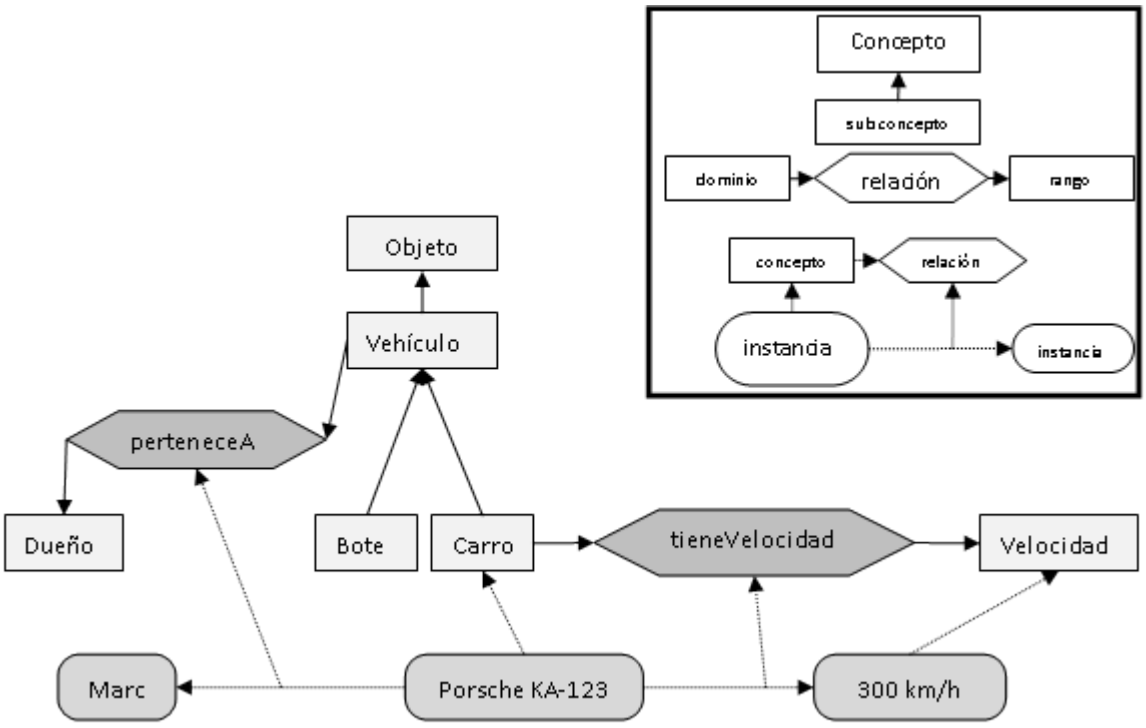

Fig. 2. Representación de una ontología según Ehrig

Los conceptos son mostrados como rectángulos, las relaciones como hexágonos, y las instancias como rectángulos redondeados. Las relaciones de subsunción son dibujadas como flechas sólidas. Una relación tiene una flecha de entrada de su dominio y una flecha de salida a su rango. Las instanciaciones de conceptos y relaciones son dibujadas como flechas de puntos. El ejemplo contiene seis conceptos objeto, vehículo, dueño, bote, carro, y velocidad, dos relaciones de pertenencia a alguien y velocidad, y tres instancias: Marc, Porsche KA-123, y 300 km/h. Hay una relación de subsunción entre objeto, vehículo, y bote y carro; un vehículo es un objeto, un bote es un vehículo, etc. Cada vehículo pertenece a un dueño y cada carro tiene una velocidad específica. En el nivel de instancia, el Porsche KA-123 pertenece a Marc y tiene la velocidad de 300 km/h. Además, el axioma es que cada carro necesita tener al menos un dueño definido. Los axiomas no están dibujados en el grafo, pero son presentados en el próximo párrafo.

Formalmente esta ontología es definida de acuerdo a $O = (S, A, KB, Lex)$. Para mantener la representación lo más corta posible, esta no contiene todas las construcciones del ejemplo anterior.

- esquema $S = (C, \leq\_C, R, \leq\_R) = (\{objeto, vehículo, dueño, \dots\},$
  $\{(vehículo, objeto), (bote, vehículo), \dots\}, \{perteneceA, tieneVelocidad\},$
  $\{perteneceA \rightarrow (vehículo, dueño), tieneVelocidad \rightarrow (carro, velocidad)\}, \{\})$
- axiomas $A = \{\forall x\ car(x) \Rightarrow \exists y\ perteneceA(x, y)\}$
- base de conocimiento $KB = (C, R, I, \iota_C, \iota_R) =$
  $(\{objeto, vehículo, dueño, \dots\}, \{perteneceA, tieneVelocidad\},$
  $\{Mark, Porsche\ KA\ 123, 300\ km/h\},$
  $\{dueño \rightarrow \{Marc\}, carro \rightarrow \{Porsche\ KA\ 123\}, velocidad \rightarrow \{300\ km/h\}\},$
  $\{perteneceA \rightarrow \{(Porsche\ KA\ 123, Marc)\},$
  $tieneVelocidad \rightarrow \{(Porsche\ KA\ 123{,}300\ km/h)\}$
- los lexicones sólo contienen los identificadores como entradas léxicas, es decir, $Lex =$
  $(\{\text{"objeto"}, \text{"vehículo"}, \dots\}, \dots, \{\text{"objeto"}, objeto),$
  $(\text{"vehículo"}, vehículo), \dots\}, \dots)$

2.1.2 Definición de ontologías por Euzenat y Shvaiko

La definición de ontologías de Euzenat y Shvaiko [2] está basada en los tipos de entidades que están presentes en los lenguajes de ontologías, con los cuales se expresan las ontologías. A continuación se describen cuáles son las entidades que se encuentran en un lenguaje de ontología. Para facilitar la comprensión de los ejemplos, presentamos la sintaxis en OWL [12-13], un lenguaje de ontologías recomendado por W3C[1].

Entidades de una ontología

Clases o **conceptos** son las entidades principales de una ontología. Son interpretadas como un conjunto de individuos en un conjunto. Son introducidas en OWL por la construcción owl:Class.

Individuos u **objetos** o **instancias** son interpretados como un individuo particular de un dominio. Son introducidas en OWL por la construcción owl:Thing.

Relaciones son la noción ideal de una relación independientemente al tipo que se aplique. Las relaciones son interpretadas como un subconjunto de producto del dominio. Son introducidas en OWL por owl:ObjectProperty u owl:DatatypeProperty.

Tipos de dato son una parte particular del dominio que especifica valores. Opuestamente a los individuos, los valores no tienen identidades.

Valores de dato son valores propiamente que un objeto puede tomar.

Especialización entre dos clases o dos propiedades es interpretada como la inclusión de las interpretaciones. La especialización es introducida por OWL por rdfs:subClassOf o rdfs:subPropertyOf.

Exclusión entre dos clases o dos propiedades es interpretada como la exclusión de sus interpretaciones, por ejemplo, cuando su intersección es vacía. La exclusión es introducida por OWL por owl:disjointWith.

[1] W3C (es un consorcio internacional donde las organizaciones miembros, personal a tiempo completo y el público en general, trabajan conjuntamente para desarrollar estándares Web.

Instanciación o **tipado** entre individuos y clases, instancias de propiedades y propiedades, valores y tipos de datos es interpretado como una membresía. La instanciación es interpretada en OWL con owl:type.

Definición 6 (Ontología). Una ontología es una tupla $o = \langle C, I, R, T, V, \leq, \perp, \in, = \rangle$, tal que:

- C es el conjunto de clases
- I es el conjunto de individuos o instancias
- R es el conjunto de relaciones
- T es el conjunto de tipos de datos
- V es el conjunto de valores (C, I, R, T, V siendo pares disjuntos)
- $\leq$ es una relación en $(C \times C) \cup (R \times R) \cup (T \times T)$ llamado especialización
- $\perp$ es una relación en $(C \times C) \cup (R \times R) \cup (T \times T)$ llamada exclusión
- $\in$ es una relación sobre $(I \times C) \cup (V \times T)$ llamada instanciación
- $=$ es una relación sobre $(I \times C) \times (I \cup V)$ llamada asignación

En la **Fig. 3** se muestra gráficamente una ontología de acuerdo con la definición anterior.

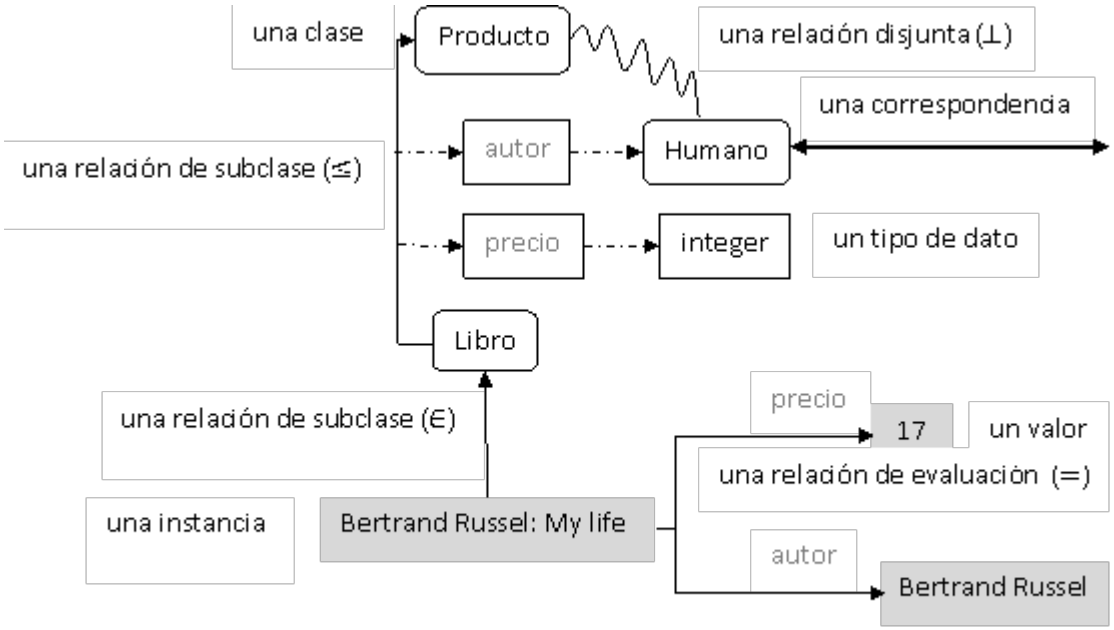

Fig. 3. Representación gráfica de una ontología según Euzenat y Shvaiko

2.1.3 Definición de ontologías según Hess et al.

Una definición menos elaborada pero con un sentido más práctico es la propuesta por Hess et al. [4]. Esta definición servirá para dar la definición de geo-ontologías propuesta por estos autores en la sección 3.1.

Definición 7 (Ontología): Una ontología puede ser definida como una 4-tupla $O = < C, P, I, A >$ donde:

- C es un conjunto de conceptos.
- P es un conjunto de propiedades.

- I es un conjunto de instancias.
- A es un conjunto de axiomas.

Un concepto $c \in C$ es cualquier fenómeno de interés a ser representado en la ontología y es definido por un término t que es usado como su nombre. El nombre de un concepto está dado por una función unaria $t(c)$.

Una propiedad $p \in P$ es una componente que es asociada a un concepto c con el objetivo de caracterizarlo, pero es definido fuera del ámbito de un concepto. Puede ser una propiedad de tipo de dato, lo que significa que su valor es un tipo de dato, como *string*, *integer*, *double*, etc., o una propiedad de tipo objeto. La propiedad tipo de dato puede ser vista como un atributo de base de datos, mientras que la propiedad tipo de objeto puede ser vista como una relación de base de datos.

Una instancia $i \in I$ es una ocurrencia de un concepto c, con un valor para cada propiedad p asociada al concepto y un único identificador.

Un axioma describe una relación jerárquica entre conceptos, o provee una asociación entre una propiedad y un concepto, o asocia una instancia con el concepto al cual pertenece, o define restricciones sobre las propiedades dentro del contexto de un concepto.

De esta última definición se puede criticar el tratamiento que le da a las definiciones. La componente relación de las dos primeras definiciones puede ser vista como la componente propiedad de la última definición de tipo objeto. En las dos primeras definiciones, las relaciones son consideradas como una componente elemental de las ontologías y no como un caso particular de la componente propiedad. También en la última definición se generaliza la relación como axioma a la relación jerárquica $\leq$ planteada en las dos primeras definiciones, considerándose como otra especialización de la componente axioma, lo cual no es así ya que la relación de taxonomía es una componente elemental en las ontologías.

2.2 Alineamiento de ontología

En esta sección se mostrará la definición de alineamiento de ontología según Ehrig [3] y se esclarecerán algunos términos relacionados con el alineamiento de ontologías.

2.2.1 Definición de alineamiento de ontologías

Alinear algo significa “traer en línea”. Esta es una breve definición que enfatiza que el alineamiento es una actividad en la cual después haberse realizada, los objetos involucrados están en mutua relación.

Ehrig [3] plantea que alinear una ontología con otra significa que, por cada entidad (concepto, relación, o instancia) en la primera ontología, se trata de buscar una entidad correspondiente, la cual pretenda tener el mismo significado en la segunda ontología.

Definición 8 (Alineamiento de ontologías). Una función de alineamiento, $align$, basada en el conjunto E de todas las entidades $e \in E$ y basada en el conjunto de posibles ontologías O es una función parcial

$$align: E \times O \times O \rightarrow E$$

Se denota $align_{O_1,O_2}(e)$ para $align(e, O_1, O_2)$. Se puede omitir O_1, O_2 cuando son evidentes a partir del contexto y escribir en su lugar $align(e)$. Una vez establecido un alineamiento

(parcial) $align$ entre dos ontologías O_1 y O_2, se dice que la entidad e está alineada con la entidad f cuando $align(e) = f$. Un par de entidades (e, f) que no están todavía alineadas y para las cuales el criterio apropiado de alineamiento todavía necesita ser probado es llamado un alineamiento candidato.

Aparte de los alineamientos de equivalencia uno-a-uno, una entidad, a menudo, tiene que ser alineada no solo con entidades equivalentes, sino basadas en otra relación (por ejemplo, subsunción). Podemos extender la definición anterior de alineamiento para introducir un conjunto de relaciones de alineamiento M. M incluye la identidad, subsunción, instanciación y ortogonalidad.

Definición 9 (Alineamiento general de ontología). Una función de alineamiento general de ontología $genalign$, basada en el vocabulario E de todos los términos $e \in E$, basada en el conjunto de posibles ontologías O, y basadas en posibles relaciones de alineamiento M, es una función parcial

$$genalign: E \times O \times O \rightarrow E \times M.$$

2.2.2 Representación de alineamientos de ontologías

Actualmente, no existe un consenso generalizado de un formato estándar para salvar alineamientos de ontologías. Ehrig [3] propone dos posibles formatos de representación que son aceptados y utilizados en la comunidad de alineamiento.

La primera posibilidad es adherirse a las construcciones existentes, por ejemplo en OWL. OWL provee axiomas de igualdad para conceptos, relaciones e instancias: owl:equivalentClass, owl:equivalentProperty, y owl:sameAs. También es posible expresar desigualdad a través de owl:differentFrom. La ventaja de esta representación es que el motor de inferencias de OWL interpretará automáticamente el alineamiento y razonará a través de varias ontologías. La desventaja es que la relación de equivalencia es muy estricta. Un valor de confianza no puede ser interpretado consecuentemente. Alineamientos complejos como los antes mencionados no son posibles.

La segunda posibilidad está basada en los trabajos de Euzenat [14]. La representación utiliza RDF/XML para formalizar alineamientos de ontologías. Después de la definición general de las ontologías involucradas, los alineamientos individuales son representados en celdas, con cada celda teniendo los atributos entidad 1, entidad 2, medida (la confidencia), y la relación (normalmente '='). Esta representación se corresponde con la definición de Alineamiento general de ontología (Definición 9). Debido a sus diferentes parámetros, puede ser utilizada en diferentes aplicaciones con alineamientos. Desafortunadamente, no es directamente un formato de ontología. Consecuentemente, se requiere una importación explícita para transformar el alineamiento en un formato adecuado para realizar inferencias. Para esta importación, se necesita definir cómo manejar los valores de confianza de un alineamiento.

Representaciones alternativas son MAFRA, *Semantic Bridging Ontology* (SBO), OWL contextualizado, el lenguaje de regla SWRL, el OMWG *Mapping Language* (OML), y SKOS. Una visión general de ello es encontrada en Euzenat et al. [15].

2.2.3 Términos relacionados

Las siguientes definiciones son tomadas de Klein [16], Ding et al. [17] y de Bruijn et al. [18] y comentadas por Ehrig [3]. Desafortunadamente, el uso de estos términos difiere considerablemente.

Combinación:
Dos o más ontologías diferentes son usadas en una tarea en la que su relación mutua es irrelevante. La relación combinación puede ser de cualquier tipo, no solo la identidad. Consecuentemente, no se tiene información de cómo es establecida la relación.
Integración:
Para la integración, una o más ontologías son reutilizadas para una nueva ontología. Los conceptos originales son tomados sin alterarlos, posiblemente son extendidos, pero su origen permanece claro, por ejemplo, a través de su nombre. Las ontologías son sencillamente integradas en lugar de ser mezcladas completamente. Esta aproximación es especialmente interesante, si las ontologías dadas difieren en sus dominios. A través de la integración, la nueva ontología puede abarcar un dominio mayor. El alineamiento de ontologías puede ser visto como un paso previo para detectar entre las ontologías involucradas, donde existen solapamientos y puedan ser enlazadas una con otra. Los métodos de integración más prominentes son unión e intersección [19], donde pueden ser tomadas todas las entidades de ambas ontologías o solamente aquellas que tienen correspondencias en ambas ontologías.
Emparejamiento, comparación (Matching):
Para el emparejamiento, se trata de buscar dos entidades correspondientes. Estas no tienen que ser necesariamente las mismas. Un emparejamiento puede ser, por ejemplo, en términos de un cerrojo y la llave que le sirva. Un grado de similitud sobre una dimensión especificada es suficiente, por ejemplo, el patrón de cerrojo/llave. La combinación permite varias relaciones al mismo tiempo, mientras que la comparación implica un tipo específico de relación. Un escenario típico para el emparejamiento es la composición de servicios web, donde la salida de un servicio tiene que corresponderse con la entrada correspondiente del próximo servicio. Cualquier esquema de emparejamiento o algoritmo de emparejamiento de ontologías puede ser utilizado para implementar el operador *Match*. El emparejamiento concuerda con la definición de alineamiento general, sin embargo, donde hay una relación fija entre las entidades alineadas, expresa un tipo de emparejamiento.
Asociación (Mapping):
La asociación de ontologías es utilizada para consultar diferentes ontologías. Una asociación de ontologías representa una función entre ontologías. Las ontologías originales no son cambiadas, pero los axiomas adicionales de asociaciones describen cómo expresar conceptos, relaciones o instancias, en términos de la segunda ontología. Ellas son almacenadas separadas de las ontologías. A menudo, las asociaciones pueden ser aplicadas en una dirección, por ejemplo, las instancias de un concepto en la ontología 1 pueden ser instancias de un concepto de la ontología 2, pero no viceversa. Éste es el caso, si las asociaciones sólo tienen restringida su expresividad y la relación teórica de asociación completa no puede ser encontrada en la representación actual. Un caso típico de uso de asociación es una consulta en una representación de ontología, la cual es después reescrita y manejada a otra ontología. Las respuestas son asociadas hacia atrás de nuevo. El alineamiento simplemente identifica la relación entre las ontologías, la asociación se concentra en la representación y ejecución de las relaciones para una cierta tarea.
Mediación:
La mediación de ontología es el proceso de alto nivel de reconciliar diferencias entre las ontologías heterogéneas para lograr interoperación entre los orígenes de datos y las aplicaciones usando estas ontologías. Esto incluye el descubrimiento y especificación de alineamientos de ontologías, así como el uso de estos alineamientos para una tarea, tal como la asociación para reescritura de consultas y transformación de instancias. Adicionalmente, el término mediación de ontología subsume la mezcla de ontología.

Mezcla:
Para la mezcla, una nueva ontología es creada a partir de dos o más ontologías. En este caso, la nueva ontología unificará y reemplazará las ontologías originales. Esto usualmente requerirá adaptaciones considerables y extensiones. Elementos individuales de las ontologías originales están presentes dentro de la nueva ontología, pero no pueden ser rastreadas hacia atrás. El alineamiento es un paso previo para detectar la superposición de entidades.
Transformación:
Cuando se transforman ontologías, su semántica cambia (posiblemente también cambie su representación) para hacerla adecuada para otros propósitos distintitos de los originales. Esta definición es tan general que es difícil relacionarla con el alineamiento.
Traducción:
Se define la traducción como una operación restringida a la traducción de datos [20], la cual puede incluir una sintaxis, por ejemplo, traducir una ontología de RDF(S) a OWL. El formato de representación de una ontología es cambiado mientras que la semántica se preserva. Como se está hablando acerca de alineamiento semántico, la traducción es un requerimiento fundamental cuando los formatos difieren, pero no nos ocupamos de traducción en sí.

2.3 Similitud entre ontologías

El objetivo del alineamiento de ontologías es encontrar qué entidad o expresión en una ontología se corresponde con otra en la segunda ontología. A continuación se presentan los métodos básicos que permiten evaluar esta correspondencia a nivel local, es decir, comparando un elemento con otro y no trabajando a escala global con las ontologías. Usualmente, estas relaciones son descubiertas a través de medidas de similitudes entre las entidades de las ontologías.

Similitudes, distancias y otras medidas

Existen varias vías para calcular la similitud entre dos ontologías. La manera más común es definir una medida de esta similitud. A continuación se presentan algunas características de estas medidas.

Definición 10 (Similitud). Una similitud $\sigma: o \times o \to \mathbb{R}$ es una función que, a partir de un par de entidades, calcula un número real que expresa la similitud entre dos objetos, tal que:

$$\forall x, y \in o, \sigma(x, y) \geq 0 \text{ (positividad)}$$
$$\forall x \in o, \forall y, z \in o, \sigma(x, x) \geq \sigma(y, z) \text{ (maximalidad)}$$
$$\forall x, y \in o, \sigma(x, y) = \sigma(y, x) \text{ (simetría)}$$

La disimilitud es la operación dual y se define como:

Definición 11 (Disimilitud). Dado un conjunto de entidades, una disimilitud $\delta: o \times o \to \mathbb{R}$ que, a partir de un par de entidades, calcula un número real que expresa la similitud entre dos objetos tal que:

$$\forall x, y \in o, \delta(x, y) \geq 0 \text{ (positividad)}$$
$$\forall x \in o, \forall y, z \in o, \delta(x, x) = 0 \text{ (minimilidad)}$$
$$\forall x, y \in o, \delta(x, y) = \delta(y, x) \text{ (simetría)}$$

Las definiciones anteriores están basadas en la simetría de las (di)similitudes. Definiciones de (di)similitudes no simétricas también han sido consideradas por algunos autores [21]. Los humanos tienden a no seguir la regla de la simetría cuando deciden la similitud entre dos objetos

[22]. Normalmente, la similitud entre el objeto 1 y el objeto 2 está valorada en el contexto del objeto 1, mientras que para la similitud entre el objeto 2 y el objeto 1 el contexto es del objeto 2.

Existen disimilitudes con más restricciones, tales como las distancias y las ultramétricas.

Definición 12 (Distancia). Una distancia o métrica $\delta: o \times o \to \mathbb{R}$ es una función de disimilitud que satisface la concreción (*definiteness*, en inglés) y la desigualdad triangular

$$\forall x, y \in o, \delta(x, y) = 0 \text{ si y solo si } x = y \text{ (concreción)}$$
$$\forall x, y, z \in o, \delta(x, y) + \delta(y, z) \geq \delta(x, z) \text{ (desigualdad triangular).}$$

Definición 13 (Ultramétrica). Dado un conjunto de entidades, una ultramétrica es una métrica tal que:

$$\forall x, y, z \in o, \delta(x, y) \leq \max\big(\delta(x, z), \delta(y, z)\big) \text{ (similitud ultramétrica).}$$

Usualmente, estas medidas son normalizadas, especialmente si la disimilitud de diferentes tipos de entidades debe ser comparada. Reducir cada valor a la misma escala, en proporción al tamaño del espacio considerado, es una manera común de normalizar.

Definición 14 ((Di)similitud normalizada). Una di(similitud), se dice que está normalizada si su valor está en el intervalo [0,1].

Podemos denotar a una (di)similitud normalizada σ (respectivamente, δ) como $\overline{\sigma}$ (respectivamente $\overline{\delta}$).

