PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

Extracción de información para la generación de reportes estructurados a partir de noticias peruanas relacionadas a crímenes

TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERA INFORMÁTICA

AUTOR

Gina Bustamante Alvarez

ASESOR:

Félix Arturo Oncevay Marcos

Lima, Agosto, 2019

RESUMEN

Actualmente, múltiples fuerzas policiales y agencias de inteligencia han decidido enfocar parte de sus esfuerzos en la recolección de todo tipo de información sobre crímenes. Esto con el objetivo de poder analizar los datos disponibles y utilizar los resultados de esta tarea para la mejora de procesos actuales, e incluso, para la prevención de ataques y delitos futuros.

No obstante, debido a la velocidad con la que se generan datos e información en la última década, las técnicas de análisis tradicional han resultado en baja productividad y en un uso ineficiente de recursos. Es por esta razón, que desde el campo de la informática, y específicamente desde las ciencias de la computación, se vienen realizado múltiples intentos para ayudar a identificar y obtener la información más importante dentro de estos grandes volúmenes de datos.

Hasta el momento los estudios previos realizados para este dominio, abarcan desde la predicción del lugar de un delito utilizando data numérica, hasta la identificación de nombres y entidades en descripciones textuales. En este contexto, este trabajo propone la creación de una herramienta de extracción de información para noticias relacionadas al dominio criminal peruano. Buscando identificar automáticamente culpables, víctimas y locaciones mediante los siguientes pasos: (1) Procesamiento y generación de un conjunto de datos en base a noticias criminales, (2) Implementación y validación de algoritmos de extracción e información, y (3) Elaboración de una interfaz de programación de aplicaciones para el consumo del modelo desarrollado.

Los resultados obtenidos evidencian que el enfoque utilizado, basado en dependencias sintácticas y reconocimiento de entidades nombradas, es exitoso. Además, se espera que

en el futuro se puedan mejorar los resultados obtenidos con técnicas de procesamiento de lenguaje natural para dominios con pocos recursos.

A mis padres, por criarme libre y siempre pensar que podía lograr todo lo que me proponga

A mi familia, por siempre alentarme a cumplir todas mis metas

A mis amigos, por su compañía y apoyo durante estos años

A mis profesores, por enseñarme a ver el mundo de otra manera

Y a todos los que sueñan con crear tecnología

TABLA DE CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

Capítulo 1. Generalidades

1.1 Problemática

El análisis criminal, entendido como el conjunto de técnicas utilizadas para extraer datos e información valiosa de las descripciones de un delito (LeBlanc, Elder, Bruce, & Santos, 2014), se ha vuelto una actividad vital en la era actual debido a que el número de crímenes cometidos a nivel mundial aumenta día con día (Jayaweera et al., 2015). Por otro lado, la revolución tecnológica de estos últimos años ha permitido que cada vez sea más fácil generar y compartir datos e información, causando que el volumen de estos aumente a una velocidad nunca antes vista (McAfee & Brynjolfsson, 2012). Considerando estos dos hechos, es de esperarse que las fuerzas del orden y agencias de inteligencia se hayan dedicado a la misión de recolectar grandes cantidades de datos (también llamados *"Big Data*") sobre crímenes, con el fin de analizarlos y utilizar los resultados obtenidos para tratar de prevenir ataques o delitos futuros (Chibelushi, Sharp, & Shah, 2006).

Sin embargo, puesto que ahora se está tratando con un gran volumen de datos, las técnicas manuales usadas tradicionalmente para el análisis de información relacionada a crímenes donde, por ejemplo, un analista se dedica a identificar culpables, víctimas y locaciones basándose en el juicio experto (Crime, 2011) han resultado en baja productividad y en un uso ineficiente de recursos humanos (Jayaweera et al., 2015). Esto ha causado que el procesamiento de esta gran cantidad de datos se haya vuelto uno de los principales desafíos de las organizaciones dedicadas a la tarea del análisis de crímenes, creándose así la necesidad de buscar nuevos enfoques para esta (Gupta, 2014).

En este contexto, en el que se tiene una gran cantidad de datos de donde se puede extraer información valiosa, la minería de datos definida como el proceso que se encarga de descubrir información oculta en grandes conjuntos de datos (Hassani, Huang, Silva, & Ghodsi, 2016), posee un gran potencial para ayudar a identificar y extraer la información más importante

oculta dentro del *"Big Data"* del crimen (Thongtae & Srisuk, 2008). Prueba de lo mencionado anteriormente son las diferentes técnicas de minería de datos que fueron diseñadas para la clasificación, predicción y perfilamiento del comportamiento humano durante los años ochenta, con el objetivo de rastrear y detener criminales. Demostrando así que es posible el análisis automático de tendencias de crimen y comportamientos criminales, sin la necesidad de la intervención humana (Pramanik, Lau, Yue, Ye, & Li, 2017). En la última década, también son varias las técnicas de minería de datos que ya han sido aplicadas y probadas exitosamente en tareas de análisis criminal a fin de identificar y prevenir crímenes a través del tiempo (Hassani et al., 2016).

Del mismo modo, el procesamiento de lenguaje natural, entendido como el uso de técnicas computacionales para aprender y entender el lenguaje de los humanos (Hirschberg & Manning, 2015), es también de gran ayuda cuando se necesita procesar grandes cantidades de datos no estructurados, y sus técnicas son completamente aplicables al análisis criminal (van Banerveld, Le-Khac, & Kechadi, 2014). Ejemplo de esto son las investigaciones realizadas utilizando técnicas de extracción de información, específicamente el reconocimiento de entidades nombradas, para identificar y extraer automáticamente entidades significativas como nombres propios de personas, direcciones y nombres de narcóticos a partir de reportes policiales (Chau, Xu, & Chen, 2002).

A pesar de que se ha demostrado la utilidad y los beneficios de la aplicación de técnicas pertenecientes al área de ciencias de la computación, varios autores han identificado que el acceso a la información sobre crímenes juega un papel importante en la efectividad de estas, y que varios problemas surgen cuando, a pesar de las grandes cantidades de información existentes, este acceso es obstaculizado por cuestiones de privacidad o confidencialidad (Gupta, 2014). Estos problemas han sido evaluados desde diferentes perspectivas. Primero, se encuentran los países que poseen esta información pero no la ponen a disponibilidad de la

comunidad, o solamente hacen pública información de alto nivel (como el número total de robos en un distrito o provincia), la cual, sin el contexto adecuado, no es muy útil para un ciudadano promedio (Arulanandam, Savarimuthu, & Purvis, 2014). En segundo lugar, se encuentran los países donde todavía se usa un sistema de registro y análisis de crímenes manual, basado plenamente en la habilidad del analista, como el caso de Sri Lanka y demás países no desarrollados (Jayaweera et al., 2015).

Frente a estas limitaciones en el acceso a datos e información, se ha propuesto en diferentes investigaciones el uso de noticias relacionadas a crímenes extraídas de internet como conjunto de datos alternativo. Entre los casos destacados está el sistema de monitoreo VnLoc el cual busca extraer eventos de las noticias en idioma vietnamita publicadas en internet (Tran, Nguyen, Nguyen, Nguyen, & Phan, 2012), y el sistema I-JEN desarrollado en Malasia para la recuperación de información de las noticias criminales (Mohd & Ali, 2011). El principal argumento a favor de este enfoque es que los periódicos contienen una vasta cantidad de información pública sobre crímenes que puede ser explotada y organizada a fin de ser usada posteriormente por la comunidad, para su seguridad personal, o incluso por fuerzas policiales como ayuda para el análisis criminal (Arulanandam et al., 2014).

Los periódicos son conocidos ya como una amplia fuente de diversos tipos de información (Krtalic & Hasenay, 2012), puesto que las noticias contenidas en ellos poseen datos con potencial para rastrear situaciones como eventos, personas, organizaciones y otros problemas relacionados (Kittiphattanabawon & Theeramunkong, 2009). Prueba de lo mencionado son los estudios como el de Tibbo (Tibbo, 2002) donde se pone a los periódicos como una fuente de información altamente usada por historiadores. Por otro lado, en estudios como el de Patterson et all (Patterson, Emslie, Mason, Fergie, & Hilton, 2016) se han utilizado noticias para analizar problemas sociales tales como los estereotipos de género presentes al momento de narrar o reportar excesos en el consumo de alcohol en el Reino Unido. En el caso particular de las

noticias sobre crímenes, son pocas las investigaciones o herramientas dedicadas a explotar el potencial de información presente en ellas, principalmente por la tarea tediosa que representa generar un conjunto de datos en base a estas. Ocasionando así que una gran cantidad de datos se desperdicie y no sea usada para el beneficio de comunidades mediante la elaboración de herramientas o modelos predictivos (Netsuwan & Kesorn, 2017).

Las bases de conocimiento sobre crimen que se pueden elaborar a partir de noticias son beneficiosas para una gran variedad de usuarios finales, ya que contienen información pre-procesada y organizada sobre actividades delictivas y posibles delincuentes asociados a estas (Westphal, 2008). Sin embargo, la mayoría de aplicaciones de extracción de información de noticias mencionadas anteriormente son de uso restringido, ocasionando así que el problema de acceso a datos e información no se solucione del todo.

Por tanto, en el presente proyecto se propone la implementación de una herramienta de extracción de información de noticias de dominio criminal, utilizando técnicas de procesamiento de lenguaje natural para datos textuales no estructurados. La herramienta será capaz de recolectar las noticias de internet y extraer información relevante de éstas, para posteriormente presentarla en forma de un reporte estructurado. Una aplicación de este tipo sirve no solo para poner a disposición de un público general la información extraída, sino también para la generación de conjuntos de datos a utilizar en el desarrollo de modelos predictivos en un futuro.

1.2 Objetivos

En esta sección se indicarán los objetivos que se desea alcanzar y los resultados esperados del presente proyecto de fin de carrera.

1.2.1 Objetivo general

Implementar un modelo algorítmico de extracción de información de descripciones de delitos en noticias peruanas relacionadas a crímenes.

1.2.2 Objetivos específicos

- O1: Procesar y generar un conjunto de datos estructurado en base a noticias criminales en la web.
- O2: Implementar y validar un algoritmo para la extracción automática de información de las noticias.
- O3: Elaborar una interfaz de programación de aplicaciones para la presentación de funcionalidades del modelo desarrollado

1.2.3 Resultados esperados

- R1: Para el objetivo 1.- Programa de búsqueda y extracción automática de noticias relacionadas a crímenes en la web.
- R2: Para el objetivo 2.- Comparación de módulos de software para la extracción de entidades en los textos
- R3: Para el objetivo 2.- Personalización del módulo de reconocimiento de entidades nombradas para el dominio y vocabulario específico
- R4: Para el objetivo 2.- Recurso léxico con términos de dominio criminal
- R5: Para el objetivo 2.- Programa de generación de lenguaje natural para la emisión de reportes estructurados

- R6: Para el objetivo 3.- Interfaz de programación de aplicaciones que utilice el algoritmo implementado para extraer la información importante del texto de una noticia y genere un reporte con esta.

1.2.4 Mapeo de objetivos y resultados esperados

Tabla 1 Mapeo de objetivos y resultados (elaboración propia)

Objetivo: Procesar y generar un conjunto de datos estructurado en base a noticias criminales en la web		
Resultado	**Meta física**	**Medio de verificación**
Programa de búsqueda y extracción automática de noticias criminales en la web	Conjunto de datos	Pruebas de integridad de datos
Objetivo: Implementar y validar un algoritmo para la extracción automática de información de las noticias.		
Resultado	**Meta física**	**Medio de verificación**
Comparación de módulos de software para la extracción de entidades en los textos	Software	Experimentación numérica
Personalización del módulo de reconocimiento de entidades nombradas para el dominio y vocabulario específico	Módulo de reconocimiento de entidades nombradas tuneado	Métricas de precisión y exactitud
Recurso léxico con términos de dominio criminal	Tesauro	Métricas de esquema y métricas de la base de conocimiento
Programa de generación de lenguaje natural para la	Software	Métricas de precisión y exactitud

emisión de reportes estructurados		
Objetivo: Elaborar una interfaz de programación de aplicaciones para la presentación de funcionalidades del modelo desarrollado.		
Resultado	**Meta física**	**Medio de verificación**
Interfaz de programación de aplicaciones que utilice el algoritmo implementado para extraer la información importante del texto de una noticia y genere un reporte con esta.	Software	Plataforma web para pruebas de la interfaz de programación de aplicaciones.

1.3 Herramientas y métodos

1.3.1 Herramientas

Python

Python es un lenguaje de programación de alto nivel orientado a objetos, que se caracteriza por fomentar la reutilización de código y modularidad de programas mediante el usos de módulos y paquetes (Python Software Foundation, 2017). Del mismo modo, Python se distingue de otros lenguajes por sus excelentes funcionalidades para procesar datos lingüísticos (Bird, Klein, & Loper, 2009) razón por la cual fue elegido como leguaje principal en el desarrollo del presente proyecto.

Jupyter Notebook

Jupyter Notebook es una aplicación web utilizada para la creación de documentos que contengan código, ecuaciones, gráficas y textos (Proyect Jupyter, 2017). La utilidad de esta aplicación en el proyecto propuesto se ve justificada con la necesidad de presentar el código y resultados parciales de una manera ordenada y fácil de visualizar.

Beautiful Soup

Beautiful Soup es una librería de Python para extraer datos de archivos HTML Y XML. La biblioteca crea una estructura con todos los elementos del documento y permite su acceso mediante métodos propios de esta (Richardson, 2016). Dentro del proyecto a realizar, esta librería será utilizada para la extracción del texto de las noticias recolectadas de periódicos publicados en internet.

MongoDB

MongoDB es una base de datos documental de código abierto que ofrece altos niveles de rendimiento, disponibilidad y escalabilidad. Cada registro dentro de MongoDB se conoce como “documento”, una estructura de datos compuesta por pares de campo y el valor respectivo para este (MongoDB, 2017). Las bases de datos documentales como MongoDB son ideales

para el manejo de data semiestructurada y aplicaciones que involucren la agregación constante de registros a colecciones de documentos (Gudivada, Rao, & Raghavan, 2014). Es por las razones anteriormente expuestas, que se ha seleccionado como base de datos para el presente proyecto.

Protégé

Protégé es un software de ayuda a la construcción de recursos léxicos y sistemas basados en conocimiento desarrollado por la Universidad de Stanford. Es por amplio margen el software más utilizado para los propósitos mencionados, registrando hasta la fecha más de 250000 usuarios (Musen & Protégé Team, 2015). Durante el desarrollo del proyecto propuesto, se utilizará Protégé para la creación y edición del tesauro de dominio criminal.

Hermit

Hermit es un algoritmo de razonamiento para recursos léxicos escrito utilizando Web Ontology Language (OWL). Dado un archivo OWL, Hermit puede detectar automáticamente si la ontología o tesauro es consistente e identificar relaciones entre las clases (Horrocks, 2013). Se utilizará para hacer las verificaciones de consistencia respectivas del tesauro desarrollado.

Wordnet

Wordnet es una base de datos léxica diseñada especialmente para ser utilizada por algoritmos computacionales. Sustantivos, verbos, adjetivos y adverbios están agrupados en conjuntos de sinónimos, donde cada uno de estos grupos representa un concepto (Miller, 1995).

Natural Language Toolkit

Natural Language Toolkit es una librería de Python que posee una amplia gama de funcionalidades destinadas al procesamiento de lenguaje natural (Bird et al., 2009). Algunas de las funcionalidades ofrecidas por esta librería serán utilizadas en el pre procesamiento de las noticias recolectadas.

Stanford CoreNLP

El Stanford CoreNLP es un *pipeline* (conjunto de elementos de procesamiento de datos conectados en serie) que provee herramientas para la mayoría de pasos del procesamiento de lenguaje natural. Este conjunto de herramientas es actualmente usado tanto en investigaciones, como aplicaciones comerciales y gubernamentales de código abierto (Manning et al., 2014). En el proyecto a realizar, se usarán algunas de las herramientas brindadas por el Stanford CoreNLP para el proceso de extracción de información.

SpaCy

SpaCy es una biblioteca gratuita y de código abierto para el procesamiento avanzado de lenguaje natural en Python. Se puede usar para construir sistemas de extracción de información o de comprensión del lenguaje natural, o para pre procesar texto para aprendizaje profundo (spaCy, 2018). Durante el presente proyecto, se utilizarán algunas de las funcionalidades brindadas por SpaCy para la extracción de información de las noticias.

PyTorch

PyTorch es un librería de Python diseñada para agilizar la investigación en modelos de aprendizaje de máquina (Paszke et al., 2017). Será utilizada para el entrenamiento de modelos de extracción de información durante el proyecto.

Flask

Flask es un marco de trabajo para Python, que busca crear aplicaciones simples y fácilmente extensibles (Ronacher, 2010). Dentro del proyecto propuesto, se usará en la creación de la plataforma web para pruebas de la interfaz de programación de aplicaciones.

Django

Django es marco de trabajo gratuito y de código abierto escrito en Python, utilizado en el desarrollo de páginas web. Sigue el patrón modelo-vista-plantilla (MVT, por sus siglas en inglés) y fue diseñado especialmente para hacer las tareas comunes del desarrollo web fáciles

y sencillas (Django Software Foundation, 2016). Será utilizado en el desarrollo de la plataforma web para las pruebas del presente proyecto.

1.3.2 Métodos

Metodología de desarrollo

- **Extracción de noticias de sitios web**

Se conoce como *web scraping* o extracción web a la práctica de recolectar datos mediante un programa automatizado que consulta a un servidor web. Este programa solicita datos, generalmente en formato HTML, y luego analiza su estructura para obtener exclusivamente los datos que necesita (Mitchell, 2015). Durante el proyecto presentado, se utilizarán técnicas de extracción web para recolectar el texto de las noticias desde la página donde han sido publicadas.

- **Extracción de información**

La tarea de extracción de información (IE, por sus siglas en inglés) se basa en analizar textos en lenguaje natural a fin de extraer fragmentos de información, que pueden ser publicados directamente o almacenados para un análisis posterior. El tipo de información a extraer como las técnicas y algoritmos utilizados para obtenerla varían dependiendo del objetivo de cada proyecto (Cunningham, 2006).

En la presente investigación, se planea utilizar técnicas de extracción de información para el dominio de noticias peruanas relacionadas a crímenes y de este modo obtener información como tipo de crimen, ubicación, actores, instrumentos, acciones y relaciones entre entidades de cada noticia.

Algunas de las técnicas a utilizar se mencionan a continuación:

- **Reconocimiento de entidades nombradas**

  La primera mención del término "Entidad nombrada" se dio durante la sexta *Message Understanding Conference* (MUC-6) por Grishman y Sundheim. El objetivo era

identificar, utilizando las técnicas de extracción de información desarrolladas hasta ese entonces, los nombres de personas, organizaciones y ubicaciones geográficas en un texto (Grishman & Sundheim, 1996).

