

Procesamiento de lenguaje natural tóxico en redes sociales.

**Nombre Estudiante:** Esperanza Rodríguez-Correa Fernández

Inteligencia de Negocio y Big Data

B2.593 TFM Big data

**Nombre Consultor/a** José Luís Gómez

**Nombre Profesor/a responsable de la asignatura** Josep Curto Díaz

**Fecha Entrega** 06/2019

  
Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](http://creativecommons.org/licenses/by-nc-nd/3.0/es/)

© (Esperanza Rodríguez-Correa Fernández)

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

**FICHA DEL TRABAJO FINAL**

|  |  |
| --- | --- |
| **Título del trabajo:** | *Procesamiento de lenguaje natural tóxico en redes sociales.* |
| **Nombre del autor:** | Esperanza Rodríguez-Correa Fernández |
| **Nombre del consultor/a:** | *José Luis Gómez García* |
| **Nombre del PRA:** |  |
| **Fecha de entrega (mm/aaaa):** | 06/2019 |
| **Titulación:** | *Plan de estudios del estudiante* |
| **Área del Trabajo Final:** | *El nombre y código de la asignatura de TF* |
| **Idioma del trabajo:** | *Español* |
| **Palabras clave** | *NLP, Machine Learning*  *Toxic messages* |
| **Resumen del Trabajo (máximo 250 palabras):** *Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.* | |
| El principal propósito de este proyecto es el reconocimiento de lenguaje tóxico en redes sociales o en cualquier conversación informal realizada entre humamos, como es la partición en foros universitarios.  La propuesta ha sido inspirada por 'perspective' tecnología creada por Jigsaw y el equipo de tecnología de lucha contra el abuso de Google en un proyecto de investigación colaborativo llamado Conversation-AI y finalmente integrada en la biblioteca Tensorflow.  La contribución de este estudio no es solo la comparación del rendimiento de los algoritmos estándar en dos conjuntos de datos con diferentes idiomas (español e inglés) sino que también prueba la necesidad de una biblioteca similar a Vader en español con los comentarios tóxicos más comunes. | |
| **Abstract (in English, 250 words or less):** | |
| The main purpose of this project is to recognize toxic language in the social network or any conversation between humans, such as interactions in a university forum.  This approach has been inspired by 'perspective' technology created by Jigsaw and Google’s Counter Abuse Technology team in a collaborative research project called Conversation-AI and finally integrated into the Tensorflow library.  What this study contributes is not only a project which compares the performance of the gold standard algorithms on two datasets with different languages (Spanish and English) but also probes the necessity of a library similar to Vader in Spanish with the most common toxic comments. | |

**Tabla de contenido**

[1. Introducción 1](#_Toc12749944)

[1.1 Contexto y justificación del Trabajo 1](#_Toc12749945)

[1.2 Objetivos del Trabajo 2](#_Toc12749946)

[1.3 Enfoque y método seguido 3](#_Toc12749947)

[1.4 Planificación del Trabajo 3](#_Toc12749948)

[1.5 Breve sumario de productos obtenidos 1](#_Toc12749949)

[1.6 Breve descripción de los otros capítulos de la memoria 1](#_Toc12749950)

[2. Creación del corpus 2](#_Toc12749951)

[3. Estado del arte 3](#_Toc12749952)

[3.1 LIWC (Linguistic Inquiry and Word Count) 4](#_Toc12749953)

[3.2 VADER 4](#_Toc12749954)

[3.3 IBM Watson Sentiment Analyzer. 6](#_Toc12749955)

[3.4 TextBlob 8](#_Toc12749956)

[4. EDA de los corpus 10](#_Toc12749957)

[4.1. Datos del Jigsaw Unintended Bias in Toxicity Classification Challenge 10](#_Toc12749958)

[4.1.1 Origen de los datos 10](#_Toc12749959)

[4.1.2 Datos de descarga 11](#_Toc12749960)

[4.1.3 Análisis de los datos 12](#_Toc12749961)

[4.2 Datos del Jigsaw Unintended Bias in Toxicity Classification Challenge 16](#_Toc12749962)

[4.2.1 Origen de los datos 16](#_Toc12749963)

[4.2.2 Pre procesamiento de los datos de twitter 16](#_Toc12749964)

[4.2.3 Pre procesamiento de los datos de twitter 17](#_Toc12749965)

[5. Feature\_Engineering 22](#_Toc12749966)

[5.1 Representación de las palabras 23](#_Toc12749967)

[5.2 División del dataset en el de entrenamiento y validación. 24](#_Toc12749968)

[6. Estudio de modelos de Machine Learning 24](#_Toc12749969)

[6.1 Multinomial Naive Bayes 24](#_Toc12749970)

[6.2 SVM. Support Vector Machine 25](#_Toc12749971)

[7. Conclusiones 26](#_Toc12749972)

[Will Sentiment Analysis ever be 100% accurate, or close? 26](#_Toc12749973)

[4. Glosario 27](#_Toc12749974)

[5. Bibliografía 28](#_Toc12749975)

[6. Anexos 30](#_Toc12749976)

**Lista de figuras**

[Ilustración 1 Sesgo en la herramienta perspective. Fuente: civic.mit.edu 2](#_Toc12749977)

[Ilustración 2 Acoso/Toxicidad en redes sociales (Fuente: AMERICAblog) 2](#_Toc12749978)

[Ilustración 3 Planificación del TFM 1](#_Toc12749979)

[Ilustración 4 Vader lengua inglesa matriz de confusión 5](#_Toc12749980)

[Ilustración 5 IBMWatson lengua inglesa matriz de confusión 7](#_Toc12749981)

[Ilustración 6 TextBlow lengua inglesa matriz de confusión 9](#_Toc12749982)

[Ilustración 7. Pantalla usada por la plataforma Civil Comments 10](#_Toc12749983)

[Ilustración 8 Cantidad de comentarios tóxicos 12](#_Toc12749984)

[Ilustración 9 Número de palabras por comentario 13](#_Toc12749985)

[Ilustración 10 Distribución de los comentarios tóxicos 13](#_Toc12749986)

[Ilustración 11 Matriz de confusión 14](#_Toc12749987)

[Ilustración 12 Distribución temporal por grupo de identidades 14](#_Toc12749988)

[Ilustración 13 Número de comentarios tóxicos y no tóxicos 19](#_Toc12749989)

[Ilustración 14 100 palabras consideradas como tóxicas 20](#_Toc12749990)

[Ilustración 15 100 palabras consideradas como no tóxicas 20](#_Toc12749991)

[Ilustración 16 Coeficiente de correlación de Pearson 21](#_Toc12749992)

[Ilustración 17 Resultado sobre el proceso de stemming en castellano. 22](#_Toc12749993)

[Ilustración 18 Mal comportamiento lemma en español 23](#_Toc12749994)

[Ilustración 19 Fórmula explicativa TF-IDF[14] 23](#_Toc12749995)

[Ilustración 20 Clasificador SVM[15] 25](#_Toc12749996)

**Lista de tablas**

[Tabla 1 Modificaciones librería emoticono 2](#_Toc12749620)

[Tabla 2 Resultado traducción en Español 3](#_Toc12749621)

[Tabla 3 Ejemplos de funcionamiento Vader 4](#_Toc12749622)