Se puede observar que a una similitud normalizada $\overline{\sigma}$, le corresponde una disimilitud normalizada $\overline{\delta}=1-\overline{\sigma}$ y viceversa.

2.3.1 Clasificaciones de las técnicas para el cálculo de medidas de similitud entre ontologías

2.3.1.1 Clasificación de las similitudes basados en el modelo de Ehrig

Para Ehrig [3], las similitudes entre dos ontologías están organizadas en dos dimensiones ortogonales. Puede ser vista como dimensiones horizontales y verticales como se muestra en la **Fig. 4**.

La dimensión horizontal incluye tres capas, una construida encima de otra.

Capa de datos:

En esta primera capa, se comparan entidades considerando los valores de los tipos de datos simples o complejos, como los enteros (int) y las cadenas (string). Para comparar los valores de datos, podemos utilizar una función de similitud genérica como la distancia de edición para cadenas y la distancia relativa entre enteros. Tipos de datos complejos son creados a partir de tipos de datos simples, por lo que requiere medidas complejas, que pueden ser medidas simples compiladas efectivamente.

Capa de ontología:

Se consideran las relaciones semánticas entre las entidades. De hecho, se puede separar esta capa otra vez en dependencia de la complejidad semántica, lo cual es derivado de las capas (*cake layer*) de Berners-Lee [23]. En el nivel más bajo, se trata a las ontologías solamente como un grafo con conceptos y relaciones. Estas redes semánticas fueron introducidas por Quillan [24]. Este nivel es mejorado por la lógica de descripción, como la semántica [25], por ejemplo, una taxonomía es creada sobre conceptos, en la cual un concepto hereda todas las relaciones de sus

superconceptos. Por ejemplo, si ciertas aristas son interpretadas como una jerarquía de subsunciones, es posible determinar la similitud taxonómica basada en el número de aristas *es-un* que separan dos conceptos. Además de los rasgos intencionales, también nos basamos en la dimensión extensional, es decir, se evalúan dos conceptos como que son lo mismo, si sus instancias son similares. Para restricciones, como en el lenguaje OWL, se usan diferentes heurísticas. Los niveles superiores de la capa de ontología también son interesantes para la similitud. Especialmente, si existen reglas de similitud entre entidades, estas entidades se estimarán como similares. Para este tipo de similitud, un experto humano es el que tiene que procesar relaciones de orden superior. Las funciones de similitud de la capa de ontología recurren a las funciones de similitud de la capa de datos.

Capa de contexto:

En esta capa, consideramos cómo son utilizadas las entidades de las ontologías en un contexto externo. Esto implica que usamos información externa a las ontologías. Consideramos el contexto como modelos locales que codifican el punto de vista subjetivo de una parte. Aunque existen muchos contextos en los cuales una ontología pueda considerarse (por ejemplo, el contexto en el que una ontología es desarrollada, en el que ha sido modificada), desde el punto de vista de determinar la similitud; el más importante es el contexto de la aplicación, por ejemplo, una entidad específica de una ontología ha sido usada en el contexto de una aplicación dada. Un ejemplo de esto es el portal de Amazon[2], en el cual, dando información acerca de qué personas compran cuáles libros, uno puede decidir si dos libros son similares o no. Consecuentemente, la similitud entre dos entidades de ontologías es fácilmente determinada comparando su uso en una aplicación basada en ontologías. Una explicación sencilla es que ontologías similares tienen un patrón similar de uso. El problema principal resta en cómo definir esos patrones de uso [26] para descubrir la similitud de la manera más eficiente. Para generalizar la descripción de esos patrones, reusamos el principio de similitud en términos de uso: entidades similares son usadas en contextos similares. Usamos ambas direcciones de la implicación en el descubrimiento de similitudes: si dos entidades son usadas en el mismo contexto (relacionado), estas entidades son similares y viceversa; si en dos contextos las mismas entidades (relacionadas) son usadas entonces esos contextos son similares.

La dimensión vertical representa el *dominio de conocimiento* específico que puede ser situado en cualquier capa de la dimensión horizontal. Aquí, la ventaja de un recurso externo específico de dominio, por ejemplo, Dublin Core[3] para el dominio bibliográfico, es considerado para estimar la medida de similitud entre entidades de ontologías.

[2] http://www.amazon.com/

[3] Dublin Core es un modelo de metadatos elaborado y auspiciado por la DCMI (*Dublin Core Metadata Initiative*), una organización dedicada a fomentar la adopción extensa de los estándares interoperables de los metadatos y a promover el desarrollo de los vocabularios especializados de metadatos para describir recursos para permitir sistemas más inteligentes del descubrimiento del recurso.

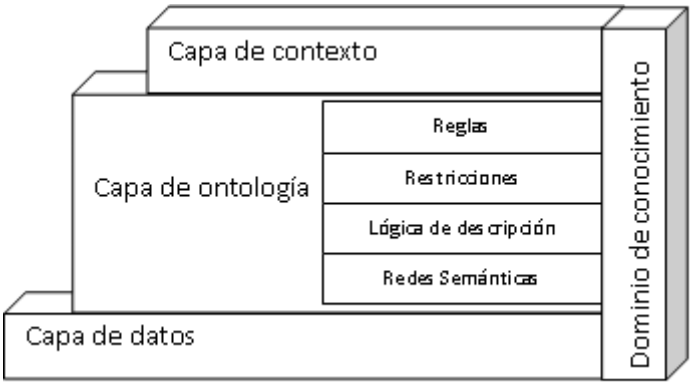

Fig. 4. Capas de similitud según la clasificación de Ehrig

2.3.1.2 Clasificación de las similitudes según Euzenat y Shvaiko

Estos autores clasifican las similitudes desde el punto de vista de técnicas de comparación. Para clasificar las técnicas de comparación elementales, Shvaiko y Euzenat [27] introdujeron dos clasificaciones basadas en las propiedades más salientes de las dimensiones de comparación. Estas dos clasificaciones son presentadas como dos árboles que comparten sus hojas. Las hojas representan las clases de las técnicas elementales de comparación y sus ejemplos concretos (ver **Fig. 5**). Dos clasificaciones son:

- Clasificación de la *Granularidad/Interpretación de la entrada* que está basada (i) en la granularidad del comparador, es decir, a nivel de los elementos o a nivel de estructuras, y (ii) como las técnicas generalmente interpretan la información de entrada.
- Clasificación por *tipo de entrada* que está basada en el tipo de entrada que es usada por las técnicas de comparación elementales.

La clasificación global se puede leer de forma descendente (enfocándose en cómo las técnicas interpretan la información de entrada) y de forma ascendente (concentrándose en los tipos de objetos manejados) hasta alcanzar la capa de técnicas básicas.

Los comparadores elementales están distinguidos por la capa *Granularidad/Interpretación de la entrada* de acuerdo al siguiente criterio de clasificación:

- *Nivel de elementos vs. nivel de estructuras*: Las técnicas de comparación que trabajan a nivel de elementos calculan correspondencias analizando las entidades o instancias de esas entidades aisladas, ignorando su relación con otras entidades o sus instancias. Las técnicas de nivel de estructuras calculan correspondencias analizando cómo las entidades o sus instancias aparecen juntas en una estructura. Este criterio, para los método basados en esquemas, fue introducido por Rahm y Berstein [28], mientras la separación *nivel de elementos* vs. *nivel de estructura*, para los métodos basados en instancias, fue por Kang y Naughton [29].
- *Sintáctico vs. externo vs. semántico*: La característica clave de las técnicas sintácticas es que ellas interpretan la entrada respetando su estructura siguiendo algún algoritmo indicado. Las técnicas externas explotan recursos auxiliares (externos) de un dominio y conocimiento común para interpretar la entrada de los datos. Estos recursos pueden ser

entrada de datos por humanos o algún tesauro que exprese la relación entre términos. Las técnicas semánticas usan alguna semántica formal, por ejemplo, semánticas modelo teórico (*model-theoretic semantics*) para interpretar la entrada de datos y justificar los resultados. En caso del sistema de comparación semántica, los algoritmos con resultados exactos son completos con respeto a la semántica, es decir, ellos garantizan un descubrimiento de todos los posibles alineamientos, mientras que los algoritmos aproximados tienden a ser incompletos.

La clasificación de la capa *tipo de entrada* tiene que ver con el tipo de entrada considerado por una técnica particular.

- El primer nivel está categorizado dependiendo del tipo de datos en los que trabajan los algoritmos: cadenas (*terminológicos*), estructura (*estructural*), modelos (*semánticas*) o instancias de datos (*extensional*). Los dos primeros son encontrados en la descripción de la ontología. El tercero requiere alguna interpretación semántica de la ontología y usualmente utiliza algún razonador semántico para deducir correspondencias. El último lo constituye la población de la ontología.
- El segundo nivel descompone estas categorías si es necesario: los métodos terminológicos pueden estar basados en cadenas (considerando los términos como una secuencia de caracteres) o basados en la interpretación de esos términos como objetos lingüísticos (lingüística). La categoría de métodos basados en la estructura es divida en dos tipos de métodos: aquellos que consideran la estructura interna de las entidades, por ejemplo: los atributos y sus tipos (internos), y los que consideran la relación de entidades con otras entidades (relacional).

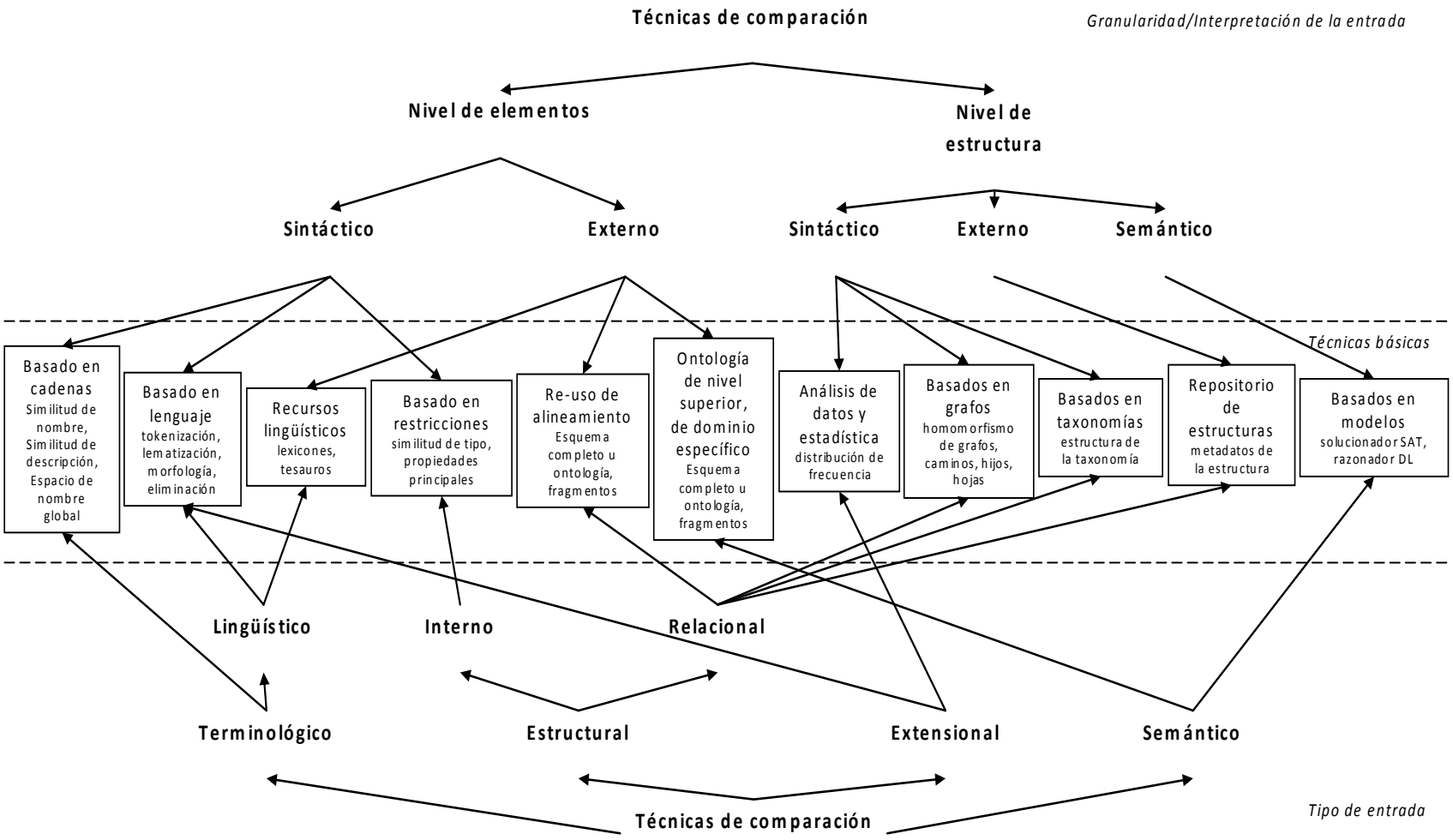

Fig. 5. Clasificación de los métodos de comparaciones principales. La clasificación superior está basada en la granularidad y la interpretación de la entrada; la clasificación inferior está basada en el tipo de entrada. La capa intermedia presenta las clases de las técnicas básicas

2.3.1.2.1 Técnicas en el nivel de elementos

Las técnicas en el nivel de elementos consideran las entidades de ontologías o sus instancias aisladamente de sus relaciones con otras entidades o instancias.

Técnicas basadas en cadenas

Las técnicas basadas en cadenas son usualmente usadas para comparar nombres y descripciones de nombres de las entidades de ontologías. Estas técnicas consideran las cadenas como secuencias de letras en un alfabeto. Ellas son típicamente basadas en la siguiente intuición: mientras más similares sean las cadenas, más parecen denotar el mismo concepto. Usualmente, funciones de distancias asocian un par de cadenas a un número real, donde un valor pequeño del número real indica una mayor similitud entre las cadenas. Algunos ejemplos de técnicas basadas en cadenas que son extensivamente usadas son los prefijos, sufijos, distancia de edición y distancia *n*-gram.

Técnicas basadas en el lenguaje

Las técnicas basadas en el lenguaje consideran los nombres como palabras en algún lenguaje natural, por ejemplo, Español. Están basadas en técnicas de procesamiento del lenguaje natural explotando propiedades morfológicas de las palabras de entrada.

Usualmente, son aplicadas a los nombres de las entidades antes de ejecutar técnicas basadas en cadenas o lexicones para mejorar resultados. Estás técnicas basadas en lenguaje pueden ser consideradas como una clase separada de técnicas de comparación, debido a que pueden ser extendidas, por ejemplo, en el cálculo de una distancia (comparando la cadenas resultantes o conjuntos de cadenas).

Técnicas basadas en restricciones

Las técnicas basadas en restricciones son algoritmos que tratan con restricciones internas siendo aplicadas a las definiciones de entidades como tipos, cardinalidad o multiplicidad de los atributos y llaves.

Recursos lingüísticos

Los recursos lingüísticos como los lexicones o dominios específicos como los tesauros, son usados para comparar palabras (en este caso, los nombres de entidades de ontologías son considerados como palabras del lenguaje natural) basados en relaciones lingüísticas entre ellos; por ejemplo, sinónimos, hipónimos.

Re-uso de alineamientos

Las técnicas de re-uso de alineamientos representan una vía alternativa de explotar recursos externos, las cuales graban alineamientos de ontologías previamente comparadas. Por ejemplo, cuando necesitamos comparar las ontologías o' y o'', dados los alineamientos entre o y o', y entre o y o'' disponible de un recurso externo. El re-uso de alineamiento está motivado por la intuición de que, muchas ontologías, al ser comparadas, son similares a ontologías previamente comparadas, especialmente si ellas describen el mismo dominio de aplicación. Estas técnicas son prometedoras cuando se trata de ontologías grandes consistentes en cientos de miles de entidades. En este caso, primero, los problemas de comparaciones grandes son descompuestos en pequeños subproblemas, generándose un conjunto de problemas de comparación de fragmentos de ontologías; entonces, el re-uso de resultados de comparación previos puede ser aplicado más efectivamente en el nivel de fragmentos de ontologías en lugar del nivel de ontología completa.

Ontologías de alto nivel y ontologías de dominio formal específico

Las ontologías de alto nivel pueden ser utilizadas como recurso externo de conocimiento común. Algunos ejemplos son: la ontología de alto nivel Cyc [30], *Suggested Upper Merged Ontology* (SUMO) [31] y *Descriptive Ontology for Linguistic and Cognitive Engineering* (DOLCE) [32]. La característica principal de estas ontologías es que son sistemas basados en la lógica y por lo tanto, las técnicas de comparación que utilizan la lógica se basan en la semántica por ejemplo, la ontología DOLCE apunta a proveer una especificación formal para el alto nivel de *WordNet*. Por lo tanto, los sistemas que explotan *WordNet* y su sistema de comparación pueden considerar utilizar DOLCE como una extensión semántica.

Las ontologías específicas de dominio pueden ser utilizadas como fuentes externas de conocimiento. Tales ontologías se concentran en un dominio particular y usan los términos en un sentido que es relevante sólo en este dominio y que no está relacionada a conceptos similares en otros dominios. Por ejemplo, en el dominio de anatomía, una ontología como *The Foundational Model of Anatomy* (FMA) puede ser usada como contexto para otras ontologías médicas, al ser comparadas. Esto puede ser usado para proveer la estructura faltante cuando se comparan recursos con estructuras pobres [33].

2.3.1.2.2 Técnicas del nivel de estructura

Contrario a las técnicas del nivel de elementos, las técnicas del nivel de estructuras consideran las entidades de ontologías o sus instancias para comparar sus relaciones con otras entidades o sus instancias.

Técnicas basadas en grafos

Las técnicas basadas en grafos son algoritmos de grafos que consideran las ontologías de entrada como grafos etiquetados. Las ontologías (incluyendo los esquemas de bases de datos y taxonomías) son vistas como estructuras de grafos etiquetados. Usualmente, la comparación de la similitud entre un par de nodos de dos ontologías está basada en el análisis de sus posiciones dentro del grafo. La intuición detrás de esto es que si dos nodos de dos ontologías son similares, sus vecinos deben ser similares de alguna forma.

Junto con las técnicas basadas puramente en grafos, existen otras técnicas con estructuras más específicas, como los árboles.

Técnicas basadas en taxonomías

Las técnicas basadas en taxonomías son también algoritmos de grafos que consideran sólo la relación de especialización. La intuición detrás de las técnicas es que los enlaces *es-un* (*is-a*) conectan términos que son similares (siendo interpretado como un subconjunto o superconjunto de cada uno), por lo tanto, sus vecinos deben ser similares de alguna forma.

Repositorio de estructuras

Los repositorios de estructuras almacenan las ontologías y sus fragmentos juntos con un par de medidas de similitud, por ejemplo, coeficientes en el rango [0,1]. A diferencia del re-uso de alineamientos, los repositorios de estructuras almacenan sólo la similitud entre ontologías, no los alineamientos. A continuación, a las ontologías o a sus fragmentos los llamaremos simplemente estructuras. Cuando nuevas estructuras deben ser comparadas, primero se revisa la similitud entre las estructuras que están disponibles en el repositorio. El objetivo es identificar estructuras que sean lo suficientemente similares para que valga la pena la comparación a un nivel más detallado, o re-usar alineamientos existentes, evitando la operación de comparación sobre estructuras disimilares. La determinación de la similitud entre estructuras debe ser

computacionalmente menos costosa que la comparación de ellas a todo detalle. El método de Rahm et al. [34] para comparar dos estructuras propone el uso de algún metadato que describa las estructuras, como el nombre de la estructura, el nombre de la raíz, el número de nodos, la longitud del camino máximo, etc. Estos indicadores son analizados y agregados en un solo coeficiente, el cual estima la similitud entre ellos. Por ejemplo, dos estructuras pueden ser encontradas apropiadas para comparase si ambas tienen la misma cantidad de nodos.

Técnicas basadas en modelos

Los algoritmos basados en modelos manejan la entrada de datos basada en su interpretación, por ejemplo, semántica del modelo teórico (*model-theoretic semantic*). La intuición es que si dos entidades son iguales, entonces ellas comparten la misma interpretación. Ejemplos son la satisfacibilidad proposicional y las técnicas de razonamiento de la lógica de descripción.

Técnicas de análisis de datos y estadística

Las técnicas de análisis de datos y estadística son las que toman ventaja de una muestra representativa de una población para encontrar regularidades y discrepancias. Agrupan elementos o calculan la distancia entre ellos. Entre las técnicas de análisis de datos se encuentran la clasificación basada en distancia, análisis de conceptos formales y análisis de correspondencias; entre las técnicas de análisis estadístico podemos citar la distribución de frecuencias.

2.3.1.3 Otras clasificaciones

Doan y Halevy [35] clasifican las técnicas de comparación en (i) basadas en reglas y (ii) basadas en aprendizaje. Típicamente, las técnicas basadas en reglas trabajan con información del nivel de esquema, como los nombres de las entidades, los tipos de datos, las estructuras. Ejemplo de reglas es que dos entidades coinciden si sus nombres son similares o si tienen el mismo número de entidades vecinas. Los métodos basados en aprendizaje usualmente trabajan con información a nivel de instancias, por ejemplo, comparando los formatos de los valores y la distribución de las instancias subyacentes a las entidades consideradas. Sin embargo, el aprendizaje también puede ser realizado a nivel de esquemas y a partir de comparaciones anteriores. Zanobini [36] clasifica los métodos de comparación en tres categorías:

Sintáctico: Esta categoría representa a los métodos que son puramente sintácticos. Algunos ejemplos de esos métodos incluyen técnicas basadas en cadenas, por ejemplo, la distancia de edición entre cadenas y técnicas de comparación de grafos, por ejemplo, distancia de edición de árboles.

Pragmático: Esta categoría representa a los métodos que confían en la comparación de instancias subyacentes a las entidades consideradas para calcular alineamientos. Algunos ejemplos de esos métodos incluyen clasificadores automáticos (clasificadores Bayesianos y análisis formal de conceptos).

Conceptual: Esta categoría representa a los métodos que trabajan con conceptos y comparan sus significados para calcular los alineamientos. Algunos ejemplos de estos métodos incluyen técnicas que explotan un tesauro externo, como *WordNet* para comparar el significado entre conceptos.

Giunchiglia y Shvaiko [37] clasifican los métodos de comparación en *sintácticos* y *semánticos*. En la dimensión proceso de comparación, esto corresponde a las categorías sintácticas y conceptual de Zanobini [36], respectivamente. Sin embargo, estas han sido restringidas por una segunda condición que trata con la dimensión de salida: las técnicas

sintácticas retornan un coeficiente en el rango [0,1], mientras que las técnicas semánticas retornan relaciones lógicas, como la equivalencia o la subsunción.

2.3.2 Medidas de similitud

En esta sección, detallaremos las distintas técnicas para el cálculo de las medidas de similitud existentes. Para ello nos basaremos en la clasificación realizada por Euzenat y Shvaiko [2] explicada en el epígrafe 2.3.1.2. Las técnicas utilizadas se clasificarán según el tipo de entrada que se puede observar en la **Fig. 5**.

2.3.2.1 Técnicas basadas en nombres

Algunos métodos terminológicos comparan cadenas. Pueden ser aplicados al nombre, etiqueta o comentarios de las entidades para encontrar cuáles son similares. Pueden ser usadas para comparar nombre de clases o URIs.

Representaremos con $\mathbb{S}$ el conjunto de cadenas, es decir, la secuencias de letras de cualquier longitud sobre un alfabeto $\mathbb{L}$: $\mathbb{S} = \mathbb{L}*$. La cadena vacía es denotada por ϵ, y $\forall s, t \in \mathbb{S}, s + t$ es la concatenación de las cadenas s y t, $|s|$ denota la longitud de la cadena (la cantidad de caracteres que contiene), $s[i]$ para $i \in [1, |s|]$ la letra en la posición i de s.

El problema principal en comparar entidades de ontologías por sus etiquetas ocurre debido a la existencia de sinónimos y homónimos:

Sinónimos: palabras diferentes usadas para nombrar la misma entidad.

Homónimos: palabras iguales usadas para nombrar entidades diferentes. El hecho de que una palabra pueda tener múltiples significados es también conocido como *polisemia*.

Existen dos categorías principales de métodos para comparar términos: considerando solamente las cadenas de caracteres, o usando algún conocimiento lingüístico para interpretar estas cadenas.

2.3.2.1.1 Métodos basados en cadenas

Los métodos basados en cadenas toman ventajas de la estructura de la cadena (como una secuencia de letras).

Existen varias vías para comparar cadenas dependiendo de cómo es vista la cadena: una secuencia exacta de letras, una secuencia errónea de letras, un conjunto de letras, un conjunto de palabras. Cohen et al. [38] comparan varias técnicas de comparación de cadenas.

