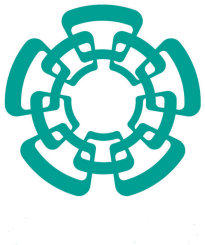


Centro de Investigación y de Estudios Avanzados
del Instituto Politécnico Nacional

Unidad Cinvestav Tamaulipas

Método de aprendizaje probabilístico para el reconocimiento de entidades nombradas

Tesis que presenta:

Cristopher Roberto Gaytán Díaz

Para obtener el grado de:

Maestro en Ciencias
en Ingeniería y Tecnologías Computacionales

Dr. Iván López Arévalo, Co-Director
Dr. Edwyn Javier Aldana Bobadilla, Co-Director

Cd. Victoria, Tamaulipas, México Diciembre, 2018

La tesis presentada por Cristopher Roberto Gaytán Díaz fue aprobada por:

Dr. Hiram Galeana Zapién

Dr. José Luis González Compeán

Dr. Iván López Arévalo, Co-Director

Dr. Edwyn Javier Aldana Bobadilla, Co-Director

Cd. Victoria, Tamaulipas, México, 13 de Diciembre de 2018

Hakuna Matata

Agradecimientos

Gracias a Dios porque de él, y por él, y para él, son todas las cosas.

A mi padre, que sin él nada de lo que hasta hoy he logrado hubiese sido posible. Gracias a él aprendí que con amor y sudor todo se puede.

A mi madre, que sin ella no sería la persona en la que hasta ahora me he convertido, pues de ella aprendí lo que ninguna escuela puede enseñar. Gracias por cada día aligerar el gran peso y esfuerzo que demandó este trabajo.

A mi hermano y a su esposa, por formar parte de esta importante etapa de mi vida. Gracias por cada vez que se les presentaba la ocasión preguntarme como iba mi trabajo de tesis, por preocuparse por mi y por todo el apoyo brindado.

Al Dr. Edwyn Javier Aldana Bobadilla, porque gracias a él logré culminar este trabajo. Por compartir su conocimiento y filosofía, pero sobre todo, gracias por a cada observación y opinión prepararme para la siguiente etapa de mi vida.

Al Dr. Iván López Arévalo, por guiarme en la senda de la investigación. Por enseñarme la manera correcta de hacer las cosas y mi trabajo. Pero sobre todo, gracias por brindarme su apoyo, confianza y paciencia a lo largo de este trabajo de tesis.

Al Dr. José Luis González Compeán y al Dr. Hiram Galeana Zapién por tomarse el tiempo de comprender mi tema. Por sus valiosas observaciones, pero sobre todo, por preocuparse de que realizara un mejor trabajo.

A mis estimados, por haber estado ahí siempre que los necesité, aún y cuando el tiempo expandió el espacio que había entre cada uno de nuestros universos.

A mis mejores conocidos, Los Gatos, que aunque siempre estuvieron metiéndome en problemas, nada de esto se hubiese convertido en una grata e inolvidable aventura. Meow.

Gracias a todas aquellas personas, investigadores, compañeros y personal administrativo de la unidad CINVESTAV Tamaulipas que de una u otra manera contribuyeron en mi formación.

Al CONACyT por el apoyo económico que me permitió terminar mis estudios profesionales y al CINVESTAV por la oportunidad de estudiar una maestría.

Índice General

Índice de Figuras

Índice de Tablas

Índice de Algoritmos

Resumen

Método de aprendizaje probabilístico para el reconocimiento de entidades nombradas

por

Cristopher Roberto Gaytán Díaz
Unidad Cinvestav Tamaulipas
Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, 2018
Dr. Iván López Arévalo, Co-Director
Dr. Edwyn Javier Aldana Bobadilla, Co-Director

El reconocimiento de entidades nombradas (NER, por sus siglas en inglés) es una tarea en el área de extracción de información (IE, por sus siglas en inglés) que tiene como objetivo buscar, localizar y clasificar los elementos clave en un texto. Estos elementos pueden representar personas, organizaciones, lugares, etc. La tarea de NER es utilizada en la mayoría de las tareas de IE, así como en algunas tareas de procesamiento de lenguaje natural (NLP, por sus siglas en inglés).

Usualmente los métodos de NER incorporan análisis de estructuras propias del lenguaje que las hacen dependientes del mismo, tales como reglas gramaticales, sintácticas y morfológicas. Sin embargo, dado que la gramática del lenguaje natural es dependiente del contexto, dichas técnicas no pueden resolver efectivamente todas las ambigüedades que pudieran presentarse en un texto dado. Por lo anterior, han surgido métodos basados en estadística y en aprendizaje automático que abordan de manera distinta dicho problema. Estos métodos requieren de un gran volumen de datos de entrenamiento (ejemplos de entidades nombradas conocidas a priori). Sin embargo, estos conjuntos de entrenamiento son obtenidos a partir de procesos de etiquetado manual, que es la introducción de anotaciones por humanos que identifican a las entidades nombradas en los textos que son usados para entrenar métodos de NER, los cuales implican un importante costo en tiempo, personas e infraestructura.

En esta tesis se presenta un método de NER basado en un clasificador Bayesiano simple que reduce su dependencia del lenguaje ya que no requiere procesos de etiquetado manual. Para ello

se propone un proceso automatizado de recolección de datos que permite obtener documentos de texto, a partir de la Web, que incluyen entidades nombradas. Estos textos son procesados para extraer información circundante (n-gramas) de cada entidad nombrada, los cuales son utilizados para generar un clasificador basado en un modelo probabilístico. Este clasificador, sin la necesidad de utilizar reglas gramaticales, sintácticas o morfológicas, dada las palabras que contiene un n-grama determina probabilísticamente si éste está o no asociado a una entidad nombrada, así como el tipo al que pertenece.

La experimentación realizada para evaluar el desempeño del método propuesto se compone de dos escenarios de pruebas. En el primer escenario el método fue entrenado y probado con conjuntos de documentos de texto en español recolectados con la herramienta de filtrado y búsqueda web. En un segundo escenario el método fue evaluado con el benchmark CoNLL para el idioma español. Los resultados de ambos escenarios fueron evaluados con las métricas de desempeño *precision*, *recall* y *f-measure*. Los resultados de dicha experimentación arrojan resultados prometedores que permiten concluir que el método funciona adecuadamente, cumpliendo así con los objetivos planteados.

Abstract

Probabilistic Learning Method for Named Entity Recognition

by

Cristopher Roberto Gaytán Díaz
Cinvestav Tamaulipas
Research Center for Advanced Study of the National Polytechnic Institute, 2018
Dr. Iván López Arévalo, Co-Advisor
Dr. Edwyn Javier Aldana Bobadilla, Co-advisor

Named-Entity Recognition (NER) is a task of Information Extraction (IE) for locating and classifying important elements in a given text. This elements represent objects, persons, organizations, places, etc. The NER is used in some subtasks of IE and Natural Language Processing (NLP).

The NER methods usually include analysis of languages structures that make them language-dependent, such as grammatical, syntactic and morphological rules. However, because the grammar of natural languages are context-dependent, these methods can't manage effectively all the ambiguities that a given text could have. For this reason some methods based on statistical and machine learning approaches has been proposed, which deals with the NER problem in a different way. These methods, for their effectiveness, require a large volume of training data (examples of named entities known a priori). However, these datasets are obtained from a manual tagging processes, which is the insertion of tags by humans to identify named entities in texts, that are used to train NER methods. This manual tagging processes involves importants costs in time, people, and infrastructure.

This work presents a NER method based on a naive Bayes classifier that reduces its language dependence because it doesn't depend directly on a manual tagging processes. An automatic process of data recollection is proposed to collect text documents from the Web, that include named entities. The text of each document is taken to extract the surrounding n-grams of each named entity. This set of extracted n-grams is the training dataset for the probabilistic method. This classifier, without using grammatical, syntactic and morphological rules, determines probabilistically from the words of

a given n-gram if such words are associated or not with a named entity, as well as its type.

The experimentation of the proposed method was divided in two escenarios. In the first one, the method was evaluated (trained and tested) with text documents written in Spanish collected by the process of data recollection described before. In the second one, the method was evaluated with the benchmark CoNLL for Spanish language. The results of both escenarios were measured with the metric precision, recall, and f-measure. The results of this evaluation allow to conclude that the method works properly and that it complies with the objectives of this work.

Acrónimos

CRF	Conditional Random Field
CoNLL	Computational Natural Language Learning
HMM	Hidden Markov Model
IE	Information Extraction
IR	Information Retrieval
ME	Maximum Entropy
MUC	Message Understanding Conference
NER	Named Entity Recognition
NLP	Natural Language Processing
RDF	Resource Description Framework
SPARQL	SPARQL Protocol and RDF Query Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator

1

Introducción

En este capítulo se presenta el contexto y la problemática asociada al reconocimiento de entidades nombradas. Asimismo, se presenta la hipótesis de investigación y los objetivos planteados en este trabajo de tesis, así como la metodología empleada en el desarrollo del mismo. Finalmente, se describe la organización del presente documento.

1.1 Contexto

La Web es un repositorio de grandes volúmenes de datos estructurados y no estructurados. Estos últimos son de particular interés ya que no es posible encontrar directamente información y conocimiento en ellos. Una instancia de este tipo de datos son los documentos de texto. Éstos son objeto de estudio del área denominada procesamiento de lenguaje natural (NLP, por sus siglas en inglés). Entre las tareas fundamentales del NLP que dan estructura a la información contenida en un documento se encuentra el etiquetado gramatical, la búsqueda de respuestas, el análisis de sentimiento y el reconocimiento de entidades nombradas. Esta última tarea es la que se aborda en este trabajo de tesis.

El reconocimiento de entidades nombradas (NER, por sus siglas en inglés) es una búsqueda de estructura semántica que consiste en encontrar aquellos elementos de un documento que hacen referencia a lugares, personas, organizaciones u otros conceptos y objetos de interés que tienen un significado universal [77], tales como Torre Eiffel, Nueva York, Paris Hilton, Michael Jordan, ONU, OMS, etc. Desde el punto de vista computacional, detectar y clasificar en un texto este tipo de elementos, conocidos como entidades nombradas, no es fácil. Típicamente los textos que las contienen están conformados por sentencias en lenguaje natural cuya gramática es dependiente del contexto, razón por la cual resulta complicado resolver adecuadamente el universo de posibles ambigüedades. Por ejemplo, dada la siguiente sentencia y habiendo eliminado tildes:*"**Paris Hilton** es famosa, ella es dueña de los hoteles **Hilton** en **Paris**"*, la palabra "Paris", al igual que "Hilton", representarán un ambigüedad si se desconoce el contexto que indica que "Paris Hilton" es una persona.

La mayoría de los métodos de NER propuestos en la literatura usualmente incorporan reglas y estructuras gramaticales propias del lenguaje para realizar la detección y clasificación de entidades nombradas [8, 17, 45, 64, 76]. Sin embargo, dado que la gramática del lenguaje natural es dependiente del contexto, dichas técnicas no pueden resolver efectivamente todas las ambigüedades que pudieran presentarse en un texto dado [53].

En vista de lo anterior han surgido enfoques basados en métodos estadísticos que abordan de manera distinta dicho problema [4, 7, 43, 60, 72]. Si bien estos últimos enfoques han mejorado en efectividad, también inducen otros problemas entre los que destaca el proceso de etiquetado manual. Este proceso de etiquetado consiste en la introducción manual de anotaciones que identifican a las entidades nombradas en los textos que son usados para entrenar métodos de NER. Evidentemente este proceso es ineficiente ya que la mayoría de estas técnicas estadísticas requieren de un gran volumen de datos para obtener un alto grado de eficiencia.

En este trabajo de tesis se presenta un enfoque probabilístico para el NER que no depende directamente de procesos de etiquetado manual y que reduce la incorporación de reglas basadas en estructuras gramaticales propias de un idioma.

1.2 Definición del problema

Los métodos de NER basados en reglas, como su nombre lo indica, incorporan conjuntos de reglas gramaticales, morfológicas o sintácticas para resolver su tarea [8, 17, 45, 64, 76]. Estos son rápidos y eficientes, sin embargo, definir un conjunto de reglas para el NER implica importantes costos en esfuerzo por parte de lingüistas expertos. Ademas, también implica costos en tiempo, infraestructura y personas ya que, dado que el idioma para el cual han sido diseñados se encuentra en constante evolución (textos nuevos), es necesario mantenerlos actualizados para que resuelvan adecuadamente las diferentes ambigüedades. Dada la cardinalidad del espacio de posibles ambigüedades del lenguaje natural, es prácticamente imposible incluirlas como reglas en una base de conocimiento que permita a una computadora inferir cuándo una palabra, o más, representan una entidad de cierto tipo.

Dado esta gran desventaja que presentan los métodos de NER basados en reglas, recientes enfoques han recurrido a modelar el problema como un problema estadístico [4, 7, 43, 47, 60, 72]. Estos métodos encuentran un modelo probabilístico a partir del análisis de estructuras gramaticales, sintácticas y morfológicas de conjuntos de textos que contienen ejemplos de entidades nombradas previamente identificadas mediante etiquetas. Este modelo probabilístico permite a la computadora determinar con un grado de incertidumbre cuándo una o varias palabras representan una entidad y su tipo. Si bien estos métodos basados en estadística logran resolver algunos de los problemas de los métodos basados en reglas, continúan utilizando supuestos basados en estructuras propias del lenguaje, los que los hace dependientes del lenguaje. Por otro lado, también inducen nuevos problemas, dentro de los que destaca el proceso de etiquetado manual en los textos con entidades nombradas que son utilizados para el entrenamiento.

Dado los anteriores problemas que presentan los métodos de NER basados en reglas o en estadística, en este trabajo de tesis se aborda el problema de dependencia de lenguaje que presentan los métodos de NER al utilizar supuestos basados en estructuras gramaticales, sintácticas y morfológicas. Para ello se propone un método de NER que genera un modelo probabilístico a partir

de únicamente información sintáctica, reduciendo así su dependencia del lenguaje al dejar de utilizar información gramatical y morfológica. Por otro lado, también se aborda el problema de etiquetado manual en los textos que son utilizados para entrenar métodos de NER basados en estadística.

1.3 Hipótesis

Con base en el problema anterior, se plantea la siguiente hipótesis de investigación.

> *Dado que los enfoques actuales de NER dependen en gran medida de características del lenguaje, es posible crear un método de NER basado en un modelo probabilístico que incorpore información sintáctica que reduzca las dependencias del lenguaje, con respecto a las técnicas actuales del estado del arte.*

1.4 Objetivos

A continuación se presentan los objetivos de este trabajo para confirmar la hipótesis anterior.

1.4.1 Objetivo general

Obtener un método de NER basado en un modelo probabilístico que reduzca, con respecto a técnicas existentes, la dependencia de información asociada a estructuras particulares del lenguaje.

1.4.2 Objetivos particulares

1. Contar con un método de NER que reduzca la dependencia de información asociada a estructuras gramaticales, sintácticas y morfológicas propias del lenguaje.
2. Contar con un método de aprendizaje probabilístico que permita reconocer entidades nombradas a partir de información sintáctica.

3. Definir un método de filtrado de búsqueda web que permita explorar de forma sistemática y focalizada diferentes recursos textuales de la Web para formar un corpus que contenga entidades nombradas.

1.5 Metodología

Para corroborar la hipótesis de investigación y cumplir con los objetivos propuestos, se llevó a cabo cada uno de los procesos que se ilustran en la Figura 1.1. Estos son descritos a continuación.

1. **Definición y alcance de la investigación.** Se realizó una revisión de la literatura actual con el fin de definir los alcances de la propuesta y determinar sus contribuciones.

2. **Estudio de enfoques de NER.** Del análisis de trabajos relacionados, se definieron sus ventajas y desventajas, así como técnicas de benchmarking y métricas de desempeño.

3. **Adecuación y configuración de herramienta de filtrado de búsqueda web.** Esta herramienta permitió explorar de forma sistemática y focalizada diferentes recursos textuales de la Web que contienen entidades nombradas.

4. **Recolección de datos.** Utilizando la herramienta mencionada se recolectaron recursos textuales de la Web de diferentes contextos para formar un corpus que contenga entidades.

5. **Procesamiento y transformación.** Se aplicaron tareas de preprocesamiento sobre el corpus con entidades nombradas recolectado. Además, se extrajo información circundante de cada entidad nombrada, la cual fue utilizada para entrenar y probar el método propuesto.

6. **Diseño del modelo de NER.** Se realizó el análisis de la información circundante de cada entidad nombrada para generar un modelo probabilístico con el cual fue posible determinar, a partir de únicamente información sintáctica, cuando una palabra es una entidad nombrada.

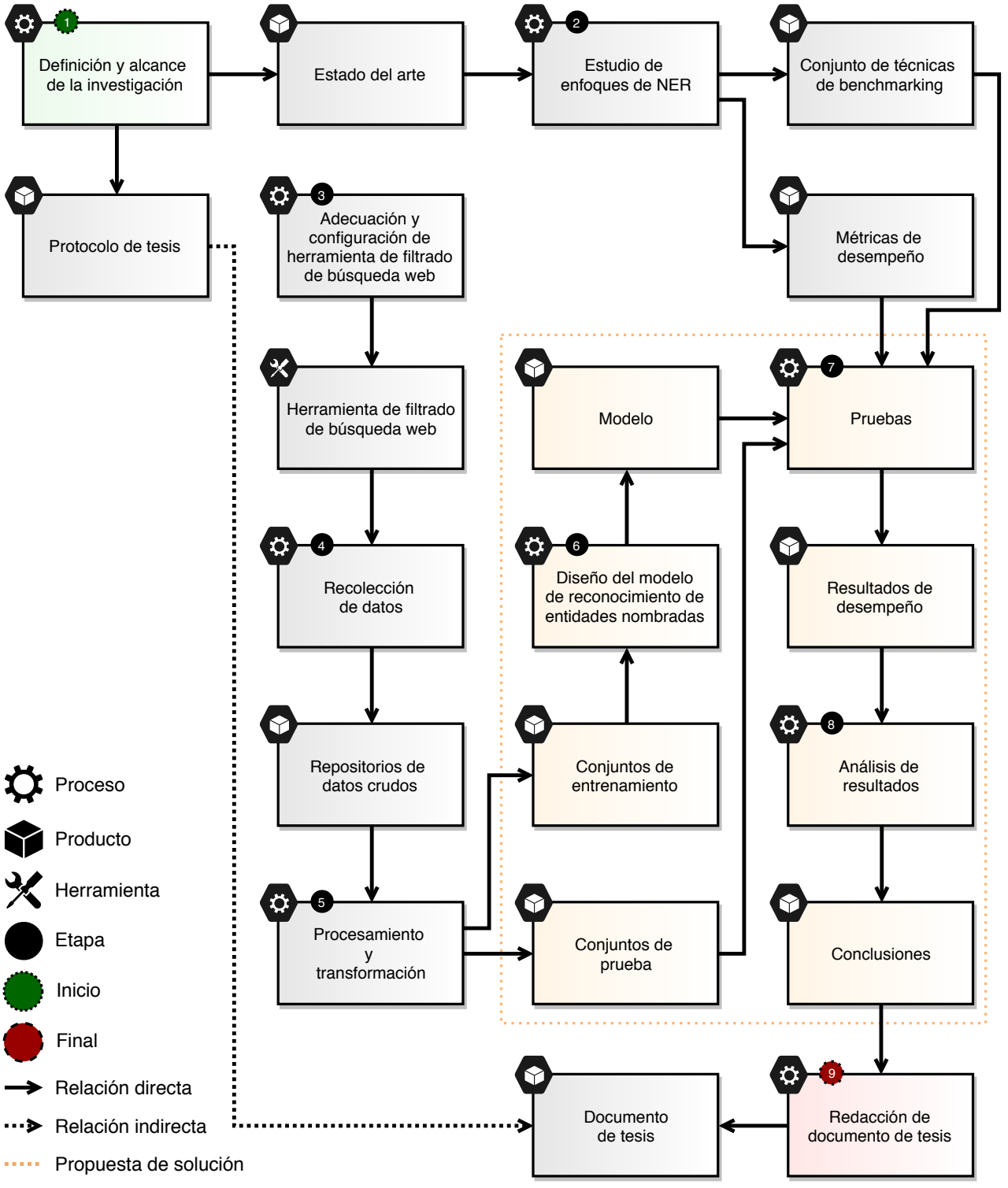


Figura 1.1: Metodología de investigación.

7. **Pruebas.** Tomando el modelo probabilístico, se realizaron pruebas con los repositorios de datos previamente creados, con un conjunto de métricas de desempeño y con benchmarks.

8. **Análisis de resultados.** Los resultados obtenidos se compararon con aquellos analizados en el estudio de enfoques de NER. Esto permitió determinar el desempeño del método propuesto relativo a otras propuestas, corroborando así la hipótesis planteada en este trabajo de tesis.

