|  |  |
| --- | --- |
| **Detección de preguntas deshonestas en Quora** | |
|  | |
|  | **Alberto Sánchez Mazarro**  Máster Universitario en Ciencias de datos  Procesamiento de Lenguaje Natural    **Tutor/a de TF**  Joaquim Moré López  **Profesor/a responsable de la asignatura**  Toni Lozano Bagén    Fecha Entrega |

[Llicència de Creative Commons](http://creativecommons.org/licenses/by-nc-nd/3.0/es/)

GNU Free Documentation License (GNU FDL)

Copyright © 2023 Alberto Sánchez Mazarro.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

Ficha del Trabajo Final

|  |  |
| --- | --- |
| **Título del trabajo:** | Detección de preguntas deshonestas en Quora |
| **Nombre del autor/a:** | Alberto Sánchez Mazarro |
| **Nombre del Tutor/a de TF:** | Joaquim Moré López |
| **Nombre del/de la PRA:** | Nombre y dos apellidos |
| **Fecha de entrega:** | Enero 2023 |
| **Titulación o programa:** | Máster Universitario en Ciencia de Datos |
| **Área del Trabajo Final:** | Área de Procesamiento de Lenguaje Natural |
| **Idioma del trabajo:** | Castellano |
| **Palabras clave** | PNL, Sentiment Analysis |
| **Resumen del Trabajo** | |
| Este trabajo pretende utilizar técnicas de Procesamiento de Lenguaje Natural (PLN) para tratar de detectar preguntas deshonestas o insinceras en Quora.  Quora es una plataforma digital que permite a sus usuarios la interacción mediante la formulación de preguntas y la contestación de las mismas. En ocasiones, se detectan en la plataforma algunos comentarios que no tienen como fin enriquecer el contenido de esta, sino utilizar mensajes propagandísticos o con fines no del todo honestos.  El objetivo del presente trabajo es tratar de ver hasta dónde el PLN y Análisis de Sentimientos es capaz de llegar a la hora de detectar este tipo de opiniones.  También se pretende analizar si existen aspectos o temas sobre los que existe una mayor probabilidad de encontrar opiniones deshonestas o, por el contrario, éstas son independientes del “topic” sobre el que se esté debatiendo.  Se estudiará cómo la aparición en las preguntas de nombres de entidades influye en el hecho de que la pregunta sea calificada como insincera.  Partiendo de un conjunto de datos de entrenamiento que califica previamente las opiniones en “sinceras” o “deshonestas”, se entrenará un sistema que sea capaz de obtener una tasa de acierto suficientemente alta en la detección. Se usan diferentes técnicas para intentar obtener un modelo lo más preciso posible.  Este trabajo está inspirado en una competición que existe en la web Kaggle para ayudar a Quora en la detección de usuarios y preguntas insinceras que pretenden generar spam en la plataforma. Sin embargo, queda fuera del objetivo de este proyecto participar en dicho concurso. | |

Index

[1. Introducción 1](#_Toc120895991)

[1.1. Contexto y justificación del Trabajo 1](#_Toc120895992)

[1.2. Objetivos del Trabajo 2](#_Toc120895993)

[1.3. Impacto en sostenibilidad, ético-social y de diversidad 2](#_Toc120895994)

[1.4. Enfoque y método seguido 2](#_Toc120895995)

[1.5. Planificación del trabajo 3](#_Toc120895996)

[1.6. Breve sumario de productos obtenidos 3](#_Toc120895997)

[1.7. Breve descripción de otros capítulos de la memoria 4](#_Toc120895998)

[2. Estado del arte 5](#_Toc120895999)

[2.1. Introducción 5](#_Toc120896000)

[2.2. Estado del arte de trabajos similares 5](#_Toc120896001)

[2.3. Conceptos teóricos de aplicación en este trabajo 7](#_Toc120896002)

[3. Materiales y métodos 10](#_Toc120896003)

[3.1. Juego de datos 10](#_Toc120896004)

[3.2. Análisis de las opiniones 12](#_Toc120896005)

[3.3. Clasificación de las opiniones. 16](#_Toc120896006)

[4. Resultados 21](#_Toc120896007)

[5. Conclusiones y trabajos futuros 22](#_Toc120896008)

[6. Glosario 23](#_Toc120896009)

[7. Bibliografía 24](#_Toc120896010)

[8. Anexos 26](#_Toc120896011)

# Introducción

El Procesamiento del Lenguaje Natural (PLN) es un campo dentro de la Inteligencia Artificial (IA) que permite a las máquinas “comprender” el lenguaje humano,

Existe mucha controversia sobre el propósito ético de aplicaciones utilizadas mediante estas técnicas, por la cantidad de información que, sin nuestro consentimiento, algunas máquinas podrían obtener de nuestras conversaciones.

Sin embargo, es una ciencia emergente, que cada vez tiene aplicación en más áreas y que permite automatizar tareas que antes tenían que ser realizadas por humanos, en ocasiones de una manera bastante tediosa.

El PLN se ha aplicado en aspectos cotidianos de nuestra vida diaria, como la traducción de textos o la predicción de las siguientes palabras cuando tecleamos un texto en un buscador. Pero su uso va más allá: es posible extraer automáticamente la información de un texto, detectar los temas que trata o extraer los sentimientos que se desprenden del mismo. Pero, ¿hasta qué punto con la IA somos capaces de realizar tareas finas como detectar si una opinión está escrita de forma insincera?

En los próximos apartados de este capítulo se enmarcará el contexto en el que se realiza este trabajo, se definirán los objetivos del mismo y su enfoque. Adicionalmente se incluye una planificación que fue cumplida durante el desarrollo del trabajo y se indican los resultados a obtener, que serán detallados en posteriores capítulos.

## Contexto y justificación del Trabajo

Este trabajo plantea hasta qué punto puede llegar la IA a ser capaz de detectar preguntas insinceras entre los usuarios de la web Quora. También se estudia si los resultados obtenidos pueden ser extrapolables a contextos similares o se pierde capacidad de predicción si el ámbito del conjunto de opiniones se modifica.

Quora es una plataforma que permite a los usuarios el intercambio de información mediante la realización de preguntas que pueden ser contestadas por otros usuarios. Sin embargo, Quora ha detectado que existen en su plataforma opiniones deshonestas con fines propagandísticos no éticos que acaban tergiversando el propósito inicial de Quora, al convertir algunos foros en discusiones manipuladas y donde se vierte información no real.

