ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE SISTEMAS

UN MODELO PARA LA OBTENCIÓN DE INTERACCIONES MEDICAMENTOSAS MEDIANTE APRENDIZAJE PROFUNDO SOBRE EL CORPUS 'DDI EXTRACTION 2013'

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL GRADO DE MAGÍSTER EN COMPUTACIÓN

María Cristina Jiménez Hernández

maria.jimenez01@epn.edu.ec

Director: Phd Marco Molina

marco.molinab@epn.edu.ec

Co-Director: Msc Carlos Montenegro

carlos.montenegro@epn.edu.ec

2020

APROBACIÓN DEL DIRECTOR

Como director del trabajo de titulación UN MODELO PARA LA OBTENCIÓN DE INTERACCIONES MEDICAMENTOSAS MEDIANTE APRENDIZAJE PROFUNDO SOBRE EL CORPUS 'DDI EXTRACTION 2013' desarrollado por Cristina Jiménez , estudiante de Maestría en Computación, habiendo supervisado la finalización de este trabajo y realizado las correcciones correspondientes, apruebo la redacción final del documento escrito para continuar con el correspondiente procedimientos para apoyar la defensa oral.

Phd Marco Molina
DIRECTOR

APROBACIÓN DEL CO-DIRECTOR

Como codirector del trabajo de grado UN MODELO PARA LA OBTENCIÓN DE INTERACCIONES MEDICAMENTOSAS MEDIANTE APRENDIZAJE PROFUNDO SOBRE EL CORPUS 'DDI EXTRACTION 2013' desarrollado por Cristina Jiménez , estudiante de Maestría en Computación, habiendo supervisado la finalización de este trabajo y realizado las correcciones correspondientes, apruebo la redacción final del documento escrito para continuar con el correspondiente procedimientos para apoyar la defensa oral.

Msc Carlos Montenegro
CO-DIRECTOR

DECLARACIÓN DE AUTORÍA

Yo, María Cristina Jiménez Hernández, declaro bajo juramento que el trabajo descrito aquí es mi responsabilidad; que no ha sido presentado previamente para ningún título o calificación profesional; y que he consultado las referencias bibliográficas incluidas en este documento. La Escuela Politécnica Nacional puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

María Cristina Jiménez Hernández

DEDICATORIA

Esta tesis se la dedico a Dios quién supo guiarme, darme fuerzas para seguir adelante y no desmayar en los momentos difíciles. A mi madre que está en el cielo y a mi padre por su apoyo, consejos, amor, ayuda en los momentos difíciles. A mis hijas y a mi esposo por ellos soy lo que soy, contituyen el motor de mi motivación, inspiración y felicidad.

AGRADECIMIENTOS

Primero y antes que nada, mi agradecimeinto es a Dios, por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente y por haber puesto en mi camino a aquellas personas que han sido mi soporte y compañía durante todo el periodo de estudio.
Agradecer hoy y siempre a mi familia. A mi esposo Byron y a mis maravillosas hijas Francis y Wendy, que con su apoyo y alegría que me brindan me dan la fortaleza necesaria para seguir adelante. A mi padre Segundo Rafael que siempre ha sido mi soporte y mi luz.
De igual manera mi más sincero agradecimiento al Phd Marco Molina, director del presente trabajo de investigación, muchas gracias por su guía científico-técnica; es una persona apasionada por su trabajo, con mucha vocación y servicio impulsando a que los profesionales alcancen sus objetivos.
Un agradecimiento especial al Msc Carlos Montenegro, decano de la Facultad de sistemas y codirector de este trabajo de investigación, por la colaboración brindada durante toda la tesis, gracias por su visión de desarrollo y crecimiento de la investigación y sobre todo la excelente gestión que realiza en favor de la facultad.

Índice general

Índice de figuras

Índice de tablas

Resúmen

Las interacciones medicamentosas (DDI) constituyen información importante y útil para el personal médico y los pacientes, puesto que proporcionan información sobre los efectos que producen durante una terapia los medicamentos coadministrados a un paciente. Este estudio utiliza un modelo convolucional por partes PCNN para capturar eficientemente la relación que existe entre entidades farmacológicas descrita en la literatura biomédica. Adicionalmente, este modelo utilizó la incorporación de palabras multicanal para ampliar el vocabulario y disminuir la cantidad de palabras desconocidas, y el optimizador estocástico Adam para aprender automáticamente los parámetros de red y se añadió una capa de ruido Gausiano para la extracción eficaz de relaciones DDI. Los experimentos muestran una mejora en el rendimiento del nuevo modelo, en relación con los modelos encontrados en la literatura técnica actual, relacionados con el desafío DDIExtraction2013, con resultados verificables y reproducibles.

Keywords: drug-drug interaction, deep learning, relations extraction, DDIExtraction2013 Challenge

Abstract

Drug-Drug interactions (DDI) constitutes essential and useful information for medical staff and patients, since they provide information on the effects of co-administered medications to patients, during therapy. This study uses a Piecewise Convolutional Neural Network (PCNN) to capture the relationship between pharmacological entities described in the biomedical databases. Additionally, the model incorporates multichannel words to expand vocabulary and decrease the number of unknown words. The stochastic optimizer Adam is used for learning the network parameters automatically, and Gaussian noise layer is added to improve the extraction of DDI relationships. Experiments show an improved performance of the new model, regard to the models found in the current technical literature, related to the DDIExtraction2013 Challenge, with verifiable and reproducible results.

Keywords: drug-drug interaction, deep learning, relations extraction, DDIExtraction2013 Challenge

Capítulo 1

Introducción

El procesamiento del lenguaje natural (PLN) es una rama de las ciencias computacionales que se ocupa de la investigación de mecanismos con alto rendimiento, para lograr una comunicación efectiva entre la máquina y el ser humano, a través del lenguaje natural. El objetivo principal del PLN es construir sistemas computacionales que permitan relacionar al hombre con la computadora por medio de lenguajes naturales. La comprensión y generación del lenguaje natural no es un proceso simple más bien es complejo debido al alto nivel de ambigüedad e interpretabilidad que tiene el lenguaje. Exactamente, el objetivo de la minería de texto es comprender el lenguaje humano, a partir de una basta colección de documentos que se recibe como datos de entrada. Fundamentalmente, la extracción de información de estos documentos es el proceso de transformar datos no estructurados en datos estructurados. Para realizar esta tarea se plantean dos pasos: el Reconocimiento de elementos como: nombre de personas, organizaciones, lugares, expresiones de tiempo o cantidades de oraciones llamadas entidades; y la extracción de relaciones, que define las asociaciones más importantes entre las entidades. En el campo de la medicina, existen numerosas problemáticas relacionadas con la literatura y la práctica cotidiana del ejercicio profesional. La comprensión completa de los documentos que conforman la literatura biomédica se dificulta en la práctica, debido a que presentan estructuras gramaticales complejas e incluyen símbolos de compuestos, expresiones alfanuméricas o representaciones genéticas complejas. Por lo tanto, aplicar enfoques de PLN a la literatura biomédica es un reto para los expertos en el campo de las ciencias computacionales, específicamente de la inteligencia artificial.
En (Lazarou, Pomeranz, y Corey, 1998) (Businaro, 2013) (Landau, 2009)se menciona que 2.2 millones de personas , entre 57 y 85 años, en USA y Europa han ingerido combinaciones potencialmente peligrosas de fármacos; se estima que las muertes por interacciones medicamentosas accidentales aumentaron en un 68 % en un período de 5 años, entre 1999 y

2004. Por otro lado, en (Johnell y Klarin, 2007) se afirma que la ingesta de varios fármacos puede aumentar el riesgo de efectos secundarios y su número está correlacionado con la cantidad de medicamentos que toma, de forma simultánea, cada paciente. La definición de interacción medicamentosa (DDI) se describe como: Un cambio que se produce en el efecto de un fármaco, provocado por otro, al ser ambos ingeridos por un paciente dentro del mismo periodo de tiempo (Mahendran y Nawarathna, 2016) (Y. Zhang y cols., 2017).
A partir de la caracterización de los DDI, es extremadamente importante la prescripción médica, debido a posibles reacciones adversas, en desmedro de la calidad de vida del paciente además que incrementan los costos de atención médica. La información sobre DDI se obtiene, mayoritariamente, de revistas especializadas que se convierten en la fuente de información principal para detectarlas. Varios son los beneficios que ofrece la extracción automática de DDI; por un lado, las industrias farmacéuticas ganan conocimiento para fabricar medicamentos más seguros y, por otro, los galenos pueden prescribir fármacos evitando los peligros que las DDI significan para el paciente.
Los primeros enfoques para la extracción automática de DDI se basan principalmente en reglas elaboradas manualmente debido a la falta de conjuntos de datos etiquetados (Segura Bedmar, Martinez, y Sánchez Cisneros, 2011). Con la introducción de los desafíos académicos de DDIExtraction en 2011 y 2013 (Segura Bedmar, Martínez, y Herrero Zazo, 2013), cuyo objetivo es mejorar sucesivamente los modelos de extracción, y la disponibilidad de conjuntos de datos DDI etiquetados, se han propuesto un mayor número de modelos basados en el aprendizaje automático. La tarea DDIExtraction 2013 se refiere al reconocimiento de drogas y la extracción de interacciones entre drogas que aparecen en la literatura biomédica. El objetivo de la tarea es proporcionar un marco común para la evaluación de las técnicas de extracción de información aplicadas al reconocimiento de sustancias farmacológicas y la detección de interacciones farmacológicas a partir de textos biomédicos (Segura-Bedmar et al., 2013, 2014) es un esfuerzo a nivel comunitario para promover la implementación y la evaluación comparativa de las técnicas de PLN.
Estos enfoques generalmente se basan en un conjunto de características cuidadosamente diseñadas para entrenar clasificadores supervisados. Un caso emblemático constituye la máquina de vectores de soporte (SVM), cuyos resultados de la extracción de DDI dependen de la calidad de las características utilizadas (Zelenko, Aone, y Richardella, 2003) (Reichartz, Korte, y Paass, 2010). Resulta arduo el proceso de diseño de características, necesario en el modelo SVM, el mismo que puede ser evitado con el uso de técnicas de aprendizaje profundo, mismas que, en los últimos años, han sido propuestas, con buenos resultados. Tales técnicas aprenden automáticamente representaciones de características, a partir de grandes

cantidades de datos no etiquetados (Bengio, Courville, y Vincent, 2013). Entre métodos, se han propuesto aquellos basados en redes neuronales convolucionales (CNN) (Suárez-Paniagua y Segura-Bedmar, 2018), (Y. Zhang y Lu, 2019) y redes neuronales recurrentes (RNN) (Hou y Ceesay, 2018), (Shen y cols., 2018), (B. Xu, Shi, Zhao, y Zheng, 2018).
Los métodos de aprendizaje profundo en la actualidad, han llegado a ser considerados como una excelente alternativa debido a que permiten extraer automáticamente las características más apropiadas para representar un problema relacionado con una tarea determinada. Los antecedentes anotados, junto con la posibilidad de encontrar nuevos métodos más eficientes, constituyen razones suficientes para dar paso al planteamiento de una hipótesis:
Es posible mejorar la exactitud predictiva de los modelos de clasificación de las frases del corpus "DDI Extraction 2013", aplicando técnicas de aprendizaje profundo.

1.1. Formulación del problema

En esta sección, se presentan algunas definiciones indispensables para introducir el concepto de extracción de información relacionada, a partir de grandes volúmenes de documentos de texto.

1.1.1. Clasificación:

En el área de la inteligencia artificial denominada aprendizaje automático, se le conoce como clasificación a la tarea de asignar una clase a un objeto que posee un patrón asociado a esa clase, dado un conjunto fijo predefinido C de clases. Este es uno de los problemas centrales en el aprendizaje automático y tiene una gran variedad de aplicaciones prácticas en PLN.

Formalmente, la clasificación como problema de aprendizaje supervisado se define como:

Sea el conjunto de entrenamiento, $D = \{(x_1, y_1), ..., (x_n, y_n)\}$

donde dado un patrón de entrada $x_i \in R^d$

dado $y_i \in Y$ que es su clase

y$_n$ indica varias instancias de entrenamiento,

el modelo de aprendizaje supervisado aprenderá una hipótesis $g : X \rightarrow Y$

del conjunto de datos D para asignar una entrada $x \in X$ a su clase $y \in Y$.

Aquí, X e Y representan los espacios de entrada y salida respectivamente.

1.1.2. Clasificación de relaciones

La clasificación de relaciones es el proceso de identificar cómo se relacionan semánticamente, las entidades dentro de un texto. En la literatura biomédica, comúnmente se proponen 4 tipos de relaciones entre fármaco - fármaco: Mecanismo, Consejo, Efecto e Interaccion. En la tarea de clasificación de relaciones, suponemos que las entidades ya han sido reconocidas y el objetivo es determinar el tipo de relación semántica que existe. En general, estas tareas se encuentran en un escalón más elevado de complejidad, porque demandan una mayor comprensión semántica de los fragmentos de texto. En nuestros experimentos, formulamos esta tarea como una clasificación supervisada , donde la finalidad es aprender una hipotesis g, que asigna una clase y a una oración de entrada dada s con dos entidades

$$e_1 \text{ y } e_2$$

es decir,

$$g(s, e_1, e_2) = y$$

En general, una oración puede tener múltiples entidades y múltiples relaciones entre ellas. En tales casos, se procede a separar y crear instancias para cada par de entidades, donde la oración seguirá siendo igual, pero las entidades objetivo serán diferentes. De esta manera, el modelo puede clasificar las relaciones entre todos los pares de entidades en una oración dada

1.2. Métodos estándares para el problema de clasificación

La clasificación de relaciones en textos no estructurados se ha modelado aplicando diferentes métodos. Los métodos basados en co-ocurrencia son los más utilizados en el campo biomédico y dominio clínico, debido a su simplicidad y flexibilidad. En el análisis de concurrencia, se entiende que, si dos entidades ocurren juntas en muchas oraciones, debe existir una relación entre ellos (Bunescu, Mooney, Ramani, y Marcotte, 2006). Sin embargo, estos métodos no pueden diferenciar entre diferentes tipos de relaciones lo cual afecta el rendimiento de este método. Diferentes enfoques estadísticos, como 'point wise', 'mutual information', 'chi-cuadrado' o 'log-likelihood ratio' se han utilizado para mejorar el rendimiento (Stapley y Benoit, 1999). Los métodos basados en reglas es otro enfoque comúnmente utilizado para la

tarea de clasificación de relaciones. Se crean reglas observando cuidadosamente los patrones sintácticos y semánticos de instancias de relación. Para mejorar el rendimiento de la técnica basada en reglas se usa el método de Bootstrapping (K.-L. Xu, 2008) el cual utiliza un pequeño número de pares de relaciones conocidas de cada tipo como semillas y utiliza estas semillas para buscar patrones en una gran cantidad de texto no anotado, de forma iterativa. Además, el método Bootstrapping genera muchos patrones irrelevantes, que pueden ser controlados por el enfoque de supervisión a distancia. Un método supervisado a distancia utiliza una gran base de conocimiento como UMLS (Unified Medical Language System) o Freebase como su entrada y extrae patrones de un corpus grande para todos los pares de relaciones presentes en la base de conocimiento. La ventaja de bootstrapping y métodos supervisados a distancia sobre los métodos supervisados es que no requieren grandes cantidades de datos de entrenamiento etiquetados manualmente que generalmente es difícil de obtener. Los métodos basados en características usan oraciones con entidades predefinidas para construir un vector de características, construido a partir de la extracción de las mismas. La extracción de características se basa principalmente en la producción de herramientas de dominio específico. Los vectores de características extraídos se utilizan para determinar la clase correcta de la relación usando una técnica de clasificación dada. Los Métodos del núcleo son una extensión de los métodos basados en características que utilizan las funciones del núcleo para explotar la información sintáctica enriquecida , como en árboles de análisis (Zeng y cols., 2014). Los resultados más modernos sean obtenido por esta clase de métodos. Sin embargo, el rendimiento de la función y los métodos basados en el núcleo dependen en gran medida de la selección adecuada del conjunto de características, que no solo es una tarea tediosa y lenta, sino que también requiere conocimiento del dominio y depende de otros sistemas de PLN. A menudo, estos métodos resultan en una gran cantidad de características y pueden verse afectados por el problema de la maldición de la dimensionalidad (Collobert y cols., 2011). Otro problema que enfrentan estos métodos es que el proceso de extracción de características debe ajustarse a las características de la fuente de datos

1.3. Objetivo de investigación

Proponer un modelo de extracción de interacción medicamentosa a partir del corpus “DDI Extraction 2013”, y compararlos con modelos reportados en la literatura técnica.
Objetivos específicos:

- Identificar el estado del arte con relacion a los modelos para la extraccion de DDI

existentes en la literatura técnica.

- Plantear un modelo alternativo para la tarea de extracción de DDI.
- Realizar un analisis comparativo del desempeño del modelo propuesto con el de otros modelos, a partir de su aplicación en el corpus “DDI Extraction 2013”.

1.4. Esquema de la tesis

Brevemente describiremos como se encuentra organizado este documento: Iniciamos revisando algunas definiciones, técnicas y desafíos de extracción de interacciones medicamentosas y los conceptos de clasificación de relaciones en PLN. Capítulo 2, explica los pasos dados para la revisión de la literatura sobre modelos de extracción de DDI. El capítulo 3, Describe la metodología y arquitectura del modelo utilizado. El capítulo 4, presenta el proceso seguido para realizar los experimentos y los resultados de la evaluación del modelo. El capítulo 5, contiene las conclusiones del trabajo realizado. El capítulo 6, sugiere algunos trabajos futuros y finalmente el capítulo 7, detalla las referencias bibliográficas.

