UNIVERSIDAD
NACIONAL
DE LA PLATA

Facultad de Informática
Tesis Doctoral

Integración de métodos de descubrimiento de conocimiento embebido en fuentes de información desestructuradas

Autor

Juan M. Rodríguez

Directores

Patricia Pesado
Rodolfo Bertone

Asesor

Hernán Merlino

UNIVERSIDAD NACIONAL DE LA PLATA

Abstract

Facultad de Informática

Doctorado en Ciencias Informáticas

Integration of embedded knowledge discovery methods in unstructured information sources

by Juan Manuel Rodríguez

Existing Open Information Extraction methods have considerably low precision and recall, around 60%, and although they have great potential to be used in applications, their performance needs to be improved. There are also other open problems that are being addressed by different authors such as: the extraction of non-informative semantic relationships, the extraction of subjective information and the support for languages other than English.

The main contribution of this thesis consists in the publication of 3 new methods of Open Information Extraction, one for the English language: ATP-OIE and two for the Spanish language: TP-OIE-ES and ECMes. Also a reference framework is proposed for the evaluation of the methods, that is, the construction of a test dataset and a precise definition of the metrics to be used and how to implement them.

ATP-OIE is an autonomous algorithm, able to learn from examples and able to learn new extraction patterns while running productively. TP-OIE-ES replicates the behavior of ATP-OIE for the Spanish language, with the exception that it is not capable of learning new patterns while it runs productively. Finally ECMes is a re-trained version of TP-OIE-ES with additional improvements. ECMes got a better performance in Spanish language, in the evaluated datasets, than other similar methods in the state of the art.

UNIVERSIDAD NACIONAL DE LA PLATA

Resumen

Facultad de Informática

Doctorado en Ciencias Informáticas

Integración de métodos de descubrimiento de conocimiento embebido en fuentes de información desestructuradas

por Juan Manuel Rodríguez

Los métodos existentes de extracción de conocimiento para la Web (*Open Information Extraction*) tienen una precisión y una exhaustividad considerablemente baja, de alrededor del 60% y si bien tienen un gran potencial en cuanto a su aplicabilidad, es necesario mejorar su desempeño. Existen además otros problemas abiertos que están siendo abordados por varios autores como por ejemplo: la extracción de relaciones semánticas no informativas, la extracción de información subjetiva y el soporte para idiomas distintos del inglés.

El principal aporte de esta tesis consiste en la publicación de 3 métodos novedosos de extracción de conocimiento para la Web, uno para idioma inglés: ATP-OIE y dos para idioma español: TP-OIE-ES y ECMes. Así mismo, se propone un marco de referencia único para la evaluación de los métodos, esto es la construcción de un conjunto de pruebas y una definición precisa de las métricas a utilizar y de cómo implementarlas.

ATP-OIE es un algoritmo autónomo, capaz de aprender de ejemplos y capaz de aprender nuevos patrones de extracción mientras se está ejecutando de forma productiva. Por su parte, TP-OIE-ES replica el comportamiento de ATP-OIE para idioma español, con la salvedad de que no es capaz de aprender nuevos patrones mientras se ejecuta de forma productiva. Por último, ECMes es una versión reentrenada de TP-OIE-ES con otras mejoras adicionales. ECMes ha obtenido un mejor desempeño en idioma español, en los conjuntos evaluados, que otros métodos similares en el estado del arte.

Dedicatoria

A Estefanía mi compañera y amiga

A mis padres, Mónica y Juan

A mi hermana Jimena

A Leandro mi hijo, la razón de todo

Agradecimientos

Quisiera agradecer primeramente a Hernán Merlino, mi director de tesis y amigo, por toda la confianza que siempre me tuvo desde el primer momento y por ayudarme a concluir este proyecto de investigación.

También a Patricia Pesado directora principal de esta tesis y por supuesto a Ramón García–Martínez que a pesar de no estar más entre nosotros fue el motor inicial que permitió a este trabajo despegar y tomar vuelo.

Publicaciones

Parte del trabajo realizado en esta tesis ha sido presentado en diversos congresos y publicaciones en revistas nacionales e internacionales:

Publicaciones en revistas internacionales:

Mejoras aplicadas a la extracción de relaciones semánticas para la Web en español. Rodríguez, J.M., Merlino, H., Pesado, P.; Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN), número 66, pág. 133-140, 2021.

Presentaciones en congresos internacionales:

ATP-OIE: An Autonomous Open Information Extraction Method. Rodríguez, J.M., Merlino, H., Patricia, P.; Presentado en International Conference On Compute And Data Analysis (iccda 2020).

TP-OIE-ES: Método autónomo de extracción de relaciones semánticas para la Web en Español. Rodríguez, J.M., Merlino, H.; Presentado en Conferencia Iberoamericana de Complejidad, Informática y Cibernética: CICIC 2020.

Evaluation of open information extraction methods using Reuters-21578 database. Rodríguez, J.M., Merlino, H.D., Pesado, P., García-Martínez, R.; Presentado en 2nd International Conference on Machine Learning and Soft Computing (ICMLSC 2018)

Performance evaluation of knowledge extraction methods. Rodríguez, J.M., Merlino, H., Pesado, P., García-Martínez, R.; Presentado en International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems. IEA/AIE 2016.

Presentaciones en congresos nacionales:

Automatización de la extracción de características en tareas de análisis de sentimiento. Rodríguez, J.M., Merlino, H.D., Pesado, P., García-Martínez, R.; Presentado en XXIII Congreso Argentino de Ciencias de la Computación (CACIC 2017).

Clasificación de Distintos Conjuntos de Datos Utilizados en Evaluación de Métodos de Extracción de Conocimiento Creados para la Web. Rodríguez, J.M.,

Merlino, H.D., Pesado, P., García-Martínez, R.; Presentado en XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016).

Presentaciones en simposios:

Integración de métodos de descubrimiento de conocimiento embebido en fuentes de información desestructuradas. Rodríguez, J.M., Merlino, H.D; Presentado en El Simposio Argentino de Inteligencia Artificial (ASAI) 2020, parte de las Jornadas Argentinas de Informática (JAIIOs 49).

Publicaciones relacionadas

Además de las publicaciones realizadas durante el desarrollo de la tesis, se publicó un estudio preliminar en el año 2015 en el congreso nacional CACIC, que sirvió como punto de partida para el armado de la propuesta:

Revisión Sistemática Comparativa de Evolución de Métodos de Extracción de Conocimiento para la Web. Rodríguez, J.M., Merlino, H., García-Martínez, R.; Presentado en: XXI Congreso Argentino de Ciencias de la Computación (CACIC 2015).

Índice general

Índice de figuras

Índice de tablas

Glosario

árbol(es) de decisión: se trata de un modelo de clasificación, generalmente generado por un algoritmo de aprendizaje automático, que permite decidir a que clase o categoría pertenece una observación dada, haciendo preguntas sobre sus atributos. Dependiendo de las respuestas se va recorriendo el árbol (el cual es un tipo de grafo), desde el nodo raíz hasta algún nodo hoja, el cual contendrá la respuesta o categoría buscada. Genéricamente se mencionan como "árboles de decisión" a un conjunto de algoritmos que son capaces de generar este tipo de árboles a partir de datos de ejemplo.

árbol de dependencias sintácticas: existe un tipo de análisis sintáctico que descompone una oración dada en sus unigramas constituyentes, coloquialmente sus palabras, encuentra luego la categoría gramatical de cada unigrama (es decir, identifica si es un verbo, sustantivo, adjetivo, etc.) y las relaciones que existen entre estas. Estas relaciones se llaman relaciones sintácticas y son del tipo: "es determinante de" o "es el pivote de la frase nominal", etc. El resultado de este análisis es el árbol de dependencias sintácticas, el cual, partiendo de una palabra raíz, y utilizando las relaciones sintácticas como aristas, va conectando cada una de las palabras en la oración.

array: en programación un *array* o colección es un tipo de variable que permite almacenar N elementos de forma consecutiva y acceder a cualquiera de ellos utilizando su indice. Puede ser unidimensional, en dicho caso solo habrá un indice que irá desde 0 a N-1, siendo N la cantidad de elementos en el a*rray*, o bien multidimensional y en ese caso habrá que establecer un indice con varios dígitos, uno por cada dimensión.

ATP-OIE: acrónimo de *Autonomous Tree Pattern Open Information Extraction*, uno de los métodos de extracción de relaciones semánticas para la Web propuesto en este trabajo. Se caracteriza por su habilidad para aprender nuevos casos, mientras se ejecuta de forma productiva.

AUC: acrónimo de *Area Under Curve*, es una métrica para estimar la eficacia de un clasificador y consiste en el calculo del área bajo la curva ROC (otra métrica, ver más abajo).

auto-supervisados, métodos: se trata de un tipo de algoritmo utilizado en aprendizaje automático para estimar un valor, capaz de aprender a partir de ejemplos (llamado conjunto de entrenamiento), con la diferencia respecto de otros métodos, de que es capaz de generar o ampliar el conjunto de entrenamiento de forma autónoma, es decir sin intervención humana.

automatizable, proceso: cualquier algoritmo capaz de ser ejecutado en una computadora. Software.

Bayes naïve: algoritmo de clasificación basado en el calculo de la probabilidad condicional.

BERT: acrónimo de *Bidirectional Encoder Representations from Transformers*, es una tecnología basada en redes neuronales preentrenadas utilizada y creada para tareas de procesamiento de lenguaje natural. Fue desarrollada por Google.

bigrama: un bigrama o digrama es un grupo de dos letras, dos sílabas, o dos palabras. Los bigramas son utilizados comúnmente como base para el análisis estadístico de texto

C4.5: es un algoritmo usado para generar un árbol de decisión desarrollado por Ross Quinlan. Es un algoritmo que mejora a uno anterior llamado ID3.

ClausIE: algoritmo de extracción de conocimiento para la Web (*Open IE*).

clustering: es una tarea de aprendizaje automático no supervisada. Implica descubrir automáticamente la agrupación natural de los datos de entrada. A diferencia del aprendizaje supervisado, los algoritmos de agrupación o *clustering* sólo interpretan los datos de entrada para encontrar grupos o agrupaciones naturales en el espacio de características.

conjunciones coordinantes: son aquellas conjunciones que unen palabras, frases u oraciones, que tienen el mismo nivel jerárquico, o sea, que realizan la misma función o pertenecen a la misma categoría gramatical. Por ejemplo la palabra “y” que une dos oraciones similares.

conocimiento embebido: se trata de conocimiento (es decir, el derivado último de la información) plasmado en alguna fuente, por ejemplo un documento de texto, pero que no es directamente accesible por un software o sistema infor-

mático. Por ejemplo, un libro de recetas de cocina tiene el conocimiento necesario para preparar cientos de platillos distintos. Este conocimiento está embebido en el libro y está directamente disponible para aquellas personas que sepan leer el lenguaje en el cual está escrito el libro. Pero no es un conocimiento accesible para una máquina. Ésta, previamente, deberá realizar algún proceso de análisis, manipulación y refinamiento de los datos (ya sea de forma manual o automática) para poder acceder a dicho conocimiento.

DARPA: acrónimo de *Defense Advanced Research Projects Agency*, (Agencia de Proyectos Avanzados de Defensa), la cual es una agencia del Departamento de Defensa de Estados Unidos responsable del desarrollo de nuevas tecnologías para uso militar

dependencias sintácticas: en el modelo teórico del lenguaje propuesto por el lingüista francés *Lucien Tesnière*, los elementos léxicos, que conforman la estructura sintáctica de la oración, están relacionados entre sí. De acuerdo con esta teoría, existe un vínculo que relaciona las ideas expresadas por las palabras para formar un pensamiento organizado. Estas relaciones conllevan un orden jerárquico lo que implica dependencias entre las diversas palabras, siendo el verbo el elemento con mayor jerarquía. Estas relaciones se llaman relaciones o dependencias sintácticas y son del tipo: “es determinante de...” o “es el pivote de la frase nominal...”, etc. Conjuntamente forman el árbol de dependencias sintácticas.

Depparse: es un algoritmo de análisis sintáctico basado en redes neuronales profundas, el cual forma parte de la biblioteca *Stanford CoreNLP*. Genera, entre otras cosas el árbol de dependencias sintácticas de una oración.

desestructuradas, fuentes de información: fuentes de información creadas para ser consumidas por seres humanos y no necesariamente por máquinas, como por ejemplo: un libro, una película o una foto. Cualquier información allí contenida, es decir embebida, deberá ser extraída previamente y presentada de una forma estructurada para que un sistema informático sea capaz de manipularla.

ECMes: acrónimo de *Extractor de Conocimiento Mejorado en Español*, uno de los métodos de extracción de relaciones semánticas para la Web propuesto en este trabajo. Se caracteriza por trabajar en idioma español.

entity1: primera parte de una relación semántica, en ocasiones se trata de un nombre propio (una entidad), pero más genéricamente es una frase nominal. Esta entidad o frase nominal está vinculada (en la oración) con la *entity2* (en una relación ternaria) mediante una relación denominada: *relation*. Se expresan las tres como la tupla: (*entity1*, *relation*, *entity2*).

entity2: tercera parte de una relación semántica ternaria, en ocasiones se trata de un nombre propio (una entidad), pero más genéricamente es una frase nominal. Esta entidad o frase nominal está vinculada (en la oración) con la *entity1* mediante una relación denominada: *relation*. Se expresan las tres como la tupla: (*entity1*, *relation*, *entity2*).

estructurada, información: información que está disponible para ser manipulada por un sistema informático ordinario, por ejemplo registros en una tabla en una base de datos relacional.

estructurados, datos: datos que están disponibles para ser manipulados por un sistema informático ordinario, por ejemplo registros en una tabla en una base de datos relacional. En este caso información estructurada o datos estructurados se han utilizado como términos intercambiables.

exhaustividad: traducción al español de la métrica llamada: *recall*, la cual se calcula como los verdaderos positivos sobre la suma de los verdaderos positivos y los falsos negativos.

expresiones regulares: secuencia de caracteres que conforma un patrón de búsqueda. Se utilizan principalmente para la búsqueda de patrones en cadenas de caracteres u operaciones de sustitución.

F1, medida: la medida F, es una métrica que combina precisión y *recall* en un sólo valor. En particular cuando toma un parámetro interno llamado beta, igual a 1, ambas métricas (precisión y *recall*) son evaluadas con la misma ponderación. En este caso hablamos de F1, F1-*meassure* o medida F1.

GPT-3: acrónimo de *Generative Pre-trained Transformer 3*, un modelo de lenguaje autorregresivo que emplea aprendizaje profundo para tareas de PLN, principalmente para la producción de textos que simulan la redacción humana.

gramaticales, categorías: es una forma de clasificar las palabras según su tipo. Modernamente el término "categoría gramatical" se refiere a una variable lingüística que puede tomar diferentes valores que condicionan la forma morfológica concreta de una palabra. La gramática tradicional castellana distingue nueve categorías gramaticales: Sustantivo, Adjetivo, Artículo, Pronombre, Verbo, Adverbio, Interjección, Preposición y Conjunción. Sin embargo en los últimos años se han aunado esfuerzos por crear categorías gramaticales universales, estas últimas son las que se utilizan en los modernos trabajos de PLN. Más información en: universaldependencies.org

informativas, relaciones semánticas: una relación semántica es informativa cuando captura la esencia de lo que la oración intenta transmitir. Por ejemplo, asumiendo la oración: "Albert Einstein, quien nació en Ulm, ganó el Premio Nobel", se podrían generar dos relaciones semánticas validas: (Albert Einstein, ganó, Premio Nobel) y (Albert Einstein, nació, Ulm). En este ejemplo, la primera relación semántica es informativa y la segunda no, o su aporte de información es ínfimo. En el apartado 5.2 hay una descripción más detallada.

IOB: técnica para etiquetar un conjunto de datos, particularmente oraciones, intentando respetar el orden de las palabras. Utilizada generalmente en algoritmos de detección de nombres de entidades (NER).

J48: J48 es una implementación *open source* en lenguaje de programación Java del algoritmo C4.5 en la herramienta *Weka* de minería de datos.

JSON: acrónimo de *JavaScript Object Notation*, es un formato de texto sencillo para el intercambio de datos entre sistemas.

JSoup: es una biblioteca Java de código abierto diseñada para analizar, extraer y manipular datos almacenados en documentos HTML y XML.

Kaggle: sitio web que funciona como un repositorio *online* de diversos conjuntos de datos, muchos de ellos creados con la finalidad de ser utilizados en minería de datos. Kaggle.com

Medida-F: ver "F1, medida".

Medida-F1: ver "F1, medida".

MinIE: método de extracción de relaciones semánticas para la Web (*Open IE*), sucesor de ClausIE.

NP-Chunking: es una técnica de análisis sintáctico que consiste en dividir una oración en frases, sin que haya solapamiento. Este tipo de técnica es llamada también análisis sintáctico superficial ya que no genera un árbol sino que las palabras las agrupa usando sólo aquellas que son consecutivas.

OIE: acrónimo de *Open Information Extraction*, lo que se traduce como extracción de conocimiento para la Web o extracción de relaciones semánticas para la Web.

OLLIE: método de extracción de relaciones semánticas para la Web (*Open IE*), sucesor de ReVerb.

Open IE: *Open Information Extraction.* Ver *OIE*.

Parser: algoritmo de análisis sintáctico.

Penn Treebank: es un corpus lingüístico en idioma inglés en el que cada frase fue analizada, es decir anotada con su estructura sintáctica, y en donde cada palabra fue etiquetada con su categoría gramatical. Las categorías gramaticales utilizadas en *Penn Treebank* no se corresponden a las categorías gramaticales universales, ya que es un trabajo realizado en los años 1989 a 1996, sin embargo el corpus es y fue ampliamente utilizado para crear herramientas y modelos lingüísticos, por lo que es bastante común que muchos algoritmos de PLN en idioma inglés trabajen con estas categorías gramaticales.

Pos-tags: acrónimo de ***Part-of-Speech tags***, se refiere a las etiquetas gramaticales para el idioma inglés. Es decir: Verbo (verb), Sustantivo (noun), etc.

precision: o **precisión** en castellano, es una métrica utilizada para medir el desempeño de un algoritmo predictivo. Se calcula como los verdaderos positivos sobre la suma de los verdaderos positivos más los falsos positivos.

Pregunta-respuesta: sistema informático para interactuar con el usuario a través de una interfaz de texto utilizando diálogos, en donde el usuario siempre formula una pregunta que es respondida, en la medida de lo posible, por el sistema informático.

recall: ver exhaustividad.

redes neuronales: el término hace referencia, concretamente, a las redes neuronales artificiales. Éstas son un modelo computacional, que busca imitar el comportamiento del cerebro humano para realizar tareas complejas, como el reconocimiento de patrones. Estas redes están constituidas por varias unidades, llamadas neuronas artificiales, las cuales están interconectadas entre sí y cada una realiza un cómputo relativamente pequeño.

relaciones semánticas: se llaman de este modo a las relaciones que existen entre dos elementos con significado en una oración, típicamente la relación entre dos entidades. Por ejemplo, asumiendo la oración: "Albert Einstein ganó el Premio Nobel", tenemos dos elementos con significado, dos entidades: "Albert Einstein" y "Premio Nobel", en este caso ambos son nombres propios. Estos dos elementos están conectados por la relación "ganó". Albert Einstein "ganó" y el Premio Nobel "fue ganado". Se llama a este tipo de relación, relación semántica.

relation: segunda parte de una relación semántica ternaria, en ocasiones se trata de un verbo o frase verbal. La *relation*, vincula a la *entity1* con la *entity2*. Se expresan las tres como la tupla: (*entity1, relation, entity2*). Es la relación propiamente dicha.

Reuters-103: subconjunto de pruebas creado con 103 cables de noticias tomados al azar de la base de datos Reuters-21578.

Reuters-21578: base de datos con textos de cables de noticias en idioma inglés.

Reuters-55: subconjunto de pruebas creado con 55 cables de noticias tomados al azar de la base de datos Reuters-21578.

Reuters: agencia de noticias con sede en el Reino Unido, conocida por suministrar información a medios de comunicación y mercados financieros.

ReVerb: método de extracción de relaciones semánticas para la Web (*Open IE*), sucesor de *TextRunner*.

ROC: acrónimo de *Receiver Operating Characteristic* (Característica Operativa del Receptor), es la representación de la razón o proporción de verdaderos

positivos frente a la razón o proporción de falsos positivos según varía el umbral de discriminación

semisupervisado: es una técnica de aprendizaje automático que utiliza datos de entrenamiento tanto etiquetados como no etiquetados. Normalmente una pequeña cantidad de datos etiquetados junto a una gran cantidad de datos no etiquetados.

SMO: es la sigla de *Sequential Minimal Optimization*, y es una implementación *open source* en lenguaje de programación Java del algoritmo SVM en la herramienta *Weka* de minería de datos.

Stanford CoreNLP: biblioteca para el lenguaje Java que contiene la implementación de diversos algoritmos para el procesamiento de lenguaje natural.

supervisados, métodos: técnicas para crear modelos predictivos a partir de datos de entrenamiento utilizados como ejemplos. Los datos de entrenamiento consisten de pares de objetos: una componente del par son los datos de entrada y el otro, los resultados deseados.

SVM: sigla de *Support-Vector Machines*, son un conjunto de algoritmos de aprendizaje supervisado desarrollados por *Vladimir Vapnik* y su equipo en los laboratorios AT&T.

Text-chunking: ver *NP-Chunking.*

TP-OIE-ES: acrónimo de *Tree Pattern Open Information Extraction Español*, es uno de los métodos de extracción de relaciones semánticas para la Web propuesto en este trabajo.

TP-OIE: acrónimo de *Tree Pattern Open Information Extraction*, es uno de los métodos de extracción de relaciones semánticas para la Web propuesto en este trabajo y es también el sistema base que sirvió para construir métodos derivados que lo superaron

trigramas: grupo de tres letras, tres sílabas, o tres palabras. Los trigramas son utilizados comúnmente como base para el análisis estadístico de texto.

tupla: es una lista o secuencia ordenada y finita de elementos. En general se utiliza este termino para referir la salida de un método de extracción de relaciones

semánticas, ya que las relaciones semánticas son representadas como una tupla de 3 elementos con la forma: (elemento 1, elemento 2, elemento 3). En donde los elementos 1 y 2 son entidades o argumentos y el elemento 2 es la relación propiamente dicha.

unigrama: elemento singular, tal que, una secuencia de ellos constituye una oración o frase en lenguaje natural, coloquialmente una palabra. Los unigramas se obtienen al dividir una oración por los caracteres de espacio, salto de línea o bien por los signos de puntuación.

1 Objetivos

En esta sección se detallan los objetivos primarios y secundarios de este trabajo de investigación. Tal y como se definió en el plan de tesis y anteproyecto, los objetivos principales son los de construir una familia de métodos de extracción de conocimiento tal que dada una estructura de información inicial como entrada, que contenga conocimiento embebido, entiéndase aquí un documento en lenguaje natural, sean capaces de generar un conjunto de piezas de conocimiento fácilmente manipulables por un sistema informático.

1.1 Objetivo primario

Dentro de esta familia de métodos de extracción de conocimiento que se pretende crear, hay un objetivo primario que es el de construir un método de extracción de conocimiento para la Web que soporte idioma español y que sea capaz de extraer piezas de información con la misma efectividad con la que otros métodos en el estado del arte lo hacen para idioma inglés.

1.2 Objetivos secundarios

Uno de los objetivos secundarios es proponer un marco de referencia para la evaluación de los métodos de *Open Information Extraction* existentes para idioma inglés y español, que permita medir el desempeño de estos con criterios similares. Se trata de crear un conjunto de pruebas o conjunto de evaluación para estos métodos, con resultados verificados de forma manual, y una definición precisa de las métricas a utilizar y de como calcularlas. Este marco de referencia podrá luego ser utilizado en trabajos futuros o ampliado, llegado el caso, para nuevos trabajos.

Un segundo objetivo es construir un método de *Open Information Extraction* novedoso, capaz de desempeñarse de forma similar a los métodos utilizados y considerados en el estado del arte. Este método deberá funcionar en idioma inglés y proponer alguna técnica original que le permita diferenciarse de otros existentes, buscando fortalecer puntos débiles detectados en métodos actuales.

2 Introducción

Desde el año 2000 aproximadamente la Web se ha convertido en un repositorio emergente de conocimiento embebido y este crecimiento continúa su marcha de forma exponencial. La necesidad de explotar estos conocimientos ha servido para recuperar la tradición de computación cognitiva (*cognitive computing*) pero en un nuevo contexto de grandes datos. En este nuevo contexto aparecen necesidades específicas dentro de las técnicas de extracción de conocimiento como lo son las técnicas de extracción de relaciones semánticas para grandes volúmenes de datos no estructurados, en particular en lenguaje natural [Rodríguez et al., 2015].

2.1 Extracción de conocimiento

La extracción de conocimiento es cualquier técnica mediante la cual un proceso automatizable es capaz de analizar fuentes de información no estructurada, como es el caso de textos escritos en lenguaje natural y extraer el conocimiento allí embebido para representarlo de una manera estructurada, manipulable en procesos de razonamiento automático, como por ejemplo: una regla de producción o un subgrafo en una red semántica. A la información obtenida como salida de este tipo de procesos se la llama pieza de conocimiento [García-Martínez y Britos, 2004; Gómez et al., 1997]. Si se piensa a la extracción de conocimiento como una transformación algebraica podría plantearse:

$$extracción\ de\ conocimiento\ (datos) = piezas\ de\ conocimiento \quad \textbf{(1)}$$

2.1.1 Surgimiento de la extracción de conocimiento

El desafío de la extracción de conocimientos comienza a fines de la década de 1970 como es señalado en [Cowie y Lehnert, 1996]. Más tarde en los años 90 la investigación fue alentada y financiada por la Agencia de Proyectos Avanzados de Defensa (DARPA) [Konstantinova, 2014].