Este TFM pretende explorar diferentes aspectos del PLN para encontrar características que caractericen a opiniones insinceras. Se hará un estudio sobre la influencia de los nombres de entidades en las opiniones para que estas sean calificadas como insinceras, y se analizará si estos modelos de detección pueden ser utilizados sobre conjuntos de datos diversos o, por el contrario, pierden precisión.

Desde un punto de vista personal, esta área de trabajo me interesa particularmente. En el momento en el que lo inicio, estoy embarcado en un proyecto profesional en el que pretendemos clasificar diferentes tipos de documentos recibidos en papel en base a su contenido. En general el PLN es una técnica que, con su evolución y refinamiento, puede eliminar mucho trabajo rutinario de personas que podrían emplear sus destrezas en otras labores más técnicas

## Objetivos del Trabajo

* Implementar un sistema que evalúe si una pregunta realizada en Quora es legítima o no.
* Evaluar si hay temas o aspectos donde son más frecuentes las opiniones fraudulentas.
* Analizar la relación entre la aparición de nombres de entidades en la pregunta y el hecho de que ésta sea catalogada como insincera
* Proponer posibles mejoras del sistema que no pueden llegar a ser implementadas en el ámbito de este proyecto.

## Impacto en sostenibilidad, ético-social y de diversidad

La metodología y procedimientos empleados para la elaboración de este trabajo tendrá en cuentas los principios de Comportamiento Ético global.

Las opiniones escritas en Quora serán analizadas independientemente del sexo, raza o condición del autor o autora. Por la propia naturaleza de los datos, que no contienen información acerca del usuario que realizó la pregunta, se garantiza que no habrá sesgos con factores como el sexo o raza del usuario o usuaria para determinar si éste puede hacer más o menos probable que una opinión la podamos considerar legítima. En ningún caso sería propósito del presente trabajo estigmatizar a un colectivo en función de los resultados obtenidos. De ahí que estas líneas se redacten antes de iniciarse el trabajo, para garantizar la independencia entre el propósito inicial del estudio y las conclusiones que se puedan extraer del mismo.

Como resultado de este trabajo se pretende poder contribuir a ayudar a detectar opiniones que tal vez no respeten los principios éticos que deseamos en la sociedad actual. Quora parece interesado en eliminar todos los comentarios potencialmente dañinos e irrespetuosos, que es lo que se pretende identificar en este trabajo.

## Enfoque y método seguido

El trabajo consiste en la realización de un proyecto nuevo, no en la continuación de ningún trabajo de análisis previo hecho sobre la propuesta de Quora en Kaggle.

Para ello la idea será aplicar las técnicas aprendidas durante el máster, especialmente en la asignatura optativa que cursé de “Análisis de textos y sentimientos” utilizando Python como lenguaje de programación y herramientas para la visualización de gráfiicos.

El objetivo será entrenar un modelo en base a un juego de datos de entrenamiento que proporciona Kaggle, comprobar los resultados obtenidos con un conjunto de prueba y analizar los resultados obtenidos. Estos juegos de datos contienen ya una etiqueta en la que las preguntas están calificadas como honestas o deshonestas, por lo que, aunque haremos algún análisis de las opiniones para encuadrarlas, la labor de clasificación previa no será necesaria. También se realizará un análisis para ver si este modelo es extrapolable a otras plataformas o marcos temporales, ya que, en una sociedad tan cambiante como la actual, el tema del que trata la opinión es relevante en el hecho de que la pregunta sea calificada como insincera.

Adicionalmente a este modelo de predicción, se estudiará información cualitativa con influencia en la detección de preguntas insinceras, haciendo especial hincapié en la aparición de nombres de entidades en preguntas insinceras.

## Planificación del trabajo

Tomando en cuenta el esquema de entregas propuesto, el plan de trabajo realizado se puede asemejar al siguiente diagrama de Gantt.

Escala de tiempo

Descripción generada automáticamente

En cierto modo, se puede tratar de un “scrum modificado” donde cada entrega es un sprint y donde las tareas de bajo nivel se han ido revisando conforme ha ido avanzando el proyecto.

## Breve sumario de productos obtenidos

Como se ha indicado anteriormente, se pretende obtener:

* Un SW capaz de determinar con un grado de acierto elevado si una pregunta es “legítima” o “tendenciosa”. ¿Cómo puede cambiar el porcentaje de acierto si cambian las características de los datos?
* Información cualitativa sobre las características de las preguntas insinceras en la plataforma Quora.
* Información sobre el impacto de la aparición de nombres propios de entidades en las opiniones insinceras.
* Una lista de puntos que pudieran ser interesantes aplicar en siguientes trabajos para dar continuidad al actual.

## Breve descripción de otros capítulos de la memoria

En los siguientes capítulos, se revisará el estado del arte del PLN, introduciendo conceptos que se utilizarán en este trabajo, posteriormente se explicará la tecnología y el método utilizado para llevar a cabo el estudio, se presentarán los resultados, se revisarán las conclusiones obtenidas de los mismos y se propondrán líneas de investigación que puedan ser de interés para trabajos sucesivos.

# Estado del arte

## Introducción

En este capítulo se pretende poner en contexto el estado actual del conocimiento y las aplicaciones existentes sobre los distintos puntos que trata el proyecto.

El PLN es una disciplina que abarca tanto campos relacionados con la lingüística como con tecnológicos y computacionales donde el avance y la evolución de las aplicaciones todavía posee un mayor recorrido.

En general, el procesamiento del lenguaje natural por parte de las máquinas es un procedimiento realmente complicado. La tarea de entender una frase “sencilla” ya implica muchos matices para un procesador automático. Pero el lenguaje natural con el que nos expresamos, tanto de manera oral como escrita, contiene problemas de resolución mucho más difícil.

Por ejemplo, en ocasiones utilizamos pronombres para referirnos a un nombre que ha salido previamente en el discurso, palabras polisémicas cuyo significado pretendido solo puede deducirse dentro de un contexto, expresiones formadas por varias palabras que no son “la suma del significado” de las mismas o expresiones en las que la ironía cambia por completo el sentido de la expresión.

En este capítulo vamos a intentar desgranar cada uno de los elementos que se van a utilizar en este trabajo. Se van a utilizar autores y artículos de referencia para intentar aplicar el conocimiento extraído de ellos a la detección de preguntas “insinceras”.

## Estado del arte de trabajos similares (Estudios sobre Yahoo answers, Reddit)

En este apartado se pretenden recoger las publicaciones más relevantes que se han encontrado sobre estudios similares al que se pretende llevar a cabo en este trabajo así como en trabajos que permiten introducir los conceptos básicos del PLN que serán de aplicación en los siguientes capítulos.