[Tabla 4 Resultados sobre 253 palabras tóxicas en castellano, Vader 5](#_Toc12749623)

[Tabla 5 Análisis de BIAS en IBM Watson 6](#_Toc12749624)

[Tabla 6 Resultados sobre 253 palabras tóxicas en castellano, IBMWatson 7](#_Toc12749625)

[Tabla 7 Análisis de Bias en TextBlow 8](#_Toc12749626)

[Tabla 8 Resultados sobre 253 palabras tóxicas en castellano, TextBlow 8](#_Toc12749627)

[Tabla 9 Tamaño de los dataset 12](#_Toc12749628)

[Tabla 10 Nuevas columnas del dataframe de Twitter 18](#_Toc12749629)

[Tabla 11 Resultados Bigram 20](#_Toc12749630)

# 1. Introducción

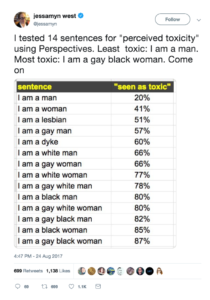
## 1.1 Contexto y justificación del Trabajo

En la actualidad nos encontramos con un problema de lenguaje tóxico y acoso en las redes sociales que herramientas como la de perspective intentan solventar. Uno de los problemas a resolver en el procesamiento de lenguaje natural es la existencia del sesgo en determinadas palabras que según el contexto tienen significados diferentes, así como la existencia de ironía que hace que el significado cambie. Ejemplo de ello, es que las palabras interesante y enriquecedor son positivas pero la frase ‘La película no fue ni enriquecedora ni interesante’ es negativa.

Por otro lado, se identifica otra problemática y es que la mayoría de los desarrollos en librerías y estudios NLP han sido realizados sobre la lengua inglesa o alemana haciendo que los resultados utilizando esas herramientas sobre el análisis en la lengua española dependan de lo bueno o malo que sea un traductor, de si expresiones populares han sido incorporadas al dataset que se utilizó en el entrenamiento, de lo contrario será necesario el desarrollo de librerías propias que contengan las expresiones propias de cada lengua y dialecto .

La plataforma Civil comments fue creada por dos mujeres cuya intención era acabar con los ataques verbales en las redes sociales, planteaban que cuando a un usuario se le indicaba que el lenguaje que estaba utilizando era tóxico, modificaba su mensaje de forma más respetuosa. Sus datos fueron utilizados por el equipo de investigación de Google para la implementación de la API de perspective y es la utilizada como fuente de inspiración en éste estudio.

A través de muestras de los datos de Kaggle y muestras propias tomadas de Twitter vamos a realizar un estudio de cuál es el comportamiento actual de técnicas NLP de reconocimiento de emociones y o lenguaje tóxico en ambos lenguajes (inglés y castellano). Se demostrará la necesidad de elaborar herramientas propias de cada dialecto, los insultos y las expresiones en argentino o mexicano no son los mismos que los utilizados en castellano.



La aparición de la competición de Kaggle este año expone, como muestra la siguiente ilustración, que uno de los grandes problemas que presentan los modelos de Machine Learning es que los datos que se utilizan pueden presentan un sesgo, corriendo el riesgo de que ante nuevos datos los resultados presenten frases como ‘Yo soy una mujer gay negra’ presentan un 87% de toxicidad.

Ilustración 1 Sesgo en la herramienta perspective. Fuente: civic.mit.edu

## 1.2 Objetivos del Trabajo

El objetivo del proyecto es poner en práctica los conocimientos adquiridos sobre análisis de datos, así como la ampliación de ellos con un análisis en profundidad del estado de las técnicas de NLP en el lenguaje español. Se quiere demostrar cómo actualmente las técnicas de procesamiento de lenguaje natural en castellano no tienen un rendimiento adecuado y se busca obtener un método de análisis de lenguaje tóxico inclusivo sin sesgos asociados debido al incremento de acoso y ataques en redes sociales:



Ilustración 2 Acoso/Toxicidad en redes sociales (Fuente: AMERICAblog)

Se concretan en los siguientes puntos:

* Análisis actual de las librerías de Python que permiten el análisis de sentimientos tanto en inglés como en español.
* Análisis actual de los métodos NLP de análisis de sentimientos
* Análisis de los datos con bias de Kaggle
* Creación del corpus con lenguaje tóxico en castellano a través de la API de Twitter
* Creación de librerías que permitan reconocer palabras tóxicas del castellano
* Creación de librerías que permitan sustituir un emoticono por un valor numérico de sentimiento.
* Eliminación de contracciones, @, \_ y símbolos propios de las redes sociales.
* Análisis del corpus de Twitter y de las librerías creadas
* Análisis de los resultados obtenidos con el modelo anterior sobre los nuevos datos de la red social.

Queda fuera de este estudio el reconocimiento de la ironía, siendo cualquier palabra despectiva reconocida como tóxica independientemente del contexto. Se considera que frases como ‘soy lerdo XD’ siguen siendo tóxicas, aunque se utilicen con lenguaje jocoso.

## 1.3 Enfoque y método seguido

Se comenzará analizando los estudios actuales de lenguaje tóxico y cómo se comportan los diferentes enfoques tanto en un dataset propio creado en castellano como en el utilizado en el desarrollo del modelo de Jigsaw. El modelo en castellano se quiere enriquecer con una librería de insultos recopilados en internet que puedan incrementar el peso del análisis, sustitución de abreviaturas, medida del número de signos de admiración, reconocimiento de palabras ofensivas y peso emocional de los emoticonos.

## 1.4 Planificación del Trabajo

En este proyecto, debido al volumen de ambos dataset, en vez de utilizar una plataforma como databricks para el uso de Pyspark que nos permite el análisis de grandes volúmenes de datos por su computación en paralelo, se realizará sobre la plataforma Anaconda con la librería Pandas sobre cuadernos de Jupyter. Ambos datasets no superan en volumen el orden de los GB, por lo que la facilidad de implementación sin tener en cuenta la distribución de los datos en distintos cluster, la flexibilidad y un mayor número de librerías hace que Pandas sea la elección de este proyecto.

Pandas a día de hoy sigue siendo más rápido y más fácil de implementar debido a sus múltiples librerías.

Debido a la necesidad de copiar y pegar código en la memoria final se utilizará la herramienta externa <http://www.planetb.ca/syntax-highlight-word> que permitirá mantener el formato dependiendo del lenguaje.

La forma de citar documentos, páginas web y datos se realizará como aconseja la UOC en su artículo publicado ‘[Citación bibliográfica’](http://biblioteca.uoc.edu/es/recursos/recurso/mendeley-guia-para-la-citacion-de-social-media) . Se usará la extensión de Chrome, el programa y el complemento de Word de Mendeley para el estándar IEEE a la hora de citar los recursos utilizados

Para la planificación de los tiempos se utiliza el programa GanttProject que nos permite realizar y modificar los tiempos de cada tarea para su correcta estimación.