Los métodos de extracción de conocimiento comenzaron trabajando en la detección y clasificación de nombres propios, utilizando como entrada fuentes de información no estructurada. Este tipo de extracción de conocimiento es llamado Reconocimiento de Nombres de Entidades (NER según sus siglas en inglés). En

general, estos sistemas de extracción de conocimiento buscan nombres de personas, compañías, organizaciones y lugares geográficos [Konstantinova, 2014]. El siguiente paso que dieron los métodos de extracción de conocimiento fue el de resolver correferencias y el de extraer relaciones entre nombres de entidades [Jurafsky y Martin, 2000].

Hacia finales del año 2000 los métodos de extracción de conocimiento se habían diversificado y especializado. En [Jurafsky y Martin, 2000] se reconocen distintos tipos de piezas de conocimiento susceptibles de ser extraídas: nombres de entidades, expresiones temporales, valores numéricos, relaciones entre entidades y expresiones previamente identificadas, eventos, entre otras.

La extracción de conocimiento tradicionalmente ha requerido de participación humana en la forma de reglas de extracción o bien de ejemplos de entrenamiento etiquetados de forma manual. En particular, para los casos de extracción de relaciones entre entidades, es el usuario quien debe explícitamente especificar cada relación que le interese, tarea ardua, sobre todo cuando se trabaja con fuentes heterogéneas de información no estructurada y con volúmenes de datos demasiado grandes, como podría ser la Web. Debido a ello, en general, los sistemas de extracción de conocimiento fueron utilizados sobre fuentes de información no estructurada más bien pequeñas y homogéneas [Banko et al., 2007].

2.1.2 Extracción de relaciones semánticas

Una subtarea comprendida dentro del conjunto de métodos de extracción de conocimiento es la de extraer de relaciones semánticas. En [Culotta et al., 2006] se define a la extracción de relaciones semánticas como: “la tarea de descubrir conexiones semánticas entre entidades”, y se agrega que es de uso común realizar esta tarea utilizando como entrada textos en lenguaje natural en los cuales se suele identificar primeramente grandes cantidades de pares de entidades por documento para luego determinar si existe una relación entre éstas utilizando pistas basadas en las características del lenguaje analizado.

En [Banko et al., 2008] se clasifican los métodos de extracción de relaciones semánticas en tres clases:

- métodos basados en conocimiento (*knowledge-based methods*)
- métodos supervisados (*supervised methods*)
- métodos auto-supervisados (*self-supervised methods*)

2.1.2.1 Métodos basados en conocimiento

Los primeros sistemas de extracción de relaciones eran específicos para un dominio, por ejemplo en 1991 DARPA desafió a la comunidad que estaba trabajando en procesamiento de lenguaje natural a "construir sistemas robustos capaces de llenar plantillas con piezas de conocimiento sobre el terrorismo en América Latina". Los campos requeridos eran: fechas, ubicaciones, perpetradores, armas, víctimas y objetivos físicos. Más adelante los dominios fueron cambiando y se centraron en *joint ventures*, microelectrónica y planes para la sucesión de gestiones empresariales.

Este tipo de sistemas estaban basados en reglas de coincidencia de patrones (*pattern-matching*) creadas a mano para cada dominio. Estos sistemas tenían la desventaja de no ser escalables ni portables entre dominios diferentes [Banko et al., 2008].

2.1.2.2 Métodos supervisados

Este tipo de métodos trabaja con un conjunto de datos de entrenamiento en donde ciertos ejemplos específicos, para un dominio de interés, son previamente etiquetados. Luego se utilizan dichos ejemplos para entrenar un extractor de forma automática. La principal contra de este tipo de métodos radica en el tiempo y el esfuerzo que se requiere para construir el conjunto de datos de entrenamiento [Konstantinova, 2014].

2.1.2.3 Métodos auto-supervisados

En 2005 Oren Etzioni en [Etzioni et al., 2005] presenta un método de extracción de relaciones semánticas llamado KNOWITALL, el cual es capaz de aprender a etiquetar sus propios ejemplos de entrenamiento utilizando sólo un conjunto pequeño de patrones de extracción, independientes de cualquier dominio. Este fue el primer sistema publicado capaz de encarar la extracción de conocimiento de páginas Web ya que era no supervisado, independiente del dominio y escalable [Banko et al., 2008; Etzioni et al., 2005]. Los métodos estudiados y desarrollados en la presente tesis pertenecen a esta última categoría.

2.1.3 Métodos de extracción de conocimiento para la Web (*OIE*)

En el año 2007 Michele Banko introduce un nuevo concepto en materia de extracción de conocimiento, al que llama en inglés: *Open Information Extraction*, abreviado muchas veces como *Open IE* o simplemente como *OIE* y al que nos

referiremos en español como: extracción de conocimiento para la Web o de forma más especifica: extracción de relaciones semánticas para la Web. Se trata de un paradigma de extracción de conocimiento en donde un sistema informático realiza una sola pasada sobre el total de las fuentes de información no estructurada en formato de lenguaje natural (llamado corpus de documentos), dadas como entrada y extrae un gran conjunto de tuplas relacionales sin requerir ningún tipo de intervención humana. Cabe aclarar que este paradigma de extracción de conocimiento pertenece a la clase de métodos auto-supervisados. En el mismo trabajo Banko presenta un método llamado TEXTRUNNER, el cual es el primer método que trabaja dentro de este nuevo paradigma [Banko et al., 2007].

A partir de este trabajo se propusieron otros métodos de extracción de conocimiento bajo el paradigma que Banko llamó *Open Information Extraction* y que en español se lo puede llamar de forma más concreta: métodos de extracción de conocimiento para la Web.

Los métodos de *Open IE* devuelven tuplas que constan de 3 partes: argumento primero, relación y argumento segundo. Para ilustrar esto, considérese la oración del Ejemplo 1:

Albert Einstein, quien nació en Ulm, ganó el Premio Nobel [1]

Extrayendo las relaciones semánticas presentes en la oración y expresándolas como una tupla en la forma: “(Argumento 1, Relación, Argumento 2)” obtenemos lo siguiente:

- (Albert Einstein, *ganó el*, Premio Nobel)
- (Albert Einstein, *nació en*, Ulm)

3. Revisión sistemática de literatura

Para conocer el estado del arte en relación a los métodos de extracción de conocimiento para la Web se realizó primeramente una exploración sistemática de la bibliografía existente, utilizando como principal criterio de búsqueda, el de revisar aquellos artículos que citasen a [Banko et al., 2007]. Para esta primera etapa exploratoria se contó además con el artículo de [Konstantinova, 2014] en donde se presenta una descripción general de las principales líneas de investigación y los avances hasta ese momento en el campo de la extracción de conocimiento, prestando especial atención a los métodos pensados para la Web. El detalle de esta primera revisión de la bibliografía que abarca hasta el año 2015 se describe en la sección 3.1.

Para cubrir la brecha hasta el año 2018 se cuenta con el exhaustivo trabajo de [Glauber y Barreiro Claro, 2018]. Dicho trabajo es un estudio de mapeo sistemático sobre métodos de extracción de relaciones semánticas para la Web (*Open Information Extraction*). Los autores analizaron más de 2000 artículos extraídos de 5 bases de datos en línea diferentes, para quedarse luego con 73 estudios primarios. Este trabajo fue citado y utilizado para definir el estado del arte en tres publicaciones posteriores, en las que se presentaron respectivamente tres métodos diferentes para la extracción de conocimiento para la Web: ATP-OIE [Rodríguez et al., 2020], TP-OIE-ES [Rodríguez y Merlino, 2020] y ECMes [Rodríguez et al., 2021]. Estos tres métodos surgieron como resultado directo de esta tesis. El artículo de Glauber y Barreiro Claro se trata en detalle en la sección 3.2

Finalmente para cubrir el estado del arte hasta el año 2021, momento de redacción de esta tesis, se realizó una nueva revisión sistemática utilizando la metodología descripta en el reporte: *Guidelines for performing Systematic Literature Reviews in Software Engineering* publicado por [Keele Staffs, 2007]. En la sección 3.3 se describe en detalle esta revisión.

3.1 Revisión de la evolución de los métodos de extracción de conocimiento para la Web (2015)

Para este primer trabajo de investigación documental se establecieron las siguientes preguntas de investigación:

- ¿Cuáles son los métodos de extracción de conocimiento para la Web en el estado del arte?
- ¿Están dichos métodos disponibles de forma pública para su evaluación?
- ¿Su código fuente está disponible para entender su funcionamiento?

Para responder estas preguntas se utilizó como base de datos de consulta: Google Académico[1]. Se realizó la búsqueda de todas aquellas publicaciones científicas que citasen al artículo de [Banko et al., 2007].

Tabla 1. Métodos relevados y sus artículos originales

Método	Artículo	Disponible	Código fuente
KnowItAll	[Etzioni et al., 2005]	sí	sí
TEXTRUN-NER	[Banko et al., 2007]	sí	sí
WOE	[Wu y Weld, 2010]	no	no
SONEX	[Mesquita et al., 2010]	no	no
SRL-IE-UIUC	[Christensen et al., 2011]	no[2]	no
SRL-IE-Lund		no	no
ReVerb	[Fader et al., 2011]	sí	sí
OLLIE	[Mausam et al., 2012]	sí	sí
ClausIE	[Del Corro y Gemulla, 2013]	sí	sí
ReNoun	[Yahya et al., 2014]	no	no
TRIPLEX	[Mirrezaei et al., 2015]	no	no

El portal web Google Académico devolvió un total de 1140 resultados. Los resultados devueltos, ordenados por relevancia (según criterios del propio portal), fueron filtrados para retener sólo aquellos que presentaran un nuevo método de extracción de relaciones semánticas para la Web. De la lista anterior sólo se retuvieron 10 trabajos incluyendo el de [Banko et al., 2007]. El resumen se muestra en la Tabla 1 en donde también se añadió al método KnowItAll, anterior a TEXTRUNNER, pero sobre el cual este último está basado.

Del análisis de los artículos anteriores, y teniendo en cuenta las comparaciones entre distintos métodos que presentan los autores, se puede concluir que el mejor de los métodos es ClausIE, seguido por OLLIE y luego por ReVerb. Sin embargo, las pruebas presentadas son diversas, no todos los autores utilizan las mismas fórmulas

[1] https://scholar.google.com/

[2] Los autores convirtieron métodos existentes de etiquetamiento secuencial (*SRL*) en métodos de Open IE. Los métodos de SRL originales están disponibles, pero no las versiones modificadas

para calcular la efectividad de un algoritmo. La fórmula más comúnmente utilizada es la precisión, la cual se calcula como los casos de éxito sobre las extracciones totales, o más específicamente en estos casos, como se indica en la Fórmula 2.

$$Precisión = \frac{relaciones\ semánticas\ extraídas\ correctamente}{relaciones\ semánticas\ extraídas\ totales} \quad \textbf{(2)}$$

La segunda fórmula más utilizada, pero prácticamente en conjunto con la precisión, fue la exhaustividad (*recall* en inglés), la cual se calcula como la cantidad de casos de éxito sobre la cantidad de casos relevantes totales, o más específicamente en este ámbito como se indica en la Fórmula 3.

$$Exhaustividad = \frac{relaciones\ semánticas\ extraídas\ correctamente}{relaciones\ semanticas\ existentes\ en\ el\ texto} \quad \textbf{(3)}$$

En la mayoría de los casos la cantidad de piezas totales de conocimiento fueron etiquetadas a mano, en ocasiones por más de una persona.

Otra fórmula utilizada es la Medida-F, la cual se calcula utilizando las dos medidas anteriores más un parámetro β que indica a cuál de las dos se le da una ponderación mayor. La Fórmula 4 muestra como se calcula la Medida-F. En los artículos relevados en donde se utiliza esta medida, siempre se lo hace con β igual 1, en este trabajo se referirá a la Medida-F con β igual a 1, como Medida-F1 o simplemente F1.

$$F_\beta = \frac{\left(1+\beta^2\right) \cdot Precisión \cdot Exhaustividad}{\left(\beta^2 \cdot Precisión\right) + Exhaustividad} \quad \textbf{(4)}$$

Por último, en algunos artículos se utiliza el área bajo la curva *Receiver Operating Characteristic* (ROC) como medida de la calidad de las piezas de conocimiento extraídas. Ésta medida, abreviada muchas veces como AUC por sus siglas en inglés, se basa en una representación gráfica de la tasa de verdaderos positivos contra la tasa de falsos positivos, donde el área que encierra dicha curva es una medida de la calidad. Un área de 1 representa una calidad perfecta, significaría que el método extrajo correctamente todas las piezas de conocimiento sin extraer ninguna de más ni de menos. Un área de 0,5 representa una calidad nula, éste caso significa que el

método no logró extraer ninguna pieza de conocimiento correctamente [Bradley, 1997].

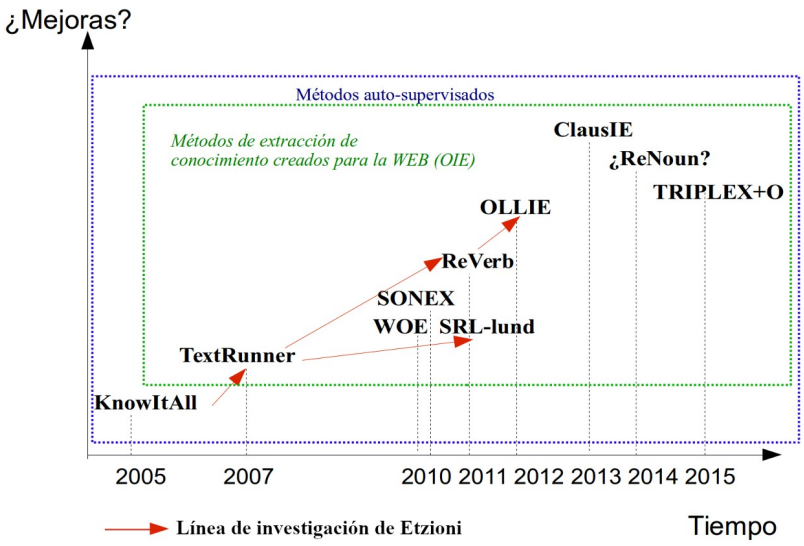


Figura 1: Mejora supuesta entre los distintos métodos de *Open IE* versus el tiempo

Hay que tener en cuenta también que los métodos no siempre fueron comparados utilizando los mismos conjuntos de datos de entrada. Las publicaciones en donde se presentan los métodos: KnowItAll, TextRunner, ReVerb, SRL-IE-LUND, SRL-IE-UIUC y OLLIE tienen entre sus autores a Oren Etzioni; es decir que en el desarrollo de las comparaciones entre métodos estuvo involucrada al menos una misma persona. Esta clase de continuidad sobre una línea de investigación es un fuerte indicio a favor de los datos presentados.

En la Figura 1, publicada originalmente en [Rodríguez et al., 2015] se puede ver de forma cualitativa como fue evolucionando el desempeño de los distintos métodos a lo largo del tiempo, hasta el año 2015. Las comparaciones fueron obtenidas al analizar los distintos trabajos recogiendo la información allí presentada.

Respecto del método llamado SONEX, en el trabajo original de [Mesquita et al., 2010] no hay comparaciones realizadas contra otros métodos. Sin embargo, existe una comparación contra ReVerb en el trabajo de [Merhav et al., 2012], pero como

SONEX trabaja de una forma completamente diferente a ReVerb, la salida de este último tuvo que ser adaptada para poder ser comparado. Las fórmulas utilizadas para medir el desempeño de ambos métodos fueron: *Purity* e *Inverse Purity*, medidas propuestas en [Amigó et al., 2009] para determinar la distancia entre dos soluciones de *clustering*. Los resultados muestran que ReVerb es mejor en *Purity* y SONEX en *Inverse Purity*. Sin embargo, esas medidas no son suficientes para evaluar el desempeño de SONEX como método de *Open IE y SONEX* no está disponible al público para realizar otras evaluaciones.

Respecto del modo llamado ReNoun, no hay evidencia suficiente en el artículo original de [Yahya et al., 2014] para deducir que tan bueno es su desempeño respecto a otros métodos. Si bien el artículo menciona a ReVerb, OLLIE y ClausIE no hay comparaciones contra éstos y ReNoun no está disponible públicamente para realizar nuevas pruebas.

El enfoque de TRIPLEX es ligeramente distinto ya que funciona como un complemento de ReVerb o bien de OLLIE, y en el estudio realizado en [Mirrezaei et al., 2015], TRIPLEX por sí solo no logra superar a OLLIE (se compara utilizando la medida F1 en este caso) y es el uso conjunto de OLLIE más TRIPLEX el que arroja un mejor resultado, aunque no muy lejano al que arroja OLLIE por sí solo.

En la Tabla 2 se muestra qué método fue comparado contra qué otro, indicando cuál resultó mejor en dicha comparación. La Tabla 2 es una tabla de doble entrada, en donde cada celda debe entenderse como una comparación hecha entre el método indicado en la columna contra el método indicado en la fila. En la celda se indica de manera genérica qué método logró una mayor calidad y cantidad de piezas de conocimiento extraídas, independientemente de la medida utilizada en el artículo. Se indica también la referencia al artículo o los artículos de donde fue relevada la comparación.

Tabla 2. Resumen de comparaciones relevadas entre métodos

Método	TR	WOE	SRL-IE-Lund	ReVerb	Ollie	ClausIE	Triplex
KnowItAll	TR[1]						
TR*		WOE[2,4,7]	SRL-IE-Lund[3]	ReVerb[4,7]		ClausIE[7]	
WOE				ReVerb[4,7]	Ollie[5,7]	ClausIE[7]	
SRL-IE-Lund					SRL-IE-Lund[5]		
ReVerb					Ollie[5,6], ReVerb[7]	ClausIE[7]	Triplex, T.+Re-Verb[6]
Ollie						ClausIE[7]	Ollie, T.+Ollie[6]
ClausIE							
ReNoun							
Triplex							

Referencias: 1. [Banko et al., 2007], 2. [Wu y Weld, 2010], 3. [Christensen et al., 2011], 4. [Fader et al., 2011], 5. [Mausam et al., 2012], 6. [Mirrezaei et al., 2015], 7. [Del Corro y Gemulla, 2013], *****TR: TextRunner

3.2 Estudio de mapeo sistemático sobre métodos extracción de conocimiento para la Web (2018)

En el artículo de [Glauber y Barreiro Claro, 2018] se presentan los resultados de un estudio de mapeo sistemático sobre métodos de extracción de conocimiento para la Web, realizado con el objetivo de responder la siguiente pregunta de investigación:

¿Cuál es el estado del arte respecto de la extracción de conocimiento para la Web?

Sin embargo, como consideraron demasiado ambiciosa la pregunta anterior, propusieron ocho preguntas de investigación secundarias, las cuales, en conjunto podrían ayudar a responder la pregunta principal. Las mismas se listan a continuación:

- **PI1**: ¿Qué términos son usados como sinónimos de *Open Information Extraction*? (pregunta formulada en idioma inglés)
- **PI2**: ¿Cuáles son las fuentes de publicaciones en el área de *Open IE*?
- **PI3**: ¿Qué tipos de contribuciones fueron hechas por estudios de *Open IE*?
- **PI4**: ¿Cuáles son los métodos existentes de *Open IE*?

- **PI5**: ¿Cómo son usados los métodos de *Open IE*?
- **PI6**: ¿Cómo son evaluados los métodos de *Open IE*?
- **PI7**: ¿Cuáles son las herramientas utilizadas en *Open IE*?
- **PI8**: ¿Cuáles son los problemas abiertos en el área de *Open IE*?

De la lista anterior las preguntas más relevantes para esta tesis son la pregunta de investigación número 4 y la número 8. Aunque por supuesto todas son en alguna medida útiles ya que están relacionadas directamente con la propuesta de la tesis.

Los autores no sólo definieron las preguntas a responder sino que también definieron cuáles serían las fuentes de los artículos a consultar, es decir las bases de datos que utilizarían, junto con los criterios de búsqueda tales como: palabras claves, años válidos para las publicaciones, idioma, etc. En la Tabla 3 se resumen estos resultados parciales.

Tabla 3. Artículos obtenidos en [Glauber y Barreiro Claro, 2018]

Base de datos (fuente)	Artículos devueltos
Science Direct	47
IEEE Xplore	233
ACM Digital Library	164
Scopus	226
Google Scholar	1813

Sobre estos 2483 artículos encontrados, los autores aplicaron luego 6 filtros, para quedarse sólo con los estudios primarios pertinentes. Se detallan estos filtros a continuación:

- **F1**: Eliminar artículos no escritos en idioma inglés
- **F2**: Eliminar artículos no publicados en *journals* o conferencias.
- **F3**: Eliminar artículos cortos.
- **F4**: Eliminar encuestas o artículos de revisión.
- **F5**: Eliminar artículos que incluyen los términos: “Open IE” o similares pero que no son artículos sobre el tópico.
- **F6**: Eliminar artículos duplicados (mismo artículo obtenido de fuentes distintas).

Luego de aplicar los filtros mencionados los autores sólo retuvieron 73 estudios primarios. La distribución de los mismos por base de datos de origen se muestra en la Figura 2, publicada originalmente en el artículo de [Glauber y Barreiro Claro, 2018].

En la Tabla 4 se muestran un total de 28 métodos de extracción de conocimiento para la Web, los cuales fueron presentados en los estudios primarios relevados por Glauber y Barreiro Claro. Se los muestra ordenados por año de aparición junto el artículo de referencia.

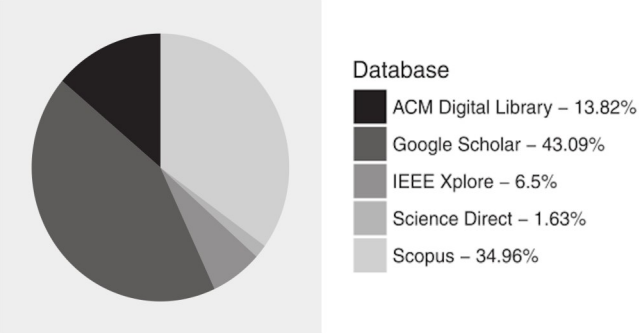


Figura 2: Porcentaje de cada base de datos en la selección de estudios primarios [Glauber y Barreiro Claro, 2018]

Tabla 4. Métodos relevados en el artículo de [Glauber y Barreiro Claro, 2018]

Método	Artículo	Año	Lenguaje
MinIE	[Gashteovski et al., 2017]	2017	inglés
RelP	[de Abreu y Vieira, 2017]	2017	portugués
C-COERE	[Wu y Wu, 2017]	2017	chino
vnOIE	[Truong, D. et al., 2017]	2017	vietnamita
R-OpenIE	[Lin et al., 2016]	2016	inglés
SemIE	[Tan et al., 2016]	2016	inglés
REALTEXT	[Perera y Nand, 2015]	2015	inglés
ClausORE	[Xu et al., 2015]	2015	chino
CORE	[Petroni et al., 2015]	2015	inglés
GCORE	[Wang et al., 2015]	2015	chino
ArgOE	[Gamallo y Garcia, 2015]	2015	multilenguaje
LSOE	[Xavier et al., 2015]	2015	inglés
Stanford OpenIE	[Angeli et al., 2015]	2015	inglés
WEBCHILD	[Tandon et al., 2014]	2014	inglés
CORE	[Tseng et al., 2014]	2014	chino
DepOE+	[Garcia y Gamallo, 2014]	2014	multilenguaje
LEGALO	[Consoli et al., 2015]	2014	inglés
ReNoun	[Yahya et al., 2014]	2014	inglés
AWAKE	[Boschee et al., 2014]	2014	inglés
ZORE	[Qiu y Zhang, 2014]	2014	chino

ClausIE	[Del Corro y Gemulla, 2013]	2013	inglés
EXEMPLAR	[Mesquita et al., 2013]	2013	inglés
CSD-IE	[Bast y Haussmann, 2013]	2013	inglés
SONEX	[Merhav et al., 2012]	2012	inglés
OLLIE	[Mausam et al., 2012]	2012	inglés
OntExt	[Mohamed et al., 2011]	2011	inglés
REVERB	[Fader et al., 2011]	2011	inglés
RDROIE	[Kim et al., 2011]	2011	inglés

3.2.1 Revisión de los métodos de *Open IE* relevados por Glauber y Barreiro Claro

En esta sección se analizarán los nuevos métodos relevados en [Glauber y Barreiro Claro, 2018] que trabajan en idioma inglés, con el objeto de entender cuáles de ellos, según lo que indique cada artículo, superan a los ya conocidos en el estado del arte: ClausIE, OLLIE y/o ReVerb, según lo relevado en la sección 3.1.

MinIE es comparado en precisión, en el artículo de [Gashteovski et al., 2017], con OLLIE, ClausIE y Standford OpenIE, utilizando dos conjuntos de datos diferentes. En una de sus variantes (MinIE-C) resulta ser más preciso que cualquiera de los otros tres métodos, y aunque en el artículo no miden la exhaustividad (*recall*), este método logra la mayor cantidad de extracciones correctas. MinIE está disponible de forma pública junto a su código fuente.

R-OpenIE, publicado por [Lin et al., 2016] es comparado con ReVerb, OLLIE y ClausIE. Es más rápido que éstos y también más preciso. R-OpenIE extrae además mayor cantidad de relaciones semánticas correctas, aunque no se mide la exhaustividad del mismo. Lamentablemente este método no se encuentra disponible de forma pública.

SemIE en [Tan et al., 2016] es comparado con OLLIE, al cual mejora en precisión y exhaustividad. Sin embargo, SemIE utiliza aprendizaje semisupervisado para mejorar las extracciones para un dominio específico, debido a esto, no sería un método de *Open IE* según la definición más estricta. Este método no se encuentra disponible de forma pública.

REALTEXT [Perera y Nand, 2015] no es un método de *Open IE*, sino que es un *framework* capaz de generar patrones que pueden ser usados para la lexicalización de conocimiento estructurado como tuplas, donde cada tupla consta de un sujeto, un predicado y un objeto. Este *framework* utiliza internamente un método de Open IE.

Más precisamente OLLIE. En el artículo se proporciona un enlace web al proyecto descripto, pero este enlace está roto.

CORE, presentado en [Petroni et al., 2015] es un método de extracción de relaciones semánticas, pero no de *Open IE*. El objetivo de CORE es el de "razonar" sobre hechos extraídos, utilizando sistemas de *Open IE*, añadiendo a estos hechos, información de contexto. Los autores afirman que CORE es agnóstico respecto al método de *Open IE* utilizado. En su trabajo buscan ir más allá de las relaciones semánticas, buscan una definición precisa de lo que estas significan.