- **Tesauros**

Un tesauro es una representación computacional del conocimiento sobre un dominio específico. Son comúnmente utilizados en la tarea de extracción de información, como parte del proceso de entendimiento e interpretación de un texto para poder extraer la información relevante necesitada (Nédellec & Nazarenko, 2006). En el presente proyecto se propone la elaboración de un tesauro de dominio criminal para detectar entidades y relaciones entre estas en las noticias utilizadas como corpus.

- **Etiquetado gramatical**

Las etiquetas gramaticales como nombre, verbo, participio, artículo, pronombre, preposición, adverbio y conjunción que se pueden asignar a las palabras dentro de una oración han sido utilizadas por lingüistas desde hace mucho tiempo. La tarea de etiquetado gramatical (llamado también part-of-speech tagging o POS tagging, en inglés), desde el punto de vista computacional, consiste en asignar a un texto las etiquetas mencionadas automáticamente (Voutilainen, 2005).

- **Análisis de dependencias sintácticas**

El análisis de dependencias sintácticas se realiza mayormente a nivel de oraciones, en este se busca describir la estructura gramatical de una oración en términos de las palabras que la componen y de las relaciones sintácticas binarias entre estas (Martin & Jurafsky, 2008).

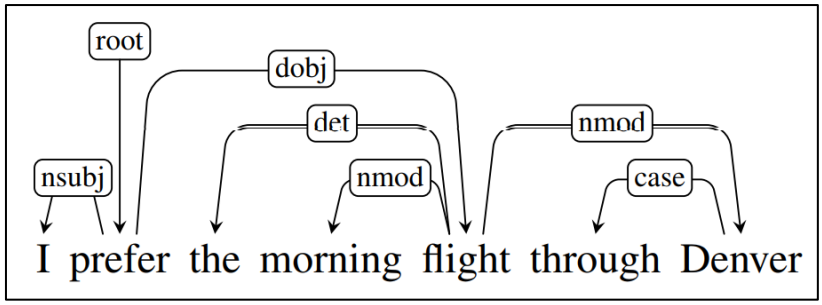

Figura 1: Análisis de dependencias sintácticas de una oración. Adaptado de (Martin & Jurafsky, 2008)

Desde las ciencias de la computación se ha buscado automatizar esta tarea, creando algoritmos y programas que reciban como entrada una oración y devuelvan las relaciones de dependencia encontradas entre las palabras que la conforman (Covington, 2001).

- **Extracción de relaciones**

Las entidades dentro de una oración no se encuentran aisladas, normalmente existe un vínculo entre ellas. Estas relaciones pueden ser de diferentes tipos, y en su mayoría, pueden definirse previamente (como en el caso de la relación de propiedad, empleo, pertenencia, etc.). La tarea de la extracción de relaciones (RE, por sus siglas en inglés) es identificar y extraer automáticamente las relaciones de interés en cada una de las oraciones de un documento, a fin de ayudar con otras tareas dentro del procesamiento de lenguaje natural tales como la extracción de información, sistemas de pregunta/respuesta (*Question Answering*) o lectura de máquina (*Machine Comprehension*) (Pawar, Palshikar, & Bhattacharyya, 2017).

- **Representación vectorial de palabras**

La representación vectorial de palabras permite que estas sean representadas en un espacio continuo. Utilizando métodos de aprendizaje no supervisado sobre grandes conjuntos de datos,

estos modelos capturan aspectos claves de la función y significado de la palabra (Garten, Sagae, Ustun, & Dehghani, 2015).

La habilidad de este tipo de modelos para organizar conceptos y aprender relaciones implícitas entre ellos se puede ver en la Figura número 2, donde se puede visualizar que le modelo ha aprendido la relación de "capital" entre ambos grupos de palabras.

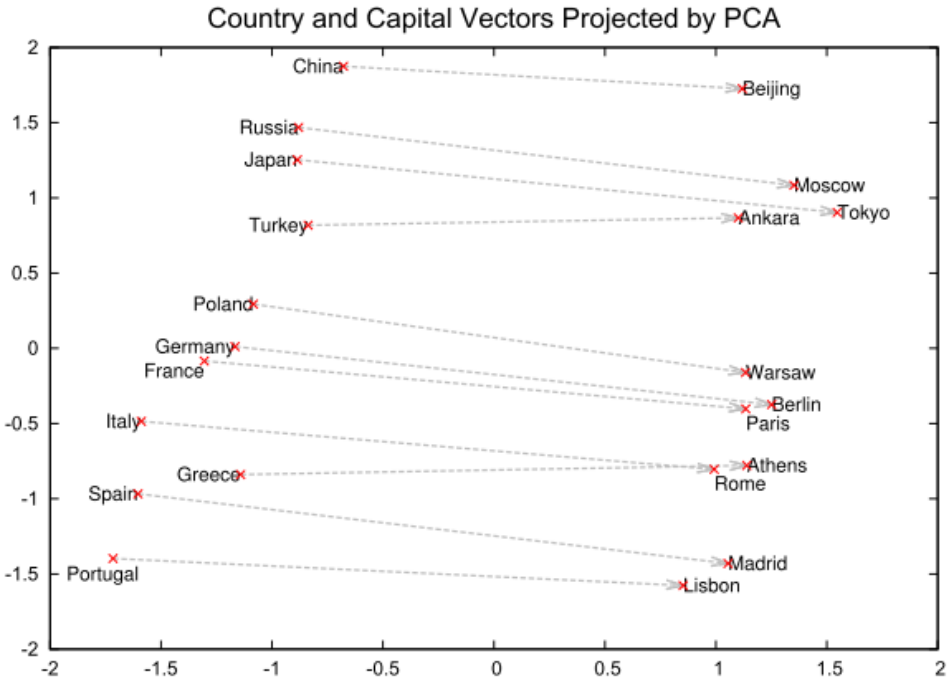

Figura 2: Proyección en 2D de vectores de 1000 dimensiones generados con Skip-gram sobre países y sus capitales (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013)

- **Representación distribuida de documentos**

La mayoría de modelos de aprendizaje de máquina requiere recibir entrada un vector de características. Cuando se trabaja con textos, una de las representaciones más comunes para la representación vectorial de estos es el método de bolsa de palabras. Sin embargo, en el año 2014 se introdujeron los algoritmos de *Paragraph Vector* el cual busca representar cada documento como vector denso de características (Le & Mikolov, 2014).

Para lograr esta representación se proponen dos arquitecturas. La primera, llamada PV-DM *(Paragraph Vector - Distributed memory*) la cual trata de modelar el vector de documento como la información faltante en el contexto actual. Luego, está PV-DBOW *(Paragraph Vector - Distributed bag of words*) donde el vector de documento es modelado tratando de predecir cierta cantidad de palabras pertenecientes al documento. Las Figuras 3 y 4 muestras las arquitecturas mencionadas respectivamente:

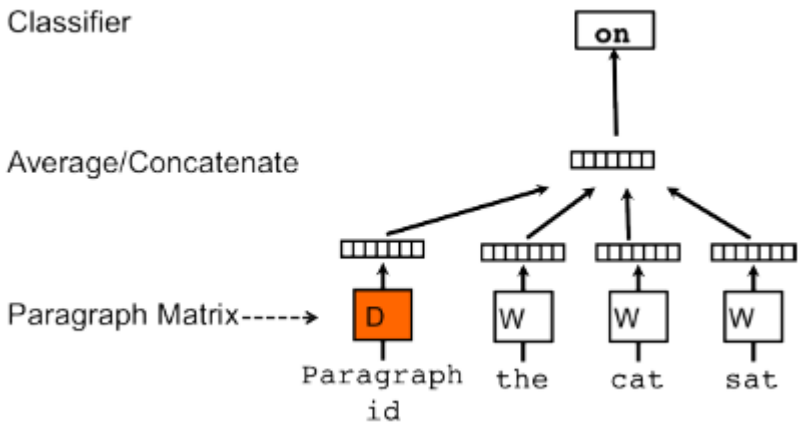

Figura 3: Arquitectura PV-DM para el aprendizaje de vectores de documentos (Le & Mikolov, 2014)

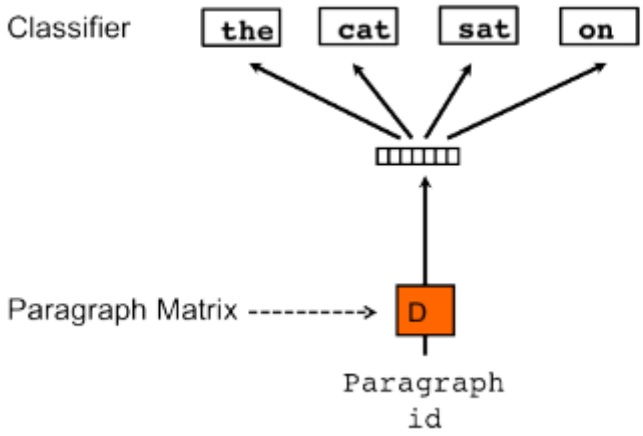

Figura 4: Arquitectura PV-DBOW para el aprendizaje de vectores de documentos (Le & Mikolov, 2014)

- **Métricas de evaluación**

Las métricas de evaluación descritas a continuación sirven para describir el desempeño del nuevo algoritmo implementado en comparación con el definido como línea base.

- **Precisión**

  La precisión se define como el porcentaje de entidades extraídas por el modelo que fueron correctas (Chinchor & Sundheim, 1993).

$$Precisición = \frac{Entidades\ identificadas\ correctamente\ y\ selecionadas}{Cantidad\ total\ de\ entidades\ indentificadas}$$

- **Exactitud**

  Se define como el porcentaje de las entidades correctas que fueron extraídas (Chinchor & Sundheim, 1993).

$$Exactitud = \frac{Entidades\ identificadas\ correctamente\ y\ selecionadas}{Cantidad\ total\ de\ entidades\ indentificadas\ correctamente}$$

- **Medida F**

  La medida F provee una forma de combinar las métricas de precisión y exactitud, la fórmula para calcularla es la siguiente:

$$F = \frac{(\beta^2 + 1.0) \times P \times R}{(\beta^2 \times P) + R}$$

  Donde P es la precisión, R es la exactitud (*Recall*, en inglés) y β es la importancia relativa de R sobre P (Chinchor & Sundheim, 1993).

Las siguientes métricas de evaluación son la utilizadas para analizar las características de esquema, clase y términos de la base de conocimiento.

- **Riqueza de clases**

Esta métrica se relaciona con que tan bien los términos están distribuidos a través de las clases del recurso léxico, si esta es muy cercana a cero indica que el tesauro no tiene términos suficientes para la representación de conocimiento. La fórmula para el cálculo de la riqueza de clase es la siguiente, donde C' es el número de clases con entidades y C número total de clases:

$$CR = \frac{C'}{C}$$

- **Población promedio**

Es definida como la división de I (número de términos en la base de conocimiento) entre C (número de clases). Una población promedio muy baja, indica que hay varias clases sin la cantidad de términos suficientes.

$$P = \frac{I}{C}$$

- **Importancia de clase**

Se calcula como la división de la cantidad de términos que pertenecen a una clase (C_i) entre el total de términos en la base de conocimiento (I).

$$Imp = \frac{C_i}{I}$$

Si se tiene muchas clases con una importancia de clase muy baja, quiere decir que talvez el esquema de la base de conocimiento no es el adecuado.

- **Generación de lenguaje natural**

Para poder generar un texto, un sistema debe ser capaz de determinar qué información incluir y como organizarla. McKeown propuso que los textos siguen ciertos patrones de organización y que estos pueden ser utilizados para la generación de documentos (McKeown, 1985). En el proyecto a desarrollar, se planea utilizar técnicas de generación de lenguaje natural basadas en plantillas para la elaboración de los reportes estructurados.

Cuadro de herramientas y métodos

Tabla 2 Cuadro de objetivos y herramientas a usar (elaboración propia)

Objetivo: Procesar y generar un conjunto de datos estructurado en base a noticias criminales en la web		
Resultado	**Medio de verificación**	**Herramientas y métodos**
Programa de búsqueda y extracción automática de noticias criminales en la web	Pruebas de integridad de datos	- Python - Jupyter - Beautiful Soup - Extracción de noticias de sitios web - MongoDB
Objetivo: Implementar y validar un algoritmo para la extracción automática de información de las noticias.		
Resultado	**Medio de verificación**	**Herramientas y métodos**
Comparación de módulos de software para la extracción de entidades en los textos	Experimentación numérica	- Stanford CoreNLP - SpaCy - Python - Jupyter - Reconocimiento de entidades nombradas
Personalización del módulo de reconocimiento de entidades nombradas	Métricas de precisión y exactitud	- Python - Jupyter - SpaCy

para el dominio y vocabulario específico.		- Reconocimiento de entidades nombradas - Métricas de evaluación
Recurso léxico con términos de dominio criminal	Métricas de esquema y métricas de la base de conocimiento	- Protégé - Hermit - Wordnet - Métricas de evaluación
Programa de generación de lenguaje natural para la emisión de reportes estructurados	Calificaciones humanas	- Python - Jupyter - Métricas de evaluación - Generación de lenguaje natural
Objetivo: Elaborar una interfaz de programación de aplicaciones para la presentación de funcionalidades del modelo desarrollado.		
Resultado	**Medio de verificación**	**Herramientas y métodos**
Interfaz de programación de aplicaciones que utilice el algoritmo implementado para extraer la información importante del texto de una noticia y genere un reporte con esta.	Plataforma web para pruebas de la interfaz de programación de aplicaciones.	- Flask - Django

1.4 Alcance y limitaciones

El presente proyecto de fin de carrera se llevará a cabo utilizando como conjunto de datos noticias de los últimos 4 años extraídas de la sección de "Policiales" para el diario El Comercio; "Asaltos", "Robos", "Feminicidio" y "Asesinatos" en el caso del diario La República y "Policiales/Crímenes" en el diario RPP Noticias. Parte del proyecto será la extracción, filtrado y pre procesamiento de un aproximado de tres mil noticias, para su posterior almacenamiento en un base de datos no relacional.

Una vez almacenado el conjunto de datos, se procederá con la selección de un módulo de reconocimiento de entidades nombradas potencialmente adaptable al dominio y vocabulario específico. Luego, se buscará tunear el módulo seleccionado, utilizando anotaciones manuales hechas en el mes de julio del 2018 sobre el 10% del corpus obtenido.

Posteriormente se construirá un tesauro de dominio criminal, la cual poseerá algunos términos iniciales relacionados al dominio tratado (también llamados *seed terms*) que servirán para popular las clases en base a los documentos presentes en el conjunto de datos. Finalmente usando las entidades detectadas por el módulo de reconocimiento de entidades nombradas, junto los conceptos presentes en el tesauro, se buscará extraer información como la víctima, el culpable, la locación y tipo de crimen de la descripción de un delito para la generación de un reporte estructurado.

En base al modelo de extracción de información generado se creará una interfaz de programación de aplicaciones capaz de recibir el texto de una noticia, procesarla y devolver como resultado la información extraída automáticamente junto con el texto generado con esta. Por último, se elaborará una plataforma web para las pruebas de dicha interfaz.

1.5 Viabilidad

1.5.1 Viabilidad técnica

Los lenguajes y herramientas a utilizar durante el desarrollo del proyecto son de acceso libre. Además, quien suscribe posee experiencia en su uso.

1.5.2 Viabilidad temporal

Se estima una duración de 6 meses para el desarrollo de las actividades del presente proyecto. En el Anexo 1 se presentan las tareas a realizar y el tiempo estimado para la ejecución de cada una.

1.5.3 Viabilidad económica

Debido a que las herramientas, lenguajes y datos a utilizar son de libre acceso, no existen limitaciones financieras ni de accesibilidad para el proyecto presentado.

1.6 Alcance limitaciones y riesgos

La tabla presentada a continuación muestra los riesgos identificados en el proyecto, así como las medidas de mitigación y contingencia a ejecutar en caso lleguen a materializarse.

Tabla 3. Riesgos identificados en el proyecto (elaboración propia)

Descripción	Síntomas	Probabilidad	Impacto	Severidad	Mitigación	Contingencia
Bloqueo del acceso a las páginas web de periódicos de donde se extraerán las noticias	El extractor automático de noticias puede ser confundido con software malicioso por las políticas de	0.5	0.7	0.6	Puede utilizarse noticias de otros periódicos peruanos publicados en internet	Puede hacerse la extracción de noticias desde una IP diferente, utilizando máquinas virtuales

	seguridad del sitio					

Capítulo 2. Marco Conceptual

A continuación, se hace una breve descripción de los conceptos teóricos que se utilizarán para abordar el problema presentado y en el desarrollo del presente proyecto. Se mencionan definiciones sobre criminología y ciencias de la computación.

2.1 Objetivos del marco conceptual

Con el presente marco conceptual se busca definir los conocimientos necesarios para poder comprender los problemas expuestos previamente, así como los términos y soluciones del área de extracción de información mencionadas.

2.2 Conceptos sobre criminología

2.2.1 Análisis criminal

Proceso en el cual se utiliza un conjunto de técnicas cuantitativas y cualitativas con el fin de analizar datos valiosos para las agencias policiales y sus comunidades. El análisis criminal incluye el análisis de crimen y criminales, víctimas de delitos y tráfico de personas. Sus resultados están orientados a apoyar las investigaciones criminales y acusaciones, actividades de patrullaje, las estrategias de reducción y prevención del crimen, y la evaluación de los esfuerzos de la policía (LeBlanc et al., 2014).

2.3 Conceptos de ciencias de la computación

2.3.1 Procesamiento del lenguaje natural

También llamado lingüística computacional, es la rama de las ciencias de la computación que se encarga del uso computacional de técnicas para aprender, entender y producir contenido en lenguaje humano (Hirschberg & Manning, 2015).

Las técnicas de procesamiento del lenguaje natural pueden ser muy útiles para las fuerzas defensoras de la ley, particularmente en el caso donde grandes cantidades de data no estructurada debe ser procesada por investigadores criminales. Técnicas como el resumen

automático de textos, extracción de información y extracción de entidades son de gran interés por las agencias de defensa de la ley y de inteligencia (van Banerveld et al., 2014).

2.3.2 Macro datos

También llamado *Big Data*, en inglés, el término es utilizado para referirse a la cantidad de datos más allá de las herramientas, técnicas y tecnologías para el almacenamiento, administración y procesamiento eficiente existentes (Kaisler, Armour, Espinosa, & Money, 2013). Actualmente, la mayoría de departamentos de policía y agencias de inteligencia recolectan volúmenes extraordinarios de datos de diferentes fuentes, los cuales deben ser procesados y convertidos en información beneficiosa para la seguridad de los ciudadanos rápidamente (Pramanik et al., 2017).