A continuación, se presenta el Diagrama de Gantt con las tareas y los plazos que se desean realizar, los plazos son asignados según los tiempos de las entregas de cada una de las PEC. Se puede encontrar el archivo:

*Referencia:* [Gannt/Proyecto.pdf](https://github.com/SpRdgz/TFM/blob/master/Gannt/Proyecto.pdf)

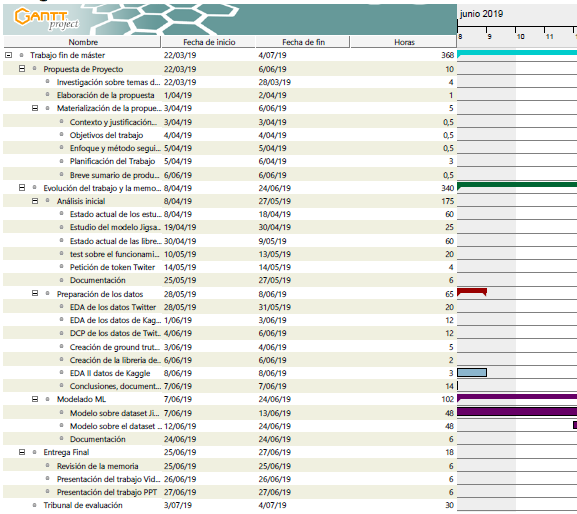


Ilustración 3 Planificación del TFM

## 1.5 Breve sumario de productos obtenidos

La entrega final del proyecto incluye los elementos:

* Repositorio Github
* Memoria del proyecto
* Presentación del trabajo (PPT)
* Comparación de modelos de Machine Learning de reconocimiento de lenguaje tóxico
* Librería de abreviaturas en redes sociales en castellano
* Librería de palabras tóxicas en español
* Corpus de tweets en español expresando emociones
* Comparativa de modelos NLP sobre textos tanto inglés como castellano
* Modelo de predicción de lenguaje tóxico en castellano
* Video con una breve explicación del trabajo realizado

## 1.6 Breve descripción de los otros capítulos de la memoria

Los capítulos que se presentan a continuación son los que incluyen el estudio, recopilación de todo tipo de datos para las pruebas, formación de librerías para finalmente presentar los resultados.

# 2. Creación del corpus

Para la realización de este trabajo se han utilizado las siguientes librerías:

1. Para el análisis del lenguaje tóxico en inglés se ha tomado una muestra de 700 elementos del proporcionado por la competición de Kaggle ‘[Toxic Comment Classification Challenge](https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data).
2. Destinados al análisis de palabras tóxicas ha sido necesaria la creación de una lista de palabras ofensivas tomadas de internet y ampliadas mientras se realizaba el análisis de los tweets así como de noticias que han aparecido en los periódicos.

*Referencia:* [libreria\_insultos.txt](https://github.com/SpRdgz/TFM/blob/master/00_Dataset/libreria_insultos.txt)

1. Corpus de 762 tweets con emociones: Extraídos mediante la API de twitter una vez obtenidas las credenciales a través de su web de desarrolladores: <https://developer.twitter.com/en/apps> . Se han tenido que ir buscando por temáticas para poder conseguir los comentarios de agravio, una vez extraídos han sido etiquetados como tóxicos o no tóxicos también de forma manual

*Referencia:* [data\_sentiment\_labels\_completo.csv](https://github.com/SpRdgz/TFM/blob/master/00_Dataset/data_sentiment_labels_completo.csv)

1. Abreviaturas: De nuevo hay muchos documentos con las abreviaturas más utilizadas en redes sociales en inglés como son ASAP, WTF,.. Pero no se ha encontrado un listado completo en castellano por lo que es producto de los encontrados en el corpus del punto 3.

*Referencia:* [abreviaturas.csv](https://github.com/SpRdgz/TFM/blob/master/00_Dataset/abreviaturas.csv)

1. Emoticonos: Origen en ‘[Emoji Sentiment Ranking v1.0](http://kt.ijs.si/data/Emoji_sentiment_ranking/). Aunque han sido tomados como referencia, se han modificado algunos valores como los siguientes:

Tabla 1 Modificaciones librería emoticono

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| score | emoji | score | emoji | score | emoji |
| -0.017 | 👪 | -0.486 | 🔋 | -0.348 | 👮 |
| -0.313 | 🍱 | -0.033 | ☠ | -0.15; | 💂 |

Emoticonos como la familia, sushi o la policía, tenían valores negativos cuando no es su valor. Mientras que iconos como la muerte tenían muy poco significado negativo, como líneas futuras se quiere realizar un nuevo modelo numérico para este diccionario.

*Referencia:* [Libraries\_Emojis\_sent.csv](https://github.com/SpRdgz/TFM/blob/master/00_Dataset/Libraries_Emojis_sent.csv)

# 3. Estado del arte

En la actualidad existen numerosas librerías cuya intención es la de reconocer los sentimientos en textos escritos en la lengua inglesa. Sin embargo, no se encuentra ningún estudio, librería que haya sido especialmente desarrollado y probado en castellano.

Dentro de los sentimientos, son de especial interés aquellos que desencadenan situaciones de ira o de acoso producto de un sentimiento negativo hacia un tema o persona, por ello se realizará un análisis sobre las librerías y metodologías que han sido desarrollados a día de hoy.

El análisis de comportamiento se ha realizado construyendo una librería de lenguaje tóxico con 242 palabras tomadas del artículo ‘Insultos en castellano que deberías conocer y su significado’ [1] que más tarde servirá para el modelo creado, se ha ampliado a medida que se observaba el lenguaje informal a través de Internet a un total de 253 palabras. La intencionalidad de la creación de esta librería era demostrar si las traducciones podrían llegar a captar las palabras utilizadas como insultos en castellano.

Antes de avanzar hacia el análisis de los métodos implementados. Se realiza un estudio sobre los traductores de Python y el porcentaje de aciertos que tiene sobre la librería de insultos.

Tabla 2 Resultado traducción en Español

|  |  |  |
| --- | --- | --- |
| librería | función | % de aciertos |
| translate | translator.translate(sentence) | 12 de 242 pero a |
| goslate | gs.translate(sentence, 'en') | Corta la comunicación con 20 consultas |
| googletrans | translator.translate(res\_estand\_lib\_ins, dest='en') | 4 de 242 |

Aunque la primera es la que mejores resultados logra incluso por delante de Google translator. Mymemory que es la API con la que se comunica, al igual que Goslate, corta la comunicación con el servidor al pasar alrededor de las 400 consultas, lo que hace que nos sea imposible trabajar sin una traducción manual. MyMemory es una memoria de traducción de escala web colaborativa con más de 40 mil millones de palabras de traducciones profesionales, creada mediante recopilación de TM de la Unión Europea, Naciones Unidas y alineando los mejores sitios web multilingües de dominio específico.

A continuación, se analizan las herramientas, diccionarios y librerías que son más utilizadas en el análisis de sentimientos y que nos permiten realizar una implementación no supervisada, es el caso de la librería NLTK que para realizar una clasificación de sentimientos debe ser entrenado.

## 3.1 LIWC (Linguistic Inquiry and Word Count)

Se destaca esta librería por ser uno de los modelos de referencia en multitud de tareas de análisis de texto como es la detección temprana de depresión en foros de cáncer.