ArgOE [Gamallo y Garcia, 2015] es uno de los dos métodos relevados que soporta múltiples idiomas. Los autores lo evalúan en 3 idiomas diferentes: inglés, portugués y español. Para medir el desempeño del método en idioma inglés, ArgOE es comparado con otros métodos ya conocidos: ReVerb, WOE, OLLIE, TEXTRUNNER y ClausIE. ArgOE logra superar a todos en cuanto cantidad de extracciones semánticas correctas extraídas, excepto a ClausIE, que lo supera ampliamente. Los métodos también son comparados en precisión, en exhaustividad y utilizando la medida F1. ReVerb y ClausIE obtienen un valor para la medida F1 superior a ArgOE, aunque éste supera a OLLIE, WOE y TEXTRUNNER. ArgOE se encuentra disponible de forma pública junto a su código fuente.

LSOE [Xavier et al., 2015] es comparado con ReVerb y DepOE. Mejora en precisión a ambos pero la diferencia con ReVerb no es muy grande, además ReVerb obtiene una mayor cantidad de extracciones correctas. No hay cálculos ni comparaciones de la exhaustividad ni de la medida F1. En el artículo hay un enlace para descargar el proyecto, pero está roto.

Stanford OpenIE [Angeli et al., 2015] es comparado en precisión, exhaustividad y medida F1 con otros dos métodos: OLLIE y uno llamado *UW Official*, en donde UW es la sigla para *University of Washington*. *UW Official* es un método no citado pero que identifican como el sucesor de OLLIE (posiblemente se trata de una versión preliminar del método que luego sería publicado luego como OpenIE4). Standford OpenIE supera a ambos métodos. Lamentablemente el mismo no está disponible de forma pública.

WEBCHILD [Tandon et al., 2014] no es un método de *Open IE*. Según los autores es un método para construir de forma automática una gran base de datos de sentido

común (c*ommonsense knowledge base*) utilizando contenido Web. El mismo está disponible públicamente.

DepOE+ no es un método de *Open IE* pero sí lo es DepOE. En [Garcia y Gamallo, 2014] los autores presentan un sistema de resolución de correferencias llamado LinkPeople. Los autores sugieren que LinkPeople puede mejorar el rendimiento de otros algoritmos, entre ellos los métodos de *Open IE*. Para probar su propuesta utilizan un método de *Open IE* llamado DepOE. Hacen dos ejecuciones, en la primera ejecutan DepOE tomando como entrada un archivo en texto plano en lenguaje natural. En la segunda prueba, ejecutan DepOE sobre la salida del algoritmo LinkPeople. A esta segunda ejecución de DepOE los autores la han llamado DepOE+.

DepOE fue presentado originalmente en el artículo de [Gamallo y Garcia, 2012] como un método de *Open IE* multilenguaje, el cual es evaluado en idioma inglés y comparado con ReVerb. Los resultados de las pruebas realizadas en el artículo indican que DepOE es más preciso que ReVerb, sin embargo, ReVerb tiene una mayor exhaustividad y una mayor medida F1. Además se indica que ReVerb es ligeramente más rápido. DepOE está disponible de forma pública, también su código fuente.

LEGALO [Consoli et al., 2015], es similar a un método de *Open IE* pero su salida es mucho más compleja, ya que en lugar de devolver una tupla devuelve un grafo. En el artículo de [Consoli et al., 2015] no hay comparaciones hechas contra otros métodos de *Open IE*. Los autores señalan que si bien sería natural comparar los resultados de LEGALO con los de otros sistemas de extracción de relaciones semánticas, esa comparación requeriría manipular la salida de los otros métodos. Tarea que no estaría contemplada dentro del alcance de su artículo. Los autores además identifican a LEGALO, no como un sistema de *Open IE*, sino como un sistema de *Open Knowledge Extraction* (*OKE*). Éste sería un concepto similar, pero más enfocado en que la salida represente conocimiento. Existe una versión de prueba en línea, pero no se dispone de su código fuente.

AWAKE no es un método de *Open IE*. En el artículo de [Boschee et al., 2014] se explica que AWAKE es un *framework* extensible que cuenta con herramientas para el entendimiento automático del lenguaje. La idea de los autores es que AWAKE sea capaz de analizar grandes cantidades de artículos científicos para luego armar una base de conocimiento y de esta forma ahorrarle a los investigadores la ardua tarea de

tener que leer cientos de artículos distintos. Los investigadores podrían entonces concentrarse sólo en el análisis de los mismos. AWAKE internamente utiliza ReVerb.

EXEMPLAR [Mesquita et al., 2013] es comparado con varios métodos de *Open IE*, entre los que se encuentran: ReVerb, SONEX, OLLIE y Lund, utilizando para ello 3 conjuntos de datos diferentes. En cada conjunto EXEMPLAR obtiene la medida F1 más alta y la exhaustividad más alta. Pero la precisión más alta la obtienen Lund y SONEX. Lamentablemente EXEMPLAR no está disponible de forma pública.

CSD-IE [Bast y Haussmann, 2013] es comparado por los autores con ClausIE, ReVerb y OLLIE. Para ello utiliza 2 conjuntos de datos diferentes, los mismos propuestos por Del Corro y Gemulla en el artículo original de ClausIE [Del Corro y Gemulla, 2013]. En ambas pruebas se mide la precisión y la cobertura (*coverage)*, medida definida como: *el porcentaje de apariciones de palabras en las tuplas extraídas por un sistema respecto a las existentes en el texto de entrada* [Bast y Haussmann, 2013]. La cobertura es una forma indirecta y menos precisa de medir la exhaustividad. CSD-IE obtiene, en ambos conjuntos de datos, una cobertura mayor que los otros 3 métodos, sin embargo, ReVerb obtiene la mayor precisión para ambos conjuntos de datos. Esto se debe a que ReVerb extrae pocas tuplas, pero el porcentaje de error en ellas es menor. El que más tuplas totales extrae, en ambos escenarios, es CSD-IE y también es quien más tuplas correctas extrae. Seguido en ambos casos por ClausIE. Es interesante remarcar acá que MinIE [Gashteovski et al., 2017], es también evaluado con los mismos dos conjuntos de datos y logra extraer, en ambos casos, aún más cantidad de tuplas correctas que CSD-IE. De todos modos las pruebas no son directamente comparables ya que los criterios de evaluación, sobre si una extracción es correcta o no, pueden diferir en ambos artículos. Este método no está disponible de forma pública.

OntExt [Mohamed et al., 2011], es un método híbrido entre un método de *Open IE* y un método tradicional de extracción de relaciones semánticas, entiéndase en este caso, un método creado para extraer relaciones previamente conocidas en un dominio específico. Para ello, OntExt utiliza una ontología que le permite mejorar, según sus autores, la calidad de las relaciones semánticas extraídas. Sin embargo, este método no es comparado con otros métodos existentes y tampoco está disponible.

RDROIE [Kim et al., 2011] es presentado por los autores como un método de *Open IE*, aunque en las conclusiones sugieren utilizarlo para dominios específicos,

realizando previamente un entrenamiento acorde al dominio de interés. Por otro lado, el método es comparado con TEXTRUNNER y si bien obtiene una mayor precisión que éste, la medida F1 es superior en TEXTRUNNER, un método ampliamente superado por otros sistemas de *Open IE*. RDROIE no está disponible al público.

En base al análisis anterior se puede concluir que sólo los siguientes métodos son serios candidatos para ser considerarlos dentro del estado del arte (al menos hasta 2018): MinIE, R-OpenIE, Standford OpenIE, Exemplar y CSD-IE. De esta lista sólo MinIE, R-OpenIE y CSD-IE son comparados contra ClausIE, el método más relevante hasta 2015 y si bien los respectivos autores no han medido la exhaustividad, sí han medido la cantidad de extracciones totales y correctas en los artículos de MinIE y CSD-IE. En ambos casos la cantidad de extracciones correctas es superior a ClausIE, lo que sugiere que estos métodos tendrían también una exhaustividad más alta y por lo tanto una medida F1 mayor a ClausIE.

Lamentablemente de la lista aterior de métodos candidatos, sólo disponemos de MinIE para realizar nuevas pruebas y comparaciones. Afortunadamente MinIE parece ser el método que mejor desempeño tiene y es el más moderno de todos ya que fue publicado en 2017.

3.3 Revisión sistemática de literatura: nuevos métodos de Open IE (2021)

Para cubrir el intervalo de tiempo transcurrido entre el artículo de [Glauber y Barreiro Claro, 2018] y el momento de redacción de este documento de tesis se realizó una nueva revisión sistemática de literatura, utilizando la metodología descripta en [Keele Staffs, 2007]. Esta nueva revisión se concibió también como una continuación del trabajo realizado por Galuber y Barreiro Claro.

La pregunta de investigación principal fue:

¿Cuáles son los métodos de extracción de conocimiento para la Web (Open Information Extraction) en el estado del arte?

Las bases de datos o fuentes de publicaciones utilizadas para la búsqueda de artículos fueron las siguientes:

- Science Direct
- IEEE Xplore
- ACM Digital Library

- Google Scholar

Los criterios utilizados y las palabras claves o términos para realizar la búsqueda fueron:

- Aparición de los términos: “Open Information Extraction” y “Open IE”
- Publicaciones hechas a partir del año: 2018
- Sólo en conferencias y revistas especializadas (*journals*)

La búsqueda anterior, arrojó los resultados parciales que se listan en la Tabla 5.

Tabla 5. Cantidad de artículos obtenidos por base de datos

Base de datos (fuente)	Cantidad de artículos
Science Direct	53
IEEE Xplore	23
ACM Digital Library	124
Google Scholar	981

Sobre estos 1181 artículos encontrados, se aplicaron tres filtros para agilizar el análisis:

- Se eliminaron artículos duplicados.
- Se dejaron sólo métodos en idioma inglés o español.
- Se descartaron las publicaciones de tipo póster.

Luego se procedió a una revisión más detallada de los artículos restantes con el fin de dejar sólo aquellos que presentasen un método novedoso de extracción de conocimiento para la Web. Los métodos encontrados se resumen en la Tabla 6, en donde se indica también el artículo en el cual fue presentado y si el método está disponible al público general.

Tabla 6. Métodos de extracción de conocimiento para la Web, desde 2018.

Método	Artículo	Lenguaje	Disponible
DSN	[Song et al., 2018]	inglés	No
LS3RyIE	[Vo y Bagheri, 2018]	inglés	No
LS3RyIE+BT	[Vo y Bagheri, 2018]	inglés	No
WW-PIE	[Li et al., 2018]	inglés	No
ReMine	[Zhu et al., 2018]	inglés	No
BioOpenIE	[Wang et al., 2018]	inglés	No
StuffIE	[Prasojo et al., 2018]	Inglés	Sí
Graphene	[Cetto et al., 2018]	inglés	Sí
CALM/ OpenIE5	[Saha y Mausam, 2018]	inglés	Sí
Neural OpenIE	[Cui et al., 2018]	inglés	No

RnnOIE-aw	[Stanovsky et al., 2018]	inglés	Sí
RnnOIE-verb	[Stanovsky et al., 2018]	inglés	Sí
NEURON	[Bhutani et al., 2019]	inglés	No
MinScIE	[Lauscher et al., 2019]	inglés	Sí
SenseOIE	[Roy et al., 2019]	inglés	Sí
Multi^{2}OIE	[Ro et al., 2020]	multilenguaje	Sí
OpenIE4	[Christensen et al., 2011; Kolluru et al., 2020a]	inglés	Sí
OpenIE6	[Kolluru et al., 2020a]	inglés	Sí
IMoJIE	[Kolluru et al., 2020b]	inglés	Sí
CrossOIE	[Cabral et al., 2020]	multilenguaje	Sí
SpanOIE	[Zhan y Zhao, 2020]	inglés	Sí
RnnOIE-Full	[Tang et al., 2021]	inglés	Sí
ReLink	[Tran y Nguyen, 2021]	inglés	Sí

3.3.1 Revisión de los métodos de *Open IE* publicados a partir de 2018

En esta sección se analizarán los métodos relevados en la sección 3.3 que trabajan en idioma inglés con el objeto de entender cuáles de ellos, según se indique en cada artículo, superan en desempeño a los ya relevados en el estado del arte.

DSN, publicado en [Song et al., 2018] supera en precisión, según sus autores, a ClausIE, ReVerb y OLLIE. La precisión promedio calculada para este método es de 0,67. Sin embargo, la exhaustividad y la medida F1 no son calculadas en el artículo. Este método no está disponible al público general.

LS3RyIE [Vo y Bagheri, 2018] supera en precisión a ClausIE, ReVerb y OLLIE en el conjunto de datos propuesto en el artículo de ReVerb [Fader et al., 2011]. La precisión en dicho conjunto de datos es de 0,68. Sin embargo, la exhaustividad y la medida F1 no son calculadas en el artículo. Este método tampoco se encuentra disponible al público general.

LS3RyIE+BT, presentado en el mismo artículo que el método anterior [Vo y Bagheri, 2018], supone una mejora respecto al método LS3RyIE al ejecutarlo conjuntamente con un algoritmo de *bootstrapping*. Esta combinación de algoritmos logra un mejor desempeño global ya que la precisión aumenta a 0,72. La exhaustividad y la medida F1 son también calculadas y sus valores son de 0,64 y 0,68 respectivamente. LS3RyIE+BT no sólo supera a ClausIE, ReVerb y OLLIE, cosa que ya hacía el método LS3RyIE solo, sino que además supera a ClausIE, ReVerb y OLLIE ejecutados con el algoritmo de *bootstrapping*. Lamentablemente este método y su código fuente no se encuentran disponibles.

WW-PIE [Li et al., 2018] es un método de *Open IE*, pero específico para literatura biomédica, al ser específico para un dominio estaría violando la definición más estricta de cómo debe ser un método de *Open IE* según [Banko et al., 2007], sin embargo, es posible entender como los demás principios de estos métodos son aplicados, pero dentro de un dominio específico para textos de diversas fuentes. En textos de literatura biomédica WW-PIE supera a MinIE, OLLIE, Stanford OpenIE y ClausIE en precisión, la cual llega a 0,73. Otras medidas no son calculadas. El método no está disponible al público.

ReMine [Zhu et al., 2018] sí es un método de *Open IE* de propósito general y el mismo supera a OLLIE, Stanford OpenIE y ClausIE en cuanto a precisión, la misma llega a ser 0,74 en cierto conjunto de datos, aunque es 0,59 en otro. La exhaustividad y la medida F1 no son calculadas pero se indica que la cantidad de tuplas extraídas por este método es menor que la cantidad de tuplas extraídas por los otros métodos utilizados como referencia. ReMine no se encuentra disponible al público general.

BioOpenIE [Wang et al., 2018] es un método de *Open IE* específico para literatura biomédica. Alcanza una precisión de 0,96 superando a OLLIE, MinIE, ClausIE y Stanford OpenIE en su dominio de aplicación. No calculan la exhaustividad y los autores aclaran en su artículo que no lo hacen debido a la dificultad para calcular esta medida.

StuffIE [Prasojo et al., 2018] es un método de *Open IE* que extrae relaciones semánticas anidadas y sus facetas. No devuelve tuplas. Según sus autores, supera a ClausIE aunque en el artículo no se calculan ni la precisión ni la exhaustividad. Se encuentra disponible para descargar junto a su código fuente.

Graphene [Cetto et al., 2018] es un método de *Open IE* que usa una transformación de dos capas: una capa de incrustación de cláusulas y una capa de desintegración de frases en conjunto con una identificación de relaciones retóricas. Su precisión promedio es de 0,5 y su exhaustividad es de 0,27. La medida F1 no es calculada. Supera en la precisión promedio a OpenIE4 aunque no los supera en exhaustividad. El método y su código fuente se encuentran disponibles.

CALM [Saha y Mausam, 2018] es en realidad un algoritmo pensado para mejorar el *parser* de los métodos de *Open IE*. Este algoritmo al ser aplicado al *parser* de OpenIE4 logra mejorar su rendimiento. CALM forma parte integral del método

OpenIE5, el sucesor de OpenIE4. En el artículo no se calculan la precisión ni la exhaustividad. CALM se encuentra disponible para su uso junto a su código fuente.

Neural OpenIE es presentado en [Cui et al., 2018] como el enfoque a la extracción de conocimiento para la Web desde las redes neuronales. Para una exhaustividad máxima de 0,6 este método logra una precisión de 0,45. La medida F1 no es calculada, pero se calcula en su lugar el área bajo la curva ROC y es 0,47, superando a OpenIE4, OLLIE y ClausIE. Este método no está disponible al público.

RnnOIE-aw [Stanovsky et al., 2018] se trata de un método *Open IE* supervisado. Logra una precisión de 0,6 y una exhaustividad de 0,62. Su medida F1 es de 0,62 también. En las pruebas realizadas por los autores supera a OpenIE4 y ClausIE. Se encuentra disponible para su uso junto a su código fuente.

RnnOIE-verb también fue publicado en [Stanovsky et al., 2018], se trata de un método de *Open IE* supervisado. Su rendimiento general no es tan bueno comparado con el de RnnOIE-aw. La medida F1 de RnnOIE-verb es de 0,59 y la de RnnOIE-aw es de 0,62. No hay comparaciones entre éste y otros métodos de *Open IE*. RnnOIE-verb se encuentra disponible para su descarga junto a su código fuente.

NEURON [Bhutani et al., 2019] no es un método de *Open IE* de propósito general, sino que se trata de un método para extraer tuplas de pares pregunta-respuesta. En el artículo no se calcula la precisión ni la exhaustividad. Tampoco hay comparaciones con otros métodos. No está disponible al público general.

MinScIE [Lauscher et al., 2019] es un método de *Open IE* específico para trabajar con citas científicas. Es comparado en precisión con MinIE sobre un conjunto de datos específico de citas científicas y lo supera en precisión en un 3%. No se calculan otras medidas. Se cuenta con acceso al código fuente.

SenseOIE [Roy et al., 2019] es un método de *Open IE* basado en redes neuronales. Es comparado usando la medida F1 en 5 conjuntos de datos diferentes con otros 4 métodos: RnnOIE [Stanovsky et al., 2018], OpenIE5, Stanford OpenIE y UKG (un método de *Open IE* propietario creado por los mismos autores). En todos los conjuntos de datos, excepto en uno, SenseOIE logra una mejor medida F1, siendo su máximo: 0,79 y su mínimo 0,41. No se dispone de acceso al método ni a su código fuente.

Multi^{2}OIE [Ro et al., 2020] es un método de *Open IE* que soporta múltiples idiomas, en el artículo de referencia los autores lo evalúan en inglés, español y portugués. Para cada uno de estos idiomas calculan la precisión, la exhaustividad y la medida F1. En idioma inglés es comparado con Stanford OpenIE, OLLIE, ClausIE, OpenIE4, RnnOIE y SpanOIE en dos conjuntos de datos diferentes. En ambos Multi^{2}OIE logra una mayor medida F1 y una mayor exhaustividad. Su medida F1 en el primer conjunto es de 0,84 y de 0,52 en el segundo. Los métodos también son evaluados utilizando el área bajo la curva ROC (AUC), en ambos conjuntos Multi^{2}OIE logra el valor más alto (0,75 y 0,33 respectivamente). En su modalidad para varios idiomas Multi^{2}OIE es evaluado en un conjunto de datos multilenguaje y comparado con dos métodos multilenguaje: ArgOE y PredPatt. Multi^{2}OIE obtiene la mayor precisión, exhaustividad y medida F1 para inglés, español y portugués. Los resultados de dicha evaluación se muestran en la Tabla 7. Si bien en [Ro et al., 2020] utilizan PredPatt como un método similar con el cual comparar, PredPatt [White et al., 2016] no es un método de *Open IE*, sino que es una herramienta ligera para identificar la estructura de predicados y argumentos de las dependencias sintácticas de tipo *Universal Dependencies* (dependencias universales), la salida de este método es utilizada por los autores en [White et al., 2016] como entrada para los protocolos de anotación desacoplados, que es el punto central de su trabajo.

OpenIE4, OpenIE5 y OpenIE6 son una familia de métodos de *Open IE* tal que cada nueva versión supera a la anterior. Cada uno de estos métodos fue liberado al público general junto con su código fuente pero sólo OpenIE6 fue presentado formalmente en un artículo científico [Kolluru et al., 2020a, p. 6]. Sin embargo, en dicho artículo se citan a los otros dos métodos y se los asocia a publicaciones previas. OpenIE4 está asociada a la publicación de [Christensen et al., 2011] aunque en dicho artículo no se menciona a ningún método con el nombre de OpenIE4. OpenIE5 está asociado al artículo [Saha y Mausam, 2018] en donde presentan el algoritmo CALM, el cual al ser usado para mejorar a OpenIE4 daría como resultado OpenIE5 (posiblemente con alguna mejora adicional). Finalmente OpenIE6 es presentado como un método superador de los anteriores en el artículo de [Kolluru et al., 2020a], en donde se puede observar el rendimiento de los tres métodos en 4 conjuntos de datos diferentes utilizando las métricas: medida F1 y AUC. OpenIE6 no sólo es comparado con sus versiones anteriores, sino también con MinIE, ClausIE, SenseOIE, SpanOIE, RnnOIE y ImoJIE. En todos los conjuntos de datos, OpenIE6 obtiene la medida F1

más alta y la AUC más alta, con excepción del primer conjunto en donde obtiene el segundo lugar luego de IMoJIE. El detalle de los valores se puede ver en la Tabla 8.

Tabla 7. Resultados de la evaluación de Multi^{2}OIE en modo multilenguaje y su comparación con ArgOE y PredPatt [Ro et al., 2020].

Idioma	Método	F1	Precisión	Exhaustividad
Inglés	ArgOE	0,43	0,57	0,35
	PredPatt	0,53	0,54	0,52
	Multi^{2}OIE	0,69	0,67	0,72
Español	ArgOE	0,39	0,48	0,33
	PredPatt	0,44	0,45	0,44
	Multi^{2}OIE	0,6	0,59	0,61
Portugués	ArgOE	0,38	0,46	0,33
	PredPatt	0,43	0,44	0,42
	Multi^{2}OIE	0,59	0,56	0,63

Tabla 8. Métodos evaluados en el artículo de [Kolluru et al., 2020a, p. 6].

Método	Bases de datos de prueba						
	Conjunto A		Conjunto B		Conjunto C		Conjunto D
	F1	AUC	F1	AUC	F1	AUC	F1
MinIE	0,419	-	0,384	-	0,523	-	0,285
ClausIE	0,450	0,220	0,402	0,177	0,610	0,380	0,332
OpenIE4	0,516	0,295	0,405	0,201	0,543	0,371	0,344
OpenIE5	0,480	0,250	0,427	0,206	0,599	0,399	0,354
SenseOIE	0,282	-	0,239	-	0,311	-	0,107
SpanOIE	0,485	-	0,379	-	0,540	-	0,319
RnnOIE	0,490	0,260	0,395	0,183	0,560	0,320	0,264
IMoJIE	0,535	0,333	0,414	0,222	0,568	0,396	0,360
OpenIE6	0,527	0,337	0,464	0,268	0,656	0,484	0,400

IMoJIE publicado por [Kolluru et al., 2020b], es un método de *Open IE* de propósito general para idioma inglés. En el artículo de referencia se calcula su medida F1, la cual es de 0,535 y se lo compara con otros métodos en el estado del arte: MinIE, OpenIE4, OpenIE5, SenseOIE y SpanOIE, a los cuales supera. Este método se encuentra disponible junto con su código fuente.

CrossOIE [Cabral et al., 2020] no es un método de *Open IE* sino que se trata de un sistema de clasificación, su propósito es decidir si una extracción es correcta o no. A cada extracción semántica realizada le asigna un puntaje. Este sistema trabaja con

redes neuronales y soporta los idiomas: inglés, español y portugués. Se dispone de acceso a su código fuente.

SpanOIE [Zhan y Zhao, 2020] es un método de *Open IE* en idioma inglés y de propósito general. Los autores lo comparan con Stanford OpenIE, OLLIE, ClausIE y OpenIE4 en dos medidas: F1 y AUC, utilizando dos conjuntos de datos. En ambos casos SpanOIE obtiene los valores más altos. Su medida F1 es de 0,69 en un conjunto de datos y de 0,79 en otro. Se dispone del código fuente del método.

RnnOIE-Full [Tang et al., 2021] es en verdad un *framework* de entrenamiento, y es utilizado para entrenar al método RnnOIE (RnnOIE-aw) descripto anteriormente. Para diferenciar el rendimiento de este nuevo modelo respecto del anterior, llaman al método original RnnOIE-Supervised y al nuevo RnnOIE-Full. El propósito de este *framework* es eliminar la parte del etiquetamiento manual (realizado por personas) del proceso de entrenamiento. Como se mencionó RnnOIE es un método basado en redes neuronales. Este nuevo modelo es comparado con ClausIE, Stanford OpenIE, OpenIE4 y por supuesto con RnnOIE-Supervised (incluso con distintas variantes de éste) en 3 conjuntos de datos, utilizando dos métricas AUC y la medida F1. Los resultados son mixtos, en un conjunto de datos RnnOIE-Full obtiene la mayor medida F1 y la mayor AUC, en otro conjunto obtiene la mayor medida F1 pero la mayor AUC la obtiene una de sus variantes supervisadas. En el último de los conjuntos logra la mayor AUC (su versión supervisada obtiene el mismo valor), pero OpenIE4 obtiene la mayor medida F1. Se dispone de acceso a su código fuente.

ReLink [Tran y Nguyen, 2021] es un método de *Open IE* de propósito general en idioma inglés. Este método es comparado por los autores contra ClausIE, OLLIE y ReVerb. Logra superar a OLLIE y ReVerb pero no a ClausIE. Las métricas calculadas para este método son la precisión, la exhaustividad y la medida F1, siendo esta última de 0,38 en promedio. Supuestamente se dispone de acceso público al método aunque no está el enlace de acceso en el artículo.