9. **Redacción del documento de tesis.** Al termino del desarrollo de todos y cada uno de los puntos anteriormente descritos, se realizó la redacción del presente documento de tesis donde se reporta cada uno de los resultados obtenidos en cada una de estas actividades.

1.6 Organización del documento

El resto de este documento está organizado de la siguiente manera. En el Capítulo 2 se describe el fundamento teórico requerido para el desarrollo de este trabajo de investigación. En el Capítulo 3 se presenta una revisión de los trabajos del estado del arte con respecto a la tarea de NER. En el Capítulo 4 se describe el método de aprendizaje propuesto en este trabajo de acuerdo a la metodología ilustrada en la Figura 1.1, describiendo a detalle la recolección, procesamiento y transformación de datos, así como el diseño del método de NER. En el Capítulo 5 se presenta la evaluación experimental y el análisis de los resultados obtenidos a partir de la implementación del método propuesto. En el Capítulo 6 se muestran las conclusiones, contribuciones y limitaciones de este trabajo, así como el trabajo futuro.

2

Fundamento teórico

En este capítulo se presenta el fundamento teórico necesario para el desarrollo de este trabajo, así como los conceptos básicos que involucra el reconocimiento de entidades nombradas en texto.

2.1 Procesamiento de lenguaje natural

El procesamiento de lenguaje natural (NLP, por sus siglas en inglés) es el estudio del modelado matemático y computacional de varios aspectos del lenguaje, así como del desarrollo de una amplia gama de sistemas [29]. En este sentido, NLP pretende modelar cómo es que los humanos entendemos y usamos el lenguaje con el propósito de desarrollar herramientas y técnicas que ayuden a los sistemas de cómputo a entender y manipular lenguajes para realizar tareas deseadas [13].

Para modelar dichos aspectos del lenguaje, usualmente se hacen uso de estructuras propias del lenguaje, tales como las gramaticales, sintácticas y morfológicas. Dichas estructuras son representadas a manera de reglas y aplicadas sobre conjuntos de datos para analizar y extraer información con la cual sea posible generar modelos matemáticos o estadísticos. Las estructuras que usualmente se aplican en este análisis lingüístico se dividen en los siguientes niveles [37].

- Fonético: el estudio de los sonidos físicos del discurso humano (pronunciaciones y discursos).
- Morfológico: el estudio de la estructura interna de las palabras (lexemas, morfemas, etc).
- Léxico: el significado de una palabra en su uso común (lexicones[1], gazetteers[2], etc).
- Sintáctico: es la parte de la gramática que estudia las formas en que se combinan las palabras, tales como sintagmas y oraciones gramaticales.
- Semántico: el estudio del significado de las palabras dentro de oraciones en diferentes contextos, tales como la gramática y discursos.
- Discurso: el estudio de textos conformados por mas de una sentencia (temas, contextos, etc).
- Pragmático: el estudio del modo en que el contexto influye en la interpretación del significado.

Un sistema NLP utiliza, en cierta manera, uno o algunos de los niveles listados anteriormente para resolver alguna tarea en específico. Por ejemplo, un sistema NLP típicamente empieza con un análisis a nivel de palabra para determinar su estructura morfológica, continua a nivel de sentencia para determinar el orden y el significado de las palabras a partir de la gramática y semántica. Y termina a nivel de documento para determinar el contexto y dominio de lo que se habla.

Las tareas de NLP, según Nadkarni y Ohno-Machado [51], pueden dividirse en 2 categorías. La primera categoría corresponde a las tareas de *bajo-nivel*, las cuales intentan resolver problemas básicos que usualmente se presentan en el lenguaje natural. Algunas de estas tareas de bajo-nivel se describen a continuación.

- Lematización: encuentra el lema de una palabra dada.
- Segmentación morfológica: encuentra los morfemas de cada palabra, así como su categoría.

[1] Colección o lista de palabras de una lengua.
[2] *Gazetteer* es un término para referirse a un lexicón de un dominio en específico (e.g. países, ciudades, colegios).

- Etiquetado gramatical: encuentra la categoría gramatical de cada palabra de un texto dado, tales como sustantivos, adjetivos, artículos, pronombres, verbos, adverbios, preposiciones, etc.
- *Parsing* (análisis sintáctico): encuentra la estructura sintáctica (árbol) de una oración dada.
- Segmentación de sentencias: encuentra el inicio y fin de todas las sentencias de un texto.
- *Stemming*: encuentra la raíz (en inglés *stem*) de una palabra dada.
- Segmentación de palabras: encuentra las palabras de una sentencia dada.

La segunda categoría corresponde a las tareas de *alto-nivel*, las cuales intentar resolver problemas específicos a partir de tareas de bajo-nivel. Algunas de estas tareas se describen a continuación.

- Extracción de terminología: extrae términos relevantes de un conjunto de documentos.
- Traducción automática: traduce un texto dado de un lenguaje natural a otro.
- Búsqueda de respuestas: encuentra la respuesta a una pregunta planteada y escrita en lenguaje natural.
- Reconocimiento y segmentación de temas: extrae los temas de un texto dado y separa su contenido (sentencias o párrafos) por temas.
- Resumen automático: sintetiza un texto de forma automática.
- Reconocimiento de entidades nombradas: extrae las elementos de un texto que hacen referencia a nombres propios, como lugares, personas, organizaciones, etc.
- Extracción de relaciones: identifica las relaciones entre las entidades nombradas de un texto.
- Resolución de correferencia: encuentra los vínculos (referencias) entre las entidades nombradas de un texto dado.

El área de NLP puede ser dividido en diferentes subáreas, tales como Lingüística Computacional, Comprensión del Lenguaje Natural, Generación de Lenguajes Naturales, Recuperación de Información y Extracción de Información [30]. Estas subáreas utilizan tareas de alto-nivel en específico, dependiendo del subárea de estudio. Por ejemplo, la extracción de información tiene como objetivo extraer información útil de textos escritos en lenguaje natural [21], es decir, descubrir nueva información de la cual no se tenía conocimiento. La extracción de información utiliza 3 tareas de NLP para lograr su objetivo: resolución de correferencia, extracción de relaciones y reconocimiento de entidades nombradas.

2.2 Extracción de información

En los últimos años la Web se ha convertido en un repositorio de información textual y multimedia. La información que la Web contiene usualmente es de libre acceso, lo que a convertido a la Web en un foco de atención para explotar la información que contiene para la toma de decisiones, análisis de tendencias, o por la simple necesidad de búsqueda de información. Sin embargo, debido a la gran cantidad de información que la Web almacena, es prácticamente imposible que los humanos procesen toda esa información para convertirla en información útil, importante y de rápido acceso, con la cual sea posible generar/encontrar nuevo conocimiento. Por otro lado, debido a la escasez de estructura global en la información que contiene la Web, es imposible para los sistemas de cómputo acceder a ésta y utilizarla para responder consultas, al menos no igual que en las bases de datos, donde la información se encuentra estructurada. Esta necesidad de transformar la información no estructurada en formatos estructurados para su posterior uso, procesamiento o análisis, se ha convertido en un objeto de estudio en los últimos años. La subárea del NLP que trata de resolver este tipo de problemas es la *extracción de información*.

La extracción de información (IE, por sus siglas en inglés) es la subárea de NLP que tiene como tarea generar información estructurada a partir de texto no estructurado [53]. Su principal objetivo es el de analizar texto escrito en lenguaje natural para encontrar hechos o eventos en dicho texto, como "quién" hizo "qué" a "quién", "cuándo" y "dónde" [49]. Por lo anterior, IE genera de manera automática nuevo conocimiento a partir de procesar información no estructurada [46].

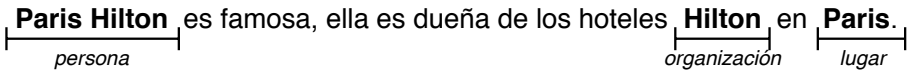
IE muchas veces es confundido con la recuperación de información (IR, por sus siglas en inglés). Este último, su principal tarea es el de recuperar un subconjunto de documentos textuales que sean relevantes a una consulta en particular [71], tal y como lo hacen los motores de búsqueda web. Sin embargo, esta lista de documentos relevantes recuperada usualmente no provee ninguna información extra del contenido de cada documento. Para obtener información extra de estos documentos, IE intenta procesar su contenido/información textual con la finalidad de descubrir nuevo conocimiento que facilite su posterior análisis, procesamiento o búsqueda [53]. En otras palabras, IR intenta recuperar documentos relevantes de un conjunto dado, mientras que IE intenta extraer información relevante de un conjunto de documentos [59]. IE e IR son técnicas que se pueden complementar y combinar adecuadamente en diferentes maneras de acuerdo a las necesidades del usuario.

El interés que se ha tenido en utilizar en sistemas informáticos tareas de IE ha incrementado en los últimos años, algunas de las razones han sido por: las tareas de extracción están bien definidas, utiliza textos del mundo real, resuelve problemas difíciles relacionadas con el NLP y porque su desempeño en algunos casos puede compararse con el de un humano [14]. Sin embargo, existen numerosos casos en los que dicho desempeño no es comparable y esto se debe a que los sistemas de IE se enfrentan a las complejidades y ambigüedades del lenguaje natural, ya que existen infinidad de maneras en las que se pueden expresar hechos o eventos en un texto, los cuales se pueden encontrar en diferentes sentencias, documentos o repositorios [53].

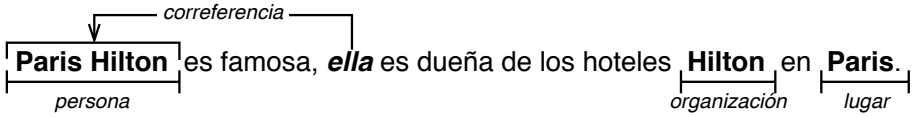
Estos sistemas de IE suelen clasificarse en dos categorías, según el enfoque que utilicen para resolver su tarea, los cuales son: los basados en reglas y los estadísticos. Los métodos basados en reglas comúnmente son aquellos que logran desempeños comparables con el de un humano, como

se dijo anteriormente, ya que éstos utilizan reglas que lingüistas expertos definen para la extracción, así como para un idioma y dominio en específico. Por otro lado, los métodos basados en estadística son aquellos que utilizan ejemplos etiquetados manualmente para entrenar modelos probabilísticos [19]. Estos dos enfoques son utilizados para resolver las tareas de IE, las cuales son:

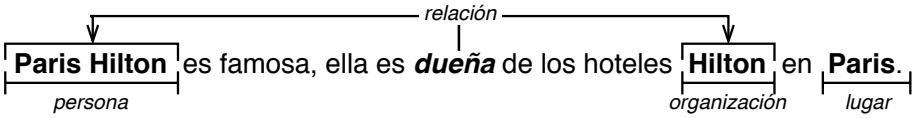
- **Reconocimiento de entidades nombradas.** Buscar, localizar y clasificar elementos clave en un texto sobre categorías predefinidas, tales como nombres de personas, organizaciones, lugares, eventos, actividades, objetos, entre otros, así como expresiones de hora, cantidad, valores monetarios y porcentajes [77]. Un ejemplo de ésto se muestra a continuación.



- **Resolución de la correferencia.** Tiene como objetivo detectar la correferencia de los vínculos entre las entidades nombradas de un texto dado, por ejemplo:



- **Extracción de relaciones.** Requiere la detección y clasificación de las menciones a relaciones semánticas. Un ejemplo de lo anterior se muestra a continuación.



Gracias a las tareas de NLP, IE logra dar estructura a documentos escritos en lenguaje natural de manera automática, permitiendo así inferir a partir de éstos nuevo conocimiento [31].

La arquitectura que típicamente se utiliza en un sistema de IE, según lo proponen Piskorski y Yangarber [53], así como Turmo y Ageno [71], se ilustra en la Figura 2.1. La tarea de reconocimiento de entidades nombradas de la IE es el tema principal que aborda este trabajo de tesis.

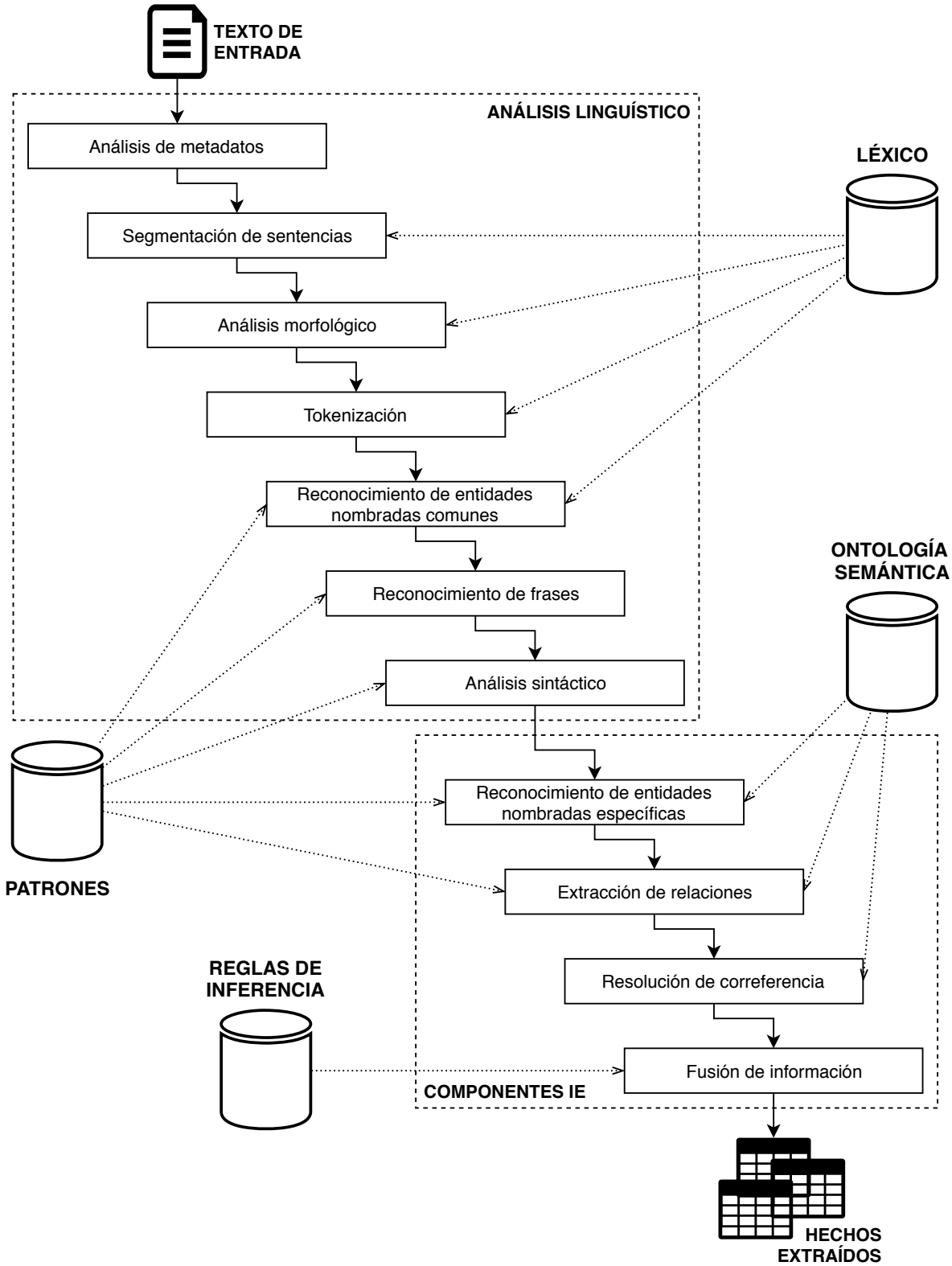


Figura 2.1: Arquitectura típica de un sistema de IE.

2.3 Reconocimiento de entidades nombradas

El elemento principal que utilizan la mayoría de las tareas de IE son las entidades nombradas. Una entidad nombrada es un elemento atómico de un texto el cual hace referencia a una persona, lugar, organización, etc. En otras palabras, una entidad nombrada es un objeto físico, imaginario o conceptual del cual se hace referencia por un nombre [66]. La tarea más importante de la IE es el de reconocer las entidades nombradas de un texto dado, ya que esta tarea afecta directamente el desempeño de algunas tareas de NLP [12, 30]. Formalmente el reconocimiento de entidades nombradas (NER, por sus siglas en inglés) es definido como el proceso de detectar las menciones de entidades nombradas que se encuentren en algún texto para después clasificarlas según su tipo [54].

Desde un punto de vista computacional, reconocer entidades nombradas no es una tarea fácil debido a la ambigüedad y segmentación [24, 30]. Por lo anterior, la tarea de NER se divide en la detección de nombres y en la clasificación de éstos, las cuales se detallan a continuación.

1. **Detección de los límites de la entidad.** Una entidad puede ser denotada a través de un nombre compuesto por una o más palabras. Para reconocer una entidad es necesario detectar, como se muestra en el siguiente ejemplo, dónde inicia y termina la entidad "Paris Hilton", "Hilton" y "Paris".

   *<inicio>***Paris Hilton***<fin>* es famosa, ella es dueña de los hoteles *<inicio>***Hilton***<fin>* en *<inicio>***Paris***<fin>*.

2. **Clasificación de la entidad.** Una entidad puede representar un lugar, una persona, una organización o algún otro objeto de interés. Toda entidad pertenece a una clase/categoría. Esta clasificación presenta un gran reto pues, como se muestra en el siguiente ejemplo, el lugar "Paris" y la organización "Hilton" representan una ambigüedad con la persona "Paris Hilton".

   *<persona>***Paris Hilton***</persona>* es famosa, ella es dueña de los hoteles *<organización>***Hilton***</organización>* en *<lugar>***Paris***</lugar>*.

Los sistemas de IE deberían ser fáciles de ajustar a nuevos dominios [32], ya que al momento de aplicar tareas, como la de NER, el tipo de dominio y documentos a ser usados son desconocidos en la mayoría de los casos. Por lo anterior los sistemas de NER deben ser robustos para ser capaces de adaptarse a diferentes y múltiples dominios, ya que se espera que sea aplicado a diversos conjuntos de documentos (e.g. textos históricos, artículos, noticias, informes médicos, páginas web, etc.) [56].

Muchas propuestas de NER han logrado ajustar sus sistemas a un dominio, aplicación y lenguaje en específico, a costo de procesos manuales o semiautomáticos [42]. Estos procesos comúnmente involucran la extracción de reglas gramaticales, morfológicas y sintácticas propias de un lenguaje, así como el etiquetado manual de sentencias que contienen ejemplos de entidades nombradas. Desafortunadamente, realizar estos procesos cada vez que se cambia de aplicación, domino o lenguaje, implica grandes costos en tiempo y esfuerzo.

Las fuentes de información que actualmente se encuentran disponibles para el entrenamiento o desarrollo de nuevos sistemas de NER son muy escasas ya que gran parte de las investigaciones realizadas en el área de NER se han enfocado en el estudio del Inglés, dejando de lado otros idiomas y dominios de los cuales existen pequeñas cantidades de datos de entrenamiento disponibles [24, 35, 50, 73]. La escasez de esta información, así como los grandes costos que implica realizar procesos manuales y semiautomáticos para ajustar los sistemas de NER a un dominio, aplicación y lenguaje en específico han planteado un nuevo reto en el área del NER: la independencia del lenguaje y los métodos no supervisados para el etiquetado del corpus[3] [26, 35].

En este trabajo se propone un método automático de recolección de datos para generar un corpus de sentencias etiquetadas, así como un método de NER que reduce su dependencia del lenguaje en comparación a las técnicas existentes.

[3]Conjunto extenso y ordenado de datos o textos científicos, literarios, etc., de situaciones reales que pueden servir de base a una investigación.

3

Estado del arte

En este capítulo se presenta una revisión de la literatura actual en el contexto del trabajo propuesto. Esta revisión incluye un análisis de trabajos relacionados, donde se indican las ventajas y desventajas que estos presentan al resolver el problema de NER.

3.1 Enfoques de NER

En la literatura se ha propuesto una diversidad de enfoques que resuelven los problemas de NER. Estas propuesta suelen clasificarse en dos tipos: basados en reglas y basados en estadística. A continuación se analizan algunas propuestas relacionadas con estos dos tipos de enfoques.

3.1.1 Enfoques basados en reglas

Los enfoques basados en reglas fueron las primeras propuestas que surgieron para el NER. Estos enfoques en su mayoría se componen de lexicones y gazetteers, así como de conjuntos de reglas gramaticales propias del lenguaje para el cual fue propuesto.