Capítulo 2

Revision de la Literatura

2.1. Introducción

Una interacción medicamentosa (DDI) es un cambio que se produce en el efecto de un fármaco, provocado por otro, al ser ambos ingeridos por un paciente dentro del mismo periodo de tiempo. La caracterización de los DDI es extremadamente importante en la prescripción médica, debido a posibles reacciones adversas, en desmedro de la calidad de vida del paciente. Por otro lado, las enfermedades coexistentes son también motivo de alerta. Estos dos temas confluyen, provocando una sobrecarga de información para el médico a la hora de prescribir fármacos, pues no solo deberá concentrarse en los medicamentos que tratan la enfermedad del paciente, sino también en sus interacciones que a veces resultan ser de mucha complejidad (V. Miranda y cols., 2011) (Bond y Raehl, 2006).
En (Lazarou y cols., 1998) (Businaro, 2013) (Landau, 2009)se mencionan que 2.2 millones de personas en USA y Europa, entre 57 y 85 años, han ingerido combinaciones potencialmente peligrosas de fármacos; también, se estima que las muertes por interacciones medicamentosas accidentales aumentaron en un 68 % en un período de 5 años, entre 1999 y 2004 [6]. Por otro lado, en [7] se afirma que las DDI también pueden aumentar el riesgo de efectos secundarios, y se demuestra que su número está correlacionado con la cantidad de medicamentos que toma, de forma simultánea, cada paciente. La información sobre DDI se obtiene, mayoritariamente, de revistas especializadas que se convierten en la fuente de información principal para detectarlas. Varios son los beneficios que ofrece la extracción automática de DDI; por un lado, las industrias farmacéuticas ganan conocimiento para fabricar medicamentos más seguros y, por otro, los galenos pueden prescribir fármacos evitando los peligros que las DDI significan para el paciente.
Los primeros enfoques para la extracción automática de DDI se basan principalmente

en reglas elaboradas manualmente debido a la falta de conjuntos de datos etiquetados (Segura Bedmar y cols., 2011). Con la introducción de los desafíos de DDIExtraction en 2011 y 2013 (Segura Bedmar y cols., 2013), cuyo objetivo es mejorar sucesivamente los modelos de extracción, y la disponibilidad de conjuntos de datos DDI etiquetados, se han propuesto un mayor número de modelos basados en el aprendizaje automático. Estos enfoques generalmente se basan en un conjunto de características cuidadosamente diseñadas para entrenar clasificadores supervisados. Un caso emblemático constituye la máquina de vectores de soporte (SVM), cuyos resultados de la extracción de DDI dependen de la calidad de las características utilizadas (Zelenko y cols., 2003) (Reichartz y cols., 2010). Resulta arduo el proceso de diseño de características, necesario en el modelo SVM, el mismo que puede ser evitado con el uso de técnicas de aprendizaje profundo, mismas que, en los últimos años, han sido propuestas, con buenos resultados. Tales técnicas aprenden automáticamente representaciones de características, a partir de grandes cantidades de datos no etiquetados (Bengio y cols., 2013). Entre otros modelos para obtener, de forma automática, vectores de características de oraciones, útiles para la extracción de DDI, con un alto rendimiento, se han propuesto aquellos basados en redes neuronales convolucionales (CNN) (Suárez-Paniagua y Segura-Bedmar, 2018), (Y. Zhang y Lu, 2019) y redes neuronales recurrentes (RNN) (Hou y Ceesay, 2018), (Shen y cols., 2018), (B. Xu, Shi, Zhao, y Zheng, 2018).
En este trabajo, además de presentar los fundamentos teóricos de la extracción de las DDI, se reporta una revisión sistemática de la literatura que resume el desarrollo reciente de los modelos para recuperar información presente en la literatura biomédica, y se sugiere futuras direcciones de investigación.

2.2. Extracción automática de interacciones medicamentosas

La tarea de extracción de interacciones medicamentosas se alivianó con la aparición de los desafíos de DDIExtraction en 2013(Segura Bedmar y cols., 2013), se resolvió el problema de la falta de datos etiquetados. La extracción de interacciones medicamentosas de los textos biomédicos consiste en clasificar los tipos de interacción (Mecanismo, Efecto, Consejo, Interacción, Negativo) entre dos entidades farmacológicas en una oración de la literatura biomédica. Por ejemplo:
`Pantoprazole (entidad1) has a much weaker effect on clopidogrel (entidad2) pharmacokinetics and on platelet reactivity during concomitant use.`

`entidad1 = 'pantoprazol' y entidad2 = 'clopidogrel'`
El objetivo es reconocer automáticamente que esta oración expresa una interacción de tipo "mecanismo" entre entidad1 y entidad2. En los últimos años, se han dedicado esfuerzos considerables a la extracción de DDI. En este desafío tanto los enfoques basados en reglas como los enfoques de aprendizaje automático dependen en gran medida de la ingeniería de características y un buen conocimiento del dominio para lo cual se necesita la ayuda de un experto en el campo, adicionalmente que sufren de un procesamiento redundante de características. La aparición de enfoques de aprendizaje profundo alivia este problema de manera efectiva al definir automáticamente el conjunto de características más adecuado para una tarea en particular. A continuación, se resumen conceptos de técnicas procesamiento de lenguaje natural para extracción de interacciones medicamentosas (ER), necesarios para el análisis de la literatura técnica en el tema, así como los procesos de minería de texto involucrados, y los modelos de redes de neuronas presentes en la literatura especializada.

2.2.1. Procesamiento de lenguaje natural

El procesamiento del lenguaje natural (PLN) cubre varias técnicas que constituyen pasos de preprocesamiento para las tareas descritas en la siguiente sección. Las técnicas de PLN frecuentemente se combinan para obtener un mayor rendimiento.

1.- Tokenización: Tiene como fin dividir el texto en tokens para procesarlos individualmente o como una secuencia. El camino más directo de tokenización es dividir el texto de entrada por espacios en blanco. Sin embargo, en la literatura científica biomédica, se debe tener en cuenta que existen entidades complejas como los términos del fenotipo humano (compuesto de varias palabras), genes (representados por símbolos) y otros tipos de entidades estructuradas. Estas entidades tienden a ser complejas, algunos investigadores utilizan un algoritmo de compresión (Sennrich, Haddow, y Birch, 2015), codificación de par de bytes (BPE) que representa vocabularios abiertos a través de un vocabulario de tamaño variable de secuencias de caracteres de longitud variable, lo que lo hace adecuado para modelos de redes neuronales.

2.- Lematización: Palabras que comparten una raíz común, que hacen referencia al mismo concepto básico. Ej. Matriz, matrices son palabras con el mismo lema (Schütze, Manning, y Raghavan, 2008). La raíz puede corresponder solo a un fragmento de una palabra, pero el lema es siempre una palabra real. Por ejemplo, la raíz de la palabra matriz es matri y el lema

es matriz.

3.- Etiquetado (Part-of-Speech): consiste en asignar cada palabra de una oración a la categoría a la que pertenece teniendo en cuenta su contexto (por ejemplo, verbo o preposición). Cada palabra puede corresponder a más de una categoría. Esta característica es útil para obtener información sobre el papel de una palabra en una oración dada.

4.- Árbol de análisis: Un árbol sintáctico refleja cómo los componentes de una frase se estructuran en diferentes categorías sintácticas. Representa la estructura sintáctica de una oración. Hay dos tipos diferentes de árboles de análisis: árboles de análisis basados en la circunscripción y árboles de análisis basados en la dependencia. Los árboles de análisis basados en circunscripciones de las gramáticas constitutivas, distinguen entre nodos terminales y no terminales. Los nodos interiores están etiquetados por categorías no terminales de la gramática, mientras que los nodos hoja están etiquetados por categorías terminales (Tai, Socher, y Manning, 2015) (Aho, Sethi, y Ullman, 1986). Los árboles de análisis basados en la dependencia de las gramáticas de dependencia ven todos los nodos como terminales, lo que significa que no reconocen la distinción entre categorías terminales y no terminales. En promedio, son más simples que los árboles de análisis basados en la circunscripción porque contienen menos nodos.

2.2.2. Tareas principales de minería de texto

La minería de texto se utiliza para identificar y extraer información a partir de grandes volúmenes de texto no estructurado o altamente heterogéneo (Westergaard, Stærfeldt, Tønsberg, Jensen, y Brunak, 2018).Igualmente, se usa para extraer datos y relaciones de forma estructurada, que puede utilizarse para etiquetar bases de datos especializadas y permitir así la transferencia de conocimiento entre dominios (Fleuren y Alkema, 2015). Se puede considerar a la minería de texto como un subcampo de la minería de datos. Por ende, los algoritmos de minería de datos se pueden aplicar si transformamos el texto en una representación de datos adecuada, es decir, vectores numéricos. En los últimos años las herramientas de minería de texto han evolucionado sorprendentemente en número y calidad, aún hay muchos desafíos en la aplicación de minería de texto a la literatura científica biomédica.Las principales tareas de la minería de texto necesaria para ER son: a) Reconocimiento de entidades con nombre, que busca reconocer y clasificar las entidades mencionadas en el texto; b) Enlace de entidades con nombre, que asigna las entidades reconocidas, a entradas en una base de conocimiento

dada; y, c) Extracción de relaciones, que identifica relaciones entre entidades dentro de un texto (Singhal, Simmons, y Lu, 2016).
El flujo de trabajo de un sistema de extracción de relaciones típico se presenta en la fig. 2.1, gráfico obtenido de (Alves y Wijnholds, 2018). DET es un determinante, NP es un sustantivo, VP es un verbo, AD es un adjetivo y PP es una preposición.

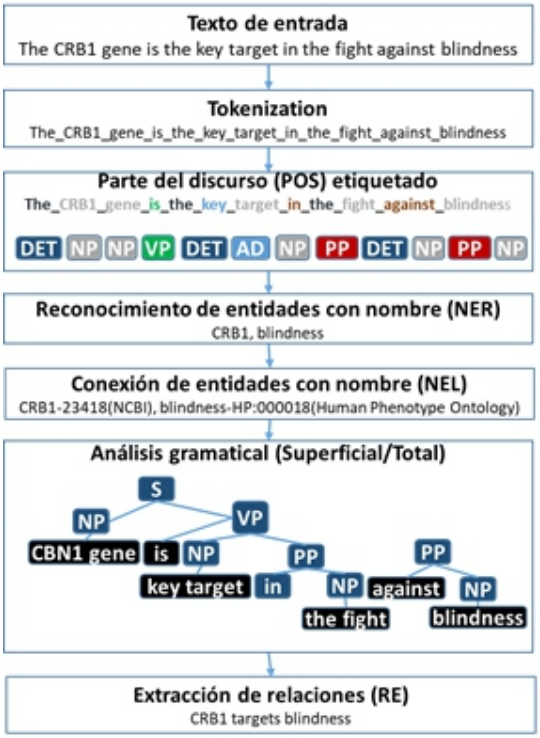

Figura 2.1: Flujo de trabajo de un sistema RE simplificado.

2.3. Redes Neuronales para ER

Los sistemas actuales de aprendizaje profundo proporcionan potentes capacidades de aprendizaje automático que no necesitan una amplia ingeniería de características para proporcionar un rendimiento adecuado (Goodfellow, Bengio, y Courville, 2016). Aun cuando los métodos de aprendizaje profundo logran un buen rendimiento del uso de una gran cantidad de datos de entrenamiento necesarios para construir modelos, su utilidad se basa en su capacidad para aprender representaciones de propósito general de vastos cuerpos sin etiquetar (Mikolov, Sutskever, Chen, Corrado, y Dean, 2013). Estas representaciones proporcionan información estructurada para el posterior análisis de aprendizaje automático que toma de

manera efectiva la semántica léxica sin requerir que los investigadores utilicen enfoques de procesamiento de lenguaje natural para representar ese significado explícitamente como características.
El aprendizaje profundo permite que los modelos computacionales compuestos de múltiples capas de procesamiento aprendan representaciones de datos con múltiples niveles de abstracción (LeCun, Bengio, y Hinton, 2015). Esto ha mejorado dramáticamente el rendimiento del aprendizaje automático en muchos dominios, como la visión por computadora (Tompson, Jain, LeCun, y Bregler, 2014), el procesamiento de lenguaje natural (Sutskever, Vinyals, y Le, 2014) y el reconocimiento de voz (Hinton, Deng, y cols., 2012) y también ha demostrado un gran rendimiento en la salud y en los dominios médicos, en especial en la tarea de extracción de interacciones medicamentosas. A continuación tenemos algunas arquitecturas profundas de uso común:

Redes neuronales recurrentes (RNN)
Son una extensión de las redes neuronales avanzadas para modelar datos secuenciales, como series temporales (Lipton, Kale, Elkan, y Wetzel, 2015) y texto en lenguaje natural (Jagannatha y Yu, 2016). En particular, la estructura recurrente de las redes neuronales recurrentes forman una herramienta muy adecuada para modelar series temporales. Se trata de un tipo de redes con una arquitectura que implementa una cierta memoria y, por lo tanto, un sentido temporal. Esto se consigue implementando algunas neuronas que reciben como entrada la salida de una de las capas e inyectan su salida en una de las capas de un nivel anterior a ella, lo que los convierte en la arquitectura preferida para varias tareas de modelado en la extraccion de información de literatura biomédica. Ver figura 2.2. Se puede procesar una secuencia de vectores mediante la aplicación de la siguiente formula:

$$h_t = F_w(h_{t-1}, x_t)$$

h_t : Nuevo estado
F_w : Función con parámetros w
h_{t-1} : Estado antiguo
x_t : Vector de entrada en algún paso del tiempo

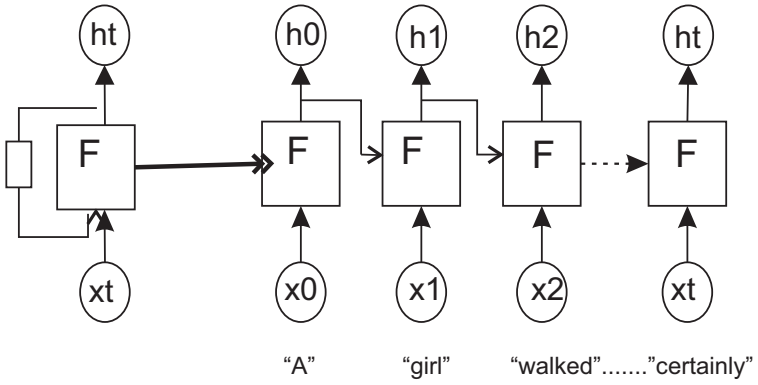

Figura 2.2: Red Neuronal Recurrente.

Redes de memoria a corto plazo (LSTM) Son una alternativa a la RNN normal (Hochreiter y Schmidhuber, 1997). Las LSTM son un tipo de RNN que maneja dependencias largas (por ejemplo, oraciones), lo que hace que este clasificador sea más adecuado para el dominio biomédico, donde las oraciones suelen ser largas y descriptivas. Los LSTM bidireccionales usan dos capas LSTM, en cada paso, una que lee la oración de derecha a izquierda y otra que lee de izquierda a derecha. La salida combinada de ambas capas produce una puntuación final para cada paso. Los LSTM bidireccionales tienen rendimiento mejores resultados que los LSTM tradicionales cuando se aplican a los mismos conjuntos de datos (S. Zhang, Zheng, Hu, y Yang, 2015). Ver figura 2.3

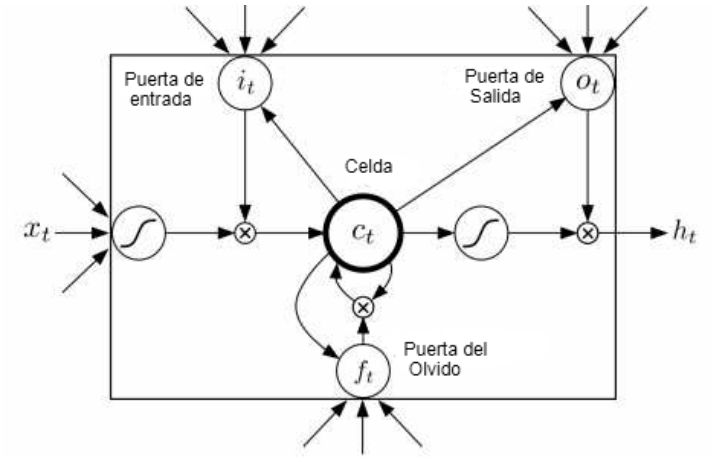

Figura 2.3: Red Neuronal que maneja largas dependencias.

No supervisado ‘embedding’
Un modelo especialmente eficiente desde el punto de vista computacional es Word2vec, tiene dos variantes: la bolsa de palabras continua (CBOW) que predice el objetivo en los

contextos vecinos, y el Skipgram que predice los contextos vecinos. En particular, con el modelo “Skip-Gram” lo que se quiere decir es: dado un conjunto de frases (también llamado corpus) el modelo analiza las palabras de cada sentencia y trata de usar cada palabra para predecir que palabras serán vecinas. Por ejemplo, a la palabra “finalin” le seguirá “forte” con más probabilidad que cualquier otra palabra. El objetivo de estos modelos es integrar terminologías de diferentes dominios en el mismo espacio para descubrir las relaciones entre ellos (por ejemplo, relaciones entre enfermedades y medicamentos) (Choi y cols., 2016). ver figura 2.4 tomada de (SrirangamSridharan, Srivatsa, Ganti, y Simpkin, 2018)

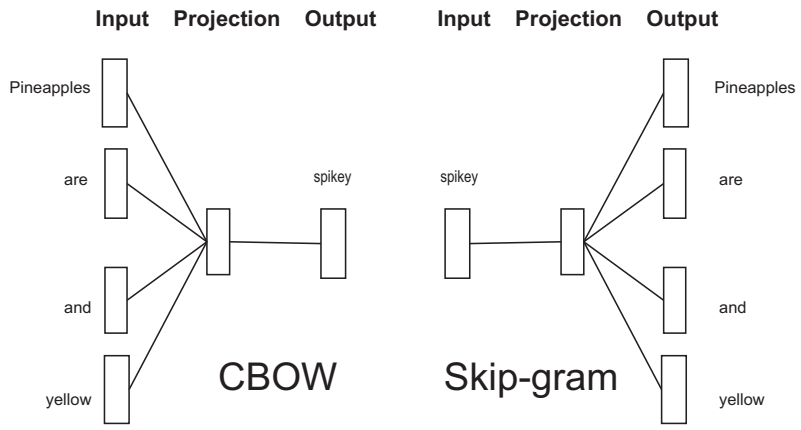

Figura 2.4: Modelo Word2vec.

Redes neuronales convolucionales (CNN)
En análisis de imagen, voz y video, las CNN explotan las propiedades locales de datos (estacionariedad y la composicionalidad a través de estadísticas locales) y utilizar capas convolucionales y de agrupación para extraer progresivamente patrones abstractos. Los CNN funcionan de la siguiente manera: las capas convolucionales conectan múltiples filtros locales con sus datos de entrada (datos sin procesar o salidas de capas) y producen características locales invariantes de transcripción (Nguyen, Tran, Wickramasinghe, y Venkatesh, 2016). Entonces, la agrupación de capas reduce progresivamente el tamaño de la salida para evitar sobreajuste. Aquí, tanto la convolución como el agrupamiento se realizan localmente, tal que (en el análisis de la imagen) la representación de una característica local no influirá en otras regiones. ver figura 2.5.

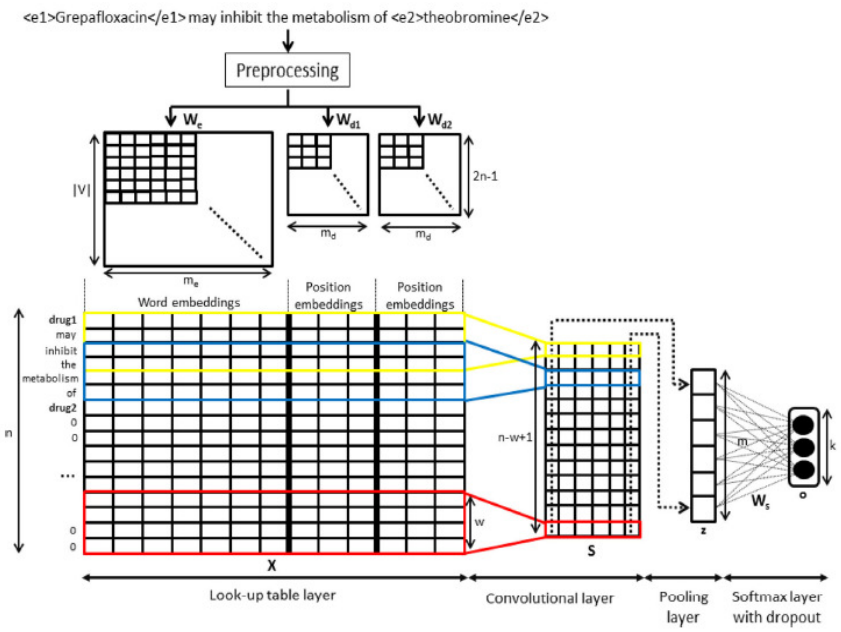

Figura 2.5: Modelo CNN. Figura obtenida del trabajo [Suárez-Paniagua, Segura-Bedmar, y Martínez, 2017]

Efecto de la variación de hiperparámetros en el rendimiento

Las redes neuronales profundas han avanzado rápidamente en los últimos años. Sin embargo, todavía parece un arte negro para algunos investigadores utilizarlo de manera eficiente. La razón de esta complejidad es que obtener un resultado consistente y sobresaliente de una arquitectura profunda requiere la optimización de muchos parámetros conocidos como hiperparámetros. El ajuste de hiperparámetros es una tarea esencial en el aprendizaje profundo, que puede realizar cambios significativos en el rendimiento de la red. En este estudio se proporcionamos una lista de hiperparámetros para ajustar, además de su impacto en el rendimiento de la red. Se provee de este listado a los investigadores interesados en modificar su arquitectura profunda para obtener un mejor rendimiento con el menor esfuerzo (Aghaebrahimian y Cieliebak, 2019) (Pasi y Naik, 2016) (Quan, Hua, Sun, y Bai, 2016).