Luego de revisar minuciosamente los distintos artículos, en donde son presentados y evaluados diversos métodos de extracción de conocimiento para la Web o simplemente métodos de *Open IE*, según su nombre en inglés, se puede concluir que hay 5 métodos que, según sus respectivos autores, superan en rendimiento a los demás, y éstos son: SenseOIE, MultiOIE, OpenIE6, IMoJIE y SpanOIE. De esta lista

OpenIE6 es el claro candidato a ser el más efectivo de todos ya que en sus evaluaciones supera a SenseOIE y a EMoJIE.

3.4 Revisión de los métodos de *Open IE*, en idioma español

La lista de métodos de extracción de conocimiento para la Web, que funcionan con texto en lenguaje natural en idioma español no es muy extensa. El primero que aparece es DepOE en el año 2012, el cual no es un método exclusivo de idioma español sino un método multilenguaje que soporta: portugués, español, gallego e inglés. En 2014 aparece ExtrHech, el único método de esta lista que no fue relevado en las secciones anteriores y que es exclusivo para idioma español. En 2015 [Gamallo y Garcia, 2015] publican un nuevo método multilenguaje llamada ArgOE y en dicho trabajo realizan una comparación entre ese método y ExtrHech en idioma español en donde concluyen que ArgOE es más preciso, aunque no por mucho. La precisión calculada es 0,55 y 0,50 respectivamente. Por último, aparece un nuevo método multilenguaje con soporte para idioma español en el año 2020 llamado Multi^{2}OIE [Ro et al., 2020]. Multi^{2}OIE es evaluado en idioma español y comparado con ArgOE al que supera en precisión, exhaustividad y medida F1, como se resume en la Tabla 7.

En la Tabla 9 se presenta un listado de todos los métodos de extracción de conocimiento para la Web que fueron relevados para idioma español junto con su año de aparición y el artículo científico en el cual fueron presentados.

Tabla 9. Métodos de *Open IE* que soportan idioma español.

Método	Artículo	Año	Lenguaje
DepOE	[Gamallo y Garcia, 2012]	2012	multilenguaje
ExtrHech	[Zhila y Gelbukh, 2014]	2014	español
ArgOE	[Gamallo y Garcia, 2015]	2015	multilenguaje
Multi^{2}OIE	[Ro et al., 2020]	2020	multilenguaje

4. Conjunto de pruebas

Para contar con un marco de referencia único con el cual medir el desempeño de diversos métodos de extracción de conocimiento para la Web, se construyó una base de datos con textos en lenguaje natural y sus relaciones semánticas asociadas. Esta base de datos sirvió no sólo para establecer el desempeño de diversos métodos sino también para evaluar los nuevos métodos propuestos construidos como objeto de esta tesis.

Se escogió la base de datos con textos de cables de noticias en idioma inglés conocida como Reuters-21578 [Lewis, 1997]. Esta base de datos es ampliamente conocida y utilizada en diversos trabajos de procesamiento de lenguaje natural en idioma inglés [Joachims, 1998; Steinbach et al., 2000; Yang y Liu, 1999; Zhao y Karypis, 2001]. Otra de las razones para la elección de esta base de datos es que cada cable de noticias es un fragmento corto de texto en lenguaje natural, en general, con información fáctica fácilmente reconocible por una persona, a diferencia de otros tipos de textos como podrían ser un fragmento de poesía, una metáfora o una analogía.

La cantidad de noticias en la base de datos es demasiado grande para poder realizar una extracción manual de todas las relaciones semánticas presentes, hay en ella 21578 noticias. Además hay que tener en cuenta que cualquier extracción automática que se quiera hacer deberá ser corregida luego por una persona. Planteado de esta forma, se hace casi imposible tomar una base de datos tan grande como Reuters-21578 y usarla para medir el desempeño de métodos de *Open IE* utilizando supervisión manual. Por ello se decidió tomar sólo un subconjunto aleatorio de la base de datos.

El objetivo fue evaluar un método de *Open IE* en un subconjunto la base Reuters-2157, tal que fuese posible afirmar que el desempeño medido en dicho subconjunto es el mismo que el desempeño del método en la totalidad de la base, con al menos un 95% de certeza y un margen de error máximo de un 10%. Para calcular la cantidad de textos necesarios en dicho subconjunto se utilizó la Fórmula 5 tomada de [Lubov et al., 1979].

$$n = \frac{N \cdot Z^2 \cdot p \cdot (1-p)}{(N-1) \cdot e^2 + Z^2 \cdot p \cdot (1-p)} \quad \textbf{(5)}$$

En donde:

- *N*: es el número total de textos en el base de datos, por lo tanto su valor es 21578
- *Z*: es la desviación del valor medio aceptada para lograr el nivel de confianza deseado. Para un nivel de confianza de 95% , Z debe ser 1,96
- *p*: es la proporción que esperamos encontrar. Para una muestra desconocida se toma 50% que es lo más usual.
- *e*: es el margen de error máximo admitido

De la Fórmula 5 se desprende que *n* deberá ser igual a 96 para los valores propuestos, es decir el subconjunto a crear deberá contener al menos 96 cables de noticias. El trabajo de [Rodríguez et al., 2016a] constituye un estudio preliminar, en el mismo se presentó un subconjunto de pruebas con 55 cables de noticias tomados al azar de la base de datos Reuters-21578, conjunto que será llamado a lo largo de este trabajo: Reuters-55. Finalmente en el trabajo de [Rodríguez et al., 2018] se construyó un subconjunto de pruebas con 103 cables de noticias tomados al azar de la base de datos, la confianza calculada para ese conjunto es ligeramente superior al 95%. A este subconjunto de Reuters-21578 se lo mencionará como Reuters-103[3] a lo largo de este trabajo.

4.1 Extracción manual de relaciones semánticas

Luego de haber seleccionado 103 cables de noticias de forma aleatoria de la base Reuters-21578 se procedió a la extracción manual de las relaciones semánticas presentes en cada texto. Se contó para ello con la ayuda de estudiantes avanzados de Ingeniería en Informática. A cada uno de los estudiantes se les explicó qué eran las relaciones semánticas y se les proveyó de un ejemplo general. Sin embargo, los detalles finos sobre el proceso de extracción fueron dejados a criterio de cada estudiante. Finalmente el autor realizó una revisión de todas las relaciones semánticas extraídas para unificar criterios.

A continuación se muestran una serie de relaciones semánticas extraídas de forma manual del texto en inglés presentado en el Ejemplo 2, el cual pertenece al cable de noticias identificado con el número 44 en base de datos.

[3] *Dump* de mySQL con el conjunto de datos Reuters-103 y el resultado de todas las pruebas realizadas: https://github.com/juanma1982/atp-oie/blob/master/data_tests/reuters.sql

McLean Industries Inc's United States Lines Inc subsidiary said it has agreed in principle to transfer its South American service by arranging for the transfer of certain charters and assets to Crowley Mariotime Corp's American Transport Lines Inc subsidiary. U.S. Lines said negotiations on the contract are expected to be completed within the next week. Terms and conditions of the contract would be subject to approval of various regulatory bodies, including the U.S. Bankruptcy Court. [2]

De este texto se extrajeron las siguientes relaciones semánticas de forma manual:

- (McLean Industries Inc, is subsidiary of, United States Lines Inc)
- (McLean Industries Inc, said, it has agreed in principle to transfer its South American service by arranging for the transfer of certain charters and assets to Crowley Mariotime Corp's American Transport Lines Inc subsidiary)
- (McLean Industries Inc, has agreed to transfer, its South American service by arranging for the transfer of certain charters and assets to Crowley Mariotime Corp's American Transport Lines Inc subsidiary)
- (U.S. Lines, said, negotiations on the contract are expected to be completed within the next week)
- (negotiations on the contract, are expected to be completed, within the next week)
- (Terms and conditions of the contract, would be, subject to approval of various regulatory bodies)

Nótese que cada extracción manual tiene la forma de una tupla de 3 partes, en donde primero aparece el sujeto o argumento primero, luego la relación propiamente dicha y por último el predicado o argumento segundo. También puede encontrarse en la bibliografía que las secciones de la tupla son: entidad 1, relación y entidad 2, lo cual es cierto en muchos casos donde las relaciones semánticas son simplemente relaciones entre entidades. En la tupla del Ejemplo 1: (Albert Einstein, *nació en*, Ulm), por ejemplo, se tienen dos entidades: Albert Einstein, una entidad de tipo *Persona* y Ulm una entidad de tipo *Ciudad*, conectadas mediante la relación: “nació en”. Sin embargo, los métodos de *Open IE* suelen tener estrategias agresivas para recolectar relaciones semánticas y no sólo capturan entidades sino que muchas veces trabajan con frases nominales completas. Un método de *Open IE* podría haber extraído perfectamente “una pequeña ciudad alemana” como argumento segundo, suponiendo que en la oración de entrada dijese eso en lugar de Ulm. Es por eso que en este trabajo de tesis la forma utilizada para referir la estructura de una relación semántica es: (Argumento 1, relación, Argumento 2), ésta fue la forma utilizada para las relaciones semánticas extraídas de forma manual.

4.2 Evaluación de las extracciones automáticas

El siguiente paso fue ejecutar los métodos candidatos (candidatos a ser los mejores), utilizando como entrada para cada uno de ellos cada uno de los 103 cables de noticias existentes en el subconjunto Reuters-103. Luego se realizó una validación manual de las relaciones semánticas extraídas. En [Rodríguez et al., 2018] se evaluaron los métodos: ClausIE, OLLIE y ReVerb. En [Rodríguez et al., 2020] se añadió a la lista MinIE que como se mencionó en la sección 4.2.2, no sólo es el método de *Open IE* más prometedor hasta el año 2018, sino que es el único de la lista de métodos candidatos que está disponible de forma pública para realizar pruebas.

Para evaluar cada relación semántica extraída por un método dado se utilizaron 3 categorías: correcta, incorrecta y casi-correcta. Este último valor fue utilizado para marcar relaciones semánticas difíciles de evaluar. Existen multiples casos en donde es complejo para una persona discernir si una extracción es o no es válida. Si una extracción dada fue marcada como casi-correcta, entonces ésta será ignorada para el cálculo de la precisión y de la exhaustividad. El motivo de esta decisión es el de no penalizar a un método por un trabajo hecho prácticamente bien. Además esta penalización cuenta doble cuando se calcula la medida F1. También se marcaron como casi-correctas relaciones semánticas prácticamente igual a otras marcadas como correctas, en general, se trata de relaciones semánticas sobre un mismo hecho, pero expresadas de forma ligeramente diferente por el mismo método. A estas relaciones semánticas se las llama duplicadas, aunque no son exactamente iguales, ambas refieren al mismo hecho. Por dicho motivo es que sólo una de estas es marcada como correcta y la otra como casi-correcta, con lo cual será ignorada. Se muestra a continuación una extracción realizada por ClausIE, utilizando el texto del Ejemplo 2, que ilustra el caso anterior.

- (it, has agreed, to transfer its South American service)
- (it, has agreed, in principle to transfer its South American service)

Como puede observarse ambas extracciones son correctas. Ambas relaciones semánticas referencian a la misma oración y al mismo hecho concreto. Para este caso puntual, se marcó a la primera como correcta y a la segunda como casi-correcta. Si se marcasen ambas como correctas, aumentaría la precisión de un método respecto a otro y también su exhaustividad, pero esas métricas no estarían reflejando el desempeño real del método.

Se tuvieron que sortear dos dificultades adicionales a la hora de calcular la exhaustividad, la primera de ellas tiene que ver con cómo identificar una coincidencia entre una extracción manual y una extracción automática. Como se mencionó, dos relaciones semánticas pueden ser diferentes pero referir al mismo hecho puntual, por lo cual para entender si un método automático ha encontrado una de las relaciones semánticas presentes en el grupo de extracciones manuales hay que realizar una tarea a conciencia. Se requiere que una persona entienda primero los hechos, las relaciones semánticas existentes en un texto, para luego decidir si dos tuplas, en principio disímiles, están refiriéndose al mismo suceso. Por supuesto, en ambas tuplas se espera que la mayoría de las palabras coincidan pero una pequeña diferencia entre ambas ya es suficiente para que esta tarea no pueda ser realizada de forma automática. Para ilustrar este punto, téngase en cuenta que las siguientes tuplas, extraídas también del texto dado en el Ejemplo 2, son consideradas equivalentes:

- (McLean Industries Inc, has agreed to transfer, its South American service by arranging for the transfer of certain charters and assets to Crowley Mariotime Corp's American Transport Lines Inc subsidiary)
- (it, has agreed, to transfer its South American service)

La primera de ellas es una tupla extraída de forma manual, mientras que la segunda es una tupla extraída de forma automática.

La segunda dificultad para calcular la exhaustividad tuvo que ver con relaciones semánticas válidas, extraídas por un método automático, pero que no estaban presentes en el conjunto de relaciones semánticas extraídas de forma manual. Una posibilidad hubiese sido incorporar dichas relaciones semánticas al conjunto de extracciones manuales, sin embargo, esto podría haber sido injusto al momento de comparar la exhaustividad en distintos métodos ya que la mayoría de estas nuevas relaciones descubiertas hacen referencia a sucesos poco relevantes. En general, decimos que se trata de relaciones poco informativas: válidas, pero con poco o ningún valor informativo. Considérese el texto del Ejemplo 3.

The rise was also slightly below the 3.3 pct growth rate Finance Minister Michael Wilson predicted for 1986 in February's budget. [3]

Una extracción válida pero poco informativa sería:

- (*February*, has, *budget*)

En general, hay una razón por la cual las personas que realizaron el trabajo de detectar relaciones semánticas en estos cables de noticias ignoraron este tipo de relaciones, aunque lo hayan hecho de forma inconsciente, y es que suelen ser irrelevantes en la mayoría de los casos. Es además muy deseable que un método de *Open Information Extraction* extraiga sólo las relaciones semánticas más informativas, ya que estos métodos están pensados para trabajar con volúmenes enormes de datos. Para solventar esta segunda dificultad se agregaron las nuevas relaciones semánticas extraídas por el método automático al grupo de extracciones manuales, pero sólo para el cálculo de la exhaustividad de dicho método, no para todos. Esta decisión implicó que la fórmula tradicional para el cálculo de la exhaustividad (*recall*) sea ligeramente diferente. Sin embargo, considerando que las relaciones adicionales son en verdad un porcentaje pequeño del total de relaciones semánticas extraídas creemos que es una buena aproximación. Además, como se verá más adelante, los resultados obtenidos están en concordancia con los valores medidos por otros autores.

Las fórmulas utilizadas para los cálculos de precisión y exhaustividad pueden describirse claramente con las Fórmulas 5 y 6 respectivamente.

$$precisión = \frac{relaciones\ semánticas\ extraídas\ marcadas\ como\ correctas}{relaciones\ semánticas\ extraídas\ no\ ignoradas} \quad \textbf{(6)}$$

$$exhaustividad = \frac{relacion\ es\ semánticas\ extraídas\ marcadas\ como\ correctas}{extracciones\ manuales + nuevas\ extracciones\ correctas} \quad \textbf{(7)}$$

La medida F1, se calculó utilizando la Fórmula 4, con el parámetro β igual a 1 por lo que quedó como se muestra en la Fórmula 8.

$$F_1 = \frac{2 \cdot Precisión \cdot Exhaustividad}{Precisión + Exhaustividad} \quad \textbf{(8)}$$

4.2.1 Consideraciones sobre ClausIE

En el artículo original de [Del Corro y Gemulla, 2013] ClausIE fue evaluado en 4 modalidades diferentes, que se enumeran a continuación:

1 Con procesamiento de conjunciones coordinantes
2 Sin procesamiento de conjunciones coordinantes
3 Contando extracciones redundantes como correctas
4 Ignorando extracciones redundantes

En el experimento conducido en [Rodríguez et al., 2018] ClausIE se ejecutó sin procesar conjunciones coordinantes y se ignoraron las extracciones redundantes como se explicó en esta misma sección, se las marcó como casi-correctas para no contabilizarlas.

El uso de conjunciones coordinantes es una modalidad de procesamiento de ClausIE en donde una sentencia es dividida en varias distintas utilizando sus conjunciones. De este modo ClausIE logra, idealmente, extraer múltiples tuplas, una con cada relación semántica encontrada. A partir del texto del Ejemplo 4:

Bell makes and distributes electronic, computer and building products. [4]

ClausIE extraería las siguientes tuplas en su modalidad de conjunciones coordinantes:

- (Bell, makes, electronic products)
- (Bell, makes, computer products)
- (Bell, makes, building products)
- (Bell, distributes, electronic products)
- (Bell, distributes, computer products)
- (Bell, distributes, building products)

Sin embargo, éste no es el comportamiento de ClausIE por defecto. Lo que ClausIE produciría como salida por defecto sería lo siguiente:

- (Bell, makes, electronic computer and building products)
- (Bell, distributes, electronic computer and building products)

Evaluar a ClausIE utilizando su modo de conjunciones coordinantes aumentaría el volumen de relaciones semánticas correctamente extraídas aunque los hechos referenciados sean los mismos. Los otros métodos evaluados OLLIE y ReVerb no

trabajan de esa manera y en cambio trabajan de la misma forma en que ClausIE funciona por defecto. Por estas razones, se decidió evaluar en [Rodríguez et al., 2018] a ClausIE en su modalidad por defecto.

4.2.2 Consideraciones sobre MinIE

MinIE está construido utilizando a ClausIE como base pero con la premisa de generar relaciones semánticas más compactas y precisas. MinIE trabaja sobre la salida de ClausIE minimizando los componentes de las tuplas extraídas, haciendo que éstos tengan menos palabras pero sean más significativos. Además MinIE produce extracciones adicionales (a las de ClausIE) para capturar relaciones semánticas implícitas. Para demostrar como MinIE mejora la salida de ClausIE, se cita a continuación un ejemplo de [Gashteovski et al., 2017].

Pinocchio believes that the hero Superman was not actually born on beautiful Krypton. [5]

ClausIE extraería las siguientes tuplas:

- (Pinocchio, believes, that the hero [...] beautiful Krypton)
- (the hero Superman, was not born, on beautiful Krypton)
- (the hero Superman, was not born, on beautiful Krypton actually)

Pero MinIE, extraería por ejemplo las siguientes tuplas:

- (Superman, was born actually on, beautiful Krypton)
- (Superman, "is", hero)

En el Ejemplo 5 se ve claramente como MinIE por un lado minimiza el argumento primero de la tuplas para convertir la frase nominal: “the hero Superman” en tan solo “Superman”, al mismo tiempo que añade una relación semántica utilizando el verbo “is” que no está presente en la oración original. De esta forma indica que “Superman” es un “hero” y no hay pérdida de información por la minimización. Además de lo anterior, corrige el error de ClausIE que no considera que una *creencia* no implica un hecho y genera una tupla con la relación “was not born” cuando en verdad la oración está dando por sabido que sí nació en Krypton (*actually born*). MinIE también cuenta con varias opciones de ejecución que se describen en la Tabla 10.

Tabla 10. Modos de ejecución de MinIE

Modo	Nombre	Descripción
MinIE-C	Complete	No realiza minimizaciones
MinIE-S	Safe	Sólo ejecuta minimizaciones consideradas universalmente seguras
MinIE-D	Dictionary	Utiliza estadísticas a nivel de corpus para informar el proceso de minimización
MinIE-A	Aggressive	Minimiza todo y sólo deja partes que son consideradas universalmente necesarias.

Según su autores [Gashteovski et al., 2017], MinIE obtiene un mejor desempeño cuando es ejecutado con la opción Complete (MinIE-C) y es por ello que en el trabajo realizado por [Rodríguez et al., 2020] MinIE fue evaluado en dicha modalidad.

4.3 Resultados obtenidos

Según los respectivos autores [Del Corro y Gemulla, 2103; Fader et al., 2011; Gashteovski et al., 2017; Mausam et al., 2012] los valores esperados de precisión para los diferentes métodos se muestran en la Tabla 11, según tres conjuntos de pruebas diferentes utilizados en todas las evaluaciones y que se componen de la siguiente forma:

- 200 oraciones tomadas de forma aleatoria del conjunto llamado *New York Times collection* [Sandhaus, 2008].
- 200 oraciones tomadas de forma aleatoria de páginas de Wikipedia.
- 500 oraciones tomadas de forma aleatoria de un servicio de Yahoo llamado: *Yahoo'srandom link* (conjunto ReVerb)

Tabla 11. Precisión esperada

Conjunto de datos	Precisión			
	ClausIE	OLLIE	ReVerb	MinIE
ReVerb	0,615	0,440	0,534	--
Wikipedia	0,670	0,414	0,663	0,750
NYT	0,648	0,425	0,550	0,750
Todos	**0,633**	**0,431**	**0,563**	**0,750**

Luego de evaluar los mismos métodos: ClausIE, OLLIE, ReVerb y MinIE sobre el conjunto de datos propuesto (Reuters-103), se obtuvieron los resultados mostrados en la Tabla 12.

Tabla 12. Métricas calculadas sobre Reuters-103

Medida	Métodos			
	ClausIE	OLLIE	ReVerb	MinIE
Precisión	0,467	0,456	**0,633**	0,612
Exhaustividad	0,519	0,416	0,319	**0,593**
Medida-F1	0,492	0,435	0,424	**0,602**

Lo primero que se observa al comparar las tablas 11 y 12 es que ReVerb obtiene una precisión más alta que la esperada, OLLIE obtiene una precisión relativamente similar a la esperada y ClausIE y MinIE una precisión bastante más baja que la esperada. Sin embargo, al mirar la medida F1 de todos los métodos entendemos que el orden de mejor a peor es esperado: MinIE, ClausIE, OLLIE y por último ReVerb. En el caso de ReVerb, si bien tiene la precisión más alta su exhaustividad, es comparativamente más chica que la de los otros métodos, puede entenderse esto como que ReVerb es un método *conservador:* extrae pocas relaciones semánticas pero esas pocas son mayoritariamente correctas.

4.3.1 Consideraciones sobre los conjuntos de datos de entrada

El primer enfoque propuesto para entender las diferencias observadas entre la precisión esperada y la calculada, en particular con el método de ClausIE, fue el de evaluar los diferentes conjuntos de datos de entrada para discernir si a priori es posible decidir que un método funcionará mejor que otro en determinado conjunto.

Para ello se crearon dos conjuntos de datos, el conjunto “C-ClausIE” y “C-ReVerb”, cada uno de estos conjuntos estuvo conformado por una selección de textos tomados de los 3 conjuntos anteriores (*New York Times collection*, Wikipedia y ReVerb) y también de Reuters-55, el cual como se mencionó es subconjunto de Reuters-103.

El conjunto “C-ReVerb” se conformó con todos aquellos textos tal que, al ser utilizados como entrada de los métodos ReVerb y ClausIE, ReVerb extrajo relaciones semánticas con una precisión al menos 50% superior a ClausIE. A su vez, el conjunto “C-ClausIE” quedó conformado con todos aquellos textos en los cuales ClausIE logró una precisión al menos 50% superior a ReVerb en la extracción de relaciones semánticas [Rodríguez et al., 2016b].

Sobre estos dos conjuntos se aplicaron métodos de clasificación de textos utilizando los siguientes algoritmos: *Bayes Naïve* [McCallum et al., 1998], una

implementación de *Support Vector Machines* (SVMs) [Joachims, 1998] llamada SMO [Platt, 1998] y una implementación de los árboles de decisión C4.5 [Quinlan, 2014] llamada J48 [Gholap, 2012].

Además, el texto original en idioma inglés, fue convertido palabra por palabra a sus categorías gramaticales (*pos-tags*) y también se lo convirtió en etiquetas de IOB [Ramshaw y Marcus, 1999] según la técnica de *text-chunking*. La técnica de *text-chunking*, consiste en dividir frases en segmentos de texto que no se superponen, en base a un análisis superficial. En [Abney, 1991] se propuso este método como un precursor útil y simple de implementar para detectar principalmente frases nominales y verbales. Para convertir una porción de texto en sus categorías gramaticales y también para obtener las etiquetas de IOB correspondientes al *text-chunking* se utilizó el mismo programa ReVerb modificado para que convierta el texto de la forma mencionada utilizando las mismas bibliotecas y versiones de las mismas que utiliza para extraer relaciones semánticas [Rodríguez et al., 2016b]. Para ilustrar estas transformaciones considérese el Ejemplo 6.

She has done so with depth and confidence. [6]

Convertido a categorías gramaticales (*pos-tags*) quedó como:

- PRP VBZ VBN RB IN NN CC NN .

Convertido a etiquetas IOB de *text-chunking* quedó como:

- B-NP B-VP I-VP B-ADVP B-PP B-NP I-NP I-NP O

Luego, cada uno de estos tres conjuntos: el de oraciones en lenguaje natural, el de categorías gramaticales y el de etiquetas IOB de *text-chunking*, fue utilizado para entrenar un clasificador de texto. No sólo se utilizaron los unigramas de cada conjunto para entrenar al clasificador, sino que además se crearon conjuntos adicionales de entrenamiento usando los bigramas y trigramas de cada uno de los conjuntos anteriores. Continuando con el texto dado en el Ejemplo 6, los unigramas, bigramas y trigramas para lenguaje natural, quedarían así:

- **Unigramas**: (She, has, done, so, with, depth, and, confidence)
- **Bigramas**: (<start>-She, She-has, has-done, done-so, so-with, with-depth, depth-and, and-confidence, confidence-<end>)
- **Trigramas**: (<start>-She-has, She-has-done, has-done-so, done-so-with, so-with-depth, with-depth-and, depth-and-confidence, and-confidence-<end>)

Entonces, cada uno de los métodos de clasificación, fue entrenado 9 veces: con los unigramas, bigramas y trigramas del texto en lenguaje natural, los unigramas, bigramas y trigramas de sus categorías gramaticales y con los unigramas, bigramas y trigramas del etiquetado IOB.

Los resultados, publicados en [Rodríguez et al., 2016b], indican que no hay ningún conjunto de palabras, de categoría gramaticales o de etiquetado IOB, ni secuencia de estos elementos que pueda decirnos a priori que un método funcionará mejor que otro. Es decir que son, en principio, independientes del dominio y que su éxito al extraer relaciones semánticas está determinado por características complejas de las oraciones, como su estructura sintáctica.