La mayor desventaja que este enfoque presenta es que los lexicones y gazetteers con los que son entrenados tienen que ser actualizados regularmente para resolver las diferentes ambigüedades que se van presentando conforme el idioma va evolucionando (textos nuevos). Por otro lado, las reglas que estas propuestas utilizan necesitan ser de buena calidad para dar buenos resultados, lo que implica conocimientos avanzados en el idioma para el que es propuesto. Los enfoques basados en reglas son rápidos y eficientes, sin embargo, estos son completamente dependientes del lenguaje e implican un importante esfuerzo/costo para mantenerlos actualizados [61].

En su mayoría se pueden identificar tres grupos de enfoques basados en reglas: (1) aquellos que utilizan Wikipedia[1] como fuente de datos, (2) los que se basan en árboles de decisión y (3) los que hacen uso de bases de conocimiento para recolectar, organizar y recuperar información para el NER de manera rápida y eficiente. A continuación se analizan estos enfoques.

3.1.1.1. Basados en Wikipedia

Wikipedia es una enciclopedia libre, donde cada uno de sus más de 45 millones de artículos están relacionadas con una entidad o concepto, lo que la hace una gran fuente de información útil para crear repositorios de entrenamiento [36]. Para el NER, Wikipedia ha sido utilizado para la creación de lexicones y gazetteers, sin embargo, el principal motivo de su utilización ha sido para la desambiguación de entidades nombradas. Por ejemplo, un sistema NER en la fase de detección podría detectar la palabra "Mercurio" como una entidad nombrada, sin embargo, al momento de realizar la clasificación podría no saber si etiquetarla como elemento químico, mitología o planeta.

Por lo anterior han surgido propuestas, como la de Bunescu y Pasca [8], en la cual extrajeron y etiquetaron manualmente artículos de Wikipedia. A partir de estos artículos entrenaron un método de NER capaz de desambiguar entidades nombradas. Dada esta capacidad de desambiguación, los autores argumentan que consiguen un sistema NER libre del dominio ya que, dependiendo de los artículos de Wikipedia que se utilicen para entrenar el método de NER, logran reconocer las entidades

[1] https://www.wikipedia.org/

nombradas de un texto dado sin importar el tema del que trate (e.g. deportes, economía, etc).

Dada esta necesidad de desambiguar entidades nombradas, Yosef y Hoart [76] utilizaron Wikipedia para proponer un nuevo método basado en grafos. Cada artículo de Wikipedia lo representaron como una entidad y, a la vez, los relacionaron a través de hipervínculos con otos artículos/entidades. En esta propuesta se recolectaron a priori los hipervínculos (relaciones) que existen entre cada entidad nombrada de Wikipedia. Estas relaciones son utilizadas para generar un grafo, con el cual logran la desambiguación. Si bien esta propuesta, dado un texto, logra reconocer las entidades nombradas que contiene, depende del contexto y de las entidades nombradas que éste contiene para lograr la desambiguación; tal y como se ilustra en la Figura 3.1.

Por otro lado, además de utilizar un método similar a los anteriormente mencionados, Dojchinovski y Kliegr [17] hacen uso de hipónimos para desambiguar entidades nombradas a grano fino o grueso. Con los hipónimos logran clasificar entidades desde clases muy genéricas como animales y comidas, hasta peces y frutas respectivamente.

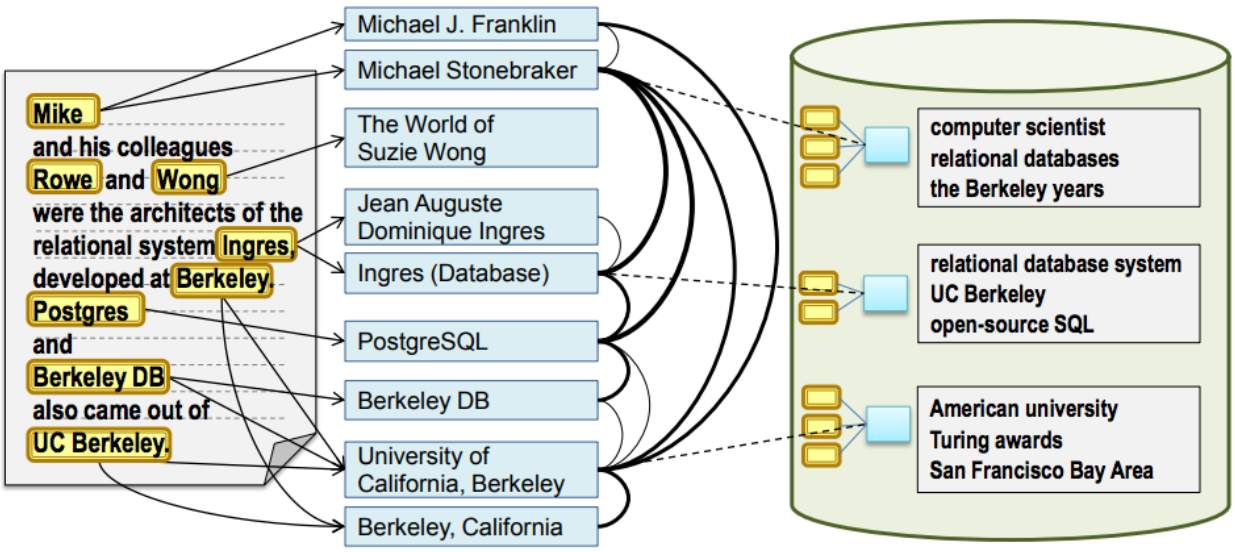


Figura 3.1: Ejemplo de un grafo de menciones de entidades nombradas [76].

3.1.1.2. Basados en árboles de decisión

Estos enfoques utilizan un conjunto de características, como gazetteers, secuencias de palabras, reglas gramaticales y conjuntos de patrones para construir árboles de decisión. Con estos árboles logran predecir, a partir de condiciones que ocurren de forma sucesiva, cuándo una o más palabras pertenecen a una entidad nombrada.

Por ejemplo, Abdallah y Shaalan [1] utilizaron el algoritmo J48 para construir un árbol de decisión que reconoce entidades nombradas en textos escritos en idioma árabe. Este árbol resultante contiene 1684 nodos; una parte de éste se puede observar en la Figura 3.2.

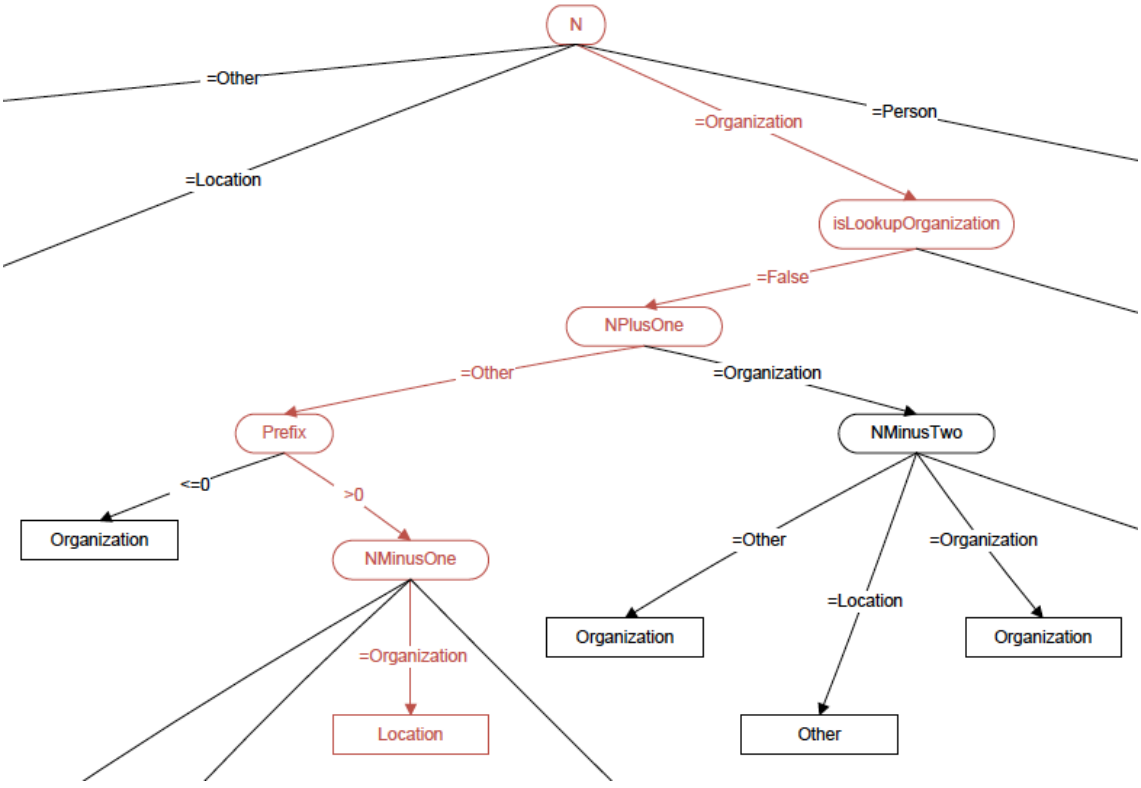


Figura 3.2: Ejemplo de un árbol de decisión para el NER [1].

Por otro lado, Szarvas y Farkas [64] utilizaron el algoritmo C4.5 para construir un árbol con las mismas características de la propuesta descrita anteriormente. Además, aplican el meta-algoritmo AdaBoostM1 para generar un conjunto de clasificadores. A partir de estos clasificadores y de un

sistema de votos, utilizan C4.5 para generar el árbol de decisión con mejor desempeño. Con esto realizan una combinación de modelos, con lo cual generan un enfoque de NER híbrido. La estructura del método de esta propuesta se ilustra en la Figura 3.3.

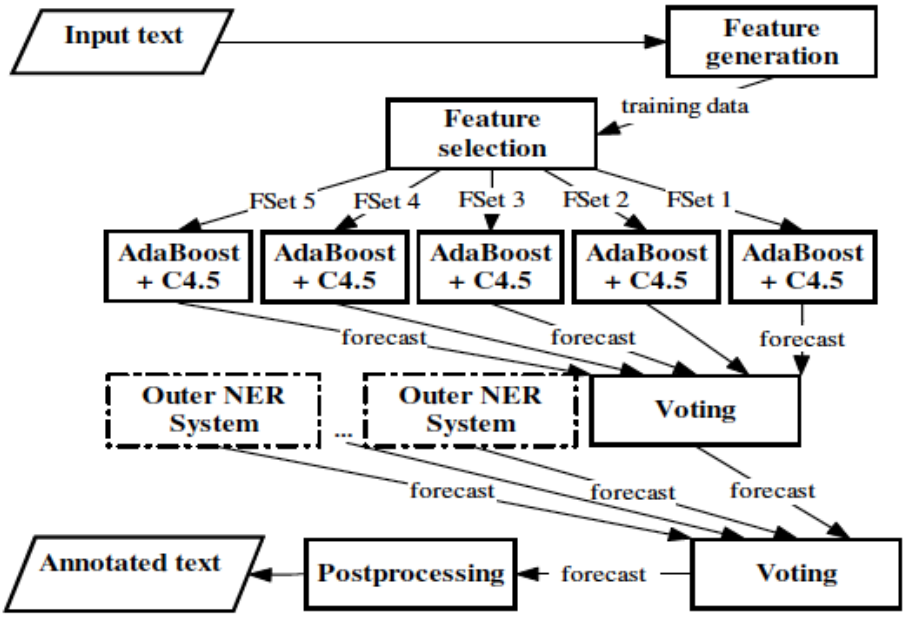


Figura 3.3: Estructura del sistema de NER híbrido propuesto en [64].

3.1.1.3. Basados en bases de conocimiento

Los métodos basados en bases de conocimiento, al igual que aquellos que se basan en Wikipedia, son utilizados para la creación de lexicones y gazetteers, así como para desambiguar entidades nombradas. Sin embargo, la diferencia que estos dos presentan es que las bases de conocimiento utilizan ontologías para especificar su estructura, lo cual permite recolectar, organizar y recuperar información para el NER de manera sintética, más rápida y eficiente en comparación a aquellos enfoques basados en Wikipedia.

Uno de los enfoques de NER basados en bases de conocimiento más conocidos es DBpedia Spotlight[2], propuesto por Mendes y Jakob [45]. Esta propuesta utiliza lexicones creados a partir de la base de conocimiento DBpedia[3] [5]. Utilizan grafos creados a partir de Wikipedia para desambiguar

[2]http://wiki.dbpedia.org/projects/dbpedia-spotlight
[3]http://wiki.dbpedia.org

entidades, por lo que este método depende del contexto para realizar una correcta desambiguación. El propósito de DBpedia Spotlight es reconocer mediante un identificador de recursos uniforme (URI, por sus siglas en inglés) todas las entidades nombradas de un texto dado. Cada URI corresponde a un concepto (entidad nombrada) de DBpedia.

Por otro lado, existen propuestas que utilizan múltiples bases de conocimientos, como la de Yosef y Hoart [76] que utiliza DBpedia, Yago [63] y Freebase[4]. En esta propuesta se genera un grafo basado en estas tres bases de conocimiento para realizar la clasificación y desambiguación de entidades. Sin embargo, utilizan Stanford NER Tagger[5] [68, 69] para la detección de entidades nombradas, es decir, hacen uso de una herramienta externa para detectar las entidades nombradas de un documento.

3.1.2 Enfoques basados en estadística

Los enfoques de NER basados en estadística surgen como solución a los grandes problemas que presentan los enfoques basados en reglas, tales como la construcción de gazetteers y lexiciones, así como la definición de reglas gramaticales de buena calidad.

Típicamente las propuestas de NER basadas en estadística presentan los siguientes dos componentes. Primero, éstos utilizan corpora[6] de textos en donde se tienen identificadas mediante una etiqueta las entidades nombradas que contiene. Para etiquetar dichos textos se realiza un proceso a priori en el que manualmente se marcan dichas entidades. Como segundo componente estas propuestas utilizan un modelo estadístico que, a partir de estos textos etiquetados, les permita representar probabilísticamente una entidad nombrada.

En el caso de los métodos de NER basados en estadística, un modelo estadístico está compuesto de parámetros con los cuales es posible mapear un evento lingüístico a una probabilidad. Como un problema de aprendizaje supervisado, la tarea de NER puede ser modelada como una tarea

[4] http://developers.google.com/freebase
[5] https://nlp.stanford.edu/software/tagger.shtml
[6] Corpora es el plural de corpus.

de clasificación para cada palabra individual. Usualmente esta clasificación es realizada sobre una secuencia de palabras (sentencia/oración) en el que se predicen las etiquetas de todas las palabras de la secuencia.

En la literatura se pueden identificar tres grupos de enfoques basados en estadística: (1) aquellos que están basados en campos aleatorios condicionales, (2) basados en modelos ocultos de Markov y (3) basados en máxima entropía. A continuación se describen estos enfoques.

3.1.2.1. Basados en campos aleatorios condicionales

Un campo aleatorio condicional (CRF, por sus siglas en inglés) es un modelo estocástico que utiliza una estructura probabilística para el etiquetado y segmentado de datos secuenciales, el cual utiliza un enfoque condicional [11]. Un CRF puede entenderse como un modelo de grafo no dirigido, globalmente condicionado sobre una observación [34]. Esta observación, como se muestra en la Figura 3.4, corresponde a una variable aleatoria sobre la secuencia de datos a ser etiquetados, que en el caso del NER es una palabra dentro de una sentencia.

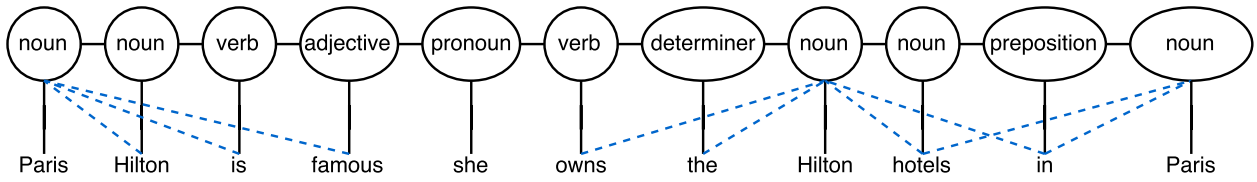


Figura 3.4: Representación gráfica de una secuencia de datos etiquetados con CRF.

Como propuestas de NER basadas en CRF podemos encontrar la de McCallum y Li [43], quienes propusieron un método de inducción de características, el cual se basa en el principio de seleccionar únicamente aquellas características que mejor desempeño tengan al reconocer entidades nombradas. También utilizan GoogleSets[7] para crear lexiciones. Esta propuesta infirió alrededor de 6423 características.

[7]Es una herramienta de Google que actualmente ya no se encuentra en servicio. Ésta, a partir de dos o más palabras, regresa un conjunto de palabras similares, tantas y como el sistema permita. Por ejemplo, de las palabras "mazda" y "honda", regresa "bmw", "ford", "toyota", "nissan", "audi", etc.

Por otro lado, Castillo y Gutierrez [47] propusieron un prototipo para NER en el idioma Español basado en CRF, el cual implementa segmentación por sentencias y palabras para aplicar herramientas de etiquetado gramatical sobre los conjuntos de entrenamiento y prueba, permitiéndoles así detectar entidades nombradas.

3.1.2.2. Basados en modelos ocultos de Markov

El modelo oculto de Markov (HMM, por sus siglas en inglés) puede ser considerado como un modelo probabilístico de una secuencia. Formalmente HMM es un proceso estocástico conformado por dos procesos. El primer proceso genera la secuencia de estados, que en el caso del NER son el tipo de clases a las que puede pertenecer una entidad nombrada (e.g. persona, organización, lugar, etc.). Esta secuencia de estados se genera siguiendo la siguiente suposición: el estado actual depende sólo del estado anterior (se conoce como suposición de Markov de primer orden) [67]. Un ejemplo de esta secuencia de estados se ilustra en la Figura 3.5. El segundo proceso genera la secuencia de observaciones de la secuencia de estados. Para ello se utiliza la decodificación de Viterbi [72], la cual encuentra la secuencia óptima de estados dada la secuencia de palabras y sus características. Típicamente, estas características son: primera letra de la palabra inicia en mayúscula, primera palabra de la sentencia, longitud de la palabra, etiqueta gramatical, etc.

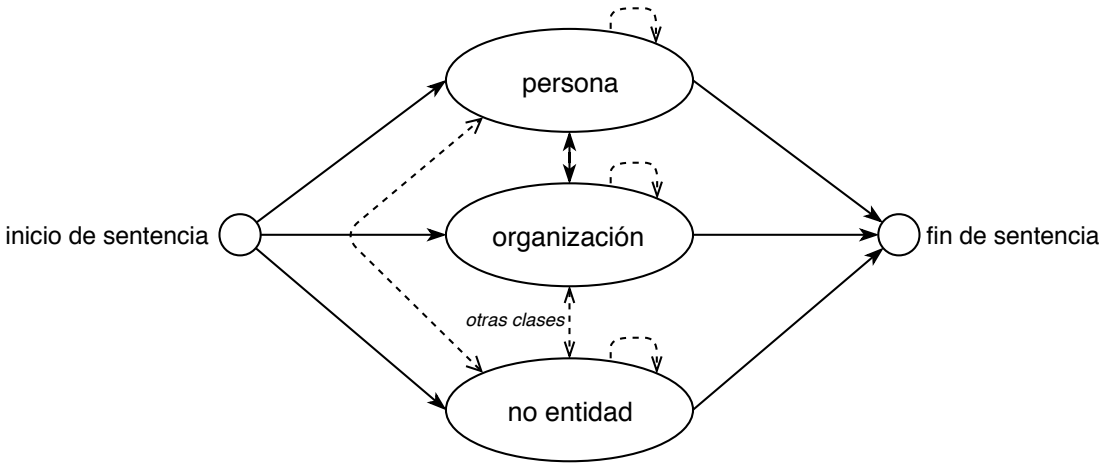


Figura 3.5: Representación gráfica de una secuencia de estados generada con HMM [4].

Nymble es una herramienta de NER basada en HMM propuesta por Bikei y Miller [4], la cual genera una máquina de estado finito de ocho estados, es decir, con ocho tipos de entidades nombradas: persona, organización, fecha, hora, porcentaje, cantidades de dinero y una última denominada "nada" para clasificar una observación como no entidad.

Por otro lado, Malouf [40] realizó una experimentación diseñando cinco modelos de HMM:

- Primero realiza un simple reconocedor basado en HMM, el cual no incluye las típicas características gramaticales, sino simplemente toma en cuenta la etiqueta de las palabras que le preceden y proceden.
- Como segundo se agregan características al primer modelo HMM descrito anteriormente, tales como: mayúsculas, primera palabra, longitud, mayúscula inicial, entre otros.
- El tercero integra el primer modelo en un enfoque de máxima entropía y lo enriquece agregándole las características del segundo modelo.
- Por último, el cuarto y quinto modelo, siguen preservando el enfoque de máxima entropía, el cual integra el primer modelo de HMM simple, pero con la diferencia de que estos dos son entrenados con más información y con más características. La información extra que utilizaron fueron bases de datos de entidades y gazetteers.