Similitud de cadenas

La similitud de cadenas retorna 0 si las cadenas bajo consideración no son idénticas y 1 si son idénticas. Esta medida puede ser tomada como una medida de similitud.

Definición 15 (Igualdad de cadenas). La igualdad de cadenas es una similitud $\sigma: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que $\forall x, y \in \mathbb{S}, \sigma(x, x) = 1$ y si $x \neq y, \sigma(x, y) = 0$.

Esta medida no explica cuán diferentes son las cadenas. Una vía más inmediata de comparar dos cadenas es la distancia de Hamming, esta cuenta el número de posiciones en que difieren dos cadenas [39].

Definición 16 (Distancia de Hamming). La distancia de Hamming es una disimilitud $\delta: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que:

$$\delta(s,t) = \frac{\left(\sum_{i=1}^{\min(|s|,|t|)} s[i] \neq t[i]\right) + \big||s| - |t|\big|}{\max(|s|,|t|)}$$

Esta es una versión normalizada por la longitud de la cadena más larga.

Prueba de subcadenas

Diferentes variaciones pueden ser obtenidas a partir de la igualdad de cadenas, como considerar que dos cadenas son muy similares cuando una es subcadena de otra.

Definición 17 (Prueba de subcadena). La prueba de subcadena es una similitud $\sigma: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que $\forall x, y \in \mathbb{S}$, si $\exists p, s \in \mathbb{S}$, donde $x = p + y + s$ o $y = p + x + s$, entonces $\sigma(x,y) = 1$, en otro caso $\sigma(x,y) = 0$.

Esta medida se puede refinar en una similitud de subcadena que mide la proporción de la parte común entre dos cadenas.

Definición 18 (Similitud de subcadenas). La similitud de subcadenas es una similitud $\sigma: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que $\forall x, y \in \mathbb{S}$, y sea t la cadena común más larga de x y y:

$$\sigma(x,y) = \frac{2|t|}{|x| + |y|}$$

Esta definición puede ser usada para construir funciones basadas en el prefijo común más largo o en el sufijo común más largo.

Por ejemplo, si escribimos por error *aricle*, queriéndonos referir la palabra *article*, es deseable que la similitud entre estas dos palabras sea elevada, se analizará el comportamiento de la similitud entre dos palabras que se refieren al mismo concepto pero se escriben diferente, *article* y *paper*, y la similitud entre dos palabras que se tienen varios caracteres en común, pero con significados diferentes, *article* y *particle*.

Usando la similitud de subcadenas obtenemos que la similitud entre *article* y *aricle* sería 4/7 = 0.57. mientras que entre *article* y *paper* sería 1/7 = 0.14, y finalmente, entre *article* y *particle* sería 6/7 = 0.86.

Una similitud de prefijo o sufijo puede ser definida de este modelo a partir de pruebas de prefijos y sufijos, que probarían cuándo una cadena es prefija o sufija de otra. Estas medidas no son simétricas.

La similitud *n*-gram es también usada para comparar cadenas. Esta calcula el número de *n*-grams comunes, es decir, secuencias de *n* caracteres, entre ellas. Por ejemplos, *trigrams* para la cadena *article* son: art, rti, tic, icl, cle.

Definición 19 (Similitud *n*-gram). Sea $ngram(s,n)$ un conjunto de subcadenas de s de longitud n. La similitud *n*-gram es una similitud $\sigma: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que:

$$\sigma(s,t) = |ngram(s,n) \cap ngram(t,n)|$$

La versión normalizada de esta función

$$\bar{\sigma}(s,t) = \frac{|ngram(s,n) \cap ngram(t,n)|}{\min(|s|,|t|) - n + 1}$$

Esta función es muy eficiente cuando sólo algunos caracteres faltan.

Por ejemplo, la similitud entre *article* y *aricle* sería 2/4 = 0.5, mientras que entre *article* y *paper* sería 0, y finalmente, entre *article* y *particle* sería 5/6 = 0.83.

Distancia de edición

Intuitivamente, una distancia de edición entre dos objetos es el costo mínimo de operaciones que deben ser aplicadas a uno de los objetos para obtener el otro. Las distancias de edición fueron diseñadas para medir la similitud entre cadenas que puedan tener errores de escritura.

Definición 20 (Distancia de edición). Dado un conjunto Op de operaciones de cadenas ($op: \mathbb{S} \to \mathbb{S}$) y una función de costo $w: Op \to \mathbb{R}$, tal que para cualquier par de cadenas existe una secuencia de operaciones que transforman la primera cadena en la segunda (y viceversa), la distancia de edición es una disimilitud $\delta: \mathbb{S} \times \mathbb{S} \to [0,1]$ donde $\delta(s,t)$, es la función de costo de la secuencia menos costosa de operaciones que transforma s en t.

$$\delta(s,t) = \min_{(op_i)_I; op_n(\ldots op_1(s))=t} \left(\sum_{i \in I} w_{op_i} \right)$$

En la distancia de edición de cadenas, las operaciones que son consideradas usualmente incluyen la inserción de un caracter $ins(c,i)$, reemplazo de caracteres por otro $sub(c,c',i)$ y eliminación de un caracter $del(c,i)$. A cada operación se le asigna un costo y la distancia entre dos cadenas es la suma de los costos de cada operación en el conjunto de operaciones menos costoso.

La distancia de Levenshtein [40] es el mínimo número de inserciones, eliminaciones y sustituciones de caracteres requeridos para transformar una cadena en otra. Es la distancia de edición con costo 1.

La medida de Jaro ha sido definida para comparar nombres propios que pueden tener errores de escritura [41-42]. No está basada en la distancia de edición, pero sí en el número y proximidad de caracteres comunes entre dos cadenas.

Definición 21 (Medida de Jaro). La medida de Jaro es una medida no simétrica $\sigma: \mathbb{S} \times \mathbb{S} \to [0,1]$ tal que

$$\sigma(s,t) = \frac{1}{3} \times \left(\frac{|com(s,t)|}{|s|} + \frac{|com(t,s)|}{|t|} + \frac{|com(s,t)| - |transp(s,t)|}{|com(s,t)|} \right),$$

con

$s[i] \in com(s,t)$ si y sólo si $\exists j \in [i - (min(|s|,|t|)/2, i + (min(|s|,|t|)/2)]$

y $transp(s,t)$ son los elementos de $com(s,t)$ que ocurren en un orden diferente en s y t.

Por ejemplo, si comparamos *article* con *aricle*, *article* con *paper*, el número de letras comunes será 6, 7 y 1 respectivamente (debido a que en el último caso, la letra 'e' en *paper* está muy lejos de la de *article*). El número de letras comunes traspuestas será 0, 1 y 0 respectivamente. Como consecuencia, la similitud entre las cadenas son: 0.95, 0.90 y 0.45.

Esta medida fue mejorada favoreciendo la comparación de cadenas con prefijos comunes largos [43].

Definición 22 (Medida de Jaro-Winkler). La medida de Jaro-Winkler $\sigma: \mathbb{S} \times \mathbb{S} \to [0,1]$ es:

$$\sigma(s,t) = \sigma_{Jaro}(s,t) + P \times Q \times \frac{(1 - \sigma_{Jaro}(s,t))}{10}$$

tal que P es la longitud del prefijo común y Q es una constante.

En este caso, la similitud para un valor de $Q = 4$ para las tres cadenas (*aricle*, *paper* y *particle*) comparado con *article* son: 0.99, 0.98 y 0.45, respectivamente.

2.3.2.1.2 Métodos basados en lenguaje

Anteriormente, considerábamos a las cadenas como secuencia de caracteres. Cuando se considera el lenguaje, estas cadenas se transforman en texto. Los textos pueden ser segmentados en palabras.

Los métodos basados en lenguaje se basan en el uso de técnicas del procesamiento del lenguaje natural (*Natural Language Processing*, NLP) para extraer términos del texto. La comparación de estos términos y sus relaciones debe ayudar a calcular la similitud de las entidades de ontologías. Aunque estos métodos están basados en algún conocimiento lingüístico, se hace distinción a métodos que se basan solamente en algoritmos y a métodos que hacen uso externo de recursos externos como diccionarios.

Métodos intrínsecos: Normalización lingüística

La normalización lingüística apunta a reducir cada término a una forma estándar que pueda ser reconocida fácilmente. El trabajo de Maynard y Ananiadou [44] distingue tres tipos principales de variaciones de términos: morfológicos (variación de la forma y función de una palabra basada en la misma raíz), sintáctico (variación de la estructura gramatical de un término) y semántico (variación de un aspecto del término, usualmente usando una hiperonimia o hiponimia).

Una cadena lingüística ha sido desarrollada para obtener la forma normal de una cadena que denota términos. En ella usualmente se realizan las siguientes funciones:

Tokenización: Consiste en segmentar cadenas en secuencias de *tokens* por un tokenizador que reconozca símbolos de puntuación, letras mayúsculas, caracteres en blanco, dígitos, etc.

Lematización: Las cadenas de los *tokens* son analizadas morfológicamente para ser reducidas a una forma normalizada. El análisis morfológico hace esto posible encontrando las inflexiones y derivaciones de una raíz. Esto involucra la supresión del tiempo, género y número. A la recuperación de la raíz se le llama lematización. Actualmente, los sistemas usan sistemas técnicas aproximadas de lematización llamadas *stemming*, las cuales eliminan los sufijos de los términos. Por ejemplo, *reviewed* se trasforma en *review*.

Extracción de términos: Está relacionado con lo que se llama cuerpo lingüístico y requiere una gran cantidad de texto. Los extractores de términos identifican a los términos de la repetición de frases morfológicamente similares en el texto y el uso de patrones.

Eliminación de *stopwords*: Los *tokens* que son reconocidos como artículos, preposiciones, conjunciones, etc. son descartados, porque son considerados palabras sin significado para la comparación.

Una vez que estas técnicas han sido aplicadas, las entidades de ontologías son representadas con un conjunto de términos, no palabras, que pueden ser comparadas con las mismas técnicas presentadas anteriormente.

Métodos extrínsecos

Los métodos lingüísticos extrínsecos usan recursos externos, como los diccionarios y lexicones. Varios tipos de recursos lingüísticos pueden ser explotados para encontrar la similitud entre términos.

Lexicones. Un lexicón, o diccionario, es un conjunto de palabras con una definición en lenguaje natural de esas palabras.

Lexicones multilenguaje. Son lexicones en los cuales la definición es reemplazada por términos equivalentes en otro lenguaje.

Lexicones semántico-sintácticos. Los lexicones semántico-sintácticos y lexicones semánticos son recursos usados en los analizadores del lenguaje natural. Ellos usualmente no solamente contienen nombres, sino también categorías, por ejemplo, no animados, líquidos, y contiene los tipos de argumentos tomados por los verbos y adjetivos. Por ejemplo, *fluir* toma un líquido como sujeto y no tiene objeto. Son difíciles de crear y no son muy usados en la comparación de ontologías.

Tesauros. Son un tipo de lexicón a los que se les ha agregado algún tipo de relación, por ejemplo, hiperonimia. *WordNet* [45] es un tesauro que distingue el significado de la palabra agrupándolas en conjuntos de sinónimos (*synsets*).

Terminologías. Una terminología es un tesauro de términos que contienen frases en lugar de palabras. Usualmente son específicos del dominio.

Estos recursos pueden ser definidos para un lenguaje o ser específicos para un dominio.

Los recursos lingüísticos son introducidos para tratar con los sinónimos. Incrementando el sentido de interpretación de las palabras, ellos incrementan las posibilidades de encontrar términos que coincidan. Por otro lado, incrementan los homónimos y la posibilidad de encontrar correspondencias de términos inexistentes (falsos positivos). El tratamiento de este problema es conocido como desambiguación de palabras. La desambiguación de palabras trata de restringir los candidatos a partir del contexto.

Definición 23 (Recurso de sinónimos parcialmente ordenado). Un recurso de sinónimos parcialmente ordenado Σ sobre un conjunto de palabras W, es una tripla $\langle E, \leq, \lambda \rangle$ tal que $E \subseteq 2^W$ es un conjunto de *synsets*, $\leq$ es la relación de hiperonimia entre *synsets* y λ es una función de *synsets* a su definición (un texto que es considerado como un paquete de palabras en W). Para un término t, $\Sigma(t)$ denota el conjunto de *synsets* asociados a t.

Las siguientes medidas hacen uso de los sinónimos:

Definición 24 (Similitud de sinonimia). Dados dos términos s y t y un recurso de sinónimos Σ, la sinonimia es una similitud $\sigma: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que:

$$\sigma(s,t) = \begin{cases} 1, si\ \Sigma(s) \cap \Sigma(\mathrm{t}) \neq \emptyset \\ 0, en\ otro\ caso \end{cases}$$

Esto permite considerar que la similitud entre autor y escritor sea 1.

Definición 25 (Similitud de Cosinonimia). Dados dos términos s y t y un recurso de sinónimos Σ, la cosinonimia es una similitud $\sigma: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que:

$$\sigma(s,t) = \frac{|\Sigma(s) \cap \Sigma(t)|}{|\Sigma(s) \cup \Sigma(t)|}$$

Otras medidas de similitud se basan en la información, desde un punto de vista teórico. Se basan en que el concepto menos probable, es el que tiene más información. Así que el contenido de información de un concepto es inversamente proporcional a su probabilidad de ocurrencia. En la similitud propuesta en Resnik [46-47], cada *synset* (c) está asociado a una probabilidad de ocurrencia $(\pi(c))$ de una instancia de un concepto en un cuerpo particular. Usualmente, $\pi(c)$ es la suma de la ocurrencia de la palabra del *synset* divididas por el número total de conceptos. Esta probabilidad se obtiene de un cuerpo de estudio. Mientras más específico es el concepto, más baja es su probabilidad. La similitud semántica de Resnik entre dos términos es una función del *synset* más general común a ambos términos. Este considera el contenido máximo de información (entropía) del posible *synset*.

Definición 26 (Similitud semántica de Resnik). Dados dos términos s y t y un recurso de sinónimos parcialmente ordenado $\Sigma = \langle E, \leq, \lambda \rangle$ provisto con una medida de probabilidad π, la similitud semántica de Resnik es una similitud $\sigma: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que:

$$\sigma(s,t) = \max_{k;\exists c,c' \in E; s \in c \wedge t \in c' \wedge c \leq k \wedge c' \leq k} \left(-log\left(\pi(k)\right)\right)$$

Otras similitudes de teoría de la información dependen del crecimiento de la medida del contenido de información de los términos a sus hiperónimos comunes en lugar del contenido de información compartido. Este es el caso de la similitud de la teoría de información de Lin [48]. Este método especifica el grado de probabilidad de solapamiento entre dos *synsets*.

Definición 27 (Similitud del modelo teórico). Dados los términos s y t y un recurso de sinónimos parcialmente ordenado $\Sigma = \langle E, \leq, \lambda \rangle$ con una probabilidad π, la medida de la similitud de la teoría de información de Lin es una similitud $\sigma: \mathbb{S} \times \mathbb{S} \rightarrow [0,1]$ tal que:

$$\sigma(s,t) = \max_{k;\exists c,c' \in E; s \in c \wedge t \in c' \wedge c \leq k \wedge c' \leq k} \frac{2 \times log\left(\pi(k)\right)}{log\left(\pi(s)\right) + log\left(\pi(t)\right)}$$

Una vía final para comparar términos encontrados en cadenas a través de tesauros, como *WordNet*, es usar la descripción (*gloss*) dada a esos términos en *WordNet*. En este caso, alguna entrada de diccionario $s \in \Sigma$ es identificada por el conjunto de palabras correspondientes a $\lambda(s)$. Entonces, cualquier medida basada en cadenas puede servir para comparar cadenas [49].

Definición 28 (Solapamiento de descripción (*gloss overlap*)). Dado un recurso de sinónimos parcialmente ordenado $\Sigma = \langle E, \leq, \lambda \rangle$, el solapamiento de descripción entre dos cadenas s y t es definida por la similitud de Jaccard entre sus definiciones:

$$\sigma(s,t) = \frac{|\lambda(s) \cap \lambda(t)|}{|\lambda(s) \cup \lambda(t)|}$$

2.3.2.2 Métodos basados en la estructura

La estructura de las entidades que pueden ser encontradas en las ontologías pueden ser comparadas en lugar o en adición al comparar sus nombres o identificadores.

Esta comparación puede ser subdividida en una comparación de la estructura interna de una entidad, es decir, su nombre, sus propiedades, o en la comparación de una entidad con otras entidades con las cuales está relacionada. La primera es llamada estructura interna y la última

estructura relacional. La estructura interna es la definición de entidades sin referenciar a otras entidades; la estructura relacional es el conjunto de relaciones que una entidad tiene con otras entidades. La estructura interna es explotada en la comparación de esquemas, mientras que la estructura relacional es más importante en la comparación de ontologías formales.

2.3.2.2.1 Estructura interna

Los métodos basados en la estructura interna son algunas veces referidos en la literatura como métodos basados en restricciones [28]. Estos métodos están basados en la estructura interna de las entidades y usan tales criterios como el conjunto de sus propiedades, el rango de sus propiedades (atributos y relaciones), su cardinalidad o multiplicidad, y la transitividad o simetría de sus propiedades para calcular la similitud entre ellos.

Entidades con estructura interna comparables o propiedades con dominios y rangos similares entre dos ontologías pueden ser numerosas. Por esa razón, este tipo de métodos son comúnmente usados para crear agrupamientos de correspondencias en lugar de descubrir correspondencias entre entidades. Usualmente son combinadas con otras técnicas del nivel de elementos, como los métodos terminológicos, que son los responsables de reducir el número de correspondencias candidatas.

Comparación de tipo de datos

Existen reglas para las que un objeto de un tipo puede parecer como un objeto de otro tipo y existen reglas para las que un valor de un tipo puede ser convertido en la representación de la memoria a otro tipo (conocido como *casting* en los lenguajes de programación).

Idealmente, la proximidad entre tipos de datos debe ser maximal cuando estos son del mismo tipo, menor cuando los tipos son compatibles (por ejemplo, enteros y flotantes) y mucho menor cuando no son compatibles. La compatibilidad entre tipos de datos puede ser alcanzada usando una tabla de búsqueda (*lookup table*). Un ejemplo de dicha tabla se muestra en la **Tabla 1**.

Tabla 1. Fragmento de una tabla de compatibilidad de tipos

	char	*fixed*	*enumeration*	*int*	*number*	*string*
string	0.7	0.4	0.7	0.4	0.5	1.0
number	0.6	0.9	0.0	0.9	1.0	0.5

Comparación de dominio

Dependiendo de las entidades a ser consideradas, lo que se puede alcanzar mediante una propiedad puede ser diferente: en clases esto es el dominio mientras que en individuos estos son valores. Aún más, ellos pueden ser estructurados en conjuntos o en secuencias.

Valtchev [50] propuso un marco de trabajo (*framework*) en el cual los tipos o dominios de propiedades deben ser comparados sobre la base de su interpretación: conjunto de valores. La comparación de tipos está basada en su tamaño, en el que el tamaño de un tipo es la cardinalidad o multiplicidad del conjunto de valores que este define. La distancia entre dos dominios está dada por la diferencia entre su tamaño y el de su generalización común. Esta medida suele ser normalizada por el tamaño de la mayor distancia posible con un tipo de dato particular. A continuación se muestra una instancia de este tipo de medida.

Definición 29 (Distancia tamaño relativo (*Relative distance size*)). Dadas dos expresiones del dominio e y e' sobre un tipo de dato τ, la distancia tamaño relativo $\delta: 2^{\tau} \times 2^{\tau} \rightarrow [0,1]$, es como sigue:

$$\delta(e, e^{'}) = \frac{|gen_{\tau}(e \vee e^{'})| - |gen_{\tau}(e \wedge e^{'})|}{|\tau|},$$

tal que $gen_{\tau}(.)$ provee la generalización de un tipo de expresión y $\vee$ y $\wedge$ corresponde a la unión e intersección de los tipos.

Ejemplo: Considere la propiedad edad en una clase al ser comparada con la propiedad edad de otras tres clases (alumno, adolecente, adulto). La primera propiedad tiene un dominio de $[6,12]$, mientras que las otras tienen el dominio expresado por: $[7,14]$, $[14,22]$ y ≥ 10. Todas estas propiedades tienen el mismo tipo de dato entero. La generalización de estos cuatro dominios son los dominios en sí mismos, la unión con $[6,12]$, es $[6,14]$, $[6,22]$, $[6, +\infty[$, y la intersección es $[7,12]$, $\emptyset$ y $[10,12]$, respectivamente. Como consecuencia, la distancia será $3/|\tau|$, $17/|\tau|$ y $|\tau| - 3/|\tau|$, respectivamente también. Esto corresponde con la intuición de que la distancia entre dominios depende de la diferencia entre los valores que ellos cubran aisladamente o en común.

Usualmente, una generalización común depende del tipo:

Comparando multiplicidades y propiedades

Las propiedades pueden ser restringidas mediante las multiplicidades. Las multiplicidades son las cardinalidades aceptables del conjunto de valores de una propiedad. Similarmente a la compatibilidad de tipos de datos, la compatibilidad entre cardinalidades puede ser establecida mediante una tabla de búsqueda.

Las multiplicidades pueden ser expresadas como un intervalo de un conjunto de enteros positivos $[0, +\infty[$. Dos multiplicidades son compatibles si la intersección de los intervalos correspondientes no es vacía. Cualquier medida sobre el tipo de dato entero puede ser usada para calcular la similitud entre multiplicidades. En este caso, se presenta una distancia simple inspirada en la similitud de Jaccard.

Definición 30 (Similitud de multiplicidad). Dadas dos expresiones de multiplicidad $[b, e]$ y $[b', e']$, la similitud de multiplicidad es una similitud entre intervalos no negativos $\sigma: 2^{\tau} \times 2^{\tau} \rightarrow [0,1]$ tal que:

$$\sigma([b, e], [b^{'}, e']) = \begin{cases} 0 & , \text{si } b^{'} > e \text{ o } b > e' \\ \dfrac{\min(e, e^{'}) - \max(b, b')}{\max(e, e^{'}) - \min(b, b')} & , \text{en otro caso} \end{cases}$$

Por ejemplo, si tenemos que comparar la multiplicidad $[0,6]$ con $[2,8]$, $[8,12]$ y $[0, +\infty[$, la comparación producirá respectivamente 0.5, 0.0 y 6/MAXINT.

2.3.2.2.2 Estructura relacional

Una ontología puede ser considerada como un grafo cuyas aristas están etiquetadas por los nombres de la relación. Existen tres tipos de relaciones que han sido consideradas en las técnicas de estructura relacional: estructura de taxonomía, estructura de mereología y todas las relaciones involucradas.

Estructura de taxonomía

En una jerarquía de taxonomía, las relaciones entre entidades son relaciones de especialización, es decir, es el grafo construido con la relación subclase-de. La subentidad es una especialización de la superentidad y viceversa, la superentidad es una generalización de la subentidad. Una superentidad puede tener relaciones con una o más subentidades y similarmente, una subentidad puede tener relaciones con una o más superentidades.

Matemáticamente podemos denotar a una taxonomía como $H = \langle o, \leq \rangle$ donde o es la ontología y $\leq$ es la relación de especialización entre los elementos de la ontología o.

Existen varias medidas propuestas para comparar clases basadas en la estructura taxonómica. Las más comunes están basadas en el conteo del número de aristas en la taxonomía entre dos clases. La disimilitud estructural topológica sobre una jerarquía sigue la distancia de grafo (*graph distance*), es decir, el camino más corto en un grafo [51].

Una medida genérica posible para determinar la similitud semántica de conceptos C dentro de una ontología, en la jerarquía de conceptos, ha sido presentada por Rada et al. [52].

Definición 31 (Similitud de taxonomía). La similitud entre dos conceptos en una taxonomía $H = \langle o, \leq \rangle$ es una función de similitud $\sigma: o \times o \to \mathbb{R}$ tal que:

$$\sigma(c, c') = \begin{cases} e^{-\alpha l \cdot \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}}}, si\ c \neq c' \\ 1, \quad \text{en otro caso} \end{cases}$$

donde $\alpha \geq 0$ y $\beta \geq 0$ son parámetros que escalan la contribución del camino más corto de longitud l y profundidad h en la jerarquía de conceptos, respectivamente. La longitud del camino más corto es una métrica para medir la distancia conceptual de c_1 y c_2. La idea de usar la profundidad del concepto directo común que los subsume es que esos conceptos en las capas superiores de la jerarquía de conceptos son más generales y son semánticamente menos similares que los conceptos a un nivel inferior.