4.4 ClausIE mejorado: identificador de oraciones

Un estudio minucioso de las relaciones semánticas incorrectas extraídas por ClausIE, reveló que dentro de una misma relación semántica aparecen elementos de dos o más oraciones diferentes, no siempre relacionadas entre sí. En las evaluaciones anteriores del método, principalmente en el artículo de [Del Corro y Gemulla, 2013] los textos de entrada utilizados consistieron en oraciones simples, mientras que en el conjunto de entrada Reuters-103 se cuenta con párrafos enteros como textos de entrada. A partir de dicha observación, se propuso la hipótesis de que ClausIE no es bueno para analizar textos grandes, compuestos por varias oraciones. Para validar esta hipótesis se agregó una nueva función a ClausIE para dividir cada texto de entrada en oraciones independientes.

La idea principal fue la de dividir el texto de entrada en oraciones independientes usando el mismo *parser* de Stanford que ClausIE utiliza para construir el árbol de dependencias. Sin embargo, el *parser* no funcionó como se esperaba, particularmente con abreviaturas y acrónimos. Hay que considerar que los textos de Reuters tienen muchas palabras de este tipo. Un error común fue que algunas oraciones fueron divididas a la mitad porque un punto al término de una abreviatura era confundido por el *parser* con un punto final. Para mejorar el comportamiento del *parser* en este tipo de situaciones, se agregó un diccionario de abreviaturas en inglés con el fin de detectar posibles abreviaturas en el texto de entrada. Se utilizaron expresiones regulares para detectar siglas, números y abreviaturas candidatas.

La nueva funcionalidad para detectar oraciones quedó conformada por los siguientes pasos:

- **Primero**: realizar una búsqueda de expresiones regulares usando patrones para números.
- **Segundo**: realizar una búsqueda usando patrones para acrónimos.
- **Tercero**: en cada porción de texto, coincidente con un patrón, reemplazar todos los puntos por un carácter comodín. Después de extraer la oración del texto de entrada, el carácter comodín será reemplazado por el carácter original (el carácter punto).
- **Cuarto**: utilizar dos expresiones regulares para buscar abreviaturas, una para buscar abreviaturas dobles, por ejemplo: "Nat. Hist." y otra para buscar abreviaturas simples. En cada caso, la parte correspondiente del texto es verificada con un diccionario de abreviaturas, si hay coincidencias, todos los puntos de ese texto son reemplazados por el carácter comodín. La Tabla 13 muestra las expresiones regulares utilizadas.

Tabla 13. Expresiones regulares utilizadas para detectar puntos no finales

Elemento a detectar	Expresión regular
Números	\d+\.\d+
Acrónimos	(?:[a-zA-Z]\.){2,}
Abreviaciones dobles	\w+\. \w+.
Abreviaciones simples	\w+\.

4.4.1 Nuevos resultados para ClausIE

Se desarrolló una nueva versión de ClausIE[4] utilizando la solución descripta en la sección 4.4 y se ejecutó utilizando el conjunto Reuters-103 como entrada. La precisión y la exhaustividad mejoraron considerablemente. Los nuevos valores obtenidos se muestran en la Tabla 14.

Tabla 14. Métricas calculadas para ClausIE mejorado usando Reuters-103

Medida	Métodos	
	ClausIE	ClausIE mejorado
Precisión	0,467	0,602
Exhaustividad	0,519	0,641
Medida-F1	0,492	0,621

[4] El código fuente y una versión ejecutable del mismo pueden ser descargadas del sitio: https://github.com/juanma1982/clausIEwss

La Figura 3 muestra las diferencias en precisión, exhaustividad y medida-F1 de los métodos ClausIE y ClausIE mejorado utilizando el conjunto Reuters-103 como conjunto de pruebas.

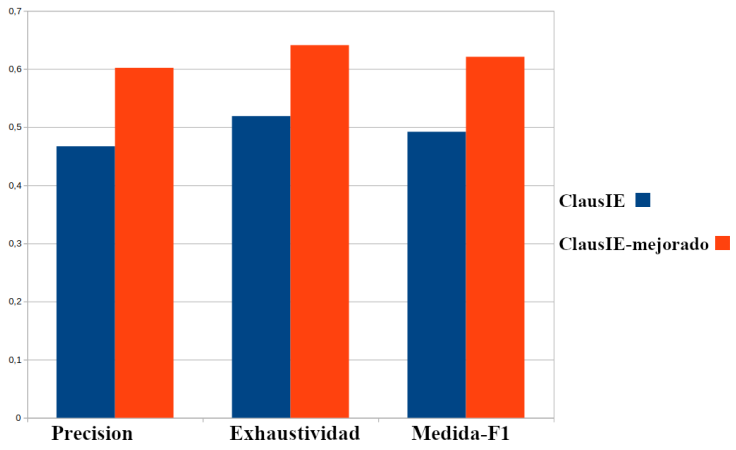


Figura 3: Diferencia de métricas entre ClausIE original y mejorado

Como resultado adicional de la nueva funcionalidad agregada para dividir el texto en oraciones, el tiempo que tarda ClausIE en procesar un texto de entrada se ha mejorado considerablemente. Es difícil medir el tiempo promedio que tarda ClausIE en procesar un texto de entrada ya que depende de la extensión del mismo. Cuanto más largo sea un texto, más complejo será el árbol de dependencias sintácticas que construye ClausIE. El texto más largo en el conjunto Reuters-103 es un párrafo de 248 palabras y 1620 caracteres. Éste fue procesado por ClausIE en 6 minutos y 13,914 segundos. Sin embargo, la nueva versión de ClausIE demoró sólo 16,6 segundos para procesar el texto completo. Si bien no era el objetivo principal mejorar los tiempos de procesamiento de ClausIE, la mejora en la velocidad cobra relevancia si se tiene en cuenta que este método está pensado para trabajar con grandes volúmenes de datos, típicamente en la Web. Si bien en el trabajo [Rodríguez et al., 2017] se demuestra que ClausIE es un método útil para automatizar tareas manuales como la extracción de características en análisis de sentimientos, su viabilidad decrece en tareas en donde se tienen que procesar textos no previamente separados en oraciones, como los del conjunto Reuters-103.

El tiempo se midió con la herramienta de comando *time* de Linux, de la siguiente manera:

```
time ./clausie.sh -f  18745.txt
```

Las especificaciones de la computadora y la versión de Java utilizadas en este experimento se resumen en la Tabla 15.

Tabla 15. Especificaciones de la PC usada para medir el tiempo de ClausIE

Componente	Valor
CPU	AMD Phenom™ 8450 Triple-Core Processor
RAM	4 Gb RAM DIMM DDR2 Síncrono 333 MHz
Versión SO	Linux Mint 17.2 Rafaela
Versión del kernel Linux	GNU/Linux 3.16.0-38-generic x86_64
Versión de Java	Java version "1.7.0_80". Java HotSpot(TM) 64-Bit Server VM (build 24.80-b11, mixed mode)

Para concluir esta sección sobre la evaluación de los métodos existentes en el estado del arte y por lo tanto, candidatos a ser los mejores en términos de las métricas utilizadas: precisión, exhaustividad y medida-F1 (al menos hasta el año 2018), es posible afirmar que los valores propuestos por los distintos autores para sus métodos son aproximadamente los mismos que se obtuvieron en el trabajo de [Rodríguez et al., 2018] con excepción de ClausIE y MinIE. Sin embargo, como se probó en la sección 4.4, la diferencia observada se debe a que ClausIE hace un mal manejo de textos con múltiples oraciones. MinIE por su parte está basado en ClausIE y arrastra los mismos problemas.

5. Problemas abiertos

Hay diversos problemas con los sistemas de extracción de conocimiento para la Web, los más significativos son cómo incrementar la precisión y la exhaustividad. Otros problemas usuales son la extracción de relaciones semánticas poco informativas y el manejo de información subjetiva. Estos problemas serán discutidos en las siguientes secciones.

5.1 Precisión y exhaustividad

La precisión, la exhaustividad y la medida-F1 fue calculada para 4 métodos en el estado del arte en el punto 3: ReVerb, OLLIE, ClausIE y MinIE, los resultados pueden observarse en la Tabla 12. Allí se muestra que los máximos obtenidos apenas superan el 60% para cualquiera de estas medidas, por lo que existe un margen de casi 40 puntos porcentuales hasta que alguno de estos métodos logre igualar a un ser humano.

Entre los métodos más recientes, relevados en el punto 4.3 al momento de escribir esta tesis, se encuentran algunos que logran, según sus autores, una precisión superior al 90 por ciento, como por ejemplo BioOpenIE [Wang et al., 2018] pero en este caso se trata de un método que funciona sólo en un dominio específico. Otros métodos como SpanOIE que logra, según sus autores, una medida F1 de más de 0,73 al ser evaluado en otros conjuntos de datos, como los propuestos en [Kolluru et al., 2020a], obtiene una medida F1 de entre 0.319 y 0.540, dependiendo del conjunto. Lo que sucede con SpanOIE, que su desempeño dependa del conjunto de datos de entrada, ocurre, en general, con la mayoría de los métodos.

En la Tabla 8 de la sección 3.3.1 se muestra una comparativa de varios métodos de *Open IE* en el estado del arte, tomada del trabajo de [Kolluru et al., 2020a]. En dicha tabla se muestran los resultados de nueve métodos diferentes, evaluados en cuatro conjuntos de datos. Los métodos más antiguos que aparecen allí son ClausIE y MinIE y el más moderno es OpenIE6. En la Tabla 8 se pueden notar dos cosas interesantes: la primera es que ClausIE logra mejores resultados que MinIE en todos los conjuntos evaluados y la segunda es que la medida F1 más alta alcanzada por un método es de 0,656.

A pesar de que pasaron varios años desde la publicación de ClausIE (2013), parece que es muy pequeño el avance logrado en la mejora de la precisión y de la exhaustividad de los distintos métodos de *Open IE*. Según los trabajos relevados el límite superior alcanzado para la medida F1 es un valor que supera a 0.6 pero no llega a 0.7. Sólo en dominios particulares o con conjuntos de datos específicos algún método logra posicionarse por encima de este límite. Por lo cual, ampliar la precisión y la exhaustividad es el mayor problema abierto para esta familia de métodos.

5.2 Relaciones semánticas poco informativas

Algunas tuplas extraídas pueden ser correctas, es decir que corresponden a una relación semántica presente en una oración dada, pero que sin embargo, son poco o nada útiles ya que aportan poca o ninguna información relevante. Esto ya se mencionó en el punto 3.2 y se ilustró con el texto del Ejemplo 3, en donde se mostró que la siguiente extracción es válida pero poco informativa:

- (February, has, budget)

La extracción anterior corresponde a un ejemplo real, la misma fue hecha por ClausIE. El problema es menos grave que el problema de la precisión mencionado en la sección 5.1, ya que aquél consiste en no tener relaciones semánticas válidas en absoluto y en cambio este problema puede ser mitigado descartando las relaciones poco informativas en un análisis posterior o bien éstas podrían ser incorporadas a una base general de conocimiento sin que se modifiquen hechos importantes. El problema principal es que si este tipo de extracciones se computan como válidas, pueden distorsionar los valores reales de precisión y exhaustividad de un método. Idealmente, si extracciones poco informativas como la del ejemplo pueden ser detectadas, el método debería descartarlas para generar *piezas de conocimiento* de mejor calidad, según la definición dada en [García-Martínez y Britos, 2004; Gómez et al., 1997].

5.3 Manejo de información subjetiva

Considérese el Ejemplo 7 y la siguiente extracción:

Early astronomers believed that the earth is the center of the universe. [7]

- (earth, is, the center of the universe)

Esta extracción es correcta desde el punto de vista sintáctico. Pero la información allí presente no es objetiva, corresponde a una opinión. Es información de tipo no-fáctica o subjetiva. Este tipo de extracciones no son tenidas en cuenta de forma particular por métodos como Reverb o ClausIE. Sí es manejada por métodos más nuevos como OLLIE y MinIE.

Por ejemplo MinIE anota cada extracción con información acerca de su *factualidad*. MinIE representa la *factualidad* de una extracción con dos piezas de información: polaridad (+ o -) y modalidad (CT o PS; para indicar certeza o posibilidad) [Gashteovski et al., 2017].

5.4 Autonomía de los métodos

Uno de los problemas más interesantes entre los métodos relevados hasta el año 2018 era la autonomía de los mismos, la mayoría de ellos: ReVerb, OLLIE, ClausIE y MinIE, por ejemplo, descansan en la lógica original dada por cada uno de sus desarrolladores. Suele ser por lo general una lógica compleja, difícil de seguir y por lo tanto difícil de mejorar. Quizás como excepción se puede citar a ReVerb que tiene una lógica bastante simple. En todo caso, son métodos con poca autonomía, con poca capacidad para transcender su programación. La propuesta obvia es construir métodos que aprendan a través de ejemplos, para de esta forma, mantener una lógica dinámica que pueda ser mejorada a lo largo del tiempo. Una propuesta más avanzada sería la de construir un método capaz de ir aprendiendo en tiempo de ejecución, es decir, en modo productivo.

Este problema ha sido mitigado en los métodos más modernos, relevados en la sección 3.3 ya que muchos utilizan redes neuronales y construyen la lógica interna

utilizando conjuntos de entrenamiento. El método OpenIE6 también utiliza un conjunto de entrenamiento para construir la lógica de su funcionamiento.

5.5 Métodos en lenguaje español

Todos los problemas señalados en las secciones 5.1, 5.2, 5.3 y 5.4 son también problemas presentes en los métodos de extracción de conocimiento en lenguaje español, ya que éstos son apenas un subconjunto de aquella familia. Los métodos de extracción de conocimiento para la Web en lenguaje español se enumeran en la Tabla 9 y son los siguientes: ExtrHech, ArgOE, DepOE y Multi^{2}OIE. En la Tabla 16 se muestran la precisión, la exhaustividad y la medida F1 de cada uno de estos métodos según mediciones realizadas por sus respectivos autores, en los artículos en donde fueron presentados.

Tabla 16. Métodos de *Open IE* en idioma español y sus medidas de rendimiento

Método	Artículo	Precisión	Exhaustividad	Medida F1
DepOE	[Gamallo y Garcia, 2012]	0,68	0,38	0,49
ExtrHech	[Zhila y Gelbukh, 2014]	0,55	0,49	0,52
ArgOE	[Gamallo y Garcia, 2015]	0,50	–	–
Multi^{2}OIE	[Ro et al., 2020]	0,59	0,61	0,60

En la Tabla 16 se puede ver que aún tomando en consideración los valores calculados por los autores de cada método, éstos apenas sí logran superar el 60% en cualquiera de las 3 métricas en que fueron evaluados. Sin embargo, para entender mejor estos valores, hay que tener en cuenta lo siguiente: DepOE en el artículo de [Gamallo y Garcia, 2012] es evaluado en idioma inglés y no en español ya que el método es considerado multilenguaje, categoría que también se aplica a ArgOE y Multi^{2}OIE. Por otro lado, Multi^{2}OIE fue evaluado en idioma español pero utilizando un conjunto de datos de prueba en inglés llamado Re-OIE2016 [Ro et al., 2020] el cual fue traducido al español. Esta traducción fue realizada de forma automática por un producto de Google[5], aunque los autores tuvieron cuidado al realizar la traducción ya que modificaron las oraciones traducidas para que éstas no perdiesen su significado original al ser traducidas de forma inversa, nuevamente a idioma inglés. A pesar de este cuidado, no deja de ser una prueba sesgada, en donde las nuevas oraciones en idioma español tendrán quizás una estructura sintáctica similar a sus

[5] https://cloud.google.com/translate

equivalentes en idioma inglés y en donde aparecerán secuencias de palabras similares que no necesariamente serán representativas o de uso frecuente en idioma español. En este conjunto de datos de prueba ArgOE también fue evaluado y su precisión fue ligeramente menor a la ya calculada: 0,48, su exhaustividad fue de 0,33 y su medida F1 0.39.

Las relaciones semánticas poco informativas son un problema vigente en los métodos en idioma español. En pruebas realizadas con métodos en español se detectó extracciones válidas pero poco informativas. Considérese el Ejemplo 8 y la siguiente extracción:

El mundo era tan reciente que muchas cosas carecían de nombre, y para nombrarlas había que señalarlas con el dedo. [8]

- (El mundo, era tan reciente que, muchas cosas)

Se trata de una extracción real realizada por ArgOE y también por DepOE, ambos métodos han extraído la misma relación semántica que si bien parece correcta al ser incompleta es poco útil.

Por otro lado, ninguno de los métodos para idioma español listados en la Tabla 16 cuenta con mecanismos para la detección de información subjetiva, o como es llamada también: información no fáctica.

Finalmente, de los métodos listados en la Tabla 16 solo Multi^{2}OIE es un método capaz de cierta autonomía al estar basado en redes neuronales. Su desempeño es en gran parte producto de los ejemplos utilizados en su entrenamiento. Sin embargo, descansa fundamentalmente en BERT [Devlin et al., 2019]. BERT es un modelo de representación del lenguaje que está preentrenado y que se puede ajustar con sólo una capa de salida adicional para crear nuevos modelos, útiles en una amplia gama de tareas relacionadas con procesamiento de lenguaje natural.

6. Soluciones propuestas

En la siguiente sección se describen las soluciones propuestas a los problemas abiertos enumerados en la sección 5.

6.1 TP-OIE (*Tree Pattern Open Information Extraction*)

TP-OIE es el primer método propuesto como una mejora a los existentes con la intención de solucionar o mitigar los problemas descriptos en la sección 5. TP-OIE no fue publicado pero sirvió como base para la creación otros 3 métodos que sí fueron publicados: ATP-OIE [Rodríguez et al., 2020], TP-OIE-ES [Rodríguez y Merlino, 2020] y ECMes [Rodríguez et al., 2021], estos dos últimos para lenguaje español.

La idea principal de TP-OIE es la de encontrar patrones en el árbol de dependencias sintácticas para una oración dada. La idea de usar *parsers* de dependencias sintácticas no es nueva, ClausIE utiliza el *parser* de Standford para crear el árbol de dependencias sintácticas de un texto dado como entrada. La diferencia con TP-OIE es que este último busca en el árbol patrones aprendidos previamente, durante una primera etapa de entrenamiento. Al generar conocimiento a partir de ejemplos se busca aumentar la autonomía del método según se describió en la sección 5.4.

6.1.1 Proceso de aprendizaje

Este proceso de entrenamiento o aprendizaje no es un paso obligatorio ya que el método fue entrenado por los autores de antemano utilizando una base de datos con ejemplos. Si bien no es necesario un reentrenamiento para que TP-OIE sea capaz de extraer relaciones semánticas, siempre se pueden agregar nuevos ejemplos al conjunto de entrenamiento para que TP-OIE mejore su proceso de extracción o bien se lo puede entrenar para textos de un dominio específico. Aquí radica la autonomía del método.

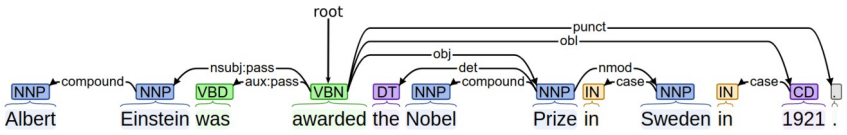
La base de datos de ejemplos consiste en un archivo JSON con el formato que se muestra en el Ejemplo 9. Este objeto JSON tiene un solo atributo llamado *examples*, el cual es un *array* de objetos JSON. Cada uno de estos objetos, contiene dos atributos: *sentence* y *relations*. El primero es una oración de ejemplo y el segundo otro *array* de objetos JSON, en donde cada objeto del *array* representa una relación semántica existente en la oración dada. Puede haber varias como se muestra en el

Ejemplo 9. Cada uno de estos objetos contiene tres atributos: *entity1*, *relation* y *entity2*. El primero de estos atributos es el argumento primero o sujeto de la relación, el atributo *relation* contiene a la relación propiamente dicha y por último *entity2* es el argumento segundo.

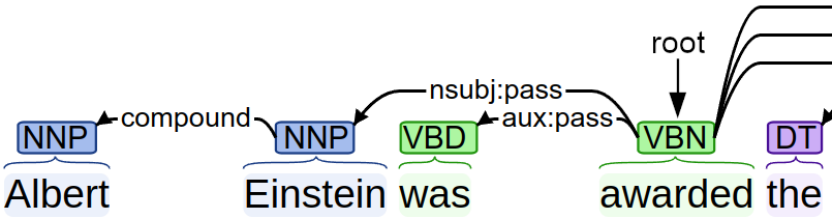
```
{
   lang: "en",
   examples:[ {
      sentence: "Albert Einstein was awarded the Nobel Prize
                    in Sweden in 1921.",
      relations: [
        {
          entity1: "Albert Einstein",
          relation: "was awarded",
          entity2: "the Nobel Prize"
        },
        {
          entity1:"Albert Einstein",
          relation:"was awarded",
          entity2:"in Sweden"
        },
        {
          entity1: "Albert Einstein",
          relation: "was awarded",
          entity2:  "in 1921"
        }
      ]
   },…
```

[9]

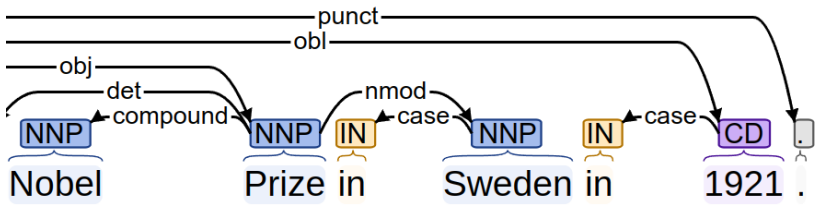
Para cada uno de los ejemplos en el *array* TP-OIE ejecutará un *parser* de dependencias sintácticas para obtener el árbol de dependencias de la oración de ejemplo. El *parser* utilizado por TP-OIE es *depparse* de la biblioteca *Stanford CoreNLP* [Danqi y Christopher, 2014]. En el mismo proceso TP-OIE identificará los nombres de entidades (NER, por sus siglas en inglés) de la oración usando la anotación “ner” de la misma biblioteca *Stanford CoreNLP*. Una representación gráfica del árbol generado para la oración del Ejemplo 9 se muestra en la Figura 4.



a. representación total



b. detalle de la parte izquierda



c. detalle de la parte derecha

Figura 4: Representación gráfica del árbol de dependencias sintácticas para la oración del Ejemplo 9.

Nota: En la primera imagen (**a**), vemos la totalidad del árbol de dependencias sintácticas. En la imagen (**b**) se ve un detalle de la parte izquierda y en la imagen (**c**) un detalle de la parte derecha.

De acuerdo con Fader en [Fader et al., 2011] conviene siempre encarar la extracción de relaciones semánticas empezando por la relación propiamente dicha, entonces para cada unigrama en la relación de un ejemplo dado, TP-OIE intentará extraer la ruta (o *path* en inglés) en el árbol de dependencias sintácticas. Siendo esta ruta la secuencia de aristas en el árbol, desde la raíz (en este caso el verbo: *awarded*) hasta el unigrama en cuestión. Por ejemplo, el unigrama: *Nobel* tiene la siguiente ruta: *root* → *obj* →*compound*.

Continuando con el Ejemplo 9, la relación *was awarded*, de la primer relación semántica, tiene dos unigramas *was* y *awarded*, la ruta de cada uno en el árbol es la siguiente:

- **was**: *root → auxpass*
- **awarded**: *root*

El siguiente paso para construir un patrón es añadir información adicional a cada unigrama, si éste fue identificado como el nombre de una entidad (NER) se agrega la anotación "ner", por ejemplo: "ner=PERSON". Si no es una entidad, se utiliza la categoría gramatical (POS), por sus siglas en inglés *part of speech*. En este caso, se añade la anotación "tag" con la categoría gramatical de la palabra, según la definición del conjunto de etiquetas *Penn Treebank POS* dada en [Ratnaparkhi, 1996]. Estas categorías gramaticales son las utilizadas por el *parser Stanford CoreNLP*. Por ejemplo "tag=VBN" para indicar que el unigrama es un verbo. Sin embargo, cuando la categoría gramatical es "IN" o "TO", se utiliza la etiqueta "word" y se le asigna directamente el unigrama correspondiente. Continuando con el Ejemplo 9 el patrón aprendido para la relación quedaría así:

- root →auxpass [tag=VBD]
- root [tag=VBD]

El patrón para el *entity1*, es decir el argumento primero quedaría como:

- root→nsubj:pass→compound[ner=PERSON]
- root→nsubj:pass[ner=PERSON]

Dado que un patrón está conformado por las rutas de uno o más unigramas, dependiendo de la oración de ejemplo utilizada para crearlo, es posible que un patrón incluya parte de otro. Si en la oración del Ejemplo 9 en lugar de "Albert Einstein" dijese sólo "Einstein" el patrón generado para la *entity1* sería sólo:

- root→nsubj:pass[ner=PERSON]

Este patrón está contenido en el anterior. A su vez, si en otra oración de ejemplo con sintaxis similar, apareciese un nombre de persona con 3 unigramas, posiblemente obtendríamos un patrón que contuviese a los anteriores. Es por ello que los patrones se van agrupando en una organización de árbol a medida que son descubiertos. Para ilustrar este punto considérese la Figura 5.

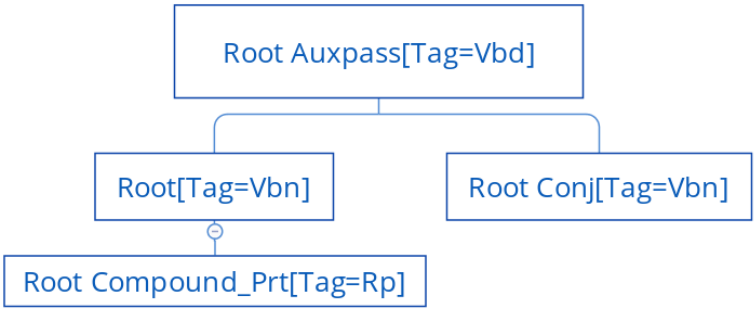


Figura 5: Árbol de patrones para detectar relaciones.