Ekbal y Bandyopadhyay [18] propusieron un método de NER basado en HMM para los idiomas bengalí e hindi. Para ello utilizan gazetteers de personas y lugares, etiquetado gramatical y características relacionadas con sufijos y prefijos de palabras. Los resultados fueron evaluados con validación cruzada de 10 iteraciones con el promedio de las métricas cobertura, precisión y medida-f.

3.1.2.3. Basados en máxima entropía

Cuando se trata de modelar un comportamiento aleatorio o un proceso estocástico a partir de un conjunto de datos se debe seleccionar un modelo que, sin hacer suposiciones, satisfaga todas las propiedades observables de la variable o del proceso. En el sentido de la teoría de la información, lo anterior resulta en un modelo de máxima entropía (ME, por sus siglas en inglés) [75].

ME es una técnica eficiente debido a que es capaz de integrar información de diferentes fuentes dentro de un único modelo. Esta información es tratada e integrada al modelo como características que, teniendo en cuenta que la tarea de NER se realiza a partir de texto plano escrito en lenguaje natural, suele ser información gramatical, morfológica y sintáctica. El objetivo de un modelo basado en ME es, a partir de estas características, definir un conjunto de funciones de probabilidad (*pmfs*[8]) y seleccionar de éstas aquellas que generen el modelo que tenga la máxima entropía[9].

Las ventajas de este modelo residen en su capacidad de combinar/integrar diferentes tipos de dependencias estadísticas en un solo modelo. Lo anterior se debe a que ME, por ejemplo, es capaz de combinar/utilizar HMM y CRF en un único modelo, obteniendo así las ventajas de estos dos. Por otro lado, debido a que es capaz de integrar mucha información proveniente de diferentes fuentes, requiere de un gran costo computacional para ser entrenado, lo que presenta una gran desventaja.

Como enfoques para el NER basado en ME, existe el propuesto por Curran y Clark [16], en el cual utilizaron un conjunto de características binarias, es decir, si ésta se cumple tendrá el valor de 1 y en caso contrario 0. Con lo anterior logran comprobar que el utilizar una gran variedad de características produce buenos resultados.

Por otro lado, Borthwick y Sterling [7] propusieron MENE, el cual utiliza características binarias léxicas, así como lexicones y gazetteers de 8 diferentes fuentes: nombres en general, nombres de empresas, nombres de empresas sin sufijos, lista de escuelas y universidades, listas de sufijos, fechas y tiempos, abreviaciones y regiones/lugares.

Gran parte de las propuestas basadas en ME han sido realizadas para el Inglés, sin embargo, se pueden encontrar muy pocas propuestas para otros idiomas, como la de Konkol y Konopík [33] para el idioma checo. Ellos, además de utilizar las características típicas mencionadas en las propuestas anteriores, agregan *semantic spaces*, el cual es un método para encontrar automáticamente relaciones entre palabras, lo que ayuda a reconocer entidades nombradas en contextos desconocidos.

[8]Función que da la probabilidad de que una variable aleatoria discreta sea exactamente igual a algún valor [62].

[9]Entiéndase a máxima entropía como la distribución de probabilidad menos sesgada, que es aquella en la que la desinformación es máxima, es decir, aquella que contenga menos información extrínseca al problema [57].

3.1.3 Benchmarking para el NER

El número de sistemas de IE ha crecido significativamente en los últimos años, especialmente en el área del NER [21]. Este mismo crecimiento ha estimulado el desarrollo de métodos de evaluaciones formales, así como la creación de conjuntos de datos para dichas evaluaciones [42], lo que ha dado lugar a los *puntos de referencia* para el NER. Un punto de referencia (en inglés benchmark) es una prueba de rendimiento que tiene como objetivo estimar el rendimiento de un sistema, método, componente, etc., así como proveer un método en el que el resultado de dicha prueba pueda compararse objetivamente/equitativamente con los resultados de otros métodos. En el NER existen benchmarks que proporcionan las herramientas y los conjuntos de datos para evaluar el desempeño de los métodos de NER, así como las métricas para poder comparar sus resultados. Dichos conjuntos de datos se componen de textos escritos en lenguaje natural, los cuales en su mayoría están etiquetados manualmente por lingüistas expertos.

Uno de los primeros benchmarks propuestos para el IE fue *Message Understanding Conferences* (MUC) que tuvo lugar desde 1987 hasta 1998. El objetivo de las conferencias MUC fue el de evaluar sistemas IE desarrollados por diferentes grupos de investigación en diferentes dominios. Por lo que se seleccionó un dominio diferente para cada edición de MUC, los cuales se describen a continuación.

- MUC-1 (1987): fue básicamente exploratorio.
- MUC-2 (1989): se usó el mismo dominio que para MUC-1.
- MUC-3 (1991): el dominio de los documentos fue cambiado a eventos de terrorismo.
- MUC-4 (1992): se utilizó la misma tarea que para MUC-3.
- MUC-5 (1993): en las conferencias anteriores, los sistemas IE de la competencia sólo se aplicaban para extraer información de documentos en inglés.
- MUC-6 (1995): se utilizó el dominio financiero.
- MUC-7 (1998): se propuso el dominio accidentes aéreos.

Este benchmark fue inicialmente desarrollado por la marina de EUA (*Naval Ocean Systems Center* de San Diego) y posteriormente paso a manos de *United States Advanced Research Projects Agency*. Por este tipo de patrocinadores, los dominios de los conjuntos de datos que el benchmark MUC provee han sido muy limitados, tales como encuentros militares, terrorismo en latinoamérica o empresas internacionales (finanzas y accidentes aéreos).

Dado que MUC fue unos de los pioneros en la evaluación de sistemas de IE, gran parte de las métricas que se utilizan actualmente para evaluar el desempeño de este tipo de sistemas fue definido durante las conferencias MUC. En estás se estableció como métricas de evaluación de sistemas IE la precisión y la cobertura [25]. Poco después, MUC agregó a su lista de métricas la medida-f.

Dada la necesidad de explorar otros dominios así como el de resolver nuevos retos y problemas, como la independencia del lenguaje, surgieron nuevos benchmarks, de entre los que destaca *Conference on Natural Language Learning* (CoNLL). CoNLL surgió de la necesidad de evaluar propuestas de NLP que estuvieran basadas en métodos de aprendizaje automático. ConLL en las ediciones 2002 y 2003, de entre las tareas de NLP que buscó evaluar, se agregó la tarea de NER. Específicamente en la edición 2002, se enfocó en reconocer entidades nombradas en el lenguaje holandés y español. En dicha edición y por cada lenguaje, se proporcionaron conjuntos de datos que se conforman por archivos de texto con sentencias etiquetadas manualmente por lingüistas expertos.

En este trabajo, el método de NER propuesto es evaluado con el benchmark CoNLL utilizando las métricas precisión, cobertura y medida-f. Dichas métricas, así como los conjuntos de datos que CoNLL en su edición 2002 proporcionó para el español, se describen a detalle en el Capítulo 5.

3.1.4 Comparativa de trabajos relacionados

Dado los trabajos relacionados previamente analizados, en la Tabla 3.1 se presenta una comparativa entre las diferentes características gramaticales, morfológicas y sintácticas que estos utilizan, así como los diferentes idiomas paras los cuales fueron propuestos, y las técnicas y métricas que utilizaron para su evaluación.

En la parte de las características podemos ver que, aún y cuando la mitad de las propuestas presentadas en esta tabla pertenecen a métodos basados en estadística, dependen de reglas gramaticales, sintácticas o morfológicas, lo que las hacen dependientes del lenguaje. En este trabajo se reduce el uso de éstas, o cualquier otras características con el fin de proponer un método de NER que reduzca, con respecto a las técnicas mostradas en la Tabla 3.1, la dependencia al lenguaje.

Por otro lado, como técnicas de benchmark se puede observar que CoNLL y MUC-7 son las más utilizadas. Dado que MUC-7 solo puede ser usada bajo una licencia de suscripción, en este trabajo se utiliza CoNLL para evaluar el desempeño del método propuesto, así como las métricas de precisión, cobertura y medida-f.

3.1.5 Resumen

En este capítulo se hizo un análisis y estudio de los trabajos de investigación que resuelven el problema de dos diferentes maneras: estadísticamente o con base a reglas propias del lenguaje. Las propuestas de este último enfoque incorporan características gramaticales y morfológicas las cuales suelen representarse en un árbol de decisión con el fin de generar una serie de reglas/condiciones que ocurren de forma sucesiva, y así lograr reconocer las entidades nombradas de un texto dado. Si bien estos enfoques utilizan herramientas y fuentes de información externas para mejorar su desempeño, tales como Stanford NER Tagger, Wikipedia y DBpedia, suelen presentar una dependencia del lenguaje debido a la utilización de reglas basados en estructuras propias del lenguaje.

Por otro lado, las propuestas basadas en un enfoque estadístico buscan resolver este problema de dependencia del lenguaje. Para ello generan modelos probabilísticos a partir de características observadas en ejemplos de textos con entidades nombradas (documentos etiquetados), los cuales son incorporados en modelos como CRF, HMM y máxima entropía. Si bien estos enfoques resuelven el problema probabilísticamente, siguen incorporando en sus modelos características basadas en estructuras gramaticales, sintácticas y morfológicas propias del lenguaje, tal y como se muestra en la Tabla 3.1.

De acuerdo a lo anterior, en este trabajo de tesis se plantea un método de NER basado en un modelo probabilístico el cual únicamente incorpore información sintáctica, dejando de lado la gramatical y morfológica, con el fin de reducir la dependencia del lenguaje, con respecto a las técnicas del estado del arte.

Tabla 3.1: Resumen de los métodos de NER propuestos en el estado del arte.

		Basados en Reglas						**Basados en Estadística**					
		Wikipedia		KB		AD		CRF		HMM		ME	
		[8]	[3]	[45]	[48]	[1]	[64]	[43]	[47]	[4]	[18]	[7]	[33]
características	lexicones	+		+	+			+			+		
	diccionarios	+	+				+		+			+	
	gazetteers				+		+		+		+	+	+
	prefijos					+			+		+		
	sufijos					+			+		+	+	
	n-gramas					+	+	+		+	+	+	+
	stopwords		+										
	lemmatization												+
	longitud					+	+		+			+	
	mayúsculas						+	+	+	+		+	+
	POST			+	+	+	+	+	+		+		+
idiomas	Inglés	+	+	+			+	+		+		+	
	Hindi										+		
	Hindú		+										
	Bengalí										+		
	Árabe					+							
	Húngaro						+						
	Checo												+
	Español				+				+	+			
benchm.	CoNLL				+		+	+	+				
	ANCORA					+			+				
	MUC-7				+					+	+	+	+
métricas	precisión		+		+	+	+	+	+	+	+	+	+
	cobertura		+		+	+	+	+	+	+	+	+	+
	medida-*F*		+		+	+	+	+	+	+	+	+	+

4

Solución propuesta basada en un método de aprendizaje probabilístico para el NER

En este capítulo se describe el desarrollo de la propuesta de solución al problema de NER. Primero se presenta una descripción general del método de aprendizaje probabilístico, seguido de una descripción detallada de las cuatro etapas por las que ésta se compone.

4.1 Descripción general de la propuesta

La propuesta de solución de este trabajo de tesis plantea un método capaz de solventar/mitigar las deficiencias de los métodos de NER descritos en el estado del arte, los cuales son: la escasez de corpus de entrenamiento dada la poca cantidad de datos disponibles, los grandes costos que implican los procesos de etiquetado manual de corpus por parte de lingüistas expertos, y la dependencia que presentan al utilizar información asociada a estructuras propias del lenguaje. Este método se compone de cuatro etapas, las cuales se ilustran en la Figura 4.1.

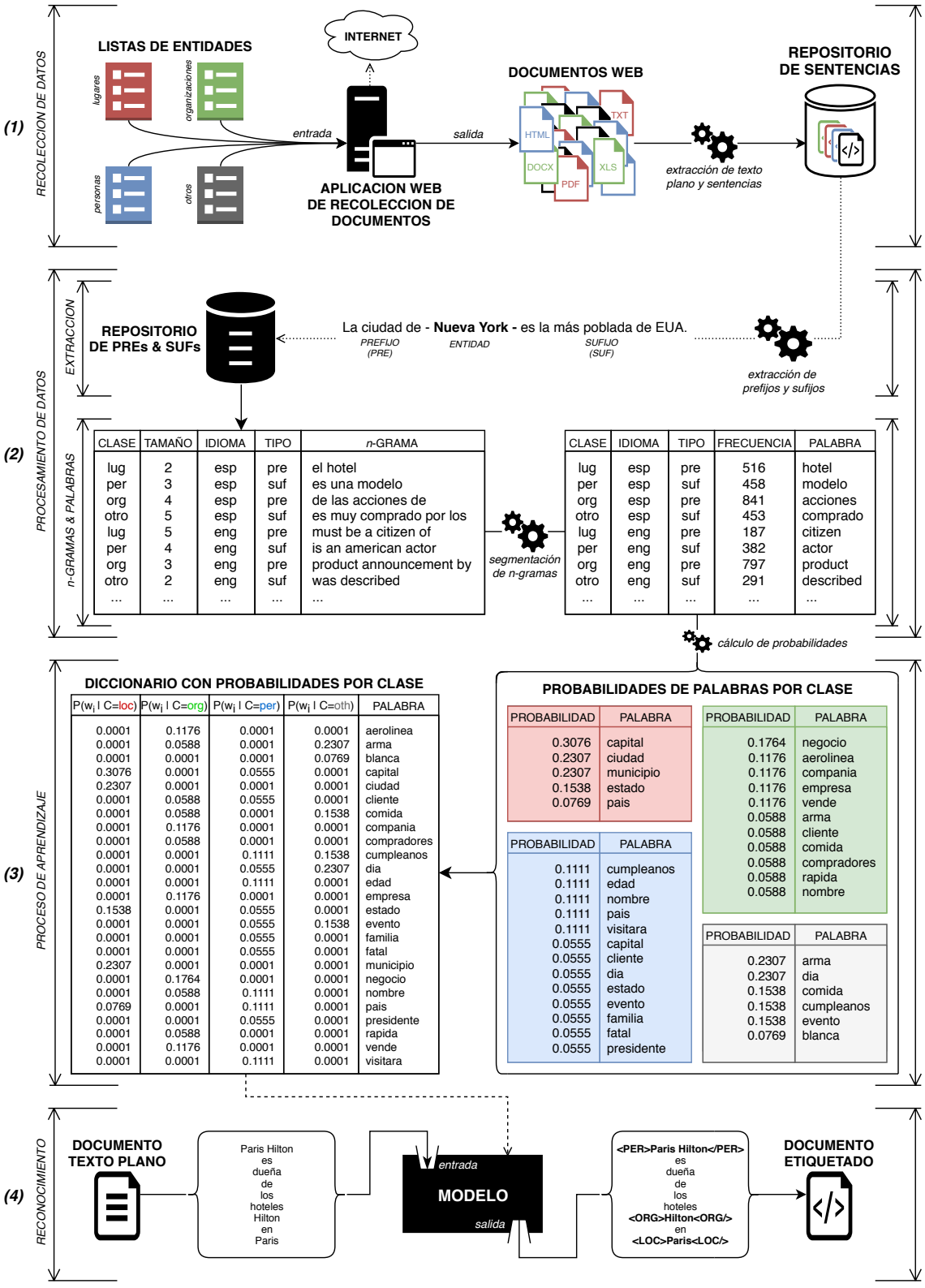


Figura 4.1: Propuesta de solución.

En la primera etapa Recolección de Datos se define un método de filtrado de búsqueda web que permite explorar de forma sistemática y focalizada diferentes recursos textuales de la Web. Ésta incluye una aplicación web que recibe como parámetro de entrada una lista de términos a ser buscados. Los términos de dicha lista pueden hacer referencia a diferentes tipos de entidades nombradas, tales como Microsoft, Nueva York y Paris Hilton que corresponden a una organización, lugar y persona respectivamente. A partir de estas entidades, la aplicación busca en la Web diferentes recursos textuales que contengan dichos términos. Tales recursos textuales que contienen entidades nombradas son descargados y procesados para extraer su contenido (texto plano). Dicho texto plano es segmentado en sentencias y almacenado en un repositorio.

En la segunda etapa Procesamiento de Datos el repositorio de sentencias con entidades nombradas es procesado con el fin de obtener conjuntos de entrenamiento y prueba. Para ello, de cada sentencia recolectada se extraen aquellas palabras circundantes que le anteceden y preceden a cada entidad nombrada. Esta información sintáctica es utilizada para determinar una frecuencia de aparición por palabra con el fin de obtener, desde la perspectiva estadística, aquellas palabras circundantes comunes a un tipo de entidad nombrada de cierto tipo. Este conjunto de palabras con su frecuencia de aparición conforma el conjunto de entrenamiento y de prueba.

En la tercera etapa Proceso de Aprendizaje se genera el modelo probabilístico de NER propuesto en este trabajo, el cual está basado en un clasificador Bayesiano simple. Este clasificador, a partir de la frecuencia de un conjunto de palabras circundantes, calcula valores de probabilidad. Este cálculo de valores de probabilidad corresponde al proceso de entrenamiento del modelo probabilístico.

En la cuarta etapa Reconocimiento de Entidades Nombradas, una vez entrenado el clasificador Bayesiano simple, el modelo reconoce probabilísticamente a partir de la información sintáctica/circundante de cada palabra las entidades nombradas que contiene un texto dado.

Las cuatro etapas de este método propuesto, las cuales son Recolección de Datos, Procesamiento de Datos, Proceso de Aprendizaje y Reconocimiento de Entidades Nombradas, se detallan en las siguientes secciones.

4.2 Recolección de datos

En el NER usualmente se emplean datos de entrenamiento que involucran procesos de etiquetado manual, los cuales implican un importante costo en tiempo, personas e infraestructura. Para evitar incurrir en estos costos, en este trabajo se propone un proceso automatizado de recolección de datos que permite obtener documentos de texto, a partir de la Web, que incluyen entidades nombradas.

Como se indica en la Figura 4.1, este proceso de recolección de datos incluye una aplicación web que permite realizar consultas al motor de búsqueda de Google. Una consulta está conformada por una cadena de texto que incluye los términos a ser buscados (entidades nombradas) y parámetros adicionales de búsqueda, tales como lenguaje y número de resultados. Toda búsqueda enviada a esta aplicación es atendida por el módulo **gestor de consultas** quien a su vez se encarga de obtener del motor de búsqueda de Google las ubicaciones de documentos (URLs, por sus siglas en inglés) que contienen cadenas de texto asociadas a los términos buscados. Cada URL es utilizada por el módulo **gestor de documentos** para obtener el documento asociado a dicha URL y almacenarlo en un repositorio denominado **documentos no estructurados**. Cada documento almacenado es procesado por el módulo **extractor de texto** encargado de transformar el contenido en texto plano y estructurarlo en sentencias. El funcionamiento anteriormente descrito se ilustra en la Figura 4.2.

Por ejemplo, aplicando el proceso anterior es posible realizar una consulta para el término "Nueva York", con el fin de obtener sentencias de la forma: "*La ciudad de* ***Nueva York*** *es la más poblaba de EUA*", "*El estado de* ***Nueva York*** *está muy urbanizado*", o "***Nueva York*** *es excepcionalmente diversa*". Por lo tanto, si garantizamos que la cadena de texto buscada hace referencia a una entidad nombrada, es posible obtener un conjunto de documentos que contengan sentencias que incluyan la entidad nombrada buscada.

Para obtener de manera sistemática un repositorio de sentencias que incluyen entidades, se creó una lista de entidades nombradas, la cual es dada como parámetro de entrada a la aplicación, tal y como se ilustra en la Figura 4.3.