- Optimizador:

  El trabajo de un optimizador es minimizar la pérdida en la función objetivo. Los métodos más ampliamente usados son los basados en gradiente y descenso de gradiente estocástico (SGD). Existen variantes de estos optimizadores como: Adagrad (Duchi, Hazan, y Singer, 2011), RMSProp (Hinton, Srivastava, y Swersky, 2012), Adam (Kingma y Ba, 2014). Se obtiene mejor rendimiento con el optimizador Adam.

- Pooling:

  El pooling se ha demostrado como una herramienta útil para extraer la mayoría de las características relevantes, ya sea en un modelo CNN después de los filtros convolucionales o en un modelo RNN después de las capas recurrentes. Se investigó tres tipos de pooling: average, max y la unión de ambos.

- Control de gradiente:

  Las derivadas que se calculan en la retropropagación en el tiempo de entrenamiento en un modelo con muchas capas se hacen cada vez más pequeñas hasta el punto de desaparecer. Esto es particularmente cierto para los RNN que tienen una gran cantidad de capas. Esto hace que el entrenamiento sea difícil y lento. Existen dos mecanismos ampliamente practicados llamados recorte de gradiente (Mikolov, 2012) y normalización de gradiente(Pascanu, Mikolov, y Bengio, 2013) para abordar este problema conocido como desaparición de gradiente. Establecemos el gradiente mecanismo de control con la mejor de las arquitecturas profundas.

- Clasificador:

  La última capa en un modelo de clasificación se considera la capa más crucial ya que todas las características calculadas en esta capa se proyectan a sus clases apropiadas. Por lo tanto, la elección de esta capa tiene un impacto esencial en el rendimiento de la red. La elección de esta capa depende en gran medida de los supuestos que hacemos sobre la tarea en cuestión. Si las etiquetas se distribuyen de forma independiente, el Sigmoid y el Softmax producen mejores resultados, mientras que si están condicionadas por sus etiquetas adyacentes (por ejemplo, etiquetado POS), el campo aleatorio condicional (CRF) (Lafferty, McCallum, y Pereira, 2001) funciona mejor. Si esperamos una distribución multinomial sobre las etiquetas, Softmax es el mejor clasificador para elegir, mientras que si esperamos una distribución de Bernoulli, Sigmoid es la elección correcta.

- Drop out:

  Las redes neuronales profundas tienden a memorizarse o sobreajustarse, lo cual es un comportamiento indeseable ya que estamos interesados principalmente en la capacidad de la red para generalizar. Drop out es una herramienta efectiva para mejorar la generalización(Srivastava, Hinton, Krizhevsky, Sutskever, y Salakhutdinov, 2014). La primera técnica conocida como drop out simple se propuso como un mecanismo que elimina aleatoriamente las conexiones entre capas profundas. (Gal y Ghahramani,

2016) propusieron un nuevo mecanismo de drop out llamado variacional, que mejora el drop out simple al definir máscaras estáticas para eliminar las conexiones entre capas profundas ('capa intermedia'), así como entre las unidades dentro de capas profundas ('intracapa' ').

- Gaussian noise:

  Como se describe en el trabajo de(Papadaki, 2017), en este método, en lugar de generar nuevas oraciones aumentadas, insertamos perturbaciones en las incrustaciones de palabras en la capa de incrustación de nuestro modelo. Para cada inserción de palabras, les agregamos un valor de ruido gaussiano según la siguiente ecuación: $w_j0 = w_j x j_(3)$ donde, w_j es la inserción de palabras original, w_j0 es la inserción de palabras resultante y x_j es el valor de ruido aleatorio. Los valores para cada elemento vectorial de ruido se muestrearon a partir de la distribución normal truncada con $\mu = 0, \sigma = 1$ y con un rango entre 0 y 0.3. A continuación, se seleccionan al azar, los elementos del vector con una probabilidad de 0.3 y el resto del vector se establece en cero. El vector resultante es considerado como un vector de ruido. Los valores se mantuvieron en un rango pequeño para evitar que la incrustación de palabras resultante se aleje demasiado del espacio de incrustación de palabras contextuales. Una excepción en este método a la de los otros métodos de aumento es que, insertamos perturbaciones en todas las palabras, independientemente de su etiqueta POS. Hemos tratado de explorar las fortalezas y debilidades de estos métodos en los experimentos.

2.4. METODOLOGÍA

Establecidos los fundamentos de ER, a continuación, se reporta una revisión sistemática de la literatura que resume el desarrollo reciente de los modelos, presente en la literatura biomédica.
El esquema desarrollado para la presente revisión está basado en la metodología de revisiones sistemáticas de Barbara Kitchenham (Kitchenham, 2004). La selección y extracción de información, se detalla a continuación:
a. Preguntas de investigación.
b. Método de revisión.

- Fuentes y estrategias de búsqueda.
- Cadenas de búsqueda.

- Criterios de selección de estudios.
- Extracción de información.

c.Estudios incluidos y excluidos.

2.4.1. Preguntas de investigación

El alcance de este trabajo se abordó con artículos relacionados con el challenge DDI 2013. Las preguntas de investigación son:

- RQ1: ¿Qué métodos de extracción de interacciones medicamentosas son los más usados?
- RQ2: ¿Cuáles son las ventajas y limitaciones de estos métodos?
- RQ3: ¿Cuál es la técnica más usada para la evaluación del desempeño?

2.4.2. Método de Revisión

2.4.2.1. Fuentes y Estrategias de Investigación

Las siguientes bases de datos de investigación fueron utilizadas para la búsqueda: PubMed, IEEEXplore and Web of Science.

2.4.2.2. Cadenas de búsqueda

Basada en la pregunta de investigación, las cadenas de búsquedas que se definieron son: (Drug interaction OR DDI) AND (extraction) AND (Data mining OR intelligent OR Predict* OR Classificat* OR Cluster* OR Associat*) AND (Model* OR algorithm* OR technique* OR rule* OR Method*)
Criterios de inclusión en el estudio: La busqueda fue ejecutada con el siguiente criterio:

- La fecha de publicación se consideró a partir del año 2018 debido a la relevancia de los enfoques que presentan un mejor aporte a nuestro trabajo.
- Los resultados de la investigación son solo en el área de Ciencias e Informática.

- Las producciones científicas son estudios primarios (artículos de conferencias, artículos de revistas).
- La búsqueda por defecto estará en el idioma inglés debido a su relevancia científica.
- Los estudios deben tener información relevante a la pregunta de investigación.

Criterios de exclusión del estudio: Los artículos con estas características fueron eliminados: 1. No disponible. 2. Los duplicados. 3. Los incompletos. 4. Los que no contestan las preguntas de investigación.

2.4.2.3. Criterios de selección de estudios.

Una vez que se obtuvieron los resultados, la selección de los estudios primarios se basa en considerar lo siguiente:

- Hay información actual del mecanismo de extracción de interacciones medicamentosas de la literatura biomédica en el resumen.
- Hay información relevante para la revisión en la conclusión o introducción.

2.4.2.4. Extracción de información

Para cada artículo seleccionado, se sintetizará al menos uno de los siguientes elementos:

- Métodos eficientes para extracción de interacciones medicamentosas DDI de la literatura biomédica.
- Resultados.
- Conclusiones relevantes.

2.4.3. Estudios incluidos y excluidos

En el primer proceso de extracción, 332 casos fueron obtenidos. Luego, otro proceso de selección fue llevado acabo en el cual 14 relevantes articulos fueron obtenidos. La tabla 2.1 nos muestra su distribución:
El siguiente diagrama de flujo describe la seleccion de articulos incluidos y excluidos, ver figura 2.6

Tabla 2.1: Articulos incluidos excluidos

Fuente	Resultados	(+-)Relevante	No Relevantes	Repetidos	Incompletos	Relevantes
IEEE	58	15	25	0	15	3
Pubmed	235	10	200	0	14	10
WebOfScience	40	10	15	14	0	1
Total	333	35	240	14	29	15

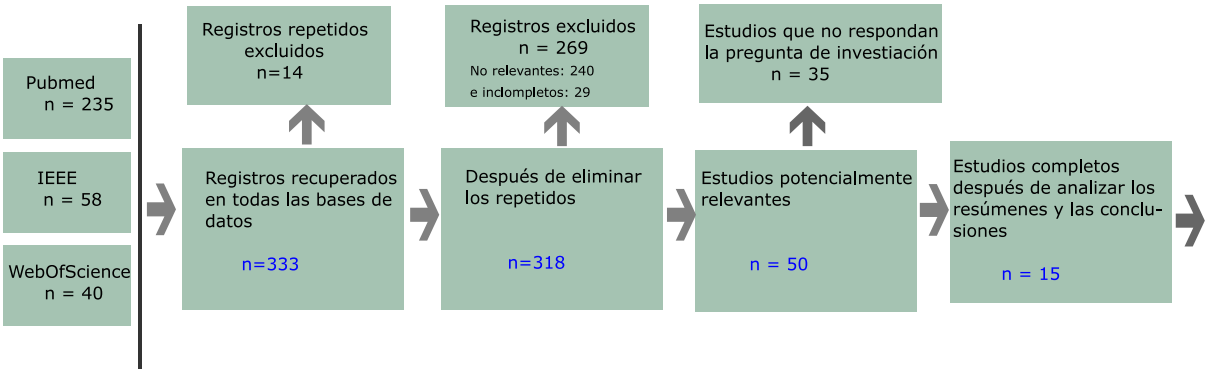

Figura 2.6: Diagrama de flujo: el proceso utilizado para seleccionar los artículos incluidos y excluidos.

2.5. Resultados obtenidos

De los estudios seleccionados, se identificó evidencia relevante para responder las preguntas de investigación, como se describe a continuación.

RQ1: ¿Qué modelos de extracción de interacciones medicamentosas son los más usados?
Para abordar los desafíos de extracción de interacciones medicamentosas de la literatura biomédica se utilizan varios modelos de arquitecturas de aprendizaje profundo, así como redes neuronales totalmente conectadas.
En la tabla 2.2 y figura 2.7, se muestra el modelo usado y el rendimiento general obtenido en la experimentación. Los modelos basados en redes neuronales generalmente logran un mejor rendimiento en este tipo de tareas. Por ejemplo, en el trabajo (X. Sun y cols., 2018) se empleó una red convolucional profunda (DCNN) y se propone una arquitectura novedosa utilizando pequeñas convoluciones, que toma la literatura biomédica como insumo y opera directamente en el nivel de palabra para obtener las características convolucionales

Tabla 2.2: Modelos de extracción de interacciones medicamentosas DDI

Cod.	Ref.	Métodos más usado	F1 score
SR1	(Suárez-Paniagua y Segura-Bedmar, 2018)	CNN	0.644
SR2	(Hou y Ceesay, 2018)	LSTM	-
SR3	(B. Xu, Shi, Zha, y cols., 2018a)	Bi-LSTM	0.712
SR4	(Shen y cols., 2018)	Bi-LSTM	0.919
SR5	(Y. Zhang y Lu, 2019)	Bi-LSTM and CNN	0.588
SR6	(Li y cols., 2019)	Bi-LSTM	0.743
SR7	(X. Sun y cols., 2019)	CNN	0.754
SR8	(B. Xu, Shi, Zhao, y Zheng, 2018)	Bi-LSTM	0.711
SR9	(Y. Zhang y cols., 2017)	Bi-LSTM	0.729
SR10	(Zhou, Miao, y He, 2018)	Bi-LSTM	0.729
SR11	(B. Xu, Shi, Zha, y cols., 2018b)	Bi-LSTM	0.712
SR12	(X. Sun, Ma, Du, Feng, y Dong, 2018)	CNN	0.845
SR13	(Zhao, Wang, Lin, Yang, y Zhang, 2019)	Bi-LSTM	0.708
SR14	(Lamurias, Sousa, Clarke, y Couto, 2019)	BO-LSTM.	0.751
SR15	(Park, Cho, Park, y Park, 2019)	MCPCNN.	0.714

basadas en incrustaciones. Finalmente, con estas características se logra una puntuación del estadístico F-score de 0.845. En el trabajo de (Park y cols., 2019)se utiliza un modelo PCNN para caracterizar el proceso de extracción y capturar la información estructural de la oración que contiene las entidades farmacológicas, se extrae satisfactoriamente las relaciones entre un par de entidades, este modelo usa cinco versiones de incrustaciones de palabras como canales para enriquecer la terminología desconocida, este modelo llega aun F-score de 0.714. En los trabajos (Hou y Ceesay, 2018), (B. Xu, Shi, Zha, y cols., 2018a), (Shen y cols., 2018), (Li y cols., 2019), (B. Xu, Shi, Zhao, y Zheng, 2018), (Y. Zhang y cols., 2017), (Zhou y cols., 2018), (B. Xu, Shi, Zha, y cols., 2018b), (Zhao y cols., 2019), (Lamurias y cols., 2019) se utiliza LSTM (Long Short Term Memory) simple y bidirecccional y se obtiene un F-score promedio de 0.758. La tabla III muestra el rendimiento comparativo de algunas estrategias que usan los modelos para mejorar su desempeño. Cuando se emplea el modelo de Bi-LSTM en la secuencia de la oración y SDP (Shortest Dependency Path), se logra una puntuación F-score

de 0.696. La secuencia de oraciones contiene todas las palabras, mientras que SDP solo conserva las palabras vitales de la oración. Por lo tanto, la secuencia de oraciones contiene información léxica y sintáctica más rica que SDP. Esta es la razón principal por la que el modelo Bi-LSTM logra un mayor rendimiento en la secuencia de oraciones que SDP. Cuando se emplea el modelo jerárquico de Bi-LSTM en la secuencia de la oración, la puntuación del F-score mejora de 0.696 a 0.707. Los bi-LSTM jerárquicos pueden capturar características más valiosas dividiendo la oración en tres subsecuencias, y mejorando el rendimiento eficazmente. Cuando se agrega el mecanismo de atención de inserción, el puntaje F-score mejora a 0.717. Esto indica que la atención integrada puede identificar y aumentar el peso de las palabras clave en la oración, lo que mejora aún más el rendimiento del modelo RNN para la extracción de DDI. Además, el modelo mejora el F-score de 0.729, al integrar el SDP en la secuencia de la oración.

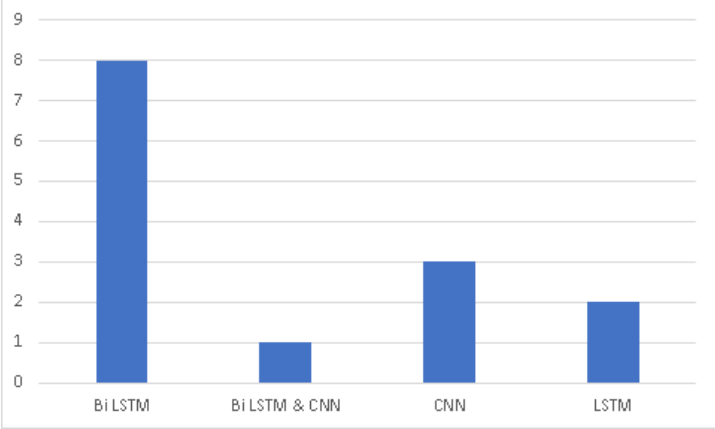

Figura 2.7: Modelos más utilizados

En el trabajo (X. Sun y cols., 2019) se propone una nueva red neuronal convolucional híbrida recurrente (RHCNN) para la extracción de DDI de la literatura biomédica. En la capa de incrustación, los textos que mencionan dos entidades (dos medicamentos) se representan como una secuencia de incrustaciones semánticas e incrustaciones de posición. En particular, la integración semántica completa se obtiene mediante la fusión de información entre la inserción de una palabra y su información contextual que se aprende por estructura recurrente. A continuación, la red neuronal convolucional híbrida se emplea para aprender las características a nivel de oración, que consiste en aquellas de contexto local de palabras consecutivas y las características de dependencia entre palabras separadas para la extracción de DDI, se obtuvo una puntuación de F-score de 0.754.
Los hallazgos encontrados en el trabajo (Lamurias y cols., 2019) demuestra que, además del

alto rendimiento de las técnicas actuales de aprendizaje profundo, las ontologías específicas de dominio pueden ser útiles para mitigar la falta de datos etiquetados.

Tabla 2.3: El efecto de la estrategia en el rendimiento

Estrategia	P	R	F1
Sequence BI-LSTM	0.702	0.691	0.696
Hierarchy BI-LSTM	0.725	0.689	0.707
Hierarchy BI-LSTM + Att.	0.73	0.703	0.717
Hierarchy BI-LSTM + Att. + SDP	0.741	0.718	0.729

RQ2: ¿Cuáles son las ventajas y limitaciones de estos métodos?

Los modelos basados en redes neuronales no solo aprenden automáticamente de la representación de características de la oración, sino que también logran un buen rendimiento. Esto indica la efectividad y el potencial de los métodos basados en redes neuronales para la extracción de DDI. Entre los modelos basados en redes neuronales, las CNN y las RNN son los dos modelos que se usan comúnmente para la tarea de extracción de DDI.

En la tabla 2.4, se muestra las fortalezas y debilidades de los métodos que se utilizan para extraccion de DDI.

Tabla 2.4: Ventajas y desventajas de métodos para extracción de DDI

Método	Ventajas	Desventajas
RNN	Habilidad para modelar relaciones complejas.	Necesidad de series temporales más largas para entrenamiento y prueba.
	Capacidad para aprender relaciones no lineales.	Difícil establecimiento de relaciones causales entre variables. Se comporta como una caja negra.
	Alta capacidad de clasificación, predicción y tolerancia a fallos.	
CNN	Puede aprender estructuras arborescentes.	Pobre desempeño en las oraciones en las que las palabras relacionadas no son adyacentes.
	Tienen alta velocidad.	

RQ3: ¿Cuál es la técnica más usada para la evaluación del desempeño?
Las técnicas de evaluación de desempeño predictivo de los modelos de clasificación se dividen en dos grupos: Métodos de escala individual y métodos gráficos (Kaymak, Ben-David, y Potharst, 2012).Los métodos de escala individual utilizan como indicadores de rendimiento los valores de sensibilidad (R), especificidad y precisión (P). La Sensibilidad, o Tasa de Verdaderos Positivos, es la proporción de casos positivos que fueron correctamente identificadas por el algoritmo. La Especificidad es la Tasa de Verdaderos Negativos; se trata de los casos negativos que el algoritmo ha clasificado correctamente. La precisión es la proporción entre las predicciones correctas que ha hecho el modelo y el total de predicciones. Los métodos gráficos utilizan figuras que representan de forma detallada el desempeño del modelo. Entre las curvas más conocidas están: la curva ROC (Receiver Operating Characteristic Curve), la línea de costo y el gráfico de elevación. Por lo general, los métodos gráficos pueden ser menos claros que los métodos de escala individual, cuando se utiliza en el proceso de análisis (Prati, Batista, y Monard, 2011).