En la primera década del siglo [[1]](#Marcador1) estudió la detección de “calificaciones spam” en valoraciones de productos vendidos en Amazon. Realiza un estudio clasificando las opiniones spam en tres tipos:

* Opiniones deliberadamente no sinceras
* opiniones para hablar de una marca en concreto
* opiniones para hablar de algo diferente al producto y hacer publicidad de otra causa

De este artículo podemos extraer mucho valor. Sin embargo, por la naturaleza de los datos de nuestro estudio de las preguntas de Quora (donde solo tenemos el título de la pregunta) nos debemos centrar en el tipo 1 y tratar de inferir de las palabras y el aspecto de la pregunta si esta es más o menos sospechosa de ser “insincera”.

Adicionalmente se introducen otras variables como el autor de la revisión, que sería un dato bastante interesante para el análisis (una persona con deseo de *spamear* es probable que lo haga más de una vez); sin embargo, en nuestro juego de datos no disponemos de información del usuario que ha realizado la pregunta, por lo que no tenemos la posibilidad de analizar características relativas al autor del comentario.

También en esa misma época [2] trata sobre la detección de spam en la web. Aunque algunos de los criterios que utiliza no aplican al estudio que se pretende llevar a cabo en este trabajo, otros conceptos pueden ser de mucha utilidad.

En concreto, se plantean los conceptos de “emotividad” y “afecto”. Además, detectan que el uso de la voz pasiva también puede ser un indicador importante.

El concepto de “emotividad” se define como la proporción de adjetivos y adverbios sobre nombres y verbos es un factor diferenciador de las opiniones insinceras.

El afecto es calculado en base a categorizar las palabras con un número que indica su positividad o negatividad. Opiniones excesivamente polarizadas tienen más probabilidad de ser insinceras.

Estos factores serán tratados en este trabajo sobre el juego de datos de Quora.

Otros aspectos estudiados en este artículo más relacionados con el aspecto formal de las oraciones no serán tenidos en cuenta en este trabajo, puesto que para detectar opiniones insinceras en Quora nos estamos fijando más en el significado de los mensajes que en su forma, que es un aspecto que se investiga en el trabajo de estos autores, especialmente para la detección de plagios en la web. Además cabe destacar que nuestro juego de datos está compuesto por “títulos de preguntas” donde el aspecto formal no cobra tanta importancia, al ser frases únicas y generalmente cortas.

En estudios más recientes como[3] se clasifican como positivas, negativas y neutras las opiniones escritas en Twitter sobre una empresa de suministro de electricidad en Indonesia.

En este caso los autores tuvieron que realizar el trabajo previo de etiquetar manualmente las etiquetas, punto que viene resuelto en el dataset proporcionado por Quora. Sin embargo, en la etapa de preprocesado de datos hay muchos pasos (eliminación de stopwords, de ruido, tokenización o tratamiento de caracteres especiales) que sí son de aplicación en este trabajo.

Además, utiliza parámetros para tratar de ver qué palabras aparecen en cada grupo con menos frecuencia en los demás, que es uno de los puntos clave a tratar en este trabajo, sobre el que se profundizará más adelante.

[5]vuelve a revisar características del lenguaje utilizado en preguntas de la web. Así, expresiones con un tono no neutral, excesivamente despectivas o con contenido demasiado explícito se detectan como más proclives a ser insinceras.

Utilizan modelos de redes neuronales que han hecho crecer rápidamente la capacidad de estos sistemas de spam, eliminando problemas tradicionales como la captación del sentido en oraciones largas. Utilizando modelos bidireccionales un modelo BERT y llegan a resultados cercanos al 95% de acierto en la detección de spam.

[6]incluye el concepto de “mecanismo de atención” utilizando el concepto de “capsule network”. Indican que puede suceder que no todo un comentario sea insincero, sino fragmentos del mismo. Las cápsulas son grupos de neuronas que permiten separar en un espacio las palabras incluso cuando son muy cercanas en su significado.

## Conceptos teóricos de aplicación en este trabajo

En esta sección se pretende introducir los conceptos teóricos y las referencias a los libros y artículos de donde han sido obtenidos.

[7], [8], [9], [10], [11] son las referencias que introducen los conceptos que serán utilizados para el desarrollo de este trabajo. A continuación, se realiza una introducción de los mismos, explicando su naturaleza y su propósito. Otras publicaciones de interés complementarias serán referenciadas en puntos concretos del siguiente capítulo.

Para el procesado de lenguaje natural existen una serie de técnicas para poder descomponer las frases en unidades que nos permitan trabajar con ellas y procesarlas. Para ello es necesario una labor de "preprocesado”, consistente en determinar si queremos eliminar ciertos caracteres que aparecen en un texto (por ejemplo los paréntesis) o queremos quedarnos simplemente con la raíz o lexema de las palabras. Lenguajes de programación como Python contienen las librerías NLTK y Spacy que ofrecen funciones para estos propósitos. En [12] se añaden algunas nuevas funciones de utilidad, como la posibilidad de obtener “frases outliers”; esto es, frases que, dentro de un texto, difieren considerablemente del contexto general del mismo. En este trabajo no serán de aplicación, puesto que estamos analizando frases aisladas, pero para un análisis de posibles respuestas dentro del foro de Quora, sería de mucha utilidad disponer de una herramienta que “limpiara” el texto de frases que escapan del contexto general de la respuesta.

* + 1. Procesamiento del texto.

Las principales técnicas de PLN consisten en la descomposición del texto mediante una serie de pasos para poder trabajar con las unidades que componen el lenguaje. En función del propósito final de cada análisis estos pasos pueden tener matices diferentes. A continuación se comentan los procesos y técnicas más habituales, que se desarrollarán conforme vayan siendo necesarios en posteriores capítulos.