2.3.3 Minería de datos

La minería de datos es una forma de extraer conocimiento de conjuntos de datos generalmente grandes. En otras palabras, es un enfoque para descubrir relaciones ocultas entre datos mediante el uso de métodos de aprendizaje de maquina (Keyvanpour, Javideh, & Ebrahimi, 2011). Implica definir algoritmos que exploren los datos y el desarrollo de un modelo que sirva para identificar patrones, así como para su análisis y predicción (Maimon & Rokach, 2005).

En lo relacionado a crímenes, uno de los desafíos de las agencias de inteligencia y el cumplimiento de la ley es el análisis de las grandes cantidades de datos involucradas en la actividad criminal. La minería de datos, sostiene la promesa de hacer fácil, conveniente y práctica la exploración de grandes bases de datos para organizaciones y usuarios (Chen et al., 2002).

2.3.4 Extracción de información

Proceso de extraer hechos relevantes de un texto y representarlos de forma útil. Los "hechos relevantes" están restringidos a aquellos que están explícitamente presentes en el texto.

Además, el rango de hechos de interés se especifica de antemano y se limitará a un pequeño número de eventos y relaciones aplicado a las entidades en el texto (Appelt, 1999).

Un ejemplo de la aplicación de este proceso en el campo del análisis de crímenes es el Sistema de Reporte de Crímenes Online desarrollado por Ku, Iriberri, & Leroy (2006), donde se buscaba facilitar la obtención de reportes hechos por testigos y víctimas de un crimen. El sistema contaba con un módulo de extracción de información para poder extraer información relaciona al crimen reportado automáticamente, logrando así facilitar y agilizar la tarea de los investigadores.

Capítulo 3. Estado del Arte

Esta sección busca mostrar los algoritmos, técnicas, y resultados de trabajos anteriores en el campo de la extracción de información en base a noticias criminales extraídas de la web, a fin de contextualizar la investigación a realizar.

3.1 Estrategia de búsqueda

3.1.1 Palabras clave

Para la búsqueda de investigaciones anteriores se utilizaron las palabras clave listadas a continuación, según la relación entre ellas:

- *Crime, Crime analysis*
- *News*
- *Text, Language*
- *Information extraction*
- *Entities*

3.1.2 Cadenas de búsqueda

En base a la combinación de las palabras clave, se obtuvieron las siguientes tres cadenas de búsqueda con resultados significativos:

- *TITLE-ABS-KEY (crime AND news AND (text OR language)) AND (LIMIT-TO (SUBJAREA , "COMP"))*
- *TITLE-ABS-KEY ("information extraction" AND crime AND news)*
- *TITLE-ABS-KEY (crime AND news AND entities) AND (LIMIT-TO (SUBJAREA , "COMP"))*

3.1.3 Preguntas de revisión

En la revisión de los artículos presentados a continuación, se buscaba responder a las siguientes preguntas:

- *Pregunta 1: ¿Cuál es la motivación detrás del uso de noticias como conjunto de datos para una investigación?*
- *Pregunta 2: ¿Qué técnicas o enfoques se utilizaron previamente para la extracción de información de noticias criminales?*
- *Pregunta 3: ¿Qué tipo de entidades extrajeron los trabajos previos realizados con noticias criminales?*
- *Pregunta 4: ¿De qué forma se validaron los resultados obtenidos?*
- *Pregunta 5: ¿Qué resultados se obtuvieron con las técnicas y enfoques utilizados?*

3.2 Revisión y discusión

En esta sección se presenta la revisión de investigaciones recientes desde diferentes técnicas de extracción de información en el área de procesamiento de lenguaje natural. Se utilizó como principal buscador Scopus y la búsqueda se realizó en los meses de abril y mayo del año 2018.

CrimeProfiler: Crime Information Extraction and Visualization from News Media (Dasgupta, Naskar, Saha, & Dey, 2017)

En el presente trabajo se busca extraer y clasificar entidades dentro de las noticias relacionadas a crímenes. Se argumenta a favor de este enfoque que, si bien es común que las agencias defensoras de la ley y ciudadanos tengan amplios conocimientos sobre la actividad criminal en su propia localidad, una base de conocimiento con información extraída de fuentes abiertas (como es el caso de las noticias) puede ayudar a conocer la actividad criminal en otros lugares. La investigación buscaba extraer por cada noticia la siguiente información: nombre del criminal, nombre de la víctima, naturaleza del crimen, ubicación geográfica, sanción contra el criminal, fecha y hora. Para la extracción de potenciales entidades nombradas se utilizó el *Stanford Named Entity Recognizer,* el cual manejaba las siguientes clases: persona,

organización, locación, fecha, hora, dinero y porcentaje. Del mismo modo, para la ayuda a la clasificación de estas entidades se utilizó una ontología con conceptos relacionados a crímenes. Finalmente, para la validación de resultados se seleccionaron mil documentos aleatoriamente para la anotación de expertos, donde se debía identificar los siguientes cuatro factores relacionados a un crimen: naturaleza del crimen, nombre del acusado, ubicación y fecha. Los resultados obtenidos se muestran en la siguiente tabla:

Tabla 4. Evaluación de los clasificadores de factores de un crimen. Los valores se encuentran en porcentajes (Dasgupta et al., 2017)

	Nombre del acusado	Naturaleza del crimen	Ubicación	Fecha
Precisión	93	71	89	87
Exactitud	49	74	87	88
Medida F1	64	72	88	87

An Interactive Malaysia Crime News Retrieval System (Mohd & Ali, 2011)

En esta investigación se discuten las técnicas usadas para la implementación del *Interactive Malaysia Crime News Retrieval System (i-JEN)*, el cual tiene como fin organizar, recuperar información y presentar las noticias sobre crímenes de forma más efectiva e interactiva. La utilidad del proyecto se justifica en el hecho que el aumento del crimen es uno de los principales problemas en Malasia. Además, considerando que la información oficial sobre crímenes es difícil de obtener, gracias a políticas de privacidad y accesibilidad, las noticias son entonces uno de los principales recursos para los ciudadanos y medios de comunicación en el rastreo y monitoreo de actividad criminal.

El sistema mencionado utiliza modelos de espacio vectorial para representar las noticias, asignándole diferentes pesos a cada palabra y particularmente un peso adicional a las entidades

nombradas identificadas por cada noticia. No se muestran los resultados obtenidos en este proceso de extracción de información.

Crime analytics: Analysis of crimes through newspaper articles (Jayaweera et al., 2015)

Se propone el desarrollo de un sistema de análisis de crímenes basado en noticias publicadas en la web. Como principal motivación para el uso de noticias está el hecho que en Sri Lanka (país donde se desarrolla el sistema) todavía se utiliza un sistema de manual de registro y análisis de crímenes. Además, también es objetivo de la investigación brindar un sistema de mapeo de crímenes abierto, ya que los existentes hasta ahora han sido desarrollados especialmente para departamentos de policía y no están disponibles para un público general. Como conjunto de datos inicial se utilizan noticias de los periódicos en ingles más populares de Sri Lanka: *Daily Mirror, The Island y Caylon Today.*

Inicialmente, el sistema clasifica las noticias entre "Noticia relacionada al crimen" y "Noticia no relacionada" utilizando una máquina de vectores de soporte (SVM, por sus siglas en inglés). Después de esta clasificación, se busca extraer las siguientes entidades relevantes de cada noticia: fecha, ubicación, policías involucrados y número de víctimas. Para esto se utiliza diferentes aplicaciones ya elaboradas dentro del software GATE (*General Architecture for Text Engineering*), particularmente la extracción de información se propone un enfoque híbrido utilizando la combinación de los etiquetadores gramaticales ANNIE y el *Stanford POS tagger*. Finalmente, para la validación de la extracción de entidades hay dos resultados. Primero, la comparación del enfoque híbrido usado para la extracción de información frente a otros enfoques como el de aprendizaje de máquina o el basado en contexto; luego, se muestran las medidas de nivel de acierto obtenidas para la extracción de entidades del tipo fecha, ubicación y tipo de crimen.

Tabla 5. Comparación del rendimiento entre el enfoque híbrido, aprendizaje de máquina y basado en contexto. Los valores se encuentran en porcentajes (Jayaweera et al., 2015)

	Híbrido	Aprendizaje de máquina	Basado en contexto
Precisión	82.93	53.84	91.1
Exactitud	80.95	43.75	10
Medida F1	|81.93	48.28	18.02

Tabla 6. Evaluación de entidades de tipo y fecha de crímenes. (Jayaweera et al., 2015)

Categoría de la entidad	Nivel de acierto (%)
Lugar del crimen	82
Tipo de crimen	82
Fecha del crimen	74

Extracting Crime Information from Online Newspaper Articles (Arulanandam et al., 2014)

Los autores buscan extraer información sobre locaciones de crímenes, utilizando noticias sobre crímenes publicadas en periódicos en internet. La utilidad del proyecto se justifica en el hecho que, si bien esta información es conocida por la policía, en muchos países no se hace pública, dejando a los ciudadanos sin este tipo de conocimiento importante. Por esta razón, se busca extraer este tipo de información "escondida" sobre la ubicación de un crimen en una fuente de información ya pública, como lo son los artículos periodísticos.

Para propósitos de la investigación desarrollada, los autores se concentran en un solo tipo de crimen, en este caso robo, y en extraer un solo tipo de información, la ubicación. Primero se extraen las locaciones en general de las noticias recolectadas, para realizar esta tarea se comparan 4 tipos diferentes de algoritmos de extracción de entidades nombradas, utilizando un conjunto de datos de 70 noticias del periódico *Ontago Daily Times*. La validación de estos

algoritmos se realizó utilizando un subconjunto de 50 noticias anotadas del conjunto de datos original, obteniéndose los siguientes resultados:

Tabla 7. Comparaciones de cuatro algoritmos NER diferentes basados en la identificación de ubicación realizada por Arulanandam et al (Arulanandam et al., 2014)

Algoritmo	Precisión	Exactitud	Medida F	Nivel de acierto
NLTK pre-trained named entity chunker	0.93	0.78	0.85	0.74
Stanford NER	0.93	0.80	0.86	0.76
NLTK Chunkparser using Gazateer	0.88	0.91	0.90	0.81
LBJ Tagger	0.98	0.96	0.97	0.94

Posteriormente, se buscan características en las oraciones que contienen las ubicaciones identificadas con el algoritmo que obtuvo mayor nivel de acierto, en este caso *LBJ Tagger,* en el paso anterior. La detección de estas características servirá para etiquetar las oraciones en dos tipos: oración con ubicación de un crimen (CLS, por sus siglas en inglés) y oración sin la ubicación de un crimen (NO-CLS, por sus siglas en inglés).

3.3 Conclusiones

Luego de haber revisado y analizado los estudios más relevantes y con mejores resultados, se puede concluir lo siguiente:

- La principal motivación para el uso de noticias y no información oficial de la policía como datos de entrada es el querer dejar como aporte de la investigación un conjunto de datos sobre crímenes, sea reportes estructurados o resúmenes, más accesible al público general. Esto mayormente se da en países donde la policía no dispone de información digitalizada o no la hace pública, por lo tanto, se considera que es un proyecto completamente aplicable a la realidad peruana.

- Los tipos de información extraída más frecuentes son tipo de crimen y ubicación. Dejando de lado otros tipos de datos que podrían ser relevantes tales como: los actores involucrados en la descripción de un delito, es por esta razón, que el presente proyecto buscará extraer este tipo de información.

- Si bien las técnicas de extracción de entidades nombradas son útiles en este proceso, estas por si solas no son suficientes para la extracción de información. Se requiere de la ayuda de otros métodos y técnicas para poder decir que se posee información valiosa, por lo tanto, en esta investigación se propone el uso de recursos léxicos (como es un tesauro) y el análisis del árbol de dependencias sintácticas.

Capítulo 4. Procesamiento y generación de un conjunto de datos estructurado en base a noticias criminales publicadas en internet.

4.1 Introducción

En el presente capítulo se muestra el desarrollo del objetivo número 1, el cual tiene como propósito la generación de un conjunto de datos estructurado en base a noticias peruanas relacionadas al dominio del crimen. A fin de lograr este objetivo, primero se desarrolló un componente de software para la extracción automática de textos de páginas web y se creó una base de datos documental para el almacenamiento de los textos recolectados. Posteriormente, se detectó que, si bien todas las noticias obtenidas se relacionaban al ámbito criminal o policial, algunas de estas no contenían la descripción de un delito propiamente, por lo que fue necesario la creación de un módulo complementario para el filtrado de documentos.

4.2 Descripción del objetivo

El conjunto de datos generado posee noticias publicadas durante los cuatro últimos años en los portales web de diarios peruanos más visitados: El Comercio, La República y RPP (Radio Programas del Perú). Además del título, resumen y texto de cada noticia, se guardan otros atributos como el URL, fecha de publicación y etiquetas relacionadas (asignadas por los autores) que podrían ser útiles posteriormente. Además, se realizó y se muestran los resultados del análisis exploratorio del conjunto de datos obtenido.

4.3 Desarrollo del objetivo

4.3.1 Programa de extracción y recolección de noticias

El programa descrito a continuación se implementó debido a que obtener manualmente una cantidad tan grande de noticias demandaría mucho tiempo. Para la elaboración de este se utilizó la librería BeautifulSoup4 y se tomó en cuenta la estructura particular de cada página, por lo tanto, este es exclusivo para el archivo y las noticias publicadas en cada uno de los portales web seleccionados.

- El Comercio

La página web del diario el comercio (www.elcomercio.pe) divide las noticias que publica en diferentes secciones, para fines de este proyecto la sección de interés identificada es "Policiales" en la cual se encontraron noticias hasta el 16 de mayo del 2017. Las direcciones URL correspondientes al archivo de esta sección tienen el siguiente formato:

www.elcomercio.pe/archivo/lima/policiales/<año>-<mes>-<día>

Tomando en cuenta esta estructura, las noticias policiales correspondientes al 13 de diciembre del año 2017 se encontrarán en la siguiente dirección:

https://elcomercio.pe/archivo/lima/policiales/2017-12-13

Para la extracción, primero se creó una función que identifique y obtenga todos los links a noticias publicadas para una fecha en el archivo de la sección de policiales. Una vez obtenidos, cada link se mandaba como parámetro a otra función con el fin de extraer la información de la noticia.

Después de la creación de las funciones mencionadas, bastó con iterar desde una fecha de inicio hasta una final para obtener todas las noticias de la sección de policiales.

- La República

En el caso de la página web del diario La República (https://larepublica.pe) las noticias se clasifican por etiquetas, siendo de nuestro interés las siguientes: “Asaltos”, “Robos”, “Feminicidio” y “Asesinatos”. Las URL de las secciones mencionadas tiene la siguiente estructura:

www.larepublica.pe/tag/<nombre de la etiqueta>/page/<número de página>

De este modo, la dirección para la página número 15 de la etiqueta “Asesinatos” sería la siguiente:

https://larepublica.pe/tag/asesinatos/page/15

Dentro del proceso de extracción, primero se creó una función que obtenga las direcciones a las noticias completas de los titulares contenidos en cada página. Luego, se implementó una función que reciba una dirección a una noticia completa y extraiga la información de esta. Finalmente, se creó un bucle que recorra cada página de un *tag* específico, y se repitió el proceso para cada *tag* seleccionado.

- RPP

Para la página web del diario RPP (Radio Programas del Perú) (https://rpp.pe/) se identificó como sección de interés “Policiales/Crímenes”. Sin embargo, acceder a un histórico exclusivo de las noticias de esta sección no es posible, razón por la cual se utilizará como URL base la dirección al archivo general del diario RPP. A partir de ahí, se descartarán noticias de otras secciones, con el fin de extraer solamente las que pertenezcan a la de “Policiales/Crímenes”.

La estructura del URL correspondiente al archivo de noticias del diario RPP es la siguiente:

www.rpp.pe/archivo/lima/<año>-<mes>-<día>

Siguiendo este esquema, si queremos las noticias del día 8 de marzo del 2018, la dirección a utilizar sería:

https://rpp.pe/archivo/lima/2018-03-08

A fin de obtener las noticias, se creó una función que identifique los links y las secciones de cada titular dentro de una dirección del archivo. De este modo se descartaron aquellas direcciones que no pertenecieran a la sección “Policiales/Crímenes”. Posteriormente, y de forma similar a los dos periódicos anteriores, se creó una función para que, dada la dirección de una noticia, obtuviera la información de esta de forma automática.

Para obtener todas las noticias en el archivo, solo fue necesario repetir la ejecución de las funciones mencionadas arriba desde una fecha de inicio hasta una final pasadas como parámetros.

4.3.2 Filtrado de noticias

Como se mencionó anteriormente, si bien todas noticias extraídas pertenecen al dominio del crimen, no todas contienen exactamente la descripción de un delito, sino que son una crónica de alguna secuencia de hechos o describen una problemática. Es por esta razón que se implementó este componente de filtrado de noticias, capaz de detectar los valores atípicos o *outliers* presentes en el conjunto de datos utilizando el método de representación distribuida de documentos (Le & Mikolov, 2014), particularmente la implementación de la librería gensim. La secuencia de pasos seguida se muestra en la Figura número 5 y se detalla a continuación:

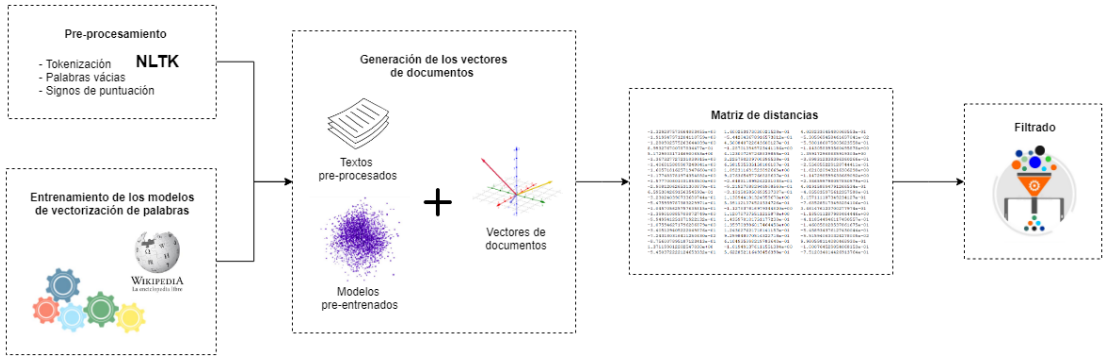

Figura 5: Flujo de trabajo para el filtrado de valores atípicos en los documentos (elaboración propia)

Primero, se tuvo que crear una función para el pre-procesamiento de las noticias recolectadas utilizando la librería nltk. Esta función divide el texto de una noticia en palabras individuales, convierte cada palabra a letras minúsculas y filtra signos de puntuación y palabras vacías. Una vez que se tenían los textos pre-procesados, se infirieron los vectores correspondientes a estos haciendo uso de un modelo de vectorización de documentos generado de la concatenación de los modelos de *distributed memory* y *distributed bag of words*. Los modelos mencionados

fueron entrenados previamente con información de wikipedia en español para asegurar el mantenimiento del contexto al convertir el documento a vector.