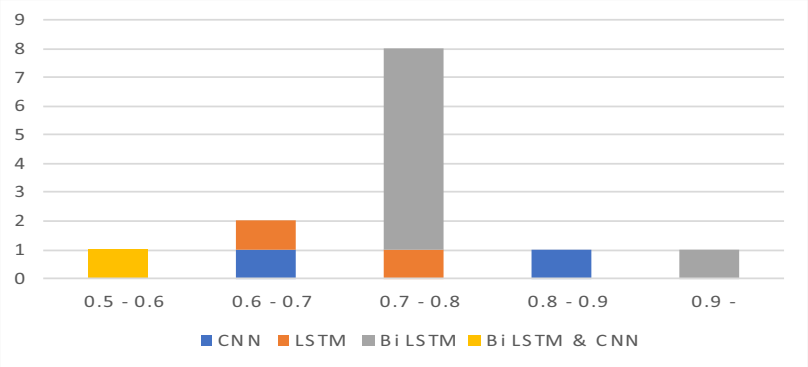

Figura 2.8: Evaluación métrica F1 score

Al revisar los trabajos seleccionados, se observa que, a fin de evaluar el desempeño predictivo de los modelos de clasificación, los indicadores más utilizados son la sensibilidad, la especificidad y la precisión, con 75 %. Prácticamente, todos los trabajos combinan los indicadores anteriores y presentan valores de F-score (F1) como medida del desempeño del modelo, ver 2.8. Mientras tanto, un 25 % de los trabajos seleccionados adoptan métodos gráficos.

2.6. Discusión

En esta revisión, proporcionamos una visión general de los métodos actuales de aprendizaje profundo para la extracción de los datos de la literatura biomédica. Los resultados de los

trabajos revisados han demostrado que, en comparación con otros enfoques de aprendizaje automático, los modelos de aprendizaje profundo son excelentes para modelar estos datos sin procesar, minimizando la necesidad de preprocesamiento e ingeniería de características, y mejorando significativamente el rendimiento en muchas tareas analíticas. Cabe destacar que los modelos de aprendizaje profundo son herramientas ideales para extraer este tipo de relaciones de interacciones medicamentosas DDI. Sin embargo, aunque las técnicas de aprendizaje profundo han mostrado resultados prometedores al realizar muchas tareas de análisis, varios desafíos abiertos permanecen.

En el presente trabajo, se ha proporcionado una visión panorámica de los modelos de aprendizaje profundo, más utilizados en la actualidad, para la extracción de datos de la literatura biomédica. Los resultados de los trabajos revisados han demostrado que, en comparación con otros enfoques de aprendizaje automático, los modelos de aprendizaje profundo son excelentes a la hora de modelar datos sin procesar, puesto que se minimiza el pre-procesamiento y se obvia la ingeniería de características, con lo cual mejora significativamente el rendimiento en muchas tareas analíticas. Se destaca que los modelos de aprendizaje profundo son herramientas apropiadas para extraer DDI de la literatura biomédica. Sin embargo, aunque las técnicas de aprendizaje profundo han mostrado resultados prometedores en muchas tareas de análisis, aún permanecen varios desafíos abiertos.

En la mayoría de los estudios seleccionados se utilizaron redes neuronales artificiales, aprovechando el hecho de que estos algoritmos revelan un alto rendimiento en la tarea de extracción de interacciones medicamentosas. En la búsqueda de mejores rendimientos, en los últimos años, los investigadores han realizado grandes esfuerzos para crear modelos novedosos y más complejos. Es así como algunos investigadores aplican modelos híbridos, enfoques de LSTM + Att. + SDP y enfoques jerárquicos, para combinar las potencialidades de varios algoritmos, a fin de aumentar el rendimiento global de los modelos. Los algoritmos de aprendizaje deben proporcionar un aspecto explicativo, es decir, la capacidad de interpretación de un modelo debe permitir al usuario obtener valioso conocimiento del conjunto de datos sin procesar. Este aspecto es primordial en la extracción de interacciones medicamentosas. Las redes neuronales artificiales presentan una marcada desventaja con respecto a los algoritmos tradicionales de aprendizaje automático, es el fenómeno de la caja negra; estos modelos son capaces de una predicción con alto rendimiento, pero el modelo obtenido es demasiado complejo para ser comprendido e interpretado por expertos humanos (Â. Miranda y cols., 2016), (Goldstein, Kapelner, Bleich, y Pitkin, 2015).Para superar el problema de la interpretabilidad en los modelos de aprendizaje automático, se proponen dos estrategias principales en la literatura: extracción de reglas y técnicas de visualización (Cortez y Embrechts, 2013),

(Prati y cols., 2011). Hay varias razones por las que los modelos deben ser interpretables; por ejemplo, el objetivo final de modelar en medicina no es una cuestión puramente de alto rendimiento predictivo, además, el modelo debe ser comprensible de tal manera que sirvan como herramienta para que los médicos definan estrategias de atención al paciente.
La evaluación del desempeño de los algoritmos de Aprendizaje de Maquina es esencial, porque sus resultados se usan como indicadores para determinar la calidad de tales algoritmos (Wang y cols., 2006), (Ozcift y Gulten, 2011). La sensibilidad es el indicador más usado para algoritmos de clasificación, dentro de los documentos seleccionados. La limitación más importante de esta medida es que no considera los costos de clasificación errónea, y puede ser engañosa, especialmente cuando las clases tienen varias probabilidades previas (Coad, Cathers, Ball, y Kadluczka, 2014). En los últimos años, se han presentado diferentes métodos gráficos para la evaluación del desempeño. Las métricas más conocidas son la curva ROC, las líneas de costo, la curva de retorno de la inversión (ROI) y la curva de elevación (Arji y cols., 2019). Cada una de estas métricas tiene sus pros y contras, por lo que no es pertinente asumir la superioridad de una sobre las otras.
No obstante, a pesar de que la sensibilidad y ROC son los indicadores más usados en los artículos seleccionados, ellos proporcionan una imagen incompleta del rendimiento de los modelos y se podría perder información valiosa. Así, ROC ha sido criticada en (Wang y cols., 2006) (Ramaswami, 2014) , pues puede usar diferentes distribuciones de costos de clasificación errónea para varios clasificadores.

2.7. Conclusión

En este trabajo, realizamos una revisión sistemática de los métodos de extracción de interacciones medicamentosas, se analizó el estado de arte actual, sus retos y desafíos. El proceso de extracción de relaciones biomédicas es una tarea crucial en el procesamiento de lenguaje natural y es el primer paso en la minería y explotación de información valiosa oculta en los textos biomédicos. RNN y CNN son actualmente los principales modelos basados en redes neuronales, para la extracción de relaciones biomédicas, cada uno tiene sus propias ventajas. En esta revisión se propone realizar un modelo híbrido basado en RNN y CNN, además del uso de SDP que contiene valiosa información sintáctica y semántica para la tarea de extracción de DDI. La mayoría de los métodos basados en redes neuronales solo usan la secuencia de oraciones como entrada de los modelos, lo que limita el rendimiento de la tarea de extracción DDI. Adicional se va a introducir mecanismos de atención de inserción para Identificar y realzar las palabras clave que existen en la semántica cercana a dos entidades.

Capítulo 3

Método

Una red neuronal convolucional CNN posee una arquitectura de aprendizaje profundo, robusta y que presenta excelente rendimiento en muchas tareas de PNL; como en la clasificación de oraciones, agrupamiento semántico y análisis de sentimientos (Dos Santos y Gatti, 2014).Una de sus principales ventajas es que no requiere la definición de características hechas a mano; en su lugar, puede aprender automáticamente, las características más adecuadas para la tarea. Una CNN combina las incrustaciones de palabras de una instancia (es decir, una frase que contiene una relación candidata entre dos entidades) usando filtros para construir un vector que represente esta instancia. Finalmente, la capa con la función Softmax, que normalmente es la última capa de la red CNN, asigna una etiqueta de clase a cada vector.

3.1. Arquitectura del modelo

En el presente trabajo, se abordó un enfoque basado en una red convolucional multicanal MCCNN. El concepto de canal en MCCNN está inspirado en el procesamiento de imágenes RGB de tres canales (R. Zhang y cols., 2018),lo que significa que la inclusión de palabras diferentes representa canales diferentes y aspectos diferentes de las palabras de entrada. Este enfoque multicanal realiza una representación más precisa de la información en donde se utiliza múltiples versiones de incrustación de palabras sugerido en (Quan y cols., 2016) esto enriquece el significado de términos desconocidos utilizados en la literatura biomédica. Adicionalmente se aborda la tarea de extracción de relaciones DDI mediante la aplicación de un modelo de extracción de características en particular. Se propone la aplicación del modelo PCNN (CNN por partes) sugerido por (Zeng, Liu, Chen, y Zhao, 2015) para superar las limitaciones durante el proceso de Max Pooling, donde el tamaño de la capa oculta aumenta

rápidamente, por lo cual es difícil capturar información estructural, es decir información valiosa de la oración, entre dos entidades farmacéuticas. La Figura 3.1, muestra el fundamento de la arquitectura de red neuronal propuesta, que permitirá la efectiva extracción de relaciones medicamentosas, a partir de un corpus formado por múltiples documentos de la literatura biomédica. Esta arquitectura está basada en el modelo (Park y cols., 2019). Se ilustra el procedimiento que maneja una instancia del corpus. Este procedimiento incluye seis fases:

- integración multicanal,
- incrustación de palabras,
- convolución,
- agrupación máxima por partes 'Piecewise Max Pooling',
- ruido gausiano,
- salida Softmax.

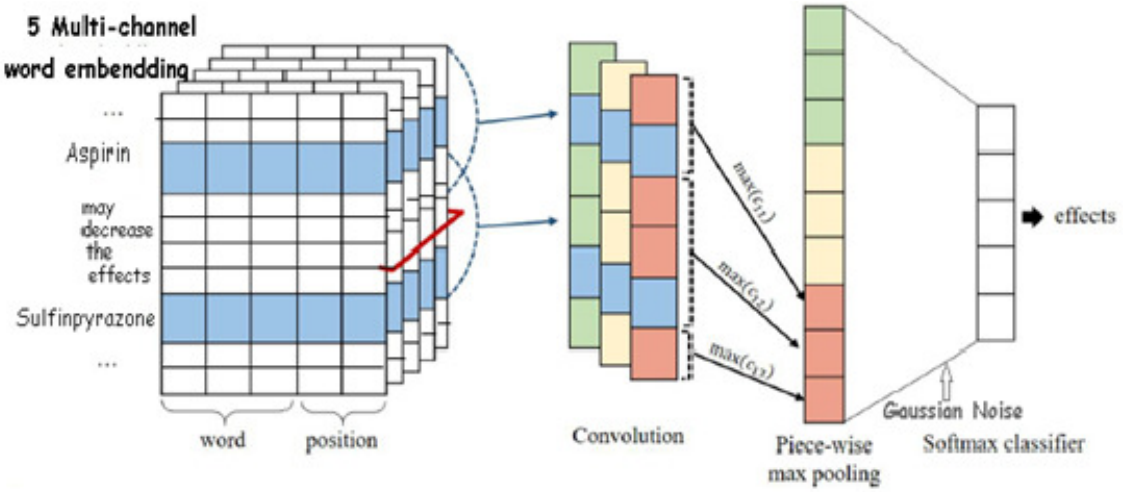

Figura 3.1: Arquitectura modelo PCN multicanal

En las próximas subsecciones, estas seis fases del procedimiento se describen de forma detallada:

3.1.1. Integración multicanal

El modelo propuesto integra cinco versiones de incrustaciones de palabras para representar mejor las palabras de entrada.
Para entender un poco más el concepto de palabra incrustada, es necesario empezar visualizándola como un vector de valor real que representa una sola palabra en función del contexto en que aparece. Esta representación de palabras nos permite mapear a cada palabra de un vocabulario con un punto en un espacio vectorial, como se muestra en la fig.3.2. Una vez mapeadas las palabras en el espacio vectorial, es posible la aplicación de la llamada *hipótesis de distribución*, que establece que las palabras que aparecen en los mismos contextos tienen significados similares o relacionados. Por lo tanto, se deduce que, en el espacio vectorial, las incrustaciones de palabras que tienen la misma relación semántica o sintácticamente están más cercanas entre sí, mientras que, las palabras que no posean una relación directa, aparecerán como incrustaciones alejadas entre sí. En el estudio actual, esta relación dependerá completamente de los datos del corpus.

Figura 3.2: Incrustaciones como puntos en el espacio 2D. Cada palabra en una oración se asigna a una incrustación

La entrada para la red neuronal consistirá en oraciones conteniendo pares de fármacos. La base del enfoque de incrustación de palabras es que las palabras que ocurren en contextos similares tienen una alta probabilidad de tener significados similares. La incrustación de palabras refleja la distribución de palabras en un corpus sin etiquetar, con el fin de garantizar la máxima cobertura de las incrustaciones de palabras, los artículos de PubMed, PMC, MedLine y Wikipedia se utilizaron en nuestro trabajo para entrenar la capa de incrustación de palabras. Se trabajó con cinco versiones de incrustación de palabras basadas en estos corpus: (Moen y Ananiadou, 2013) publicaron las primeras cuatro incrustaciones de palabras,

mientras que la inclusión de la quinta incrustación es entrenada por CBOW, está técnica esta descrita en el capítulo 2 en la sección Redes Neuronales para ER", en el corpus de MedLine (http://www.nlm.nih.gov/databases/journal.html), ver Tabla 3.1.
Con frecuencia, los problemas verbales no registrados a menudo surgen debido al uso de términos biológicos desconocidos, para compensar este problema, en nuestro proyecto se utilizó los corpus biológicos como PubMed, PMC, y los corpus comunes como MedLine y Wikipedia. La incrustación de palabras se usa para cada canal de entrada para aliviar problemas verbales no registrados. Además, el significado de cada palabra se enriqueció enormemente, es decir se mejoró su representación en el espacio.

3.1.2. Incrustación de palabras

Las incrustaciones de palabras son representaciones distribuidas de palabras, que consisten en mapeos que asignan a cada palabra de un texto, un vector de valor real de "k"dimensiones. Esta técnica, según estudios recientes, han mostrado un excelente desempeño en la captura de información semántica y sintáctica en oraciones (Mikolov y cols., 2013). La utilización de incrustación de palabras o 'word embenddings' que han sido previamente entrenadas, esto significa que son las incrustaciones aprendidas en una tarea que se utilizan para resolver otra tarea similar, en este caso implica predecir una palabra basada en una o más de estas palabras circundantes, se ha convertido en un valioso aporte para mejorar muchas tareas de PLN (Nam, Han, Kim, y Choi, 2018).
En la tarea de extraer una relación de una oración dada, se requiere traducir cada palabra a un vector de baja dimensión. En el presente trabajo, se tradujó cada palabra a un vector y se busco la respectiva representación de la incrustación de palabra pre-entrenada. Además, se utilizó las características de posición para especificar bien a los pares de entidades, que también se transformaron en vectores al buscar las incrustaciones de posición.

Tabla 3.1: Corpus entrenados

Item	No Palabras	Corpus entrenado
1	2515686	PMC
2	2351706	Pubmed
3	4087446	PMC and Pubmed
4	5443656	Wikipedia and Pubmed
5	650187	Medline

El método de incrustación de palabras generalmente aprende sin supervisión, explotando

la estructura de coincidencia de palabras en texto sin etiquetar. Los investigadores han propuesto varios métodos de incrustaciones de palabras de entrenamiento (Mikolov y cols., 2013)(Collobert y cols., 2011). Estos métodos de incrustación de palabras aprenden una representación vectorial de valor real para un vocabulario predefinido de tamaño fijo a partir de un corpus de texto. El proceso de aprendizaje está unido con el modelo de red neuronal en alguna tarea, como la clasificación de relaciones, o es un proceso no supervisado que utiliza estadísticas de documentos. En el capítulo 2 sección Redes Neuronales para ER, item "No supervisado *embeddings*",se repasan algunas técnicas que se utilizan para aprender la incrustación de una palabra en base a un corpus de texto.

3.1.3. Posición de incrustación

La posición de incrustación se añade como valor de entrada del modelo, esta posición de incrustación determina la distancia relativa entre dos entidades y las palabras restantes en una oración. Mediante este proceso se puede indicar la ubicación de dos objetos. Por ejemplo, en la siguiente oración, ver Figura 3.3:

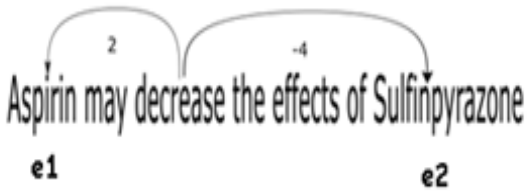

Figura 3.3: Posición de la palabra dentro de la secuencia de entrada

3.1.4. Convolución

El modelo convolucional se ha aplicado principalmente en el campo del reconocimiento de imágenes, en los últimos años se han realizado aplicaciones de estos modelos dentro del campo del procesamiento del lenguaje natural PLN, donde mostró excelente rendimiento y su aplicabilidad se está extendiendo de forma considerable en esta área. La capa de convolución es en realidad el proceso de extracción de características. La capa convolucional es igual a aplicar filtros en n-gramas de la oración de entrada, usando diferentes tamaños de ventana para obtener la información de la entidad.
Sea

$$x_i \in R^k$$

el vector de palabra de k dimensiones correspondiente a la iésima palabra en la oración. Una oración de longitud n es representada como:

$$x_{1:n} = x_1 \oplus x_2 \oplus ..... \oplus x_n$$

donde $\oplus$ es el operador de concatenación. En general, que

$$x_{i:i+j}$$

se refiera a la concatenación de palabras

$$x_i, x_{i+1}, ..., x_{i+j}$$

Una operación de convolución involucra un filtro

$$w \in R^{hk}$$

que se aplica a una ventana de h palabras para producir una nueva característica. Por ejemplo, una característica c_i se genera a partir de una ventana de palabras

$$x_{i:i+h-1}$$

mediante

$$c_i = f(wx_{i:i+h-1} + b)$$

Aquí

$$b \in R$$

es un término de sesgo y f es una función no lineal como la tangente hiperbólica. Este filtro se aplica a cada posible ventana de palabras en la oración

$$x_{1:h}, x_{2:h+1}, ..., x_{n-h+1:n}$$

para producir un mapa de características

$$c = [c_1, c_2, ..., c_{n-h+1}] \text{ , con } c \in R^{n-h+1}$$

Luego aplicamos una operación de agrupación de tiempo extra máximo (Collobert y cols.,

2011) sobre el mapa de características y tomamos el valor máximo

$$\hat{c} = max[c]$$

como la característica correspondiente a este filtro particular, la idea es capturar la característica más importante, una con el valor más alto, para cada mapa de características. Este esquema de agrupación naturalmente trata con longitudes de oración variables.
Hemos descrito el proceso mediante el cual se extrae una característica de un filtro. El modelo usa múltiples filtros (con diferentes tamaños de ventana) para obtener múltiples funciones. Estas características forman la penúltima capa y se pasan a una capa softmax totalmente conectada cuya salida es la distribución de probabilidad sobre las etiquetas.