* *Tokenización*. Es la descomposición en unidades con significado que puedan ser utilizadas en análisis. Puede ir acompañado de otros filtrados, como eliminación de *stopwords* (o palabras que no aportan semántica a la frase) o símbolos que no constituyen palabras, conversión a minúsculas para evitar duplicados, etc.
* *PoSTagging*. Clasificación en categorías gramaticales de los tokens anteriores.
* *Buscador de n-gramas*. En ocasiones las palabras están relacionadas semánticamente con las palabras que están inmediatamente por delante o por detrás en la frase. Existen técnicas para detectarlas, de modo que palabras como “Casa Blanca” o “teléfono móvil” puedan detectarse conjuntamente, ya que puede ser interesante procesarlas como una sola unidad.
* *Lematización y stemización*. Es la transformación de una palabra a su lema o raíz. Esto permite unificar palabras que aparecen en distinto género, plurales y singulares, etc.
* *Detección de sinonimia e hiperonimia*. Permite detectar que se utilizan diferentes palabras para definirse a un mismo concepto por ambos tipos de relaciones.
* *Desambiguación del término (synset)*. Un término puede tener varios significados; por medio del contexto de las palabras que le rodean se puede identificar cuál de ellos está siendo utilizado en cada ocasión.
* *Vectorización del texto (word embeddings)*. Existen diferentes técnicas para convertir los textos en vectores o matrices numéricos para poder ser procesados por modelos. Con la métrica tf.idf exto.
* *Word embedding*. Es la transformación de un token en un vector que lo sitúa en una determinada posición del espacio. Esto permite que palabras con significados similares se encuentren cercanas entre sí. [13] realizó la primera aproximación a esta idea que se ha ido evolucionando hasta representaciones como Word2Vec, que utiliza internamente sistemas de redes neuronales obtiene muy buenos resultados en este ámbito.
* *Bag of words (BOW)*. Modelo que trata las palabras pertenecientes a un texto de forma aislada, despreciando la posible relación que tengan con las palabras que la rodean y su contexto.
* *TF-IDF* (Term frequency – Inverse document frequency) es una métrica que pretende identificar cómo de relevante es una determinada palabra en un t
* *Detección de temas o aspectos*. Permite obtener de un texto los temas o conceptos de los que trata. En 2003 [14] introdujo la Latent Dirichlet Allocation (LDA) como un modelo probabilístico que le asocia a cada palabra la probabilidad de pertenecer a un tema concreto. Para esta aproximación se utiliza un concepto de “bolsa de palabras” (BOW de “Bag of Words”) en inglés donde el orden en que aparecen las palabras no es tenido en cuenta.
* *Named entitity Recognition (NER)*. Es la técnica que permite reconocer entidades con nombre propio dentro de un texto y clasificarlas en tipos.
  + 1. Deep Learning aplicado al PLN

El Deep Learning es una rama del aprendizaje automático que pretende utilizar estructuras lógicas para abordar problemas complejos. Estas estructuras simulan el aprendizaje humano por medio de redes neuronales.

En el estudio [15] se demostró la eficacia de estas redes para realizar algunas de las tareas clásicas del PLN que hemos visto en el apartado anterior (PoS Tagging) y otras como reconocimiento de nombres de entidades o etiquetados semánticos. Una nueva herramienta podía impulsar el alcance del PLN.

En [16] se demostró la eficacia de las redes convolucionales (que son un tipo particular de redes neuronales) para tareas automáticas de traducción, respuesta a preguntas o resúmenes de textos.

[17]presenta un muy interesante resumen de la evolución de las redes neuronales en la práctica del PLN y ha servido de referencia para una mejor comprensión de la evolución histórica del PLN.

Aunque no es objeto del presente TFM adentrarnos en la arquitectura de estas redes, en la bibliografía requerida existe suficiente información para comprender las bases que las soportan. En todo caso, se nombran algunos conceptos que, aunque serán desarrollados posteriormente, conviene citar aquí.

* *LSTM (long short-term memory).* Es un tipo de red neuronal recurrente que permite utilizar el conocimiento extraído de procesar las palabras previas a la actual dentro de un texto
* *BERT (Bidirectional Encoder Representations from Transformers)*. Es otro tipo de red neuronal que permite el aprendizaje en ambas direcciones; esto es, palabras situadas con posterioridad en el texto pueden ser utilizadas para obtener información sobre palabras anteriores.

Estos conceptos y la bibliografía asociada son la base sobre los que vamos a trabajar en el presente trabajo.

# Materiales y métodos

## Juego de datos

* + 1. Introducción

Lo primero a tener en este trabajo es la magnitud del juego de datos y un pequeño análisis exploratorio del mismo. El dataset que ofrece Kaggle tiene 1306122 opiniones, de las cuales 1225312 son consideradas “sinceras” y 80810 “insinceras”.

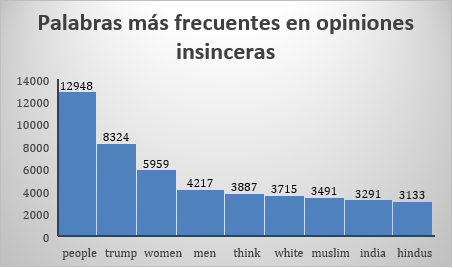
Gráfico

Descripción generada automáticamente

Esta situación nos indica que el juego de datos de que disponemos es grande (más de un millón de ejemplos) y que está “descompensado” en el sentido de que existen muchas más opiniones sinceras que insinceras. Este factor será tenido en cuenta como un posible factor de riesgo de *overfitting*, ya que es posible que algunos modelos tiendan a clasificar masivamente las opiniones como “sinceras”, ya que en el proceso de entrenamiento serán las mayoritarias si tomamos conjuntos aleatorios.

* + 1. Preparación y análisis de los datos

Para poder realizar un trabajo algo más profundo, nos proponemos inspeccionar características de cada uno de los dos tipos de opiniones. Pero antes de eso, es preciso eliminar “ruido” de los datos. Para ello, transformamos los textos a minúsculas, suprimimos las stopwords y analizamos qué palabras son las más frecuentes en opiniones insinceras:



Esta figura ya nos aporta cierta información. Vemos cómo temas relacionados con Donald Trump y, previsiblemente con temas de igualdad de género, problemas raciales o tolerancia religiosa son candidatos a aparecer en las opiniones insinceras.

Sin embargo, parece preciso realizar un estudio más profundo, ya que algunas palabras como “people”, “women” o “men”, que también aparecen con mucha frecuencia parecen ser lo suficientemente genéricas como para que, si bien aparecen en este tipo de preguntas, no sean características de las mismas, ya que también aparecen en las preguntas sinceras.

## Análisis de las opiniones

En estos apartados, para trabajar con conjuntos de la misma magnitud para el estudio de opiniones sinceras e insinceras, hemos tomado un subconjunto aleatorio de preguntas sinceras del mismo tamaño que el conjunto de opiniones insinceras contenidas en el dataset.

A continuación se realizará un análisis más profundo para intentar encontrar características que diferencian a las opiniones sinceras de las insinceras. Para ello se utilizarán diferentes técnicas y métricas y se observará cómo afectan al hecho de considerar una opinión como insincera.

* + 1. TF-IDF

Para tratar de encontrar palabra específicas, utilizamos la mética TF-IDF que se ha introducido anteriormente y que es aplicada en numerosos trabajos previos.