Posteriormente se calcula el vector medio de todos los vectores de documentos obtenidos y utilizando la implementación de la métrica similitud coseno de la librería sklearn se mide la distancia de cada uno de los vectores del modelo al vector media. De esta forma se obtiene una matriz de distancias, junto con la media y la desviación estándar de esta. Finalmente, las distancias obtenidas son utilizadas para filtrar aquellos documentos que se encuentren a más de una desviación estándar del vector media.

La distribución de distancias respecto al vector media para los tres periódicos existentes en el conjunto de datos se puede visualizar en las Figuras 6, 7, y 8. Del mismo modo, los resultados con la cantidad de *outliers* detectados en cada periódico se presentan en la Tabla 8 presentada a continuación:

Tabla 8. Resultados del filtrado de noticias utilizando el método de representación distribuida de documentos (elaboración propia)

Diario	**Cantidad original de noticias**	**Cantidad de valores atípicos**	**Cantidad de noticias después del filtrado**
El Comercio	1423	217 (15.24%)	1206
La República	3605	574 (15.92%)	3031
RPP	1011	130 (12.85%)	881

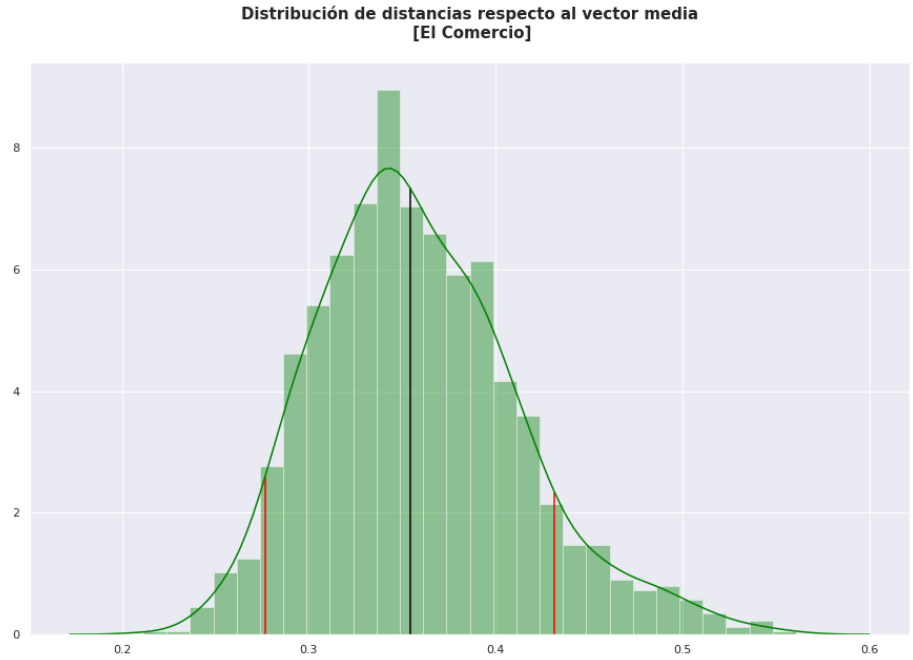

Figura 6: Distribución de distancias respecto al vector media para el diario El Comercio (elaboración propia)

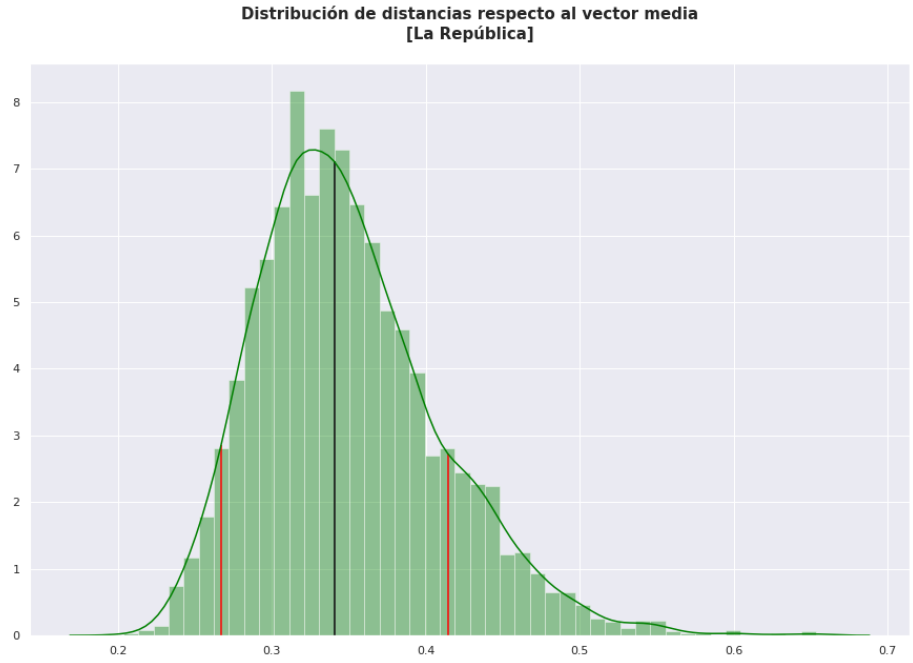

Figura 7: Distribución de distancias respecto al vector media para el diario La República (elaboración propia

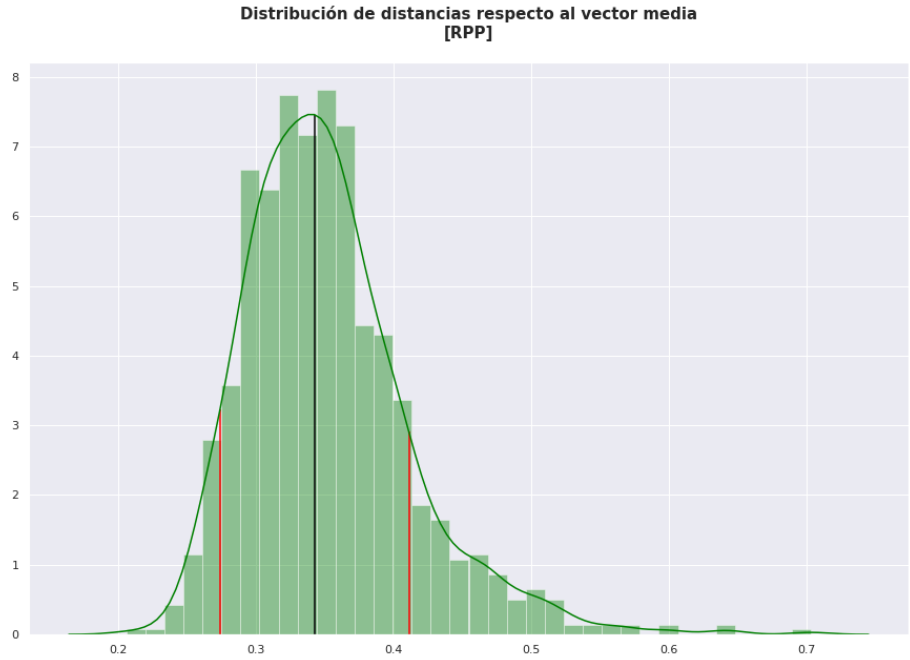

Figura 8: Distribución de distancias respecto al vector media para el diario RPP (elaboración propia)

4.3.3 Base de datos con los textos recolectados

Para el almacenamiento de las noticias recolectadas y textos pre-procesados se construyó una base de datos siguiendo el esquema mostrado en la Figura número 9, sobre una instancia levantada utilizando el gestor de base de datos MongoDB.

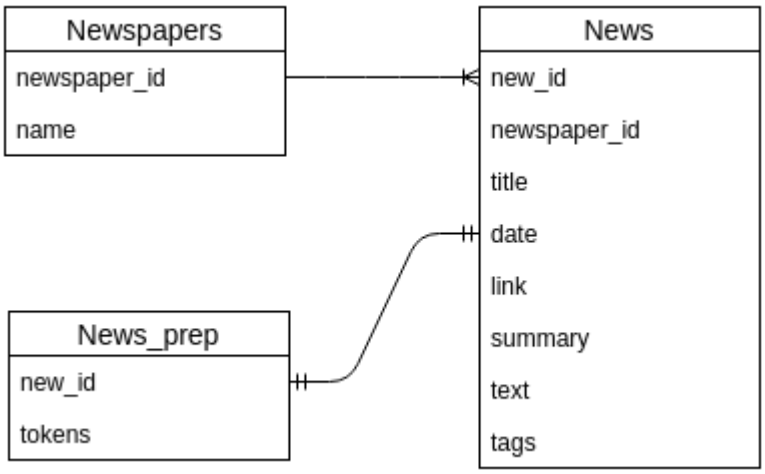

Figura 9: Esquema de la base de datos para el almacenamiento de los documentos recolectados (elaboración propia)

Cada noticia posee como identificador un id creado al momento de insertarla a la base de datos y tiene asociado el id del periódico de donde se extrajo. Además, se guardan los atributos relacionados directamente al análisis de textos: título, resumen, texto; junto con otros como fecha, link y etiquetas que se pueden utilizar para otros propósitos.

Por otro lado, en una colección aparte se guarda el texto de la noticia ya dividido en palabras individuales, a fin de agilizar el procesamiento de diferentes tareas que se realizarán a futuro.

4.3.4 Análisis exploratorio de datos

En el análisis exploratorio del conjunto de datos obtenido se observó la cantidad de palabras por cada noticia, a fin de conocer el tamaño de los textos con los que se está trabajando. En las Figuras 10, 11 y 12 presentadas a continuación; se muestra la cantidad de noticias por número de palabras (sin considerar palabras vacías o *stopwords*) para los tres periódicos a los que pertenecen los documentos recolectados.

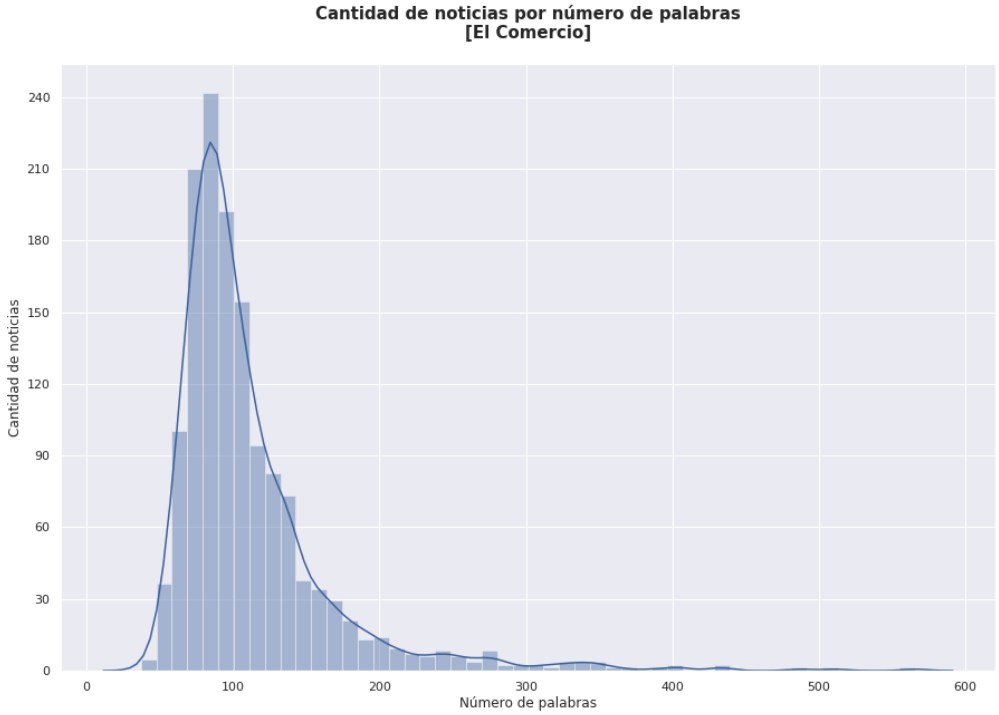

Figura 10: Cantidad de noticias por número de palabras para el diario El Comercio (elaboración propia)

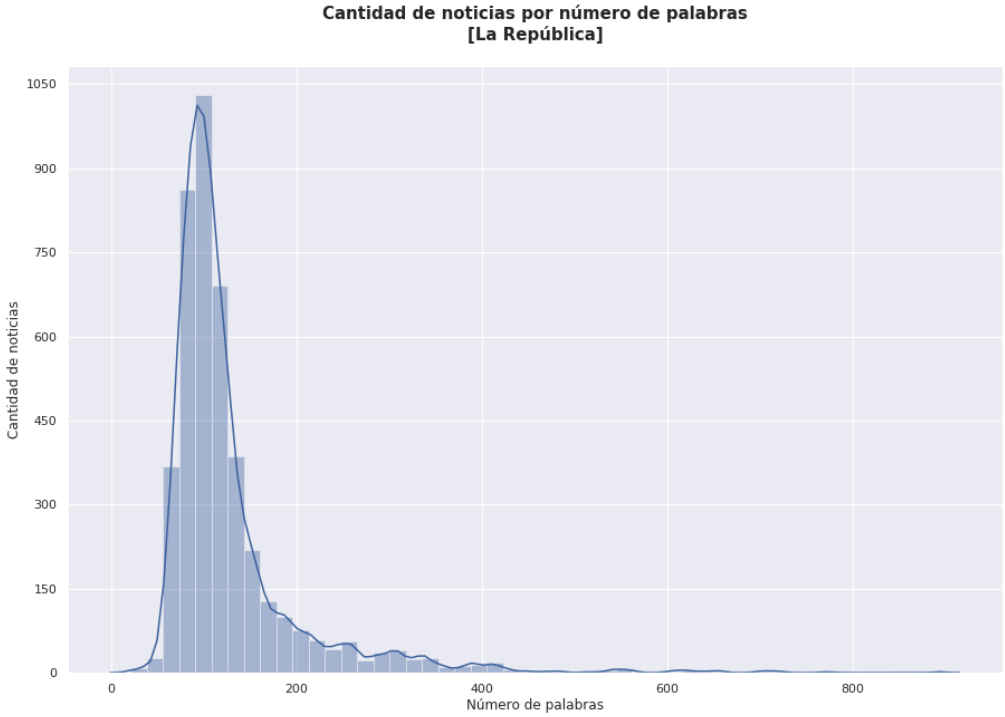

Figura 11: Cantidad de noticias por número de palabras para el diario La República (elaboración propia)

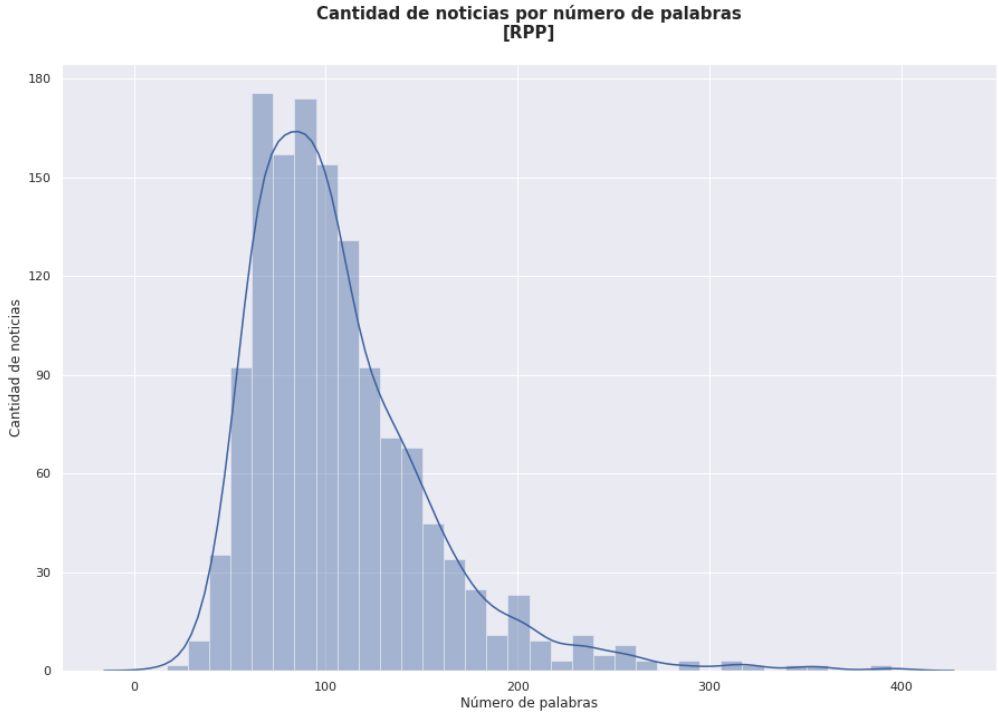

Figura 12: Cantidad de noticias por número de palabras para el diario RPP (elaboración propia)

Como se puede ver, la gran mayoría de las noticias en los tres diarios utilizados tienen aproximadamente entre 100 y 120 palabras. Por otro lado, es interesante también observar que mientras El Comercio y La República tienen gráficas bastante parecidas, RPP presenta una distribución mucho más uniforme. Si consideramos junto a esta observación que en el filtrado de documentos RPP fue el periódico que menos valores atípicos presentaba, se podría inferir cierta relación entre la cantidad de palabras y la probabilidad de que un documento no contenga la descripción de un delio propiamente.

Capítulo 5. Comparación de módulos de software para la extracción de entidades en los textos

5.1 Introducción

En este capítulo se explicará el desarrollo del resultado número 2. En el cual se busca comparar los módulos de reconocimiento de entidades nombradas (NER, por sus siglas en inglés) populares entre las investigaciones actualmente, a fin de seleccionar el mejor para el dominio y vocabulario específico del presente proyecto. En la revisión del estado del arte se encontró que uno de los módulos más utilizados es la librería CoreNLP desarrollada por el grupo de procesamiento de lenguaje natural de la Universidad de Stanford (Manning et al., 2014). En el presente capítulo, esta librería es comparada contra la recientemente publicada librería para procesamiento avanzado de lenguaje natural SpaCy (spaCy, 2018).

5.2 Descripción del resultado

Se seleccionó una muestra representativa del 10% del conjunto de datos para anotaciones manuales. Estas fueron utilizadas para obtener las métricas de precisión, exactitud y medida F1 de los módulos de NER pertenecientes a las librerías mencionadas. Posteriormente, mediante la técnica *bootstrap sampling,* se recolectaron 30 muestras de la medida F1 obtenida por cada módulo sobre 30 particiones aleatorias del conjunto de datos anotados. Finalmente, se realizó la experimentación numérica correspondiente para determinar cuál de los dos módulos tiene un mejor despeño para el dominio y vocabulario específico de los textos utilizados en el presente proyecto. La secuencia de pasos descrita, se muestra gráficamente en la figura número 13.