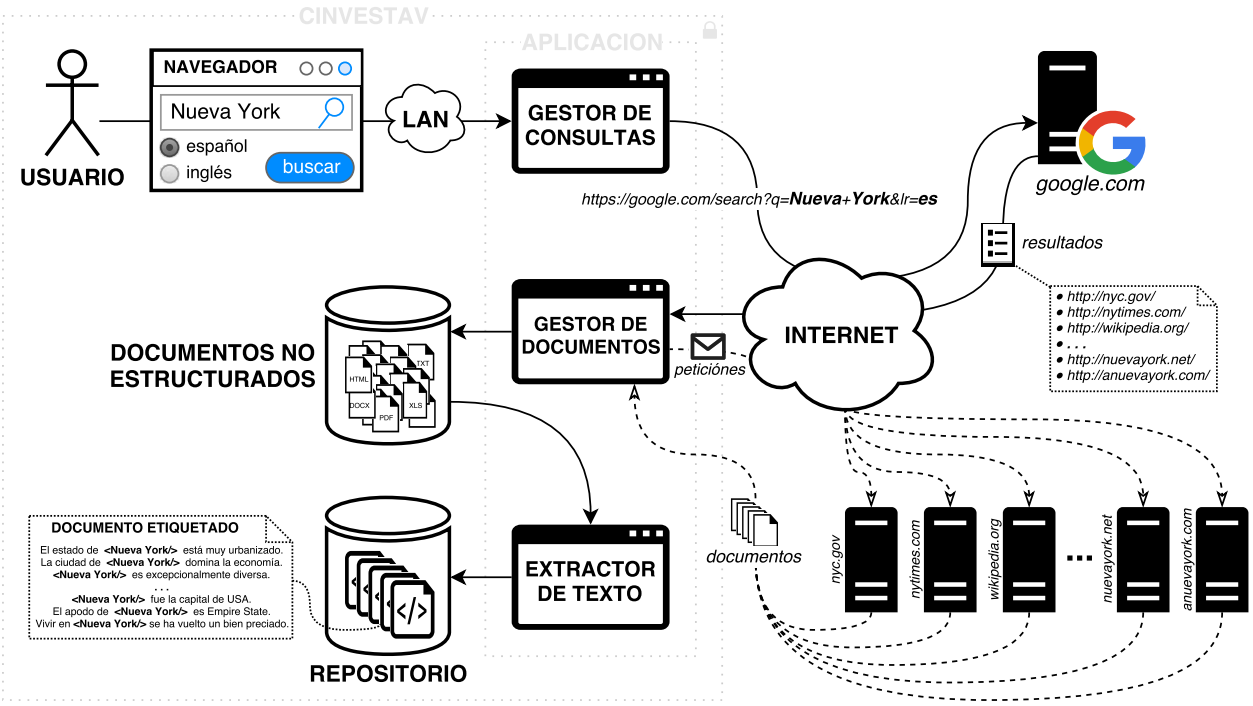


Figura 4.2: Funcionamiento general de la aplicación web de recolección de documentos web.

Para efectos prácticos, se ideó crear la lista de entidades nombradas a partir de información ya existente. Dicha lista se creó a partir de las entidades nombradas de DBpedia, la cual es una base de conocimiento que describe más de 3.77 millones de conceptos y entidades nombradas [36] clasificadas en más de 50 clases y 700 subclases.

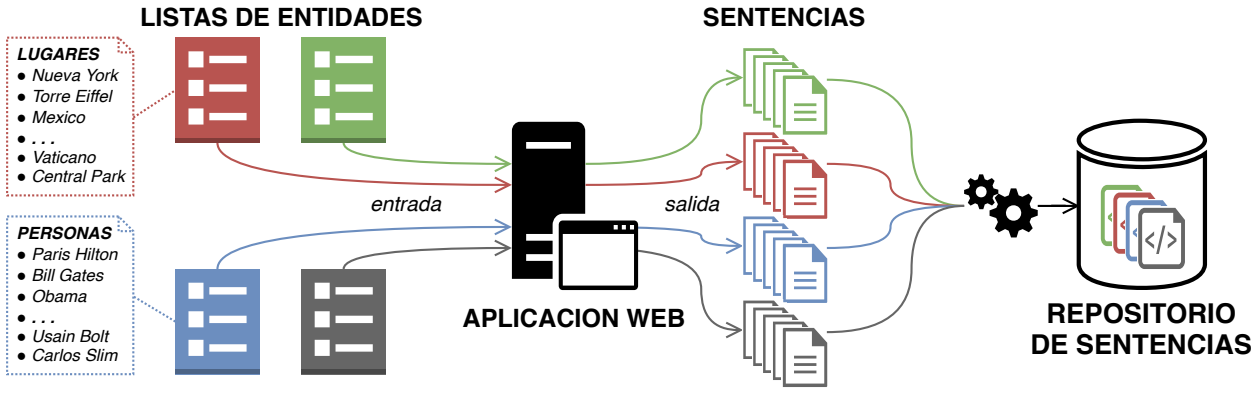


Figura 4.3: Proceso de obtención de textos etiquetados a partir de listas de entidades nombradas.

DBpedia permite extraer nombres de entidades asociadas a lugares, personas, organizaciones, entre otros tipos; así como agruparlas en subtemas más específicos, por ejemplo iglesias y parques, actores y cantantes, o aerolíneas y bandas de música. La lista creada incluye entidades de cuatro tipos: (1) *location*, (2) *person* y (3) *organisation*, así como una denominada (4) *others* que aglomera *events*, *foods*, *species*, *chemicals*, *holidays*, *activities* y *artefacts*.

El proceso utilizado en esta etapa de recolección de datos se describe en el Algoritmo 1. Éste recibe como parámetro de entrada una lista de entidades nombradas, el número de resultados deseados por entidad nombrada, idioma de los textos que se recolectarán y una ruta de un directorio local donde se desea que se almacenen los documentos que se obtendrán. El resultado será un conjunto de documentos que contienen sentencias en donde las entidades nombradas se identifican mediante etiquetas insertadas, por ejemplo: "El estado de *<location>* **Nueva York** *</location>* está muy urbanizado".

Algoritmo 1 Recolección de datos.

Entrada:
$entidades[]$: a ser buscadas;
$numero$: de resultados por entidad;
$lenguaje$: de los textos que se recolectaran;
$directorio$: local donde se almacenaran los documentos etiquetados;

```
1:  para todo entidad ∈ entidades hacer
2:      archivoLocal = crearNuevoArchivoDeTexto(directorio)
3:      urls[] = consultarGoogle(entidad, numero, lenguaje)
4:      para todo url ∈ urls hacer
5:          documento = descargarRecursoWeb(url)
6:          tipo = obtenerTipoDeDocumento(documento)
7:          texto = extraerTextoPlano(documento, tipo)
8:          sentencias[] = segmentarTextoEnSentencias(texto)
9:          para todo sentencia ∈ sentencias hacer
10:             sentenciaEtiquetada = etiquetarSentenia(sentencia, entidad)
11:             guardarSentencia(archivoLocal, sentenciaEtiquetada)
12:         fin para
13:     fin para
14: fin para
```

4.3 Procesamiento de datos

El proceso de recolección de datos anteriormente descrito permite obtener un conjunto de sentencias las cuales al menos incluyen una entidad nombrada. Debido a que dichas sentencias pueden contener información redundante o carente de valor, al menos desde el punto de vista de análisis de datos, se aplicaron algunas tareas de preprocesamiento, tales como eliminación de caracteres especiales, dígitos, signos diacríticos y normalización de mayúsculas a minúsculas.

Dado este conjunto de sentencias preprocesadas, y asumiendo que éstas incluyen una entidad, es posible realizar un análisis que permita inferir su orden estructural intrínseco con el fin de inferir a qué tipo de entidad nombrada pertenece. Dicho análisis requiere de las siguientes consideraciones:

- A partir de una sentencia que contiene una entidad nombrada, es posible extraer aquellas n palabras precedentes y subsecuentes que circundan dicha entidad (n-grama). Por ejemplo, en la siguiente sentencia se extrajeron n-gramas con $n = 5$:

$$\text{El número de} \mid \underbrace{\text{visitas a la ciudad de}}_{n\text{-grama precedente}} \mid \textbf{Nueva York} \mid \underbrace{\text{ha aumentado de forma sorprendente}}_{n\text{-grama subsecuente}} \mid$$

- Al extraer los n-gramas de todas las sentencias, tendremos, al menos desde la perspectiva estadística, aquellos n-gramas comunes a una entidad nombrada. Por ejemplo, n-gramas como "*la ciudad de*", "*el ciudadano de*", "*ha aumentado de*" o "*es muy visitada*", podrán tener una probabilidad alta en aquellas sentencias que contienen una entidad nombrada de tipo lugar.
- A partir de un conjunto de n-gramas precedentes y subsecuentes es posible realizar un análisis de los elementos que contienen estos n-gramas que permita inferir una estructura común para un tipo de entidad nombrada.
- Este análisis requiere generar un modelo probabilístico con el cual sea posible determinar probabilísticamente a qué tipo de entidad está asociado un n-grama dado.

Con base en lo anterior, a continuación se describen los siguiente aspectos: (1) extracción de n-gramas y (2) extracción de palabras.

4.3.1 Extracción de *n*-gramas

Dado un conjunto de sentencias preprocesadas que contienen entidades nombradas etiquetadas, se extraen por cada entidad nombrada de cada sentencia los n-gramas precedentes y subsecuentes de longitud n correspondientes a la entidad nombrada etiquetada. Un ejemplo de este proceso, con $n = 3$, se muestra en la Figura 4.4.

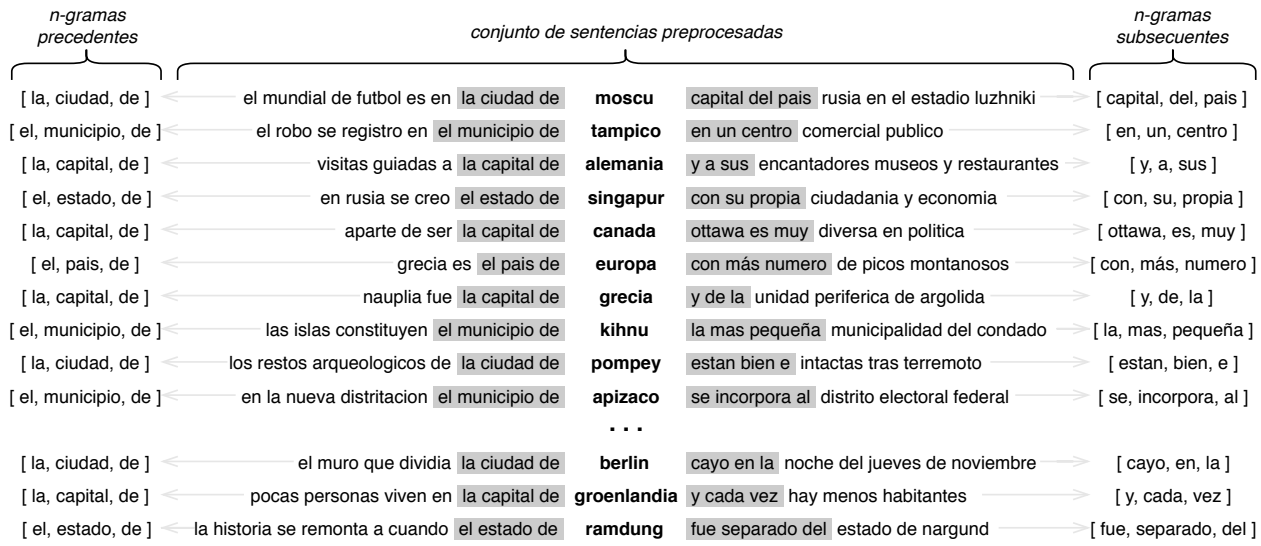


Figura 4.4: Extracción de n-gramas precedentes y subsecuentes de longitud $n = 3$.

Aplicando este proceso se extrajeron n-gramas de longitud 3 y 5. Este conjunto de n-gramas fue almacenado en un repositorio con información adicional respecto a su clase, longitud y tipo. Ejemplos de estos n-gramas almacenados se muestran en la Tabla 4.1.

Tabla 4.1: Ejemplos de n-gramas extraídos y almacenados.

CLASE	LONGITUD	TIPO	*n*-GRAMA	ENTIDAD NOMBRADA
lugar	3	precedente	en el hotel	Hilton
persona	3	subsecuente	es una modelo	Paris Hilton
organización	5	precedente	logros más destacados de la	ONU
otro	5	subsecuente	es una obra maestra literaria	Divina Comedia

Estas longitudes de n-grama, con $n = 3$ y $n = 5$, fueron seleccionados experimentalmente después de hacer pruebas con longitudes de n-gramas de 1, 2, 3, 4, 5, 6, 7 y 8. Las longitudes $n = 3$ y $n = 5$ fueron las que mejor resultado/desempeño presentaron en el proceso de detección y clasificación de entidades nombradas.

4.3.2 Extracción de palabras

De las palabras de los n-gramas es posible determinar la frecuencia de aparición que tiene la palabra en un conjunto de n-gramas dado, un ejemplo de este cálculo se ilustra en la Figura 4.5.

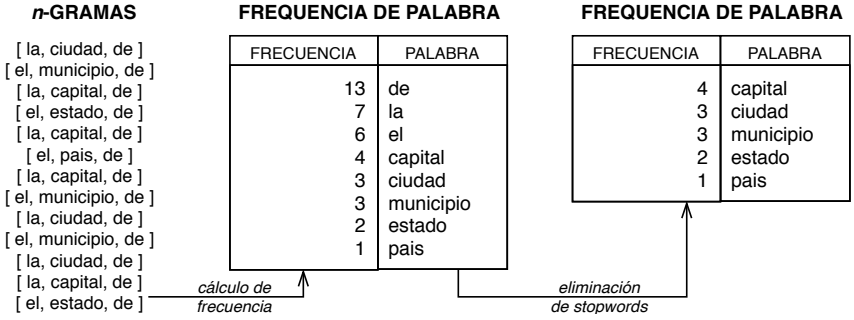


Figura 4.5: Cálculo de frecuencia de palabras.

En este cálculo de frecuencia no se incluyen *stopwords* ya que, de acuerdo al estado del arte [22, 58], éstas no aportan valor para discriminar cuando una entidad nombrada pertenece a cierto tipo. Este proceso se realizó para los conjuntos de n-gramas asociados a lugares, organizaciones, personas y otros. Un ejemplo de este proceso se ilustra en la Figura 4.6.

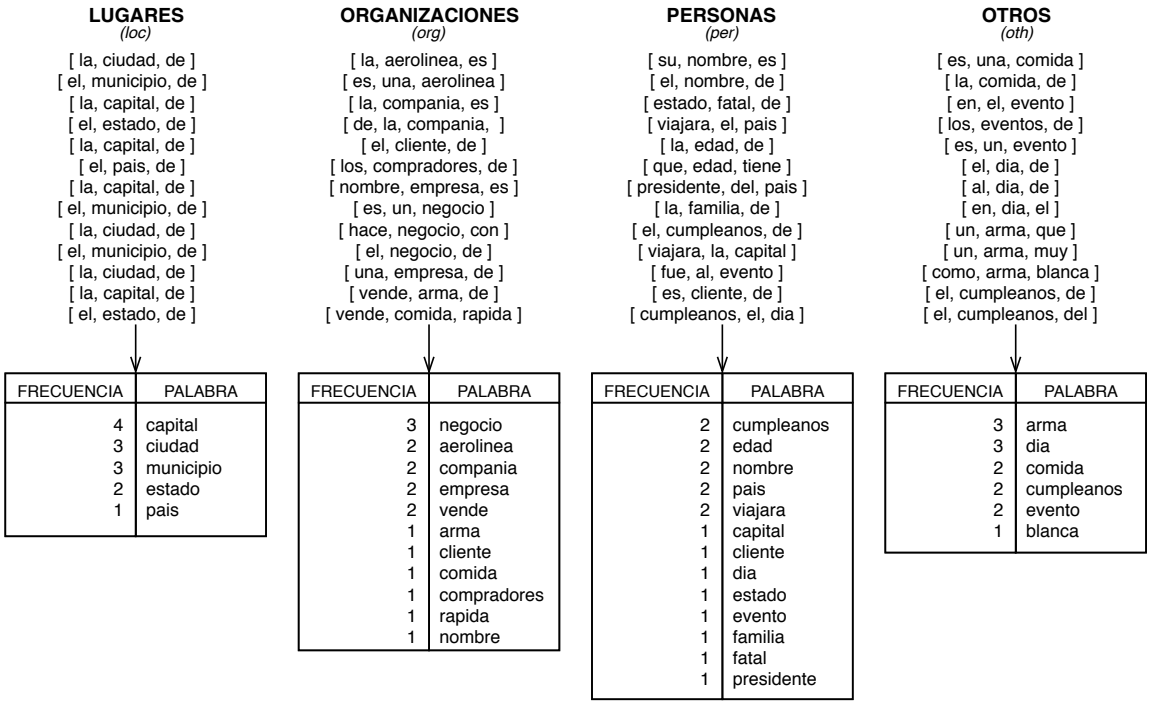


Figura 4.6: Cálculo de frecuencia de palabras por tipo.

Al calcular estas frecuencias de aparición se obtienen, desde la perspectiva estadística, aquellas palabras circundantes comunes a una entidad nombrada de tipo lugar, organización, persona y otro. Este conjunto de palabras con su frecuencia de aparición conforman el conjunto de entrenamiento del modelo probabilístico propuesto en este trabajo.

4.4 Proceso de aprendizaje

El proceso de aprendizaje propuesto en este trabajo se compone de un modelo probabilístico basado en un clasificador Bayesiano simple. Los clasificadores bayesianos son clasificadores basados en el teorema de Bayes. Éstos, dado un ejemplo x representado por n valores, devuelven la clase C_i más probable que describa x. Para ello se basan en el supuesto de que todas las variables predictoras x_n son condicionalmente independientes dada la clase. Esta suposición se denomina *independencia condicional de clase*. Por otro lado, los clasificadores bayesianos también han demostrado una gran precisión y velocidad cuando se aplican a grandes conjuntos de datos [2, 23, 27, 38].

El clasificador Bayesiano simple propuesto para clasificar un n-grama x de n palabras a partir de un conjunto de palabras frecuentes por clase, requiere calcular las siguientes probabilidades: la probabilidad de la clase i (Ecuación 4.1) y la probabilidad de cada palabra dado que se está observando la clase i (Ecuación 4.2).

$$P(C = i) \tag{4.1}$$

$$P(x_n|C = i) \tag{4.2}$$

La probabilidad de cada palabra dado que se está observando la clase i (Ecuación 4.2) corresponde a la probabilidad de que una palabra x_n aparezca en un n-grama asociado a la clase i. Siguiendo el ejemplo mostrado en la Figura 4.6, los resultados de calcular la probabilidad descrita en la Ecuación 4.2 se ilustran en la Figura 4.7.

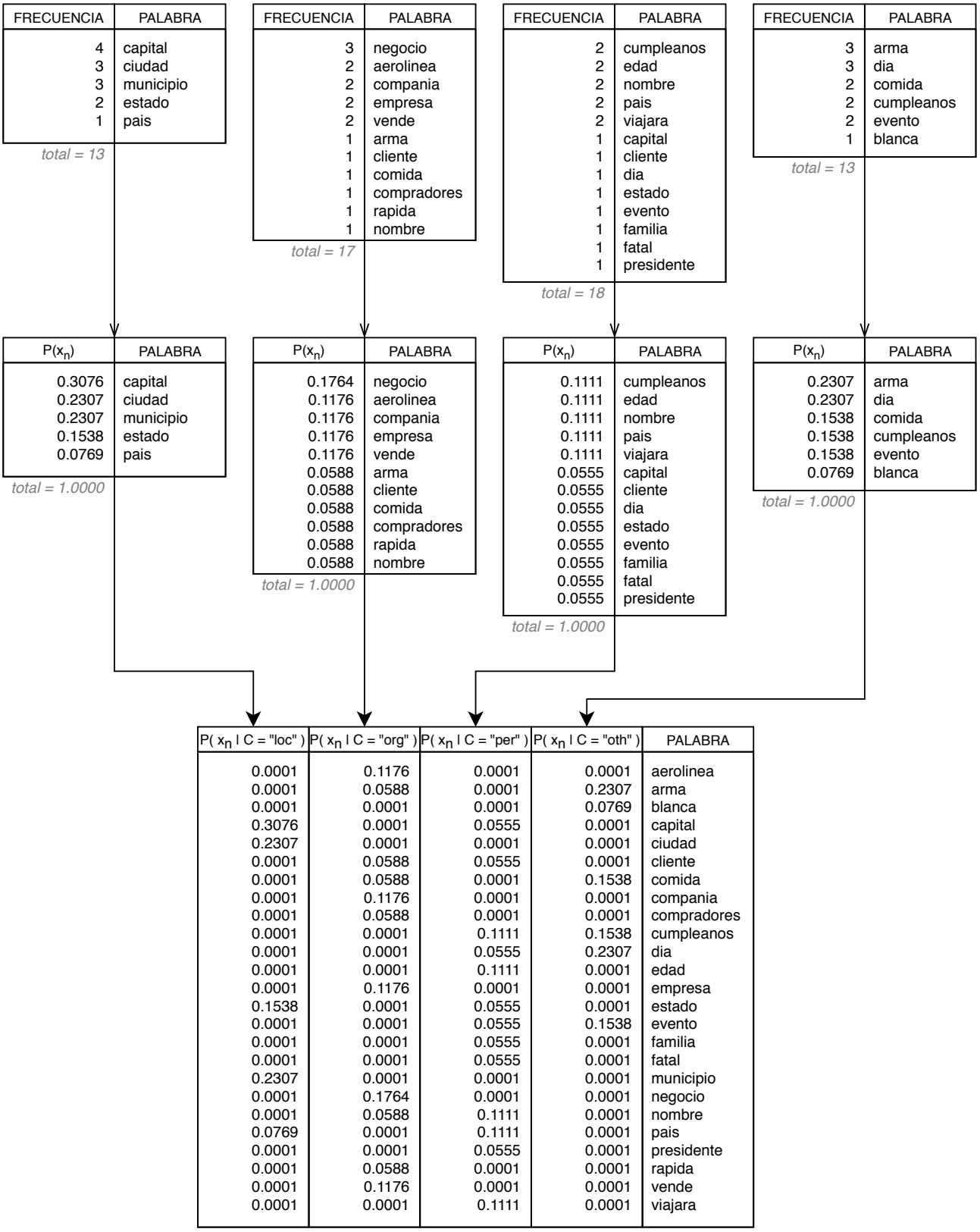


Figura 4.7: Cálculo de la probabilidad de cada palabra dado que se está observando la clase i.