Cada nodo en el árbol puede estar marcado como nodo terminal, es decir que en él termina un patrón encontrado. Por lógica, las hojas son todos nodos terminales. Una vez que termina la fase de entrenamiento, la base de datos de patrones queda almacenada como una lista de árboles indexada por el patrón raíz de cada elemento.

Una vez obtenido el patrón correspondiente a la relación para el ejemplo que se está analizando, se procede a extraer el patrón correspondiente al argumento 1. Este paso es similar al anterior. Los argumentos primeros se extraen utilizando la misma lógica. La única diferencia entre estos patrones y los patrones para extraer *relaciones* es que estos otros se almacenan referenciados a un patrón de *relación*. De esta forma al momento de producirse una coincidencia entre un patrón de *relación*, no se realiza una búsqueda exhaustiva en la lista de patrones de *argumentos primeros* sino que sólo se prueban los patrones asociados, lo cual es más eficiente.

Finalmente todos estos patrones obtenidos quedan almacenados en un archivo JSON. Este archivo es cargado cada vez que TP-OIE inicia en modo de ejecución. En el Ejemplo 10 se muestra como se ve una porción de este archivo.

[10]

```
{
 "subjects": {… }
  "relations": {
   "root auxpass[tag=VBD]": {
    "patternStr": "root auxpass[tag=VBD]",
    "isLeaf": true,
    "nextPatterns": {
     "root[tag=VBN]": {
      "patternStr": "root[tag=VBN]",
      "isLeaf": true,
      "nextPatterns": {
        "root compound_prt[tag=RP]": {
          "patternStr": "root compound_prt[tag=RP]",
```

```
            "isLeaf": true,
            "nextPatterns": {}
          }
        }
},  …
```

6.1.2 Conjunto de entrenamiento

TP-OIE fue entrenado principalmente utilizando las extracciones correctas de ClausIE en los 3 conjuntos de datos propuestos por [Del Corro y Gemulla, 2013] y que fueron descriptos en la sección 4.3. Todos conjuntos de oraciones en idioma inglés. Se agregaron además 12 ejemplos creados a mano que fueron los utilizados en las etapas tempranas de desarrollo para realizar pruebas y validaciones. Finalmente el conjunto de entrenamiento quedó conformado de la siguiente forma:

- 131 oraciones de las 200 tomadas aleatoriamente por [Del Corro y Gemulla, 2013] del subconjunto *New York Times collection* [Sandhaus, 2008] y las respectivas relaciones semánticas correctas extraídas por ClausIE.
- 102 oraciones seleccionadas de las 200 oraciones tomadas de forma aleatoria por [Del Corro y Gemulla, 2013] de páginas de Wikipedia y las respectivas relaciones semánticas correctas extraídas por ClausIE.
- 322 oraciones seleccionadas de las 500 oraciones tomadas de forma aleatoria de un servicio de Yahoo llamado: *Yahoo's random link* por [Fader et al., 2011] (conjunto ReVerb) y las correspondientes relaciones semánticas extraídas por ClausIE en dicho conjunto.
- 12 oraciones en inglés creadas de forma manual originalmente con fines de prueba.

El número total de oraciones en el conjunto de entrenamiento fue de 567 y el número total de diferentes relaciones semánticas es 1425.

6.1.3 Proceso de extracción

El proceso de extracción es el proceso por el cual TP-OIE extrae relaciones semánticas de textos dados como entrada. Consta de los siguientes pasos, para cada texto de entrada:

1 Divide cada párrafo en oraciones. En este punto se utiliza la misma funcionalidad descripta en la sección 4.4.

2 Para cada oración ejecuta el *parser* de dependencias sintácticas y un *parser* de análisis sintáctico superficial (*NP-Chunking*). El *parser* de dependencias sintácticas es el mismo utilizado en el proceso de aprendizaje. El *parser* de *NP-chunking* es el de biblioteca *OpenNLP*, el mismo utilizado por *ReVerb* [Fader et al., 2011].

3 TP-OIE genera internamente un XML en memoria que representa el árbol de dependencias sintácticas enriquecido con las etiquetas *ner*, *tag* y *word* descriptas en 6.1.1. Éstas indican el tipo de entidad, la categoría gramatical y la palabra. La razón de construir un XML es la de poder utilizar JSoup[6] una poderosa herramienta de Java para poder encontrar patrones en árboles, habitualmente en el árbol DOM de HTML. El XML generado para la oración del Ejemplo 9 puede verse en la Figura 6.

4 Luego para cada patrón relación en la lista interna, generada durante el proceso de entrenamiento, JSoup intenta buscar una coincidencia en el árbol XML. Si un elemento raíz coincide en la lista de patrones, continua buscando coincidencias en los nodos hijos hasta encontrar todos los nodos hijos que estén marcados como nodos terminales. De esta forma obtiene una serie de relaciones candidatas.

5 Para cada una de las relaciones candidatas, obtiene la lista asociada de patrones para detectar argumentos primeros. Con cada uno de los patrones obtenidos procede de igual forma que en el punto 4.

```
▼<root word="awarded" tag="VBN" ner="O">
 ▼<nsubjpass word="Einstein" tag="NNP" ner="PERSON">
    <compound word="Albert" tag="NNP" ner="PERSON"> </compound>
  </nsubjpass>
  <auxpass word="was" tag="VBD" ner="O"> </auxpass>
 ▼<dobj word="Prize" tag="NNP" ner="MISC">
    <det word="the" tag="DT" ner="O"> </det>
    <compound word="Nobel" tag="NNP" ner="MISC"> </compound>
   ▼<nmod word="Sweden" tag="NNP" ner="COUNTRY">
      <case word="in" tag="IN" ner="O"> </case>
    </nmod>
  </dobj>
 ▼<nmod word="1921" tag="CD" ner="DATE">
    <case word="in" tag="IN" ner="O"> </case>
  </nmod>
  <punct word="." tag="." ner="O"> </punct>
</root>
```

Figura 6: XML generado en memoria.

6 TP-OIE se queda con las relaciones para las cuales encontró argumentos primeros candidatos. Procede luego para cada par: relación, argumento primero a buscar un argumento segundo.

7 El método de extracción de argumentos segundos es diferente. En este punto TP-OIE utiliza el mismo enfoque que ReVerb en [Fader et al., 2011]. Intenta encontrar la frase nominal más cercana a la derecha de la relación en la oración. Si el argumento primero está a la derecha de la relación busca la

[6] https://jsoup.org/

frase nominal más cercana a izquierda. Para obtener la frase nominal utiliza el *parser* de análisis sintáctico superficial (*NP-Chunking*).

6.1.4 Problemas encontrados con este enfoque

La estrategia utilizada para mejorar la exhaustividad en este primer intento fue la de ir agregando más y más ejemplos al conjunto de entrenamiento, con lo cual se fueron generando también más y más patrones de extracción. Al principio se tenía una precisión muy alta, cercana al 70% pero con una exhaustividad muy pobre de apenas 15%. Finalmente, luego de agregar varios ejemplos en varios entrenamientos se logró subir la exhaustividad, aunque no demasiado y por otro lado la precisión cayó considerablemente. En esta primera versión los números fueron variando mucho en la medida en que se agregaban ejemplos, o se refinaba alguno existente pero, en general, fueron resultados pobres.

Si bien el algoritmo principal parecía tener potencial, este método no logró solucionar el principal problema descripto. Su precisión y su exhaustividad fueron considerablemente bajas. Entre los problemas identificados se encuentran la imposibilidad para detectar correctamente argumentos primeros utilizando la lista de patrones aprendidos, la generación de extracciones duplicadas, relaciones semánticas extraídas poco informativas y sobre todo un gran número de extracciones inválidas. Todos estos problemas serán mitigados o solucionados en las próximas versiones que se describen en las siguientes secciones.

6.2 ATP-OIE (*Autonomous Tree Pattern Open Information Extraction*)

En esta sección se describe como ATP-OIE logró corregir alguno de los errores encontrados en TP-OIE y también manejar los errores generales de los métodos de extracción de conocimiento para la Web descriptos en la sección 5. ATP-OIE fue presentado en la publicación de [Rodríguez et al., 2020].

6.2.1 Precisión y exhaustividad

Una de las principales modificaciones sobre el método anterior se agregó en el paso 5, cuando el método intenta extraer el argumento primero de la oración utilizando la lista de patrones. Para muchas oraciones el método no lograba encontrar un patrón de coincidencia que le permitiese identificar el argumento primero. Para solucionar este problema ATP-OIE intenta buscar, como argumento primero, la frase nominal más cercana a la izquierda de la relación, si es que no logró encontrarla

utilizando la lista de patrones. Para encontrar esta frase nominal, se utiliza el resultado del *parser* superficial *NP-chunking* de la biblioteca *OpenNLP*. Este resultado es el mismo que se usa para encontrar al argumento segundo según se describió en el paso 7 de la sección 6.1.3. Para ilustrar el funcionamiento de este mecanismo, supóngase la oración del Ejemplo 9:

- *Albert Einstein was awarded the Nobel Prize in Sweden in 1921.*

El *parser* superficial devolverá por un lado las etiquetas gramaticales de cada una de las palabras en la oración (nomenclatura de *Penn Treebank Project*[7]):

- NNP NNP VBD VBN DT NNP NNP IN NNP IN CD .

En donde, cada etiqueta significa lo siguiente:

- **NNP:** sustantivo propio en singular (*Proper noun, singular*).
- **VBD:** verbo en pasado.
- **VBN:** verbo en participio pasado.
- **DT:** determinante.
- **IN:** preposición o conjunción subordinante.
- **CD:** número.

Y por otro lado devolverá un etiquetado secuencial sobre las palabras de la oración para indicar donde comienzan o terminan las frases verbales, nominales o preposicionales. Continuando con el ejemplo anterior, el resultado del etiquetamiento secuencial sería el siguiente:

- B-NP I-NP B-VP I-VP B-NP I-NP I-NP B-PP B-NP B-PP B-NP

En donde cada etiqueta significa lo siguiente:

- **B-NP**: comienza frase nominal.
- **I-NP**: continúa frase nominal.
- **B-VP**: comienza frase verbal.
- **I-VP**: continúa frase verbal.
- **B-PP**: comienza frase preposicional.
- **I-PP**: continúa frase preposicional.

El resultado de este *parser*, palabra por palabra, puede verse de forma más clara en la Tabla 17. Utilizando estos resultados resulta sencillo para el método detectar cual es la frase nominal más cercana a la relación tanto a derecha como a izquierda. Suponiendo que para la oración de ejemplo se haya detectado, utilizando la lista de

[7] https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

patrones, la secuencia de unigramas: *was awarded* como una relación candidata y que no se hubiese detectado un argumento primero utilizando la correspondiente lista de patrones, entonces el método procederá de la siguiente forma:

1 Buscará la coincidencia de los unigramas de la relación, según su posición dentro de la oración, en el resultado del etiquetamiento secuencial.
2 Una vez posicionado en el primer unigrama de la relación se moverá una posición a izquierda y verá qué tipo de etiqueta es. Repetirá este movimiento hasta encontrar una etiqueta I-NP o B-NP.
3 En este ejemplo la etiqueta a izquierda es I-NP, eso indica que hay una frase nominal que continua. Se moverá entonces hacia la izquierda hasta encontrar la etiqueta de inicio (B-NP).
4 Una vez obtenida la frase nominal, y sabiendo su posición, se quedará con los unigramas correspondientes de la oración. En este ejemplo éstos corresponden a la secuencia: *Albert Einstein*.

Tabla 17. Oración del Ejemplo 9 junto con la salida del *parser* superficial.

	Oración analizada										
U	Albert	Einstein	was	awarded	the	Nobel	Prize	in	Sweden	in	1921
CG	NNP	NNP	VBD	VBN	DT	NNP	NNP	IN	NNP	IN	CD
ES	B-NP	I-NP	B-VP	I-VP	B-NP	I-NP	I-NP	B-PP	B-NP	B-PP	B-NP

U: unigrama, **CG**: Categoría gramatical, **ES**: Etiquetado secuencial

Con esta sencilla mejora respecto de TP-OIE se logró aumentar considerablemente la cantidad de relaciones semánticas obtenidas.

6.2.1.1 Puntaje de las extracciones realizadas

Para descartar las extracciones erróneas y mejorar así la precisión se implementó un sistema de puntaje para cada extracción. El sistema es similar al que utiliza ReVerb [Fader et al., 2011] con algunas modificaciones. La principal de ellas es que le asigna un puntaje muy malo a extracciones en donde el argumento es una sola palabra y ésta es un determinante, un pronombre posesivo o un conector entre frases. Otras validaciones adicionales que se agregaron fueron las siguientes: que los argumentos 1 y 2 sean distintos, que la relación no esté dentro del argumento 1, que el argumento 1 no termine con la misma palabra con la cual comienza la relación. En la Tabla 18 se muestran la totalidad de reglas utilizadas.

Tabla 18. Reglas utilizadas por ATP-OIE para puntuar una relación semántica

Regla	Puntaje
a1+**r**+**a2** incluidos en **s** y longitud(**s**) ~= longitud(**a1**+**r**+**a2**)	116
Última palabra de **r** es *FOR*	50
Última palabra de **r** es *ON*	49
Última palabra de **r** es *OF*	46
Cantidad palabras en **s**< 11	43
Cat. gramatical de la palabra a izquierda de **r** empieza con W	43
r cumple con la regla: **VW*P**, es decir que tiene 3 o más palabras y la primera es un verbo o adverbio, la segunda es sustantivo, adjetivo, adverbio, pronombre o determinante y la última es una preposición o una partícula gramatical.	42
Última palabra de **r** es *TO*	39
Última palabra de **r** es *IN*	25
Cantidad palabras en **s**<21	23
s comienza con **a1**	21
a2 es un sustantivo propio	16
a1 es un sustantivo propio	1
A la izquierda de **a1** hay una frase nominal	-30
a1 termina con la misma palabra con que empieza **r**	-50
Si **r** es una frase de 3 palabras y estas tienen las siguientes categorías gramaticales: verbo, partícula gramatical y adverbio	-61
Si a la izquierda de **a1** hay una preposición	-65
Si hay una frase nominal a la derecha de **a2**	-81
Si la palabra a la izquierda de **r** es una conjunción	-93
Si **a1** es igual a **a2**	-100
Si **a2** es *IT*	-100
r está incluida completamente en **a1**	-100
a2 es un determinante	-1000

Nota: **s** es la oración original dada como entrada, **a1** es el argumento primero, **r** es la relación y **a2** es el argumento segundo.

Utilizando este sistema de puntajes se logra descartar muchas extracciones erróneas y aumentar considerablemente la precisión. Cabe mencionar que los valores asignados a los puntajes son los mismos que utiliza ReVerb en [Fader et al., 2011], y las nuevas reglas adicionadas solo añadieron puntajes con valores negativos por debajo de los ya existentes. La última regla añade un valor negativo exageradamente alto para descartar de lleno las relaciones semánticas que coinciden con ella.

6.2.1.2 Utilización de métodos auxiliares

Una forma de aumentar la exhaustividad de ATP-OIE fue la de incorporar en su programación los métodos ReVerb y ClausIE. Cuando ATP-OIE no encuentra una

relación semántica en una oración dada o bien las que encuentra son descartadas por el sistema de puntaje, es capaz de invocar primero a ReVerb y luego a ClausIE. Si ReVerb tampoco es capaz de encontrar una relación semántica válida o las que encuentra son descartadas nuevamente por el sistema de puntaje, entonces se invoca a ClausIE. Este mecanismo permitió aumentar la cantidad de extracciones realizadas en un 35.4 %. En las pruebas efectuadas se pasó de 647 a 876 extracciones.

6.2.1.3 Aprendizaje en línea

La razón principal de incorporar métodos auxiliares fue la de probar la capacidad de ATP-OIE para generalizar a partir de ejemplos mientras está siendo ejecutado en su modalidad de extracción. Esta forma de aprendizaje en línea funciona de la siguiente manera: cuando ATP-OIE no logra encontrar una relación semántica en una oración dada, pero sí es encontrada por uno de los métodos auxiliares (ReVerb o ClausIE) y esta relación supera un umbral de 45 puntos, ATP-OIE crea un ejemplo similar a los utilizados en la fase de entrenamiento, descripta en la sección 6.1.1. A partir de dicho ejemplo, genera los patrones de extracción asociados y los incorpora a su base de datos de conocimiento. De esta forma, si aparece una oración similar más adelante debería ser capaz de extraer la relación semántica directamente sin recurrir a sus métodos auxiliares. En esta modalidad de funcionamiento, ATP-OIE logra extraer algunas relaciones semánticas más que en el caso anterior aunque no muchas más, apenas un 1.3%, pasando de 876 relaciones semánticas extraídas a 887. Si se mira en cambio las relaciones semánticas extraídas sólo por ATP-OIE (sin contar las extraídas por los métodos auxiliares) éstas aumentaron un 3.9%, de 647 a 672, según las mediciones realizadas en el conjunto de datos de prueba Reuters-103.

6.2.2 Extracciones poco informativas

No hay en principio una forma infalible de saber si una relación semántica es o no poco informativa. La estrategia propuesta por ATP-OIE para mitigar este problema fue la de añadir más información en los argumentos. Cuando un argumento candidato es detectado, ya sea primero o segundo, ATP-OIE verifica si existe un conector que lo vincule a una frase nominal. Los conectores son, en este caso, las palabras (en inglés): *to*, *of* y *at*. Estas palabras en idioma inglés corresponden a la categoría gramatical IN, según el *parser* de Stanford *CoreNLP*, el cual utiliza las etiquetas gramaticales de *Penn Treebank*. Estos conectores suelen añadir información conectando dos frases nominales. En términos generales se puede decir que se trata de información sobre el

lugar (generalmente luego del conector *at*), de información sobre el propósito de algo, o del destino (conector *to*) o bien información sobre el tipo, la categoría o cualquier especificación adicional dada por el conector *of.* Por supuesto estas palabras no son los únicos conectores de uso frecuente en inglés, están también: *with*, *by*, *in*, etc. Pero un análisis rápido realizado sobre el conjunto Reuters-103 arrojó que estos últimos conectores mencionados suelen formar parte más bien de la relación que de los argumentos, por ejemplo: "*created by*" o "*awarded in"*. De todas formas quedó como un punto de mejora futura, agregar más conectores y ver si permiten aumentar o no la cantidad de información en las relaciones semánticas extraídas. Para ilustrar como funciona esta estrategia considérese la oración del Ejemplo 11, tomada del conjunto Reuters-103.

The Delta was the first of six rockets [11]

ATP-OIE extraerá la siguiente relación semántica:

- (The Delta, was, the first **of** six rockets)

En la extracción del Ejemplo 11, el argumento segundo en vez de ser sólo *the first*, lo cual sería poco informativo, se conecta utilizando la palabra *of* con *six rockets* para añadir información adicional y generar una tupla más completa y útil.

6.2.3 Manejo de información subjetiva

Para detectar información no fáctica o subjetiva ATP-OIE cuenta con dos mecanismos: el primero consiste en reemplazar todo texto entre comillas por una palabra comodín, la cual se analizará como un sustantivo simple. De esta manera se evita confundir al *parser* y que éste mezcle palabras del texto citado con palabras de la oración original. El segundo mecanismo consiste en verificar la existencia de determinadas palabras en la oración de entrada, palabras como: *said, told, added, announced, asserted, believe* o *believed.* Estas palabras son usadas frecuentemente en idioma inglés para expresar que lo que viene a continuación corresponde a lo expresado por un tercero. En este caso todo el texto entre dicha palabra y el siguiente signo de puntuación será reemplazado por la misma palabra comodín.

En ambos casos, la palabra comodín es considerada un sustantivo simple y luego de realizada la extracción de la relación semántica este comodín es reemplazado por el texto original. Por otro lado, la porción del texto original que fue reemplazada por la palabra comodín será considerada una nueva oración de entrada sobre la cual se

realizará otra extracción. La relación semántica extraída de esta porción de texto será marcada como dependiente de la original y, por lo tanto, no fáctica (subjetiva). En la oración del Ejemplo 7, que se vuelve a copiar abajo, ATP-OIE realizará las siguientes extracciones:

Early astronomers believed that the earth is the center of the universe. [7]

- 1 (astronomers, believed, that the earth is the center of the universe) => (23)
- 5 (the earth, is, the center of the universe) => (23) DEPENDS OF 1

En este escenario se agregó a la relación semántica toda la información adicional que genera el método. El primer número a la izquierda de la tupla es el identificador de la relación semántica (ID), en este caso 1 y 5. Los identificadores 2, 3 y 4 pertenecen a tuplas descartadas por el sistema de puntajes. El valor indicado entre paréntesis y continuación de los caracteres "=>", indica el puntaje de la relación semántica, en ambos casos 23. Por último, en la segunda relación semántica extraída se indica que depende de la primera, identificada con el ID 1. La frase "DEPENDS OF" indica que se trata de información subjetiva y que el contexto general de dicha relación semántica se encuentra en la tupla indicada, en este caso la primera.

Para que ATP-OIE muestre toda esta información adicional a la extracción debe ser ejecutado con la opción "-full".

6.2.4 Resultados de ATP-OIE

Para medir el desempeño de ATP-OIE se lo evaluó en el conjunto de datos Reuters-103 conjuntamente con otros 4 métodos: ReVerb, ClausIE, OLLIE y MinIE. MinIE fue ejecutado utilizando la opción "C" (modo completo) con la cual tiene un mejor desempeño según los autores [Gashteovski et al., 2017]. Estos 4 métodos corresponden a los métodos en el estado del arte según el corte hecho hasta el año 2018 y presentado en la sección 3.2. ATP-OIE fue evaluado en sus 3 modalidades:

- **Autónomo** (*standalone*): sin la utilización de los métodos auxiliares ReVerb y ClausIE.
- **Asistido**: con la utilización de los métodos auxiliares ReVerb y ClausIE.
- **Aprendizaje en-línea**: con la utilización de los métodos auxiliares ReVerb y ClausIE y la generación de nuevas reglas de extracción en tiempo de ejecución.

Los resultados obtenidos se muestran en la Tabla 19.

Tabla 19. Medidas calculadas para ATP-OIE, ClausIE, OLLIE, ReVerb y MinIE

Métodos	Precisión	Exhaustividad	Medida F1
ClausIE	0.467	0.519	0.492
OLLIE	0.456	0.416	0.435
ReVerb	0.633	0.319	0.424
MinIE-C	0.612	**0.593**	**0.6022**
ATP-OIE autónomo	0,650	0,294	0,401
ATP-OIE asistido	**0,680**	0,401	0,504
ATP-OIE aprendizaje en-línea	0,670	0,390	0,493

En los resultados de la Tabla 19 puede observarse que ATP-OIE en su modalidad asistido tiene una precisión mayor que los demás métodos, seguido por ATP-OIE aprendizaje en-línea y en tercer lugar por ATP-OIE autónomo. Sin embargo, MinIE tiene la exhaustividad más alta, seguido por ClausIE y luego por ATP-OIE asistido. Finalmente la medida F1 más alta corresponde a MinIE, seguido por ATP-OIE asistido y luego por ATP-OIE aprendizaje en línea.

6.3 TP-OIE-ES (*Tree Pattern Open Information Extraction Español*)

TP-OIE-ES fue presentado en el artículo [Rodríguez y Merlino, 2020], se trata de un método de extracción de conocimiento para la Web en idioma español basado en el método TP-OIE descripto en la sección 6.1. Este método utiliza la misma base de conocimiento que TP-OIE y ATP-OIE, es decir que no fue entrenado una segunda vez sino que el proceso de aprendizaje utilizado es el ya descripto en la sección 6.1.1. TP-OIE utiliza la misma lista de patrones para la detección de relaciones semánticas y de argumentos primeros que TP-OIE. Como el árbol de dependencias generado por el *parser* de dependencias sintáctico *Stanford CoreNLP* es universal, en el sentido de que las aristas que conectan las palabras son siempre las mismas independientemente del idioma [Buchholz y Marsi, 2006], es posible en principio usar los mismos patrones generados para idioma inglés en idioma español. Por otro lado, como las categorías gramaticales utilizadas por el *parser* Stanford CoreNLP en español, conocidas como *Universal POS tags* [Petrov et al., 2011], son distintas a las utilizadas en idioma inglés, conocidas como *Penn Treebank POS tags* [Ratnaparkhi, 1996], se tuvieron

que convertir unas en otras. Esta traducción fue realizada utilizando las equivalencias mostradas en la Tabla 20.

6.3.1 Precisión y exhaustividad

Para mejorar la precisión y la exhaustividad de TP-OIE-ES, respecto de la base provista por TP-OIE, se heredaron dos de las funcionalidades adicionales de ATP-OIE. La primera, es la posibilidad de recuperar la frase nominal a la izquierda de la relación cuando no se encuentra un argumento primero viable utilizando los patrones de extracción, mecanismo descripto en la sección 6.2.1. La segunda funcionalidad heredada es el sistema de puntaje, el cual es esencialmente el mismo que el utilizado por ATP-OIE, con la salvedad principal de que se han corregidos las reglas necesarias para su aplicación en idioma español. Se han removido además tres reglas. El sistema final de reglas implementado puede verse en detalle en la Tabla 21.

Tabla 20. Conversión de categorías gramaticales

Penn Treebank POS tags a *Universal POS tags*			
#→SYM	EX→PRON	NNPS→PROPN	UH→INTJ
$→SYM	FW→X	NNS→NOUN	VB→VERB
"→PUNCT	HYPH→PUNCT	PDT→DET	VBD→VERB
, →PUNCT	IN→ADP	POS→PART	VBG→VERB
-LRB-→PUNCT	JJ→ADJ	PRP→PRON	VBN→VERB
-RRB-→PUNCT	JJR→ADJ	PRP$→DET	VBP→VERB
. →PUNCT	JJS→ADJ	RB→ADV	VBZ→VERB
: →PUNCT	LS→X	RBR→ADV	WDT→DET
AFX→ADJ	MD→VERB	RBS→ADV	WP→PRON
CC→CCONJ	NIL→X	RP→ADP	WP$→DET
CD→NUM	NN→NOUN	SYM→SYM	WRB→ADV
DT→DET	NNP→PROPN	TO→PART	. →PUNCT

Tabla 21. Reglas utilizadas por TP-OIE-ES para puntuar una relación semántica.