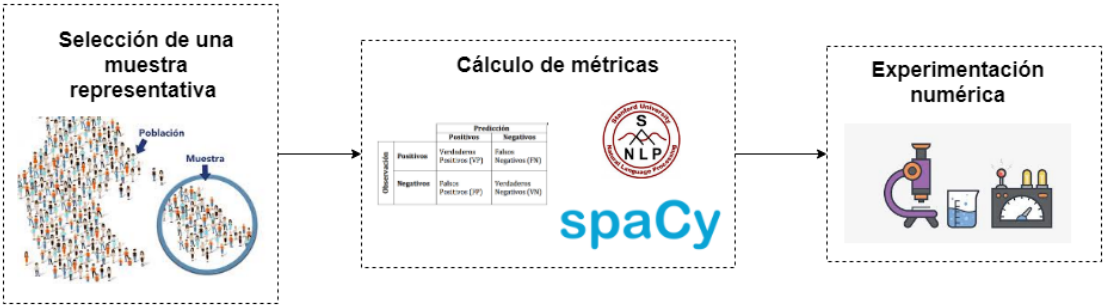

Figura 13: Flujo de trabajo a seguir para la comparación de módulos de reconocimiento de entidades nombradas (elaboración propia)

5.3 Desarrollo del resultado

5.3.1 Selección de la muestra representativa del conjunto de datos

Los métodos descritos a continuación se utilizaron a fin de asegurar que la muestra extraída para las anotaciones manuales fuera lo más representativa posible en cuando a diversidad de vocabulario, y de esta manera garantizar que las métricas obtenidas en los pasos siguientes fueran certeras.

- Algoritmo de agrupamiento basado en densidad

Utilizando la concatenación de los modelos de memoria distribuida y bolsa de palabras distribuida para la vectorización de documentos, se crearon vectores de 600 dimensiones para cada una de las noticias en el conjunto de datos. Luego, mediante la técnica estadística de análisis de componentes principales (PCA, por sus siglas en inglés) se redujo el número de dimensiones de los vectores obtenidos a 300 y se obtuvo la matriz de distancias usando la métrica de similitud coseno. Finamente se agruparon los nuevos vectores obtenidos mediante el algoritmo OPTICS (*Ordering points to identifico the clustering structure*), teniendo como parámetros: 50 en la mínima cantidad de puntos en un vecindario y 0.8463 como distancia máxima entre un núcleo y un dato perteneciente al vecindario. Los resultados obtenidos se muestran en la Figura número 14.

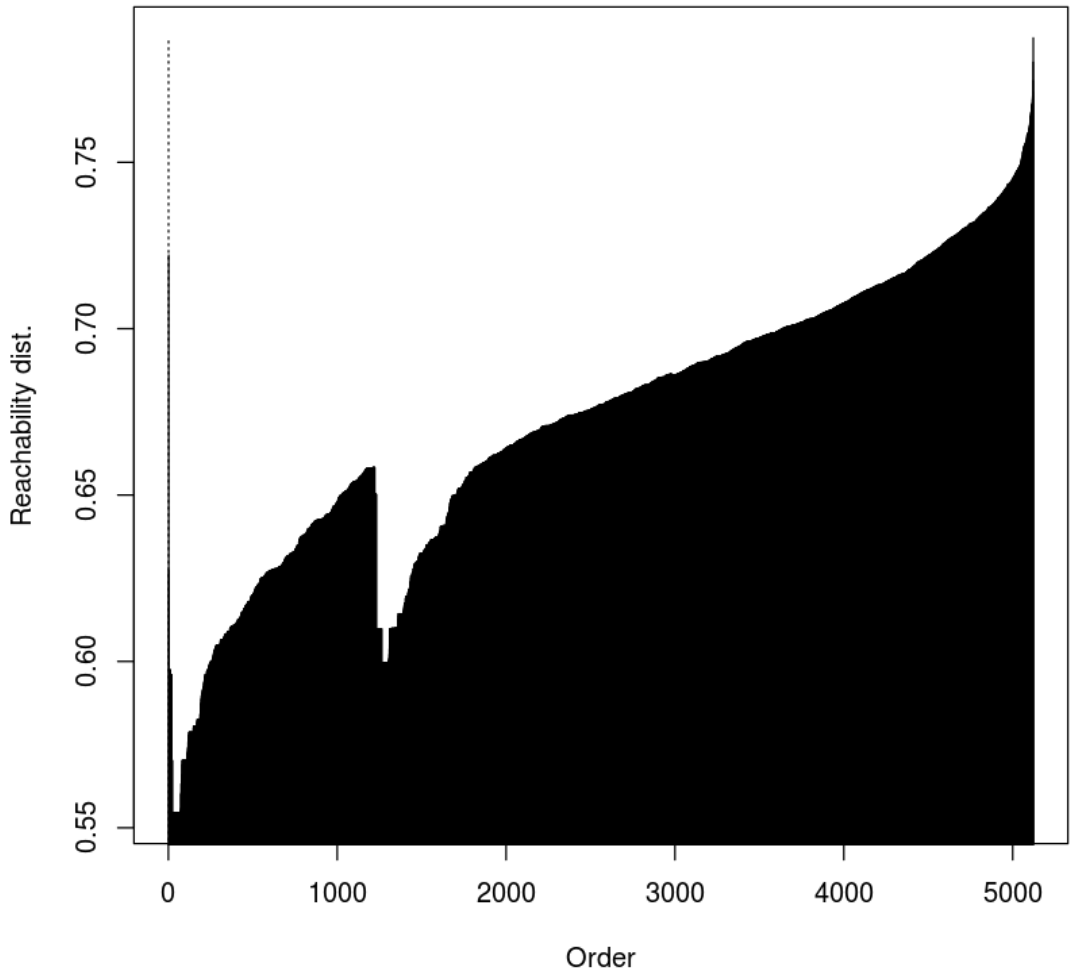

Figura 14: Gráfica de distancias de cada elemento a un núcleo de vecindario (elaboración propia)

Aunque no tan marcada como se esperaba, se puede visualizar una depresión en las distancias a los núcleos de vecindario, razón por la cual se decidió considerar la existencia de dos grandes grupos de datos utilizados en la experimentación futura.

- Experimentación y contraste contra una muestra aleatoria

Se realizaron 1000 iteraciones, en las cuales se extraían 500 documentos de todo el conjunto de datos de dos formas diferentes: primero, se consideró como línea base una extracción de forma totalmente aleatoria; segundo, tomando en cuenta los agrupamientos obtenidos con

OPTICS, se seleccionaban documentos de cada uno de estos de forma proporcional a su tamaño.

En cada iteración se calculó el tamaño de vocabulario, junto con la varianza y la desviación estándar de los vectores de palabras pertenecientes a ese vocabulario. Los resultados obtenidos se muestran a continuación:

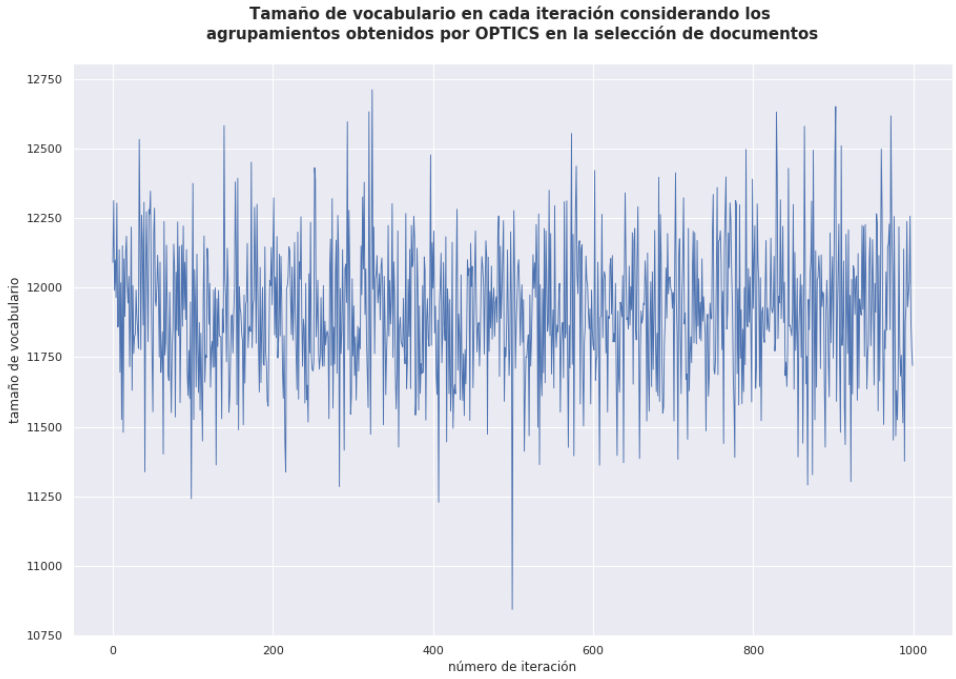

Figura 15: Tamaño de vocabulario en cada iteración considerando los agrupamientos obtenidos por OPTICS en la selección de documentos (elaboración propia)

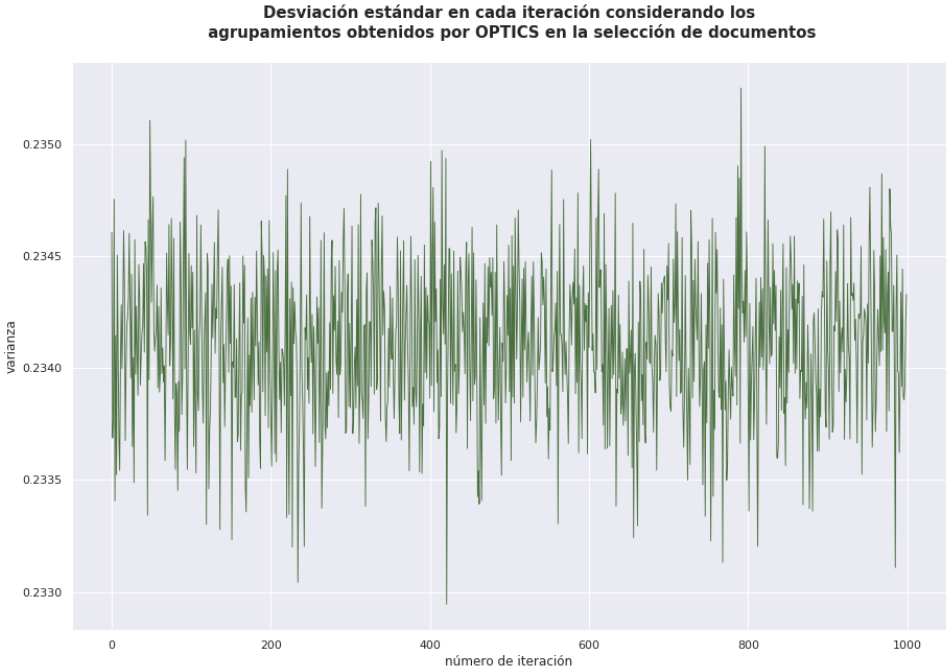

Figura 16: Desviación estándar en cada iteración considerando los agrupamientos obtenidos por OPTICS en la selección de documentos (elaboración propia)

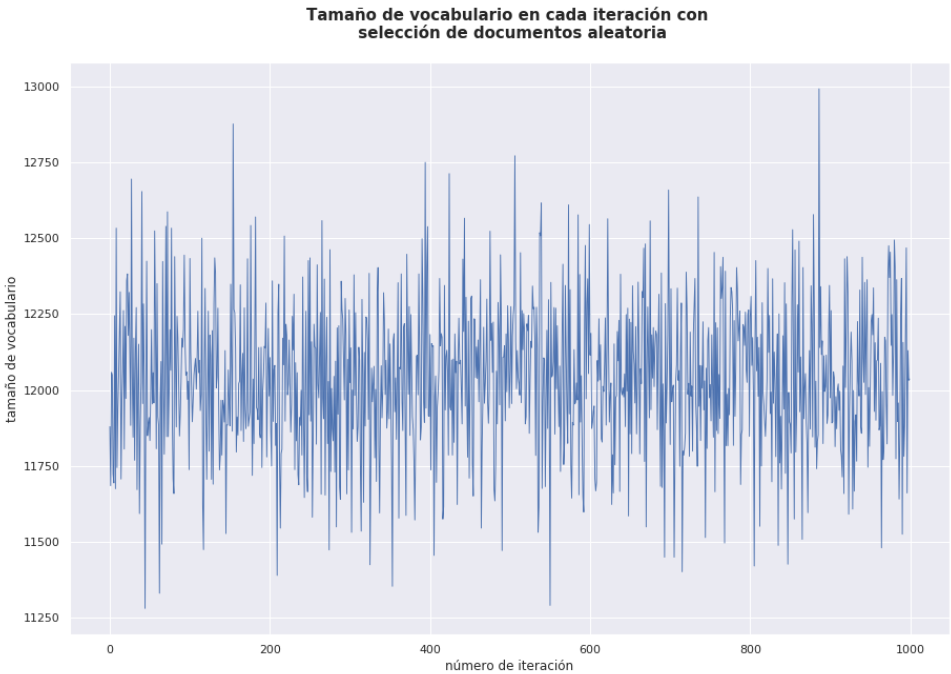

Figura 17: Tamaño de vocabulario en cada iteración con selección de documentos aleatoria (elaboración propia)

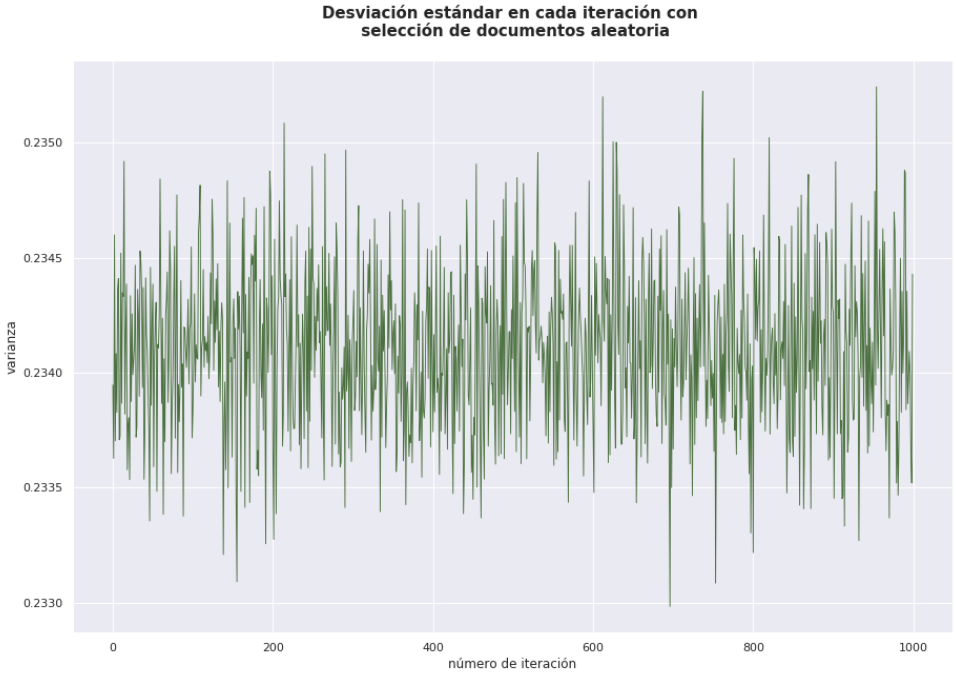

Figura 18: Desviación estándar en cada iteración con selección de documentos aleatoria (elaboración propia)

En ambos métodos, se puede observar que el tamaño de vocabulario promedio rodea las 12000 palabras diferentes. En el caso de la desviación estándar, vemos que esta no varía de forma directamente proporcional al tamaño de vocabulario y que en realidad es un valor casi constante en los dos métodos comparados, razón por la cual no se tomará en cuenta como un factor relevante en la selección de la muestra.

- Resultados

Tomando en cuenta exclusivamente el tamaño de vocabulario, se optó por la muestra obtenida a través del método aleatorio. Las características de este sub-conjunto de los datos se muestran en la Tabla número 9.

Tabla 9. Características de la muestra significativa obtenida mediante el método aleatorio (elaboración propia)

Tamaño de vocabulario	129994
Varianza	0.054696675
Desviación estándar	0.2338732

5.3.2 Cálculo de las métricas de precisión, exactitud y medida F1

Se utilizaron los módulos de NER de las librerías SpaCy y CoreNLP para el reconocimiento de entidades nombradas en todos los textos de la muestra. Para los resultados de cada librería, haciendo uso de las anotaciones manuales, se calcularon los valores de la matriz de clasificación de cada documento, utilizando el siguiente criterio:

- Verdaderos positivos: aquellas entidades presentes en las anotaciones que el módulo de NER identificó y clasificó en la categoría correcta.
- Falsos positivos: entidades presentes en las anotaciones que el módulo de NER identificó, pero clasificó en una categoría incorrecta.
- Verdaderos negativos: entidades identificadas por el módulo de NER que no están presentes en las anotaciones.
- Falsos negativos: entidades presentes en las anotaciones que el módulo de NER no identificó.

Finalmente, con los valores de la matriz de clasificación se calcularon las métricas de precisión, exactitud y medida F1 para cada documento; las cuales serán utilizadas en la experimentación numérica.

5.3.3 Experimentación numérica

- Recolección de muestras utilizando *bootstrap sampling*

El método de *bootstarp sampling* es utilizado para estimar el valor de un estadístico mediante el muestreo con repetición de una población (Efron, 1979). Dado que la métrica a comparar en

esta experimentación numérica es la medida F1, se recolectaron 30 *samples* de esta medida para cada módulo de NER utilizando el método mencionado. Es así, como las variables a comprar serán:

- X: medidas F1 obtenidas por el módulo de NER de la librería SpaCy
- Y: medidas F1 obtenidas por el módulo de NER de la librería CoreNLP

- Prueba de Kolmogorov-Smirnov

El primer paso de la experimentación numérica es averiguar si las variables a evaluar siguen o no una distribución normal, para lo cual se utiliza la prueba de Kolmogorov-Smirnov. Las hipótesis nulas y alternativas en los casos de ambas variables serán las siguientes:

- H0: la variable X sigue una distribución normal
- H1: la variable X no sigue una distribución normal
- H0: la variable Y sigue una distribución normal
- H1: la variable Y no sigue una distribución normal

Se aplicó la prueba de Kolmogorov-Smirnov en las dos variables definidas con un nivel de significancia del 5%, los resultados obtenidos se muestran en la Tabla número 10.