El tf.idf [[18]](#Marcador19) lo podemos ver cómo un medidor de la relevancia de una palabra en un texto con respecto a otros. Es decir, no valoramos solo el número de apariciones de la palabra, sino cómo esa palabra aparece más en un texto que en otros, lo que la hace más representativa. TF (Term Frequency) indica la frecuencia de un texto en un documento, calculándolo como el número de apariciones dividido por el total de términos, mientras que IDF (Inverse Document Frequency) le asigna más peso a las palabras que son infrecuentes pero que aparecen en un documento. IDF = log\_e (numDocumentos/ numAparicionesTérmino)

La multiplicación de TF e IDF dan como resultado la métrica tf.idf, que aplicada a nuestro juego de datos arroja los siguientes resultados:

Tabla

Descripción generada automáticamente

Esta lista hace ver que temas relacionados con la gente, con Donald Trump, con las mujeres o con ciertas religiones o razas son característicos de preguntas insinceras.

En [19] se mejora el concepto de TF-IDF para tener en cuenta las relaciones semánticas entre las palabras, mediante la construcción de un tesauro basado en agrupaciones de palabras sinónimas. Este trabajo es particularmente útil cuando se está trabajando sobre un tema conocido y preferentemente de ámbito científico, donde es más fácil realizar estas agrupaciones. En nuestro caso resultaría de mayor complejidad, puesto que tendríamos que interpretar las intenciones de la persona que redacta la pregunta, lo cual es más complicado en un foro de internet abierto a la realización de preguntas sobre cualquier tema.

* + 1. Emotividad

Otro aspecto a valorar para detectar las opiniones falsas es la “emotividad”, que [2] define como la proporción de adjetivos y adverbios frente a nombres y verbos:

*Emotiveness = (adjectives & adverbs) / (nouns & verbs).*

En nuestro juego de datos obtenemos los siguientes resultados:

Insinceras = (165743 + 36235) / (200375 + 102913) = 0.66

Sinceras = (102493+ 19856) / (186424+ 68913) = 0,47

Se deduce aquí que las opiniones insinceras tienen un grado de emotividad mayor, lo cual parece estar en sintonía con lo que cabría esperar: opiniones exageradas y apelando a la emotividad tienden a ser más insinceras que las que tratan temas más neutrales y objetivos.

* + 1. Polaridad y subjetividad

Con la librería TextBlob de Python es sencillo analizar la polaridad y la subjetividad de una frase.

En el caso de la polaridad, esta se mide entre -1 y 1 (siendo -1 una opinión extremadamente negativa y 1 una opinión completamente positiva).

Haciendo una comparativa de diagramas de cajas de los dos conjuntos, obtenemos los siguientes resultados:

Gráfico

Descripción generada automáticamente con confianza baja

En contra de lo que cabría esperar, no existen diferencias relevantes entre ambos conjuntos, por lo que la subjetividad no parece ser un factor decisivo en la clasificación de las preguntas. La ligera desviación positiva en preguntas insinceras resulta despreciable

Un posible motivo que explique estas situación es el hecho de que las frases sean preguntas. Por esta razón, es posible que las preguntas insinceras estén exista una mayor relevancia en el “topic” que se introduce que en la polaridad de la frase.

Desde el punto de vista de la subjetividad, se detecta un índice superior en las insinceras, con una media de 0.36 en la escala que ofrece TextBlob frente al 0.27 de las sinceras. De nuevo la diferencia no parece demasiado significativa, si bien la mayor subjetividad de las insinceras concuerda con lo que nos dicta el sentido común.

* + 1. Temas o topics

Hacer un análisis de los temas sobre los que tratan mayoritariamente las preguntas insinceras se antoja como una herramienta imprescindible para poder detectarlas. De los resultados obtenidos en el apartado referente a TF-IDF ya se pueden identificar algunos asuntos recurrentes en este tipo de opiniones. Sin embargo, utilizando la técnica de LDA que se introdujo anteriormente, vamos a intentar profundizar algo más en la detección de los temas más frecuentes en las opiniones insinceras.

LDA tiene su implementación en Python, pero revisando brevemente su origen ([14]), lo podemos ver cómo un algoritmo probabilístico para la detección de temas en un texto. Es por ello que, para detectar los temas más frecuentes de todas las opiniones, se procede primero a incluir todas las opiniones en un mismo documento y, a partir de ahí, aplicar las técnicas de preprocesado adecuadas para procesar el texto y extraer sus temas principales.

Fijando en 5 los temas a detectar se obtuvieron los siguientes resultados, representados en diagramas de tartas para ver la influencia de cada palabra.

Gráfico, Gráfico circular

Descripción generada automáticamente

Los cinco temas detectados corresponden claramente con topics susceptibles de ser polémicos y de que se traten en ellos cuestiones insinceras.

* + 1. Categorías sintácticas

Otro punto en el que merece la pena pararse es el análisis de las categorías sintácticas de las palabras que componen cada tipo de pregunta. Este análisis ya se sugería en [2].

Con las funciones de las librería NLTK de Python resulta sencillo realizar esta agrupación de palabras en sus categorías. A continuación se presenta un gráfico que ilustra la cantidad de palabras en cada categoría semántica para cada grupo de opiniones (insinceras y sinceras).

Lo primero que llama la atención en este punto es que, habiendo cogido conjuntos de la misma cardinalidad para preguntas sinceras e insinceras, para todas las categorías se tienen más apariciones en preguntas insinceras. Sin embargo, vemos que en los adjetivos y en los nombres en plural, la diferencia es aún superior, lo que nos permite inferir que frases con estas palabras tienen más probabilidad de ser insinceras.

A raíz de estos datos, cabe preguntarse si la longitud de la frase (entendida como la cantidad de palabras que contiene) es un elemento importante para detectar las opiniones insinceras.

Y detectamos que la media de palabras en una frase insincera es de casi 9 términos mientras que para las sinceras lo es de poco más de 6, por lo que frases más extensas tienen mayor probabilidad de ser insinceras.

## Clasificación de las opiniones.

En este apartado se explorarán diferentes técnicas de clasificación de las opiniones y se analizarán y compararán los resultados obtenidos.

* + 1. Métodos BOW

El método *Bag of Words,* en adelante BOW[10] se caracteriza por el hecho de que las palabras son tenidas en cuenta de una manera individual, sin prestar atención al contexto en el que se encuentran. Esto provoca problemas desde el punto de vista de la desambiguación de las palabras, ya que no podemos utilizar las palabras de su entorno para especificar el significado de la palabra en un contexto, lo cual parece un claro inconveniente en palabras polisémicas.

Para poder procesar las preguntas sinceras o insinceras, lo primero que se precisa es convertir las palabras en vectores numéricos para así poder ser procesadas.