Regla	Puntaje
a1+**r**+**a2** incluidos en **s** y longitud(**s**) ~= longitud(**a1**+**r**+**a2**)	116
Última palabra de **r** es *POR o PARA*	50
Última palabra de **r** es *EN*	49
Última palabra de **r** es *DE*	46
Cantidad palabras en **s**< 11	43

r cumple con que tiene 3 o más palabras y la primera es un verbo, la palabra segunda, tercera, etc. es sustantivo, adjetivo, adverbio, pronombre o determinante y la última es una adposición.	42
Última palabra de **r** es *A o HACIA*	39
Cantidad palabras en **s**<21	23
s comienza con **a1**	21
a2 es un sustantivo propio	16
a1 es un sustantivo propio	1
A la izquierda de **a1** hay una frase nominal	-30
a1 termina con la misma palabra con que empieza **r**	-50
Si **r** es una frase de 3 palabras y estas tienen las siguientes categorías gramaticales: verbo, partícula gramatical y adverbio	-61
Si a la izquierda de **a1** hay una preposición	-65
Si hay una frase nominal a la derecha de **a2**	-81
Si la palabra a la izquierda de **r** es una conjunción	-93
Si **a1** es igual a **a2**	-100
r está incluida completamente en **a1**	-100
a2 es un determinante	-1000

Nota: **s** es la oración original dada como entrada, **a1** es el argumento primero, **r** es la relación y **a2** es el argumento segundo.

El proceso de extracción es el mismo que el descripto en el punto 6.1.3, con la diferencia de que en el paso 2 se realiza la conversión de las etiquetas gramaticales.

TP-OIE-ES a diferencia de ATP-OIE no cuenta con ningún método auxiliar que pueda invocar cuando no encuentra una relación semántica válida.

6.3.2 Relaciones semánticas poco informativas

El mecanismo utilizado por TP-OIE-ES para mejorar las oraciones poco informativas consistió en adaptar el método de ATP-OIE basado en las palabras conectores. Para idioma español se utilizaron las palabras: *a*, *de*, *en* y *los* como conectores, y siguiendo la misma lógica que en ATP-OIE cuando se detecta una palabra conector a la izquierda de uno de los argumentos candidatos seguida de una frase nominal, el conector y la frase nominal son añadidos al argumento candidato. Un ejemplo de cómo este mecanismo puede mejorar la información en una relación semántica se muestra a continuación en el Ejemplo 12.

El vocativo épico es un enunciado exclamativo intercalado en una oración. [12]

TP-OIE-ES extraerá la siguiente relación semántica:

- (El vocativo épico, es, un enunciado exclamativo intercalado **en** una oración)

Al igual que en el ejemplo de la sección 6.2.2, el método añadió información al argumento segundo. En este caso añadió: *en una oración*, de lo contrario la tupla hubiese terminado con la palabra: *intercalado*.

6.3.3 Manejo de información subjetiva

TP-OIE-ES utiliza los mismos dos mecanismos para detectar información subjetiva que los ya descriptos para ATP-OIE en la sección 6.2.3. El primer mecanismo, el que consiste en reemplazar todo texto entrecomillado por una palabra comodín, funciona de igual manera sin ninguna modificación adicional. El segundo mecanismo, el que recorta una porción del texto cuando detecta en la oración una palabra que indica opinión, cuenta con una lista diferente de palabras para idioma español. La lista de palabras utilizadas por TP-OIE-ES es la siguiente: dijo, pensó, contó, añadió, anunció, aseveró, cree, creen, creyó, pensaban, creían y sostenían. Esta lista es bastante corta y quedó como un punto de mejora a futuro el ampliarla. Esto se hará en el siguiente método publicado: ECMes.

Se repite a continuación en el Ejemplo 13 el ejemplo dado en la sección 6.2.3, pero para idioma español. Se muestran también las relaciones semánticas generadas por TP-OIE-ES utilizando estos mecanismos.

Los primeros astrónomos creían que la Tierra era el centro del universo. [13]

- 1 (Los primeros astrónomos, creían, que la tierra era el centro del universo) => (44)
- 5 (la tierra, era, el centro del universo) => (23) DEPENDS OF 1

Este ejemplo muestra como el método detectó que la segunda extracción es en realidad una relación semántica dependiente de otra y es por lo tanto subjetiva o no-fáctica. También puede verse como el mismo mecanismo funciona tanto para idioma inglés como para idioma español.

6.3.4 Resultados de TP-OIE-ES

Para medir la precisión, la exhaustividad y la medida F1 se utilizó una base de datos de 68 oraciones extraídas de textos escolares mexicanos, propuesta por [Zhila y Gelbukh, 2014], llamada *Parallel English-Spanish corpus*. En esta base de datos cada una de las oraciones fue traducida a idioma inglés de forma manual y su finalidad original era comprobar la efectividad del método ExtrHech en ambos idiomas. En el artículo de [Rodríguez y Merlino, 2020] se comparó TP-OIE-ES con otros dos

métodos de extracción de conocimiento para la Web en español. Con el más reciente (al momento de realizar las mediciones): ArgOE [Pablo y Marcos, 2015] y con el más preciso: DepOE [Gamallo y Garcia, 2012], según se mostró en la sección 3.4. Multi^{2}OIE no había sido aún publicado.

Se detallan a continuación las fórmulas utilizadas para calcular la precisión (9) y la exhaustividad (10).

$$precisión = \frac{relaciones\ semánticas\ extraídas\ marcadas\ como\ correctas}{relaciones\ semánticas\ extraídas\ no\ ignoradas} \quad \textbf{(9)}$$

$$exhaustividad = \frac{relacion\ es\ semánticas\ extraídas\ marcadas\ como\ correctas}{totalidad\ de\ hechos\ fácticos} \quad \textbf{(10)}$$

La medida F1, se calculó utilizando la Fórmula 8. En la Fórmula 10 para el valor indicado como "totalidad de hechos fácticos" se tomó el valor dado en [Zhila y Gelbukh, 2014] que es 137. Los resultados obtenidos se resumen en la Tabla 22.

Tabla 22. Medidas calculadas para TP-OIE-ES, DepOE y ArgOE.

Métodos	Precisión	Exhaustividad	Medida F1
TP-OIE-ES	0,62	**0,36**	**0,46**
DepOE	**0,89**	0.29	0,44
ArgOE	0,67	0,29	0,40

A partir de los resultados de la Tabla 22 puede concluirse que TP-OIE-ES es un método de extracción de conocimiento para la Web en el estado del arte para idioma español, al menos al momento de su publicación. Si bien no logró la mejor precisión obtuvo la mayor exhaustividad y la medida F1 más alta. Los puntos de mejora futuros que quedan pendientes para este método son los de aumentar la precisión y la exhaustividad y obtener métricas de rendimiento en un segundo conjunto de pruebas. Con un segundo conjunto de pruebas se puede mejorar la evidencia disponible para entender que tan bien se desempeña el método en comparación a otros. Hay que tener en cuenta que el conjunto de datos utilizado para la evaluación de los métodos está compuesto por oraciones extraídas de textos escolares, como se mencionó, y es por ello un conjunto bastante homogéneo. Por último, será necesario evaluar también al método Multi^{2}OIE en idioma español en el mismo conjunto de datos que a los demás.

6.4 ECMes (Extractor de Conocimiento Mejorado en Español)

ECMes es un método de extracción de conocimiento para la Web en idioma español construido sobre la arquitectura base de TP-OIE-ES. ECMes implementa una serie de mejoras sobre el algoritmo original que le permiten incrementar considerablemente su precisión, elevar su exhaustividad y por lo tanto, mejorar la medida F1. ECMes fue presentado en 2021 en un *journal* de procesamiento de lenguaje natural en español [Rodríguez et al., 2021].

6.4.1 Puntos de mejora sobre TP-OIE-ES

ECMes nace con el propósito de mejorar a TP-OIE-ES, poniendo especial atención en los puntos más débiles de su predecesor. Algunos de los puntos de mejora fueron comentados en la sección 6.3.4 pero se expondrán aquí en mayor detalle. Aunque TP-OIE-ES logra mejorar la exhaustividad y con ello logra una medida F1 más alta que los otros métodos evaluados en la sección 6.3.4, quedan al menos tres problemas importantes por ser resueltos.

6.4.1.1 Mejorar la precisión

Para intentar mejorar las medidas de rendimiento se implementaron tres mejoras en el algoritmo original de TP-OIE-ES, las mismas se describen en las secciones siguientes.

Regeneración de los patrones de búsqueda

La principal medida que se tomó para mejorar la precisión y exhaustividad del método original fue la de reentrenar al mismo con un nuevo conjunto de ejemplos, esta vez en idioma español. Esto implicó convertir los métodos nativos que trabajaban con las categorías gramaticales en idioma inglés en el formato de *Penn Treebank POS tags* al formato *Universal POS tags*. No sólo para el sistema de búsqueda de coincidencias de patrones, sino también para el sistema de puntajes.

La nueva base de datos de entrenamiento se construyó con un total de 122 oraciones en idioma español. De dichas oraciones se extrajo un total de 229 relaciones semánticas de forma manual. El resumen se muestra en la Tabla 23[8].

[8] El detalle de oraciones y relaciones semánticas puede hallarse *online* en https://github.com/juanma1982/ECMes/tree/master/datasets

Los *tweets* y las frases de periódicos de noticias fueron obtenidas al azar de tres conjuntos de datos separados disponibles públicamente en el sitio web Kaggle[9]. Las frases de libros fueron extraídas manualmente de diversos libros, para ello se solicitó a un estudiante de Doctorado en Ciencias Informáticas, sin relación con este proyecto de investigación, que escogiera frases de libros en español, sin importar si eran o no traducciones.

A este conjunto de datos de entrenamiento se sumó uno adicional con 12 oraciones creadas especialmente para propósitos de prueba durante la fase de desarrollo del método. La mayoría de estas frases son traducciones de las frases propuestas en [Del Corro y Gemulla, 2013] como ejemplos de los diferentes tipos de oraciones para el idioma inglés.

Tabla 23. Oraciones y extracciones manuales por origen, en el nuevo conjunto.

Fuente	Oraciones	Relaciones semánticas
es.wikipedia.org	33	62
tweets Covid-19	13	13
tweets municipalidad	9	15
libros	23	54
periódicos de noticias	32	65
oraciones de prueba	12	20

Mejora en el sistema de puntaje

La otra tarea que se adicionó para mejorar la precisión fue mejorar el sistema de puntaje. En este caso, el cambio consistió principalmente en la utilización de las categorías gramaticales nativas del *parser* para idioma español, es decir las etiquetas *Universal POS* en lugar de hacer una traducción de éstas a las etiquetas *Penn Treebank POS* utilizadas en idioma inglés y heredadas de la versión del sistema de puntajes de ATP-OIE. Además se añadieron 2 reglas a las ya utilizadas en TP-OIE-ES. El sistema de reglas final utilizado por ECMes queda reflejado en la Tabla 24, las nuevas reglas están marcadas en negrita.

[9] https://www.kaggle.com

Tabla 24. Reglas utilizadas por ECMes para puntuar una relación semántica.

Regla	Puntaje
a1+**r**+**a2** incluidos en **s** y longitud(**s**) ~= longitud(**a1**+**r**+**a2**)	116
Última palabra de **r** es *POR o PARA*	50
Última palabra de **r** es *EN*	49
Última palabra de **r** es *DE*	46
Cantidad palabras en **s**< 11	43
r cumple con que tiene 3 o más palabras y la primera es un verbo, la palabra segunda, tercera, etc. es sustantivo, adjetivo, adverbio, pronombre o determinante y la última es una adposición.	42
Última palabra de **r** es *A o HACIA*	39
Cantidad palabras en **s**<21	23
s comienza con **a1**	21
a2 es un sustantivo propio	16
r es una sola palabra y es un verbo	**10**
a1 es un sustantivo propio	1
A la izquierda de **a1** hay una frase nominal	-30
a1 termina con la misma palabra con que empieza **r**	-50
Si **r** es una frase de 3 palabras y estas tienen las siguientes categorías gramaticales: verbo, partícula gramatical y adverbio	-61
Si a la izquierda de **a1** hay una preposición	-65
Si hay una frase nominal a la derecha de **a2**	-81
Si la palabra a la izquierda de **r** es una conjunción	-93
Si **a1** es igual a **a2**	-100
r está incluida completamente en **a1**	-100
r es una sola palabra y es un determinante	**-200**
a2 es un determinante	-1000

Nota: **s** es la oración original dada como entrada, **a1** es el argumento primero, **r** es la relación y **a2** es el argumento segundo.

Detección de sujetos tácitos

Un problema adicional del idioma español es el sujeto tácito, si bien éste existe en idioma inglés, es poco frecuente y en general, sólo se omite el sujeto si ya fue nombrado antes en la misma oración. Esta particularidad del idioma español provocaba que el algoritmo de TP-OIE-ES no detectase relaciones semánticas en muchas oraciones, ya que no encontraba dentro de la misma un sujeto candidato para el primer argumento. Supóngase la oración del Ejemplo 14.

Jugábamos al fútbol. [14]

La relación semántica en este caso, indica que *nosotros* es el argumento primero o el sujeto, que la relación es *jugar al* y que el argumento segundo es *fútbol*. La tupla debería quedar de la siguiente forma:

- (Nosotros, jugábamos al, fútbol)

Sin embargo, el algoritmo original no encontrará nunca una palabra en la oración de entrada que pueda asociar al argumento primero, simplemente porque esa palabra no está presente. Para resolver este problema, cuando el algoritmo no es capaz de hallar un argumento primero, añadirá al comienzo de la oración una palabra comodín, la cual será analizada como un pronombre personal. Si con esta nueva palabra encuentra una relación semántica, tal que el comodín coincide con el argumento primero, el algoritmo devolverá una tupla con el argumento primero vacío. Siguiendo con el ejemplo anterior:

- (, jugábamos, al fútbol)

El espacio inicial vacío indica que hay un sujeto tácito en la relación semántica devuelta. Esta mejora busca no sólo aumentar la precisión sino también la exhaustividad del método.

6.4.1.2 Mejorar la evidencia disponible

Se construyó un conjunto de datos de prueba usando las mismas fuentes utilizadas para la construcción del conjunto de datos de entrenamiento. Este conjunto de datos consta de 55 oraciones diferentes y un total de 120 relaciones semánticas extraídas de forma manual. El resumen se muestra en la Tabla 25[10].

Tabla 25. Oraciones y extracciones manuales por origen en el nuevo conjunto de datos de pruebas.

Fuente	Oraciones	Relaciones semánticas
es.wikipedia.org	16	36
tweets Covid-19	6	10
tweets municipalidad	5	9
libros	12	29
periódicos de noticias	16	36

[10] El detalle de las oraciones y sus respectivas relaciones semánticas puede hallarse *online* en: https://github.com/juanma1982/ECMes/tree/master/datasets

6.4.1.3 Mejorar las relaciones poco informativas

Para mejorar la calidad de las extracciones realizadas, se implementaron tres mejoras al algoritmo original de TP-OIE-ES, las cuales se detallan en las secciones siguientes.

Expandir la relación

El algoritmo original está pensado para construir la relación propiamente dicha según patrones de coincidencia en el árbol de dependencias sintácticas. Esto implica que puede tomar palabras no consecutivas dentro de la oración para formar la relación. Utilizando como entrada la oración del Ejemplo 15, TP-OIE-ES podría construir un relación como: *fue galardonado en*, aunque las palabras *galardonado* y *en* no son consecutivas. Si bien en este ejemplo esto es correcto, hay muchos casos en los cuales se generan relaciones poco informativas o bien el algoritmo no logra encontrar un argumento segundo para la relación armada. Para estos casos se decidió ampliar la relación y que ésta contenga todas las palabras existentes entre su palabra inicial y final. Continuando con el Ejemplo 15, la nueva relación quedaría: *fue galardonado con el Premio Nobel en*.

Albert Einstein fue galardonado con el Premio Nobel en Suecia. [15]

Agregar nombres de entidades

Existen casos en donde una oración contiene un nombre de entidad (NER) y sin embargo, ésta no aparece en la extracción realizada. Por ejemplo, TP-OIE-ES para la oración del Ejemplo 16, extrajo las tuplas que se muestran a continuación:

La civilización china nos heredó el papel, la pólvora, una forma de imprenta rudimentaria, y la brújula. [16]

- (La civilización, nos heredó, el papel)

Esto se debe a que el *parser* superficial que busca un posible argumento primero, no detecta a la entidad (en este caso *china*) como parte de la frase nominal. Se agregó en este caso una corrección al algoritmo para que no ignore entidades detectadas que están junto a frases nominales candidatas.

Tener en cuenta múltiples verbos

El método original fallaba al extraer la relación en oraciones donde aparecen dos o más verbos seguidos, ya que, por lo general, sólo extrae uno solo de los verbos, respetando las coincidencias del patrón de extracción. Para ilustrar este punto, supóngase que una oración como la del Ejemplo 17, cuyo árbol de dependencias sintácticas se muestra en la Figura 7, fue utilizada para entrenar al algoritmo.

La ciencia mejoró la sociedad. [17]

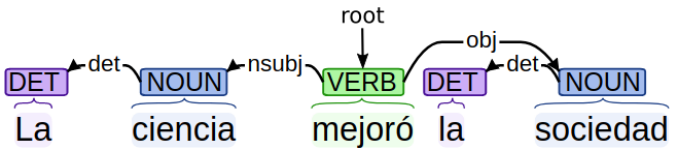


Figura 7: Árbol de dependencias sintácticas y categorías gramaticales. Oración del ejemplo 17.

La relación que en este caso es: *mejoró*, es el verbo raíz en el árbol de dependencias sintácticas. Este ejemplo generará un patrón que servirá para identificar a cualquier verbo raíz como una posible relación candidata para la tupla. Con lo cual, una oración como la del Ejemplo 18, cuyo árbol de dependencias sintácticas se muestra en la Figura 8, detectará como relación candidata la palabra: permitido e ignorará el verbo mejorar.

La ciencia ha permitido mejorar la sociedad. [18]

En el ejemplo de la Figura 18, el patrón de coincidencia tampoco está considerando al verbo auxiliar: *ha*, aunque éste podría ser detectado por un patrón diferente. Todos los patrones son ejecutados para detectar relaciones candidatas pero luego se retienen sólo aquellas que más valor obtienen en el sistema de puntajes.

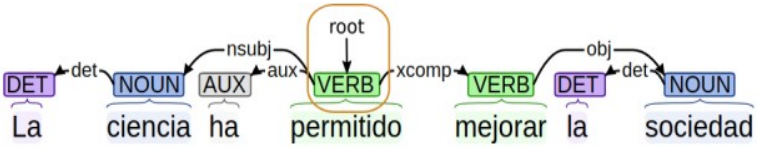


Figura 8: Árbol de dependencias sintácticas y categorías gramaticales, oración del ejemplo 18. En naranja se muestra la coincidencia del patrón utilizado.

La mejora que se introdujo al método original de TP-OIE-ES implica mantener unidos los verbos que están juntos en la oración, incluidos los auxiliares. Esto ha permitido mejorar la cantidad de información en las relaciones extraídas.

6.4.2 Evaluación y resultados de ECMes

Para medir la precisión, la exhaustividad y la medida F1 se utilizaron dos conjuntos de pruebas. El primero es el ya utilizado para medir el desempeño de TP-OIE-ES, mencionado en la sección 6.3.4, conformado por 68 oraciones en español y propuesto originalmente en [Zhila y Gelbukh, 2014]. El segundo conjunto de datos utilizado es el nuevo conjunto descripto en la Tabla 25 de la sección 6.4.1.2, creado para mejorar la evidencia disponible. A este conjunto se lo mencionará de ahora en adelante como *conjunto Rodríguez* ya que fue presentado en la publicación de [Rodríguez et al., 2021]. En el primero de estos dos conjuntos, se comparó a ECMes con otros cuatro métodos de Open IE en español: ArgOE [Gamallo y Garcia, 2015], DepOE [Gamallo y Garcia, 2012], Multi^{2}OIE [Ro et al., 2020] y TP-OIE-ES. En el segundo conjunto se lo comparó sólo contra ArgOE, DepOE y Multi^{2}OIE. Se omitió la comparación contra TP-OIE-ES ya que ECMes es esencialmente una versión mejorada de este último.

Los métodos ArgOE y DepOE están disponibles públicamente, incluyendo versiones ejecutables de los mismos, con lo cual la utilización de estos métodos para realizar pruebas no significó ningún obstáculo. Sin embargo, de Multi^{2}OIE sólo se encuentra disponible su código fuente, no se cuenta con una versión ejecutable. Cómo este método utiliza redes neuronales profundas, lo más importante para su funcionamiento es contar con el modelo entrenado. Con el fin de utilizar exactamente la misma versión del modelo que la utilizada en el artículo de [Ro et al., 2020] se intentó, sin éxito, contactar con los autores.

La versión de Multi^{2}OIE utilizada en las pruebas descriptas en esta sección corresponde a una versión entrenada a partir del código fuente disponible, utilizando

los mismos hiperparámetros sugeridos por los autores y con el mismo conjunto de datos de entrenamiento[11]. Sin embargo, los resultados esperados para los valores de la medida F1 y de la AUC, tanto en el conjunto de entrenamiento como en el de prueba, difieren ligeramente de los indicados por los autores. Estas diferencias se reflejan en la Tabla 26, en donde se muestran los resultados obtenidos en el conjunto de datos CaRB. CaRB [Bhardwaj et al., 2019] es un conjunto de datos de prueba en inglés que Multi^{2}OIE trae incorporado junto a su código fuente y que utiliza internamente para realizar validaciones.

Tabla 26. Resultado del entrenamiento de Multi^{2}OIE, valores esperados versus obtenidos.

Conjunto de datos	Conjunto CaRB entrenamiento		Conjunto CaRB pruebas	
	F1 esperada	AUC esperada	F1 obtenida	AUC obtenida
Valor esperado	0,543	0,348	0,523	0,326
Valor obtenido	0,535	0,327	0,512	0.299
Diferencia	0,008	0,021	0,011	0,027

Las diferencias observadas no son demasiado grandes, la mayor de ellas es menor a un 3% como se observa en la Tabla 26, se espera por lo tanto que el desempeño de esta versión reentrenada de Multi^{2}OIE sea similar al de la versión descripta en el artículo de [Ro et al., 2020]. En dicho artículo los autores advierten que un reentrenamiento de la red podría arrojar resultados ligeramente diferentes durante la evaluación de desempeño. En la Tabla 27 se muestran los resultados de la comparación de los distintos métodos en idioma español sobre el conjunto *Parallel English-Spanish corpus* [Zhila y Gelbukh, 2014]. En la Tabla 28 se muestran los resultados de la comparación entre los mismos método, menos TP-OIE-ES, en el conjunto de datos Rodríguez.

Tabla 27. Medidas calculadas para TP-OIE-ES, DepOE, ArgOE, ECMes y Multi^{2}OIE en el conjunto de Zhila y Gelbunk.

Métodos	Precisión	Exhaustividad	Medida F1
TP-OIE-ES	0,62	0,36	0,46
ECMes	**0,92**	**0,42**	**0,57**
Multi^{2}OIE	0,46	0,19	0,27
DepOE	0,89	0.29	0,44
ArgOE	0,67	0,29	0,40

[11] https://github.com/youngbin-ro/Multi2OIE

Tabla 28. Medidas calculadas para DepOE, ArgOE, ECMes y Multi^{2}OIE en el conjunto Rodríguez.

Métodos	Precisión	Exhaustividad	Medida F1
ECMes	0,68	**0,34**	**0,45**
Multi^{2}OIE	0,22	0,10	0,14
DepOE	**0,81**	0,18	0,30
ArgOE	0,68	0,22	0,33

De los resultados anteriores lo primero que podría sorprender es el bajo desempeño del método Multi^{2}OIE en las extracciones para idioma español ya que en el artículo de [Ro et al., 2020] los autores indican que el método obtiene una medida F1 de 0,602 en la base de datos Re-OIE2016. Sin embargo, los autores también indican que el conjunto de oraciones en español se construyó haciendo traducciones automáticas de oraciones en idioma inglés. Las oraciones originalmente formaban parte del conjunto Re-OIE2016. En cambio, los dos conjuntos de datos utilizados en las pruebas descriptas en esta sección fueron construidos con oraciones originales en español. Este podría ser el principal motivo de las diferencias observadas.

Por otro lado, se observa que Multi^{2}OIE se desempeña mejor en el conjunto de datos *Parallel English-Spanish corpus* [Zhila y Gelbukh, 2014] que en el conjunto de datos Rodríguez. Hay que tener en cuenta que en el primero de estos conjuntos se tienen oraciones relativamente sencillas, gramaticalmente correctas y en cierto sentido más fáciles de analizar, ya que son frases extraídas de textos escolares. Ésta es posiblemente la causa por la cual Multi^{2}OIE se desempeña mejor allí que en el conjunto Rodríguez en donde las oraciones en español son heterogéneas, tomadas de diversas fuentes. En el caso particular de las frases tomadas de Twitter, como su estructura sintáctica no es necesariamente correcta, todos los métodos tuvieron un desempeño peor en este conjunto de datos.

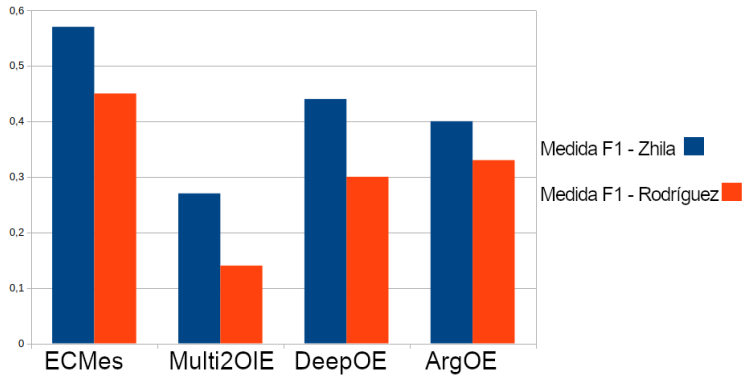


Figura 9: Medida F1 de los distintos métodos en ambos conjuntos de prueba

El resultado resumido de ambas tablas puede verse en la Figura 9. Es posible concluir a partir de los resultados arrojados, que en los conjuntos de prueba utilizados en este trabajo de investigación, ECMes supera en efectividad a otros métodos de extracción de conocimiento para la Web en español en el estado del arte.