Definición 32 (Disimilitud estructural topológica en jerarquías). La disimilitud estructural topológica $\delta: o \times o \to \mathbb{R}$ es una disimilitud sobre una jerarquía $H = \langle o, \leq \rangle$ tal que:

$$\forall e, e' \in o, \delta(e, e') = \min_{c \in o} [\delta(e, c) + \delta(e', c)]$$

donde $\delta(e, c)$ es el número de aristas intermedias entre una, elemento e y otro, elemento c.

Esto corresponde con la distancia unidad de árbol (*unit tree distance)* de Barthélemy y Guénoche [53], es decir, con peso 1 en cada arista. Esta función puede ser normalizada por la longitud máxima de un camino entre dos clases en la taxonomía:

$$\bar{\delta}(e, e') = \frac{\delta(e, e')}{\max_{c, c' \in o} \delta(c, c')}$$

Los resultados dados por esta medida no son siempre semánticamente relevantes debido a que un camino largo en una jerarquía de clases puede ser resumido por uno corto alternativo.

Una distancia más elaborada es la similitud de Wu-Palmer [54]. Esta distancia toma en cuenta el hecho que dos clases cercanas a la raíz de una jerarquía están cercanas una a la otra en términos de aristas, pero pueden ser muy diferentes conceptualmente, mientras que dos clases separadas por un gran número de aristas deberían estar conceptualmente cercanas.

Definición 33 (Similitud de Wu-Palmer). La similitud de Wu-Palmer $\sigma: o \times o \rightarrow \mathbb{R}$ es una similitud sobre una jerarquía $H = \langle o, \leq \rangle$, tal que:

$$\sigma(c, c') = \frac{2 \times \delta(c \wedge c', \rho)}{\delta(c, c \wedge c') + \delta(c', c \wedge c') + 2 \times \delta(c \wedge c', \rho)}$$

donde ρ es la raíz de la jerarquía, $\delta(c, c')$ es el número de aristas intermedias entre una clase c y otra clase c' y $c \wedge c' = \{c'' \in o, c \leq c'' \wedge c' \leq c''\}$.

La similitud ascendente de cotópico aplica la similitud de Jaccard a los cotópicos [55].

Definición 34 (Similitud ascendente de cotópico). La similitud ascendente de cotópico $\sigma: o \times o \rightarrow \mathbb{R}$ es una similitud sobre una jerarquía $H = \langle o, \leq \rangle$, tal que:

$$\sigma(c, c') = \frac{|UC(c, H) \cap UC(c', H)|}{|UC(c, H) \cup UC(c, H)|}$$

donde $UC(c, H) = \{c' \in H; c \leq c'\}$ es el conjunto de superclases de c.

Estas medidas no pueden ser aplicadas directamente en el contexto de alineamientos de ontologías debido a que se supone que las ontologías no soportan la misma taxonomía H, pero pueden ser usadas en conjunción de un recurso de conocimiento común como *WordNet*.

Además de estas medidas globales que toman en cuenta toda la taxonomía para el cálculo de la similitud entre clases, existen medidas no globales. A continuación se muestran algunas de estas medidas.

Reglas de super o subclases: Están basados en reglas que capturan la intuición de que las clases son similares si sus super o subclases son similares. Por ejemplo, si las superclases son iguales, las clases en cuestión son similares. Si las subclases son iguales, las clases en cuestión también son similares [56-57]. Esta técnica tiene al menos dos inconvenientes: (*i*) cuando hay muchas super y subclases, entonces, pudieran ser asociadas a una misma clase, y (*ii*) la similitud entre las sub o superclases se basará a su vez en sus sub o superclases. Esto convierte este problema en otro problema de similitud global.

Comparación de camino definido: Se toman dos caminos con enlaces entre las clases definidas en la estructura jerárquica, se comparan los términos y sus posiciones a lo largo de estos caminos, y se identifican los términos similares. Está técnica fue introducida por Anchor-Prompt (ver en la sección 2.6.1 del estado del arte). Esta técnica es principalmente guiada por dos anclas de caminos y utiliza técnicas alternativas para elegir la mejor coincidencia.

Estructura de mereología

En una jerarquía de mereología, las relaciones entre entidades son relaciones parte-todo. La subentidad es una "parte" de la superentidad y viceversa, la superentidad puede estar compuesta de diferentes subentidades.

La similitud entre clases se calcula basándose en la idea de que objetos similares comparten partes similares. La dificultad de esta relación es que no es fácil encontrar propiedades que tengan una estructura de mereología.

Relaciones

Además de las dos relaciones previas, uno puede considerar el problema general del alineamiento de ontologías basado en sus relaciones. Las clases se relacionan también a través de las definiciones de sus propiedades. Estas propiedades son también aristas del grafo y si son encontradas similares, pueden ser usadas para buscar clases que sean similares. Sin embargo,

contrario a las estructuras de taxonomía y mereología, el grafo de relaciones puede contener circuitos.

Si la clase A está relacionada con la clase B por una relación R en una ontología, y si una clase A' se relaciona con una clase B' mediante una relación R' en otra ontología, y si sabemos que B y B' son similares, R y R' son similares, entonces podemos inferir que A y A' son similares también. Del mismo modo, si A es similar a A', R es similar a R', B puede ser similar a B'; o R ser similar a R', si conocemos de antemano que A y A' son similares y que B y B' son similares: la similitud entre relaciones en Mädche y Staab [58] es calculada de acuerdo a este principio. Por ejemplo, las clases "*Company*" y "*University*" serán consideradas similares porque tienen una relación similar "*hasEmployee*" con la clase "*Employee*" y la clase "*Professor*", las cuales son similares.

Esto puede ser extendido a un conjunto de clases y relaciones. Esto significa que si tenemos un conjunto de relaciones $r_1 \dots r_n$ en la primera ontología que son similares a otro conjunto de relaciones $r_1' \dots r_n'$ en la segunda ontología, es posible que dos clases que sean del dominio de la relación en esos dos conjuntos sean similares también.

Uno de los problemas de este método es definir cuán similares son dos relaciones. Este método está basado en la similitud de las relaciones para inferir la similitud de las clases del dominio o las clases del rango (imagen). Las relaciones entre clases en una ontología pueden ser consideradas como entidades en esa ontología, pueden ser organizadas en una relación jerárquica y como en las clases, el cálculo de la similitud entre las relaciones es un gran problema.

2.3.2.3 Técnicas extensionales

Las técnicas extensionales (o basadas en instancias) comparan la extensión de las clases, es decir, su conjunto de instancias en lugar de su interpretación.

Los métodos extensionales se pueden dividir en tres categorías: aquellos que se aplican a ontologías con un conjunto de instancias común, aquellos que proponen técnicas de identificación individual, y aquellos que no requieren de identificación, es decir, que trabajan con conjuntos de instancias heterogéneos.

2.3.2.3.1 Comparación de extensión común

La manera más fácil de comparar clases cuando comparten instancias es probar la intersección de su conjunto de instancias A y B y considerar que estas clases son similares cuando $A \cap B = A = B$, más general cuando $A \cap B = B$ o $A \cap B = A$. Sin embargo, la disimilitud sólo puede ser 1 cuando ninguno de estos casos es aplicado, por ejemplo, cuando las clases tienen algunas instancias en común pero no todas.

Una manera de mejorar el resultado de la comparación es usar la distancia de Hamming entre dos extensiones: esto se corresponde al tamaño de la diferencia simétrica normalizado por el tamaño de la unión.

Definición 35 (Distancia de Hamming). La distancia de Hamming entre dos conjuntos es una función de disimilitud $\delta: 2^E \times 2^E \to \mathbb{R}$ tal que $\forall x, y \subseteq E$:

$$\delta(x, y) = \frac{|x \cup y - x \cap y|}{|x \cup y|}$$

Usar esta distancia para comparar conjuntos es más robusto que usar la igualdad: tolera que algunos individuos estén erróneamente clasificados y producir una distancia pequeña.

También es posible calcular la similitud basado en la interpretación probabilística del conjunto de instancias. Este es el caso de la similitud de Jaccard [59].

Definición 36 (Similitud de Jaccard). Dados dos conjuntos A y B, sea $P(x)$ la probabilidad de que una instancia aleatoria esté en el conjunto X. La similitud de Jaccard se define como:

$$\sigma(A,B) = \frac{P(A \cap B)}{P(A \cup B)}$$

Esta similitud puede ser usada con dos clases de distintas ontologías que compartan el mismo conjunto de instancias.

Análisis formal de conceptos

Una de las herramientas de análisis formal de concepto (FCA, por sus siglas en inglés)) [60] es el cálculo del retículo de concepto (*concept lattice*, en inglés). La idea básica del análisis formal de concepto es la dualidad entre un conjunto de objetos (individuos) y sus propiedades: mientras más propiedades son restringidas, menos objetos satisfacen las restricciones. Así que un conjunto de objetos con propiedades puede estar organizado en un retículo de conceptos que cubra a estos objetos. Cada concepto puede ser identificado por sus propiedades (la intención) y cubre el individuo satisfaciendo estas propiedades (la extensión).

En el emparejamiento de ontología, las propiedades simplemente pueden ser las clases para las cuales se sabe que los individuos tienen un sitio y la técnica es independiente del origen de las entidades, es decir, si provienen de la misma ontología o no. De este conjunto de datos, el análisis formal de concepto calcula el retículo de conceptos (o retículo de Galois). Esto es realizado calculando la clausura de la conexión de Galois de instancias × propiedades. Esta operación comienza con el retículo completo del conjunto potencia de extensiones e intenciones, respectivamente) y mantiene sólo los nodos que son cerrados por la conexión, es decir, comenzando con un conjunto de propiedades, se determina el conjunto correspondiente de individuos, el cual se provee a sí mismo un conjunto de propiedades correspondientes; si este es el conjunto inicial, entonces está cerrado y es conservado; de otra manera, el nodo es descartado. El resultado es un retículo de conceptos.

A continuación se muestra un ejemplo del cálculo de un retículo a partir de la tabla de la **Fig. 6**. La tabla representa un pequeño conjunto de instancias y las clases a que pertenecen (de ambas ontologías). El lado derecho de la **Fig. 6** representa el retículo de conceptos correspondiente. A partir de este retículo, las siguientes correspondencias pueden ser extraídas.

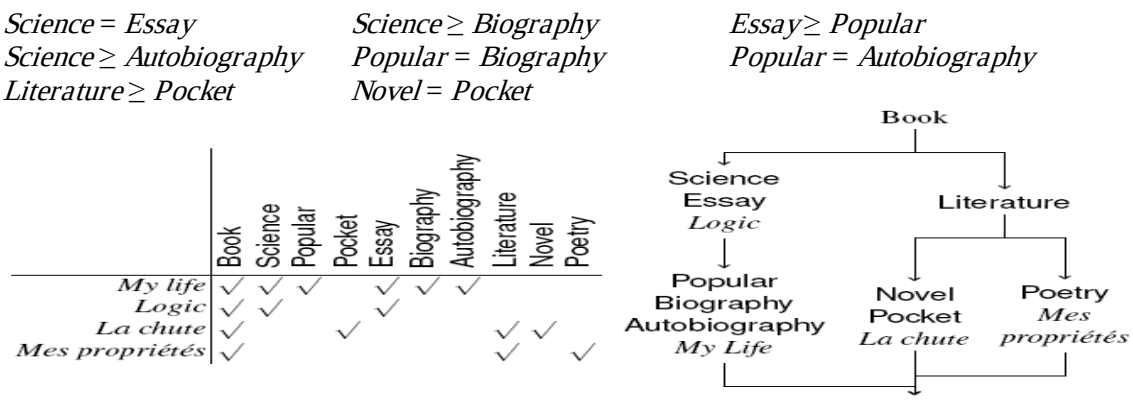

Fig. 6. Un contexto formal y el retículo de concepto correspondiente

2.3.2.3.2 Técnicas de identificación de instancias

Si no existe un conjunto común de instancias, cabe intentar identificar cuál instancia de un conjunto concuerda con otra instancia del otro conjunto. Este método es utilizable cuando uno sabe que las instancias son las mismas. Esto funciona, por ejemplo, cuando se integran dos bases de datos de recursos humanos de la misma compañía, pero no es aplicable a aquellas de compañías diferentes o bases de datos de eventos que no tienen relaciones.

Las llaves pueden ser internas para la base de datos, es decir, identificadores únicos generados (*surrogates*, en inglés), en cuyo caso no son muy útiles para identificación, o la identificación externa, en cuyo caso hay probabilidad alta de que esta llave esté presente en ambos conjuntos de datos (aun si no son llaves). En tal caso, si son utilizadas como llaves, podemos tener la seguridad de que excepcionalmente identifiquen únicamente a un individuo (como el ISBN de los libros).

Cuando las llaves no están disponibles, o son diferentes, otros acercamientos para determinar correspondencias de la propiedad usan las instancias de los datos para comparar valores de la propiedad. En bases de datos, esta técnica ha sido conocida como record *linkage* [61-62] o identificación de objetos [63]. Ellos apuntan a identificar múltiples representaciones del mismo objeto dentro de un conjunto de objetos. Se basan usualmente en técnicas de cadenas o en técnicas sobre la estructura interna.

Si los valores no son precisamente los mismos pero sus distribuciones pueden ser comparadas, cabe aplicarle técnicas globales. Este caso es cubierto en la siguiente sección.

2.3.2.3.3 Comparación de extensiones disjuntas

Cuando no es posible directamente inferir un conjunto de datos común para ambas ontologías, es más fácil de usar técnicas de aproximación para comparar extensiones de clase. Estos métodos pueden basarse en medidas estadísticas acerca de los rasgos de los miembros de la clase, en las similitudes calculadas entre las instancias de clases o basado en una comparación entre conjunto de entidades.

Métodos estadísticos

Los datos de instancia pueden usarse para calcular algunas estadísticas acerca de los valores de las propiedades encontradas en las instancias, como el máximo, mínimo, la media, la varianza, la existencia de valores nulos, la existencia de decimales, la escala, la precisión, agrupamientos, y números de segmentos. Esto permite la caracterización de los dominios de propiedades de la clase de datos. En la práctica, si se tratan con muestras estadísticamente representativas, estas medidas deberían ser las mismas para dos clases equivalentes de ontologías diferentes.

Considere dos ontologías con instancias. El análisis de las propiedades numéricas *tamaño* y *peso* en una ontología y *hauteur*[4] y *poids*[5] en la otra revelan que ellas tienen valores promedios diferentes pero el mismo coeficiente de variación, es decir, la desviación estándar dividida entre la media, la cual, a su vez, revela variabilidad comparable del tamaño y *hauteur* por un lado, y el peso y *poids* por otra parte. Esto es típicamente lo que ocurre cuando un valor es expresado en unidades diferentes. El cociente de los valores promedios del tamaño / *hauteur* es 2.54 y el de peso / *poids* es 28.35.

Estos valores han sido establecidos basados en toda la población. Pueden ser usados para comparar las características estadísticas de estas propiedades en las clases de las ontologías. Por ejemplo, el valor medio de la propiedad de tamaño para la clase *Pocket* difiere significativamente del que tiene la población global y, una vez entre 28.35, está muy cercano al de la clase *Livre_de_poche*[6] (que también difiere de toda la población, de la misma manera). Por lo tanto, estas dos clases podrían ser consideradas como similares.

Otros acercamientos, como el de Li y Clifton [64], proponen métodos que utilizan patrones de datos y distribuciones en lugar de los valores de datos y dominios. El resultado es una mejor tolerancia de fallos y un consumo de tiempo más bajo puesto que sólo una porción pequeña de valores de datos es necesario debido al empleo de técnicas de muestreo de datos. En general, aplicar métodos de estructura interna a instancias ofrece una caracterización más precisa del contenido real de los elementos del esquema, determinando así precisamente los tipos de datos basados, por ejemplo, sobre el rango de los valores descubiertos y los patrones de los caracteres.

Estos métodos tienen, sin embargo, un prerrequisito: trabajan mejor si las correspondencias entre propiedades son conocidas (de otra manera podrían comparar propiedades diferentes sobre la base de su dominio). Esto es un problema de comparación a ser solucionado.

Similitud basada en comparación de extensión (*Similarity-based extention comparison*)

Las técnicas de similitud no consideran que las clases compartan el mismo conjunto de instancias. En particular, los métodos anteriores siempre devuelven 0 cuando dos clases no comparten ninguna instancia, pasando por alto la distancia entre elementos de los conjuntos. En algunos casos es referible calcular la distancia entre esas clases. Para comparar el conjunto de instancias ellos usan una (di)similitud entre las instancias que pueden ser calculadas con cualquiera de los métodos presentados.

En análisis de datos, los métodos de agregación de enlace (*linkage aggregation*) permiten evaluar la distancia entre dos conjuntos cuyos objetos son sólo similares.

Definición 37 (Enlace simple). La medida de enlace simple entre dos conjuntos es una función de disimilitud $\Delta: 2^E \times 2^E \to \mathbb{R}$ tal que $\forall x, y \subseteq E, \Delta(x, y) = \min_{(e,e')\in x\times y} \delta(e, e')$.

Definición 38 (Enlace completo). La medida de enlace completo entre dos conjuntos es una función de disimilitud $\Delta: 2^E \times 2^E \to \mathbb{R}$ tal que $\forall x, y \subseteq E, \Delta(x, y) = \max_{(e,e')\in x\times y} \delta(e, e')$.

---

[4] Altura en francés
[5] Peso en francés
[6] Libro de cocina en francés

Definición 39 (Enlace promedio). La medida de enlace promedio entre dos conjuntos es una función de disimilitud $\Delta: 2^E \times 2^E \rightarrow \mathbb{R}$ tal que $\forall x, y \subseteq E, \Delta(x,y) = \frac{\sum_{(e,e')\in x\times y} \delta(e,e')}{|x|\times|y|}$.

Cada uno de estos métodos tienen sus beneficios propios, por ejemplo, maximizar la distancia más corta, minimizar la distancia más larga, minimizar la distancia promedio. Otro método de la misma familia es la distancia de Hausdorff que mide la distancia maximal de un conjunto al punto más cercano de otro conjunto [65].

Definición 40 (Distancia de Hausdorff). Dada una función de disimilitud $\delta: 2^E \times 2^E \rightarrow \mathbb{R}$, la distancia de Hausdorff entre dos conjuntos es una función de disimilitud $\Delta: 2^E \times 2^E \rightarrow \mathbb{R}$ tal que $\forall x, y \subseteq E$,

$$\Delta(x,y) = \max(\max_{e\in x} \min_{e'\in y} \delta(e,e'), \max_{e'\in y} \min_{e\in x} \delta(e,e')).$$

Comparación de correspondencias

El problema con las anteriores distancias, excepto el promedio, es que su valor es una función de la distancia entre una pareja de miembros del conjunto. El enlace promedio, por el contrario, tiene su función de valor de la distancia entre todas las comparaciones posibles.

Las comparaciones basadas en correspondencias [50] consideran que los elementos a ser comparados son aquellos que se corresponden unos a los otros, es decir, los más similares.

En esa medida, la distancia entre dos conjuntos es considerada como un valor a ser minimizado y su cálculo es un problema de optimización: el de la búsqueda de los elementos de ambos conjuntos que corresponde a cada uno de los demás. Esto corresponde a la solución del problema de comparación de grafos bipartitos.

Definición 41 (Similitud de correspondencia). Dada una función de similitud $\sigma: E \times E \rightarrow \mathbb{R}$, la similitud de correspondencia entre dos subconjuntos de E es una función de similitud $MSim: 2^E \times 2^E \rightarrow \mathbb{R}$ tal que $\forall x, y \subseteq E$,

$$MSim(x,y) = \frac{\max_{p\in Pairings\ (x,y)} \left(\sum_{\langle n,n'\rangle\in p} \sigma(n.n')\right)}{\max(|x|,|y|)}$$

siendo $Pairings(x,y)$ el conjunto de elementos de x asociados a y.

Esta similitud requiere un alineamiento de entidades.

2.3.2.4 Métodos semánticos

La característica principal de los métodos semánticos es que se usa el modelo de semántica teorética para justificar sus resultados, siendo métodos deductivos. Por supuesto, los métodos deductivos puros no funcionan muy bien por sí solos para una tarea esencialmente inductiva como la comparación de ontologías. Por lo tanto, necesitan una fase de preprocesamiento en la que se proveen 'anclas', es decir, se declaran las entidades que son equivalentes, por ejemplo, basadas en la identidad de sus nombres o en la entrada del usuario. La función de los métodos semánticos es ampliar estos alineamientos bases.

Los métodos para 'anclar' ontologías (sección 2.3.2.4.1) se basan en el uso de recursos formales existentes para iniciar una alineación que puede ser usada después por métodos deductivos (sección 2.3.2.4.2).

2.3.2.4.1 Técnicas basadas en ontologías externas

Cuando dos ontologías tienen que ser comparadas, a menudo carecen de una base común en la cual las comparaciones pueden basarse. En esta sección enfocamos la atención en el uso de ontologías formales intermedias para lograr este propósito. Estas ontologías intermedias pueden definir un contexto común o un conocimiento base [66] para la comparación de dos ontologías. La intuición es que una ontología base (*background ontology*) con una cobertura comprensiva del dominio de interés de las ontologías a ser comparadas ayudan en la desambiguación de los posibles múltiples significados de los términos.

La contextualización de ontologías se puede alcanzar típicamente comparando estas ontologías con una ontología común de alto nivel, que es usado como recurso externo de conocimiento común, por ejemplo *Cyc* [30], *Suggested Upper Merged Ontology* [31], o *Descriptive Ontology for Linguistic and Congnitive Engineering* (DOLCE) [32].

Un acercamiento propuesto en Aleksovski et al. [33] trabaja en dos pasos:

Anclaje (*Anchoring*) (también conocido como contextualización) es la comparación de las ontologías o' y o'' para la ontología de base o. Esto puede ser hecho usando algunos métodos disponibles presentados.

Relaciones Derivadas es la comparación (indirecta) de ontologías o' y o'' usando las correspondencias descubiertas durante el paso de anclaje. Debido a que los conceptos de ontologías o' y o'' son convertidos mediante las anclas como parte de la ontología de fondo o, la comprobación de que estos conceptos se relacionan, puede ser realizada usando un razonador (sección 2.3.2.4.2) en la ontología de base. Intuitivamente, combinando las relaciones de anclaje con las relaciones entre los conceptos de la ontología de referencia implica derivar las relaciones entre conceptos de o' y o''.

Una vez que han sido obtenidas estas alineaciones iniciales, estas pueden ser explotadas por técnicas deductivas.

2.3.2.4.2 Técnicas Deductivas

Estas técnicas utilizan la relación de subsunción entre dos entidades e y e' y se denotam como $e \sqsubseteq e'$.

Estas técnicas semánticas pueden servir para probar la satisfacibilidad de los alineamientos, en particular, para descartar alineaciones que conduzcan a una mezcla inconsistente de ambas ontologías.

Ejemplos de técnicas semánticas son la satisfacibilidad de proposiciones, las técnicas modales de satisfacibilidad, o las técnicas de la lógica de descripción.

Técnicas proposicionales

Un método para aplicar técnicas de satisfacibilidad de proposiciones (SAT) a la comparación de ontologías, incluye los siguientes pasos [37, 67-69].

1. Construir una teoría o dominio de conocimiento (axiomas) para las dos ontologías de entradas como una conjunción de axiomas disponibles. La teoría es construida usando comparadores discutidos en las secciones previas, por ejemplo, basados en *WordNet*, o incluyendo ontologías externas (sección 2.3.2.4.1).
2. Construir una fórmula de comparación para cada par de clases c y c' de las dos ontologías. El criterio para determinar si una relación se cumple entre dos clases es el hecho de que es

implicada por las premisas (la teoría). Por consiguiente, una consulta de comparación es creada como una fórmula de la siguiente forma:

$$Axiomas \rightarrow r(c, c')$$

para cada par de clases c y c' para las que se quiere comprobar la relación $r(within = , \sqsubseteq, \perp)$, donde $=$ denota la relación de equivalencia, $\sqsubseteq$ la relación de subsunción y $\perp$ es el concepto base definido en la lógica de descripción. c y c' son, algunas veces, llamados contextos.

3. Probar la validez de la fórmula, particularmente si es verdadera para todas las asignaciones verdaderas de todas las variables de la proposición que ocurran en ella. Una fórmula de la proposición es válida si y sólo si su negación es insatisfacible, lo cual es comprobado usando un solucionador SAT.

Los solucionadores SAT son procedimientos de decisión correctos y completos para la satisfacibilidad de la proposición, y por consiguiente, pueden servir para un chequeo exhaustivo de todas las correspondencias posibles. En algún sentido, estas técnicas calculan la clausura deductiva de algún alineamiento inicial [70].