LIWC es un programa diseñado por humanos como sustitución al modelo de machine learning, aunque en muchos estudios se utiliza como complemento a los modelos utilizando su resultado como tag de un algoritmo supervisado, convierte los textos en vectores. Sin embargo, en este proyecto, aunque se menciona porque se quiere utilizar el mismo enfoque haciendo una librería en castellano no se hará una prueba de concepto por ser de pago y no ser de utilidad en textos no formales al no incluir jergas, iniciales y emoticonos demostrado como importante en el análisis de sentimientos en textos sociales[2]. Por todo ello, lo deja fuera de nuestro análisis[3].

## 3.2 VADER

Valence Aware Dictionary and Sentiment Reasoner es una librería de python que surgió de un modelo que mide no solo si un texto tiene sentimiento negativo o positivo sino también la intensidad de las emociones, tiene en cuenta los emoticonos, el argot, las mayúsculas, el número de signos de exclamación….El coeficiente de correlación (r=0.881) mostró como VADER se comportaba tan bien como la evaluación del tag realizado por los humanos de forma manual (r=0.888) y mejor si atendíamos al parámetro de precisión Vader con un F1 de 0.96, mientras que la tasa humana obtenida fue de un 0.84.

Vader mantiene la vía del ‘sentiment lexicon’ como el LIWC o el basado en un glosario definido por humanos, pero su desarrollo está basado en un tipo de texto no formal, contiene listas con los emoticonos, palabras, capitalización de palabras, signos de puntuación positivos y negativos. La última de las diferencias consiste en que es totalmente gratuito y recibe mejoras continuas de los usuarios, publicaron un paper con toda la investigación llevada a cabo [4].

Sin embargo, al estar basado en diccionarios definidos por humanos en español, al realizar la traducción no se comporta de forma satisfactoria:

Tabla 3 Ejemplos de funcionamiento Vader

|  |  |  |
| --- | --- | --- |
| tweet | traducción | puntuación |
| soy una mujer negra | am a black woman | {'neg': 0.0,  'neu': 1.0,  'pos':0.0,  'compound': 0.0} |
| zopenco | Stupidity | {'neg': 1.0,  'neu': 0.0,  'pos': 0.0,  'compound': -0.4404} |
| Soy cristiano | I'm Christian. | {'neg': 0.0,  'neu': 1.0,  'pos':0.0,  'compound': 0.0} |
| Soy un hombre | I am a man | {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0} |
| Soy una mujer | I am a woman | {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0} |

Donde para ser aceptado positivo, negativo o neutro debe tener en ese campo un valor mayor que 0,5... Sobre las 241 palabras de la librería anteriormente mencionada son reconocidas solamente 12, donde estas 12 fueron las traducidas de forma correcta.

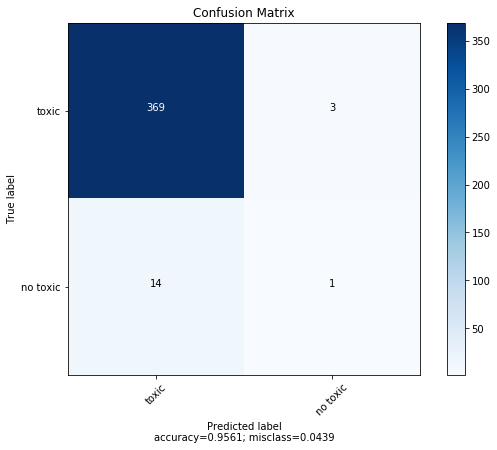
Tabla 4 Resultados sobre 253 palabras tóxicas en castellano, Vader

|  |  |
| --- | --- |
| Número de palabras consideradas como negativas | 12 |
| Número de palabras consideradas como positivas | 3 |
| Número de palabras consideradas como neutras | 233 |

*Referencia:* [pruebaVader.ipynb](https://github.com/SpRdgz/TFM/blob/master/02_State_Of_The_Art/pruebaVader.ipynb)

Una vez obtenidos los resultados con el traductor en el lenguaje español, se comprueba el funcionamiento en el corpus proporcionado por Kaggle, tomamos una muestra de 700 elementos para poder hacer una comparativa justa a la hora de finalizar el estudio.

Ilustración 4 Vader lengua inglesa matriz de confusión



Donde, aunque los resultados no estén balanceados se demuestra que en lengua inglesa , Vader es una de las mejores opciones. La intención que se sigue en este trabajo es crear un Vader en castellano siguiendo sus pautas.

*Referencia:* [PruebaVader\_Jigsaw.ipynb](https://github.com/SpRdgz/TFM/blob/master/02_State_Of_The_Art/pruebaVader_jigsaw.ipynb)

## 3.3 IBM Watson Sentiment Analyzer.

En este apartado se estudia una de las herramientas más utilizadas por las empresas hoy en día debido no a su precisión que se estudiará a continuación sino a su versatilidad, IBM Watson. En 2014, IBM se asoció con Twitter obteniendo así toda la información que esta red social le podía proporcionar para su herramienta analítica[5]. En contraste con VADER no se puede encontrar mucha información sobre el algoritmo o la estrategia que están utilizando.

. Actualmente está disponible en trece idiomas. En cuanto a la configuración, igual que en la API de Twitter hay que registrarse y explicar cuál es el objetivo de tu proyecto para poder obtener el token necesario para hacer que el programa se comunique con su API, devuelve un resultado entre -1 y 1 que indica cuanto de positivo (>0) o negativo (<0) es un comentario y dispone de un número mínimo de tres palabras para poder analizar la sentencia.

Se han realizado dos experimentos:

1. Comprobación de su rendimiento primero tanto con frases mal analizadas por la herramienta perspective como con riesgo de tener alto bias:

Tabla 5 Análisis de BIAS en IBM Watson

|  |  |
| --- | --- |
| tweet | puntuación |
| Soy una mujer gay negra | 0.64716547165 |
| I am a black gay woman | 0 |
| yo soy una mujer! | 0.441942441942 |
| I am a woman! | 0 |
| yo soy un hombre! | 0.501246 |
| I am a man! | 0.352901 |
| Eres imbecil :) | 0.7277830.727783 |
| Eres imbécil :) | -0.897086.897086 |
| yo soy musulmán | -0.65795250646 |
| I am a Muslim | 0 |
| yo soy cristiano | -0.263908 |
| es que soy cristiano | 0 |
| I am Christian | 0 |

Queda comprobada la influencia de los datos sobre el modelo resultante. Los resultados no son los mismos ante un mismo hecho como es el caso de ‘Soy una mujer gay negra’ y ‘I am a black gay woman’ donde en inglés no existe bias pero en castellano Watson lo identifica como una frase de sentimiento negativo el ser mujer gay y negra. Además, mientras que en la lengua inglesa ser hombre o ser mujer están ambos identificados con un sentimiento neutro, en castellano ser mujer tiene un sentimiento menos positivo que ser hombre.

*Referencia:* [IBM\_Watson.ipynb](https://github.com/SpRdgz/TFM/blob/master/02_State_Of_The_Art/IBM_Watson.ipynb)

1. Una vez formada la idea general del funcionamiento de la API de IBM Watson, pasa a calcular cuantas de las palabras identificadas como agravios serían identificadas como negativas, para ello hay que tener en cuenta que IBMWatson tiene un número mínimo de caracteres a la hora de analizar sentencias por lo que ha sido necesario insertar una cabecera (“Tú eres un ”: ). Se decide poner siempre la misma para que no condicione al resultado.