Para ello convertimos las opiniones utilizando de nuevo la métrica TfIdf, que se ha comentado con anterioridad, a nivel de palabra. Dado que nuestro juego de datos está previamente clasificado en opiniones sinceras e insinceras, para desarrollar un predictor simplemente lo tenemos que dividir en conjuntos de entrenamiento y validación y analizar los resultados.

Lo primero que llama la atención de este juego de datos es que está altamente desequilibrado. Se tiene aproximadamente una opinión insincera por cada 15 sinceras. Es por esto que, si cogemos datos aleatoriamente, se produce un sobreentrenamiento u *overfitting,* que ocasiona resultados no deseables, ya que el sistema se acostumbra a detectar la clase mayoritaria (en este caso opiniones sinceras) y tiende a generar un predictor demasiado acostumbrado a detectar preguntas sinceras.

En nuestro caso, tomando una muestra de 1000 preguntas, de las cuales 936 eran sinceras, un algoritmo de regresión logística clasificó las 1000 como sinceras, con lo cual, obteníamos casi un 94% de acierto, pero se queda lejos del propósito inicial de detectar preguntas insinceras.

Para solucionar este problema, es necesario elegir un conjunto equilibrado. Dado que el juego de datos es muy grande, es posible construir conjuntos de entrenamiento y validación con una cardinalidad suficiente como para poder utilizarlos. De hecho, pese a contar con ColabPro para realizar estas clasificaciones, ha sido necesario tomar solo subconjuntos del dataset inicial para poder ejecutar los algoritmos sin problemas de consumo de memoria, ya que tomando conjuntos de más de 60.000 frases se producen desbordamientos de memoria. No se ha considerado necesario aplicar mayores ampliaciones de memoria para el propósito del presente trabajo.

A continuación, se muestran los resultados obtenidos una vez se ha equilibrado el conjunto de pruebas para cada uno de los algoritmos seleccionados. En todos los casos se ha dividido el conjunto total de casos en muestras de entrenamiento y validación con la proporción 80/20, con 50.000 preguntas en el conjunto de entrenamiento y 10.000 en el de validación..

* Regresión logística [24]. Son algoritmos que pretenden explicar una variable cualitativa (en este caso binaria “SINCERO/INSINCERO)” en función de otras variables explicativas (el Tf-Idf de las palabras).

Tabla

Descripción generada automáticamente

* Naive-Bayes [25]. Es un método que interpreta que cada característica de un elemento (en este caso una palabra) contribuye de manera individual al resultado final sin influencia de las demás:

Imagen que contiene Tabla

Descripción generada automáticamente

* SVM [25]. Son las iniciales de “Support Vector Machine”. Sin entrar en consideraciones matemáticas, la idea es representar el espacio en dos regiones donde se pueda diferenciar claramente a cuál de ellas pertenece cada punto:

Tabla

Descripción generada automáticamente

* Random Forest [25]. Utiliza árboles de decisión para predecir el resultado:

Tabla

Descripción generada automáticamente

Por tanto, podemos ver que con la regresión logística y SVM obtenemos resultados cercanos al 87% de acierto, que empiezan a ser valores bastante interesantes.

Para entender mejor los resultados, se explican brevemente estos parámetros:

* La precisión *TP/(FP+TP*) es el número de verdaderos positivos dividido por el número total de positivos detectados (falsos y verdaderos). Es decir, un 87% de precisión de detección de insinceras, indica que de todas las opiniones detectadas como insinceras, el 87% lo eran realmente:
* El recall o exhaustividad *TP/(TP+FN)* indica el porcentaje de opiniones insinceras que podemos detectar. Es decir, un 87% de detección de insinceras indica que de todas las insinceras existentes, detectamos correctamente un 87%.
* El F1 es una combinación de ambas métricas en un solo valor. Se utiliza para dar un valor que tiene en cuenta los dos valores anteriores.
* La seguridad o accuracy *(TP+TN)/(TP+TN+FP+FN)* es la métrica que indica el porcentaje de acierto del algoritmo.

Como nota curiosa, si utilizamos la opción que permite realizar la vectorización TF-IDF eliminando las stopwords, los algoritmos pierden alrededor de dos puntos porcentuales de precisión. Es posible que esta situación esté motivada por el hecho de que, si bien las stopwords no ofrecen “valor semántico”, sí que pueden influir en el estilo de redacción de las preguntas según la finalidad de las mismas.

Una vez obtenidos los resultados anteriores, nos proponemos analizar si es posible obtener mejores resultados aplicando técnicas de aprendizaje profundo o deep learning.

* + 1. Redes neuronales

A continuación, se muestran dos aproximaciones utilizando la biblioteca Keras. Keras es una biblioteca de Python que permite definir modelos de redes neuronales programáticamente abstrayendo al programador de la matemática subyacente a la propia red neuronal. De este modo, con unos conocimientos medios de programación es posible definir una red neuronal.

* + - 1. LSTM

Long Short Term Memory (LSTM) es una técnica de procesamiento del lenguaje en una sola dirección: esto es, de izquierda a derecha (el sentido natural de la lectura). Esto posibilita predecir una palabra (o, en nuestro caso, encontrar el verdadero *synset* si es una palabra polisémica) basándonos en las palabras predecesoras de la misma, pero no permite utilizar palabras posteriores para desambiguar palabras anteriores.

Para el modelo LSTM realizamos dos aproximaciones, ambas respondiendo a un modelo con la siguiente estructura:

Diagrama

Descripción generada automáticamente

Brevemente, la explicación de esta red neuronal sería la siguiente:

* Una capa de entrada donde se recibe el texto
* Una capa de Embedding donde se transforman las palabras en vectores para que puedan ser tratadas por la red neuronal.
* Una capa LSTM donde se procesan estos vectores en la “dirección natural del texto”.
* Una capa de salida “Dense” donde clasificamos la salida en dos clases (opiniones sinceras o incinceras).

Con esta red neuronal obtenemos los siguientes resultados que, como se puede comprobar, son peores que los que obtuvimos con los métodos de clasificación sobre un BOW basado en TF-IDF.

Imagen que contiene Texto

Descripción generada automáticamente

Para intentar mejorar estos resultados, utilizamos un modelo preentrenado de GloVe [[23]](#Marcador23) para el inglés, que otorga unos pesos más adecuados a cada palabra en función de su relación con otras: así, palabras similares, tienen representaciones vectoriales parecidas, del mismo modo que las distancias entre las mismas se representan en ciertas dimensiones. Por ejemplo, “boy” y “girl” estarán cercanas en la “dimensión personal” pero alejadas en cuanto al sexo.