Esta técnica sólo puede ser usada para la comparación de estructuras de tipo árboles, como las taxonomías, sin tomar en cuenta sus propiedades. *Modal* SAT puede ser usado, como propuso Shvaiko [71], para extender los métodos relacionados con las proposiciones SAT a predicados binarios.

Técnicas de lógica de descripción

La lógica descripción es una familia de formalismos terminológicos con semántica de la lógica formal y diseñada para representar conocimiento y para realizar razonamientos sobre él.

Los elementos básicos en la lógica de descripción son los conceptos primitivos, los roles primitivos, el concepto universal $\top$ y el concepto base $\perp$. El concepto universal $\top$ contendrá todas las posibles instancias de la ontología. El concepto base $\perp$ no tendrá instancias y es subclase de cada concepto en la ontología. Conceptos y roles complejos pueden ser construidos a partir de los conceptos y roles primitivos, respectivamente.

La lógica de descripción provee servicios de razonamientos, siendo la subsunción el más notable. Éste puede clasificar las relaciones entre categorías y derivar una ontología integrada que sea completa y consistente. La relación de subsunción en lógica de descripción es denotada mediante el operador $\sqsubseteq$.

Más detalles sobre la lógica de descripción pueden ser encontrados en el libro de Baader, et al. [25].

En las lógicas de descripción, las relaciones $=, \sqsubseteq, \perp$, pueden ser expresadas con relación a la subsunción. La prueba de subsunción puede usarse para establecer las relaciones entre clases en una manera puramente semántica. De hecho, mezclando primeramente dos ontologías y después probando cada par de conceptos y los roles para la subsunción es suficiente para comparar términos con la misma interpretación (o con un subconjunto de las interpretaciones de los demás) [72].

2.4 Proceso de alineamiento de ontologías según Ehring

Ehrig [3] describe un procedimiento general para alinear dos ontologías. La **Fig. 7** ilustra la entrada, la salida y los seis pasos principales del proceso general de alineamiento.

En la siguiente sección consideraremos la definición de ontología por Ehrig [3] y explicada en la sección 2.1.1, para estar acorde con el autor.

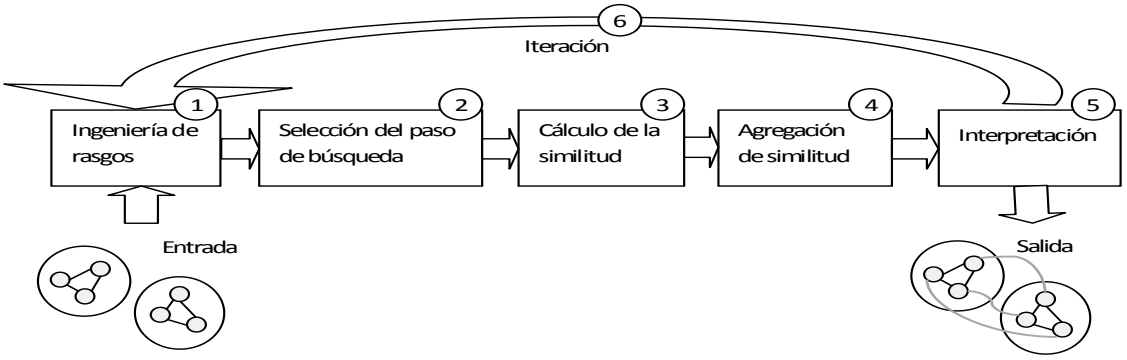

Fig. 7. Proceso de alineamiento según Ehrig

Para ilustrar el proceso de alineamiento de ontologías nos basamos en un ejemplo, mostrado en la **Fig. 8**, que consiste en dos ontologías simples que son alineadas. Las dos ontologías O_1 y O_2 describen el dominio de carros. Los alineamientos son representados mediante las líneas sombreadas que unen dos entidades. Se observa que cosa y objeto, carro y automóvil, y las dos velocidades son equivalentes. Las relaciones de tener una velocidad y propiedad se corresponden una a la otra debido a que se refieren a la velocidad. Las dos instancias Porsche KA-123 y Porsche de Marc son equivalentes.

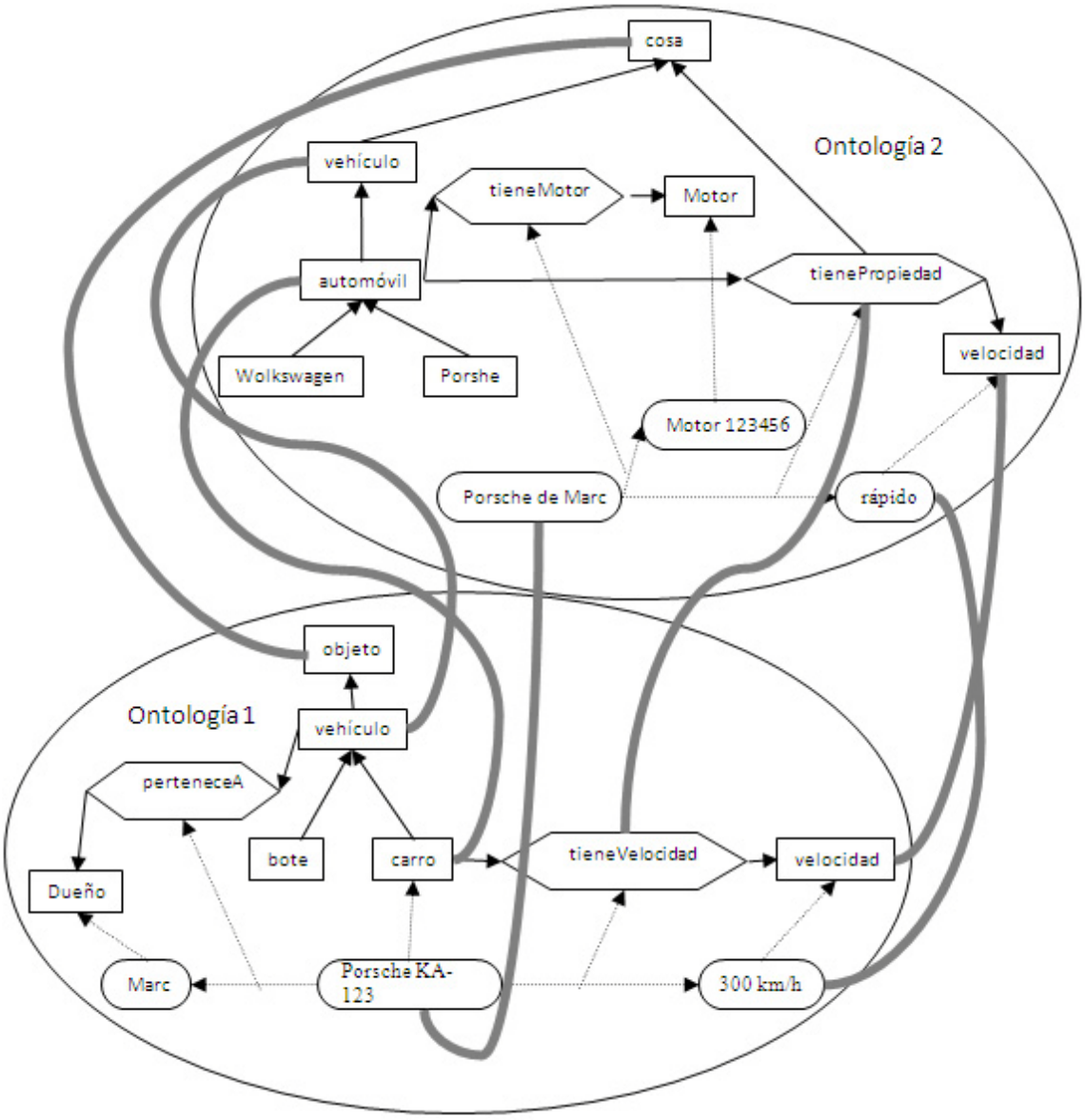

Fig. 8. Representación de alineamientos de ontologías tomando la definición de Ehrig

A continuación se expandirá el proceso general explicando sus pasos individuales en detalles.

2.4.1 Entrada

La entrada para el proceso son dos o más ontologías que necesitan ser alineadas. Adicionalmente, se pueden entrar alineamientos conocidos previamente (manual), que pueden ayudar en la búsqueda de los otros alineamientos.

Definición (Entrada): Dado un conjunto de ontologías O,

$$input: O \rightarrow \mathfrak{P}(O)$$

selecciona dos o más ontologías del mismo.

Si dos o más ontologías son tomadas, todas las ontologías son comparadas en pares. Alineamientos preconocidos pueden ser introducidos, entregándole al algoritmo de alineamiento un buen punto de comienzo, especialmente para los elementos del alineamiento estructural.

2.4.2 Ingeniería de rasgos (*Feature engineering*)

Pequeños fragmentos de la definición global de la ontología son seleccionados para describir una entidad específica.

Para comparar dos entidades a partir de dos ontologías diferentes O_1 y O_2, uno considera sus características, sus rasgos F.

Definición (Ingeniería de rasgos). A partir de dos ontologías, una lista de rasgos F es determinado a través de

$$feat: O \times O \rightarrow \mathfrak{P}(F).$$

Los rasgos seleccionados son específicos para un algoritmo de generación de alineamientos. En cualquier caso, los rasgos de una entidad ontológica (de conceptos C, relaciones R, e instancias I) necesitan ser extraídas a partir de definiciones de ontologías intencionales y extensionales. Ejemplos de rasgos son:

Identificadores: abarcan cadenas con formatos específicos, tal como las URIs (*unified source identifiers*) o etiquetas definidas en el lexicón *Lex* de la ontología. Las etiquetas son los rasgos más comunes usados cuando se consideran métodos relacionados.

Primitivas RDF-Schema: proveen un amplio rango de rasgos, por ejemplo, propiedades o las relaciones subclase/superclase definidas $(\leq_C, \leq_R)$. Esto también incluye relaciones subclase/subpropiedad inferidas.

Primitivas OWL: Extienden los rasgos de las primitivas RDF. Una entidad puede ser declarada siendo igual a otra entidad o a una unión de otras. Básicamente, cada primitiva OWL-DL puede ser usada como un rasgo para el alineamiento.

Rasgos derivados: restringen o extienden las primitivas simples (por ejemplo, clase de entidad más específica, o en inglés, *most-especific-class-of-instances*). Estas no son directamente modeladas en la ontología, pero son inferidas de ella.

Rasgos agregados: es necesario comparar más que una primitiva simple, por ejemplo, un hermano es cada instancia del concepto padre de la instancia original. Tampoco es modelado directamente en la ontología.

Axiomas complejos: pueden ser la base para identificar alineamientos. Si en una ontología sabemos que carros rápidos tienen un motor rápido y pertenece al grupo de vehículos callejeros, y en la otra, sabemos que hay un carro deportivo, el cual corre en caminos y tiene un motor rápido, inferimos que un carro deportivo es un carro rápido. Para lograr estas comparaciones, las variables individuales (típicamente entidades) y operadores de los axiomas lógicos en ambas ontologías necesitan ser comparadas. El manejo de axiomas para el alineamiento han sido tratados por Fürst y Trichet [73].

Usualmente, el alineamiento de ontologías tiene que ser ejecutado para una aplicación específica de un dominio que es expresado dentro de la definición de la ontología O. Para estos

escenarios, los *rasgos específicos del dominio* proveen valores excedentes al proceso de alineamiento. Volviendo al ejemplo, la relación velocidad no es un rasgo general de la ontología, pero un rasgo que es definido en el dominio de ontología del automóvil. De esa manera, será importante para representaciones correctas de alineamientos de vehículos concretos. Como uno puede imaginar en este contexto, el par ontología-específica rasgo-valor (velocidad, rápido) es más exacto que un par genérico (alguna relación, algún rango).

Rasgos externos: son un tipo de información que no ha sido directamente codificada en la ontología, tal como una colección de palabras (*bag-of-words*, en inglés) de un documento describiendo una instancia. De hecho, si las ontologías son débiles en expresividad, examinar los rasgos externos puede ser la única manera de encontrar el resultado del alineamiento.

En el ejemplo de la **Fig. 8**, el concepto carro en la ontología 1 está caracterizado por su etiqueta "carro", el subconcepto vehículo que está enlazado a él, su concepto hermano bote, y la relación tiene velocidad. carro también es descrito por sus instancias, aquí solo Porsche KA-123. La relación tiene velocidad, en cambio, es descrita a través del dominio carro y el rango velocidad, una instancia es Porsche KA-123, que es caracterizado a través de la propiedad instanciada pertenece a Marc y tiene una velocidad de 300 km/h.

2.4.3 Selección del paso de búsqueda (*Search Step Selection*)

Antes de empezar la comparación de dos entidades, es necesario escoger qué par de entidad (e, f) de las ontologías vamos a considerar.

El alineamiento de ontologías tiene lugar en un espacio de búsqueda de alineamientos candidatos. Este paso puede escoger, para calcular la similitud entre ciertos candidatos, un par de conceptos $\{(e, f) | e \in O_1, f \in O_2\}$ e ignorar otros (por ejemplo, sólo comparar o1:carro con o2:automóvil y no con o2:tieneMotor.

Definición (Selección del paso de búsqueda). Dadas dos ontologías para alinear, definimos

$$sele: O \times O \rightarrow \mathfrak{P}(E \times E)$$

resultando en un conjunto de pares de entidades donde E son las entidades previamente definidas.

Los métodos más comunes para alineamientos candidatos son:

- Comparar todas las entidades de la primera ontología O_1 con todas las entidades de la segunda ontología O_2: $(e, f) \in E_1 \times E_2$;
- Comparar sólo aquellas entidades del mismo tipo (concepto, relaciones e instancias)

$$(e, f) \in (C_1 \times C_2) \cup (R_1 \times R_2) \cup (I_1 \times I_2).$$

Estas estrategias son conocidas como *agendas completas (complete agendas)*.

2.4.4 Cálculo de la similitud (*Similarity Computation*)

Las similitudes representan evidencias de que dos entidades son lo mismo, entonces pueden ser alineadas. El cálculo de la similitud entre una entidad e y una entidad f es realizado usando un amplio rango de funciones de similitud. Cada función de similitud está compuesta del rasgo introducido (F) existente en ambas ontologías y la correspondiente medida de similitud.

Definición (Cálculo de similitud). Para cada par de entidades y un rasgo correspondiente, las similitudes son definidas como

$$sim: E \times E \times F \rightarrow [0,1]^k$$

Consideremos el ejemplo de la **Fig. 8**. Examinaremos los alineamientos candidatos (o1:carro, o2:automóvil). Para cada rasgo, se calcula una similitud.

Por ejemplo, si seleccionamos como rasgo las etiquetas de dichas clases, las etiquetas de las entidades padres de dichas clases, y las instancias de esas clases, entonces por cada uno de esos rasgos calculamos respectivamente las similitudes:

sim_{label} (o1:carro,o2:automóvil),
$sim_{superconcept}$ (o1:carro,o2:automóvil),
$sim_{instance}$ (o1:carro,o2:automóvil).

2.4.5 Agregación de similitud (*Similarity Aggregation*)

En general, hay muchos valores de similitud para un par candidato de entidades, por ejemplo, uno para la similitud de sus etiquetas y uno para la similitud de su relación entre otras entidades. Estos distintos valores de similitud para un par candidato deben ser agregados en un solo valor de similitud.

De acuerdo a lo planteado anteriormente, alineamos ontologías mediante la comparación de similitudes. Asumimos que una combinación de los rasgos presentados y las medidas de similitud conduce a un mejor alineamiento resultante que solamente usando uno en un momento determinado. No todos los valores de similitud tienen que ser utilizados para cada agregación, especialmente cuando algunos tienen una correlación alta.

Definición (Agregación de similitud). Valores múltiples de similitud son agregados a un valor:

$$agg: [0,1]^k \rightarrow [0,1]$$

Incluso, aunque existen varios métodos para alineamientos, no existe un artículo enfocado en la combinación e integración de estos métodos para ontologías. Do y Rahm [74] manejan este problema para estructuras de bases de datos, pero dejan la decisión de combinación al usuario al final.

Generalmente, la agregación de similitud puede ser expresada a través de:

$$sim_{agg}(e,f) = agg(sim_1(e,f), \ldots, sim_k(e,f))$$

con (e,f) un alineamiento candidatos y agg una función sobre las medidas de similitudes individuales sim_1 hasta sim_k. Usualmente, esta función conlleva a una ecuación más simple

$$sim_{agg}(e,f) = \frac{\sum_{k=1\ldots n} w_k \cdot adj_k(sim_k(e,f))}{\sum_{k=1\ldots n} w_k}$$

siendo w_k el peso para cada medida de similitud individual y adj_k una función de ajuste para transformar el valor original (adj: $[0,1] \rightarrow [0,1]$). A continuación se expone una explicación del uso de estas variables.

Promedio:

Todos los pesos individuales son fijados a 1, la función de ajuste adj_k es ajustada a la función identidad id, el valor no cambia. El resultado es el promedio sobre todas las similitudes individuales.

$$w_k = 1, adj_k(x) = id(x)$$

Suma linear:

La función de ajuste es de nuevo ajustada a la función identidad.

$$adj_k(x) = id(x)$$

Para esta agregación, los pesos w_k tienen que ser determinados. Los pesos son asignados manualmente o aprendidos, por ejemplo, utilizando un algoritmo de aprendizaje de máquina en un conjunto de entrenamiento. Berkovsky et al. [75] han investigado el efecto de los pesos en los diferentes resultandos de los alineamientos. En este método, sólo se están buscando valores de similitud que apoyen la pretensión de que dos entidades son iguales. La falta de similitud no es necesariamente tratada como una evidencia negativa.

Suma linear con evidencia negativa:

A menudo es más claro determinar que dos entidades no deberían estar alineadas que la contraparte positiva si el valor de agregación es negativo. Un valor negativo para w_k puede ser aplicado, si la similitud individual no evidencia una alineamiento, incluso señala que dos entidades no deberían ser alineadas. Los ejemplos típicos de tal caso son superconceptos de la primera entidad teniendo una similitud alta con subconceptos de la segunda entidad.

Función sigmoidal:

Un método más sofisticado enfatiza altos valores de similitud y le quita importancia a valores bajos de similitud individual. Es este caso, una función prometedora es la función sigmoidal sig, la cual tiene que ser desplazada para adecuarse al rango de entrada [0,1].

$$adj_k(x) = sig_k(x - 0.5)$$

con $sig_k(x) = \frac{1}{1+e^{\alpha_k x}}$ y α_k el parámetro de la pendiente.

El comportamiento del uso de la función sigmoidal es explicado mejor a través de un ejemplo. Cuando se comparan dos etiquetas, la posibilidad de tener la misma entidad, si sólo una o dos letras difieren, es bastante alta; esto puede ocurrir debido a error de escritura o distintas formas gramaticales. Si sólo tres o cuatro letras coinciden, allí no hay información del todo en esa similitud; una agregación de varios de esos valores bajos no deben conllevar a un alineamiento. Valores altos, por lo tanto, son incrementados y los valores bajos reducidos. Los parámetros de la función sigmoidal pueden ser estimados como una extensión de los métodos de similitud, como ellos tienen que ser ajustados de acuerdo al método sim_k al que son aplicados. Después, los valores modificados son sumados con los pesos específicos w_k adjuntados.

Para el ejemplo mostrado, usamos una agregación lineal simple. Asumimos diez similitudes individuales cuando comparamos dos conceptos de los cuales sólo las primeras se muestran en la siguiente fórmula:

$$sim_{agg}\,(\text{o1:carro,o2:automóvil}) = (1.0 \cdot sim_{label}\,(\text{o1:carro,o2:automóvil}) + 1.0 \cdot sim_{superconcept}\,(\text{o1:carro,o2:automóvil}) + 1.0 sim_{instance}\,(\text{o1:carro,o2:automóvil}) + \cdots)/10 = 0.5$$

2.4.6 Interpretación

Para los valores de similitud agregados, necesitamos deducir si existe o no un alineamiento.

Definición (Interpretación). Un valor de similitud agregado puede conducir a un alineamiento.

$$inter\colon [0,1] \rightarrow \{alignment\}$$

donde $alignment$ en esta definición es una constante.

Asignamos el alineamiento basándonos en un umbral θ aplicado a los valores de similitudes agregados. Cada entidad puede participar en uno o en múltiples alineamientos. Do y Rahm [74] presentan diferentes métodos para calcular un umbral. Cada valor de similitud por encima del umbral indica un alineamiento; cada valor por debajo del umbral es desechado.

Valor de similitud constante:

Para este método, una constante fija representa el umbral.

$$\theta = const$$

El umbral constante parece razonable mientras estemos recolectando evidencia para los alineamientos. Si es extraída poca evidencia de las ontologías, simplemente no es posible confiar en el alineamiento presentado. En esta línea se presentan resultados en Ehrig y Sure [57]. Sin embargo, es difícil determinar su valor. Una posibilidad es un promedio que maximice la calidad en varias corridas de pruebas. Alternativamente, tiene sentido dejar a expertos que determinen el valor.

Método Delta:

Para este método, el umbral para la similitud es definido tomando el mayor valor de similitud de todos y se le sustrae un valor fijo.

$$\theta = \max_{e \in O_1 f \in O_2} \left(sim_{agg}\,(e,f)\right) - const$$

Porciento N:

Este método está relacionado con el anterior. Aquí se escoge el mayor valor de similitud encontrado y se le sustrae un porcentaje p fijo.

$$\theta = \max_{e \in O_1 f \in O_2} \left(sim_{agg}\,(e,f)\right)(1-p)$$

Los últimos dos métodos son motivados a través de la idea de que la similitud es también dependiente del tipo de ontología o del dominio. La máxima similitud calculada es un indicador

para esto, y es retroalimentada al algoritmo. Desafortunadamente, si dos ontologías no tienen solapamientos, la función *máximo* no retorna resultados razonables.

Lo siguiente es decidir en cuántos alineamientos una entidad puede estar involucrada.

Enlace sencillo de alineamiento (One Alignment Link):

El objetivo de ese método es lograr un alineamiento sencillo entre dos ontologías a partir del mejor valor de similitud. Como aquí sólo hay una *buena* correspondencia (*match*), cualquier otra correspondencia es un error potencial que debiera ser desechado. Prácticamente se eliminan las entradas de las tablas que incluyan entidades ya alineadas. Del conjunto de alineamientos candidatos ($U \times V$) una estrategia voraz (*greedy*) determina primero el par con mayor similitud agregada (argmax). Este par es almacenado como un alineamiento ($aling(e,f)$). Cualquier otro alineamiento que involucre una de las entidades recién alineadas (e o f) es eliminada de los candidatos restantes. El proceso es repetido mientras que existan pares de entidades no alineados

$$align(e,f) \leftarrow (sim_{agg}(e,f) > \theta) \wedge ((e,f) = \underset{(g,h)\in(U\times V)}{\operatorname{argmax}} sim_{agg}(g,h))$$

con U y V conteniendo solamente entidades no alineadas.

Enlaces múltiples de alineamiento (Multiple Alignment Links):

A menudo, tiene sentido mantener múltiples alineamientos. En este caso, la interpretación es expresada más fácilmente a través de la siguiente fórmula:

$$align(e,f) \leftarrow sim(e,f) > \theta.$$

En el ejemplo, dos entidades, carro y automóvil, resultaron con una similitud 0.5. Aplicando un umbral fijo de $\theta = 0.7$, las dos entidades no son alineadas debido a la baja similitud.

$$align(\text{o1:carro,o2:automóvil}) = {}'\perp{}' \leftarrow sim_{agg}(\text{o1:carro,o2:automóvil}) = 0.5.$$

2.4.7 Iteración

Se puede considerar entidades como similares, si su posición en la estructura de la ontología es similar. La similitud de estructura es expresada a través de la similitud de otras entidades en la estructura. Por lo tanto, para calcular la similitud de una par de entidades, muchos de los métodos descritos dependen en la similitud de otros pares de entidades vecinos. Una primera ronda usa sólo métodos de comparación básicos basados en las etiquetas y similitudes entre cadenas para calcular la similitud entre entidades o alternativamente confiar en alineamientos entrados manualmente. Realizando cálculos en varias rondas, se puede acceder a los pares ya calculados y utilizar medidas de similitud de estructuras más sofisticadas. Esto está emparentado al algoritmo de similitud de *flooding* de Melnik et al. [76], que en contraste al paso de iteración en alineamientos de ontologías, no interpreta las aristas a través de la cual la similitud es esparcida.

Varios criterios de parada han sido descritos en la literatura: un número fijo de iteraciones, una restricción de tiempo fijo, o cambios de alineamientos por debajo de un umbral.