3.1.5. Agrupación máxima por partes (Piecewise Max Pooling)

El modelo PCNN fue propuesto por (Zeng y cols., 2014), quien planteó la convolución por partes, usando una red neuronal convolucional. Este es un modelo que tiene mucho éxito en la tarea de extracción de relaciones supervisadas a distancia. El éxito de este proceso de extracción de relaciones depende de la extracción de las características estructurales correctas de la oración que contiene el par de entidades, ver Figura 3.1. Las redes neuronales, como las redes neuronales convolucionales 'CNN', nos ayudan a aliviar el proceso de diseñar características manualmente. La salida de una red convolucional CNNs depende de la cantidad de tokens en la oración, la operación de maxpooling a menudo se aplica para eliminar esta dependencia.

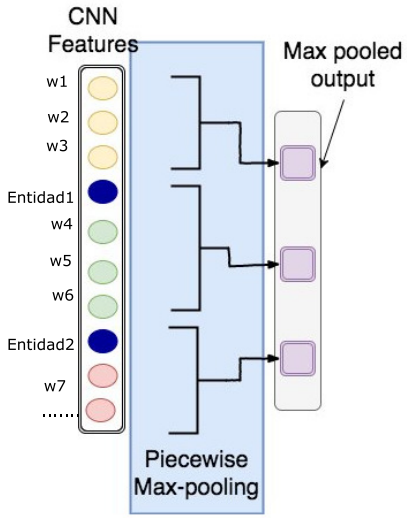

Figura 3.4: Piecewise max pooling en modelo PCNN. Modelo propuesto por [Zeng et al., 2015]

En el modelo PCNN la oración S_i que contiene dos entidades, es dividida en tres partes $c_{i1}; c_{i2}; c_{i3}$.

- c_{i1} contiene el contexto de la oración a la izquierda de la primera entidad,
- c_{i2} contiene el contexto de la oración entre las dos entidades, y
- c_{i3}:contiene el contexto de la oración a la derecha de la segunda entidad,

Luego se realiza una agrupación máxima en cada una de las tres partes, como se muestra en la Figura 3.4. De este modo, se aprovecha la información de ubicación de la entidad para retener las características estructurales de una oración después de la operación de maxpooling.

$$pc_{ij} = max(c_{ij}); 1 <= N; 1 <= j <= 3$$

El resultado de esta operación es la concatenación de

$$pc_{i1}; pc_{i2}; pc_{i3}$$

produciendo una salida de tamaño fijo.

3.1.6. Ruido Gausiano

En nuestro experimento se aplicó una capa de ruido Gaussiano, antes de la activación del clasificador softmax, para simular el efecto “data augmentation” y evitar el overfitting, de acuerdo a estudios presentes en literatura relacionada (Shorten y Khoshgoftaar, 2019) (An, 1996) (Sietsma y Dow, 1991).
Como se describe en el trabajo de (Papadaki, 2017), en este método, en lugar de generar nuevas oraciones aumentadas, insertamos perturbaciones en las incrustaciones de palabras de nuestro modelo. Para cada inserción de palabras, les agregamos un valor de ruido gaussiano según la siguiente ecuación: $w'_j = w_j + x_j$ donde, w_j es la inserción de palabras original, w'_j es la inserción de palabras resultante y x_j es el valor de ruido aleatorio. Los valores para cada elemento vectorial de ruido se muestrearon a partir de la distribución normal truncada con $\mu = 0$, $\sigma = 1$ y con un rango entre 0 y 0,3. Más tarde, seleccionamos al azar los elementos del vector con una probabilidad de 0,3 y el resto del vector se establece en cero. El vector resultante es considerado como un vector de ruido. Los valores se mantuvieron en un rango pequeño para evitar que la incrustación de palabras resultante se aleje demasiado del espacio de incrustación de palabras contextuales. Una excepción en este método a la de los otros

métodos de aumento es que, insertamos perturbaciones en todas las palabras, independientemente de su etiqueta POS. Hemos tratado de explorar las fortalezas y debilidades de estos métodos en los experimentos.

3.1.7. Salida de Softmax

Antes de realizar la clasificación, la técnica de dropout se utiliza para regularización y evitar el sobreajuste de la red. Para hacer esto, los elementos del vector z se establecen aleatoriamente en cero con una probabilidad p después de una distribución de Bernoulli para generar un vector reducido z_d. Después de eso, este vector se alimenta a una capa Softmax completamente conectada con pesos

$$W_s \in R^{mk}$$

para calcular los valores de predicción de salida para la clasificación como

$$o = z_d W_s + d$$

donde d es un término de sesgo; en el conjunto de datos, hay $k = 5$ clases (consejo, efecto, interacción, mecanismo y no DDI). En el momento de la prueba, el vector z de una nueva instancia se clasifica directamente por la capa Softmax sin un dropout.

Capítulo 4

Experimentación

La experimentación es parte substancial del presente trabajo de investigación, pues es el camino a seguir para evaluar la reproducibilidad de los experimentos ya realizados y el posible mejoramiento de los resultados de nuestra propuesta con respecto a aquellos.
En este capítulo, presentamos la configuración que se utilizó en los experimentos y los resultados obtenidos, base para la evaluación del desempeño del modelo propuesto.

4.1. Configuración experimental

Nuestro modelo PCNN multicanal es entrenado y evaluado utilizando el corpus del desafío DDIExtraction 2013 (https://www.cs.york.ac.uk/semeval-2013/task9/),cuyo corpus contiene la información que describe las DDI. Tal descripción, en estos conjuntos de documentos es extensa y desestructurada, lo que la convierte en un complejo entramado de trechos de texto, donde se encuentran escondidos las claves de la respuesta a la pregunta: ¿Dónde se encuentran las interacciones medicamentosas?
Un gran porcentaje de enfoques existentes para la extracción de DDI se basan en el aprendizaje de máquinas. Con el fin de predecir la relación entre un par de farmacos, los clasificadores generalmente se entrenan en características léxicas, sintácticas y semánticas extraídas de corpus anotados manualmente.
Brevemente revisaremos los enfoques basados en características y los basados en kernel:

- Los enfoques basados en características se centran en encontrar características potencialmente discriminatorias para representar las características de los datos. Los investigadores han explorado el uso de varios tipos de características que incluyen características de contexto (Segura-Bedmar, Martinez, y de Pablo-Sánchez, 2011), una

combinación de características léxicas, semánticas y de dominio (He, Yang, Zhao, Lin, y Li, 2013), y características heterogéneas que consisten en léxico, sintáctico , características semánticas y de negación derivadas de los árboles de análisis (Chowdhury y Lavelli, 2013a).

- Los enfoques basados en kernel emplean diferentes núcleos para calcular la similitud entre dos instancias explotando las representaciones estructurales de instancias de datos como los árboles de análisis sintácticos o los gráficos de dependencia (Tikk, Solt, Thomas, y Leser, 2013). En los desafíos anteriores de extracción de DDI, los núcleos más comúnmente utilizados son el núcleo de gráfico de todas las rutas (Airola y cols., 2008), el núcleo lingüístico poco profundo (Giuliano, Lavelli, y Romano, 2006) y el núcleo de árbol que encierra la ruta (Moschitti, 2004).

En este escenario, la utilización de métodos kernel y los métodos basados en características no aportan, de manera efectiva, a la comprensión semántica y a la extracción de características, por lo que se vuelve primordial el uso de métodos basados en redes neuronales, para evitar una muy costosa ingeniería manual de características, que si bien es verdad que poseen una alta complejidad, permitirán completar de forma efectiva, la comprensión semántica de los contenidos del corpus.

4.1.1. Interacciones medicamentosas

El corpus DDI se distribuye en documentos XML siguiendo el formato propuesto por Pyysalo et al. [31], con el fin de unificar los diferentes formatos de los principales cuerpos. El objetivo principal es garantizar un alto uso del corpus. Ver ejemplo del corpus anotado Fig. refgrafoTxtAnotado. El corpus se compone de 730 documentos que representan interacciones medicamentosas DDI de la base de datos de DrugBank y 175 resúmenes sobre DDI de la base de datos MedLine. Este vasto volumen de texto se convertirá en el dataset al que se lo dividirá en conjunto de entrenamiento y conjunto de prueba. Para ello, los documentos son sometidos a una fase procesamiento, que se describe más adelante.
Todas las instancias de DDI se clasifican en cinco categorías: mecanismo, efecto, consejo, int y falso. Las definiciones detalladas de todas las categorías se describen a continuación.

- (1) Mecanismo. Cuando los textos originales describen el mecanismo farmacodinámico o farmacocinético de interacción sobre dos fármacos, la DDI de los dos fármacos pertenece al mecanismo.

- (2) Efecto. Cuando los textos originales describen efectos farmacológicos clínicos u otros sobre dos drogas, la DDI de los dos farmacos pertenece al efecto.
- (3) Consejo. Cuando los textos originales describen consejos o sugerencias sobre dos medicamentos, la DDI de los dos farmacos pertenece al consejo.
- (4) Int. Cuando los textos originales indican una interacción simplemente declarada o descrita en una oración., la DDI de los dos farmacos pertenece a int.
- (5) Falso. Cuando un medicamento no expresa una influencia sobre el otro medicamento que se toma al mismo tiempo, el DDI de los dos medicamentos es falso.

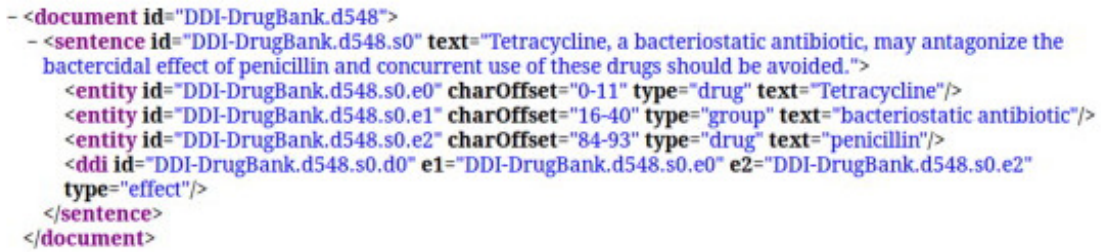

Figura 4.1: Ejemplo de un documento anotado del corpus DDIExtraction 2013. Tomado de [Segura Bedmar y cols., 2013]

La clasificación propuesta de DDI es coherente con los requisitos de información establecidos por los expertos en farmacología para un manejo adecuado de DDI en el entorno clínico (Bergk, Haefeli, Gasse, Brenner, y Martin-Facklam, 2005), (Aronson, 2004). Para este propósito, los profesionales de la salud deben recibir información sobre cómo se produce la interacción (mecanismo), qué consecuencias se pueden esperar (efecto) y cómo se puede gestionar para evitar o reducir el riesgo asociado (consejo).
Se muestran oraciones que describen la interacción medicamentosas DDI en la Fig. 4.2, la Fig. 4.3. En la Fig. 4.2, la primera oración describe dos interacciones: efecto y mecanismo, y la última también describe un DDI de tipo de efecto. En la Fig. 4.3, se describen DDI de tipo de efecto entre fenfluramina y un grupo de fármacos, fármacos antihipertensivos. La última oración da un consejo para evitar un DDI.
El detalle del corpus se muestra en la Tabla 4.1.

4.1.2. Preprocesamiento de texto

Para el preprocesamiento de los datos de entrenamiento y prueba, se llevan a cabo tres tipos de operaciones: tokenización, cegamiento de farmacos y filtrado de instancias negativas.

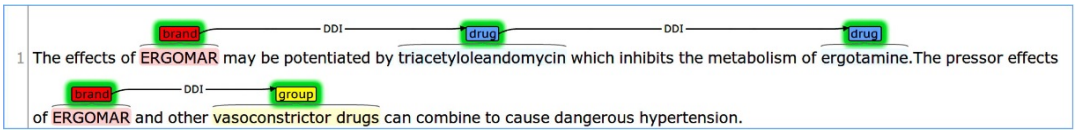

Figura 4.2: Ejemplo de interacciones medicamentosas de tipo efecto y mecanismo, recuperado de https://www.sciencedirect.com/science/article/pii/S1532046413001123.

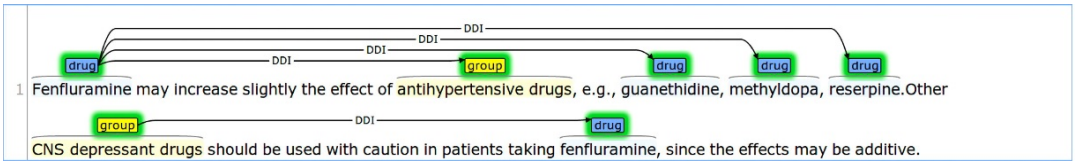

Figura 4.3: Ejemplo de interacciones medicamentosas de tipo efecto y consejo, recuperado de https://www.sciencedirect.com/science/article/pii/S1532046413001123.

- cegamiento de farmacos Se realizó una acción llamada cegamiento de farmacos que reemplaza nombres concretos de entidades farmacológicas en oraciones con caracteres especiales. El objetivo del cegamiento de drogas es mejorar la generalización del modelo , se siguió el método propuesto por (Liu, Tang, Chen, y Wang, 2016). Posteriormente se generan instancias DDI a partir del par de entidades farmacológicas detectadas en la oración que se recibe de entrada. Dada una oración con n entidades farmacológicas detectadas. El par de farmacos objetivo de las instancias DDI se anonimizan utilizando los siguientes caracteres especiales "Drugaz "Drugbz los otros farmacos son reemplazado por "Drugn". Según (Liu y cols., 2016), el número de instancia detectadas viene dada por la ecuación.

$$C_{(n,2)} = n(n-1)/2$$

  Por ejemplo, dada una oración con tres entidades farmacológicas detectadas: "La aminoglutetimida disminuye el efecto de la cumarina y la warfarina.", aplicando la ecuación resulta $C(3,2) = 3(3-1)/2 = 3$ instancias DDI, como se muestra en la siguiente tabla 4.2

- tokenización Después del cegamiento de farmacos, utilizamos el Kit de herramientas de lenguaje natural (NLTK) [1] para tokenizar las oraciones y convertirlas en secuencias de tokens. En este paso remplazamos tokens numéricos con el token general # y se remueven todos los símbolos , caracteres especiales que no son ASCII, signos de

[1] http://www.nltk.org/

Tabla 4.1: Detalle del corpus DDI extraction 2013

Clase	Train Drug-Bank	Train Medline	Total train	Test Drug-Bank	Test Medline	Total test
Documento	572	142	714	158	33	191
Consejo	818	8	826	214	33	221
Efecto	1535	152	1687	298	33	360
Mecanismo	1257	62	1319	278	33	302
Int	572	178	10	188	94	96

Tabla 4.2: Instancias generadas

(aminoglutetimida, cumarina)	La aminoglutetimida drug1 disminuye el efecto de la cumarina drug2 y la warfarina drugn
(aminoglutetimida, cumarina)	La aminoglutetimida drug1 disminuye el efecto de la cumarina drugn y la warfarina drug2
(aminoglutetimida, cumarina)	La aminoglutetimida drugn disminuye el efecto de la cumarina drug1 y la warfarina drug2

puntuación y algunas palabras sin sentido seleccionadas.

- filtrado de instancias negativas

  Evidentemente, el conjunto de datos está extremadamente desequilibrado. En varios estudios se ha demostrado que los conjuntos de datos desequilibrados causan una influencia fuerte negativamente en el rendimiento del modelo (Chowdhury y Lavelli, 2012), (A. Sun, Grishman, y Sekine, 2011). Métodos anteriores;(S. Kim, Liu, Yeganova, y Wilbur, 2015), (Liu y cols., 2016) y (Quan y cols., 2016) han verificado que el filtrado de instancias negativas es una operación práctica que puede aliviar el efecto de un conjunto de datos desequilibrado.

  En este paso, se describe el método de filtrado de instancias negativas aplicado al corpus DDI extraction 2013. Como se mencionó anteriormente, los datos desbalanceados afectan el rendimiento; por lo tanto, se implementó las siguientes dos reglas para filtrar instancias negativas que más vienen hacer “datos falsos”, y así evitar la degradación del rendimiento.

  La primera regla es eliminar cualquier par de drogas que se refiera a la misma droga.

Este tipo de par de drogas puede tener el mismo nombre o sinónimos de droga.

La segunda regla es filtrar pares de drogas que comparten relaciones de coordenadas. Una relación de coordenadas se refiere al caso en el que dos palabras están conectadas por una conjunción (por ejemplo, 'y', 'o') o una coma. En muchos casos, la relación coordinada entre tres o más fármacos es la característica de la instancia negativa.

El texto pre-procesado se encuentra organizado en las siguientes columnas: Codigo de la Fuente: Medline o DrugBank; Nombre del fármaco 1; Tipo de fármaco 1; Nombre del fármaco 2; Tipo de fármaco 2; Tipo de relación; Oración. Ver figura 4.4

```
DDI-MedLine.d79.s0.p5   didanosine  drug    _   IDV drug      _    false   if taken # hour before drugn -lcb- drugn -rcb- , druga does not affec
DDI-MedLine.d79.s1.p0   indinavir   drug    _   didanosine  drug     _   mechanism   concurrent administration of druga and drugb significantl
DDI-MedLine.d79.s1.p2   indinavir   drug    _   didanosine  drug     _   false   concurrent administration of druga and drugn significantly re
DDI-MedLine.d79.s1.p4   didanosine  drug    _   indinavir   drug     _   false   concurrent administration of drugn and druga significantly re
DDI-MedLine.d79.s1.p6   didanosine  drug    _   indinavir   drug     _   false   concurrent administration of drugn and druga significantly re
DDI-MedLine.d79.s1.p7   indinavir   drug    _   didanosine  drug     _   false   concurrent administration of drugn and drugn significantly re
DDI-MedLine.d79.s1.p9   didanosine  drug    _   indinavir   drug     _   false   concurrent administration of drugn and drugn significantly re
DDI-MedLine.d79.s2.p2   indinavir   drug    _   didanosine  drug     _   false   we compared druga pharmacokinetics and gastric ph in # human
DDI-MedLine.d79.s2.p4   indinavir   drug    _   didanosine  drug     _   false   we compared drugn pharmacokinetics and gastric ph in # human
DDI-MedLine.d79.s2.p5   indinavir   drug    _   didanosine  drug     _   false   we compared drugn pharmacokinetics and gastric ph in # human
DDI-MedLine.d79.s3.p0   indinavir   drug    _   didanosine  drug     _   mechanism   median gastric ph was significantly higher when druga was
DDI-MedLine.d79.s5.p0   Indinavir   drug    _   didanosine  drug     _   advise  druga may be taken with a light meal # h following the admini
DDI-MedLine.d59.s4.p0   Cytochalasin D  drug_n  _   carbachol   drug     _   effect  druga at # microm preferentially blocked the secretory ef
DDI-MedLine.d59.s5.p0   Cytochalasin D  drug_n  _   carbachol   drug     _   effect  druga inhibited the drugb - stimulated intracellular ca -
DDI-MedLine.d75.s3.p0   mu-selective opioids    group   _   delta(1)-selective opioids  group   _   false   the percentage of neurons hyperpo
DDI-MedLine.d133.s0.p0  verapamil   drug    _   bombesin    drug_n  _   effect  suppression by druga of drugb - enhanced peritoneal metastasi
DDI-MedLine.d133.s0.p1  verapamil   drug    _   azoxymethane    drug_n  _   false   suppression by druga of drugn - enhanced peritoneal metas
DDI-MedLine.d133.s0.p2  bombesin    drug_n      azoxymethane    drug_n      false   suppression by drugn of druga - enhanced peritoneal metas
```

Figura 4.4: Fragmento del corpus pre-procesado.