Tabla 10. Resultados de la prueba de Kolmogorov-Smirnov en las medidas F1 de los dos módulos de NER evaluados (elaboración propia)

	X	Y
Nivel de significancia	0.05	0.05
Valor crítico	0.248	0.248
Máxima diferencia	0.566	0.505

En vista que la máxima diferencia en ambos casos es mayor que el valor crítico, se rechazan las hipótesis nulas y se acepta que las variables X e Y no siguen una distribución normal.

- Prueba de Wilcoxon

La prueba de Wilcoxon se utiliza para comparar las medianas de dos muestras relacionadas, cuando estás no siguen una distribución normal. De este modo, para las variables a evaluar se plantean las siguientes hipótesis:

- H0: La mediana de X es menor igual que la de Y
- H1: La mediana de X es mayor que la de Y

Después de aplicar la prueba de Wilcoxon con un nivel de significancia del 5% se obtienen los resultados presentados en la Tabla número 11.

Tabla 11. Resultados de la prueba de Wilcoxon en las medidas F1 de los dos módulos de NER evaluados (elaboración propia)

Nivel de significancia	0.05
Valor crítico	152
Estadístico Wilcoxon	179

Dado que el estadístico obtenido es mayor que el valor crítico en la prueba a cola derecha, se rechaza la hipótesis nula y se acepta la alternativa. Por lo tanto, la mediana de X es mayor a la de Y.

5.3.4 Módulo de NER elegido

Finalmente, basándonos en los resultados de la experimentación numérica, se puede afirmar que el valor medio de las medidas F1 obtenidas con la librería SpaCy es significativamente mayor que el de las obtenidas con CoreNLP. Es por esta razón que el módulo de NER de SpaCy será el que se optimizará en el siguiente resultado.

Capítulo 6. Personalización del módulo de reconocimiento de entidades nombradas para el dominio y vocabulario específico

6.1 Introducción

En este capítulo se presenta el desarrollo del resultado esperado número 3. En este se realiza la personalización del módulo de reconocimiento de entidades nombradas de la librería SpaCy, seleccionado en base a los resultados de la experimentación numérica del capítulo anterior, para el dominio y vocabulario específico del presente proyecto. Con el objetivo de encontrar los mejores parámetros a utilizar en la optimización, se utilizó el método de validación cruzada para la evaluación de la precisión, exactitud y medida f1 en cada época de entrenamiento.

6.2 Descripción del resultado

Se evaluó la medida f1 mediante el método de validación cruzada *K-folds* por 50 iteraciones, buscando así encontrar el valor óptimo para el número de épocas de entrenamiento del módulo de reconocimiento de entidades nombradas. Una vez obtenido este valor, el modelo fue tuneado y los resultados se reflejan en la curva de aprendizaje presentada al final de este capítulo.

6.3 Desarrollo del resultado

6.3.1 Evaluación de parámetros utilizando validación cruzada

Para esta evaluación, primero fue necesario dividir el conjunto de anotaciones en dos subconjuntos de enteramiento y prueba de tamaños proporcionales al 80% y 20% respectivamente. Posteriormente se utilizó la técnica de validación cruzada *K-folds,* la cual divide los datos en k particiones para que una de estas sea utilizada como conjunto de datos de prueba y las otras k-1 como datos de entrenamiento en la evaluación de un modelo o, en este caso particular, del módulo de NER. Este proceso se repite durante k iteraciones, con cada una

de las posibles combinaciones de subconjuntos de prueba y entrenamiento que permitan las particiones, para finalmente devolver como resultado de la validación cruzada el promedio de los valores de la métrica a evaluar obtenida en cada iteración.

En el caso del presente proyecto, se utilizó el subconjunto de entrenamiento para la validación cruzada. Este se dividió en 10 particiones y la métrica que se buscaba evaluar fue la medida F1. Esta validación se repitió durante 200 iteraciones aumentado el número de épocas de entrenamiento del módulo de NER, los resultados de esta prueba se muestran en las Figuras 19 y 20:

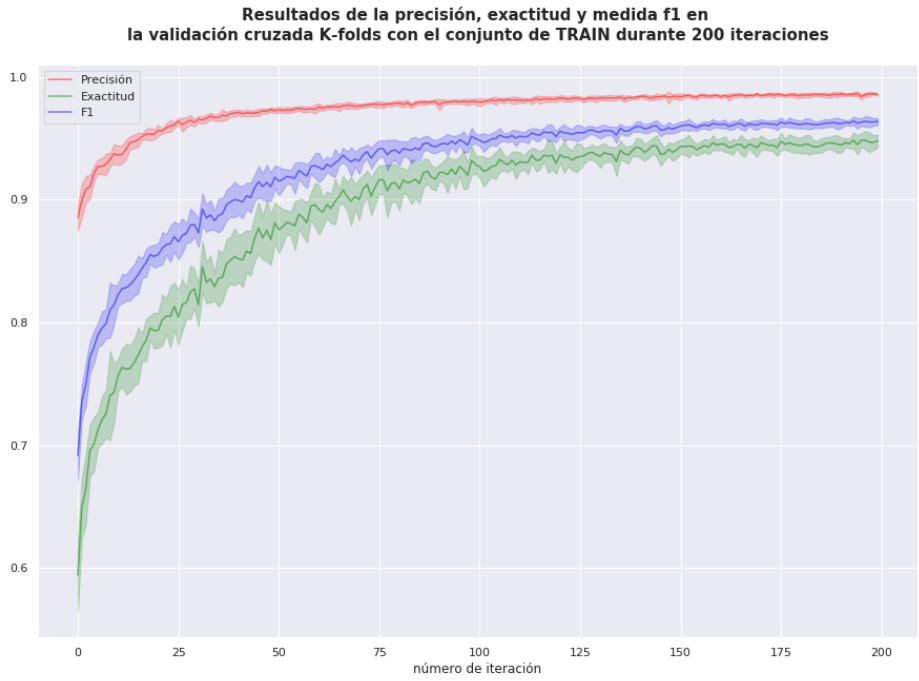

Figura 19: Resultados de las métricas de precisión, exactitud y media F1 en el subconjunto de TRAIN de la validación cruzada K-folds durante 200 iteraciones (elaboración propia)

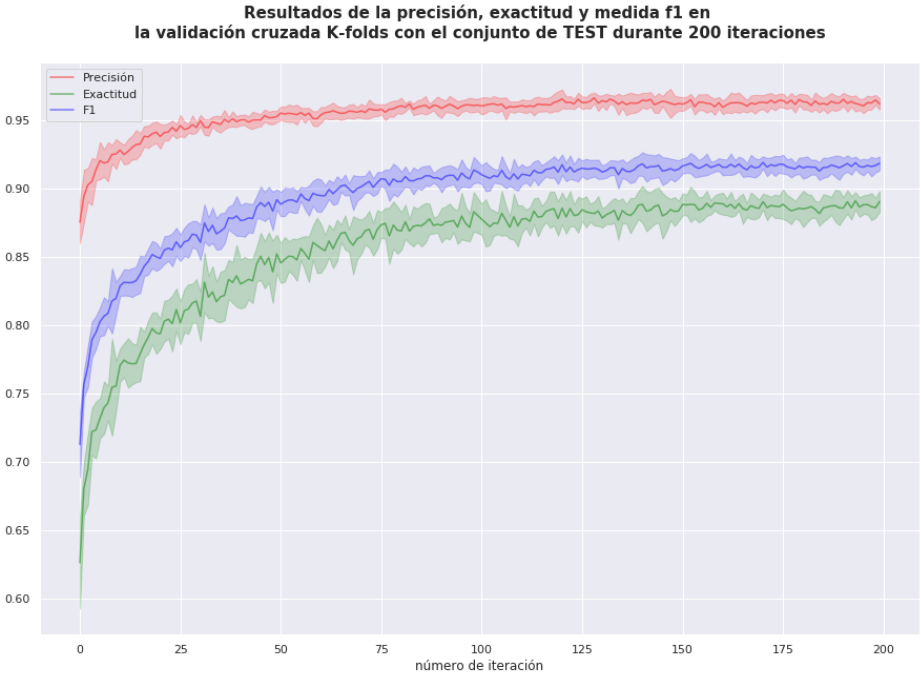

Figura 20: Resultados de las métricas de precisión, exactitud y media F1 en el subconjunto de TEST de la validación cruzada K-folds durante 200 iteraciones (elaboración propia)

Como se puede visualizar, a partir de la iteración 150 ya no hay un aumento significativo en ninguna de las métricas. Por lo tanto, se considerará este número como el valor óptimo para el número de épocas de entrenamiento en el siguiente paso.

6.3.2 Entrenamiento del módulo de NER

En este paso se utilizó como base el módulo de NER del conjunto de herramientas 'es_core_news_md' de la librería SpaCy. Este se entrenó durante 150 épocas con el subconjunto de entrenamiento, obteniéndose la siguiente curva de aprendizaje:

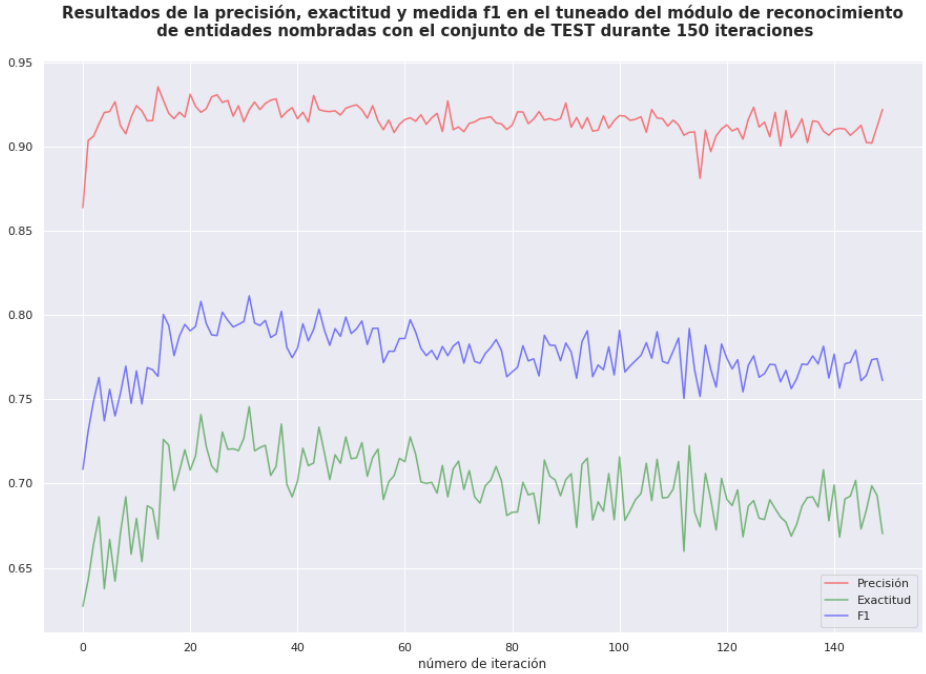

Figura 21: Curva de aprendizaje del módulo de NER con el conjunto de TEST (elaboración propia)

Una vez finalizado el entrenamiento, el NER personalizado es serializado en un archivo para su uso posterior.

Capítulo 7. Recurso léxico con términos de dominio criminal

7.1 Introducción

En el presente capítulo se explica el desarrollo del resultado esperado número 4, el cual tiene como propósito la creación de un tesauro de dominio criminal para representación computacional del conocimiento en esta área. Para la elaboración de este primero se definió la estructura y clases del recurso léxico, luego se completó cada clase con semillas o *seed terms* y finalmente se realizó la expansión de estos.

7.2 Descripción del resultado

El tesauro creado contiene clases con términos relacionados a los personajes dentro de un crimen (culpable y víctima) como también clases con términos relacionados a los tipos de crímenes a detectar (robo, asesinato, violación y venta de drogas). Esta estructura fue inicialmente poblada con las anotaciones manuales hechas sobre un 5% del corpus (250 noticias aproximadamente). Finalmente tomando estas palabras como *seed terms* se hizo una expansión de términos considerando características particulares (p.e. la categoría gramatical) de cada una de las palabras.

7.3 Desarrollo del resultado

7.3.1 Estructura del tesauro de dominio criminal

La estructura y clases del tesauro de dominio criminal se definieron en base a las relaciones que se busca identificar posteriormente. El esquema final de esta junto con algunos datos generales se muestran en la Figura número 22 y en la Tabla 12.

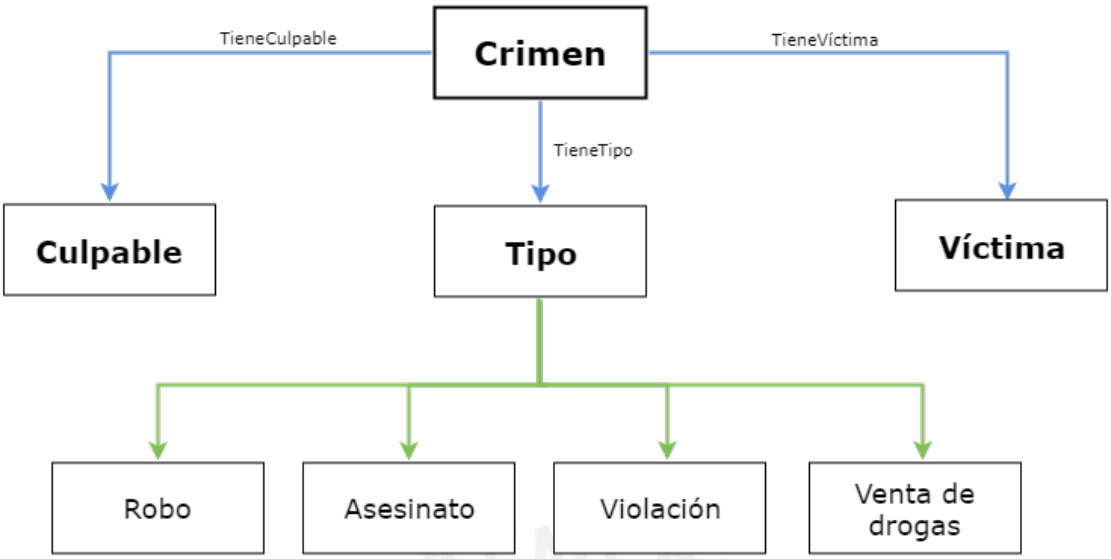

Figura 22: Estructura y clases del tesauro de dominio criminal (elaboración propia)

Tabla 12. Datos generales sobre el tesauro de dominio criminal (elaboración propia)

Datos generales	
Número de clases	8
Número de relaciones	3
Niveles jerárquicos	3

Cada concepto Crimen se relaciona con un Culpable, una Víctima y un Tipo de crimen. Además, existen cuatro subtipos de crimen los cuales tendrán diferentes términos asociados a fin de la identificación de este.

7.3.2 Términos semilla y expansión

Se seleccionó una muestra significativa de aproximadamente el 5% del conjunto de datos (250 documentos) para anotaciones manuales de términos relacionados a la víctima, culpable y tipo de crimen. En total se logró identificar 279 palabras diferentes en las anotaciones, pertenecientes a las diferentes clases del tesauro.

Los términos anotados para cada clase se utilizaron como *seed terms* y en base a estos se hizo una expansión con términos similares utilizando el analizador morfológico UniMorph, para el caso de los verbos, y vectores de palabras pre entrenados de FastText para el caso de los sustantivos.

Por ejemplo, si el término original era acusado, con el analizador morfológico se expande el género y número del verbo a: acusada, acusados, acusadas. Por otro lado, para sustantivos como narcotráfico, con los vectores de palabras pre entrenados se puede expandir el término a sus palabras más similares como: narcotraficante, narcotraficantes, narcóticos.

Las distribuciones de términos originales junto con los resultados de la expansión se muestran en la Tabla número 13. Del mismo modo, la figura número 23 muestra algunos de los términos presentes en las clases del culpable y víctima.

Tabla 13. Número de términos originales y resultados de la expansión en el tesauro de dominio criminal (elaboración propia)

Tipo de entidad	Número de términos anotados	Número de términos después de la expansión
Culpable	135	190
Víctima	55	65
Robo	14	38
Asesinato	22	83
Violación	27	39
Venta de drogas	26	34
Total	279	449

7.3.3 Métricas de la base de conocimiento

Una vez creado un tesauro es importante analizar y evaluar las características de su esquema, clase y términos a fin de conocer si tiene las características adecuadas para representar el

dominio de conocimiento deseado (Tartir, Arpinar, Moore, Sheth, & Aleman-meza, 2005). Las métricas calculadas para el tesauro desarrollado en el presente proyecto son las siguientes:

- Riqueza de clases: ya que todas las clases en el recurso léxico creado contienen individuos, la riqueza de clase en este caso es igual a 1.
- Población promedio: la población promedio en el tesauro desarrollado es 56.
- Importancia de clase

Tabla 14. Métricas de importancia de clase para cada una de las clases del tesauro de dominio criminal (elaboración propia)

Tipo de entidad	Número de términos después de la expansión	Importancia de clase
Culpable	190	0.42
Víctima	65	0.14
Robo	38	0.08
Asesinato	83	0.18
Violación	39	0.09
Venta de drogas	34	0.08

Capítulo 8. Programa de procesamiento de lenguaje natural para la generación de reportes estructurados

8.1 Introducción

En este capítulo se presenta el desarrollo del resultado esperado número 5. Este es un programa de procesamiento de lenguaje natural que utiliza los dos resultados anteriores para identificar las relaciones entre las entidades (detectadas por el módulo de reconocimiento de entidades nombradas) y términos del tesauro presentes en un texto, a fin de posteriormente utilizar esa relación para poder clasificar la entidad en una de las categorías definidas previamente (culpable, víctima y locación de un delito)

8.2 Descripción del resultado

Se seleccionó una muestra representativa de 10% de conjunto de datos (500 documentos aproximadamente) para anotaciones manuales, obteniéndose en base a estas 5000 oraciones identificadas con las clases objetivo. Utilizando las oraciones etiquetadas, se obtuvieron los patrones gramaticales que caracterizan a una entidad como culpable, víctima o locación de un crimen. Una vez obtenidos y almacenados estos patrones, se elaboró un programa de procesamiento de lenguaje natural para identificarlos en oraciones no etiquetadas y en base a esto predecir el culpable, víctima y locación de un documento.

8.3 Desarrollo del resultado

8.3.1 Definición de patrones de dependencia sintáctica

Cada una de las oraciones anotadas contenía los campos listados a continuación, además en la Tabla 15 se muestra un ejemplo de este tipo de registro.