Tabla 6 Resultados sobre 253 palabras tóxicas en castellano, IBMWatson

|  |  |
| --- | --- |
| Número total de palabras en la librería | 253 |
| Palabras etiquetadas como positivas (Falso Positivo) | 57.31% |
| Palabras etiquetadas como negativas (Positivo) | 42.68% |

1. Por último, de nuevo se analiza el comportamiento de la API en la lengua inglesa.

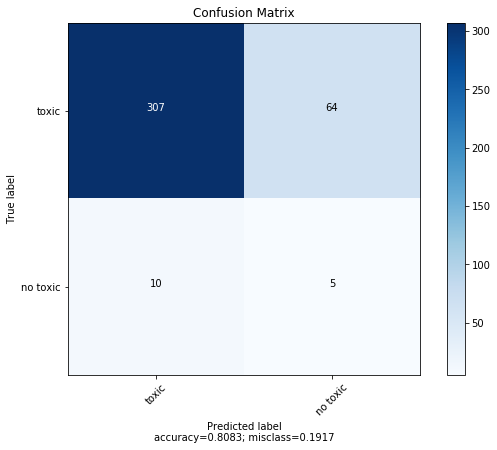


Ilustración 5 IBMWatson lengua inglesa matriz de confusión

*Referencia:* [IBM\_WATSON\_Jigsaw.ipynb](https://github.com/SpRdgz/TFM/blob/master/02_State_Of_The_Art/IBM_WATSON_Jigsaw.ipynb)

Se concluye por todo lo anterior que la herramienta de IBMWatson es capaz de clasificar textos en ingles con un error mínimo mientras que con palabras específicas en texto castellano tiene más del doble de error, un 57%.

## 3.4 TextBlob

´

Librería de Python con multitud de funciones como tokenización, corrección de ortografía y análisis de sentimiento. Devuelve dos valores polaridad que puede ir de -1 a 1 y subjetividad que va de 0 a 1 donde uno es muy subjetivo.

Se realizan las mismas pruebas que anteriormente:

1. Análisis de frases con tendencia a tener bias estudiadas en los casos anteriores:

Tabla 7 Análisis de Bias en TextBlow

|  |  |
| --- | --- |
| tweet | puntuación |
| Soy una mujer gay negra | 0.41666 |
| I am a black gay woman | 0.125 |
| I am a black woman | -0.166666 |
| I am a gay | 0.4166 |
| I am a woman! | 0.0 |
| yo soy un hombre! | 0.0 |
| I am a man! | 0.0 |
| Eres imbecil | 0.0 |
| Eres imbécil :) | 0.5 |
| yo soy musulmán | 0.0 |
| I am a Muslim | 0.0 |
| es que soy cristiano | 0.0 |
| I am Christian | 0.0 |

Textblow es un diccionario con tendencia conservadora si no conoce un símbolo lo pesa como neutro, tiene en consideración las emociones de los emoticonos, obtiene los mismos resultados para las palabras con y sin acentos. No está disponible en castellano solo en francés y alemán.

1. A continuación, se observa si reconoce las palabras tóxicas del lenguaje castellano al ser introducidas en el traductor.

Tabla 8 Resultados sobre 253 palabras tóxicas en castellano, TextBlow

|  |  |
| --- | --- |
| Número total de palabras en la librería | 253 |
| Palabras etiquetadas como positivas (Falso Positivo) | 1.1% |
| Palabras etiquetadas como negativas (Positivo) | 6.3% |

Se concluye que aunque TextBlow tenga un comportamiento adecuado a la hora de evaluar emoticonos y la mayoría de condiciones, no es capaz de reconocer el lenguaje tóxico en castellano. A continuación, se estudia si es adecuado para el reconocimiento del lenguaje negativo en la lengua inglesa.

*Referencia:* [TextBlow\_Castellano.ipynb](https://github.com/SpRdgz/TFM/blob/master/02_State_Of_The_Art/TextBlow_Castellano.ipynb)

1. Al igual que en los casos anteriores se procede a analizar el comportamiento del modelo ante la lengua inglesa con la muestra tomada del dataset de Kaggle:

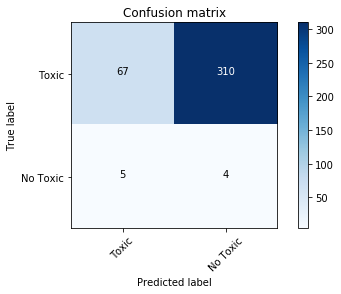


Ilustración 6 TextBlow lengua inglesa matriz de confusión

*Referencia:* [TextBlow\_Jigsaw.ipynb](https://github.com/SpRdgz/TFM/blob/master/02_State_Of_The_Art/TextBlow_Jigsaw.ipynb)

Los resultados son mejores que en el caso del reconocimiento de la lengua castellana. Pero no son satisfactorios, han sido reconocidos como falsos positivos (frases consideradas como tóxicas cuando no lo eran) 310, donde el sentimiento negativo estaba reconocido en los datos.

# 4. EDA de los corpus

Exploratory Data Analysis es el proceso en el que se realizan las primeras investigaciones en los dataset para detectar anomalías, descubrir patones, realizar pruebas que ayuden en la realización del modelo futuro , mostrar resultados de forma gráfica y encontrar posibles correlaciones[6].

## 4.1. Datos del Jigsaw Unintended Bias in Toxicity Classification Challenge

### 4.1.1 Origen de los datos

En el 2017 la plataforma llamada ‘Civil comments’ cerró su soporte y puso a disposición pública todos los comentarios que habían obtenido durante los años que estuvieron en el mercado el fruto fue este dataset.

Civil comments fue una plataforma creada como método de moderación en páginas donde se quería saber si un comentario consistía en un ataque hacia un colectivo/persona, cada vez que alguien escribía un comentario este era puntuado por el resto de los usuarios antes de ser publicado. Surgió como filtro ante comentarios obscenos o de acoso, pero las empresas quitaron sus foro en vez de invertir en su plataforma con lo que tuvieron que cerrar por falta de beneficios.

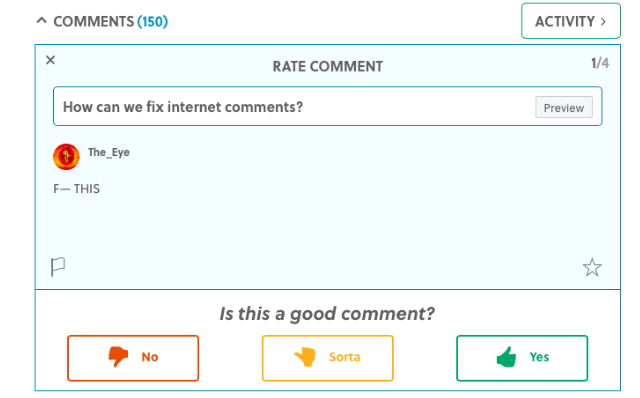


Ilustración 7. Pantalla usada por la plataforma Civil Comments

Google Ideas (**Jigsaw**) decidió utilizar este dataset y junto con ‘The times’ se propusieron lanzar una herramienta de machine learning que pudiera clasificar los comentarios en tiempo real, hasta entonces the times disponía de un equipo que censuraba manualmente[7]. El año pasado lanzaron un challenge en Kaggle que terminó en la herramienta Perspective. Sin embargo, debido a los comentarios que tuvieron en cuenta algunas frases no fueron bien clasificadas y lanzaron este año un nuevo reto cuyo contenido es el utilizado en este proyecto.