En este cálculo existen casos en las que $P(x_n|C = i)$ es igual a 0. En estos casos dicho resultado es sustituido por un valor muy cercano a 0 (corrección de Laplace [41, 55]). En el ejemplo ilustrado en la Figura 4.7 este valor cercano a 0 es igual a 0.0001.

Para calcular la probabilidad de la clase i (Ecuación 4.1) se suma la frecuencia de todas las palabra x que pertenecen a la clase i y se divide entre la suma de la frecuencia de todas las palabras x de todas las clases, es decir:

$$P(C = i) = \frac{\sum_{x \in C_i} x}{\sum_{x \in C} x} \tag{4.3}$$

Siguiendo el ejemplo mostrado en la Figura 4.7, el proceso de calcular la probabilidad de la clase lugar, organización, persona y otro se ilustra en la Figura 4.8.

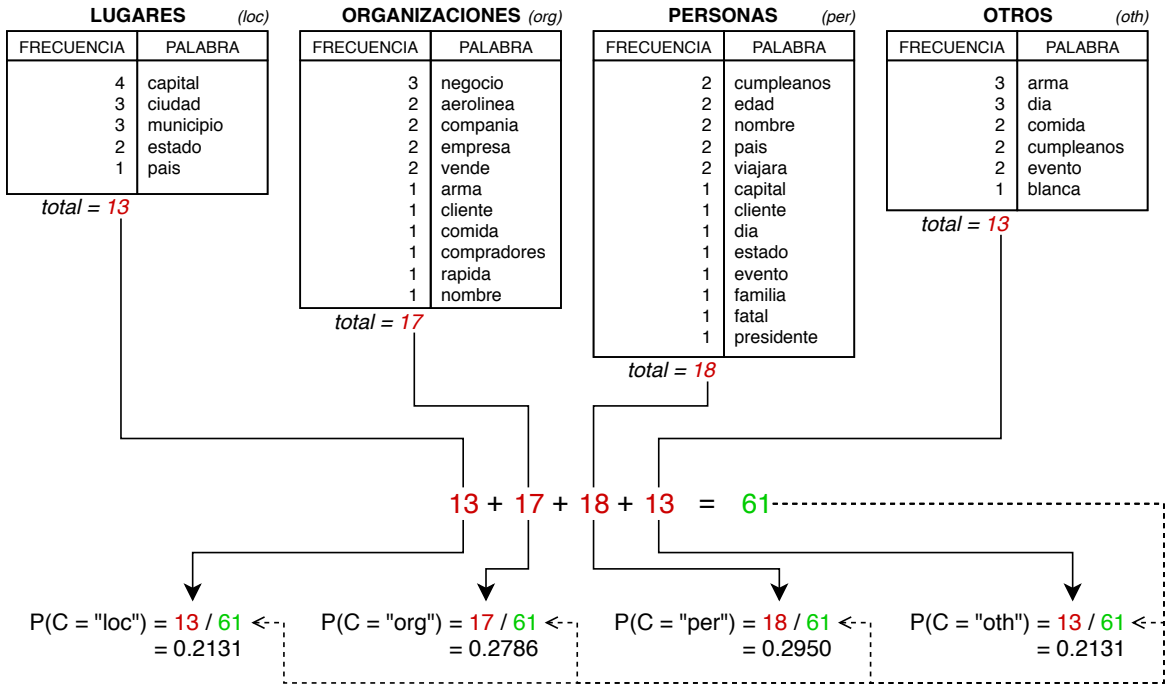


Figura 4.8: Cálculo de la probabilidad de la clase i.

Como resultado de calcular las probabilidades de las Ecuaciones 4.1 y 4.2 se obtiene la probabilidad de cada clase y un conjunto de probabilidades de palabras por clase. A partir de estos valores de probabilidad el modelo reconoce si una palabra es una entidad nombrada a partir de las palabras que contienen los n-gramas que le anteceden y preceden.

4.5 Reconocimiento de entidades nombradas

El cálculo de los valores de probabilidad obtenidos a partir de las Ecuaciones 4.1 y 4.2 corresponden al proceso de entrenamiento del modelo. Ahora, dada las palabras x_n que contiene un n-grama G, el modelo con base en el criterio mostrado en la Ecuación 4.4 determinará probabilísticamente si el n-grama está o no asociado a una entidad nombrada, así como el tipo al que pertenece.

$$\arg\max_{i=1,2,3,\ldots,n}(P(C=i)\prod_{x\in G}P(x|C=i)) \tag{4.4}$$

Por ejemplo, para reconocer las entidades nombradas de una sentencia dada, como "*El presidente Barack Obama viajará por la aerolínea WestJet hacia la Ciudad de México*", se extraen los n-gramas que le anteceden y preceden a cada palabras de dicha sentencia, tal y como se ilustra en la Tabla 4.2. Para fines ilustrativos, en este ejemplo la longitud de los n-gramas es $n = 2$, sin embargo este valor puede ser cualquier número natural que se desea configurar en el método, tal y como lo es en el caso de la experimentación donde se utiliza $n = 3$ y $n = 5$. Por otro lado, en dicha extracción también aquellas palabras que corresponden a stopwords son omitidas de este proceso.

Tabla 4.2: Extracción de n-gramas para el NER.

#	***n*-GRAMA PRECEDENTE**	**| PALABRA |**	***n*-GRAMA SUBSECUENTE**
1	-	**El**	-
2	*El*	**presidente**	*Barack Obama*
3	*El presidente*	**Barack**	*Obama viajará*
4	*presidente Barack*	**Obama**	*viajará por*
5	*Barack Obama*	**viajará**	*por la*
6	-	**por**	-
7	-	**la**	-
8	*por la*	**aerolínea**	*WestJet hacía*
9	*la aerolínea*	**WestJet**	*hacia la*
10	-	**hacia**	-
11	-	**la**	-
12	*hacia la*	**Ciudad**	*de México.*
13	-	**de**	-
14	*Ciudad de*	**México.**	-

Siguiendo el ejemplo mostrado en la Tabla 4.2 y después de normalizar el texto, para que el modelo determine si la palabra #4 "obama" con los n-gramas $[presidente, barack]$ y $[viajara, por]$ es una entidad o no, necesita calcular la Ecuación 4.4 a partir de los valores de probabilidad calculados con las Ecuaciones 4.1 y 4.2, y ejemplificados en las Figuras 4.8 y 4.7. Los resultados de dicha ecuación serán 4 probabilidades, las cuales corresponden a la probabilidad de que estos n-gramas pertenezcan a la clase *location* (loc), *person* (per), *organisation* (org) y *other* (oth). De dichos resultados, el modelo determinará que el argumento de máxima probabilidad fue $0.1818\text{x}10^2$, tal y como se ilustra en la Tabla 4.3. Por lo tanto, la palabra "obama" es etiquetada como persona ya que el argumento de máxima probabilidad corresponde a la marca de clase per.

Tabla 4.3: Cálculo del argumento de máxima probabilidad de la palabra "obama".

PALABRAS	$P(x_n\|C = loc)$	$P(x_n\|C = org)$	$P(x_n\|C = per)$	$P(x_n\|C = oth)$
1. presidente	0.0001	0.0001	0.0555	0.0001
2. barack	-	-	-	-
3. viajara	0.0001	0.0001	0.1111	0.0001
4. por	-	-	-	-
$P(C = i)$	0.2131	0.2787	0.2950	0.2131
$\arg\max$:	$0.2131\text{x}10^8$	$0.2787\text{x}10^8$	$0.1818\text{x}10^2$	$0.2131\text{x}10^8$

En este cálculo de probabilidades, como se puede ver en la Tabla 4.3, el modelo omite de la multiplicatoria aquellas palabras que corresponden a stopwords. En el ejemplo anterior la palabra "por" es omitida por ser stopword. Además, el modelo también omite aquellas palabras que no tienen una probabilidad calculada a priori, como es el caso de la palabra "barack" que es omitida ya que no aparece en los cálculos de la Figura 4.7.

Para el caso de la palabra #9 "westjet" con los n-gramas $[la, aerolinea]$ y $[hacia, la]$, el modelo determinará que el argumento de máxima probabilidad es $0.3277\text{x}10^1$, el cual corresponde a la clase org, tal y como se muestra en la Tabla 4.4.

Tabla 4.4: Cálculo del argumento de máxima probabilidad de la palabra "westjet".

PALABRAS	$P(x_n\|C = loc)$	$P(x_n\|C = org)$	$P(x_n\|C = per)$	$P(x_n\|C = oth)$
1. la	-	-	-	-
2. aerolinea	0.0001	0.1176	0.0001	0.0001
3. hacia	-	-	-	-
4. la	-	-	-	-
$P(C = i)$	0.2131	0.2787	0.2950	0.2131
arg máx:	0.2131x10^4	0.3277x10^1	0.2950x10^4	0.2131x10^4

Para el caso de la palabra #14 "mexico" con los n-gramas $[ciudad, de]$ y $[-]$, el modelo determinará que el argumento de máxima probabilidad es 0.4916x10^1, el cual corresponde a la clase loc, tal y como se muestra en la Tabla 4.5.

Tabla 4.5: Cálculo del argumento de máxima probabilidad de la palabra "mexico".

PALABRAS	$P(x_n\|C = loc)$	$P(x_n\|C = org)$	$P(x_n\|C = per)$	$P(x_n\|C = oth)$
1. ciudad	0.2307	0.0001	0.0001	0.0001
2. de	-	-	-	-
$P(C = i)$	0.2131	0.2787	0.2950	0.2131
arg máx:	0.4916x10^1	0.2787x10^4	0.2950x10^4	0.2131x10^4

En los casos en los que el argumento de máxima probabilidad corresponde a la clase oth, es decir, cuando la palabra no ha sido etiquetada como lugar, persona y organización, el modelo ha determinado que dicha palabra no es una entidad nombrada. Por lo tanto, toda aquella palabra que sea etiquetada como loc, per o org, el modelo ha determinado que es una entidad nombrada.

Este modelo es utilizado para reconocer las entidades nombradas de un documento de texto dado. Para ello, el texto de dicho documento es segmentado en sentencias. De las sentencias obtenidas se extraen los n-gramas precedente y subsecuentes de cada palabra. A partir de dichos n-gramas el modelo reconocerá las entidades nombradas del documento. Como resultado, el modelo regresará

un documento indicando mediante etiquetas las entidades nombradas que contiene. El proceso que el modelo realiza para reconocer las entidades nombradas de un texto dado, en donde la longitud de los n-gramas es $n = 2$, se ilustra en la Figura 4.9.

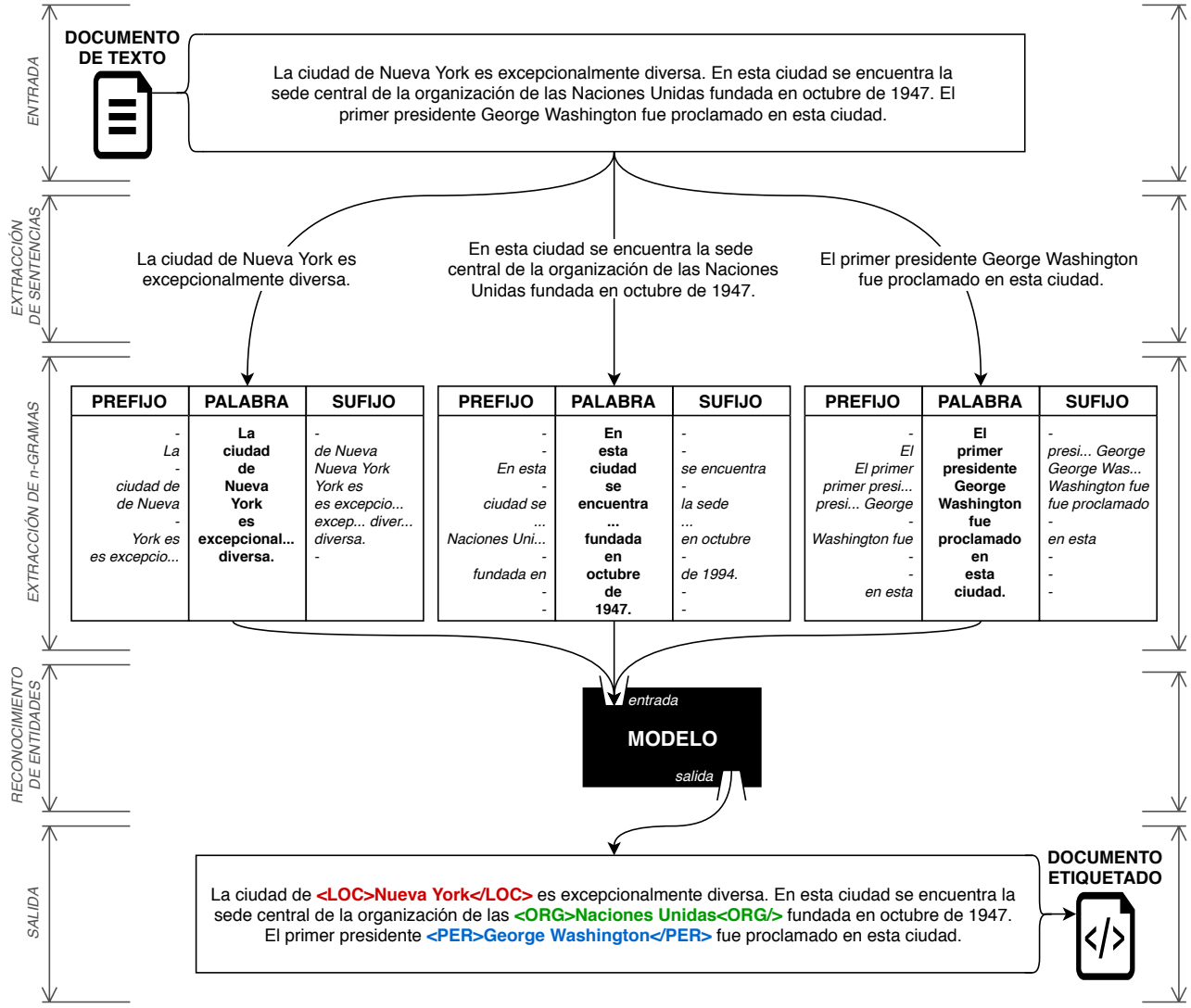


Figura 4.9: Ejemplo de reconocimiento de entidades nombradas dado un texto de entrada.

El proceso utilizado en esta etapa de reconocimiento de entidades nombradas se describe en el Algoritmo 2. Éste recibe como parámetro de entrada un texto a ser etiquetado, la longitud de los n-gramas que se utilizarán para el reconocimiento y una ruta del directorio local donde se encuentran

almacenados los documentos obtenidos en la fase de recolección. El resultado será un texto de salida el cual contiene etiquetadas aquellas entidades nombradas que el modelo reconoció.

Algoritmo 2 Reconocimiento de entidades nombradas en un texto dado.

Entrada:

$texto$: a ser anotado;
$longitud$: de los *n*-gramas a extraer;
$directorio$: local donde se encuentran almacenados los documentos etiquetados;

Salida:

$textoAnotado$: con las entidades nombradas reconocidas;

```
1: para todo documento ∈ directorio hacer
2:     entidad = extraerEntidadNombradaAsociadaAlDocumento(documento)
3:     sentenciasEtiquetadas[] = leerSentenciasDesdeDocumento(documento)
4:     para todo sentenciaEtiquetada ∈ sentenciasEtiquetadas hacer
5:         ngramas[] = extraerNgrama(sentenciaEtiquetada, entidad, longitud)
6:         agregarNgramasAlDataset(dataset, ngramas)
7:     fin para
8: fin para
9: palabras[] = segmentarNgramasDelDatasetEnPalabras(dataset)
10: frecuenciaDePalabras[] = calcularFrecuenciaDePalabrasPorClase(palabras)
11: probDeClase[] = calcularProbabilidadDeClase(frecuenciaDePalabras)
12: probDePalabras[] = calcularProbabilidadDePalabrasPorClase(frecuenciaDePalabras)
13: sentencias[] = segmentarTextoEnSentencias(texto)
14: para i = 0 hasta tamaño(sentencias) hacer
15:     sentencia = sentencias[i]
16:     tokens[] = segmentarSentenciaEnTokens(texto)
17:     para j = 0 hasta tamaño(tokens) hacer
18:         token = tokens[j]
19:         ngrama = extraerNgramaDelToken(token, tokens, longitud)
20:         marcaDeClase = clasificar(ngrama, probDePalabras, probDeClase)
21:         tokensClasificados[j] = marcaDeClase
22:     fin para
23:     sentenciasClasificadas[i] = marcarSentencia(tokensClasificados, sentencia)
24: fin para
25: textoAnotado = unirSentencias(sentenciasClasificadas)
26: devolver textoAnotado
```

5

Experimentación y resultados

En este capítulo se describen los experimentos realizados para evaluar el desempeño del método propuesto, así como los diferentes conjuntos de datos utilizados. Asimismo, se describen los escenarios probados, así como los resultados obtenidos en cada uno ellos. Este capítulo se divide en cuatro secciones: infraestructura utilizada, metodología de evaluación, conjuntos de datos y resultados.

5.1 Infraestructura utilizada

La infraestructura utilizada para la experimentación, consta de dos equipos. El primero equipo corresponde a un servidor instalado en la Unidad Cinvestav Tamaulipas donde se encuentra la aplicación web de recolección de documentos web, así como los conjuntos de datos recolectados. El segundo equipo consta de una computadora de trabajo proporcionada por la unidad para el desarrollo del proyecto. Las características de dichos equipos se describen en la Tabla 5.1.

Tabla 5.1: Descripción del equipo utilizado en la experimentación.

EQUIPO	SISTEMA OPERATIVO	PROCESADORES	CORES	RAM	DISCO DURO
1. Servidor	Ubuntu 14.04.5 LTS	4 Intel Xeon 2.10GHz	48	128GB	1500GB
2. Equipo de trabajo	Mac OS 10.13.6	1 Intel Core i5 2.5 GHz	4	4GB	500GB

5.2 Metodología de evaluación

Para evaluar el desempeño del método se diseñaron dos escenarios de pruebas. En el primer escenario el método es entrenado y probado con conjuntos de documentos obtenidos a partir de textos en español recolectados con la herramienta de filtrado y búsqueda web. En un segundo escenario el reconocedor de entidades nombradas es evaluado con un benchmark utilizado en el estado del arte para evaluar el desempeño de técnicas de NER. El benchmark utilizado en este escenario/evaluación es CoNLL 2002. Los resultados de ambos escenarios son evaluados con las mismas métricas de desempeño: *precision*, *recall* y *f-measure*. La metodología de evaluación descrita anteriormente se ilustra en la Figura 5.1.