2.4.8 Salida

La salida del proceso es una lista de alineamientos como los presentados en la **Tabla 2**.

Tabla 2. Tabla de alineamiento con similitud

Ontología O_1	Ontología O_2	Similitud	Alineamiento
objeto	cosa	0.95	Sí
vehículo	vehículo	0.9	Sí
carro	automóvil	0.85	Sí
velocidad	velocidad	0.8	Sí
tieneVelocidad	tienePropiedad	0.75	Sí
Porsche KA-123	Porsche de Marc	0.75	Sí
30 km/h	rápido	0.6	no
motor	dueño	0.3	no

Definición 42 (Salida). Dadas dos ontologías, una salida de alineamientos es creada a través de

$$output: O \times O \rightarrow E \times E \times [0..1] \times \{alignment\}.$$

Como el alineamiento ha sido calculado basado en la similitud, también se adiciona el valor de similitud agregada en esta tabla, si es necesario puede almacenarse la similitud individual de cada rasgo. Para esta representación, es necesario marcar los pares que representen alineamientos válidos.

2.5 Evaluación de los métodos de alineamientos

La manera que tenemos para comparar los métodos de alineamiento es mediante los métodos de evaluación de alineamientos.

Distintas medidas de evaluación se han propuesto, entre ellas tenemos:

- Medidas de conformidad (*compliance measures*): proveen un entendimiento de la calidad de los alineamientos identificados.
- Medidas de rendimiento (*perfomance measures*): muestran que tan bueno es el algoritmo en términos de recursos computacionales.
- Medidas relativas al usuario (*user-realted measures*): ayudan a determinar la satisfacción del usuario, por ejemplo, a través del esfuerzo necesario del usuario.
- Medidas relativas a la tarea (*task-related measures*): mide cuán bueno fue el alineamiento para un cierto uso o aplicación.

A continuación explicaremos con mayor detalle cada una de las medidas mencionadas anteriormente.

2.5.1 Medidas de conformidad

La calidad del proceso de alineamiento es presentado por dos valores: precisión y *recall*.

Definición 43 (Precisión). Dado un alineamiento de referencia R, la precisión de un alineamiento A está dada por:

$$P(A,R) = \frac{|R \cap A|}{|A|}$$

La precisión mide la proporción de los alineamientos encontrados que son correctos. Esto corresponde a la exactitud y su inversa al índice de error. Una precisión de 1 significa que todos los alineamientos encontrados son correctos, pero no implica que todos los alineamientos hayan sido encontrados. Típicamente, la precisión es balanceada con otra medida de recuperación de información llamada *recall*.

Definición 44 (*Recall*). Dado un alineamiento de referencia R, el *recall* de un alineamiento A está dado por:

$$R(A,R) = \frac{|R \cap A|}{|R|}$$

El *recall* mide la proporción de alineamientos correctos en comparación al número total de alineamientos correctos existentes. Puede también ser referido como amplitud (*completeness*). Un *recall* alto significa que muchos de los alineamientos han sido hallados, pero no hay información acerca del número de alineamientos adicionales falsamente identificados. Un *recall* alto implica varios falsos alineamientos (precisión baja) y viceversa. Por lo tanto, usualmente la precisión y el *recall* son balanceados con la medida F-*measure* [77].

Definición 45 (F-*Measure*). Dado un alineamiento de referencia $\boldsymbol{R}$, la precisión y el *recall*, la medida F-*measure* de un alineamiento $\boldsymbol{A}$ está dado por:

$$F(A,R) = \frac{(b^2+1) \cdot P(A,R) \cdot R(A,R)}{b^2 \cdot P(A,R) + R(A,R)}$$

con $b = 1$ siendo el factor de peso estándar: $F_1(A,R) = \frac{2 \cdot P(A,R) \cdot R(A,R)}{P(A,R)+R(A,R)}$.

2.5.2 Medidas de rendimiento

El rendimiento de un algoritmo es medido por los recursos que éste consume. Los recursos más importantes son el tiempo de ejecución, la escalabilidad, el uso de memoria. El tiempo es un factor crítico para muchos casos. El tiempo es medido generalmente en segundos o en milisegundos. Un aspecto cercano al tiempo es la escalabilidad. La escalabilidad determina si la ejecución de un método es también realista para grandes escalas. El uso de memoria también es usado como otra medida rendimiento.

2.5.3 Medidas relativas al usuario

Muchos de los algoritmos requieren la interacción del conocimiento humano. Sin embargo, para algunas aplicaciones, la interacción con los usuarios no es deseable o incluso imposible. Diferentes niveles de automatización varían desde el alineamiento manual, posiblemente respaldada con una GUI (interfaz gráfica), generando proposiciones a sistemas completamente automáticos. No solamente se distinguirá entre automático y manual, también se medirá el nivel

de interacción del usuario requerido, por ejemplo, contar el número de pasos de interacción. Al final, las medidas relativas al usuario incluyen la satisfacción subjetiva que tenga un usuario.

2.5.4 Medidas relativas a la tarea

Las medidas relativas a la tarea se enfocan en la aplicabilidad de un método para una cierta tarea o caso de uso. Esto tiene un nivel alto de subjetividad. Debido a este problema, esta medida se usa sólo para métodos específicos, donde el método esté optimizado para diferentes casos de uso.

2.5.5 Otras medidas de conformidad

Definición 46 (*Overall*). Dado un alineamiento de referencia R, la precisión y el *recall*, la medida *overall* de un alineamiento A está dada por:

$$O(A,R) = R(A,R) \cdot \left(2 - \frac{1}{P(A,R)}\right)$$

La medida *overall* fue usada por Do y Rahm [74] y balancea los alineamientos correctos contra alineamientos falsos. Según Ehrig [3], esta medida parece inadecuada al alineamiento de ontologías. Un falso positivo es clasificado con la misma penalidad que un falso negativo. Éste es el problema, debido a que es normalmente mucho más fácil descartar un alineamiento equivocado encontrado que encontrar otro alineamiento desconocido. Por esta razón, esta medida no es aplicada.

Receiver- Operator- Curves

En las curvas *Receiver Operating Characteristic* (curvas ROC) [78], el índice de verdaderos positivos (también llamados sensibilidad) es trazado sobre el índice de falsos positivos (1-especificidad). Mientras que la curva esté cerca del borde izquierdo y del borde superior del espacio ROC, más precisos serán los resultados. Este tipo de evaluación es para medir la influencia valores de verdaderos positivos y falsos positivos, pero es insuficiente cuando el foco está sobre los positivos. Como se mencionó anteriormente, éste es el caso del alineamiento de ontología. Además, distintos conjuntos de datos no pueden ser comparados directamente. La cercanía a la esquina superior izquierda es relativa al tamaño de la ontología.

Medidas relajadas de calidad:

Las medidas de precisión y *recall* pueden ser criticadas por dos razones:

1. Ellas no discriminan entre un alineamiento muy malo y un alineamiento menos malo.
2. Ellas no miden el esfuerzo para ajustarse al alineamiento.

Usualmente, tiene sentido no sólo tener una decisión si una correspondencia particular ha sido hallada o no, si no también medir la proximidad del alineamiento encontrado. Esto implica que fallos cercanos deberían ser tomados en cuenta. La extensión natural de precisión y *recall* consisten en reemplazar la expresión $|R \cap A|$ en la definición estándar por una proximidad de solapamiento.

Definición 47 (Generalización de precisión y *recall*). Dado un alineamiento de referencia R y una función de solapamiento ω entre alinemientos, la precisión de un alineamiento A está dado por

$$P_\omega(A,R) = \frac{\omega(A,R)}{|A|}$$

y el *recall* está dado por:

$$R_\omega(A,R) = \frac{\omega(A,R)}{|R|}$$

Definición 48 (Similitud de solapamiento). Una medida de proximidad que generaliza la precisión y el *recall* es

$$\omega(A,R) = \sum_{(a,r)\in M(A,R)} \sigma_{pair}(a,r) \cdot \sigma_{rel}(a,r) \cdot \sigma_{conf}(a,r)$$

en la cual $M(A,R) \subseteq A \times R$ es una comparación entre las correspondencias de A y R y $\sigma(a,r)$ una función de proximidad entre dos correspondencias.

La función de proximidad σ es dividida entre tres elementos individuales. Para el alineamiento de ontologías, éstas son σ_{pair}, σ_{rel} y σ_{conf}. Ellas expresan el solapamiento en términos de par de alineamiento actual, si la relación encontrada (normalmente la equivalencia, pero también es posible admitir otras como la subsunción) es la misma, y los valores de confidencia son los mismos. Para M, nos basamos en las correspondencias existentes. Éstas son las correspondencias que maximizan la proximidad de solapamiento.

Hemos elaborado tres posibles instanciaciones concretas de σ_{pair}: proximidad simétrica, esfuerzo de corrección, y medidas orientadas a la precisión y al *recall*. Como ejemplo de similitud simétrica, usamos una simple en la cual un concepto es similar a un grado de 1 consigo mismo (por ejemplo, se ha identificado el alineamiento correcto), 0.5 con sus subconceptos y superconceptos directos y 0 con cualquier otro concepto. Esta similitud es probablemente aplicada a relaciones e instancias (a través de relaciones *parte-de*). El esfuerzo de corrección mide el número de acciones necesarias para alcanzar el alineamiento correcto. Y finalmente, las medidas orientadas, por ejemplo, cuando uno quiere recuperar instancias de un concepto, un subconcepto del esperado es correcto pero no completo, entonces afecta el *recall* y la precisión.

2.6 Estado del arte sobre técnicas de alineamientos de ontologías

A continuación se mostrará un estado del arte realizado por Ehrig [3] sobre métodos de alineamientos de ontologías y métodos de alineamientos de esquemas. Los métodos de alineamientos de ontologías, como lo indica su nombre, son métodos que permiten alinear las entidades de dos o más ontologías; los métodos de alineamientos de esquemas se basan en encontrar alineamientos, pero en estructuras semánticas más simples como los esquemas relacionales de las bases de datos. Resaltemos que, a partir de estructuras como esquemas, es posible construir ontologías y es una vía alternativa a los lenguajes de ontologías (por ejemplo

OWL) para almacenarlas. Además, es interesante tratar este tópico por la idea existente detrás de la solución del problema de la heterogeneidad entre los orígenes de datos (esquemas), las cuales pueden servir de base para el desarrollo de técnicas para alinear ontologías.

2.6.1 Métodos de alineamientos de ontologías

En esta sección se mostrarán las principales herramientas desarrolladas para el alineamiento de ontologías. En este tópico se tratarán las herramientas que reciben dos ontologías que deben de ser alineadas.

ONION

En la herramienta ONION (*ONtology compositION system*), [79] proveen una aproximación para resolver la heterogeneidad entre diferentes ontologías. La asunción principal es que, mezclando ontologías completas es muy costoso e ineficiente. Por consiguiente, se enfocan en crear las llamadas *reglas articuladas (articulation rules)*, las cuales enlazan conceptos correspondientes. Como la creación manual de esas reglas no es muy eficiente, se utiliza un método semiautomático que tiene en cuenta heurísticas entre varias relaciones simples como etiquetas, jerarquías de subsunción y valores de atributos. La información de diccionarios es usada también para el proceso de alineamiento. Para estas relaciones, un emparejamiento es presentado al usuario que tiene que decidir cuándo es válido o no el alineamiento. El enlazado de las reglas de articulación puede ser aplicado cuando una aplicación busca información de dos ontologías. El trabajo está basado en la teoría de composición de álgebras [80].

SMART, PROMPT, *Anchor*-PROMPT, *PromptDiff*

SMART fue el primer paso de un número de herramientas creadas por Noy y Musen [81]. Las herramientas están accesibles como *plug-ins* para el ambiente de ontologías Protègè [82][7]. SMART es un algoritmo basado principalmente en lingüística. Éste comprueba los nombres de conceptos para la similitud y luego empareja relaciones y atributos. SMART provee alineamientos, uno a uno, de entidades de ontologías.

PROMPT [83] es una herramienta que provee un método semiautomático para la mezcla de ontologías. Está basado en el algoritmo SMART. Después de haber identificado alineamientos por emparejamiento de etiquetas, el usuario es avisado de que debe marcar los pares de entidades que deben ser mezclados. Durante la mezcla, PROMPT presenta posibles inconsistencias como conflictos de nombres o relaciones que no apunten a nadie. El usuario entonces decide cómo reaccionar y resolver la cuestión manualmente.

Anchor-PROMPT [84] representa una versión avanzada de PROMPT que incluye medidas de similitud basadas en la estructura de las ontologías. Los llamados puntos de anclaje, pares alineados en las ontologías, son identificados primeramente a través de comparación de cadenas de las entidades o directamente asignados por el usuario. Basados en estos puntos de anclaje conocidos, la estructura de las ontologías es recorrida resultando en proposiciones de alineamientos adicionales de entidades entre los puntos de anclajes conocidos. Específicamente, los caminos son recorridos a lo largo de la jerarquía así como a lo largo de otras relaciones. Después, los resultados son presentados al usuario, incluyendo una explicación y el usuario decide si mezclar o no las entidades propuestas. Este proceso es continuado en varias iteraciones.

PromptDiff [85] es una herramienta para comparar diferentes versiones de ontologías. Diferentes heurísticas de emparejamiento son usadas para determinar las entidades similares.

[7] http://protege.stanford.edu/

Los comparadores son combinados de una manera de punto fijo hasta que no ocurra ningún cambio. *PromptDiff* hace uso del hecho de que dos versiones de una ontología tienen un solapamiento considerable.

La PROMPT-Suite [86] consiste en diferentes métodos afrontando diferentes cuestiones en torno del alineamiento de ontologías. PROMPT es una de las herramientas más usadas para la mezcla de ontologías, debido a su fácil uso con el ambiente Protègè.

Chimaera

Chimaera [87] es una herramienta interactiva para mezclar ontologías. Su formato de ontología básico es OKBC[8], pero puede manejar otros lenguajes. Después de ejecutar un comparador lingüístico, *Chimaera* usa el resultado para desencadenar la operación de mezcla. Durante este proceso, el usuario tiene que decidir si mezclar o no. *Chimaera* también provee proposiciones en la reorganización de la taxonomía una vez que la mezcla haya sido procesada. Sobre todo, *Chimaera* permite el diagnóstico y la edición manual para la mezcla de ontologías. El alineamiento de entidades actual, sin embargo, está basado en medidas simples.

FCA-*Merge*

El método FCA-*Merge* fue presentado por Stumme y Maedche [88]. Como su nombre indica, su objetivo es mezclar ontologías. Se basa en análisis formal de conceptos, descrito en Ganter y Wille [60]. Dadas dos ontologías, en un primer paso, FCA-*Merge* las puebla con instancias que son extraídas de un conjunto de documentos. Este paso es necesario, debido a que la mayoría de las ontologías no tienen instancias suficientes, pero son un requerimiento para el análisis formal de conceptos. Basadas en estas instancias, las ontologías son presentadas como retículos de conceptos (*lattice*, en inglés), es decir, los conceptos son vistos como un conjunto de instancias. En este punto, la información léxica es utilizada para recuperar información específica del dominio. Usando análisis formal de conceptos, los dos contextos son integrados y un nuevo retículo es creado. Los pasos de poda son aplicados para mantener pequeño el tamaño del retículo. En un último paso, el retículo es transformado de nuevo en una ontología. Este paso debe realizarse manualmente. Para resolver conflictos, como elementos duplicados, FCA-*Merge* tiene un soporte automático para guiar al usuario a través del proceso. Se debe mencionar que FCA-*Merge* trata sólo con jerarquía de conceptos y las relaciones instancias – alineamientos no son soportadas.

LSD-GLUE

El sistema LSD (*Learning Source Descriptions*) usa técnicas de aprendizaje de máquinas (*machine learning*) para emparejar un origen de datos desconocido contra un esquema global previamente determinado [89]. Dado un alineamiento proporcionado por un usuario de un origen de datos al esquema global, el paso de preprocesamiento examina instancias de ese origen de datos para entrenar el clasificador, descubriendo patrones de características de instancias y reglas de comparación. Si las instancias de conceptos en el segundo esquema coinciden con el primer clasificador, los conceptos son considerados idénticos. Los resultados individuales de emparejamiento son usados de nuevo para entrenar un comparador "global". Aplicando este comparador, es posible determinar alineamientos entre el origen de datos global y los nuevos orígenes.

LSD fue extendido al sistema GLUE [90]. GLUE es más orientado a ontologías. Como LSD, busca el concepto más similar en dos ontologías usando varios comparadores. El componente de aprendizaje determina los clasificadores de conceptos (comparadores) para instancias basándose en descripciones de instancias, es decir, el contenido textual de una página web o su nombre. De

[8] http://www.ksl.Stanford.edu/software/OKBC/

hecho, GLUE utiliza una estrategia de aprendizaje múltiple debido a que hay diferentes tipos de información en las que los clasificadores de conceptos pueden basarse. Éstos pueden variar desde la frecuencia de palabras en un documento o los formatos de valores. De estos conceptos aprendidos, los clasificadores derivan si hay conceptos que se corresponden en dos esquemas.

Los conceptos y relaciones son además comparados utilizando *etiquetado relajado (relaxation labeling*, en inglés*)*. La intuición del etiquetado relajado es que la etiqueta de un nodo (en nuestra terminología: el alineamiento asignado a una entidad) está típicamente influenciada por los rasgos de la vecindad del nodo en el grafo. Los autores de este trabajo mencionan explícitamente subsunción, frecuencia y nodos cercanos. Un alineamiento óptimo local para cada entidad es determinado usando los valores de similitud resultantes de los pares de entidades vecinas de una ronda previa. Las restricciones de similitud individuales son sumadas para la probabilidad final de alineamiento. El etiquetado relajado adicional, que considera la estructura de la ontología, está basado solamente en reglas predefinidas codificadas manualmente. Normalmente, uno necesita verificar todas las configuraciones posibles de etiquetado, las que incluyen los alineamientos de todas las demás entidades. Los desarrolladores están conscientes del problema de la complejidad creciente, así que establecieron particiones sensibles, es decir, conjuntos de etiquetas con los mismos rasgos son agrupados y procesados sólo una vez. Las probabilidades de partición son determinadas. Una suposición es que los rasgos son independientes, lo cual admiten los autores de GLUE que no tiene que ser verdad. A través de la multiplicación de las probabilidades, finalmente se recibe la probabilidad de una etiqueta ajustable al nodo, es decir, de una entidad siendo alienada con otra. El par con máxima probabilidad es el resultado final del alineamiento.

El método de aprendizaje de máquina de GLUE es adecuado para un escenario con descripciones textuales extensivas de instancia, pero puede no ser adecuado para un escenario concentrado en el esquema de ontología.

OLA

OWL *Lite Aligner* (OLA) fue introducido por Euzenat y Valtchev [91]. Utiliza diferentes componentes de las ontologías involucradas para determinar la similitud. La similitudes bases son calculadas a partir de las etiquetas. Iterativamente, las similitudes bases influencian cada otra hasta que las similitudes estén bien balanceadas entre todos los pares de las ontologías. En cada iteración, las similitudes son recalculadas tomando en cuenta la similitud de los nodos vecinos, donde vecino se refiere a que exista una relación entre ellos. Esto hace a OLA un método que utiliza información de los elementos e información estructural. Las similitudes calculadas son pesadas diferentemente de acuerdo a la relación (por ejemplo, subsunción o instanciación). El usuario establece estos pesos de acuerdo a sus preferencias. Encontrar los alineamientos correctos es un problema de optimización de maximizar similitudes. La herramienta OWL *Lite Aligner* realiza alineamientos, uno a uno, de conceptos, relaciones e instancias.

2.6.2 Método de alineamientos de esquemas

El alineamiento de esquemas está estrechamente relacionado con el alineamiento de ontologías. A diferencia de las herramientas anteriores, las siguientes propuestas reciben esquemas como entrada, los cuales pueden ser esquemas relacionales de bases de datos, esquemas XML, grafos y otras estructuras con menor semántica que las ontologías. Estas técnicas tienden a concentrarse más en la representación de la información que en su contenido y por lo tanto, se concentran más en la estructura.

SemInt

SemInt [64] crea alineamientos entre atributos individuales de dos esquemas. A diferencia de la mayoría de los métodos, éste no provee un emparejamiento basado en nombre o un emparejamiento basado en grafos. Se basa en el análisis de la información disponible del esquema de un gestor base de datos relacional y las instancias de los datos. La distribución de los datos y la media son convertidas a signaturas. Para estas signaturas, *SemInt* aplica dos operadores de similitud. Utiliza la distancia euclidiana o una red neuronal entrenada para determinar los candidatos del emparejamiento. Los autores expresan que ambos métodos tienen ventajas y desventajas que difieren de acuerdo a la aplicación. Las redes neuronales afrontan problemas de eficiencia [92]. La contribución de SemInt es proveer uno de los primeros métodos que no opten por una combinación cableada, es decir, datos de similitudes basadas en reglas individuales insertados directamente en un programa, que no pueden ser cambiados o modificados posteriormente, excepto usando un método de aprendizaje de máquina.

DIKE

DIKE [93] es un método que determina automáticamente sinónimos e inclusión (relaciones es-un, hiperonimia). La entrada son esquemas entidad-relacional. Al final, diferentes valores de similitudes entre dos objetos son calculados basados en sus objetos relacionados como los atributos. Éstos también sólo pueden relacionarse indirectamente a través de caminos de relaciones. Mientras más distantes estén los objetos, menos importantes son para determinar la similitud. La meta no es encontrar objetos similares sino necesariamente idénticos. También identifica otros tipos de relaciones. Una relación existe si el valor de similitud está por encima de un umbral fijo.

Artemis

ARTEMIS (*Analysis of Requirements: Tool Environment for Multiple Information Systems*) [94] es un componente de MOMIS, un mediador de bases de datos heterogéneas [94]. Se integran esquemas desarrollados individualmente en un esquema virtual global. ARTEMIS está basado en diferentes similitudes (a las que los autores se refieren por afinidad), es decir, similitud de nombre (usando *WordNet*), similitud de tipos de datos y similitud estructural de las entidades involucradas. Estas similitudes son sumadas después con pesos apropiados. Basada en la similitud global y en técnicas de agrupamiento jerárquico, ARTEMIS categoriza clases en grupos, donde cada grupo presenta una clase más general con un conjunto de atributos globales. A través de una tabla de asociaciones, los esquemas originales son enlazados al esquema virtual global.

Cupid

Cupid [95] es un comparador híbrido basado en emparejamiento de elementos y de nivel estructural. En términos de datos de entrada, es muy genérico y ha sido aplicado a XML y a diferentes modelos relacionales. El algoritmo comprende tres pasos. En el primer paso, se comparan elementos (nodos del esquema) por medios lingüísticos, además, incluye información externa de sinónimos. Segundo, para el emparejamiento estructural, el modelo de datos tiene que ser transformado en un árbol. Los pares son comparados entonces examinando sus hojas. Una similitud es calculada a través de una media pesada de la similitud lingüística y estructural. En la tercera fase, un umbral es aplicado para decidir si hay un alineamiento o no. Los autores enfatizan que establecer el valor del umbral es dependiente de la aplicación y no puede realizarse de manera general.

Similarity Flooding, Rondo

Melnik, et al. [76] presentaron un método para la integración basado en el concepto de *similarity flooding*. Fue implementado después por Melnik, et al. [96]. Los esquemas de entrada son grafos

dirigidos acíclicos. La similitud del nivel de elementos básicos es determinada usando comparaciones de cadenas. A través de un cálculo de punto fijo, este valor inicial de similitud es propagado a partir de los nodos similares, a través de los grafos y hacia los vecinos adyacentes basados en coeficientes de propagación. Sin embargo, como las aristas no están etiquetadas, no se explota ninguna interpretación semántica de ellos. Después de unas iteraciones, la similitud converge a un máximo, el punto fijo es alcanzado.

Muy similar a este método es el sistema Falcon de Hu et al. [97]. Ellos usan dos grafos que se influencian uno al otro: el objeto actual y la instrucción RDF. Esto permite incluir etiquetas a las aristas, es decir, relaciones. Las similitudes son entonces calculadas en estos dos grafos.

COMA

COMA (*COmbination Matching Algorithms*) [74] es un sistema de emparejamiento de esquemas que combina múltiples comparadores individuales de un modo flexible. El objetivo es emparejar esquemas reales. Antes de comenzar el alineamiento, los esquemas son transformados en grafos dirigidos acíclicos. El proceso de alineamiento consiste en tres fases que son repetidas iterativamente. La primera fase es una retroalimentación del usuario opcional. Aquí el usuario establece los valores de los parámetros del algoritmo, por ejemplo, escoge entre los comparadores y acepta o rechaza los comparadores propuestos. Después, los emparejamientos individuales son calculados. Esto se hace principalmente utilizando información lingüística, usando también diccionarios y elementos estructurales como hijos u hojas. En la tercera fase, los comparadores son combinados usando una estrategia de peso máximo, promedio o mínimo. Se dan diferentes estrategias para determinar el umbral. Es posible ejecutar el algoritmo en un modo completamente automático.