### 4.1.2 Datos de descarga

Al realizar la descarga, obtenemos tres archivos. Una primera inspección con Linux permite un esbozo del contenido de los tres;

1. sample\_submission.csv

Este archivo muestra cómo debemos subir los archivos en la plataforma kaggle para que nuestro algoritmo sea analizado. Será el id del comentario con el grado de toxicidad que hayamos calculado.. Las columnas tendrán los nombre id

$ head -2 sample\_submission.csv

id,prediction

7000000,0.0

1. test.csv

Este documento es el núcleo del análisis donde tenemos el campo ‘id’ con el identificador del comentario y el campo ‘comment\_text’ con el contenido del comentario analizar.

$ head -2 test.csv

id,comment\_text

7000000,Jeff Sessions is another one of Trump's Orwellian choices. He believes and has believed his entire career the exact opposite of what the position requires.

1. train.csv

En este archivo ya se ha formado el archivo que será insertado en el procesamiento, tiene 45 características o columnas

$ head -2 train.csv

id,target,comment\_text,severe\_toxicity,obscene,identity\_attack,insult,threat,asian,atheist,bisexual,black,buddhist,christian,female,heterosexual,hindu,homosexual\_gay\_or\_lesbian,intellectual\_or\_learning\_disability,jewish,latino,male,muslim,other\_disability,other\_gender,other\_race\_or\_ethnicity,other\_religion,other\_sexual\_orientation,physical\_disability,psychiatric\_or\_mental\_illness,transgender,white,created\_date,publication\_id,parent\_id,article\_id,rating,funny,wow,sad,likes,disagree,sexual\_explicit,identity\_annotator\_count,toxicity\_annotator\_count

59848,0.0,"This is so cool. It's like, 'would you want your mother to read this??' Really great idea, well done!",0.0,0.0,0.0,0.0,0.0,,,,,,,,,,,,,,,,,,,,,,,,,2015-09-29 10:50:41.987077+00,2,,2006,rejected,0,0,0,0,0,0.0,0,4

,jewish,latino,male,muslim,other\_disability,other\_gender,other\_race\_or\_ethnicity,other\_religion,other\_sexual\_orientation,physical\_disability,psychiatric\_or\_mental\_illness,transgender,white,created\_date,publication\_id,parent\_id,article\_id,rating,funny,wow,sad,likes,disagree,sexual\_explicit,identity\_annotator\_count,toxicity\_annotator\_count

59848,0.0,"This is so cool. It's like, 'would you want your mother to read this??' Really great idea, well done!",0.0,0.0,0.0,0.0,0.0,,,,,,,,,,,,,,,,,,,,,,,,,2015-09-29 10:50:41.987077+00,2,,2006,rejected,0,0,0,0,0,0.0,0,4

### 4.1.3 Análisis de los datos

Se empieza analizando lo más básico del dataset como es de cuantas columnas (tags) dispone y el tamaños del dataset:

Tabla 9 Tamaño de los dataset

|  |  |
| --- | --- |
| Dimensiones del dataset de entrenamiento | 1804874 |
| Dimensión deldataset de test | 97320 |
| Columnas del archivo de entrenamiento: | Index(['id', 'target', 'comment\_text', 'severe\_toxicity', 'obscene',  'identity\_attack', 'insult', 'threat', 'asian', 'atheist', 'bisexual',  'black', 'buddhist', 'christian', 'female', 'heterosexual', 'hindu',  'homosexual\_gay\_or\_lesbian', 'intellectual\_or\_learning\_disability',  'jewish', 'latino', 'male', 'muslim', 'other\_disability',  'other\_gender', 'other\_race\_or\_ethnicity', 'other\_religion',  'other\_sexual\_orientation', 'physical\_disability',  'psychiatric\_or\_mental\_illness', 'transgender', 'white', 'created\_date',  'publication\_id', 'parent\_id', 'article\_id', 'rating', 'funny', 'wow',  'sad', 'likes', 'disagree', 'sexual\_explicit',  'identity\_annotator\_count', 'toxicity\_annotator\_count'],  dtype='object') |

La columna target es la que el modelo debe predecir. Este campo representa la fracción de personas que creyeron que ese comentario era tóxico (>=0.5). Existen 1660540 etiquetados como no tóxicos y 97320 como tóxicos, por lo que se concluye que la mayoría de los campos no son tóxicos Sin embrago no va a ser analizado el dataset completo sino 387 tomados como muestra para medir el desempeño de las herramientas del apartado anterior:

.

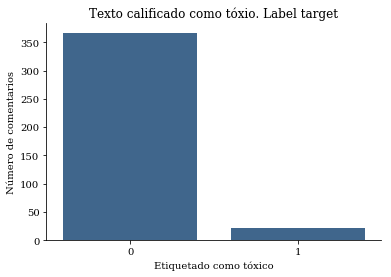


Ilustración 8 Cantidad de comentarios tóxicos

El 99.74% del dataset es de comentarios únicos. En cuanto a la longitud de los comentarios la mayoría contienen entre 5 y 25 palabras a analizar, aunque puede haber hasta 182.

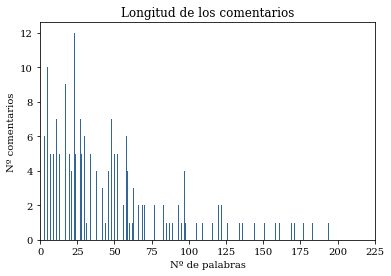


Ilustración 9 Número de palabras por comentario

Por último, se analiza una de las cosas más importantes a la hora de entrenar un dataset, si el dataset no está balanceado (cuando las clases no están representadas de la misma forma).

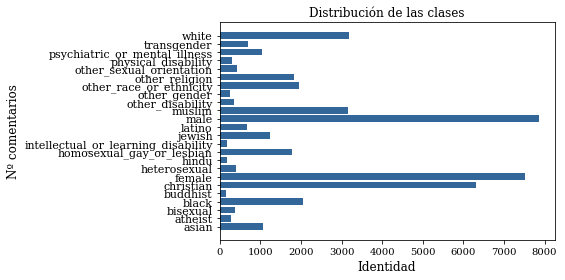


Ilustración 10 Distribución de los comentarios tóxicos

Efectivamente el dataset no está balanceado presenta categorías con una diferencia mayor de 4:1 en cuanto al volumen presente de comentarios etiquetados. En este caso, no se debe tomar el grado de acierto o exactitud como medida de evaluación del modelo. Se debe tomar alguna de las siguientes[8]:

* Matriz de confusión: Que muestra la cantidad de positivos falsos positivos, negativos y falsos negativos de modelo en forma de tabla.



Ilustración 11 Matriz de confusión

* Precisión: Cantidad de positivos reales entre todos los positivos detectados.

* Recall: La cantidad de positivos detectados entre todos los positivos del dataset.