4.1.3. Ajuste y configuración de parámetros

Para esta etapa de experimentación se usó el sistema HPC-MODEMAT, que es un clúster o conjunto de servidores de alto rendimiento computacional, ubicado en el laboratorio Ada Lovelace, de la Facultad de Sistemas de la Escuela Politécnica Nacional, con la siguiente configuración: NODELIST: quinde-4-2 NODES:1 PARTITION: quinde-G8-2695 CPUS:24 MEMORY: 257749. El modelo MCPCNN propuesto está codificado en Python v3.7 64 bits, se importaron las librerías Keras v2.2.4 (https://keras.io/), gensim v3.8.1 (https://pypi.org/project/gensim/) y TensorFlow v1.14.0 (https://www.tensorflow.org/) como backend; se consideraron cinco tipos de incrustación de palabras: PMC, PubMed, PMC y PubMed, Wikipedia y PubMed, MedLine basado en (Park y cols., 2019); y, la herramienta word2vec [http://code.google.com/p/word2vec/] fue necesaria para entrenar las representaciones de las palabras de entrada.
Se realizaron pruebas con diferentes valores de parámetros y se observó cómo rendimiento del modelo se ve afectado. A partir de las observaciones, se demuestra que, con los parámetros adecuados, se puede lograr un crecimiento de aprendizaje significativo. A continuación, se revisan esos parámetros:

La tabla 4.10 muestra los parámetros utilizados en los experimentos. Se realizo pruebas con diferentes valores de parámetros y se observó cómo afecta el rendimiento de nuestro modelo. A partir de las observaciones, se demuestra que con los parámetros adecuados puede lograr un crecimiento de aprendizaje significativo. A continuación revisaremos estos parámetros:

4.1.3.1. Epoca (Epoch)

El número de épocas es un hiperparámetro que define el número de veces que el algoritmo de aprendizaje funcionará en todo el conjunto de datos de entrenamiento. Una época significa que cada muestra en el conjunto de datos de entrenamiento ha tenido la oportunidad de actualizar los parámetros internos del modelo. Una época se compone de uno o más lotes. Puede pensarse en un lazo cíclico por el número de épocas, donde cada ciclo actúa sobre el conjunto de datos de entrenamiento. Dentro de este lazo, hay otro lazo anidado que itera sobre cada lote de muestras. Cada lote tiene el número especificado de muestras, denominado tamaño de lote. El número de épocas por lo general es grande; puede ser de cientos o miles, lo que permite que el algoritmo de aprendizaje se ejecute hasta conseguir que el error del modelo sea lo suficientemente cercano al mínimo posible. En la literatura científica relacionada con el tema, se encuentran algunos ejemplos del número de épocas utilizado en las experimentaciones; entre los utilizados se encuentran: 10, 100, 500, 1000, entre otros. Una buena idea consiste en crear un gráfico de líneas que muestre épocas a lo largo del eje x y el error o eficacia del modelo en el eje y. Estas tramas a veces se llaman curvas de aprendizaje. Esta técnica se convierte en una herramienta para diagnosticar si el modelo ha superado el aprendizaje, si el aprendizaje insuficiente o si se ajusta adecuadamente al conjunto de datos de entrenamiento. Nos indica también, el número de veces en las que el conjunto de datos de entrenamiento ha pasado por la red neuronal en el proceso de entrenamiento

En nuestro modelo seleccionamos el número de épocas de valor 30, puesto que como se observa en la tabla 4.3, para los valores más altos de 30 el aprendizaje del modelo no es significativo, ver Fig. 4.5. En algunos estudios como (Y. Kim, 2014) (Park y cols., 2019) usan el valor de epocas en el rango de 25 a 30.

4.1.3.2. Tamaño de lote (Batch Size)

El tamaño del lote es un hiperparámetro que define el número de muestras para trabajar, antes de actualizar los parámetros internos del modelo. Intuitivamente, se puede imaginar un lote como un bucle iterando sobre una o más muestras y haciendo predicciones; al alcanzar el

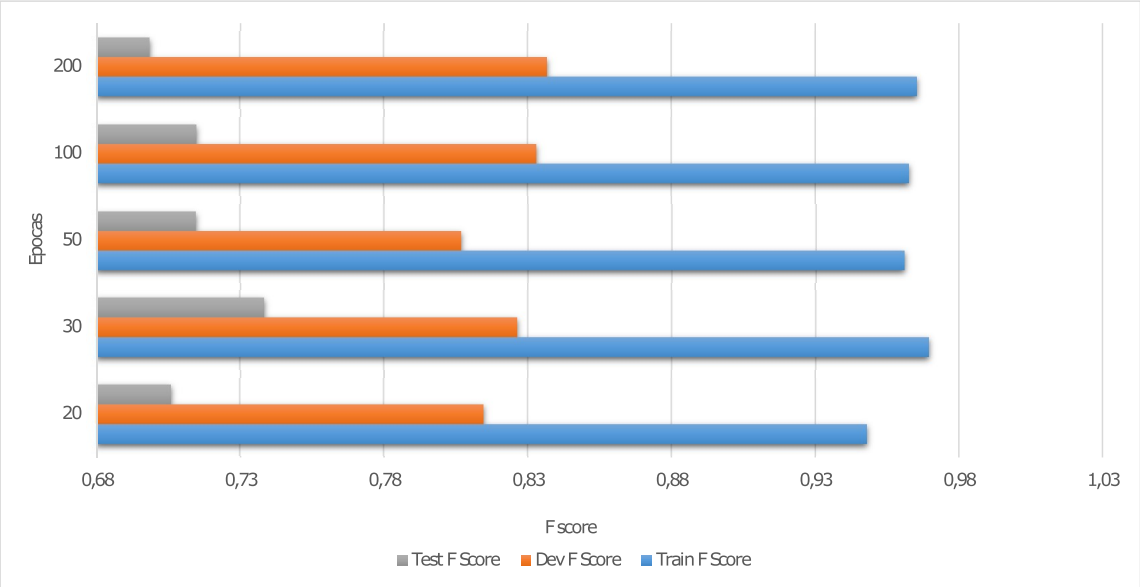

Figura 4.5: Número de épocas

	Entrenamiento			Validación			Prueba		
No. epocas	Prec	Recall	F1	Prec	Recall	F1	Prec	Recall	F1
20	0.9162	0.9819	0.9479	0.7511	0.8895	0.8145	0.6745	0.7400	0.7059
30	0.9527	0.9864	0.9693	0.7922	0.8632	0.8262	0.7359	0.7405	0.7382
50	0.9374	0.9849	0.9606	0.7668	0.8507	0.8066	0.7188	0.7106	0.7147
100	0.9392	0.9873	0.9626	0.8084	0.8590	0.8329	0.7120	0.7178	0.7149
200	0.9540	0.9769	0.9653	0.8086	0.8667	0.8366	0.7075	0.6900	0.6986

Tabla 4.3: Comparación por número de épocas

final del lote, las predicciones se comparan con las variables de salida esperadas y se calcula un error. A partir del valor de este error, se aplica el algoritmo de actualización para mejorar el modelo. Cuando se usan las librerías de Keras, este parámetro es el que indica el tamaño de los lotes (mini lotes particionados para pasarlos a la red) que se usarán en el método fit, durante el entrenamiento, para actualizar el gradiente. Este valor dependerá de factores como la capacidad de memoria principal del computador. Un buen valor predeterminado para el tamaño de lote puede ser 32. Puede también usarse 64, 128, 256, etc. En el modelo propuesto se utilizó un valor de 64.

4.1.3.3. Taza de aprendizaje (Learning rate lr)

Cambiar los pesos demasiado rápido al sumar o restar con mucha frecuencia (es decir, llegar a medidas demasiado grandes) puede dificultar la capacidad de minimización de la

función de pérdida. Para asegurarnos de que esto no suceda, utilizamos una variable llamada tasa de aprendizaje. Esta representa una cantidad pequeña, típicamente de algunas diez milésimas, por la que se multiplican los gradientes para ir escalando. Con lo mostrado, se garantiza que cualquier cambio que se realice en los pesos, será suficientemente pequeño. Hablando en términos matemáticos, tomar medidas demasiado grandes puede significar que el algoritmo pase por alto los valores que le permitan converger a un nivel óptimo. Al mismo tiempo, no es adoptar medidas que sean demasiado pequeñas, porque es posible que esto imposibilite al algoritmo, alcanzar los valores más adecuados a nuestros pesos. En matemáticas, los pasos que son demasiado pequeños pueden llevar al optimizador a converger en un mínimo local para la función de pérdida, pero no necesariamente en el mínimo absoluto. Resumiendo, hay que recordar que la tasa de aprendizaje asegura que los pesos deben cambiar al ritmo correcto, El vector de la gradiente como toda magnitud vectorial tiene direccion y magnitud. Los algoritmos de gradiente descendente multiplican la magnitud del gradiente por un escalar llamado learning rate, para determinar el siguiente punto. Por ejemplo, si la magnitud del gradiente es 1,5 y el learning rate es 0,01 entonces el algoritmo seleccionará el próximo punto a 0,015 del punto anterior. Una buena regla en general es que, si el modelo de aprendizaje no funciona, se deberá menguar el parámetro learning rate.

4.1.3.4. Optimizador (Optimizer)

Durante el entrenamiento, los parámetros del modelo se cambian en el afán de minimizar la función de pérdida y hacer nuestras predicciones lo más correctas posibles. Los optimizadores ajustan os pesos de forma progresiva, con el objetivo de lograr un modelo lo más preciso posible. La función de pérdida es la encargada de indicarle al optimizador cuándo sí y cuándo no, se mueve en la dirección correcta. Para un modelo mental útil, puede pensar en un excursionista tratando de bajar una montaña con los ojos vendados. Para empezar, es imposible saber en qué dirección va, pero al dar un paso, hay algo que sí puede saber: si está descendiendo o ascendiendo. Esta única guía, le permitirá alcanzar la base de la montaña, esto será, dando solamente pasos que lo conduzcan hacia abajo. Al hablar sobre optimizadores, se debe comenzar con el algoritmo más usado, se trata del descenso de gradiente (Gradient Descent). Este algoritmo se utiliza en los modelos de Machine Learning (y otros problemas matemáticos) cuando se trata de optimización. Es rápido, robusto y flexible. A continuación, se describe brevemente su funcionamiento:

- Calcule el efecto que un pequeño cambio en cada peso individual, provocaría en la

función de pérdida (en qué dirección está caminando el excursionista)

- Ajuste cada peso individual en función de su gradiente (es decir, dé un pequeño paso en la dirección determinada)
- Siga ejecutando los pasos anteriores hasta que la función de pérdida sea lo más baja posible

La parte difícil de este algoritmo (y los optimizadores en general) es predecir la medida en que cambiará un gradiente, a partir de un pequeño cambio en un peso o parámetro. Los gradientes son derivadas parciales, y son una medida de cambio; conectan la función de pérdida a los pesos; nos dicen qué operación específica debería realizarse con los pesos: por ejemplo: sumar 5, restar .07 o cualquier otra operación que permita reducir el rendimiento de la función de pérdida y, por lo tanto, hacer que el modelo sea más preciso.
Un inconveniente que se puede experimentar durante la optimización, es quedarse atascado en los mínimos locales. Cuando se trata de conjuntos de datos con alta dimensionalidad, es posible que encuentre un área donde parece que ha alcanzado el valor más bajo posible para su función de pérdida, pero en realidad es solo un mínimo local. En la línea de la analogía del excursionista, esto es como encontrar un pequeño valle dentro de la montaña que estás bajando. Parece que has llegado al fondo, salir del valle requiere, por intuición, escalar, pero no lo has hecho. Para evitar quedarse atascado en los mínimos locales, nos aseguramos de utilizar la tasa de aprendizaje adecuada. Es difícil exagerar cuán popular es realmente el descenso de gradiente, y se usa en todos los ámbitos, incluso en arquitecturas de redes neuronales complejas (la propagación hacia atrás es básicamente un descenso de gradiente implementado en una red). Sin embargo, existen otros tipos de optimizadores basados en el descenso de gradiente, y aquí hay algunos de ellos:

- Adagrad adapta la tasa de aprendizaje específicamente a las características individuales; esto significa que algunos de los pesos en un conjunto de datos tendrán tasas de aprendizaje diferentes a otras. Esto funciona muy bien para conjuntos de datos dispersos donde los ejemplos de entrada son dispersos. Sin embargo, Adagrad tiene un problema importante: la tasa de aprendizaje adaptativo tiende a ser muy pequeña con el tiempo. Los optimizadores a continuación buscan eliminar este problema.
- RMSprop es una versión especial de Adagrad desarrollada por el profesor Geoffrey Hinton en su clase de redes neuronales. En lugar de dejar que todos los gradientes se acumulen por impulso, solo acumula gradientes en una ventana fija. RMSprop es

similar a Adaprop, que es otro optimizador que busca resolver algunos de los problemas que Adagrad deja abiertos.

- Adam representa la estimación del momento adaptativo, y es otra forma de usar gradientes pasados para calcular los gradientes actuales. Adam también utiliza el concepto de impulso al agregar fracciones de gradientes anteriores al actual. Este optimizador se ha generalizado bastante y está prácticamente aceptado para su uso en el entrenamiento de redes neuronales. Es fácil perderse en la complejidad de algunos de estos nuevos optimizadores. Solo recuerde que todos tienen el mismo objetivo: minimizar la función de pérdida.
- Adadelta Es otra extensión del optimizador Adagrad que busca reducir su tasa de aprendizaje agresiva y monotónicamente decreciente. En lugar de reunir todos los gradientes pasados, restringe la ventana del gradiente pasado acumulado a un tamaño fijo.

En todas las configuraciones de nuestros experimentos el optimizador Adam tuvo mejor rendimiento en los resultados, ver tabla 4.4 y Fig.4.6.

Tabla 4.4: Optimizadores

Optimizador	F1 score Train	F1 score Dev	F1 score Test
Adam	0,9693	0,8262	0,7382
RMSprop	0,9408	0,8375	0,7034
Adagrad	0,7358	0,691	0,6113
Adadelta	0,2551	0,2469	0,2102

4.1.3.4.1. Word embedding size, position embedding size Este parámetro maneja la dimensionalidad de las incrustaciones y la posición de la dimensión de las incrustaciones, esta se establece en 10 y se inicializa aleatoriamente.

4.1.3.4.2. Dropout rate Es una técnica para evitar el overfitting que puede ocurrir en las redes de aprendizaje profundo. Esta técnica consiste en desechar ejemplos, de forma aleatoria, y sus conexiones durante el entrenamiento. El rango de valores $0 \leq \rho \leq 1$ se refiere al porcentaje de ejemplos desechados. Se busca valores distintos de los extremos, si $\rho = 0$ no se desechará ninguno; y si $\rho = 1$ está tendrá que desechar todos los ejemplos, es decir, la capa se quedaría unidades. Este parámetro se lo eligió a partir del análisis de los resultados.

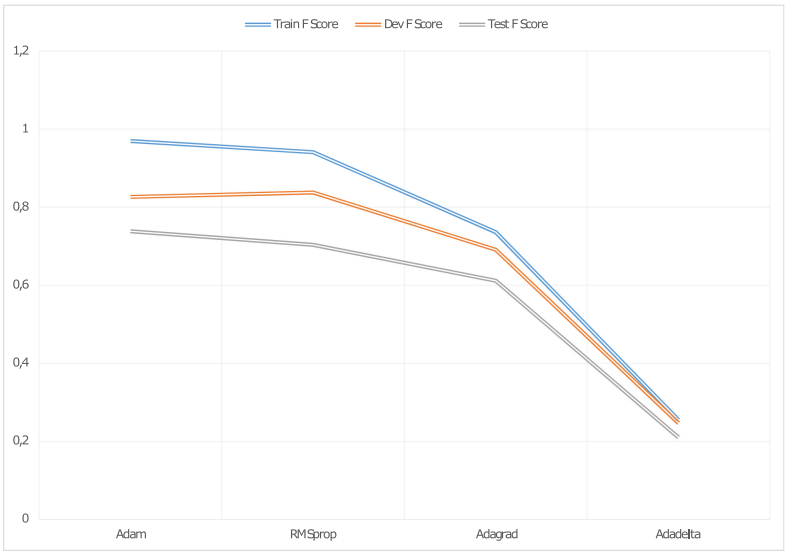

Figura 4.6: Optimizadores

4.1.3.4.3. Gauss noise El ruido gaussiano es ruido estadístico cuya función densidad de probabilidad es igual a aquella de la distribución normal, también se la conoce como distribución gaussiana. La distribución normal es la distribución más importante y más utilizada en estadística. A veces llamada curva de campana. El ruido producido es aleatorio, pero como la mayoría de los eventos aleatorios, sigue un cierto patrón. La razón por la cual la distribución gaussiana es altamente preferida en la ciencia de datos y en el aprendizaje automático, es que los eventos aleatorios de Gauss son muy comunes en la naturaleza. Cuando observamos un evento aleatorio que es la suma de muchos eventos independientes, todas las variables aleatorias parecen ser variables de Gauss (Patterson y Gibson, 2017). En el área del aprendizaje automático, es común la necesidad de agrandar el conjunto de datos; en el presente trabajo, para este fin. se utiliza la técnica del ruido gausiano, con una media de 0.0 y una desviación estándar de 0.15 a la inserción de palabras multicanal de entrada, para superar y evitar el sobreajuste.

4.1.3.4.4. Regularización La regularización es una técnica matemática aplicada a los modelos de aprendizaje automático, para evitar encajonamiento, en la fase de entrenamiento. La más común de estas técnicas es usar un término de regularización en la función objetivo para hacer que el proceso de aprendizaje sea más flexible y evitar la captura de datos irrelevantes. Concretamente, agregamos un parámetro de ajuste que multiplica y limita los valores de los pesos para penalizar la complejidad del modelo y suavizar las funciones. Existen dos términos de regularización diferentes, que pueden ser aplicados:

- Norma L1, que utiliza la media absoluta de los pesos como $J(\theta) + \lambda \sum |w|$, también conocido como estimador LASSO.
- Norma L2, que utiliza la media al cuadrado de los pesos como $J(\theta) + \lambda \sum w^2$, también conocida como regresión de Ridge. Las principales diferencias entre ambos términos radican en que L1 produce una salida dispersa que genera algunos pesos cero si no son relevantes; en contraste, la regularización Norma L2 tiene soluciones analíticas y es computacionalmente más eficiente.

  En nuestros experimentos se usó la regularización Norma L2.