- _id: Id único de la oración en la base de datos

- class: clase a la que pertenece la entidad a evaluar (culpable, víctima o locación)
- ne_text: Entidad mencionada
- ne_label: Etiqueta de la entidad mencionada
- new_id: Id de la noticia a la que pertenece la oración
- text: Texto de la oración

Tabla 15. Ejemplo de registro en la base de datos de una oración anotada (elaboración propia)

_id	ObjectId('5bd5a8b797c49217a077dd94')
new_id	ObjectId('5b5a4591b8088d165391bdc8')
class	CULPABLE
ne_label	PER
ne_text	Joel Daga Inocente
text	Joel Daga Inocente (29) asesinó a sus dos menores hijos y a su esposa

Las oraciones anotadas se dividieron en subconjuntos de entrenamiento y prueba correspondientes al 80% y 20% del total respectivamente. Para cada una de las oraciones del subconjunto de entrenamiento se construyó el árbol de dependencias sintácticas y se extrajeron las relaciones que conectaban a la entidad detectada con alguno de los términos de la base de conocimiento desarrollada en el resultado anterior.

Por ejemplo, para la oración mostrada en la Figura 23, la entidad nombrada era “Joel Daga Inocente”. En la misma oración, se encuentra el término “asesinó” el cual pertenece a la clase culpable de la base de conocimiento. La relación sintáctica entre la entidad nombrada y el término asesinó es “nsubj” (sintagma nominal) lo cual indica que Joel es el que realiza la acción en el verbo asesinó y por lo tanto es el culpable.

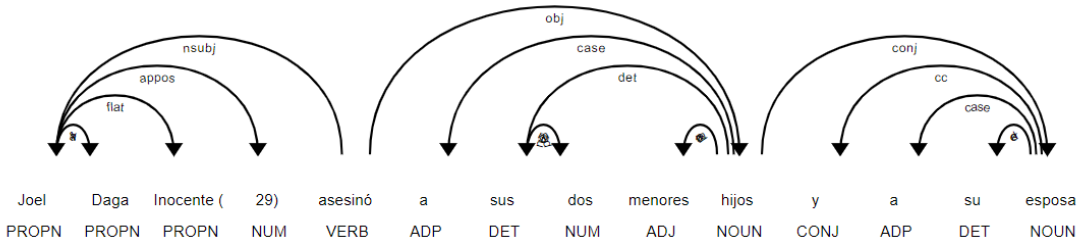

Figura 23: Ejemplo de dependencias sintácticas y etiquetado gramatical en una oración (elaboración propia)

Posteriormente se identificaron las 20 relaciones de dependencia sintáctica más frecuentes para cada clase y se establecieron como patrones pre definidos para la clasificación de oraciones no etiquetadas. En las Figuras mostradas a continuación, se visualizan las 10 primeras de las relaciones seleccionadas para cada clase:

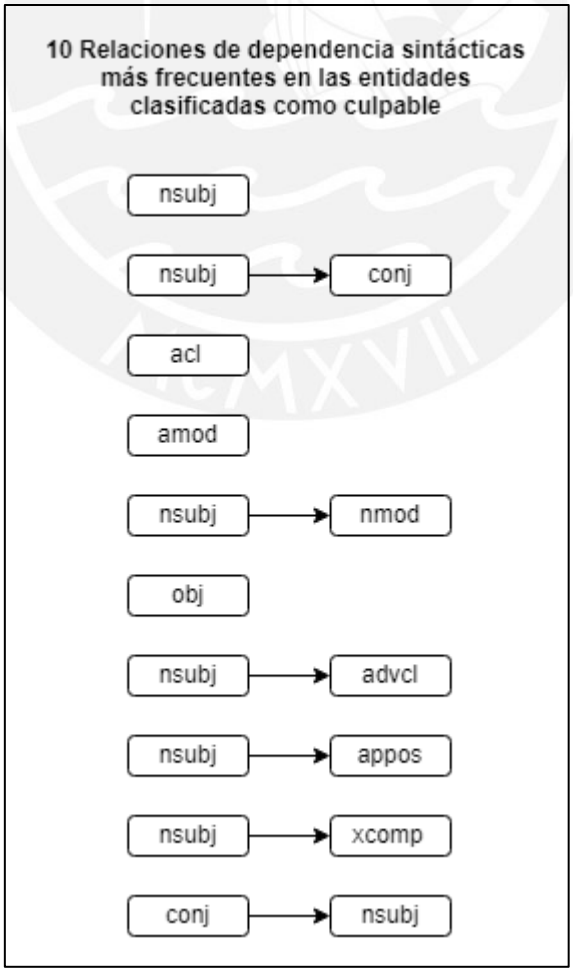

Figura 24: 10 relaciones de dependencia sintácticas más frecuentes en las entidades clasificadas como culpable

(elaboración propia)

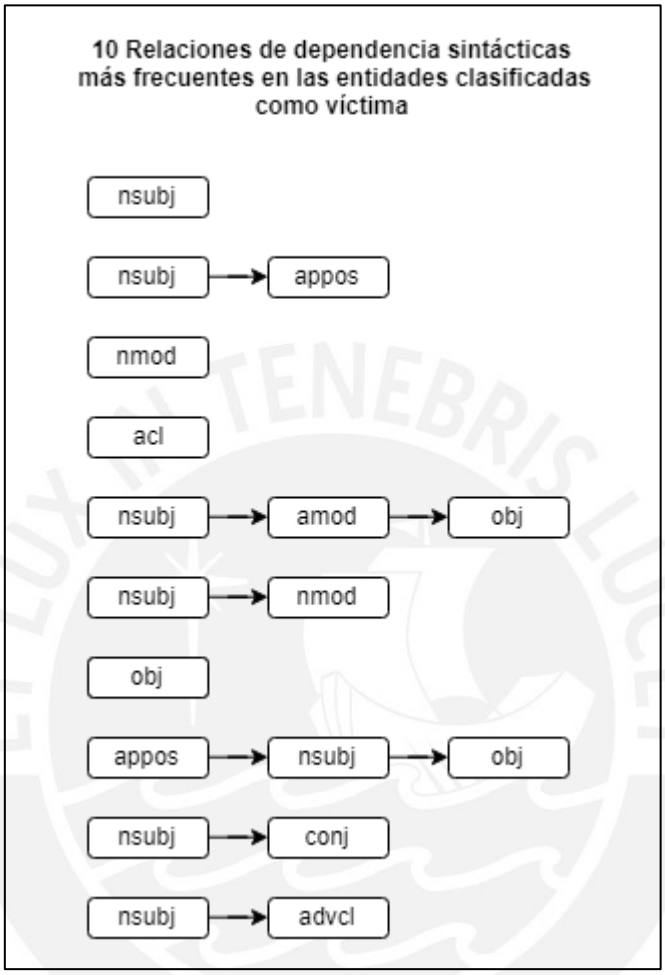

Figura 25: 10 relaciones de dependencia sintácticas más frecuentes en las entidades clasificadas como víctima

(elaboración propia)

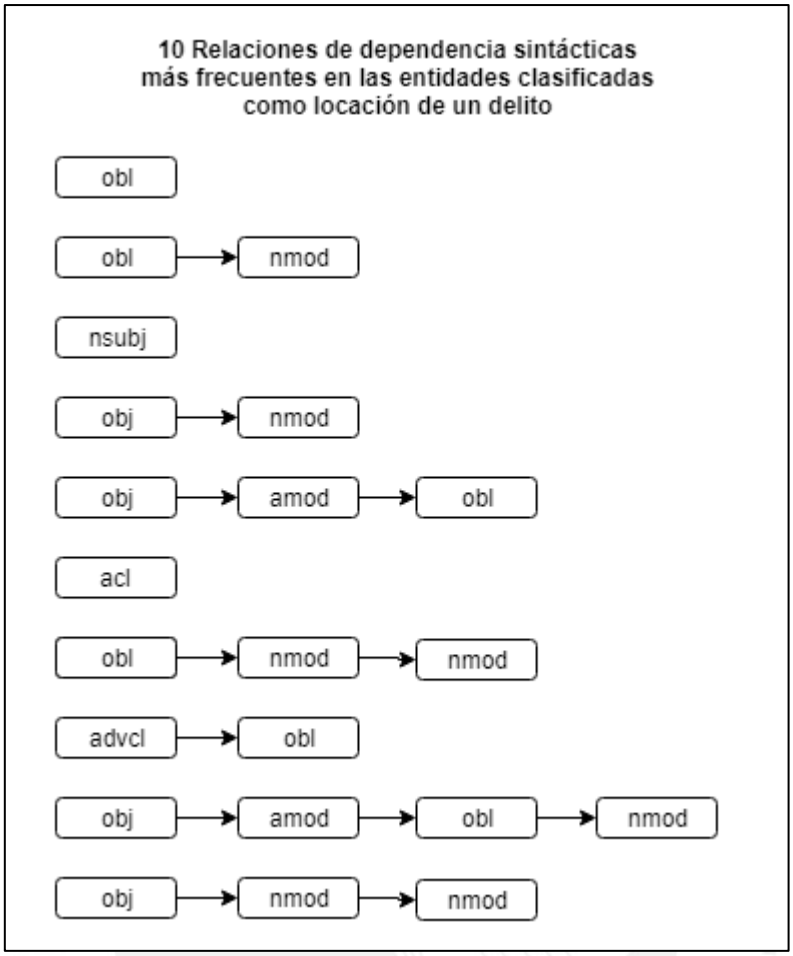

Figura 26: 10 relaciones de dependencia sintácticas más frecuentes en las entidades clasificadas como locación de un delito (elaboración propia)

8.3.2 Clasificación de entidades a nivel de oraciones

El programa desarrollado en esta parte busca clasificar la una entidad en una oración no etiquetada. Para esto, primero reconoce las entidades nombradas y los términos relacionados al culpable o víctima presentes en la oración, además de construir el árbol de dependencias sintácticas del texto. Posteriormente, evalúa la relación entre cada entidad nombrada y término encontrado. Si las dependencias sintácticas entre estos corresponden a uno de los patrones preestablecidos entonces asigna la clase correspondiente en la entidad mencionada. El flujo de trabajo explicado, se puede visualizar en la Figura número 27.

Figura 27: Flujo de trabajo para la clasificación de una entidad a nivel de oraciones (elaboración propia)

En este paso, también se evaluó cual era el número óptimo de relaciones a considerar en cada clase. Para esto se obtuvieron las métricas de precisión, exactitud y f1 score y se consideró como número optimo el punto en donde se maximizaban estas. Los resultados de esta experimentación se muestran en las Figuras 28, 29 y 30.

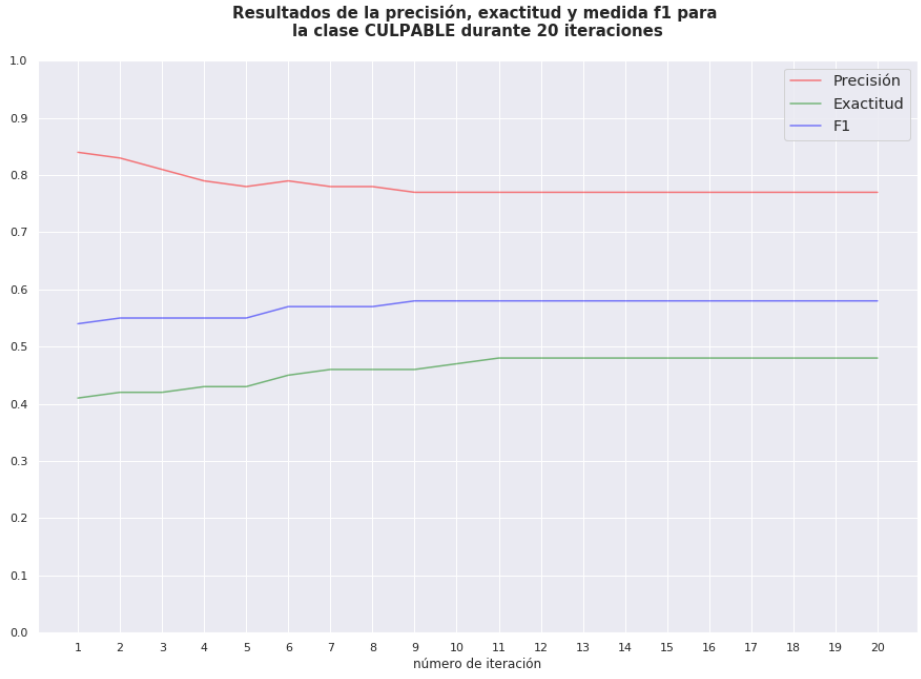

Figura 28: Resultados de precisión, exactitud y media f1 para la clase CULPABLE durante 20 iteraciones (elaboración propia)

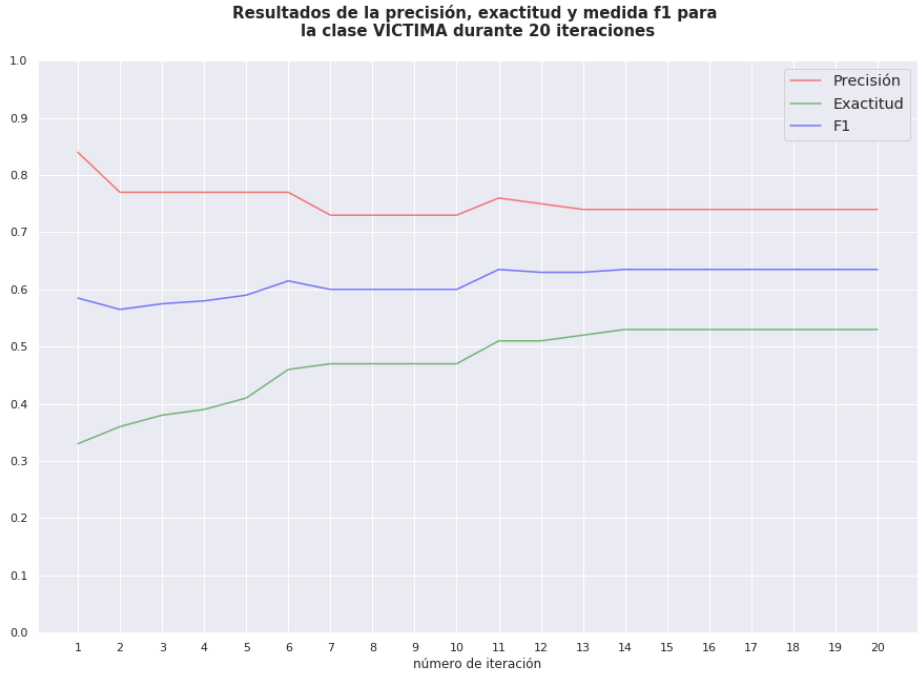

Figura 29: Resultados de precisión, exactitud y media f1 para la clase VÍCTIMA durante 20 iteraciones

(elaboración propia)

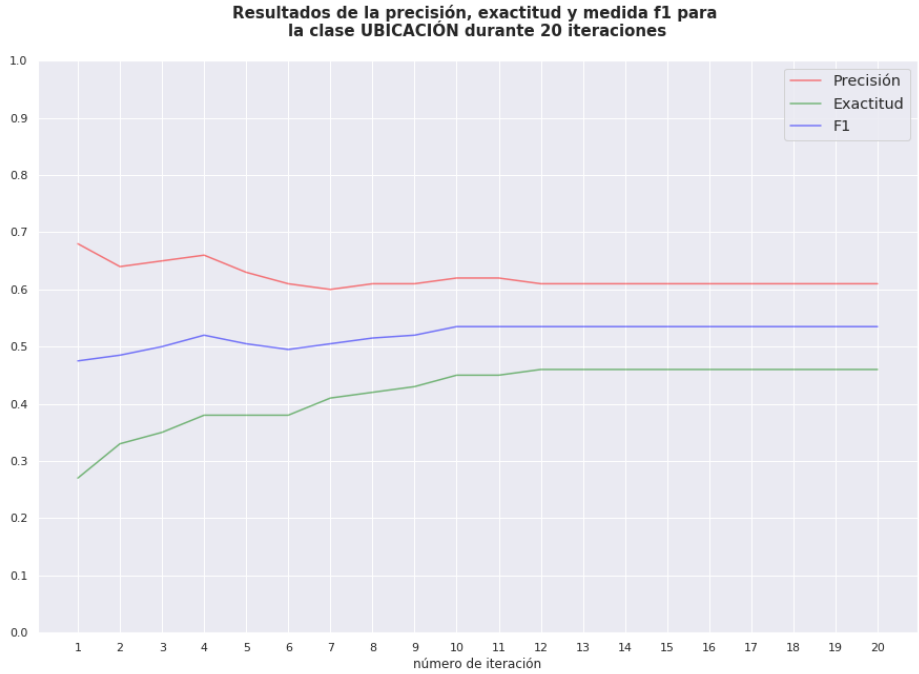

Figura 30: Resultados de precisión, exactitud y media f1 para la clase UBICACIÓN durante 20 iteraciones

(elaboración propia)

Finalmente, se decidió como número óptimo de relaciones sintácticas a considerar para cada clase los resultados de la Tabla número 16.

Tabla 16. Número óptimo de relaciones sintácticas a considerar para cada clase (elaboración propia)

Clase	Número de relaciones
Culpable	11
Víctima	11
Ubicación	13

8.3.3 Predicción a nivel de documentos

Este programa se desarrolló para poder identificar las clases objetivo a nivel de un documento (varias oraciones). Este programa recibe un documento evalúa cada oración y según el resultado de esta considera las entidades como "candidatas" a culpable, víctima y locación. Una vez obtenidos todos los candidatos se hace un análisis de correferencia, para identificar expresiones que se refieren a la misma entidad en el texto. Finalmente, se selecciona la entidad con mayor número de menciones en el documento como respuesta final en cada clase.

8.3.4 Resultados obtenidos

Los resultados de precisión obtenidos sobre el 20% de las anotaciones son los siguientes:

Tabla 17. Resultados de precisión, exactitud y media f1 para la clasificación a nivel de documentos (elaboración propia)

	Precisión	Exactitud	F1-score
Culpable	0.72	0.41	0.56
Víctima	0.69	0.39	0.54
Ubicación	0.58	0.42	0.5

Promedio	0.66	0.4	0.53

Como se puede visualizar donde hay menor precisión es identificar la locación de un delito, mientras que los culpables y víctimas los identifica bastante bien. Esta baja precisión se puede deber a que en una noticia se mencionan muchas más locaciones diferentes que actores o personas.

Capítulo 9. Interfaz de programación de aplicaciones para la presentación de funcionalidades del modelo algorítmico implementado

9.1 Introducción

El capítulo a continuación describe el desarrollo del número 3, el cual tiene como propósito implementar una interfaz de programación de aplicaciones (API, por sus siglas en inglés) a fin de hacer públicas las funcionalidades del modelo algorítmico de extracción de información desarrollado y que este pueda ser utilizado en trabajos e investigaciones futuras.

9.2 Descripción del objetivo

Para la elaboración del API se utilizó el framework Flask, el cual permite crear servicios web fácilmente. Esta contiene todas las funcionalidades del modelo algorítmico implementado, y está alojada en un servidor con las dependencias y recursos necesarios para su funcionamiento. El servicio desarrollado recibe el texto de una noticia, y devuelve en formato JSON el culpable, la víctima y la locación del crimen descrito en esta.