7. Aportes y Conclusiones

Se detalla en esta sección los aportes realizados por este trabajo de investigación en relación con los objetivos planteados en la sección 1.

7.1 Objetivo principal: creación de métodos de *Open IE* para idioma español

Se crearon no sólo uno sino dos métodos de extracción de conocimiento para la Web en idioma español: TP-OIE-ES y ECMes.

TP-OIE-ES se posicionó, en las pruebas realizadas, como el método con el mejor desempeño para el idioma español a pesar de haber sido entrenado en idioma inglés. Mostrando que la mecánica propuesta por TP-OIE y sus métodos derivados: ATP-OIE, TP-OIE-ES, ECMEs, basada en ejemplos, es suficientemente versátil para que un mismo método funcione en inglés y en español con un ajuste mínimo. Este ajuste no es otra cosa que un mapa o diccionario para convertir las categorías gramaticales de un sistema a otro (cómo se mencionó, de *Penn Treebank* a *Universal POS*).

ECMes superó a TP-OIE-ES y a los otros métodos evaluados en idioma español, demostrando que si bien la mecánica propuesta en TP-OIE funciona, se obtienen mejores resultados cuando el método es entrenado en idioma nativo. Se puede concluir que ECMes es, hoy por hoy, un método de extracción de conocimiento para Web en el estado del arte para idioma español y es el que mejor desempeño tiene en los conjuntos de pruebas utilizados.

7.2 Objetivo secundario: creación de un marco de referencia para la evaluación de los métodos de *Open IE*

Se puede afirmar que se cumplió el objetivo secundario propuesto con la creación de dos conjuntos de datos para la evaluación de métodos de extracción de conocimiento para la Web, uno en inglés: Reuters-103 y otro en español: conjunto Rodríguez. Ambos conjuntos contienen textos en lenguaje natural y una serie de relaciones semánticas, creadas a mano, asociadas a cada ejemplo.

Por otro lado, debido a que los criterios utilizados para clasificar a una relación semántica como valida o invalida pueden diferir de persona a persona, se detalló en la sección 4.2 como se calculan las métricas utilizadas en este trabajo.

7.3 Objetivo secundario: creación de un método de *Open IE* novedoso

Para cumplir con este objetivo se creó primeramente el método TP-OIE, el cual sirvió para poner a prueba la mecánica propuesta. Este primer método fue mejorado cuando se creó ATP-OIE, un método derivado, pensado también para trabajar en idioma inglés. ATP-OIE logró posicionarse por encima de los métodos de *Open Information Extraction* de la primera generación: ClausIE, OLLIE y ReVerb, aunque es superado en la medida F1 por MinIE. Sin embargo, ATP-OIE logró la precisión más alta de los cinco métodos evaluados en el conjunto Reuters-103. Otra de las mecánicas propuestas por ATP-OIE, la de aprender en línea, es decir, generar nuevos patrones de extracción mientras está realizando extracciones de forma productiva, arrojó un resultado moderado. Mejoró un poco el desempeño de ATP-OIE pero no lo suficiente para superar a MinIE.

A partir de los prometedores resultados de ATP-OIE, se diseñaron los métodos: TP-OIE-ES y ECMes para idioma español.

7.4 Aporte adicional: Mejoras en métodos existentes

Un aporte adicional de esta tesis, fue la creación de una versión mejorada del método ClausIE como se describió en la sección: 4.4.

8. Futuras líneas de investigación

Los problemas planteados en la sección 5, fueron mitigados con la familia de métodos basados en TP-OIE y sus derivados pero no fueron completamente resueltos. Estos problemas siguen pendientes al día de hoy. Incluso siguen siendo problemas pendientes para los métodos más avanzados de extracción de conocimiento para la Web como, por ejemplo, OpenIE6.

Todo método de extracción de conocimiento para la Web debe ser comparado, en última instancia, con la capacidad humana de comprender y analizar un texto. Estos métodos permiten, de forma automática y mediante algún procesamiento posterior, analizar, resumir o incluso crear mapas de conocimiento de textos en lenguaje natural. Son utilizados cuando el volumen de datos a procesar, el corpus, es demasiado grande para poder ser analizado por personas. Los errores descriptos en la sección 5: falta de precisión, falta de exhaustividad, relaciones semánticas poco informativas y manejo de información subjetiva, son todos errores debidos a la misma causa raíz: falta de entendimiento o de compresión de un texto. Una persona puede, hipotéticamente, encontrar todas las relaciones semánticas existentes, sin perder ninguna, sin perder información relevante y entendiendo cuando se trata de hechos concretos y cuando de opiniones, porque es capaz de entender el contenido de un texto dado. Y si una, o varias personas, en última instancia, no pueden comprender un texto, posiblemente sea porque el texto está mal redactado o es deliberadamente ambiguo. Sin embargo, las personas carecen de velocidad para realizar esta tarea. Todos los métodos relevados en la sección 3 tienen los mismos problemas, en particular no logran superar cierto límite en cuanto a relaciones semánticas correctas extraídas. Y estos resultados no mejoran aún agregando ejemplos a los respectivos conjuntos de entrenamiento. Lo mismo sucede con los métodos presentados en esta tesis. Muchas veces mejorar en un dominio específico no implica mejorar para cualquier texto, incluso podría suceder que mejorar para cierto conjunto de datos implique empeorar en otro.

En paralelo a la proliferación de métodos de extracción de conocimiento para la Web, han aparecido algunos modelos puntuales para intentar crear una inteligencia artificial capaz de “comprender” texto en lenguaje natural o bien acercarse lo más posible a este fin. En particular, se trata de modelos de redes neuronales artificiales, preentrenados. Una de ellas es GPT-3 [Floridi y Chiriatti, 2020] y la otra es BERT [Devlin et al., 2019]. A pesar lo promisorio de estos modelos, no logran procesar el

lenguaje natural al mismo nivel que un ser humano. Incluso uno de los métodos aquí analizados: Multi^{2}OIE, utiliza internamente BERT y aun así no es el método con mejor desempeño. De lograrse el objetivo de construir una IA capaz de comprender un texto en lenguaje natural de la misma forma en la que lo hace una persona, los métodos de extracción de conocimiento para la Web dejarían de tener sentido, ya que su funcionalidad sería sólo una de las muchas tareas factibles para estos sistemas. Por lo cual, los problemas planteados seguirán abiertos hasta que pueda resolverse de forma satisfactoria la compresión de textos de forma automática. Sin embargo, siempre hay lugar para mejorar, aunque sea mínimamente, los resultados actuales.

Referencias

Abney, S.P., 1991. Parsing by chunks, en: Principle-based parsing. Springer, pp. 257-278.

Amigó, E., Gonzalo, J., Artiles, J., Verdejo, F., 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. Inf. Retr. 12, 461-486.

Angeli, G., Johnson Premkumar, M.J., Manning, C.D., 2015. Leveraging Linguistic Structure For Open Domain Information Extraction, en: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Association for Computational Linguistics, Beijing, China, pp. 344-354. https://doi.org/10.3115/v1/P15-1034

Banko, M., Cafarella, J.M., Soderland, S., Broadhead, M., Etzioni, O., 2007. Open information extraction for the web. IJCAI 7, 2670-2676.

Banko, M., Cafarella, M., Soderland, S., Broadhead, M., Etzioni, O., 2008. Open information extraction from the web. Commun. ACM 51, 68--74.

Bast, H., Haussmann, E., 2013. Open Information Extraction via Contextual Sentence Decomposition, en: 2013 IEEE Seventh International Conference on Semantic Computing. Presentado en 2013 IEEE Seventh International Conference on Semantic Computing (ICSC), IEEE, Irvine, CA, USA, pp. 154-159. https://doi.org/10.1109/ICSC.2013.36

Bhardwaj, S., Aggarwal, S., Mausam, M., 2019. CaRB: A Crowdsourced Benchmark for Open IE, en: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Association for Computational Linguistics, Hong Kong, China, pp. 6261-6266. https://doi.org/10.18653/v1/D19-1651

Bhutani, N., Suhara, Y., Tan, W.-C., Halevy, A., Jagadish, H.V., 2019. Open information extraction from question-answer pairs. ArXiv Prepr. ArXiv190300172.

Boschee, E., Freedman, M., Khanwalkar, S., Kumar, A., Srivastava, A., Weischedel, R., 2014. Researching persons & organizations: AWAKE: From text to an entity-centric knowledge base, en: 2014 IEEE International Conference on Big Data (Big Data). IEEE, Washington, DC, USA, pp. 1030-1039. https://doi.org/10.1109/BigData.2014.7004337

Bradley, A.P., 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognit. 30, 1145--1159.

Buchholz, S., Marsi, E., 2006. CoNLL-X shared task on multilingual dependency parsing, en: In Proceedings of the tenth conference on computational natural language learning. pp. 149-164.

Cabral, B., Glauber, R., Souza, M., Claro, D., 2020. CrossOIE: Cross-Lingual Classifier for Open Information Extraction. pp. 368-378. https://doi.org/10.1007/978-3-030-41505-1_35

Cetto, M., Niklaus, C., Freitas, A., Handschuh, S., 2018. Graphene: Semantically-Linked Propositions in Open Information Extraction. ArXiv180711276 Cs.

Christensen, J., Mausam, Soderland, S., Etzioni, O., 2011. An analysis of open information extraction based on semantic role labeling, en: the sixth international conference on Knowledge capture ,ACM. pp. 113-120.

Consoli, S., Nuzzolese, A.G., Presutti, V., Reforgiato Recupero, D., Gangemi, A., 2015. Legalo: Revealing the Semantics of Links, en: Lambrix, P., Hyvönen, E., Blomqvist, E., Presutti, V., Qi, G., Sattler, U., Ding, Y., Ghidini, C. [Eds.], Knowledge Engineering and Knowledge Management, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 140-144. https://doi.org/10.1007/978-3-319-17966-7_18

Cowie, J., Lehnert, W., 1996. Information Extraction. Commun. ACM 39, 80--91.

Cui, L., Wei, F., Zhou, M., 2018. Neural open information extraction. ArXiv Prepr. ArXiv180504270.

Culotta, A., McCallum, A., Betz, J., 2006. Integrating probabilistic extraction models and data mining to discover relations and patterns in text, en: main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics. pp. 296-303.

Danqi, C., Christopher, M., 2014. A fast and accurate dependency parser using neural networks, en: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 740-750.

de Abreu, S.C., Vieira, R., 2017. Relp: Portuguese open relation extraction. Knowl. Organ. 44, 163-177.

Del Corro, L., Gemulla, R., 2103. ClausIE: clause-based open information extraction, en: 22nd international conference on World Wide Web. pp. 355-366.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, en: Proceedings of the 2019 Conference of the North. Association for Computational Linguistics, Minneapolis, Minnesota, pp. 4171-4186. https://doi.org/10.18653/v1/N19-1423

Etzioni, O., Cafarella, M., Downey, D., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A., 2005. Unsupervised named-entity extraction from the Web: An experimental study. Artif. Intell. 165, 91--134.

Fader, A., Soderland, S., Etzioni, O., 2011. Identifying relations for open information extraction, en: Association for Computational Linguistics. pp. 1535-1545.

Floridi, L., Chiriatti, M., 2020. GPT-3: Its Nature, Scope, Limits, and Consequences. Minds Mach. 30, 681-694. https://doi.org/10.1007/s11023-020-09548-1

Gamallo, P., Garcia, M., 2015. Multilingual Open Information Extraction, en: Pereira, F., Machado, P., Costa, E., Cardoso, A. [Eds.], Progress in Artificial Intelligence, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 711-722. https://doi.org/10.1007/978-3-319-23485-4_72

Gamallo, P., Garcia, M., 2012. Dependency-based open information extraction, en: Proceedings of the joint workshop on unsupervised and semi-supervised learning in NLP. pp. 10-18.

Garcia, M., Gamallo, P., 2014. Entity-Centric Coreference Resolution of Person Entities for Open Information Extraction. Proces. Leng. Nat. 53, 25-32.

García-Martínez, R., Britos, P.V., 2004. Ingeniería de sistemas expertos. Nueva Librería.

Gashteovski, K., Gemulla, R., Del Corro, L., 2017. MinIE: Minimizing Facts in Open Information Extraction, en: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 2630-2640.

Gholap, J., 2012. Performance tuning of J48 Algorithm for prediction of soil fertility. ArXiv Prepr. ArXiv12083943.

Glauber, R., Barreiro Claro, D., 2018. A systematic mapping study on open information extraction, en: Expert Systems with Applications. Elsevier, pp. 372-387.

Gómez, A., Juristo, N., Montes, C., Pazos, J., 1997. Ingeniería del conocimiento. Editorial Centro de Estudios Ramón Areces.

Joachims, T., 1998. Text categorization with Support Vector Machines: Learning with many relevant features, en: Nédellec, C., Rouveirol, C. [Eds.], Machine Learning: ECML-98, Lecture Notes in Computer Science. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 137-142. https://doi.org/10.1007/BFb0026683

Jurafsky, D., Martin, J., 2000. Speech and language processing an introduction to natural language processing, computational linguistics, and speech recognition. Prentice-Hall, Inc.

Keele Staffs, 2007. Guidelines for performing Systematic Literature Reviews in Software Engineering (technical No. EBSE-2007-01]. Keele University, UK.

Kim, M.H., Compton, P., Kim, Y.S., 2011. RDR-based open IE for the web document, en: Proceedings of the Sixth International Conference on Knowledge Capture - K-CAP ’11. ACM Press, Banff, Alberta, Canada, p. 105. https://doi.org/10.1145/1999676.1999696

Kolluru, K., Adlakha, V., Aggarwal, S., Chakrabarti, S., others, 2020a. OpenIE6: Iterative Grid Labeling and Coordination Analysis for Open Information Extraction. ArXiv Prepr. ArXiv201003147.

Kolluru, K., Aggarwal, S., Rathore, V., Chakrabarti, S., others, 2020b. IMoJIE: Iterative Memory-Based Joint Open Information Extraction. ArXiv Prepr. ArXiv200508178.

Konstantinova, N., 2014. Review of Relation Extraction Methods: What Is New Out There?, Analysis of Images, Social Networks and Texts.

Lauscher, A., Song, Y., Gashteovski, K., 2019. MinScIE: Citation-Centered Open Information Extraction, en: 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL). IEEE, Champaign, IL, USA, pp. 386-387. https://doi.org/10.1109/JCDL.2019.00083

Lewis, D., 1997. Reuters-21578 text categorization test collection.

Li, Q., Wang, X., Zhang, Y., Ling, F., Wu, C.H., Han, J., 2018. Pattern Discovery for Wide-Window Open Information Extraction in Biomedical Literature. Presentado en 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE, Madrid, Spain, pp. 420-427. https://doi.org/10.1109/BIBM.2018.8621375

Lin, H., Wang, Y., Zhang, P., Wang, W., Yue, Y., Lin, Z., 2016. A Rule Based Open Information Extraction Method Using Cascaded Finite-State Transducer, en: Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R. [Eds.], Advances in Knowledge Discovery and Data Mining, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 325-337. https://doi.org/10.1007/978-3-319-31750-2_26

Lubov, A., Hamburg, M., Hamburg, M., 1979. Study guide to accompany Basic statistics: modern approach. Harcourt, Brace, Jovanovich, New York.

Mausam, Schmitz, M., Bart, R., Soderland, S., Etzioni, O., 2012. Open language learning for information extraction, en: 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. pp. 523-534.

McCallum, A., Nigam, K., others, 1998. A comparison of event models for naive bayes text classification, en: AAAI-98 workshop on learning for text categorization. Citeseer, pp. 41-48.

Merhav, Y., Mesquita, F., Barbosa, D., Yee, W.G., Frieder, O., 2012. Extracting information networks from the blogosphere. ACM Trans. Web 6, 1-33. https://doi.org/10.1145/2344416.2344418

Mesquita, F., Schmidek, J., Barbosa, D., 2013. Effectiveness and efficiency of open relation extraction, en: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. pp. 447-457.

Mesquita, F., Yuval, M., Barbosa, D., 2010. Extracting information networks from the blogosphere: state-of-the-art and challenges, en: Fourth AAAI Conference on Weblogs and Social Media (ICWSM), Data Challenge Workshop.

Mirrezaei, S.I., Martins, B., Cruz, I.F., 2015. The Triplex Approach for Recognizing Semantic Relations from Noun Phrases, Appositions, and Adjectives, en: Knowledge Discovery and Data Mining Meets Linked Open Data (Know@LOD) co-located with Extended Semantic Web Conference (ESWC).

Mohamed, T., Hruschka, E., Mitchell, T., 2011. Discovering relations between noun categories, en: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. pp. 1447-1455.

Pablo, G., Marcos, G., 2015. Multilingual open information extraction, en: Portuguese Conference on Artificial Intelligence. pp. 711-722.

Perera, R., Nand, P., 2015. A Multi-strategy Approach for Lexicalizing Linked Open Data, en: Gelbukh, A. [Ed.], Computational Linguistics and Intelligent Text Processing, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 348-363. https://doi.org/10.1007/978-3-319-18117-2_26

Petroni, F., Del Corro, L., Gemulla, R., 2015. CORE: Context-Aware Open Relation Extraction with Factorization Machines, en: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Lisbon, Portugal, pp. 1763-1773. https://doi.org/10.18653/v1/D15-1204

Petrov, S., Das, D., McDonald, R., 2011. A Universal Part-of-Speech Tagset.

Platt, J., 1998. Sequential minimal optimization: A fast algorithm for training support vector machines.

Prasojo, R.E., Kacimi, M., Nutt, W., 2018. StuffIE: Semantic Tagging of Unlabeled Facets Using Fine-Grained Information Extraction, en: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. ACM, Torino Italy, pp. 467-476. https://doi.org/10.1145/3269206.3271812

Qiu, L., Zhang, Y., 2014. ZORE: A Syntax-based System for Chinese Open Relation Extraction, en: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Doha, Qatar, pp. 1870-1880. https://doi.org/10.3115/v1/D14-1201

Quinlan, J.R., 2014. C4. 5: programs for machine learning. Elsevier.

Ramshaw, L.A., Marcus, M.P., 1999. Text chunking using transformation-based learning, en: Natural language processing using very large corpora. Springer, pp. 157-176.

Ratnaparkhi, A., 1996. A maximum entropy model for part-of-speech tagging, en: In Conference on Empirical Methods in Natural Language Processing.

Ro, Y., Lee, Y., Kang, P., 2020. Multi2OIE: Multilingual Open Information Extraction based on Multi-Head Attention with BERT. ArXiv Prepr. ArXiv200908128.

Rodríguez, J.M., Merlino, H., 2020. TP-OIE-ES: Método autónomo de extracción de relaciones semánticas para la Web en Español, en: Conferencia Iberoamericana de Complejidad, Informática y Cibernética: CICIC 2020.

Rodríguez, J.M., Merlino, H., García-Martínez, R., 2015. Revisión Sistemática Comparativa de Evolución de Métodos de Extracción de Conocimiento para la Web, en: XXI Congreso Argentino de Ciencias de la Computación (CACIC 2015).

Rodríguez, J.M., Merlino, H., Patricia, P., 2020. ATP-OIE: An Autonomous Open Information Extraction Method, en: International Conference On Compute And Data Analysis (iccda 2020).

Rodríguez, J.M., Merlino, H., Pesado, P., 2021. Mejoras aplicadas a la extracción de relaciones semánticas para la Web en español. Soc. Esp. Para El Proces. Leng. Nat. 66.

Rodríguez, J.M., Merlino, H., Pesado, P., García-Martínez, R., 2016a. Performance evaluation of knowledge extraction methods, en: International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems. pp. 16-22.

Rodríguez, J.M., Merlino, H.D., Pesado, P., García-Martínez, R., 2018. Evaluation of open information extraction methods using Reuters-21578 database, en: 2nd International Conference on Machine Learning and Soft Computing (ICMLSC ’18). Presentado en 2nd International Conference on Machine Learning and Soft Computing (ICMLSC ’18).

Rodríguez, J.M., Merlino, H.D., Pesado, P., García-Martínez, R., 2017. Automatización de la extracción de características en tareas de análisis de sentimiento. Presentado en XXIII Congreso Argentino de Ciencias de la Computación (CACIC 2017).

Rodríguez, J.M., Merlino, H.D., Pesado, P., García-Martínez, R., 2016b. Clasificación de Distintos Conjuntos de Datos Utilizados en Evaluación de Métodos de Extracción de Conocimiento Creados para la Web. Presentado en XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016).

Roy, A., Park, Y., Lee, T., Pan, S., 2019. Supervising unsupervised open information extraction models, en: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 728-737.

Saha, S., Mausam, 2018. Open Information Extraction from Conjunctive Sentences, en: Proceedings of the 27th International Conference on Computational Linguistics. Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp. 2288-2299.

Sandhaus, E., 2008. The New York Times Annotated Corpus.

Song, S., Lin, Y., Guo, B., Di, Q., Lv, R., 2018. Scalable Distributed Semantic Network for knowledge management in cyber physical system. J. Parallel Distrib. Comput. 118, 22-33. https://doi.org/10.1016/j.jpdc.2017.11.014

Stanovsky, G., Michael, J., Zettlemoyer, L., Dagan, I., 2018. Supervised open information extraction, en: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 885-895.

Steinbach, M., Karypis, G., Kumar, V., 2000. A comparison of document clustering techniques.

Tan, S.S., Lim, T.Y., Soon, L.-K., Tang, E.K., 2016. Learning to extract domain-specific relations from complex sentences. Expert Syst. Appl. 60, 107-117. https://doi.org/10.1016/j.eswa.2016.05.004

Tandon, N., Melo, G., Weikum, G., 2014. Acquiring comparative commonsense knowledge from the web, en: Proceedings of the AAAI Conference on Artificial Intelligence.

Tang, J., Lu, Y., Lin, H., Han, X., Sun, L., Xiao, X., Wu, H., 2021. Syntactic and Semantic-driven Learning for Open Information Extraction. ArXiv Prepr. ArXiv210303448.

Tran, X.-C., Nguyen, L.-M., 2021. ReLink: Open Information Extraction by Linking Phrases and Its Applications, en: Goswami, D., Hoang, T.A. [Eds.], Distributed Computing and Internet Technology, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 44-62. https://doi.org/10.1007/978-3-030-65621-8_3

Truong, D., Vo, D.-T., Nguyen, U. T, 2017. Vietnamese open information extraction. Presentado en Symposium on Information and Communication Technology.

Tseng, Y.-H., Lee, L.-H., Lin, S.-Y., Liao, B.-S., Liu, M.-J., Chen, H.-H., Etzioni, O., Fader, A., 2014. Chinese Open Relation Extraction for Knowledge Acquisition, en: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Volume 2: Short Papers. Association for Computational Linguistics, Gothenburg, Sweden, pp. 12-16. https://doi.org/10.3115/v1/E14-4003

Vo, D.-T., Bagheri, E., 2018. Self-training on refined clause patterns for relation extraction. Inf. Process. Manag. 54, 686-706. https://doi.org/10.1016/j.ipm.2017.02.009

Wang, X., Zhang, Y., Li, Q., Chen, Y., Han, J., 2018. Open Information Extraction with Meta-pattern Discovery in Biomedical Literature, en: Proceedings of

the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics. ACM, Washington DC USA, pp. 291-300. https://doi.org/10.1145/3233547.3233594

Wang, Y.P., Qiao, X.F., Yang, D.Z., Huang, J.Y., Chen, J.S., Ma, D.G., Dong, L.S., 2015. Investigation of electron transport properties in Li2CO3-doped Bepp2 thin films. Org. Electron. 26, 86-91. https://doi.org/10.1016/j.orgel.2015.07.023

White, A.S., Reisinger, D., Sakaguchi, K., Vieira, T., Zhang, S., Rudinger, R., Rawlins, K., Van Durme, B., 2016. Universal Decompositional Semantics on Universal Dependencies, en: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Austin, Texas, pp. 1713-1723. https://doi.org/10.18653/v1/D16-1177

Wu, F., Weld, D.S., 2010. Open information extraction using Wikipedia, en: 48th Annual Meeting of the Association for Computational Linguistics. pp. 118-127.

Wu, X., Wu, B., 2017. The crfs-based chinese open entity relation extractio, en: IEEE [Ed.], DSC. Presentado en DSC, pp. 405-411.

Xavier, A.V.S., Almeida, R.C., Chaves, D.A.R., Bastos-Filho, C.J.A., Martins-Filho, J.F., 2015. Spectrum continuity based routing algorithm for flexible grid optical networks. Presentado en 2015 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC), IEEE, Porto de Galinhas, Brazil, pp. 1-5. https://doi.org/10.1109/IMOC.2015.7369203

Xu, J., Gan, L., Deng, L., Wang, J., Yan, Z., 2015. Dependency parsing based Chinese open relation extraction, en: 2015 4th International Conference on Computer Science and Network Technology (ICCSNT). IEEE, pp. 552-556.

Yahya, M., Steven, E.W., Gupta, R., Halevy, A., 2014. Renoun: fact extraction for nominal attributes, en: Empirical Methods in Natural Language Processing (EMNLP).

Yang, Y., Liu, X., 1999. A re-examination of text categorization methods, en: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. pp. 42-49.

Zhan, J., Zhao, H., 2020. Span model for open information extraction on accurate corpus, en: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 9523-9530.

Zhao, Y., Karypis, G., 2001. Criterion functions for document clustering: Experiments and analysis.

Zhila, A., Gelbukh, A., 2014. Open information extraction for spanish language based on syntactic constraints, en: Proceedings of the ACL 2014 Student Research Workshop. pp. 78-85.

Zhu, Q., Ren, X., Shang, J., Zhang, Y., Xu, F.F., Han, J., 2018. Open Information Extraction with Global Structure Constraints, en: Companion of the The Web Conference 2018 on The Web Conference 2018 - WWW '18. ACM Press, Lyon, France, pp. 57-58. https://doi.org/10.1145/3184558.3186927