4.1.4. Evaluación sobre corpus DDIExtraction2013

Para evaluar el desempeño del modelo propuesto, se usó la métrica F1 score que es la media armónica de precisión y recuperación, esta métrica se usa principalmente cuando los datos no están equilibrados, el conjunto de datos del corpus DDI extraction 2013 carece de uniformidad, por lo que el uso del F1 score, para la evaluación del rendimiento del modelo, es adecuado.
En los experimentos que se presentarán más adelante, se utilizó la siguiente configuración que se muestra en la tabla 4.5:

Tabla 4.5: Parametros de los modelos

Parametro	Valor
Epoch	50
Batch size	64
Optimizer	Adam
Word embedding size	200
Position embedding size	10
CNN kernel size	[3 5 7 9]
Number of filters	128
Dropout rate	0.3
Learning rate	3.00E-04

4.1.4.1. Experimento 1. Modelo CNN

En este experimento se construyó el modelo de red convolucional para la tarea DDIExtraction, con cuatro capas: una capa de incrustación, una capa convolucional, una capa de agrupación máxima y una capa Softmax. Las oraciones del corpus son tokenizadas, luego su caja se

convierte a minúsculas y preprocesadas usando cegamiento de entidades. Adicionalmente, se realiza un filtrado de instancias negativas, donde se descarta los pares de fármacos que no interactúan; posteriormente se aplica el proceso de max-pooling y finalmente el clasificador softmax. Este modelo presentó los siguientes resultados, se observan en la tabla 4.6 y la Fig. 4.7

Tabla 4.6: Evaluación de modelo CNN

Modelo	Precision	Recall	F1score
Train	0.7849	0.9430	0.8567
Dev	0.6151	0.7696	0.6837
Test	0.5607	0.6282	0.6000

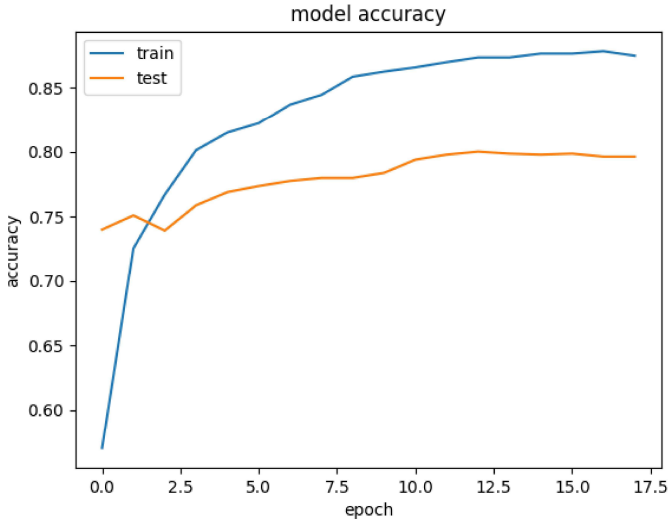

Figura 4.7: Modelo CNN

4.1.4.2. Experimento 2. Modelo PCNN

El modelo PCNN, es una extensión del modelo CNN clásico, en el cual el proceso de max pooling reduce rápidamente el tamaño del mapa de características. Este modelo utiliza este hecho en particular a su favor y realiza un proceso de separación de 3 segmentos donde se aplica el proceso de max pooling para cada uno. Esto permite capturar información estructural más detallada entre entidades. El modelo PCNN presentó los siguientes resultados, se observan en la Tabla 4.7 y en la Fig.4.8

Tabla 4.7: Evaluación de modelo PCNN

Modelo	Precision	Recall	F1score
Train	0.9834	0.9939	0.9886
Dev	0.8596	0.8728	0.8662
Test	0.7285	0.6354	0.6788

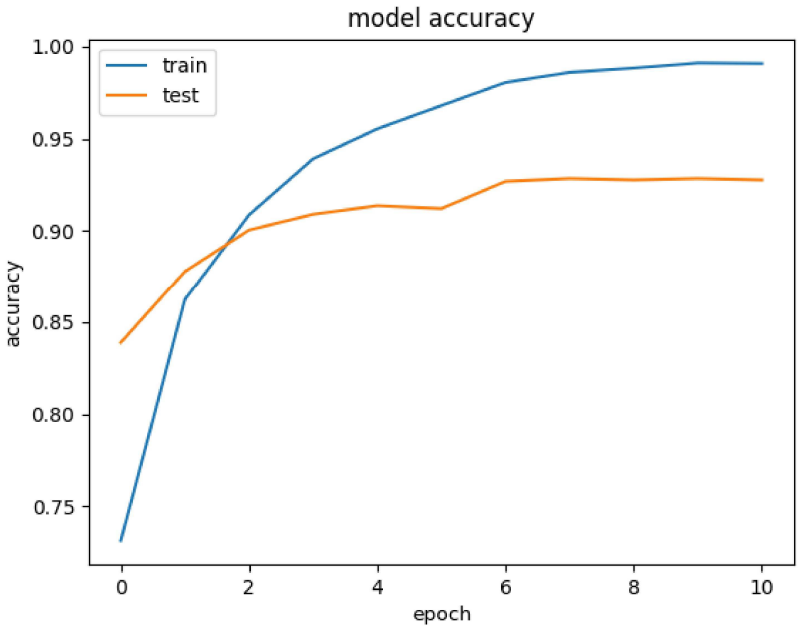

Figura 4.8: Modelo PCNN

4.1.4.3. Experimento 3. Modelo MCCNN

Se propone un tipo de CNN que toma múltiples canales para representar cada palabra en las oraciones de entrada. Basado en la idea de sumar los tres canales paralelos de la imagen (RGB) para su clasificación con CNN, el sistema integra múltiples incrustaciones de palabras de diferentes fuentes para aumentar la información semántica de cada palabra. El modelo MCCNN presentó los siguientes resultados, se observan en la Tabla 4.8 y en la Fig.4.9

Tabla 4.8: Evaluación de modelo MCCNN

Modelo	Precision	Recall	F1score
Train	0.9216	0.9676	0.9440
Dev	0.7084	0.7481	0.7277
Test	0.7283	0.6653	0.7004

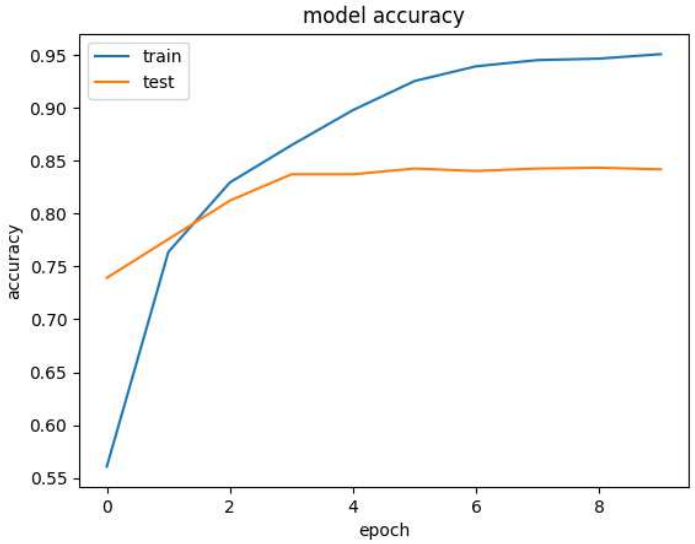

Figura 4.9: Modelo MCCNN

4.1.4.4. Experimento 4. Modelo MCPCNN

La arquitectura de este modelo se basa en los dos modelos anteriores PCNN y MCCNN. El modelo PCNN logra caracterizar el proceso de extracción y captura de la información estructural entre entidades farmacológicas; además, extrajo satisfactoriamente las relaciones entre entidades. El uso de cinco versiones de incrustaciones de palabras como canales de PCNN enriqueció la terminología desconocida. El modelo MCPCNN presentó los siguientes resultados, se observan en la Tabla 4.9 y en la Fig.4.10

Tabla 4.9: Evaluación de modelo MCPCNN

Modelo	Precision	Recall	F1score
Train	0.9478	0.9896	0.9683
Dev	: 0.7838	0.8549	0.8178
Test	0.6895	0.7477	0.7174

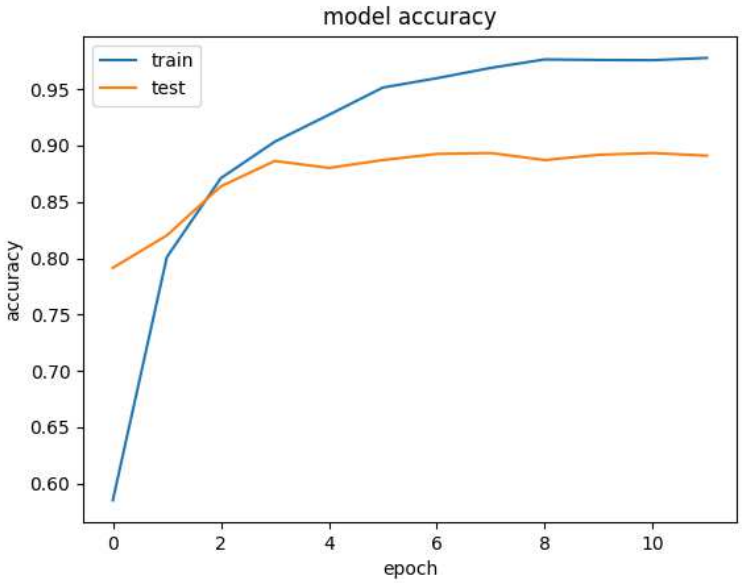

Figura 4.10: Modelo MCPCNN

Reconocer la mejor opción de hiperparámetros es a menudo un proceso engorroso a un nivel que algunas personas consideran arte negro (Snoek, Larochelle, y Adams, 2012). Hay escasez de investigaciones adecuadas sobre el impacto de estos parámetros en el rendimiento de la red, que a menudo conduce a una pérdida de tiempo, especialmente para los investigadores más jóvenes con poca experiencia. En este trabajo, se realizaron muchos experimentos descritos en la sección de ajustes y configuraciones y, de acuerdo a esto, se definió la tabla de parámetros 4.10.

Se obtiene un buen rendimiento del modelo MCPCNN con mecanismos de atención y una capa de ruido gaussiano, F1 score = 73.82. El ruido gausiano realiza un proceso de data augmentation.

Se obtiene un buen rendimiento del modelo MCPCNN con mecanismos de atención y una capa de ruido gaussiano, F1 score = 73.82. El ruido gausiano realiza un proceso de data augmentation.

La Tabla 4.11 muestra el modelo propuesto (MC-PCNN) y los resultados experimentales con los modelos de investigación anteriores (PCNN [6], MCCNN [8])

En la tabla 4.12 podemos observar una mejora en el rendimiento, que resulta luego de añadir un mecanismo de atención, el incremento es de 0.33 puntos. Al añadir ruido gausiano al modelo, la mejora es superior, el incremento es de 2.41 puntos.

Tabla 4.10: Parametros del modelo

Parametro	Valor
Epoch	30
Batch size	64
Optimizer	Adam
Word embedding size	200
Position embedding size	10
CNN kernel size	[3 5 7 9]
Number of filters	128
Dropout rate	0.3
Learning rate	3.00E-04
Gauss noise	0.15
L2	1.00E-07

Tabla 4.11: Evaluación de modelos

Modelo	Precision	Recall	F1score
CNN	0.6797	0.6601	0.6698
PCNN	0.0.7159	0.654	0.6835
MCCNN	0.7242	0.6787	0.707
MCPCNNGauss	0.7359	0.7405	0.7382

Al comparar el modelo propuesto MCPCNNGaus con el modelo CNN; PCNN y MCCNN, se aprecia una mejora en el rendimiento, de 5.47 % p, 8.66 %p y 3.12 % p, respectivamente. En el modelo Bi-LSTM existe un límite de pérdida de la información de la estructura de una oración, por lo cual hay esa diferencia de rendimiento. El modelo PCNN no solo captura mejores detalles, sino que también predice efectivamente las relaciones DDI, al igual que el modelo MCCNN a través de la incrustación multicanal, es capaz de expresar términos con mayor precisión y riqueza de su representación en el espacio.

Tabla 4.12: Efecto de la estrategia para mejor desempeño

Modelo	Precision	Recall	F1score
MCPCNN	0.7197	0.7085	0.7141
MCPCNN+ Att.	0.6895	0.7477	0.7174
MCPCNNG + Att. + Gauss	0.7359	0.7405	0.7382

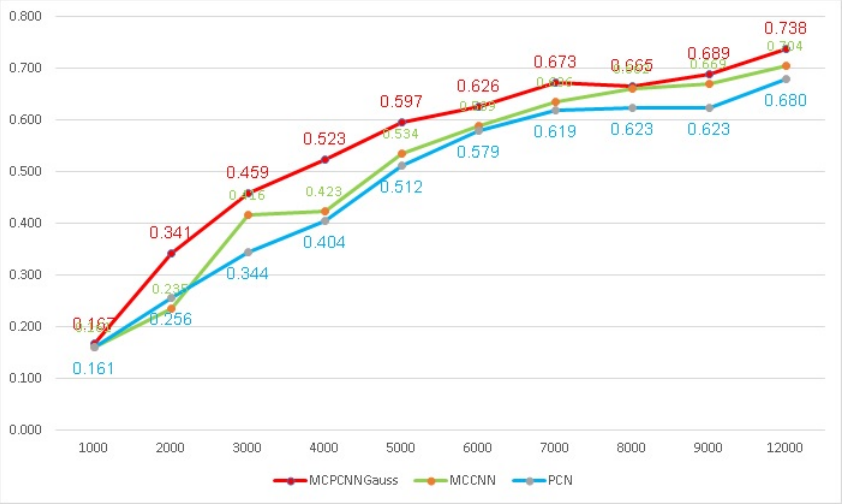

Figura 4.11: Evaluación de modelos según numero de ejemplos

Los resultados finales estan resumidos en la Fig 4.11. En general, observamos que nuestro modelo tiene mejor rendimiento que los demás con distintos números de ejemplos del conjunto de datos DDI. Se observa que nuestro modelo tiene un buen rendimiento incluso con pocos datos de entrenamiento, como se muestra en la gráfica. Esto muestra que nuestro modelo puede ser entrenado con éxito con un pequeño conjunto de entrenamiento.
Comparamos el modelo propuesto con modelos anteriores, desarrollados para la extracción de DDI. Estos enfoques de referencia se evalúan en el mismo conjunto de entrenamiento y conjunto de prueba proporcionados por el desafío DDIExtraction2013. Estos enfoques se pueden dividir en dos categorías: métodos tradicionales y enfoques basados en redes neuronales. En la figura 4.9 los metodos tradicionales se encuentran resaltados con color verde y los metodos de redes neuronales están de color azul. Los métodos tradicionales utilizan herramientas de PLN manuales y características bien diseñadas para entrenar clasificadores supervisados para la extracción de DDI: UTurku(Björne, Kaewphan, y Salakoski, 2013) fue adaptado del Sistema de extracción de eventos de Turku -tem (TEES), que usaba características de análisis de dependencia y recursos dependientes del dominio. WBI (Thomas, Neves, Rocktäschel, y Leser, 2013) combinaba características de varios enfoques

DDI. FBK-irst (Chowdhury y Lavelli, 2013b) usaba características lineales, núcleos de árboles encerrados en rutas y características lingüísticas poco profundas. Kim (S. Kim y cols., 2015) y UVM(Rastegar-Mojarad, Boyce, y Prasad, 2013) Realizaron el mejor analisis de características contextuales, léxicas, semánticas y estructuradas en árbol. Nil (Bokharaeian y Díaz, 2013) Usa un SVM multi clase con metodos de kernel en un marco de trabajo de una sola etapa.

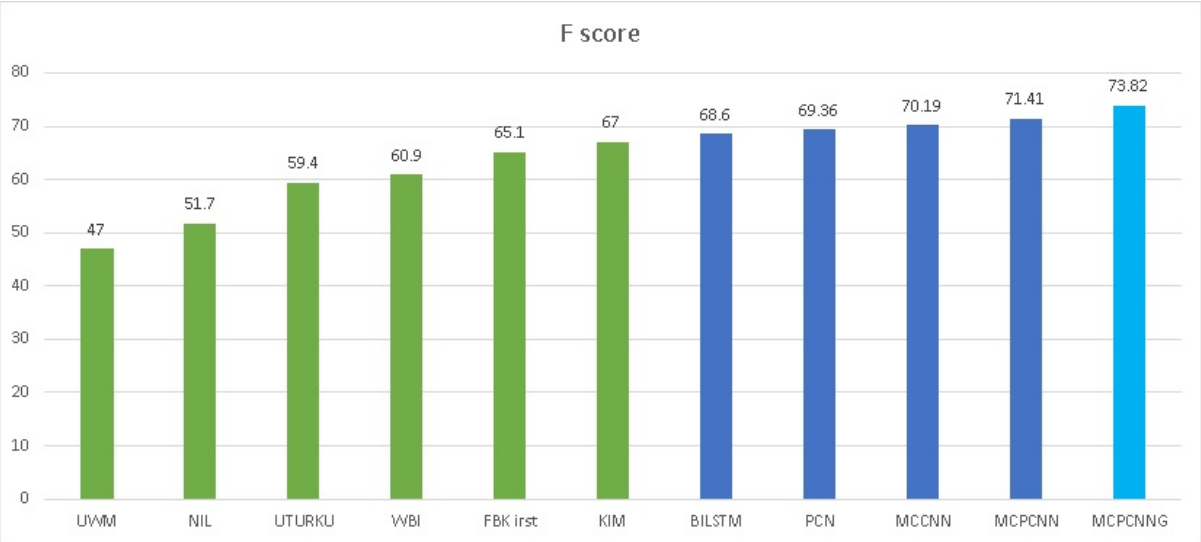

Figura 4.12: Comparación con otros modelos

Los enfoques de color azul estan basados en redes neuronales aprenden representaciones de características de instancias basadas automáticamente en diferentes estructuras de redes neuronales. Estos sistemas pertenecen a estudios recientes basados en las técnicas de aprendizaje profundo para la tarea de clasificación DDI. La barra de azul claro resalta nuestro modelo propuesto

Capítulo 5

Conclusiones

En este trabajo se propuso el modelo MCPCNNGaus para la extracción de las relaciones DDI de la literatura biomédica, el modelo propuesto se lo realizo en base a los enfoques PCNN y MCCNN. El modelo PCNN captura eficientemente la información estructural entre entidades farmacéuticas. El modelo MCCNN aplica 5 métodos de incrustación de palabras, para hacer que los términos desconocidos sean más precisos y ricos. Se aplicó una capa de ruido Gausiano para simular el efecto de data augmentation, a fin de evitar el overfiting. Se utilizó el corpus DDI extraction 2013, para verificar el modelo propuesto y se obtuvo un rendimiento de 2.41 % p en comparación con la linea base.
Se replico el trabajo realizado por los autores (Park y cols., 2019), al cual se aplicó una capa de data augmentation "Gaussian noise", con lo cual se obtuvo una mejora en el rendimiento del modelo; concepto que puede replicarse para otros modelos en el campo bioinformático. Por otro lado, el trabajo de los autores (X. Sun y cols., 2018) que tiene un F1 score de 84.5, no se pudo replicar puesto que no hay mucha información para realizar una replica de este proyecto, se enviaron correos a los autores y no existe repuesta. Estos autores emplearon una red neuronal muy profunda de 15 capas, al parecer no son muy valoradas en este campo este tipo de arquitecturas, puesto que no se logra entender el funcionamiento de la misma (por el efecto de caja negra) y son modelos muy costosos y pesados. Se prefiere la utilización de redes neuronales simples como RNN, CNN y su combinaciones.

Capítulo 6

Trabajos a futuro

En el presente estudio se exploró diferentes enfoques de clasificación de relaciones, sin embargo, sería posible usar estos estudios para aplicarlas en tareas de reconocimiento de entidades con nombre y una combinación de entidad y relación de clasificación en conjunto. En esta tesis, estudiamos diferentes modelos basados en aprendizaje profundo para tarea de clasificacion de relaciones en textos biomédicos o clínicos. Sin embargo, las relaciones de orden superior, como los eventos complejos son de vital importancia en el dominio biomédico. En el futuro, por lo tanto, será importante estudiar sistemáticamente la aplicación de métodos de aprendizaje profundo para tareas de clasificación de relaciones de orden superior.