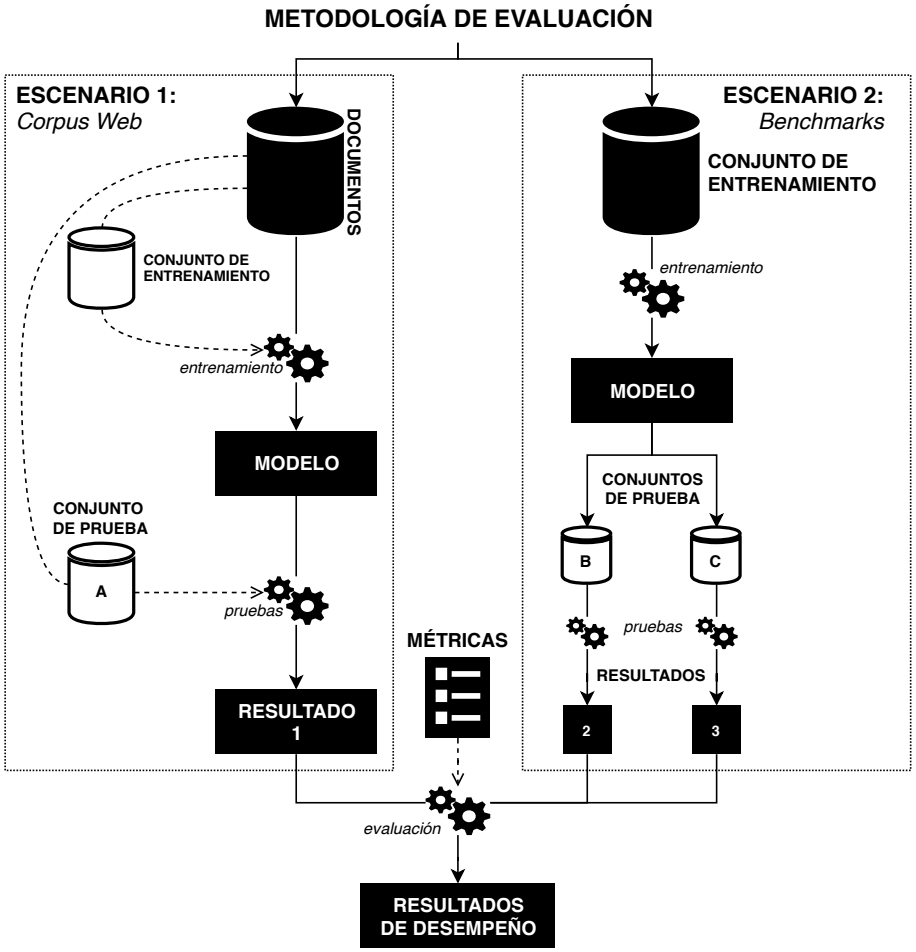


Figura 5.1: Metodología de evaluación.

5.3 Conjuntos de documentos

En esta experimentación realizada se utilizaron dos tipos de conjuntos de documentos. El primero se consiguió vía web siguiendo el proceso automatizado de recolección de datos de la propuesta de solución descrita en el capítulo anterior. Por otro lado, el segundo conjunto de datos se obtuvo de técnicas de benchmarking de NER. Estos conjuntos de documentos se describen a continuación.

5.3.1 Corpus Web

Este conjunto de documentos se obtuvo siguiendo el proceso automatizado de recolección de datos. Dicho proceso, dada una lista de entidades nombradas, obtiene un conjunto de sentencias, las cuales incluyen una entidad nombrada. La lista de entidades nombradas que se utilizó para obtener estas sentencias se creó a partir de las entidades nombradas que contiene DBpedia. Para extraer entidades nombradas de DBpedia es necesario utilizar el lenguaje SPARQL (por su acrónimo recursivo *SPARQL Protocol and RDF Query Language*), el cual permite recuperar recursos de la web semántica representados en el lenguaje RDF. Por ejemplo, para recuperar de DBpedia aquellos recursos que hacen referencia a entidades nombradas en español asociadas a la subclase país ($Country$), se tiene que ejecutar en el *endpoint*[1] de DBpedia[2] la consulta escrita en el Código 5.1.

Código 5.1: Consulta SPARQL para recuperar de DBpedia recursos en español de la clase $Country$.

```
1 PREFIX dbo:  <http://dbpedia.org/ontology/>
2 PREFIX dbr:  <http://dbpedia.org/resource/>
3 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4 SELECT *
5 WHERE {
6       ?resource a dbo:Country ;
7             rdfs:label ?label
8       . FILTER (lang(?label) = 'es')
9 }
```

[1] Servicio que acepta consultas SPARQL.
[2] https://dbpedia.org/sparql

Al ejecutar la consulta escrita en el Código 5.1 se obtendrán resultados parecidos a los mostrados en la Tabla 5.2.

Tabla 5.2: Resultados de consultar en DBpedia los recursos en español de la clase Country.

?resource	?label
dbpedia.org/Syria	"Siria"@es
dbpedia.org/Egypt	"Egipto"@es
dbpedia.org/New_Zealand	"Nueva Zelanda"@es
dbpedia.org/Afghanistan	"Afganistán"@es
dbpedia.org/Japan	"Japón"@es

Siguiendo el ejemplo descrito anteriormente, para crear la lista de entidades nombradas se recuperaron de DBpedia los recursos asociados a la clase lugar ($Place$), persona ($Person$), organización ($Organisation$), y "otros" ($ChemicalSubstance$, $Device$, $Event$, $Food$, $Holyday$, $Language$, $MeanOfTransportation$, $Species$, $Work$, $Activity$). El número de entidades nombradas recuperadas por clase fueron:

- lugar: 22,470
- persona: 16,362
- organización: 7,780
- "otros": 34,776

Esta lista de más de 80 mil entidades nombradas se le proporcionó como parámetro de entrada a la aplicación web de recolección de datos. Como resultado se obtuvo un repositorio con más de 970 mil de sentencias escritas en español. El tiempo que se requirió para obtener estas sentencias fue de casi 3 meses. Este largo tiempo se debió a que existe un número de consultas permitidas en un lapso de tiempo determinado antes de que el motor de búsqueda de Google detecte las peticiones de la aplicación como un trafico inusual (ataque DDoS).

Las sentencias de este repositorio fueron preprocesadas eliminando caracteres especiales, dígitos, signos diacríticos y cambio de mayúsculas a minúsculas. A partir de las entidades nombradas de estas sentencias preprocesadas se extrajeron n-gramas precedentes y subsecuentes de longitud 3 y 5. Dichos n-gramas se almacenaron en una base de datos MySQL.

Este repositorio de n-gramas precedentes y subsecuentes de longitud 3 y 5 conforman el corpus web. En la Tabla 5.3 se muestra el resumen de este conjunto de n-gramas.

Tabla 5.3: Resumen del corpus web.

CLASE	ENTIDADES	SENTENCIAS	*n*-GRAMAS				Tamaño en disco (mb)
			precedente		*subsecuente*		
			3	5	3	5	
LOC	22,470	251,488	663,732	536,310	701,778	596,851	438.6
PER	16,362	127,801	127,084	112,831	145,904	136,718	306.5
ORG	7,780	114,703	110,348	95,193	118,433	109,861	173.7
OTH	34,776	478,957	349,575	297,834	367,902	335,419	655.3
total	*81,388*	*972,949*	*1,250,739*	*1,042,168*	*1,334,017*	*1,178,849*	*1,574.1*

A partir del corpus web se utilizó validación cruzada para dividir el conjunto de datos en k subconjuntos. Uno de estos k subconjuntos es utilizado para probar el método, mientras los subconjuntos restantes son utilizados para entrenar el modelo. Por lo tanto, el modelo es entrenado y probado k veces. Los resaltados de estas k evaluaciones se les calcula la media para obtener un resultado general.

En este escenario de prueba la validación cruzada fue con $k = 10$, por lo tanto, se obtuvieron 10 diferentes conjuntos de entrenamiento y prueba, así como 10 resultados. Estos resultados se les calculó la media aritmética, tal y como se ilustra en la Figura 5.2.

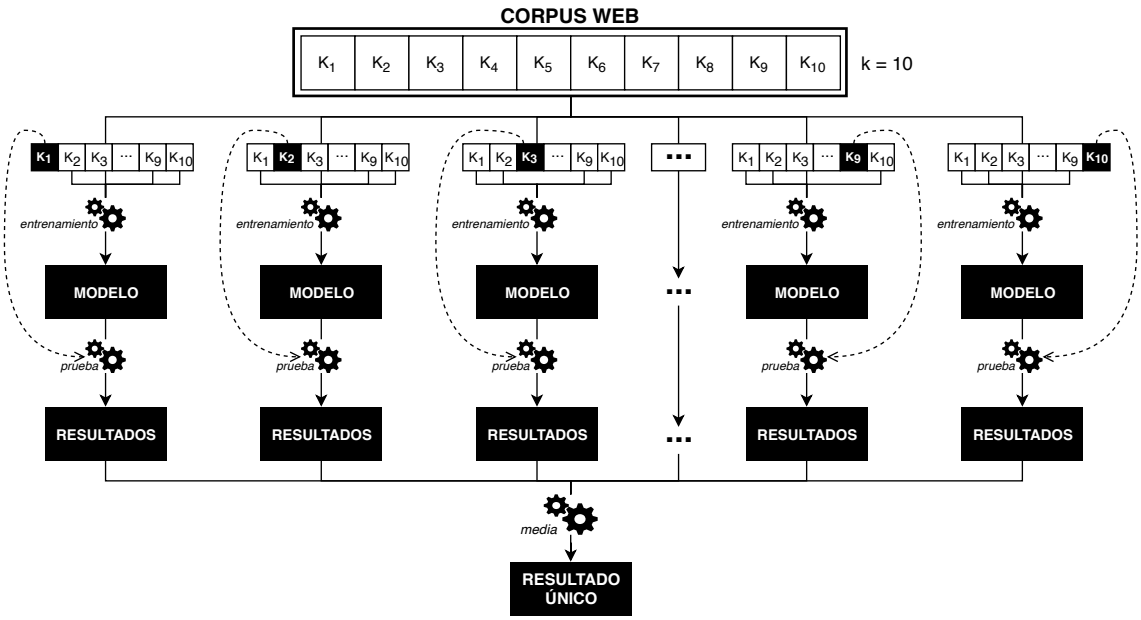


Figura 5.2: Evaluación del método propuesto con validación cruzada.

5.3.2 Benchmark

En NER, un benchmark es una prueba de rendimiento que proporciona las herramientas y los conjuntos de datos necesarios para evaluar el desempeño de un método de NER, así como las métricas para compararlos con otros. Con la finalidad de comparar con otros métodos el rendimiento del método de NER propuesto se utilizó el benchmark CoNLL. Los archivos de entrenamiento y prueba que CoNLL proporciona consisten de tres columnas separadas por un solo espacio. El primer elemento en cada línea corresponde a una palabra de una sentencia, el segundo elemento corresponde a la etiqueta gramatical de dicha palabra, y el tercero es la marca de clase de la palabra. Esta última columna correspondiente a la marca de clase presenta el formato $-TYPE, donde $ puede ser una "B" que denota el inicio de una entidad nombrada o una "I" para cualquier palabra no inicial que sea parte de una entidad nombrada, TYPE denota el tipo de entidad nombrada a la cual está asociada la palabra, tal como persona (PER), organización (ORG), lugar (LOC) y entidades diversas (MISC). Una palabra con la marca de clase "O" no es parte de una entidad nombrada, pero si de la sentencia. En la Tabla 5.4 se muestra un ejemplo de este conjunto de datos.

Tabla 5.4: Ejemplos de los conjuntos de datos etiquetados de CoNLL.

#	palabra	post	clase
1	El	DA	O
2	Museo	NC	B-ORG
3	de	SP	I-ORG
4	Arte	NC	I-ORG
5	en	SP	O
6	vidrio	NC	O
7	de	SP	O
8	Alcorcón	NC	B-LOC
9	acoge	VMI	O
10	esculturas	NC	O
11	del	SP	O
12	artista	NC	O
13	rumano	AQ	O
14	Edward	NC	B-PER
15	Leibovtiz	AQ	I-PER
16	.	.	O

Los conjuntos de datos de CoNLL se conforman de tres archivos. Uno de estos tres archivos es proporcionado exclusivamente para el entrenamiento y los otros dos restantes para pruebas individuales. El resumen de los datos de entrenamiento que CoNLL 2002 proporciona para el español se muestran en la Tabla 5.5, así como los de prueba A y B.

Tabla 5.5: Resumen de los conjuntos de datos de CoNLL.

CONJUNTO	ENTIDADES				SENTENCIAS	PALABRAS
	LOC	ORG	PER	MISC		
Entrenamiento	4,913	7,390	4,321	2,173	7,729	263,133
Prueba A	984	1,700	1,222	445	1,634	52,642
Prueba B	1,084	1,400	735	339	1,355	51,371

5.4 Métricas

En NER se han propuesto diferentes métricas para medir el desempeño de un método. De acuerdo al estado del arte, se determinó que más de 80 % de las propuestas, así como el benchmark CoNLL [66], utilizan las siguientes métricas:

- **Cobertura**, en inglés *recall*, es el número de entidades nombradas del conjunto de prueba que fueron reconocidas correctamente por el sistema.
- **Precisión**, en inglés *precision*, es el número de entidades nombradas encontradas por el sistema que son correctas.
- **Medida-F**, en inglés *F-measure*, es la media armónica de los dos valores anteriores.

Estas métricas se calculan en términos de *verdaderos positivos* (VP), *verdaderos negativos* (VN), *falsos positivos* (FP) y *falsos negativos* (FN). El término *positivo* y *negativo* corresponden a la marca de clase que se espera obtener, y el término *verdadero* y *falso* corresponden al resultado obtenido por el sistema. Un ejemplo de esto se ilustra en la Figura 5.3.

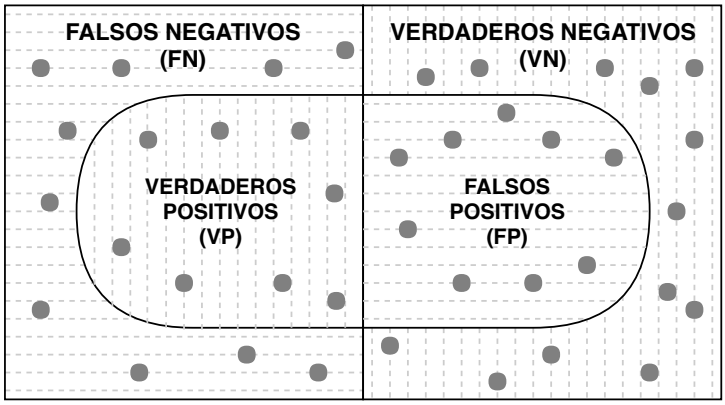


Figura 5.3: Ejemplos de elementos reconocidos en términos de VP, VN, FP y FN.

Para el problema de dos clases una manera de estructurar estos términos es a través de una matriz de confusión, tal y como se muestra en la Tabla 5.6. A partir de estos términos, las métricas de cobertura, precisión y medida-f se definen en la Ecuación 5.1, 5.2, y 5.3 respectivamente.

Tabla 5.6: Matriz de confusión para 2 clases.

	POSITIVO	NEGATIVO
VERDADERO	VP	VN
FALSO	FP	FN

$$C = \frac{VP}{VP + FP} \qquad (5.1) \qquad P = \frac{VP}{VP + FN} \qquad (5.2) \qquad MF = 2\frac{C * P}{C + P} \qquad (5.3)$$

Sin embargo, dado que en esta experimentación se utilizan cuatro tipos de entidades nombradas (loc, per, org y oth), es necesario generalizar la matriz de confusión. Para ello, se genera una tabla con m clases y k grupos, es decir, m filas y k columnas; tal y como se muestra en la Tabla 5.7, donde n_{ij} representa el número de elementos del grupo j que se clasificaron en la clase i. Con base en lo anterior, para los casos de más de dos clases las métricas de cobertura, precisión y medida-F se definen en la Ecuación 5.4, 5.5 y 5.6, respectivamente.

Tabla 5.7: Generalización de una matriz de confusión para problemas multiclase.

		GRUPOS				
		1	...	j	...	k
CLASES	1	n_1		n_{1j}		n_{1k}
	⋮		...		...	
	i	n_{i1}		n_{ij}		n_{ik}
	⋮		...		...	
	m	n_{m1}		n_{mj}		n_{mk}

$$C(i,j) = \frac{n_{ij}}{\sum_{i=1}^{m} n_{ij}} \quad (5.4) \qquad P(i,j) = \frac{n_{ij}}{\sum_{j=1}^{k} n_{ij}} \quad (5.5) \qquad F(i,j) = 2\frac{C(i,j) * R(i,j)}{C(i,j) + R(i,j)} \quad (5.6)$$

5.5 Resultados

Una vez ejecutada la evaluación en los escenarios descritos en la sección anterior, el resumen de los resultados de dichos experimentos se muestran en la Tabla 5.8. En ésta, las columnas *Escenario* y *Conjunto de Prueba* corresponden a los escenarios de evaluación y a los datos de prueba descritos anteriormente en la metodología de evaluación (Figura 5.1). La columna *n-grama* corresponde a la longitud del n-grama precedente y subsecuente que se extrajo por palabra en la fase de reconocimiento de entidades nombradas. Por último, las columnas *Cobertura*, *Precisión* y *Medida-F* corresponden a los resultados de las pruebas correspondientes a las métricas definidas en las Ecuaciones 5.4, 5.5 y 5.6 respectivamente.

Tabla 5.8: Resultados de la experimentación.

Escenario	Conjunto de Prueba	*n*-grama	Cobertura	Precisión	Medida-F
1. Corpus Web	**A**	**3**	0.5929	0.6734	0.6218
		5	0.6482	0.6835	0.6627
2. Benchmark	**B**	**3**	0.4405	0.4388	0.4359
		5	0.4212	0.4218	0.4178
	C	**3**	0.4426	0.4396	0.4376
		5	0.4255	0.4244	0.4216

Lo resultados obtenidos en el Escenario 1 del Corpus Web, los cuales están por encima del 0.64 en precisión, cobertura y medida-f para n-gramas de longitud 5, permiten afirmar que el método propuesto funciona adecuadamente. Este método de NER basado en un clasificador Bayesiano simple, permite reconocer entidades nombradas a partir de únicamente información sintáctica, es decir, de n-gramas circundantes de entidades nombradas. Esta capacidad de aprender a reconocer entidades nombradas a partir de información sintáctica, y no a partir de características asociadas a estructuras gramaticales y morfológicas, hace posible diseñar un reconocer de entidades nombradas para el idioma español sin la necesidad de contar con una amplia experiencia o grandes conocimientos en dicho lenguaje. Diseñar/entrenar el método de NER propuesto para obtener los resultados mostrados en la Tabla 5.8 no requirió costos en esfuerzo por lingüistas expertos para que definieran un conjunto de características asociadas a estructuras gramáticas, sintácticas o morfológicas y, por lo tanto, tampoco se requerirá incurrir en este tipo costos para mantenerlos actualizados o para adecuarlos/ajustaros a otros dominios o contextos.

Por otro lado, los mejores resultados de desempeño de los métodos de NER propuestos en la literatura se encuentran alrededor del 0.81 en precisión, cobertura y medida-f. Si bien el resultado de 0.64 obtenido con el método propuesto presenta una diferencia del 0.17 en comparación al 0.81 de los métodos propuestos en la literatura, esta diferencia se debe a que al dejar de utilizar información asociada a estructuras gramaticales o morfológicas el método pierda información discriminante con la cual pueda reconocer con mayor precisión una entidad nombrada.

Para compensar la pérdida de dicha información que se presenta al dejar de utilizar información gramatical o morfológica, es necesario contar con grandes fuentes de información (tamaño del conjunto de entrenamiento). Mientras mayor sea el volumen de datos con el que se entrene el modelo, mayor será su desempeño. Un ejemplo de esto se puede observar en los Escenarios 1 y 2. En el Escenario 2, correspondiente al benchmark CoNLL, los mejores resultados del método propuesto rondar por encima del 0.42 (ver Tabla 5.8). Si comparamos este resultado de 0.42 obtenido en el Escenario 2 del benchmark con el 0.64 obtenido en el Escenario 1 del Corpus Web, podemos observar

que existe una diferencia del 0.22. Sin embargo, si también comparamos la información con la que ambos fueron entrenados en términos de cantidad de entidades nombradas y sentencias, se puede observar que el Escenario 1 fue entrenado con un conjunto de datos con más de 7 mil sentencias y más de 18 mil entidades (ver Tabla 5.5) y el Escenario 2 con más de 900 mil sentencias y más de 81 mil entidades (ver Tabla 5.3). Con lo anterior se pude determinar/calcular que el Escenario 1 fue entrenado con 4 veces más entidades nombradas y 125 veces más sentencias que el Escenario 2. Por esta diferencia en el tamaño del conjunto de entrenamiento, el resultado de 0.42 del Escenario 2 es menor que el 0.64 del Escenario 1 debido a que el método no fue entrenado con la suficiente información para reconocer con un mayor desempeño entidades nombradas. El resultado de ambos escenarios están limitados por el tamaño del conjunto de documentos con el cual fueron entrenados. Por lo anterior, creemos que es posible que al aumentar el tamaño del conjunto de datos los resultados mejoren.