9.3 Desarrollo del objetivo

9.3.1 Interfaz de programación de aplicaciones (API)

Se creó un script en el lenguaje de programación Python, donde se encuentran las funciones implementadas previamente y también se cargan los componentes necesarios para la ejecución de estas (Módulo de reconocimiento e entidades nombradas, tesauro de dominio criminal, módulo para el etiquetado gramatical, etc.)

El servicio consiste en un solo método que recibe como parámetro la descripción de un delito y devuelve en formato JSON el culpable, la víctima y la locación identificados. Para acceder a

este, solo es necesario conocer la IP pública del servidor en donde se encuentra alojado y el puerto en el que se ejecuta.

La estructura para enviar una consulta al API se muestra a continuación:

http://<ip_server>:<port >/identify?new=<text>

Donde:

- ip_server: ip pública del servidor donde se encuentra alojado el servicio
- port: puerto en donde se ejecuta el servicio
- text: texto con la descripción de delito a evaluar

La respuesta exitosa a esta consulta, posee la siguiente estructura:

{

"culpable": <c_per>

"success": True

"ubicación": <loc>

"víctima" <v_per>

}

Donde <c_per> representa el nombre del culpable, <v_per> el de la víctima y <loc> la ubicación del delito. Es importante considerar que no en todos los casos todos los campos son identificados, por lo que podría haber campos vacíos dependiendo de las características del texto enviado.

9.3.2 Plataforma web

Para pruebas del funcionamiento del API implementada, se creó una plataforma web la cual puede ser consultada en la siguiente dirección: https://crime2web.herokuapp.com/. Esta recibe como entrada un texto plano proporcionado por el usuario, se conecta internamente al servicio siguiendo el formato especificado en la sección anterior, y muestra las respuestas proporcionadas por este.

Figura 31: Página de inicio de la plataforma web

Figura 32: Interfaz para el ingreso de textos por el usuario en la plataforma web

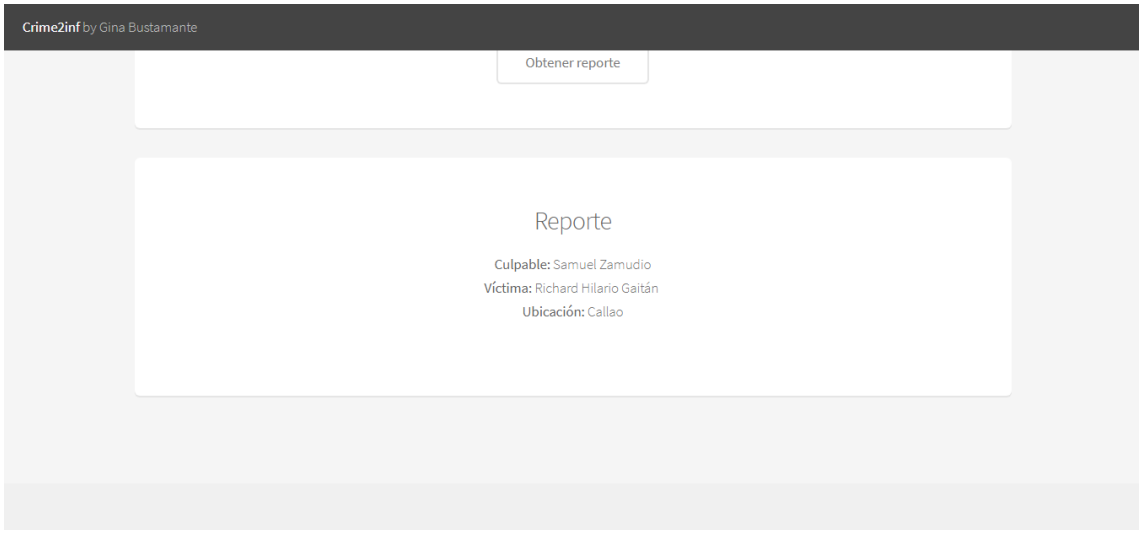

Figura 32: Ejemplo de la presentación de resultados proporcionados por el servicio en la plataforma web

Capítulo 10. Conclusiones y trabajos futuros

En este capítulo se presentan las conclusiones del presente proyecto. De mismo modo, se proponen algunos trabajos futuros a fin de seguir contribuyendo con propuestas de solución al problema planteado.

10.1 Conclusiones

En primer lugar, en el proyecto desarrollado se obtiene, pre procesa y genera un corpus en español con noticias relacionadas a crímenes. Este tipo de conjunto de datos son valiosos en el dominio del problema a tratar, y por lo tanto se considera un resultado importante de esta investigación.

Por otro lado, el módulo de reconocimiento de entidades nombradas personalizado para el dominio y vocabulario específico es también de gran ayuda para investigaciones futuras. Sobre todo, si se considera el hecho que el reconocimiento de entidades nombradas es uno de los pasos más importantes en la extracción de información.

El programa de procesamiento de lenguaje natural implementado para la identificación del culpable, víctima y locación de un crimen presenta una alta precisión, pero baja exactitud. Esto es común en problemas de extracción de información basados en patrones, y se puede mejorar utilizando modelos de aprendizaje de máquina.

Finalmente, la interfaz de programación de aplicaciones móviles permite un libre acceso al modelo algorítmico desarrollado, facilitando y fomentando así el desarrollo de más investigaciones en este campo.

10.2 Trabajos futuros

En primer lugar, en el presente proyecto se trabajó con descripciones de delitos presentes en noticias porque era el tipo de conjunto de datos al que se podía acceder sin restricciones legales. Sin embargo, sería interesante explorar otro tipo de datos oficiales como reportes policiales.

Del mismo modo, en el caso de trabajar con un nuevo conjunto de datos, también se podría popular el tesauro con términos más técnicos (p.e. los presentes en el código penal).

Finalmente, y como se mencionó anteriormente, la extracción de información basada en patrones tiene alta precisión y baja exactitud. Este tipo de problemas se puede solucionar utilizando los resultados de la extracción basada en patrones para entrenar modelos de aprendizaje de máquina, ya sean tradicionales (como máquinas de vectores de soporte) o de aprendizaje profundo (redes neuronales recurrentes), de modo que los modelos pueden identificar características latentes en las oraciones/documentos y mejorar la clasificación de entidades y relaciones.

Referencias

Appelt, D. E. (1999). Introduction to information extraction. *Ai Communications*, *12*(3), 161–172.

Arulanandam, R., Savarimuthu, B. T. R., & Purvis, M. A. (2014). Extracting Crime Information from Online Newspaper Articles. *Proceedings of the Second Australasian Web Conference (AWC 2014), Auckland, New Zealand*, (Awc), 31–38. Retrieved from https://dl.acm.org/citation.cfm?id=2667706

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python*. O'Reilly.

Chau, M., Xu, J. J., & Chen, H. (2002). Extracting meaningful entities from police narrative reports. *Proceedings of the 2002 Annual National Conference on Digital Government Research*, 1–5. Retrieved from https://dl.acm.org/citation.cfm?id=1123138

Chen, H., Chung, W., Qin, Y., Chau, M., Xu, J. J., Wang, G., & Zheng, R. (2002). Crime Data Mining : An Overview and Case Studies. *Communications of the ACM*, *2*, 165–276. Retrieved from http://ai.bpa.arizona.edu/

Chibelushi, C., Sharp, B., & Shah, H. (2006). ASKARI: a crime text mining approach. *Digital Crime and Forensic Science in Cyberspace*, *155*.

Chinchor, N., & Sundheim, B. (1993). MUC-5 evaluation metrics. In *Proceedings of the 5th conference on Message understanding - MUC5 '93* (p. 69). Morristown, NJ, USA: Association for Computational Linguistics. https://doi.org/10.3115/1072017.1072026

Covington, M. A. (2001). A Fundamental Algorithm for Dependency Parsing. *Proceedings of the 39th Annual ACM Southeast Conference*, 95–102. https://doi.org/10.1.1.136.7335

Crime, U. N. O. on D. and. (2011). *Criminal Intelligence: Manual for Analysts*. United Nations Publications.

Cunningham, H. (2006). Information Extraction, Automatic Introduction: Extraction and Retrieval. *Oxford: Elsevier. Heath S B Kortmann B Miller J*, *5*, 665–677.

Dasgupta, T., Naskar, A., Saha, R., & Dey, L. (2017). CrimeProfiler. In *Proceedings of the International Conference on Web Intelligence - WI '17* (pp. 541–549). New York, New York, USA: ACM Press. https://doi.org/10.1145/3106426.3106476

Django Software Foundation. (2016). Django overview | Django, 1–3. Retrieved from https://www.djangoproject.com/start/overview/

Efron, B. (1979). Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, *7*(1), 1–26. https://doi.org/10.1214/aos/1176344552

Garten, J., Sagae, K., Ustun, V., & Dehghani, M. (2015). Combining Distributed Vector Representations for Words. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing* (pp. 95–101). https://doi.org/10.3115/v1/W15-1513

Grishman, R., & Sundheim, B. (1996). Message Understanding Conference-6. In *Proceedings of the 16th conference on Computational linguistics* - (Vol. 1, p. 466). Morristown, NJ, USA: Association for Computational Linguistics. https://doi.org/10.3115/992628.992709

Gudivada, V. N., Rao, D., & Raghavan, V. V. (2014). NoSQL Systems for Big Data Management. In *2014 IEEE World Congress on Services* (pp. 190–197). IEEE. https://doi.org/10.1109/SERVICES.2014.42

Gupta, G. K. (2014). *Introduction to data mining with case studies*. PHI Learning Pvt. Ltd.

Hassani, H., Huang, X., Silva, E. S., & Ghodsi, M. (2016). A review of data mining applications in crime. *Statistical Analysis and Data Mining*. https://doi.org/10.1002/sam.11312

Hirschberg, J., & Manning, C. D. (2015, July 17). Advances in natural language processing. *Science*. American Association for the Advancement of Science. https://doi.org/10.1126/science.aaa8685

Horrocks, I. (2013). HermiT Reasoner. Retrieved June 9, 2018, from http://www.hermit-reasoner.com/

Jayaweera, I., Sajeewa, C., Liyanage, S., Wijewardane, T., Perera, I., & Wijayasiri, A. (2015). Crime analytics: Analysis of crimes through newspaper articles. In *MERCon 2015 - Moratuwa Engineering Research Conference* (pp. 277–282). IEEE. https://doi.org/10.1109/MERCon.2015.7112359

Kaisler, S., Armour, F., Espinosa, J. A., & Money, W. (2013). Big Data: Issues and Challenges Moving Forward. In *2013 46th Hawaii International Conference on System Sciences* (pp. 995–1004). IEEE. https://doi.org/10.1109/HICSS.2013.645

Keyvanpour, M. R., Javideh, M., & Ebrahimi, M. R. (2011). Detecting and investigating crime by means of data mining: A general crime matching framework. In *Procedia Computer Science* (Vol. 3, pp. 872–880). Elsevier. https://doi.org/10.1016/j.procs.2010.12.143

Kittiphattanabawon, N., & Theeramunkong, T. (2009). Relation discovery from thai news articles using association rule mining. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 5477, pp. 118–129). https://doi.org/10.1007/978-3-642-01393-5_13

Krtalic, M., & Hasenay, D. (2012). Newspapers as a source of scientific information in social sciences and humanities: a case study of Faculty of Philosophy, University of Osijek, Croatia. In *IFLA World Library and Information Congress* (pp. 1–7).

Ku, C. H., Iriberri, A., & Leroy, G. (2006). Natural Language Processing and e-Government : Crime Information Extraction from Heterogeneous Data Sources. *The Proceedings of the 9th Annual International Digital Government Research Conference*, 162–170. Retrieved from http://scholarship.claremont.edu/cgu_fac_pub/211

Le, Q. V., & Mikolov, T. (2014). Distributed Representations of Sentences and Documents. https://doi.org/10.1145/2740908.2742760

LeBlanc, J. B., Elder, J., Bruce, C. W., & Santos, R. B. (2014). *Definition and Types of Crime Analysis*.

Maimon, O., & Rokach, L. (2005). *Data Mining and Knowledge Discovery Handbook*. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations* (pp. 55–60). https://doi.org/10.3115/v1/P14-5010

Martin, J., & Jurafsky, D. (2008). *Speech and language processing*. Retrieved from https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf

McAfee, A., & Brynjolfsson, E. (2012). Big data: The Management Revolution. *Harvard Business Review*, (October), 1–9. https://doi.org/00475394

McKeown, K. R. (1985). Discourse strategies for generating natural-language text. *Artificial Intelligence*, *27*(1), 1–41. https://doi.org/10.1016/0004-3702(85)90082-7

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. Retrieved from https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, *38*(11), 39–41. https://doi.org/10.1145/219717.219748

Mitchell, R. (2015). *Web scraping with Python: collecting data from the modern web*. " O'Reilly Media, Inc."

Mohd, M., & Ali, N. M. (2011). An interactive Malaysia crime news retrieval system. In *2011 International Conference on Semantic Technology and Information Retrieval, STAIR 2011* (pp. 220–223). IEEE. https://doi.org/10.1109/STAIR.2011.5995792

MongoDB. (2017). The MongoDB 3.6 Manual. Retrieved from http://docs.mongodb.org/manual

Musen, M. A., & Protégé Team, the P. (2015). The Protégé Project: A Look Back and a Look Forward. *AI Matters*, *1*(4), 4–12. https://doi.org/10.1145/2757001.2757003

Nédellec, C., & Nazarenko, A. (2006). Ontologies and Information Extraction. Retrieved from http://arxiv.org/abs/cs/0609137

Netsuwan, T., & Kesorn, K. (2017). Unify framework for crime data summarization using RSS feed service. *Walailak Journal of Science and Technology*, *14*(10Special Issue), 769–781. https://doi.org/10.14456/vol14iss9pp%p

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., … Lerer, A. (2017). Automatic differentiation in PyTorch.

Patterson, C., Emslie, C., Mason, O., Fergie, G., & Hilton, S. (2016). Content analysis of UK newspaper and online news representations of women's and men's "binge" drinking: a challenge for communicating evidence-based messages about single-episodic drinking? *BMJ Open*, *6*(12), e013124. https://doi.org/10.1136/bmjopen-2016-013124

Pawar, S., Palshikar, G. K., & Bhattacharyya, P. (2017). Relation Extraction : A Survey. Retrieved from http://arxiv.org/abs/1712.05191

Pramanik, M. I., Lau, R. Y. K., Yue, W. T., Ye, Y., & Li, C. (2017). Big data analytics for security and criminal investigations. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *7*(4), e1208. https://doi.org/10.1002/widm.1208

Proyect Jupyter. (2017). The Jupyter Notebook. Retrieved June 3, 2018, from http://jupyter.org/

Python Software Foundation. (2017). What is Python? Executive Summary. Retrieved June 3, 2018, from https://www.python.org/doc/essays/blurb/

Richardson, L. (2016). Beautiful Soup Documentation, 1–72. Retrieved from http://www.crummy.com/software/BeautifulSoup/bs4/doc/

Ronacher, A. (2010). Foreword — Flask 1.0.2 documentation. Retrieved June 8, 2018, from

http://flask.pocoo.org/docs/1.0/foreword/#what-does-micro-mean

spaCy. (2018). spaCy 101: Everything you need to know. Retrieved June 3, 2018, from https://spacy.io/usage/spacy-101

Tartir, S., Arpinar, I. B., Moore, M., Sheth, A. P., & Aleman-meza, B. (2005). OntoQA : Metric-Based Ontology Quality Analysis University of Georgia. *ComputerIEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantic*. Retrieved from https://www.semanticscholar.org/paper/OntoQA-%3A-Metric-Based-Ontology-Quality-Analysis-Tartir-Arpinar/0b2c1aeb30d8961b9678b97baf011a8976c310cb

Thongtae, P., & Srisuk, S. (2008). An analysis of data mining applications in crime domain. In *Proceedings - 8th IEEE International Conference on Computer and Information Technology Workshops, CIT Workshops 2008* (pp. 122–126). IEEE. https://doi.org/10.1109/CIT.2008.Workshops.80

Tibbo, H. R. (2002). Primarily history: historians and the search for primary source materials. *Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries*, 1–10. https://doi.org/10.1145/544220.544222

Tran, M. V., Nguyen, M. H., Nguyen, S. Q., Nguyen, M. T., & Phan, X. H. (2012). VnLoc: A real-time news event extraction framework for Vietnamese. In *Proceedings - 4th International Conference on Knowledge and Systems Engineering, KSE 2012* (pp. 161–166). IEEE. https://doi.org/10.1109/KSE.2012.34

van Banerveld, M., Le-Khac, N.-A., & Kechadi, M.-T. (2014). Performance Evaluation of a Natural Language Processing Approach Applied in White Collar Crime Investigation (pp. 29–43). https://doi.org/10.1007/978-3-319-12778-1_3

Voutilainen, A. (2005). Part-of-Speech Tagging, *1*. https://doi.org/10.1093/oxfordhb/9780199276349.013.0011

Westphal, C. (2008). *Data Mining for Intelligence, Fraud & Criminal Detection*. CRC

Press. https://doi.org/10.1201/9781420067248

Anexo 1. Planificación de tareas

Tarea	Fecha de inicio	Fecha de fin
Gestión del proyecto y documentación		
Investigación de técnicas de extracción de información utilizadas anteriormente en el dominio de análisis criminal	02/04	27/04
Definición de los objetivos y resultados esperados del proyecto a realizar	30/04	20/05
Investigación de herramientas y métodos a utilizar para el desarrollo del presente proyecto	21/05	03/06
Análisis de la viabilidad del proyecto	04/06	10/06
Corrección de observaciones realizadas sobre la investigación	11/06	15/06
O1: Procesar y generar un conjunto de datos estructurado en base a noticias criminales en la web		
Exploración de las noticias a extraer	16/07	22/07
Elaboración del programa de extracción y recolección de noticias	23/07	29/07
Creación de una base de datos a partir de los textos recolectados	30/07	05/08
Exploración y filtrado de noticias	06/08	12/08

O2: Implementar y validar un algoritmo para la extracción automática de información de las noticias.		
Implementación de un módulo de software para la extracción automática de entidades y relaciones genéricas en los textos	27/08	02/09
Creación del tesauro de dominio criminal	03/09	09/09
Selección y entrenamiento de un modelo de aprendizaje de máquina para la clasificación de las entidades extraídas relacionadas al dominio del crimen	10/09	23/09
Implementación de un programa de generación de lenguaje natural para la emisión de reportes estructurados	24/09	30/09
O3: Elaborar una interfaz de programación de aplicaciones para la presentación de funcionalidades del modelo desarrollado		
Elaboración de la interfaz de programación de aplicaciones con el algoritmo generado	01/10	07/10
Elaboración de la plataforma web para las pruebas de la interfaz de programación de aplicaciones	08/10	14/10

ET LUX IN TENEBRIS LUCET
MCMXVII