Con respecto al modelo anterior, la única modificación es la capa de “Embedding”, que en lugar de utilizar una codificación cualquiera, lo hace con una más específica (la de GloVE) para cada palabra. Los resultados mejoran notablemente, aunque no llegan a los niveles de precisión obtenidos en el apartado anterior con los métodos SVM o regresión logística aplicados sobre el TF-IDF de modelo BOW.

Imagen que contiene Texto

Descripción generada automáticamente

* + - 1. BERT

BERT (Bidirectional Encoder Representations from Transformers) es otra técnica de Deep Learning que permite superar el problema de LSTM, que tan solo permite tomar los valores en una de las direcciones. Con BERT se pueden usar palabras que aparecen “después” en el texto para determinar el significado de una palabra “anterior”

A continuación, se presenta un modelo con BERT creado con Keras, con una estructura similar a la utilizada en LSTM pero con un modelo preentrenado de BERT y añadiendo una capa de “Dropout” que ofrece Keras para evitar el overfitting.

Diagrama, Esquemático

Descripción generada automáticamente

Las dos capas “KerasLayer” corresponden a sendas capas que ofrece BERT. La primera se utiliza para preprocesar el texto y la segunda para codificarlo para poder trabajar con él en la red neuronal.

Los resultados obtenidos son los siguientes:

Imagen que contiene Tabla

Descripción generada automáticamente

Estos resultados mejoran a los obtenidos con LSTM y se acercan a los mejores resultados obtenidos en el modelo BOW, pero no llegan a superarlos.

# Resultados

En el capítulo anterior se han ido comentando los resultados que se obtenían conforme se desarrollaba cada apartado. En este capítulo se hará un breve resumen de los mismos, tratando de dar una explicación a los mismos y una visión general de los elementos que caracterizan a las opiniones insinceras de este juego de datos.

El primer elemento a destacar es la necesidad de equilibrar el juego de datos para evitar modelos sobreentrenados en opiniones sinceras, que son las mayoritarias.

Desde el punto de vista lingüístico se detectan las siguientes características de las opiniones insinceras.

* Palabras como “*people*” “*women*”, “*Trump*” y “*men*” son características de opiniones insinceras. Por ejemplo, se ha obtenido que el 40% de las opiniones que contienen “Trump”, mientras que solo el 5% de las totales del dataset lo son. Por tanto, podemos decir que Trump multiplica por 8 las probabilidades de que la pregunta sea insincera, pero ni mucho menos lo garantiza (existen más preguntas que hablan de Trump sinceras que insinceras).
* Las frases insinceras tienden a ser más largas que las sinceras.
* El grado de emotividad de la frase (proporción de adverbios y verbos frente a nombres y adjetivos) es superior en las preguntas incinceras.
* Temas relacionados con Trump y su política, con religión, sexo o racismo son proclives a aparecer en las opiniones insinceras.
* La polaridad de las preguntas no parece un factor determinante. Es superior en las opiniones insinceras pero solo ligeramente. Lo mismo sucede con la subjetividad.
* Frases con más adjetivos son propias de las opiniones insinceras. Lo mismo ocurre con los nombres comunes en plural.

Desde el punto de vista de un modelo predictor, el modelo BOW con el criterio TD-IDF y métodos de clasificación SVM o Regresión lineal es capaz de precisar con un 87% de acierto si la opinión es sincera o insincera.

Con métodos de aprendizaje profundo basados en LSTM o BERT no se llega a alcanzar la misma precisión.

En el siguiente capítulo se intentará profundizar algo más en el porqué de estos resultados así como proponer líneas de investigación que se consideran interesantes para mejorar los resultados obtenidos en este trabajo.

# Conclusiones y trabajos futuros

En el presente capítulo se presenta una reflexión del trabajo realizado y un análisis de los resultados obtenidos.

Lo primero que cabe mencionar es el hecho de que este juego de datos está etiquetado con un criterio no completamente objetivo. El simple hecho de que una opinión sea categorizada por sincera o insincera, pese a realizarse desde un punto de vista lo más honesto y objetivo posible, es la decisión de una persona. Algunas de las preguntas podrían cambiar su clasificación en función de la persona que toma la decisión, puesto que pueden estar cerca del límite de lo insincero. Además, cabe preguntarse si todas las preguntas insinceras lo son en la misma intensidad, o si cabría clasificarlas en más grupos en función de “cuánto de tendenciosas” son.

Dicho esto, hemos obtenido características más comunes en opiniones insinceras que en sinceras, pero en ningún caso reglas directa de clasificación. Esto es exactamente lo que cabía esperar; no podemos aspirar a clasificar todas las opiniones correctamente por criterios lingüísticos, el tema que tratan o las palabras más habituales, pero sí que podemos encontrar aspectos que hacen a la pregunta más proclive a ser clasificada como insincera.

Del mismo modo se han generado modelos de clasificación de las opiniones por diferentes métodos. Aplicando BOW con TF-IDF se han obtenido mejores resultados que con LSTM o BERT. Esto, que no era lo que esperaba al inicio del presente trabajo, tiene que ver con las propias características de las frases que constituyen cada pregunta. Al ser frases cortas, la potencia de BERT no es completamente aprovechada, como podría ser en textos más largos. Además, al no tener las opiniones clasificadas por temas, cada pregunta es independiente; no están relacionadas unas con otras y ahí en el modelo BOW, simplemente por el hecho de que ciertas palabras aparezcan con más o menos frecuencia es suficiente para obtener resultados muy apropiados.

Por otro lado, se ha demostrado la importancia del momento en el que se recaban las opiniones; algunas palabras son relevantes porque han sido tomadas en un determinado momento de la historia (por ejemplo “Trump”). Por tanto, la vigencia de los modelos generados está condicionada a que los datos de entrenamiento de los mismos estén relacionados con los datos a clasificar; es decir, las opiniones insinceras de dentro de 30 años contendrán palabras y hablarán de temas diferentes a los aparecidos en el presente trabajo.

En virtud de lo explicado anteriormente, se consideran cumplidos los objetivos iniciales propuestos, si bien con algo más de tiempo y recursos se podrían haber realizado nuevos experimentos y refinamientos de los modelos obtenidos.

La planificación inicial ha sido seguida y cumplida, si bien en algunas situaciones se han intercalado tareas de investigación en función de los resultados obtenidos en alguna de las pruebas y de las sugerencias recibidas por parte del tutor del proyecto. El hecho de tratarse de un proyecto individual en el que solo se requerían consultas con el tutor ha minimizado los riesgos del proyecto y ha permitido cumplir con la planificación a nivel general, sin problemas por pequeñas demoras que rápidamente eran compensadas con momentos de mayor avance, en función de la gestión personal del tiempo de dedicación.