CTXMatch, S-Match

Giunchiglia et al. [68] presentaron un método para derivar relaciones semánticas entre clases de dos esquemas de clasificación, los cuales son extraídos de bases de datos o de ontologías. Basados en las etiquetas, el sistema identifica entidades equivalentes. Para esto, hace uso de sinónimos definidos en *WordNet*. A través de un solucionador SAT, el sistema identifica relaciones adicionales entre dos esquemas. El solucionador SAT toma en cuenta la estructura de los esquemas, especialmente la taxonomía y sus implicaciones inferidas, por ejemplo, cualquier objeto en una clase es también un elemento de todas las superclases. Como resultado, el sistema devuelve equivalencia, subsunción o incompatibilidad entre dos clases. En una versión reciente de *S-Match,* también provee una explicación de los alineamientos [98].

3 Alineamiento de geo-ontologías

Anteriormente, tratamos a las ontologías de manera “pura”, es decir, ontologías sin considerar rasgos específicos del dominio geoespacial. Sin embargo, para manejar las particularidades de un fenómeno geográfico, una ontología convencional puede no ser lo suficientemente expresiva. Debido al nivel de especialización de las ontologías en el dominio geoespacial, surge un concepto denominado ontología geográfica o geo-ontología.

3.1 Definición de geo-ontología por Nudelman, Iochpe y Ferrara

En la sección 2.1.3, Hess et al. [4] definieron una ontología como una 4-tupla $O =< C, P, I, A >$, donde:

- C es el conjunto de conceptos.
- P es el conjunto de propiedades.
- I es el conjunto de instancias.
- A es el conjunto de axiomas.

Según estos autores, una geo-ontología puede ser vista como una extensión de una ontología convencional.

Definición 49 (Geo-ontología). Tomando la definición de ontología anterior, una ontología geográfica o geo-ontología se define como una extensión de una ontología siendo una 4-tupla $O =< C, P, I, A >$, donde:

- C es el conjunto de conceptos.
- P es el conjunto de propiedades.
- I es el conjunto de instancias.
- A es el conjunto de axiomas.

A diferencia de los conceptos de una ontología convencional, un concepto c se puede clasificar en concepto de dominio, como *Río*, *Parque*, *Edificio*, o en concepto geométrico, como *Punto*, *Línea*, *Edificio*, o en un concepto de tiempo como *Instante* o *Periodo*. Además, un concepto de dominio geográfico gc es una especialización de un concepto de dominio que representa un fenómeno geográfico.

En una geo-ontología, una propiedad $p \in P$ puede ser de los siguientes tipos:

- Propiedad convencional: recordemos de la **Definición 7** de ontología que una propiedad es una componente que está asociada a un concepto c con el objetivo de caracterizarlo. Puede ser una propiedad de tipo (*integer, double, string*) o una propiedad de tipo objeto que permiten varios tipos de valores. Una propiedad de tipo de dato puede ser vista como un atributo de una base de datos, mientras que una propiedad de tipo objeto puede ser vista como una relación de base de datos. Una propiedad de tipo objeto representa una asociación entre un concepto de dominio, geográfico o no.
- Propiedad espacial (topológica, direccional o métrica): es siempre una propiedad de tipo objeto y representa una asociación entre dos conceptos geográficos. Las relaciones espaciales tienen una semántica predefinida y son estandarizadas por el consorcio OGC, (Open Geospatial Consortium)[9] . Las relaciones convencionales, por el contrario, pueden asumir diferentes semánticas en dependencia del concepto asociado.
- Propiedad geométrica: es una asociación entre un dominio geográfico con un concepto geométrico. Es una propiedad de tipo objeto.
- Propiedad posicional: es una propiedad de tipo de dato que debe estar asociada a un concepto geométrico para darle su ubicación (conjunto de coordenadas).
- Propiedad de tiempo: es una asociación entre un concepto de dominio y un concepto de tiempo.

Una instancia geográfica $gi \in I$ es una extensión de la instancia i. Al igual que una instancia geográfica, debe estar asociada, al menos, a una instancia de un concepto geométrico. El valor de una propiedad posicional, llamémosle *hasLocation*, devuelve la posición espacial (coordenadas) de esa instancia geográfica.

Sobre la base de este modelo de referencia es posible apuntar al menos tres diferencias entre las geo -ontologías y las ontologías convencionales.

[9] OGC (http://www.opengeospatial.org/) es una organización de normas de consenso que encabeza el desarrollo de estándares para servicios geo-espaciales y localización.

- Las relaciones espaciales tienen una semántica predefinida y están estandarizadas en la literatura [99], mientras que las relaciones convencionales pueden asumir diferentes semánticas dependiendo de los conceptos asociados.
- Cada concepto geográfico tiene, al menos, una geometría asociada que los representa. La geometría juega un papel fundamental en la definición de las posibles relaciones espaciales que pueda tener el concepto.
- Una instancia geográfica tiene un número de pares de coordenadas (x, y) que representa su posición espacial sobre la superficie. Estas coordenadas están expresadas en un sistema de coordenadas dado.

3.2 Clasificación de las heterogeneidades geográficas

Según Hess et al. [4], los tipos de heterogeneidades que se deben tomar en cuenta cuando se comparan ontologías geográficas se clasifican en heterogeneidades a nivel de concepto y heterogeneidades a nivel de instancias.

3.2.1 Heterogeneidades a nivel de concepto

Las posibles heterogeneidades son clasificadas basadas en la comparación de un concepto c que pertenece a la ontología O contra un concepto c' que pertenece a una ontología O'. Considerando la Definición 49 de geo-ontología como una tupla de conceptos, instancias, propiedades y axiomas, las posibles heterogeneidades se definen como:

Heterogeneidad de nombre: La heterogeneidad del nombre del concepto NH ocurre cuando, dados los dos nombres de conceptos $t(c)$ y $t(c')$, no son iguales, ni son sinónimos. La relación de sinonimia $SYN(t(c), t(c))$ es obtenida buscando en un tesauro externo o diccionario.

$$NH(c, c') = ((t(c) \neq t(c')) \wedge (SYN(t(c), t(c')) = false)$$

Heterogeneidad de propiedad: La heterogeneidad de concepto de propiedad PH ocurre cuando hay una heterogeneidad de atributo AH o una heterogeneidad de relación RH.

La heterogeneidad AH entre un concepto c que pertenece a la ontología O y un concepto c' que pertenece a una ontología O' ocurre cuando al menos uno de los atributos $a(t(p), dtp) \in P$ en la ontología O no se corresponde con ninguno de los atributos $a(t(p'), dtp') \in P'$ en la ontología O', donde $t(p)$ es el nombre de un atributo (propiedad) y dtp es el tipo de dato del atributo. La heterogeneidad puede ser generada debido a los diferentes nombres de atributos o los diferentes tipos de datos de los atributos.

$$AH(c, c') = \left(\exists a(t(p), dtp) \in P \middle| \forall a(t(p'), dtp') \in P', \left(t(p) \neq t(p')\right) \vee (dtp \neq dtp')\right)$$

La heterogeneidad RH entre un concepto c que pertenece a la ontología O y un concepto c' que pertenece a una ontología O' se define sobre relaciones convencionales (o sea, relaciones que no son geométricas, ni espaciales). Ocurre cuando al menos una de las relaciones $cr(t(p), t(c_x), minCard, maxCard) \in P$ de la ontología O no tiene una relación

$cr(t(p'), t(c'_x), minCard', maxCard') \in P'$ en la ontología O' que le corresponda, donde $t(p)$ es el nombre de la relación (propiedad), $t(c_x)$ es el concepto asociado y $minCard$ y $maxCard$ son, respectivamente, las cardinalidades mínima y máxima de la relación. La heterogeneidad puede ocurrir debido a conceptos diferentes c_x asociados, y a diferentes cardinalidades de la relación $minCard$ y $maxCard$ asociadas, respectivamente. A veces, los nombres de las relaciones convencionales no son significativos para identificar la relación, por lo que la componente $t(p)$ puede ser ignorada.

$$RH(c, c') = (\exists cr(t(p), t(c_x), minCard, maxCard) \in P | \forall cr(t(p'), t(c'_x), minCard', maxCard') \in P', (c_x \neq c'_x) \vee (minCard \neq minCard') \vee (maxCard \neq maxCard')).$$

Heterogeneidad de jerarquía: La heterogeneidad de jerarquía HH entre dos conceptos, c que pertenece a la ontología O y c' que pertenece a la ontología O', ocurre cuando el conjunto de superclases del concepto c es diferente del conjunto de superclases del concepto c'. Esto significa que al menos una relación jerárquica $h(c, c_x)$ de la ontología O no tiene una correspondencia $h(c, c'_x)$ en la ontología O', donde c_x es la superclase de c_x.

$$HH(c, c') = (\exists c_x \in h(c, c_x) | \forall c'_x \in h(c, c'_x), c_x \neq c'_x)$$

Heterogeneidad de relaciones espaciales: Se dividen en *heterogeneidad de relaciones topológicas* y *heterogeneidad de relaciones direccionales*. Las *relaciones métricas* no son consideradas porque en general son calculadas por los sistemas de información geográficos (GIS, por sus siglas en inglés) y no se definen como propiedad o restricciones de un concepto.

La heterogeneidad de relación direccional DH entre dos conceptos geográficos, gc que pertenece a la ontología O y gc' que pertenece a la ontología O', ocurre cuando al menos hay una relación direccional $dr(t(p), t(gc_x), minCard, maxCard) \in P$ en la ontología O sin una correspondencia $dr(t(p'), t(gc'_x), minCard', maxCard') \in P'$ en la ontología O', donde gc_x es el concepto asociado, $t(p)$ es el nombre de la relación y $minCard$ y $maxCard$ son las cardinalidades mínima y máxima, respectivamente.

$$DH(gc, gc') = \exists dr(t(p), t(gc_x), minCard, maxCard) \in P | \forall dr(t(p'), t(gc'_x), minCard', maxCard') \in P', (gc_x \neq gc'_x) \vee (t(p) \neq t(p')).$$

La heterogeneidad de relación topológica TH entre dos conceptos geográficos, gc que pertenece a la ontología O y gc' que pertenece a la ontología O', es un poco más complicada debido a que la equivalencia de las relaciones depende de las geometrías asociadas. Ocurre si la combinación de los nombres de las relaciones y las geometrías involucradas, dadas por la función $top(geo, geo_x, t(p))$ y $top(geo', geo'_x, t(p'))$, no son equivalentes, donde geo y geo' son, respectivamente, las geometrías de los conceptos gc y gc' y $t(p)$ es el nombre de la relación.

$$TH(gc, gc^{'}) = \exists tr(t(p), t(gc_x), minCard, maxCard) \in P | \forall tr\left(t(p^{'}), t\left(gc^{'}{}_x\right), minCard^{'}, maxCard^{'}\right) \in P^{'}, top\left(geo, geo_x, t(p)\right) \neq top\left(geo^{'}, geo^{'}{}_x, t(p^{'})\right).$$

3.2.2 Heterogeneidades a nivel de instancia

En el campo geográfico hay muchos rasgos que pueden influenciar el proceso del cálculo de la similitud que no están presentes cuando se tratan con datos no geográficos. Estos rasgos son, por ejemplo, la escala, la posición espacial, el tiempo en que estas instancias han sido obtenidas, etc. Sin embargo, las propiedades no espaciales, como los valores de los atributos (propiedad) no pueden ser negadas. En esta sección se definen las heterogeneidades que pueden ocurrir a nivel de instancias cuando se comparan dos geo-ontologías.

Heterogeneidad de identificador: Cuando un concepto en una ontología es instanciado, a esta instancia se le asigna un único identificador con el cual el usuario y la computadora la pueden identificar. Cuando dos instancias, i que pertenece a la ontología O e i' que pertenece a la ontología $O^{'}$, no tienen el mismo identificador (por ejemplo, en OWL se obtiene con el parámetro ID) hay una heterogeneidad de identificador IIH.

$$IIH(i, i^{'}) = \exists i \in O | \forall i^{'} \in O^{'}, id(i) \neq id(i^{'}).$$

Heterogeneidad posicional: Una de las principales características de los datos geográficos es que tiene una posición sobre la superficie. El conjunto de coordenadas de una clase es obtenido indirectamente a través de la instancia del concepto geográfico que le es asociada. Si dos instancias i que pertenece a la ontología O e i' que pertenece a la ontología $O^{'}$, no tienen la misma posición espacial, hay una heterogeneidad posicional ICH. Asumiremos la existencia de una función $pos(i)$ que da la ubicación de una instancia. Esta función devuelve el conjunto de coordenadas de la instancia geográfica.

$$ICH(i, i^{'}) = i \in O | i' \in O' \wedge pos(i) \neq pos(i^{'}).$$

Heterogeneidad de atributos: Cuando una propiedad de un concepto es una propiedad de tipo de dato, esta representa a un atributo, es decir, propiedades cuyos valores permitidos son cadenas, enteros, etc. Cuando dos instancias i que pertenece a la ontología O e i' que pertenece a la ontología $O^{'}$, tienen diferentes valores para la misma propiedad de tipo de dato, hay una heterogeneidad de atributo IAH.

$$IAH(i, i^{'}) = \left(\exists at(t(c), t(p), v) \in O | \forall at(t(c), t(p^{'}), v^{'}) \in O^{'}, (p \equiv p') \wedge (v \neq v')\right)$$

donde i es la instancia que tiene el atributo, $t(p)$ es el nombre de la propiedad $t(p)$, v es el valor de para esa propiedad, $t(c)$ es el nombre de la clase a la que pertenece la instancia i y $at(\cdot)$ es el atributo en cuestión.

Heterogeneidad de relación: Cuando una propiedad de un concepto es una propiedad de tipo objeto, esta representa a una relación, es decir, una propiedad que permite valores que son instancias de varios conceptos. Cuando dos instancias i que pertenece a la ontología O e i' que

pertenece a la ontología O' tienen asociadas, respectivamente, las instancias i_x e i'_x, las que representan diferentes conceptos, hay una heterogeneidad de relación IRH.

$$IRH(i,i') = \left(\exists rl(t(p), id(i_x)) \in O | \forall rl\left(t(p)', id(i'_x)\right) \in O' \wedge \left(id(i_x) \neq id(i'_x)\right)\right)$$

donde $t(p)$es el nombre de la propiedad p, $id(i_x)$ es la instancia asociada y $rl(\cdot)$ es la relación en cuestión.

Heterogeneidad de metadatos: Los metadatos no tienen una influencia directa en la heterogeneidad entre dos instancias geográficas. En su lugar, la influencia es indirecta, lo que significa que las diferencias en los valores de los metadatos pueden conllevar a heterogeneidades relativas a los otros elementos de la instancia (coordenadas y propiedades). Por ejemplo:

- Dependiendo del valor para el metadato *date*, el valor para algún atributo descriptivo puede variar, por ejemplo, la población de una ciudad. Incluso algunas relaciones espaciales pueden ser diferentes, por ejemplo, el área de una ciudad.
- Dependiendo del valor para el metadato *projection* (UTM, planar), la geometría y las coordenadas de una instancia cambian.

3.3 Estado del arte sobre trabajos realizados para alinear geo-ontologías

Cualquier método de alineamiento de ontologías convencional puede ser utilizado para alinear ontologías geográficas. Las propiedades no geográficas (atributos y relaciones) pueden ser alineadas por un método convencional visto en la primera parte. Sin embargo, las propiedades que representan relaciones espaciales no pueden ser alineadas debido a que estos métodos convencionales no conocen su semántica específica. Los métodos de alineamientos de geo-ontologías se basan principalmente en las instancias de los objetos que contienen la información espacial se dichos objetos.

Seguidamente, en el estado del arte, se podrá observar la evolución de métodos que utilizan la estructura jerárquica y hacen uso de métodos que trabajan a nivel de términos, que no utilizan ninguna característica propia de las geo-ontologías, a métodos que explotan la información geográfica brindada por las instancias.

Rodríguez et al. [100] propusieron una aproximación para el cálculo de las similitudes entre los rasgos geo-espaciales de las definiciones de las clases utilizando una medida de similitud asimétrica. La evaluación de la similitud es básicamente hecha sobre las interrelaciones semánticas entre las clases. En ese sentido, ellos no sólo consideran la relaciones *es-un* y *parte-de* sino también los rasgos distintivos (partes, funciones y atributos). En adición a las relaciones semánticas y a los rasgos distintivos, se toman en cuenta dos conceptos lingüísticos para la definición de clases de entidades: las palabras y los significados, la sinonimia y la polisemia (homonimia). Trabajos posteriores, usando ontologías y teoría de conjuntos [101], determinaron la similitud semántica entre clases de entidades de ontologías diferentes. Esta aproximación se enfoca en alinear vocabularios grandes con una organización jerárquica.

Hakimpour y Timpf [102] propusieron el uso de ontologías en la resolución de las heterogeneidades semánticas, especialmente para aquellas encontradas en los Sistemas de Información Geográficos (GIS). Hakimpour y Geppert [103] proponen una aproximación de integración de bases de datos que emplea la mezcla de ontologías formales. Las ontologías de origen (una por cada base de datos) son mezcladas por sistema de razonamiento que encuentra relaciones de similitudes semánticas entre las diversas definiciones usadas para cada concepto.

Un integrador de esquemas construye un esquema global de la base de datos usando los esquemas de origen y las asociaciones encontradas en el proceso de mezcla.

Uitermark [104] propuso un marco para la integración semántica de conjuntos de datos para un mismo dominio, por ejemplo, el dominio geográfico; el marco abarca ontologías de aplicación del conjunto de datos y una ontología de dominio con referencias a conceptos generales. El marco también contiene un conjunto de reglas de inspección ("*survey rules*") que determinan cómo transformar un terreno a un conjunto de datos geográficos, representado por un conjunto de instancias de objetos. Finalmente, los conceptos de la ontología de dominio son manualmente refinados para reflejar los conceptos en las ontologías de aplicación y se determinan las relaciones semánticas (equivalencia, subclase/superclase, parte/todo). Este marco realiza la integración de ontologías en dos pasos. En el primer paso, se obtienen pares de objetos que se solapan a partir de dos conjuntos de datos. En el segundo paso, estos pares son examinados para comprobar la consistencia de las reglas de inspección. Esto determina cuándo los objetos representan el mismo objeto físico o no.

Fonseca et al. [1] propusieron un marco de trabajo para el desarrollo de aplicaciones geográficas usando ontologías. El marco de trabajo usa las ontologías como elemento fundamental para la integración de la información geográfica. Fonseca creó un mecanismo que permite que la información geográfica sea integrada a un sistema de información geográfico (GIS) basado principalmente en su significado. Fonseca abrió una nueva generación para el desarrollo de los GIS, a los que se les agregan las ontologías y a esta aproximación se le llamó ODGIS (*Ontology Driven* GIS). El uso de una ontología, traducida a una componente de un sistema de información, es la base de los ODGIS.

Kavouras y Kokla [105] definen un método de mezcla de ontologías basado en el análisis formal de conceptos (*Formal Concept Analysis*, FCA). FCA está basado en una definición matemática de conceptos a través de retículos [60]. La aproximación de Kavouras y Kokla define un método que, en 7 pasos, obtiene un retículo que representa a los conceptos temáticos. En este trabajo, se necesita un usuario experto para identificar los atributos y categorías sugeridos dentro de un dominio y sus relaciones.

Sotnykova et al. [106] plantean que la integración de información espacio-temporal (esquema y después los datos) es un proceso de tres pasos: pre-integración (resolución de conflictos sintácticos), aserciones de correspondencias de esquemas (*Inter-Schema Correspondence Assertions*) (resolución de conflictos semánticos) y generación de un esquema integrado (resolución de conflictos estructurales). Para la resolución de conflictos semánticos los autores proponen un lenguaje basado en la lógica de descripción. Después, un servicio de razonamiento de la lógica de descripción es usado para comprobar la satisfacibilidad de los dos esquemas de origen y el conjunto de correspondencias inter-esquemas. El mecanismo de razonamiento basado en la lógica de descripción se utiliza para validar el conjunto de correspondencias inter-esquemas con los esquemas de origen.

Schwering y Raubal [107] definen una medida de similitud asimétrica basada en espacios conceptuales [108]. Una región conceptual es una representación de un concepto como una región convexa *n*-dimensional en un espacio vectorial, donde cada dimensión corresponde a un atributo. La medida de similitud es obtenida como el promedio de las mínimas distancias entre cada componente del vector en una región conceptual y otra región conceptual. De manera general, las medidas de similitudes entre conceptos geo-espaciales estiman la similitud entre instancias usando los puntos representados en un espacio vectorial, que es el conjunto de datos. La medida de similitud presentada por los autores está basada en medir la similitud de las

instancias, pero mide la distancia entre los conceptos representados como una región convexa en el espacio.

Duckham y Worboys [109] adoptan un método extensional (basado en instancias o individuos) para alinear ontologías. Ellos definieron un método algebraico para mezclar (generar una estructura compartida entre dos conjuntos de datos) e integrar (obtener un nuevo mapa combinando los dos conjuntos de datos de origen), basado en la distribución espacial de los datos. El método de mezcla se basa en la suposición de que, si la extensión espacial de un valor de un conjunto de datos está contenida con ese valor en el otro conjunto de datos, el primer valor es una subclase del segundo. Los autores tomaron un método extensional fundamentándose en que la información geográfica está bien estructurada y es una fuente voluminosa de instancias sobre las cuales se realizará un proceso de razonamiento inductivo. El razonamiento inductivo encuentra, a partir de casos específicos, reglas generales. En el contexto geográfico, la inferencia inductiva es usada para inferir relaciones semánticas entre categorías de entidades geográficas (reglas generales) a partir de relaciones espaciales entre conjuntos de entidades. Este trabajo permitió ver los datos geográficos desde otro punto de vista, basado en las instancias. Las instancias, en las geo-ontologías, son una fuente rica de información, en ellas se pueden encontrar datos significativos como la posición geográfica de una entidad, que ahora aportarán la información principal en el proceso de alineamiento.

Cruz et al. [110] consideran que las ontologías estén relacionadas en un mismo dominio. Aunque este trabajo se propone la aplicación para alineamientos de ontologías de un dominio geoespacial, su utilización no lo restringe a ese dominio, sino que pueden ser aplicados a las ontologías convencionales. El método seguido es considerar como jerarquías a las ontologías, como el aspecto esencial en el proceso de alineamiento. Un usuario experto inicialmente identifica los niveles de las jerarquías que son alineados. Seguidamente, este alineamiento es propagado por la jerarquía siguiendo una estrategia de abajo hacia arriba, conocida en inglés como *bottom-up*, es decir, se considerarán que dos conceptos son equivalentes si tienen hijos equivalentes. Se permite que el usuario pueda asistir al proceso de alineamiento permitiéndole introducir alineamientos manualmente. Esta aproximación es la base de la herramienta visual *AgreementMaker* [111], que permite visualizar ontologías y mostrar los alineamientos generados. *AgreementMaker* es una propuesta que consta de cuatro capas para el cálculo de la similitud entre las entidades de las ontologías. La primera capa sustituye los alineamientos iniciales introducidos por un experto o por el uso de métodos lingüísticos con el que se compararán los nombres de las entidades y hacen uso de un diccionario para permitir el análisis de sinónimos. Seguidamente, se refinan los alineamientos propagándolos por la jerarquía y permitiendo al usuario incorporar los alineamientos identificados por él. Estas tareas son realizadas en las capas dos y tres. Finalmente, la última capa es la encargada de consolidar las similitudes para obtener el resultado final. Sunna y Cruz [111] proponen una mejora introduciendo dos medidas de similitudes basándose en métodos estructurales, es decir, considerando la estructura jerárquica de las ontologías. Estas medidas son aplicadas después de haberse calculado una similitud base utilizando métodos lingüísticos, con el objetivo de alcanzar una mayor precisión. Estas nuevas medidas toman en consideración los ancestros de los conceptos y los hermanos de los conceptos, respectivamente.