* F1 score o F score: Media entre las medidas precisión y recall, su valor perfecto es 1 y el peor el 0.

* Curva ROC: Es la representación del ratio de verdaderos positivos contra el ratio de falsos positivos o lo que es lo mismo la representación del recall contra (1-especificidad)

Por esta razón fue utilizada la matriz de confusión en las conclusiones del apartado dos.

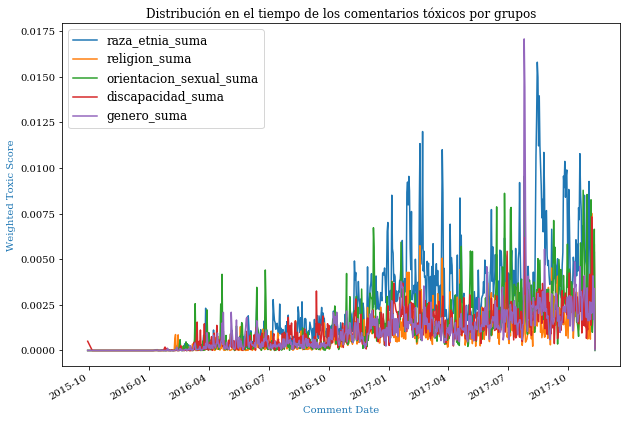


Ilustración 12 Distribución temporal por grupo de identidades

Con esto podemos analizar en que rango temporal fueron obtenidos estos datos y cuando un grupo fue atacado más que otro. Además, si fuera necesario podría ser el corte para balancear los datos y tomar un rango de tiempo en el que se tiene aproximadamente el mismo número de comentarios para cada uno de estos grupos.

*Referencia:* [EDA\_Jigsaw.ipynb](https://github.com/SpRdgz/TFM/blob/master/03_Exploratory_Data_Analysis/EDA_Jigsaw.ipynb)

## 4.2 Datos del Jigsaw Unintended Bias in Toxicity Classification Challenge

### 4.2.1 Origen de los datos

Como se ha mencionado anteriormente, debido a que no se ha encontrado un dataset en castellano con lenguaje tóxico ha sido necesaria la realización de uno propio. Así como la etiqueta manual de cada uno de los 732 comentarios indicando si pertenecía o no a un comentario ofensivo.

El desarrollo para la obtención se ha realizado con Python con llamadas a la API de Twitter, para poder obtener las credenciales de autenticación que posibilitan su uso se ha de justificar y rellenar numerosos campos en los que el usuario indica cual es el propósito de esa extracción de datos.

El código se puede obtener en el repositorio.

*Referencia:* [Twitter\_creacion\_del\_dataset.ipynb](https://github.com/SpRdgz/TFM/blob/master/01_Corpus_Creation/Twitter_creacion_del_dataset.ipynb)

### 4.2.2 Metodología

Como se muestra en el estudio realizado por CrowdFlower, el 80% del tiempo de un científico de datos, se emplea en la limpieza, procesamiento y análisis de los datos, un 20% en el estudio del modelo que se empleará en la realización del algoritmo.

Las buenas prácticas dividen este paso en otros uatro pasos que nos ayudan a entender la importancia y naturaleza de cada uno de los campos de nuestros datos:

1. Limpieza de los datos: Eliminación de caracteres sin información como son los signos de puntuación, la estandarización de los datos en un mismo formato como es la eliminación de mayúsculas (teniendo especial cuidado con las palabras como US, us que cambian el significado al cambiar su morfología) y la sustitución de contracciones por sus palabras completas.
2. Esquematización de los datos: Etiquetación, división de los datos en campos y nombres que el algoritmo pueda entender y utilizar.
3. Normalización: Uso de técnicas como son las que convierten las palabras derivadas a su raíz (escribiendo se dividirá en raíz: escribir y sufijo: -endo ) o el uso de diccionarios con el significado morfológico de cada palabra (la palabra escritura en la [RAE](https://dle.rae.es/?id=GKXHvDj) significa ‘Acción y efecto de escribir.’ el Lema devuelto será: escribir)
4. Análisis: Realización de estadísticas indicativas de los campos con más dependencias, los que no tienen valor para la realización posterior de un análisis de características.

### 4.2.3 Pre procesamiento de los datos de twitter

En el corpus se han dejado fuera los retweets por no querer información duplicada, se ha activado la opción extended para poder obtener los nuevos tweets, con más contenido que en el pasado.

A continuación se realiza, como hemos comentado, la limpieza de los datos debido a que en redes sociales hay contracciones, emoticonos…,El proceso de limpieza ha sido largo hasta poder sacar información de valor. Se han:

* Eliminado cuentas a las que hace referencia (@XXXX)
* Aglutinado múltiples espacios en uno
* Supresión de acentos y números que no afectan a la emoción del texto
* Eliminación de las palabras que comiencen por \_
* Supresión de comillas, pipes | y guiones
* Eliminación del símbolo #, así como de las comillas que no aportaban información en el análisis.
* Supresión de espacios duplicados

En cuanto a la esquematización de los datos. Además de la etiqueta que indica si un comentario es o no tóxico, se añaden indicativos de énfasis con el número de:

* Admiraciones
* Emoticonos
* Palabras en mayúsculas
* Palabras identificadas como tóxicas

La mayoría del lenguaje tóxico está dirigida hacia otras personas por lo que también se ha creado un label indicando que el comentario está dirigido a una tercera persona con el verbo ser. La última de las etiquetad creadas se realiza con la ayuda del artículo ‘Emoji Sentiment Ranking’ [9] de donde ha sido creada otra de las librerías con los emoticonos que aparecen en este corpus y el nivel de sentimiento negativo que tenía cada uno de ellos con esa información se hace una media de los sentimientos teniendo en cuenta cuantos emoticonos hay y con qué valor

Para la normalización solo ha sido necesario poner tanto el texto a analizar como la librería con el mismo formato, en minúsculas y sin tildes Esto se hace debido a que en las redes sociales cada vez más no se respetan las reglas de ortografía. Con el objetivo de no confundir palabras como moño con mono, la ñ se mantiene en el texto.

En cuanto a los resultados, finalmente fue construido un corpus con 762 comentarios tóxicos y no tóxicos obtenidos de twitter en donde la información que nos interesaba era si eran o no eran tóxicos, las etiquetas fueron añadidas manualmente por lo que no hay ningún elemento a nulo en esas dos columnas iniciales.

Tabla 10 Nuevas columnas del dataframe de Twitter

|  |
| --- |
| C:\Users\Esperanza\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\1C68EF7F.tmp |
| C:\Users\Esperanza\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\443924E5.tmp |
| C:\Users\Esperanza\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\382D0D3B.tmp |

Mediante estas representaciones se ha querido observar si había una población realmente representativa de cada una de las categorías tomadas como medios de exaltación de sentimientos y efectivamente se determina que la hay.