5.5.1 Comparativa con trabajos relacionados

La presente comparativa es realizada con respecto a los resultados de los métodos presentados en el benchmark CoNLL. Los resultados obtenidos del Escenario 2 correspondientes al experimento del benchmark, como se puede observar en la Tabla 5.8, mostraron resultados por debajo del desempeño medio en comparación de las propuestas presentadas en CoNLL, los cuales se muestran en la Tabla 5.9. Además, si comparamos el resultado de la medida-f del Escenario 2 con aquellos presentados en CoNLL, se puede observar una diferencia en desempeño del 18.81 %, hasta de un 39.23 %. Sin embargo, al comparar las diferentes características que utilizan las propuestas de CoNLL, las cuales se muestran en la Tabla 5.10 y se describen en la Tabla 5.11, se puede observar como el método propuesto no utiliza ninguna característica de las 13 mostradas, sino que únicamente se limita a utilizar *stopwords* y n-gramas. Por no utilizar información asociada al lenguaje el método propuesto presenta esta pérdida del 18.81 % al 39.23 %.

Tabla 5.9: Resultados de las propuestas presentadas en CoNLL [66].

Propuesta	**Precisión**	**Cobertura**	**Medida-F**
Carreras y Márques *et al.* [10]	0.8138	0.8140	0.8139
Radu Florian *et al.* [20]	0.7870	0.7940	0.7905
Cucerzan y Yarowsky *et al.* [15]	0.7819	0.7614	0.7715
Wu and Ngai *et al.* [74]	0.7585	0.7738	0.7661
Burger y Henderson *et al.* [9]	0.7419	0.7744	0.7578
Kim Sang y Erik F *et al.* [65]	0.7600	0.7555	0.7578
Patrick y Whitelaw *et al.* [52]	0.7432	0.7352	0.7392
Martin Jansche *et al.* [28]	0.7403	0.7376	0.7389
Robert Malouf *et al.* [39]	0.7393	0.7339	0.7366
Tsukamoto y Mitsuishi *et al.* [70]	0.6904	0.7412	0.7149
Black y Vasilakopoulo *et al.* [6]	0.6053	0.6729	0.6373
McNamee y Mayfield *et al.* [44]	0.5628	0.6651	0.6097

Tabla 5.10: Características que utilizan las propuestas presentadas en CoNLL.

Característica	[10]	[20]	[15]	[74]	[9]	[65]	[52]	[28]	[39]	[70]	[6]	[44]
1.- Morfología	+	+		+						+		
2.- POST	+	+		+	+	+				+		
3.- Mayúsculas	+	+		+				+	+	+	+	+
4.- Hipónimos	+											
5.- Acrónimos	+											
6.- Bag-of-Words	+											
7.- Gazetteers	+	+	+	+								
8.- Ortografía	+						+	+	+			+
9.- Longitud	+		+	+		+		+		+		+
10.- Sufijos	+		+	+		+	+			+	+	+
11.- Prefijos	+		+	+		+	+			+		+
12.- Lematización				+								
13.- Posición								+	+		+	+

Por otro lado, si comparamos el método con el mejor resultado presentado en CoNLL (ver Tabla 5.9), el cual corresponde a la propuesta de Carreras y Márques [10], se puede observar una diferencia de 39.23 % de mejora en comparación al aquí propuesto. Sin embargo, si también se comparan el número de características que ambos métodos usan, se puede observar que el método de Carreras y Márques utiliza 11 características, tanto gramaticales, morfológicas y sintácticas, y que, por el otro lado, el método aquí propuesto sólo utiliza 2 características: stopwords y n-gramas. Como ya se ha argumentado anteriormente, dicha diferencia en desempeño que existe entren ambos métodos se debe a las diferentes características que incorporan en sus modelos probabilísticos.

Tabla 5.11: Descripción general de las características que usualmente utilizan los métodos de NER.

CARACTERÍSTICA	DESCRIPCIÓN
1.- Morfología	Estructura interna de las palabras, como morfemas y lexemas.
2.- POST	Etiquetado gramatical, como adjetivos, verbos, sustantivos, etc.
3.- Mayúsculas	Cuanto una palabra inicia con mayúscula, comúnmente es indicio de una entidad nombrada.
4.- Hipónimos	Es una referencia general de una entidad nombrada, como fruta es hipónimo de manzana y plátano.
5.- Acrónimos	En el caso de NER, dado una lista de entidades nombradas a prori (entrenamiento), también se extrae su acrónimo.
6.- Bag-of-Words	Es utilizado para representa una entidad nombrada en ventana de palabras, sin considerar su posición.
7.- Gazetteers	Lista de palabras de un dominio en especifico. Un gazetteers de países, tendrá los términos México, USA, Panamá, etc.
8.- Ortografía	Es cuando se utilizan reglas ortográficas definidas en un lenguaje para reconocer entidades nombradas.
9.- Longitud	Es cuando se utiliza el número de letras/símbolos por las que se compone una palabra.
10.- Sufijos	Morfema o afijo que se agrega después del lexema o raíz de una palabra, como "ing", "ed" o "ly" para el inglés.
11.- Prefijos	Morfema o afijo que se agrega antes del lexema o raíz de una palabra, como "mis", "re" o "in" para el inglés.
12.- Lematización	Utilizan el lema de las palabra para representarlas en un modelo.
13.- Posición	Utilizan la posición de una palabra dentro de una sentencia. Las primeras palabras de una sentencia comúnmente son entidades.

6

Conclusiones

En este trabajo se presentó el diseño, desarrollo y evaluación de un método de aprendizaje para el NER que reduce su dependencia del lenguaje en comparación a las propuestas del estado del arte. En este capítulo se presentan las conclusiones de este trabajo de tesis, así como las contribuciones, limitantes y trabajo futuro del mismo.

6.1 Resumen

Los métodos de NER actuales para generar modelos basados en estadística o en aprendizaje automático usualmente hacen uso de estructuras propias del lenguaje, tales como las gramaticales, sintácticas y morfológicas. Estos métodos para su efectividad requieren de un gran volumen de datos de entrenamiento, los cuales usualmente son etiquetados manualmente por lingüistas expertos. Estos dos grandes problemas plantean los siguientes retos: la dependencia del lenguaje de los métodos y la necesidad de intervención humana para el etiquetado de corpus. En este trabajo se propone un método NER que reduce la dependencia del lenguaje en comparación a las técnicas actuales ya que no requiere de procesos de etiquetado manual sobre corpus de entrenamiento.

En el método propuesto se desarrolló un método de filtrado de búsqueda web que permite realizar consultas web para explorar de forma sistemática y focalizada diferentes recursos textuales de la Web que contengan entidades nombradas. Estos recursos textuales son procesados para extraer la información circundante (n-gramas) de cada entidad nombrada para obtener, al menos desde la perspectiva estadística, aquellas palabras comunes a una tipo de entidad nombrada. Con esta información extraída se genera un modelo probabilístico basado en un clasificador Bayesiano simple. Dicho modelo es capaz de reconocer probabilísticamente entidades nombradas de cuatro tipos: lugares, personas, organizaciones y otros. El desempeño del método propuesto en este trabajo fue evaluado con dos escenarios de prueba: el primero fue a partir de conjuntos de datos textuales recolectados de la Web y el segundo con el benchmark CoNLL. Los resultados obtenidos en esta experimentación mostraron que es posible contar con un método NER que reconozca entidades nombradas a partir de únicamente información sintáctica (n-gramas circundantes), dejando de lado la información gramatical o morfológica. Sin embargo, aunque el método es competitivo, el no utilizar esta información asociada al lenguaje hace que el método de NER propuesto presente un desempeño por debajo de las propuestas actuales del estado del arte, lo cual presenta una desventaja. Para resolver esto es necesario contar con grandes conjuntos de datos de entrenamiento para que el método, a partir de mayor información, pueda reconocer de mejor manera entidades nombradas.

6.2 Contribuciones

El método de aprendizaje propuesto ofrece la posibilidad de diseñar un reconocedor de entidades nombradas basado en probabilidad mediante el análisis de los elementos de n-gramas circundantes de entidades nombradas. Además, se contribuye con un método automatizado de recolección de datos, con el cual es posible buscar de manera focalizada recursos textuales de la Web. Con dicho método es posible obtener información para la creación de repositorios de datos, los cuales sean útiles para el entrenamiento de métodos de NER, evitando así procesos de etiquetado manual o costos en tiempo, personas e infraestructura.

6.3 Limitantes

Si bien los experimentos realizados se enfocaron en reconocer entidades nombradas asociadas a lugares, personas, organizaciones y nombres variados, es posible adecuar el método para que reconozca otros tipos de entidades, tales como universidades, deportes, artistas, finanzas, etc.

Por otro lado, el método automatizado de recolección de datos propuesto utiliza el motor de búsqueda de Google para obtener recursos textuales de la Web. Dicho motor de búsqueda permite cierto número de consultas en un intervalo de tiempo definido. Esta restricción es una medida preventiva que el motor implementa para evitar ataques DDoS. Dicho filtro causa que la recolección de datos sea lenta y gradual. Por ejemplo, el curpus de textos en español utilizado en la experimentación tomó 3 meses en ser recolectado. Por lo anterior, para generar otro corpus de entrenamiento es necesario considerar que el tiempo de recolección de datos es un factor muy importante a tomar en cuenta.

6.4 Trabajo futuro

A corto plazo se espera recolectar más datos para aumentar el desempeño del método de NER, así como adecuar dicho método para el reconocimiento de entidades nombradas georeferenciables. Además se realizarán experimentos con el idioma Inglés con la finalidad de ver si el método propuesto es pertinente para reconocer entidades en dicho idioma. Hasta el momento se ha recolectado un corpus en Inglés de más de 18 mil documentos con más de 150 mil sentencias.

Bibliografía

[1] Abdallah, S., Shaalan, K., and Shoaib, M. (2012). Integrating rule-based system with classification for arabic named entity recognition. *Computational Linguistics and Intelligent Text Processing*, pages 311–322.

[2] Amarappa, S. and Sathyanarayana, S. (2015). Kannada named entity recognition and classification (nerc) based on multinomial naive bayes (mnb) classifier.

[3] Bhagavatula, M., GSK, S., and Varma, V. (2012). Language-independent named entity identification using wikipedia. In *Proceedings of the First Workshop on Multilingual Modeling*, pages 11–17.

[4] Bikel, D. M., Miller, S., Schwartz, R., and Weischedel, R. (1997). Nymble: a high-performance learning name-finder. In *Proceedings of the fifth conference on Applied natural language processing*, pages 194–201.

[5] Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). Dbpedia-a crystallization point for the web of data. *Web Semantics: science, services and agents on the world wide web*, 7(3):154–165.

[6] Black, W. J. and Vasilakopoulos, A. (2002). Language-independent named entity classification by modified transformation-based learning and by decision tree induction. In *Proceedings of CoNLL-2002*, pages 159–162. Taipei, Taiwan.

[7] Borthwick, A., Sterling, J., Agichtein, E., and Grishman, R. (1998). Nyu: Description of the mene named entity system as used in muc-7. In *In Proceedings of the Seventh Message Understanding Conference (MUC-7.*

[8] Bunescu, R. and Pasca, M. (2006). Using encyclopedic knowledge for named entity disambiguation. In *11th conference of the European Chapter of the Association for Computational Linguistics*, pages 9–16.

[9] Burger, J. D., Henderson, J. C., and Morgan, W. T. (2002). Statistical named entity recognizer adaptation. In *Proceedings of CoNLL-2002*, pages 163–166. Taipei, Taiwan.

[10] Carreras, X., Màrques, L., and Padró, L. (2002). Named entity extraction using adaboost. In *Proceedings of CoNLL-2002*, pages 167–170. Taipei, Taiwan.

[11] Chellappa, R. and Jain, A. (1993). Markov random fields. theory and application. *Boston: Academic Press, 1993, edited by Chellappa, Rama; Jain, Anil*.

[12] Cho, H.-C., Okazaki, N., Miwa, M., and Tsujii, J. (2013). Named entity recognition with multiple segment representations. *Information Processing & Management*, 49(4):954–965.

[13] Chowdhury, G. G. (2003). Natural language processing. *Annual review of information science and technology*, 37(1):51–89.

[14] Cowie, J. and Lehnert, W. (1996). Information extraction. *Communications of the ACM*, 39(1):80–91.

[15] Cucerzan, S. and Yarowsky, D. (2002). Language independent ner using a unified model of internal and contextual evidence. In *Proceedings of CoNLL-2002*, pages 171–174. Taipei, Taiwan.

[16] Curran, J. R. and Clark, S. (2003). Language independent ner using a maximum entropy tagger. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 164–167.

[17] Dojchinovski, M. and Kliegr, T. (2013). Entityclassifier.eu: real-time classification of entities in text with wikipedia. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 654–658.

[18] Ekbal, A. and Bandyopadhyay, S. (2007). A hidden markov model based named entity recognition system: Bengali and hindi as case studies. *Pattern Recognition and Machine Intelligence*, pages 545–552.

[19] Fagin, R., Kimelfeld, B., Reiss, F., and Vansummeren, S. (2016). Declarative cleaning of inconsistencies in information extraction. *ACM Transactions on Database Systems (TODS)*, 41(1):6.

[20] Florian, R. (2002). Named entity recognition as a house of cards: Classifier stacking. In *Proceedings of CoNLL-2002*, pages 175–178. Taipei, Taiwan.

[21] Gaizauskas, R. and Wilks, Y. (1998). Information extraction: Beyond document retrieval. *Journal of documentation*, 54(1):70–105.

[22] Ghosh, K. and Bhattacharya, A. (2017). Stopword removal: Why bother? a case study on verbose queries. In *Proceedings of the 10th Annual ACM India Compute Conference on ZZZ*, pages 99–102. ACM.

[23] Gorla, S., Velivelli, S., Murthy, N. L., and Malapati, A. (2018). Named entity recognition for telugu news articles using naive bayes classifier.

[24] Goyal, A., Gupta, V., and Kumar, M. (2018). Recent named entity recognition and classification techniques: A systematic review. *Computer Science Review*, 29:21–43.

[25] Grishman, R. and Sundheim, B. (1996). Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, volume 1.

[26] Gunawan, W., Suhartono, D., Purnomo, F., and Ongko, A. (2018). Named-entity recognition for indonesian language using bidirectional lstm-cnns. *Procedia Computer Science*, 135:425–432.

[27] Han, J., Pei, J., and Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

[28] Jansche, M. (2002). Named entity extraction with conditional markov models and classifiers. In *Proceedings of CoNLL-2002*, pages 179–182. Taipei, Taiwan.

[29] Joshi, A. K. (1991). Natural language processing. *Science*, 253(5025):1242–1249.

[30] Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing (2Nd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[31] Kao, A. and Poteet, S. R. (2007). *Natural language processing and text mining*. Springer Science & Business Media.

[32] Karkaletsis, V., Paliouras, G., Petasis, G., Manousopoulou, N., and Spyropoulos, C. D. (1999). Named-entity recognition from greek and english texts. *Journal of Intelligent and Robotic Systems*, 26(2):123–135.

[33] Konkol, M. and Konopík, M. (2011). Maximum entropy named entity recognition for czech language. In *Text, Speech and Dialogue*, pages 203–210.

[34] Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

[35] Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition.

[36] Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al. (2015). Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

[37] Liddy, E. D. (1998). Enhanced text retrieval using natural language processing. *Bulletin of the American Society for Information Science and Technology*, 24(4):14–16.

[38] Mahalakshmi, G., Antony, J., Roshini, S., et al. (2016). Domain based named entity recognition using naive bayes classification.

[39] Malouf, R. (2002a). Markov models for language-independent named entity recognition. In *Proceedings of CoNLL-2002*, pages 187–190. Taipei, Taiwan.

[40] Malouf, R. (2002b). Markov models for language-independent named entity recognition, proceedings of the 6th conference on natural language learning. *August*, 31:1–4.

[41] Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

[42] Maynard, D., Tablan, V., Ursu, C., Cunningham, H., and Wilks, Y. (2001). Named entity recognition from diverse text types. In *Recent Advances in Natural Language Processing 2001 Conference*, pages 257–274.

[43] McCallum, A. and Li, W. (2003). Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 188–191.

[44] McNamee, P. and Mayfield, J. (2002). Entity extraction without language-specific resources. In *Proceedings of CoNLL-2002*, pages 183–186. Taipei, Taiwan.

[45] Mendes, P. N., Jakob, M., García-Silva, A., and Bizer, C. (2011). Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8.

[46] Moens, M.-F. (2006). *Information extraction: algorithms and prospects in a retrieval context*. Springer Science & Business Media.

[47] Molina, C. A. C., Gutierrez, R. E., and Solarte, O. (2015). Prototipo para el reconocimiento de entidades nombradas en el idioma español. In *Computing Colombian Conference (10CCC), 2015 10th*, pages 364–371.

[48] Mrabet, Y., Kilicoglu, H., and Demner-Fushman, D. (2016). Unsupervised ranking of knowledge bases for named entity recognition. In *ECAI*, pages 1248–1255.

[49] Márquez, L., Carreras, X., Litkowski, K. C., and Stevenson, S. (2008). Semantic role labeling: an introduction to the special issue.

[50] Nadeau, D. and Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

[51] Nadkarni, P. M., Ohno-Machado, L., and Chapman, W. W. (2011). Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551.

[52] Patrick, J., Whitelaw, C., and Munro, R. (2002). Slinerc: The sydney language-independent named entity recogniser and classifier. In *Proceedings of CoNLL-2002*, pages 199–202. Taipei, Taiwan.

[53] Piskorski, J. and Yangarber, R. (2013). Information extraction: Past, present and future. In *Multi-source, multilingual information extraction and summarization*, pages 23–49. Springer.

[54] Prabhakar, D. K., Dubey, S., Goel, B., and Pal, S. (2014). Named entity recognition for indian languages. In *Proceedings of the Forum for Information Retrieval Evaluation*, pages 98–102. ACM.

[55] Provost, F. and Domingos, P. (2003). Tree induction for probability-based ranking. *Machine learning*, 52(3):199–215.

[56] Ratinov, L. and Roth, D. (2009). Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 147–155. Association for Computational Linguistics.

[57] Ratnaparkhi, A. (1997). A simple introduction to maximum entropy models for natural language processing. *IRCS Technical Reports Series*, page 81.

[58] Rijsbergen, C. J. V. (1979). *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition.

[59] Rodrigues, M., Teixeira, A., et al. (2015). *Advanced applications of natural language processing for performing information extraction*. Springer.

[60] Sekine, S. (1998). Nyu: Description of the japanese ne system used for met-2. In *Proc. of the Seventh Message Understanding Conference (MUC-7)*.

[61] Singh, A., Rani, J., and Singh, K. (2013). Named entity recognition: A review. *International Journal of Computer Sciencie and Communication Engineering*, pages 36–39.

[62] Stewart, W. J. (2009). *Probability, Markov chains, queues, and simulation: the mathematical basis of performance modeling*. Princeton University Press.

[63] Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706.

[64] Szarvas, G., Farkas, R., and Kocsor, A. (2006). A multilingual named entity recognition system using boosting and c4. 5 decision tree learning algorithms. In *International Conference on Discovery Science*, pages 267–278.

[65] Tjong Kim Sang, E. F. (2002). Memory-based named entity recognition. In *Proceedings of CoNLL-2002*, pages 203–206. Taipei, Taiwan.

[66] Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

[67] Todorović, B. T., Rančić, S. R., and Mulalić, E. H. (2010). Context hidden markov model for named entity recognition. In *Approximation and Computation*, pages 447–460.

[68] Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180.

[69] Toutanova, K. and Manning, C. D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pages 63–70.

[70] Tsukamoto, K., Mitsuishi, Y., and Sassano, M. (2002). Learning with multiple stacking for named entity recognition. In *Proceedings of CoNLL-2002*, pages 191–194. Taipei, Taiwan.

[71] Turmo, J., Ageno, A., and Català, N. (2006). Adaptive information extraction. *ACM Computing Surveys (CSUR)*, 38(2):4.

[72] Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269.

[73] Wibawa, A. S. and Purwarianti, A. (2016). Indonesian named-entity recognition for 15 classes using ensemble supervised learning. *Procedia Computer Science*, 81:221–228.

[74] Wu, D., Ngai, G., Carpuat, M., Larsen, J., and Yang, Y. (2002). Boosting for named entity recognition. In *Proceedings of CoNLL-2002*, pages 195–198. Taipei, Taiwan.

[75] Wu, J. and Khudanpur, S. (2003). *Maximum entropy language modeling with non-local dependencies*. PhD thesis, Johns Hopkins University.

[76] Yosef, M. A., Hoffart, J., Bordino, I., Spaniol, M., and Weikum, G. (2011). Aida: An online tool for accurate disambiguation of named entities in text and tables. *Proceedings of the VLDB Endowment*, 4(12):1450–1453.

[77] Zitouni, I. (2014). *Natural language processing of semitic languages*. Springer.