Cabe resaltar que la prueba de diferentes parámetros en el modelo diseñado con BERT ha sido complicada por la duración del entrenamiento. Con más tiempo y mayor capacidad de cómputo, tal vez se podrían haber probado más combinaciones y obtener alguna con un resultado mejor. No obstante, lo obtenido se considera suficiente para la ejecución del trabajo.

No ha existido ningún impacto ético en el desarrollo del trabajo. El hecho de que algunos de los temas más característicos de opiniones insinceras estén relacionados con motivos de género, sexo o religión, nada tiene que ver con el impacto ético del presente trabajo, sino con que algunos temas sociales son más proclives a que la gente vierta sobre ellos comentarios deshonestos.

Para trabajos sucesivos, podría resultar muy interesante añadir en el juego de datos una columna que clasificara correctamente el tema en cuatro o cinco categorías y ver cuál sería el resultado de los algoritmos utilizando un dataset por cada categoría, independientemente de lo utilizado en las otras. De esta manera, palabras muy características en opiniones insinceras sobre un tema determinado, podrían tener mayor impacto sobre las opiniones de dicho tema, mientras que no influirían en otros. Por ejemplo, si hay una noticia que indica que Donald Trump asistió a un partido de fútbol, probablemente el impacto que tiene la aparición de “Trump” en dicha pregunta es bajo, porque no es un tema donde la palabra “Trump” abunde como elemento característico de una opinión insincera.

Pero para llevar a cabo este trabajo es necesaria una ardua labor manual de clasificación de las opiniones con una precisión cercana al 100%. Si se utilizara un clasificador con un porcentaje inferior, los resultados obtenidos podrían no ser válidos. La hipótesis que guiara dicha investigación sería el hecho de que un detector de opiniones insinceras podría obtener un porcentaje de acierto superior si se realizan diferentes entrenamientos con opiniones pertenecientes a un mismo tema que si se procesan todas de forma indiferente, como se ha realizado en este trabajo.

# Bibliografía

* 1. Nitin Jindal and Bing Liu, **Opinion Spam and Analysis Department of Computer Science**, University of Illinois at Chicago, 2008.
  2. J Piskorski J, M Sydow , D Weiss, **Exploring Linguistic Features for Web Spam Detection: A Preliminary Study**, Fourth International Workshop on Adversarial Information Retrieval on the Web, Beijing, China, 2008
  3. R Damarta, A Hidayat, and A S Abdullah **The application of k-nearest neighbors classifier for sentiment analysis of PT PLN (Persero) twitter account service quality**, Journal of Physics: Conference Series, Volume 1722, Tenth International Conference and Workshop on High Dimensional Data Analysis (ICW-HDDA-X) 12-15 October 2020 in Sanur-Bali, Indonesia
  4. A Rachta, G Vanmane, **Detecting Insincere Questions from Text: ATransfer Learning** Approach, ResearchGate 2020
  5. Zhengie Gao , Ao Feng , Xinyu Song, Xi Wu, **Target-Dependent Sentiment Classification With BERT** Department of Computer Science, Chengdu University of Information Technology, 2019
  6. Deepak Kumar Jain, Rachna Jain, Yash Upadhyay, Abhishek Kathuria, Xiangyuan Lan Deep. Refinement: capsule network with attention mechanism-based system for text classification, Neural Computing and Applications (2020) 32:1839–1856
  7. J Moré, **Cómo interpretar y analizar automáticamente la información textual**, Universidad Oberta de Catalunya 2019
  8. J Moré, **Evaluación de la calidad de los métodos de reconocimientos de sentimientos**, Universidad Oberta de Catalunya 2019
  9. J Moré, **Extracción de sentimientos y opiniones,** Universidad Oberta de Catalunya 2019
  10. J Moré, **Implementación de DL aplicado al procesamiento del lenguaje natural,** Universidad Oberta de Catalunya 2019
  11. A Bosch, J Casas, T Bagem, **Deep Learning, principios y fundamentos,** Editorial UOC 2019
  12. A Petukhova, Nuno Fachada, **TextCL: A Python package for NLP preprocessing tasks,** Original Software Publication Volume 19, 101122, July 01, 2022
  13. Y Bengio, R Ducharme, P Vincent, C Jauvin, **A neural probabilistic language model**, Journal of machine learning research, vol. 3, 1137–1155, 2003
  14. D Blei, M Jordan,A Y Ng**, Latent Dirichlet Allocation,** Journal of Machine Learning Research 3 993-1022, 2003
  15. R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, P. Kuksa**, Natural language processing (almost) from scratch**, Journal of Machine Learning Research, vol. 12, no. Aug, pp. 2493–2537, 2011.
  16. M Moreno, J Kalita**, Deep Learning applied to NLP,** University of Colorado Colorado Springs, 2017
  17. T Young, D Hazarika, S Poria, E Cambria, **Recent Trends in Deep Learning Based Natural Language Processing,** 2018
  18. S Qaizer, R Ali, **Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents,** International Journal of Computer Applications (0975 – 8887) Volume 181 – No.1, July 2018
  19. M Kumari, A Jain, A Bhatia, **Synonims based Term Weighted Scheme**, Twelfth International Multi-Conference on Information Processing-2016 (IMCIP-2016)
  20. A Sboev, D Gudovskih, R Rybka, **A quantitative method of text emotiveness evaluation on base of the psycholinguistic markers founded on morphlogical features**, Kurchatoc Institute, Moscow
  21. F Alattar, K Shaaland, **Emerging Research Topic Detection Using Filtered-LDA,** AI (Basel), 2021, Vol.2 (4), p.578-599
  22. «Attention Is All You Need»
  23. J Pennington, R Cocher, D Manning, **GloVe: Global Vectors for Word Representation,** [**https://nlp.stanford.edu/projects/glove/**](https://nlp.stanford.edu/projects/glove/)
  24. M Guillén, M Alonso, **Modelos de Regresión logística,** Universidad Oberta de Catalunya 2019
  25. J Gironés, J Casas, J Minguillón, Minería de datos: Modelos y algoritmos, Editorial UOC 2017

# Anexos

Los notebooks realizados para la generación del presente TFM, junto con los juegos de datos utilizados, se pueden encontrar en la siguiente dirección.

Se han separado en diferentes notebooks en función de las técnicas utilizadas en cada situación, y están pensados para poder ser ejecutados directamente, intentando facilitar la comprensión de qué se está realizando en cada punto

[AskReddit Troll Question Detection Challenge | Kaggle](https://www.kaggle.com/competitions/askreddit-troll-question-detection-challenge/data)