En el dominio de la visión por computadora, se han publicado algunos trabajos muy relevantes en la línea de funciones de interpretación, sin ir tan lejos se han presentado trabajos excelentes aquí en la EPN FIS, pero en el campo de PLN, solo se han reportado algunos trabajos exploratorios. Ante este panorama, resulta importante que se incursione en la aplicación de estas técnicas de aprendizaje profundo, aplicándolas al PLN.

Para futuros trabajos se planea aplicar las técnicas de interpolación y extrapolación donde se calcula el nuevo vector de incrustación de palabras, es otra técnica de data augmentation.

Referencias

Aghaebrahimian, A., y Cieliebak, M. (2019). Hyperparameter tuning for deep learning in natural language processing. En *4th swiss text analytics conference (swisstext 2019), winterthur, june 18-19 2019*.

Aho, A. V., Sethi, R., y Ullman, J. D. (1986). Compilers, principles, techniques. *Addison wesley*, *7*(8), 9.

Airola, A., Pyysalo, S., Björne, J., Pahikkala, T., Ginter, F., y Salakoski, T. (2008). All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. *BMC bioinformatics*, *9*(S11), S2.

Alves, C. H., y Wijnholds, J. (2018). Aav gene augmentation therapy for crb1-associated retinitis pigmentosa. En *Retinal gene therapy* (pp. 135–151). Springer.

An, G. (1996). The effects of adding noise during backpropagation training on a generalization performance. *Neural computation*, *8*(3), 643–674.

Arji, G., Safdari, R., Rezaeizadeh, H., Abbassian, A., Mokhtaran, M., y Ayati, M. H. (2019). A systematic literature review and classification of knowledge discovery in traditional medicine. *Computer methods and programs in biomedicine*, *168*, 39–57.

Aronson, J. (2004). Drug interactions–information, education, and the british national formulary. *British journal of clinical pharmacology*, *57*(4), 371.

Bengio, Y., Courville, A., y Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, *35*(8), 1798–1828.

Bergk, V., Haefeli, W. E., Gasse, C., Brenner, H., y Martin-Facklam, M. (2005). Information deficits in the summary of product characteristics preclude an optimal management of drug interactions: a comparison with evidence from the literature. *European journal of clinical pharmacology*, *61*(5-6), 327–335.

Björne, J., Kaewphan, S., y Salakoski, T. (2013). Uturku: drug named entity recognition and drug-drug interaction extraction using svm classification and domain knowledge. En *Second joint conference on lexical and computational semantics (* sem), volume 2:*

Proceedings of the seventh international workshop on semantic evaluation (semeval 2013) (pp. 651–659).

Bokharaeian, B., y Díaz, A. (2013). Nil_ucm: Extracting drug-drug interactions from text through combination of sequence and tree kernels. En *Second joint conference on lexical and computational semantics (* sem), volume 2: Proceedings of the seventh international workshop on semantic evaluation (semeval 2013)* (pp. 644–650).

Bond, C., y Raehl, C. L. (2006). Adverse drug reactions in united states hospitals. *Pharmacotherapy: The Journal of Human Pharmacology and Drug Therapy*, *26*(5), 601–608.

Bunescu, R., Mooney, R., Ramani, A., y Marcotte, E. (2006). Integrating co-occurrence statistics with information extraction for robust retrieval of protein interactions from medline. En *Proceedings of the hlt-naacl bionlp workshop on linking natural language and biology* (pp. 49–56).

Businaro, R. (2013). Why we need an efficient and careful pharmacovigilance? *Journal of Pharmacovigilance*.

Choi, E., Bahadori, M. T., Searles, E., Coffey, C., Thompson, M., Bost, J., ... Sun, J. (2016). Multi-layer representation learning for medical concepts. En *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 1495–1504).

Chowdhury, M. F. M., y Lavelli, A. (2012). Impact of less skewed distributions on efficiency and effectiveness of biomedical relation extraction. En *Proceedings of coling 2012: Posters* (pp. 205–216).

Chowdhury, M. F. M., y Lavelli, A. (2013a). Exploiting the scope of negations and heterogeneous features for relation extraction: A case study for drug-drug interaction extraction. En *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 765–771).

Chowdhury, M. F. M., y Lavelli, A. (2013b). Fbk-irst: A multi-phase kernel based approach for drug-drug interaction detection and classification that exploits linguistic information. En *Second joint conference on lexical and computational semantics (* sem), volume 2: Proceedings of the seventh international workshop on semantic evaluation (semeval 2013)* (Vol. 2, pp. 351–355).

Coad, P., Cathers, B., Ball, J. E., y Kadluczka, R. (2014). Proactive management of estuarine algal blooms using an automated monitoring buoy coupled with an artificial neural network. *Environmental modelling & software*, *61*, 393–409.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., y Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of machine learning research*,

12(Aug), 2493–2537.

Cortez, P., y Embrechts, M. J. (2013). Using sensitivity analysis and visualization techniques to open black box data mining models. *Information Sciences*, *225*, 1–17.

Dos Santos, C., y Gatti, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. En *Proceedings of coling 2014, the 25th international conference on computational linguistics: Technical papers* (pp. 69–78).

Duchi, J., Hazan, E., y Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, *12*(Jul), 2121–2159.

Fleuren, W. W., y Alkema, W. (2015). Application of text mining in the biomedical domain. *Methods*, *74*, 97–106.

Gal, Y., y Ghahramani, Z. (2016). A theoretically grounded application of dropout in recurrent neural networks. En *Advances in neural information processing systems* (pp. 1019–1027).

Giuliano, C., Lavelli, A., y Romano, L. (2006). Exploiting shallow linguistic information for relation extraction from biomedical literature. En *11th conference of the european chapter of the association for computational linguistics*.

Goldstein, A., Kapelner, A., Bleich, J., y Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, *24*(1), 44–65.

Goodfellow, I., Bengio, Y., y Courville, A. (2016). *Deep learning*. MIT press.

He, L., Yang, Z., Zhao, Z., Lin, H., y Li, Y. (2013). Extracting drug-drug interaction from the biomedical literature using a stacked generalization-based approach. *PloS one*, *8*(6).

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A.-r., Jaitly, N., ... others (2012). Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal processing magazine*, *29*.

Hinton, G., Srivastava, N., y Swersky, K. (2012). Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, *14*(8).

Hochreiter, S., y Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Hou, W. J., y Ceesay, B. (2018). Extraction of drug-drug interaction using neural embedding. *Journal of bioinformatics and computational biology*, 1840027–1840027.

Jagannatha, A. N., y Yu, H. (2016). Structured prediction models for rnn based sequence labeling in clinical text. En *Proceedings of the conference on empirical methods in natural language processing. conference on empirical methods in natural language processing* (Vol. 2016, p. 856).

Johnell, K., y Klarin, I. (2007). The relationship between number of drugs and potential

drug-drug interactions in the elderly. *Drug safety*, *30*(10), 911–918.

Kaymak, U., Ben-David, A., y Potharst, R. (2012). The auk: A simple alternative to the auc. *Engineering Applications of Artificial Intelligence*, *25*(5), 1082–1089.

Kim, S., Liu, H., Yeganova, L., y Wilbur, W. J. (2015). Extracting drug–drug interactions from literature using a rich feature-based linear kernel approach. *Journal of biomedical informatics*, *55*, 23–30.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Kingma, D. P., y Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kitchenham, B. (2004). Procedure for undertaking systematic reviews. *Computer Science Department, Keele University (TRISE-0401) and National ICT Australia Ltd (0400011T. 1), Joint Technical Report*.

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

Lamurias, A., Sousa, D., Clarke, L. A., y Couto, F. M. (2019). Bo-lstm: classifying relations via long short-term memory networks along biomedical ontologies. *BMC bioinformatics*, *20*(1), 10.

Landau, E. (2009). *Jackson's death raises questions about drug interactions [published in cnn; june 26, 2009]*.

Lazarou, J., Pomeranz, B. H., y Corey, P. N. (1998). Incidence of adverse drug reactions in hospitalized patients: a meta-analysis of prospective studies. *Jama*, *279*(15), 1200–1205.

LeCun, Y., Bengio, Y., y Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436.

Li, Z., Yang, Z., Shen, C., Xu, J., Zhang, Y., y Xu, H. (2019). Integrating shortest dependency path and sentence sequence into a deep learning framework for relation extraction in clinical text. *BMC medical informatics and decision making*, *19*(1), 22.

Lipton, Z. C., Kale, D. C., Elkan, C., y Wetzel, R. (2015). Learning to diagnose with lstm recurrent neural networks. *arXiv preprint arXiv:1511.03677*.

Liu, S., Tang, B., Chen, Q., y Wang, X. (2016). Drug-drug interaction extraction via convolutional neural networks. *Computational and mathematical methods in medicine*, *2016*.

Mahendran, D., y Nawarathna, R. (2016). An automated method to extract information in the biomedical literature about interactions between drugs. En *2016 sixteenth international conference on advances in ict for emerging regions (icter)* (pp. 155–161).

Mikolov, T. (2012). Statistical language models based on neural networks. *Presentation at*

Google, Mountain View, 2nd April, *80*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., y Dean, J. (2013). Distributed representations of words and phrases and their compositionality. En *Advances in neural information processing systems* (pp. 3111–3119).

Miranda, Â., Lavrador, R., Júlio, F., Januário, C., Castelo-Branco, M., y Caetano, G. (2016). Classification of huntington's disease stage with support vector machines: A study on oculomotor performance. *Behavior research methods*, *48*(4), 1667–1677.

Miranda, V., Fede, A., Nobuo, M., Ayres, V., Giglio, A., Miranda, M., y Riechelmann, R. P. (2011). Adverse drug reactions and drug interactions as causes of hospital admission in oncology. *Journal of pain and symptom management*, *42*(3), 342–353.

Moen, S., y Ananiadou, T. S. S. (2013). Distributional semantics resources for biomedical text processing. *Proceedings of LBM*, 39–44.

Moschitti, A. (2004). A study on convolution kernels for shallow semantic parsing. En *Proceedings of the 42nd annual meeting on association for computational linguistics* (p. 335).

Nam, S., Han, K., Kim, E.-k., y Choi, K.-S. (2018). Distant supervision for relation extraction with multi-sense word embedding. En *Proceedings of the 9th global wordnet conference (gwc 2018)* (p. 242).

Nguyen, P., Tran, T., Wickramasinghe, N., y Venkatesh, S. (2016). a convolutional net for medical records. *IEEE journal of biomedical and health informatics*, *21*(1), 22–30.

Ozcift, A., y Gulten, A. (2011). Classifier ensemble construction with rotation forest to improve medical diagnosis performance of machine learning algorithms. *Computer methods and programs in biomedicine*, *104*(3), 443–451.

Papadaki, M. (2017). Data augmentation techniques for legal text analytics. *Department of Computer Science, Athens University of Economics and Business, Athens*.

Park, C., Cho, M., Park, J., y Park, S. (2019). Relation extraction of drug-drug interaction using multi-channel pcnn model. En *Proceedings of the korean society of computer information conference* (pp. 33–36).

Pascanu, R., Mikolov, T., y Bengio, Y. (2013). On the difficulty of training recurrent neural networks. En *International conference on machine learning* (pp. 1310–1318).

Pasi, K. G., y Naik, S. R. (2016). Effect of parameter variations on accuracy of convolutional neural network. En *2016 international conference on computing, analytics and security trends (cast)* (pp. 398–403).

Patterson, J., y Gibson, A. (2017). *Deep learning: A practitioner's approach*. O Reilly Media, Inc.

Prati, R. C., Batista, G. E., y Monard, M. C. (2011). A survey on graphical methods for classification predictive performance evaluation. *IEEE Transactions on Knowledge and Data Engineering*, *23*(11), 1601–1618.

Quan, C., Hua, L., Sun, X., y Bai, W. (2016). Multichannel convolutional neural network for biological relation extraction. *BioMed research international*, *2016*.

Ramaswami, M. (2014). Validating predictive performance of classifier models for multiclass problem in educational data mining. *International Journal of Computer Science Issues (IJCSI), 11*(5), 86.

Rastegar-Mojarad, M., Boyce, R. D., y Prasad, R. (2013). Uwm-triads: classifying drug-drug interactions with two-stage svm and post-processing. En *Second joint conference on lexical and computational semantics (* sem), volume 2: Proceedings of the seventh international workshop on semantic evaluation (semeval 2013)* (pp. 667–674).

Reichartz, F., Korte, H., y Paass, G. (2010). Semantic relation extraction with kernels over typed dependency trees. En *Proceedings of the 16th acm sigkdd international conference on knowledge discovery and data mining* (pp. 773–782).

Schütze, H., Manning, C. D., y Raghavan, P. (2008). *Introduction to information retrieval* (Vol. 39). Cambridge University Press Cambridge.

Segura-Bedmar, I., Martinez, P., y de Pablo-Sánchez, C. (2011). Using a shallow linguistic kernel for drug–drug interaction extraction. *Journal of biomedical informatics*, *44*(5), 789–804.

Segura Bedmar, I., Martínez, P., y Herrero Zazo, M. (2013). Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013)..

Segura Bedmar, I., Martinez, P., y Sánchez Cisneros, D. (2011). The 1st ddiextraction-2011 challenge task: Extraction of drug-drug interactions from biomedical texts.

Sennrich, R., Haddow, B., y Birch, A. (2015). Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Shen, Y., Yuan, K., Li, Y., Tang, B., Yang, M., Du, N., y Lei, K. (2018). Drug2vec: Knowledge-aware feature-driven method for drug representation learning. En *2018 ieee international conference on bioinformatics and biomedicine (bibm)* (pp. 757–800).

Shorten, C., y Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, *6*(1), 60.

Sietsma, J., y Dow, R. J. (1991). Creating artificial neural networks that generalize. *Neural networks*, *4*(1), 67–79.

Singhal, A., Simmons, M., y Lu, Z. (2016). Text mining genotype-phenotype relationships from biomedical literature for database curation and precision medicine. *PLoS computational*

biology, *12*(11), e1005017.

SrirangamSridharan, S., Srivatsa, M., Ganti, R., y Simpkin, C. (2018). Doc2img: A new approach to vectorization of documents. En *2018 21st international conference on information fusion (fusion)* (pp. 2172–2178).

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., y Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929–1958.

Stapley, B. J., y Benoit, G. (1999). Biobibliometrics: information retrieval and visualization from co-occurrences of gene names in medline abstracts. En *Biocomputing 2000* (pp. 529–540). World Scientific.

Suárez-Paniagua, V., y Segura-Bedmar, I. (2018). Evaluation of pooling operations in convolutional architectures for drug-drug interaction extraction. *BMC bioinformatics*, *19*(8), 209.

Sun, A., Grishman, R., y Sekine, S. (2011). Semi-supervised relation extraction with large-scale word clustering. En *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies* (pp. 521–529).

Sun, X., Dong, K., Ma, L., Sutcliffe, R., He, F., Chen, S., y Feng, J. (2019). Drug-drug interaction extraction via recurrent hybrid convolutional neural networks with an improved focal loss. *Entropy*, *21*(1), 37.

Sun, X., Ma, L., Du, X., Feng, J., y Dong, K. (2018). Deep convolution neural networks for drug-drug interaction extraction. En *2018 ieee international conference on bioinformatics and biomedicine (bibm)* (pp. 1662–1668).

Sutskever, I., Vinyals, O., y Le, Q. V. (2014). Sequence to sequence learning with neural networks. En *Advances in neural information processing systems* (pp. 3104–3112).

Tai, K. S., Socher, R., y Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*.

Thomas, P., Neves, M., Rocktäschel, T., y Leser, U. (2013). Wbi-ddi: drug-drug interaction extraction using majority voting. En *Second joint conference on lexical and computational semantics (* sem), volume 2: Proceedings of the seventh international workshop on semantic evaluation (semeval 2013)* (pp. 628–635).

Tikk, D., Solt, I., Thomas, P., y Leser, U. (2013). A detailed error analysis of 13 kernel methods for protein-protein interaction extraction. *BMC bioinformatics*, *14*(1), 12.

Tompson, J. J., Jain, A., LeCun, Y., y Bregler, C. (2014). Joint training of a convolutional network and a graphical model for human pose estimation. En *Advances in neural information processing systems* (pp. 1799–1807).

Wang, M., Geng, Z., Wang, M., Chen, F., Ding, W., y Liu, M. (2006). Combination of network construction and cluster analysis and its application to traditional chinese medicine. En *International symposium on neural networks* (pp. 777–785).

Westergaard, D., Stærfeldt, H.-H., Tønsberg, C., Jensen, L. J., y Brunak, S. (2018). A comprehensive and quantitative comparison of text-mining in 15 million full-text articles versus their corresponding abstracts. *PLoS computational biology*, *14*(2), e1005962.

Xu, B., Shi, X., Zha, Z., Zheng, W., Lin, H., Yang, Z., ... Xia, F. (2018a). Full-attention based drug drug interaction extraction exploiting user-generated content. En *2018 ieee international conference on bioinformatics and biomedicine (bibm)* (pp. 560–565).

Xu, B., Shi, X., Zha, Z., Zheng, W., Lin, H., Yang, Z., ... Xia, F. (2018b). Full-attention based drug drug interaction extraction exploiting user-generated content. En *2018 ieee international conference on bioinformatics and biomedicine (bibm)* (pp. 560–565).

Xu, B., Shi, X., Zhao, Z., y Zheng, W. (2018). Leveraging biomedical resources in bi-lstm for drug-drug interaction extraction. *IEEE Access*, *6*, 33432–33439.

Xu, K.-L. (2008). Bootstrapping autoregression under non-stationary volatility. *The Econometrics Journal*, *11*(1), 1–26.

Zelenko, D., Aone, C., y Richardella, A. (2003). Kernel methods for relation extraction. *Journal of machine learning research*, *3*(Feb), 1083–1106.

Zeng, D., Liu, K., Chen, Y., y Zhao, J. (2015). Distant supervision for relation extraction via piecewise convolutional neural networks. En *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 1753–1762).

Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J., y cols. (2014). Relation classification via convolutional deep neural network.

Zhang, R., Liu, Q., Cui, H., Wang, X., Song, S., Huang, G., y Feng, D. (2018). Thyroid classification via new multi-channel feature association and learning from multi-modality mri images. En *2018 ieee 15th international symposium on biomedical imaging (isbi 2018)* (pp. 277–280).

Zhang, S., Zheng, D., Hu, X., y Yang, M. (2015). Bidirectional long short-term memory networks for relation classification. En *Proceedings of the 29th pacific asia conference on language, information and computation* (pp. 73–78).

Zhang, Y., y Lu, Z. (2019). Exploring semi-supervised variational autoencoders for biomedical relation extraction. *Methods*.

Zhang, Y., Zheng, W., Lin, H., Wang, J., Yang, Z., y Dumontier, M. (2017, 10). Drug–drug interaction extraction via hierarchical RNNs on sequence and shortest dependency paths. *Bioinformatics*, *34*(5), 828-835. Descargado de https://doi.org/10.1093/

bioinformatics/btx659 doi: 10.1093/bioinformatics/btx659

Zhao, D., Wang, J., Lin, H., Yang, Z., y Zhang, Y. (2019). Extracting drug–drug interactions with hybrid bidirectional gated recurrent unit and graph convolutional network. *Journal of Biomedical Informatics*, *99*, 103295.

Zhou, D., Miao, L., y He, Y. (2018). Position-aware deep multi-task learning for drug–drug interaction extraction. *Artificial intelligence in medicine*, *87*, 1–8.