Basados en el trabajo de Duckham y Worboys [109], el trabajo de Navarrete y Blat [112] realiza la mezcla de dos conjuntos de datos basada en la distribución espacial de los valores. Este algoritmo está basado en el nivel de solapamiento entre las extensiones espaciales (vector de unidades espaciales) de los conjuntos de valores de los conjuntos de datos. Un alto solapamiento entre dos extensiones espaciales de distintos conjuntos de datos significa que probablemente se

refiera a temas equivalentes. Si la extensión espacial del primer valor está contenida en la extensión espacial del segundo valor, probablemente indique la existencia de una relación de subclase entre las clases, es decir, la clase correspondiente al primer conjunto de datos es un subconjunto de la clase del segundo conjunto de datos. Para lograr este objetivo se definieron dos medidas de similitud asimétricas que miden el grado de pertenencia de una extensión espacial en otra, y viceversa. En caso de que se cumpla que ambas extensiones espaciales estén solapadas, se podrán considerar como equivalentes. Este algoritmo, a diferencia de Duckham y Worboys [109], permite clasificar los elementos de dos conjuntos de datos usando diferentes relaciones (equivalencia y pertenencia) de los conjuntos de datos.

3.4 Propuesta para el desarrollo de un futuro método de alineamiento de geo-ontologías

Las técnicas para el alineamiento de ontologías han ido evolucionando desde técnicas para tratar ontologías comunes hasta técnicas específicas para un dominio específico, como el geográfico, hasta el punto de llegar a definirse el término de geo-ontología. Como se ha podido apreciar en el estado del arte, las investigaciones realizadas en el campo de alineamientos de geo-ontologías están en su infancia. Todavía no existe una metodología bien definida para realizar el alineamiento. En estos pocos años, los algoritmos desarrollados se han basado en distintas estrategias. Los métodos estructurales usan medidas de similitud que consideran la estructura taxonómica de las ontologías. Los métodos terminológicos usan similitudes entre etiquetas de nombres de las entidades y pueden usar recursos externos como los tesauros y considerar relaciones como la sinonimia. Los métodos extensionales están basados en la similitud entre los individuos (instancias) de las ontologías.

Las heterogeneidades a nivel de conceptos y a nivel taxonomías pueden ser manejadas por cualquier herramienta de alineamiento convencional tratadas en el tópico 2.6. En estos niveles, las características de información geográfica no influyen en el cálculo de las similitudes. Estos comparadores no realizan la comparación a nivel de instancias. Las instancias en los datos geográficos son las que contienen la información específica de un objeto geográfico (río, montaña, ciudad). Para manejar la heterogeneidad a nivel de instancias es necesario considerar herramientas que utilicen algoritmos que trabajen con las instancias.

De las propuestas de métodos de alineamientos revisados en este reporte, no hemos hallado una que establezca una forma de asociar las similitudes obtenidas utilizando técnicas de alineamiento de ontologías convencionales con técnicas de alineamiento de geo-ontologías, en particular, las técnicas que toman en cuenta las instancias, que consideran la información y las características propias de este dominio.

Otro aspecto que no ha sido tratado en la revisión bibliográfica es el uso de las relaciones espaciales entre conceptos. Las relaciones más utilizadas por los métodos de alineamiento de ontologías son la de equivalencia y la de subsunción. Estos métodos de alineamiento generales se han visto restringidos al uso de pocas relaciones debido a que estos no pueden conocer la semántica de otros tipos de relaciones de un dominio específico. Sin embargo, la semántica implícita en las relaciones espaciales proporciona información que puede ser explotada en el proceso de alineamiento de geo-ontologías que ayuden a mejorar los resultados.

4 Conclusiones

En este documento se hizo un estudio sobre las técnicas que se han aplicado para el alineamiento de ontologías, así como la especialización de éstas para el dominio geoespacial.

El estudio de las técnicas convencionales para el alineamiento de ontologías puede servir como base para el desarrollo de métodos para alinear geo-ontologías ya que éstas son una especialización de las ontologías convencionales.

Las instancias de los conceptos de una geo-ontología son elementos importantes para el desarrollo de un método de alineamiento ya que éstas son una fuente de información con un contenido semántico que puede ser usado en el proceso de alineamiento. La distribución espacial, es decir, la posición en el espacio de un objeto, es uno de los datos geográficos de mayor interés y éste está contenido en una instancia.

Las estrategias desarrolladas principalmente para tratar las heterogeneidades se enfocan en las heterogeneidades a nivel de conceptos (analizando los términos o la estructura de la ontología) y heterogeneidades a nivel de instancias, pero no existe una estrategia que combine el análisis de ambas heterogeneidades.

Las relaciones espaciales propias del dominio semántico no han sido explotadas por las estrategias existentes. Tomar en cuenta la semántica que brindan estas relaciones entre los objetos espaciales puede servir para el desarrollo de un método que explote la información brindada por las ontologías geográficas.

En general, se ha mostrado que los trabajos que abordan la temática de alineamiento de ontologías se han comenzado a desarrollar recientemente siendo un campo de investigación joven y que no existe una estrategia bien definida. Esta afirmación también es válida para las geo-ontologías ya que como se ha planteado anteriormente, estas son una especialización de las ontologías normales al dominio geoespacial. Existe una gran variedad de métodos que analizan las ontologías sin tener en cuenta el dominio geoespacial, lo cual muestra la complejidad del tema. Para alinear geo-ontologías se pueden utilizar cualquiera de las variantes propuestas para alinear ontologías convencionales, y pueden ser ampliadas con técnicas específicas que exploten la información que nos brindan las instancias de los objetos geográficos.

Referencias bibliográficas

1. Fonseca, F.T., et al., *Using Ontologies for Integrated Geographic Information Systems.* Transactions in Geographic Information Systems, 2002. 6(3).
2. Euzenat, J. and P. Shvaiko, *Ontology Matching*. 2007, Berlin - Heidelberg - New York: Springer.
3. Ehrig, M., *Ontology Alignment: Bridging the Semantic Gap*. 2007: Springer.
4. Hess, G.N., et al. *Towards Effective Geographic Ontology Matching*. in *GeoS 2007*. 2007: Spinger-Verlag
5. Strawson, P.F. and R. Bubner, *Semantik und Ontologie*. 1975: Vandenhoeck & Ruprecht.
6. Guarino, N., *Understanding, building and using ontologies.* International Journal of Human and Computer Studies, 1997. 46(2-3): p. 293-310.
7. Gruber, T.R., *Toward Principles for the Design of Ontologies Used for Knowledge Sharing.* International Journal of Human-Computer Studies, 1995. 43(5-6): p. 907-928.
8. Studer, R., V.R. Benjamins, and F. Dieter, *Knowledge engineering principles and methods.* Data and Knowledge Engineering, 1998. 25(1-2): p. 161-197.
9. Guarino, N. *Formal Ontology and Information Systems*. in *Proceeding of FOIS '98*. 1998. Trento, Italy: IOS Press.

10. van Heijst, G., A.T. Schreiber, and B.J. Wielinga, *Using explicit ontologies in KBS development.* International Journal of Human and Computer Studies, 1997. 46(2-3): p. 183-292.
11. Stumme, G., et al., *The Karlsruhe view on ontologies*. 2003: Karlsruhe, Germany.
12. Dean, M. and G. Schreiber, *Recommendation, W3C, February 2004*, in *OWL web ontology language reference*, M. Dean and G. Schreiber, Editors. 2004.
13. Smith, M., C. Welty, and D. McGuinness, *W3C Recommendation 10 February 2004*, in *OWL Web Ontology Language Guide*, M. Smith, C. Welty, and D. McGuinness, Editors. 2004.
14. Euzenat, J. *An API for ontology alignment*. in *Proceedings of the Third International Semantic Web Conference*. 2004. Hiroshima, Japan: Springer.
15. Euzenat, J., F. Scharffe, and L. Serafini, *Knowledge Web deliverable D2.2.6:Specification of the delivery alignment format*. 2006.
16. Klein, M. *Combining and relating ontologies: an analysis of problems and solutions*. in *Proceedings of Workshop on Ontologies and Information Sharing at IJCAI-01*. 2001. Seattle, WA, USA.
17. Ding, Y., et al., *The Semantic Web:yet another hip?* Data Knowledge Engineering, 2002. 41(2-3): p. 205-227.
18. de Bruijn, J., et al., *SEKT deliverable D4.4.1: Ontology mediation management*. 2005.
19. Wiederhold, G. *An algebra for ontology composition*. in *Proceedings of 1994 Monterey Workshop on formal Methods*. 1994. Naval Postgraduate School, Monterey, CA, USA.
20. Popa, L., et al. *Translating web data*. in *Proceedings of 28th International Conference on Very Large Data Bases (VLDB-2002)*. 2002. Hong Kong, China: Morgan Kaufmann Publishers.
21. Tverski, A., *Features of similarity.* Psychological Review, 1977. 84(2): p. 327–352.
22. Bernstein, A., et al. *How similar is it? Towards personalized similarity measures in ontologies.* in *Wirtschaftsinformatik 2005: eEconomy, eCovernment, eSociety, Siebte Internationale Tagung Wirtschaftsinformatik 2005*. 2005. Bamberg, Germany: Physica-Verlag.
23. Berners-Lee, T., *Semantic Web - XML 2000*. 2000.
24. Quillan, M.R., *Word concepts: A theory and simulation of some basic capabilities.* Behavioral Science, 1967. 12: p. 410-430.
25. Baader, F., et al., *The Description Logic Handbook*. 2003: Cambridge University Press.
26. Stojanovic, N., *Semantic Query Expansion*. 2005: Karlsruhe, Germany.
27. Shvaiko, P. and J. Euzenat, *A survey of schema-based matching approaches.* Journal on Data Semantics, 2005. IV: p. 146-171.
28. Rahm, E. and P. Bernstein, *A survey of approaches to automatic schema matching.* The VLDB Journal, 2001. 10(4): p. 334-350.
29. Kang, J. and J. Naughton. *On schema matching with opaque column names and data values*. in *Proc. 22nd International Conference on Management of Data (SIGMOD)*. 2003. San Diego (CA US).
30. Lenat, D. and R. Guha, *Building large knowledgebased systems*. 1990, MA US: Addison Wesley,Reading
31. Niles, I. and A. Pease. *Towards a standard upper ontology*. in *Proc. 2nd International Conference on Formal Ontology in Information Systems (FOIS)*. 2001. Ogunquit (ME US).
32. Gangemi, A., et al., *Sweetening WordNet with DOLCE.* AI Magazine, 2003. 24(3).
33. Aleksovski, Z., et al. *Matching unstructured vocabularies using a background ontology*. in *Proc. 15th International Conference on Knowledge Engineering and Knowledge Management (EKAW)*. 2006. Praha (CZ).
34. Rahm, E., H.-H. Do, and S. Maßmann, *Matching large XML schemas.* ACM SIGMOD Record, 2004. 33(4): p. 26–31.
35. Doan, A.-H. and A. Halevy, *Semantic integration research in the database community: A brief survey.* AI Magazine, 2005. 26(1): p. 83–94.
36. Zanobini, S., *Semantic coordination: the model and an application to schema matching*. 2006: Trento (IT).

37. Giunchiglia, F. and P. Shvaiko, *Semantic matching.* The Knowledge Engineering Review, 2003. 18(3): p. 265-280.
38. Cohen, W., P. Ravikumar, and S. Fienberg. *A comparison of string metrics for matching names and records*. in *Proc. KDD Workshop on Data Cleaning and Object Consolidation*. 2003. Washington (DC US).
39. Hamming, R., *Error detecting and error correcting codes*. 1950, Bell System Technical Journal.
40. Levenshtein, V., *Binary codes capable of correcting deletions, insertions, and reversals.* Doklady akademii nauk SSSR, 1965. 4(163): p. 845-848.
41. Jaro, M., *UNIMATCH: A record linkage system: User's manual*. 1976: Washington (DC US).
42. Jaro, M., *Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida.* Journal of the American Statistical Association, 1989. 84(406): p. 414-420.
43. Winkler, W., *The state of record linkage and current research problems.* 1999.
44. Maynard, D. and S. Ananiadou, *Term extraction using a similarity-based approach.* Recent advances in computational terminology, 2001: p. 261-278.
45. Miller, G., *WordNet: A lexical database for english.* Communications of the ACM, 1995. 38(11): p. 39-41.
46. Resnik, P. *Using information content to evaluate semantic similarity in a taxonomy*. in *Proc. 14th International Joint Conference on Artificial Intelligence (IJCAI)*. 1995. Montréal (CA).
47. Resnik, P., *Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language.* Journal of Artificial Intelligence Research, 1999(11): p. 95-130.
48. Lin, D. *An information-theoretic definition of similarity*. in *Proc. 15th International Conference of Machine Learning (ICML)*. 1998. Madison (WI US).
49. Lesk, M. *Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone*. in *Proc. 5th Annual International Conference on Systems Documentation (SIGDOC)*. 1986. Toronto (CA).
50. Valtchev, P., *Construction automatique de taxonomies pour l'aide à la représentation de connaissances par objets*. 1999: Grenoble (FR).
51. Valtchev, P. and J. Euzenat. *Dissimilarity measure for collections of objects and values*. in *Proc. 2nd Symposium on Intelligent Data Analysis (IDA)*. 1997. London (UK).
52. Rada, R., et al., *Development and application of a metric on semantic nets.* IEEE Transactions on Systems, Man and Cybernetics, 1989. 19(1): p. 17-30.
53. Barthélemy, J.-P. and A. Guénoche, *Trees and proximity representations*. 1992, Chichester (UK): John Wiley & Sons.
54. Wu, Z. and M. Palmer. *Verb semantics and lexical selection*. in *Proc. 32nd Annual Meeting of the Association for Computational Linguistics (ACL)*. 1994. Las Cruces (NM US).
55. Mädche, A. and V. Zacharias. *Clustering ontology-based metadata in the semantic web*. in *Proc. 6th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)*. 2002. Helsinki (FI).
56. Dieng, R. and S. Hug. *Comparison of "personal ontologies" represented through conceptual graphs*. in *Proc. 13th European Conference on Artificial Intelligence (ECAI)*. 1998. Brighton (UK).
57. Ehrig, M. and Y. Sure. *Ontology mapping - an integrated approach.* in *Proceedings of the First European Semantic Web Symposium (ESWS-2004)*. 2004. Heraklion,Greece: Springer.
58. Mädche, A. and S. Staab. *Measuring similarity between ontologies*. in *Proc. 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW)*. 2002. Siguenza (ES),.
59. Jaccard, P. *Distribution de la flore alpine dans le bassin des dranses et dans quelques régions voisines*. in *Bulletin de la société vaudoise des sciences naturelles*. 1901.
60. Ganter, B. and R. Wille, *Formal concept analysis: mathematical foundations*. 1999, Berlin (DE): Springer Verlag.

61. Fellegi, I. and A. Sunter, *A theory for record linkage.* Journal of the American Statistical Association, 1969. 64(328): p. 1183–1210.
62. Elfeky, M., A. Elmagarmid, and V. Verykios. *Tailor: A record linkage tool box*. in *Proc. 18th International Conference on Data Engineering (ICDE)*. 2002. San Jose (CA US).
63. Lim, E.-P., et al. *Entity identification in database integration*. in *Proc. 9th International Conference on Data Engineering (ICDE)*. 1993. Wien (AT).
64. Li, W.-S. and C. Clifton. *Semantic integration in heterogeneous databases using neural networks*. in *Proc. 10th International Conference on Very Large Data Bases (VLDB)*. 1994. Santiago (CL).
65. Hausdorff, F., *Grundzüge der Mengenlehre*. 1914, Leipzig (DE): Verlag Veit.
66. Giunchiglia, F., P. Shvaiko, and M. Yatskevich. *Discovering missing background knowledge in ontology matching*. in *Proc. 16th European Conference on Artificial Intelligence (ECAI)*. 2006. Riva del Garda (IT).
67. Bouquet, P. and L. Serafini. *On the difference between bridge rules and lifting axioms*. in *Proc. 4th International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT)*. 2003. Stanford (CA US).
68. Giunchiglia, F., P. Shvaiko, and M. Yatskevich. *S-Match: an algorithm and an implementation of semantic matching*. in *Proceedings of ESWS 2004*. 2004. Heraklion (GR).
69. Shvaiko, P., *Iterative Schema-based Semantic Matching*. 2006: Trento (IT).
70. Euzenat, J. *Semantic precision and recall for ontology alignment evaluation*. in *Proc. 20th International Joint Conference on Artificial Intelligence (IJCAI)*. 2007. Hyderabad (IN).
71. Shvaiko, P., *Iterative schema-based semantic matching*. 2004.
72. Bouquet, P., et al. *Bootstrapping semantics on the web: meaning elicitation from schemas*. in *Proc. 15th International World Wide Web Conference (WWW)*. 2006. Edinburgh (UK).
73. Fürst, F. and F. Trichet, *Axiom-based ontology matching: a method and an experiment*. 2005.
74. Do, H.-H. and E. Rahm. *COMA - a system for flexible combination of schema matching approaches*. in *Proceedings of 28th International Conference on Very Large Data Bases (VLDB-2002)*. 2002. Hong Kong, China: Morgan Kaufmann Publishers.
75. Berkovsky, S., Y. Eytani, and A. Gal. *Measuring the relative performance of schema matchers*. in *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2005)*. 2005. Compeigne, France: IEEE Computer Society.
76. Melnik, S., H. Garcia-Molina, and E. Rahm. *Similarity flooding: A versatile graph matching algorithm and its application toschema matching*. in *Proceedings of the 18th International Conference on Data Engineering (ICDE-2002)*. 2002: IEEE Computer Society.
77. van Rijsbergen, C.J., *Information retrieval*. 1975, London (UK): Butterworths.
78. Egan, J.P., *Signal detection theory and ROC analysis.* Psychometrika, 1975.
79. Mitra, P., G. Wiederhold, and M. Kersten. *A graph oriented model for articulation of ontology interdependencies*. in *Proceedings of the Conference on Extending Database Technology 2000 (EDBT-00)*. 2000. Konstanz, Germany: Springer.
80. Mitra, P. and G. Wiederhold, *An ontology-composition algebra*. 2001: Stanford, CA, USA.
81. Noy, N.F. and M.A. Musen, *SMART: Automated support for ontology merging and alignment*, in *Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling, and Management*. 1999: Banff, AB, Canada.
82. Noy, N.F., et al., *Creating Semantic Web contents with Protègè-2000.* IEEE Intelligent Systems, 2001. 16(2): p. 60-71.
83. Noy, N.F. and M.A. Musen. *PROMPT: Algorithm and tool for automated ontology merging and alignment*. in *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*. 2000. Austin, TX, USA: AAAI Press / The MIT Press.
84. Noy, N.F. and M.A. Musen. *Anchor-PROMPT: Using non local context for semantic matching*. in *Workshop on Ontologies and Information Sharing at the 17th International Joint Conference on Artificial Intelligence (IJCAI-2001)*. 2001. Seattle, WA, USA.

85. Noy, N.F. and M.A. Musen. *PromptDiff: a fixed-point algorithm for comparing ontology versions*. in *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI-02)*. 2002. Edmonton, AB, Canada: AAAI Press.
86. Noy, N.F. and M.A. Musen, *The PROMPT suite: interactive tools for ontology merging and mapping*. International Journal of Human-Computer Studies, 2003. 59(6): p. 983-1024.
87. McGuinness, D.L., et al. *The Chimaera ontology environment*. in *Proceedings of the 17th National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*. 2000. Austin, TX, USA: AAAI Press / The MIT Press.
88. Stumme, G. and A. Maedche. *FCA-Merge: Bottom-up merging of ontologies*. in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2001)*. 2001. Seattle, WA, USA.
89. Doan, A., et al., *Learning to map between ontologies on the Semantic Web.* www (2002), 2002: p. 662-673.
90. Doan, A., P. Domingos, and A. Halevy, *Learning to match the schemas of data sources: A multistrategy approach.* VLDB Journal, 2003. 50: p. 279-301.
91. Euzenat, J. and P. Valtchev. *Similarity-based ontology alignment in OWL-Lite*. in *Proceedings of the 16th Eureopean Conference on Artificial Intelligence (ECAI 2004)*. 2004. Valencia, Spain: lOS Press.
92. Clifton, C., E. Housman, and A. Rosenthal. *Experience with a combined approach to attribute-matching across heterogeneous databases*. in *Data Mining and Reverse Engineering: Searching for Semantics, Seventh Conference on Database Semantics (DS-7)*. 1997. Leysin, Switzerland: Chapman and Hall.
93. Palopoli, L., G. Terracina, and D. Ursino. *The system DIKE: Towards the semi-automatic synthesis of cooperative information systems and data warehouses*. in *Symposium on Advances in Databases and Information Systems (ADBIS-DASFAA)*. 2000. Prague, Czech Republic: Matfyzpress.
94. Bergamaschi, S., et al., *Semantic Integration of Heterogeneous Information Sources.* Special Issue on Intelligent Information Integration, Journal of Data and Knowledge Engineering, 2001. 36(1): p. 215-249.
95. Madhavan, J., P.A. Bernstein, and E. Rahm. *Generic schema matching with Cupid*. in *Proceedings of the 27th International Conference on Very Large Data Bases (VLDB-01)*. 2001. San Francisco, CA, USA: Morgan Kaufmann Publishers.
96. Melnik, S., E. Rahm, and P.A. Bernstein. *Rondo: A programming plataform for generic model management*. in *Proceedings of the 2003 A CM SIGMOD International Conference on Management of Data (SIGMOD-03)*. 2003. San Diego, CA, USA: ACM Press.
97. Hu, W., et al. *GMO: A graph matching for ontologies*. in *Proceedings of the Workshop on Integrating Ontologies at K-Cap 2005*. 2005. Banff,AB, Canada: CEUR-WS Publication.
98. Shvaiko, P., et al. *Web explanations for semantic heterogeneity discovery*. in *Second European Semantic Web Conference (ESWC-2005)*. 2005. Heraklion, Greece: Springer.
99. Egenhofer, M.J. and R.D. Franzosa, *Point set topological relation.* International Journal of Geographical Information Systems, 1991. 5: p. 161-174.
100. Rodríguez, M.A., M.J. Egenhofer, and R.D. Rugg. *Assessing Semantic Similarities among Geospatial Feature Class Definitions*. in *Interoperating Geographic Information Systems, Second International Conference, Interop '99*. 1999. Zurich, Switzerland: Springer-Verlag.
101. Rodriguez, M.A. and M.J. Egenhofer, *Determining semantic similarity among entity classes from different ontologies.* IEEE Transactions on Knowledge and Data Engineering, 2003. 15(2): p. 442-456.
102. Hakimpour, F. and S. Timpf, *Using Ontologies for resolution of Semantic Heterogeneity in GIS*, in *4th AGILE Conference on Geographic Information Science*. 2001: Brno, Czech Republic.
103. Hakimpour, F. and A. Geppert. *Global schema generation using formal ontologies*. in *Proceedings of the 21st International Conference on Conceptual Modeling*. 2002: Springer.

104. Uitermark, H., *Ontology-based geographic data set integration*. 2001, Universiteit Twente: Deventer, The Netherlands.
105. Kavouras, M. and M. Kokla, *A method for the formalization and integration of geographical categorizations.* International Journal of Geographical Information Science, 2002. 16(5): p. 439-453.
106. Sotnykova, A., et al., *Semantic mappings in description logics for spatio-temporal database schema integration.* Journal on Data Semantics, 2005. III: p. 143-167.
107. Schwering, A. and M. Raubal. *Measuring Semantic Similarity Between Geospatial Conceptual Regions*. in *GeoS 2005*. 2005: Springer, Heidelberg.
108. Gärdenfors, P., *Conceptual Spaces: The Geometry of Thought*. 2000, Cambridge, Massachusetts, USA: MIT Press.
109. Duckham, M. and M. Worboys, *An Algebraic Approach to Automated Geospatial Information Fusion.* Interantional Journal of Geographic Information Science, 2005. 19(5): p. 537-557.
110. Cruz, I.F., W. Sunna, and A. Chaudhry. *Semi-automatic ontology alignment for geospatial data integration*. in *GIScience*. 2004: Springer, Heidelberg.
111. Sunna, W. and I.F. Cruz. *Structure-Based Methods to Enhance Geospatial Ontology Alignment*. in *GeoS 2007*. 2007: Springer Verlag.
112. Navarrete, T. and J. Blat. *An algorithm for Merging Geographic Datasets Based on the Spatial Distribution of Their Values*. in *GeoS*. 2007: Springer - Verlag.

RT_010, diciembre 2009
Aprobado por el Consejo Científico CENATAV

Editor: Lic. Lucía González Bayona
Diseño de Portada: DCG Matilde Galindo Sánchez
RNPS No. 2142
ISSN 2072-6287
Indicaciones para los Autores:
Seguir la plantilla que aparece en www.cenatav.co.cu
C E N A T A V
7ma. No. 21812 e/218 y 222, Rpto. Siboney, Playa;
Ciudad de La Habana. Cuba. C.P. 12200
Impreso en Cuba