El siguiente paso es igual que con el anterior dataset, determina si el corpus está balanceado para la futura toma de decisiones, para que sea considerado como no balanceado debe de haber una diferencia mayor de 4:1 algo que no ocurre como se puede ver en la figura siguiente:



Ilustración 13 Número de comentarios tóxicos y no tóxicos

Para analizar las principales palabras que aparecen y hacer un resumen de los datos se utiliza un wordcloud. Este método proporciona una forma gráfica de ver las palabras más relevantes para cada una de las categorías, en el caso de los comentarios categorizados como tóxicos

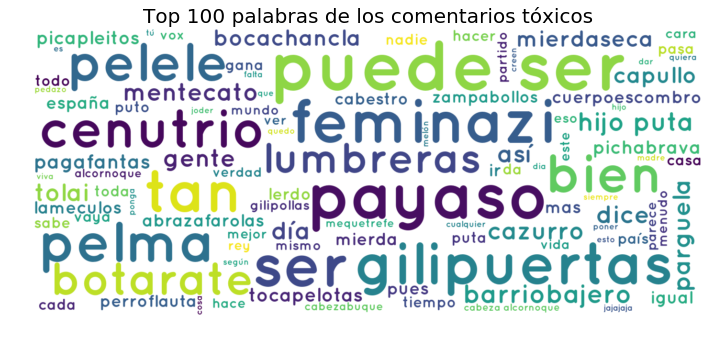


Ilustración 14 100 palabras consideradas como tóxicas



Ilustración 15 100 palabras consideradas como no tóxicas

Con estas dos ilustraciones se observa como sin ninguna intención no solo aparecen palabras que se consideran como tóxicas independientemente del contexto, sino que también hay alguna contextualizada provocada por el tiempo en el que se amplió este corpus para unos mejores resultados.

Además, se utilizaron técnicas de análisis que no llevaron a conclusiones relevantes como fue el estudio de la relación entre palabra conocido como Bigram, este busca en todo el texto manejado cuales son el grupo de palabras que más se repiten alrededor de las demás.

Tabla 11 Resultados Bigram

|  |  |
| --- | --- |
| ('no', 'se', 'puede') | 12 |
| ('se', 'puede', 'ser') | 12 |
| ('puede', 'ser', 'más') | 10 |
| ('', '', '') | 10 |
| ('data', 'science', '') | 4 |
| ('', 'ofertas', 'trabajo') | 3 |

El siguiente factor considerado en el análisis es el del Coeficiente de correlación de Pearson, coeficiente de correlación más usado para medir la asociación entre variables. Cuantifica el grado en el que la relación entre dos variables puede ser descrita por una línea, su formulación es la siguiente[10]:

Los resultados fueron los esperados, el número de insultos efectivamente tiene una alta relación con que el comentario sea tóxico y el nivel de toxicidad depende en gran medida con el número de emoticonos a pesar de que se haga una media entre sus valores, a mayor número de emoticonos más expresividad se le añade al comentario.

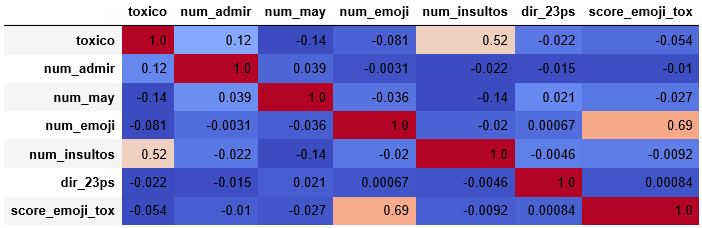


Ilustración 16 Coeficiente de correlación de Pearson

*Referencia:* [EDA\_TWITTER.ipynb](https://github.com/SpRdgz/TFM/blob/master/03_Exploratory_Data_Analysis/EDA_TWITTER.ipynb)

# 5. Feature\_Engineering

Realizado solo sobre los datos de Twitter ya que los de Jigsaw ya vienen procesados.

Denominamos la ingeniería de características como la utilización del conocimiento sobre el dominio de los datos para definir características que permiten que los algoritmos de aprendizaje automático desempeñen sus funciones.

Según Andrew Ng, gurú de la AI:

“Crear funciones es difícil, requiere mucho tiempo y requiere el conocimiento de un experto. "Aprendizaje automático aplicado" es básicamente ingeniería de características”[11]

En este apartado se crearán las mismas etiquetas que utiliza el modelo de Vader, pero primero se realizan las últimas modificaciones necesarias sobre el dataset. Pasos realizados:

* Selección de las columnas de interés
* Redondeo de los valores con coma flotante
* Supresión de las abreviaturas dentro de los comentarios de acuerdo con el diccionario creado (‘q’ es sustituido por ‘que’)
* Conversión del texto a minúsculas
* Suprimimos las palabras vacías del español dejando el ‘no’ porque en este caso si aporta valor[12].
* Tokenización: Proceso que divide las cadenas de texto como son los comentarios en piezas pequeñas de elementos o tokens generalmente son las palabras
* Lemmatización: Proceso de convertir una palabra a su raíz. La diferencia entre stemming y lemmatization es que la lematización considera el contexto y convierte la palabra a su forma básica significativa, mientras que stemming simplemente elimina los últimos caracteres, lo que a menudo conduce a significados incorrectos y errores de ortografía como podemos ver el siguiente ejemplo:

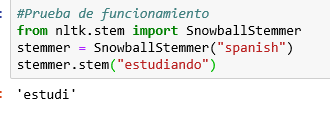


Ilustración 17 Resultado sobre el proceso de stemming en castellano.

Sustituir estudiando por ‘estudi’, simplemente por simplemente no nos proporciona información por lo que queda como prueba de concepto.

Lo que sí se implementa, aunque de nuevo no está totalmente desarrollado en castellano es la lemmanización. A continuación, se muestra otro ejemplo de mal desempeño además de los presentados en <https://github.com/explosion/spaCy/issues/2710> que siguen sin ser solucionados

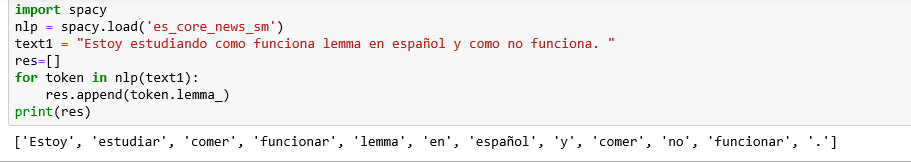


Ilustración 18 Mal comportamiento lemma en español

La frase "Estoy estudiando cómo funciona lemma en español y como no funciona. “da como resultado la lista de tokens con su lemma ['Estoy', 'estudiar', 'comer', 'funcionar', 'lemma', 'en', 'español', 'y', 'comer', 'no', 'funcionar', '.']. como de comparación es sustituido por comer que es incorrecto.

Por último guardamos el resultado en objetos pickle, utilizado cuando se tiene un conjunto de datos grande que se cargan en la memoria cada vez que ejecuta el programa [13].

## 5.1 Representación de las palabras

Consiste en representar las palabras que conforman el texto de forma numérica para que el algoritmo lo pueda entender. Entre todas las técnicas en este estudio se ha utilizado el más usado por ser el que mejor resultados da TF-IDF su significado es "Frecuencia de Término - Frecuencia del Documento Inverso", que son los valores que finalmente se asignan a cada palabra:

* **Frecuencia de término:** Cuanto aparece la palabra en ese documento.
* **Frecuencia de documentos inversa**: esta escala reduce las palabras que aparecen mucho en los documentos.

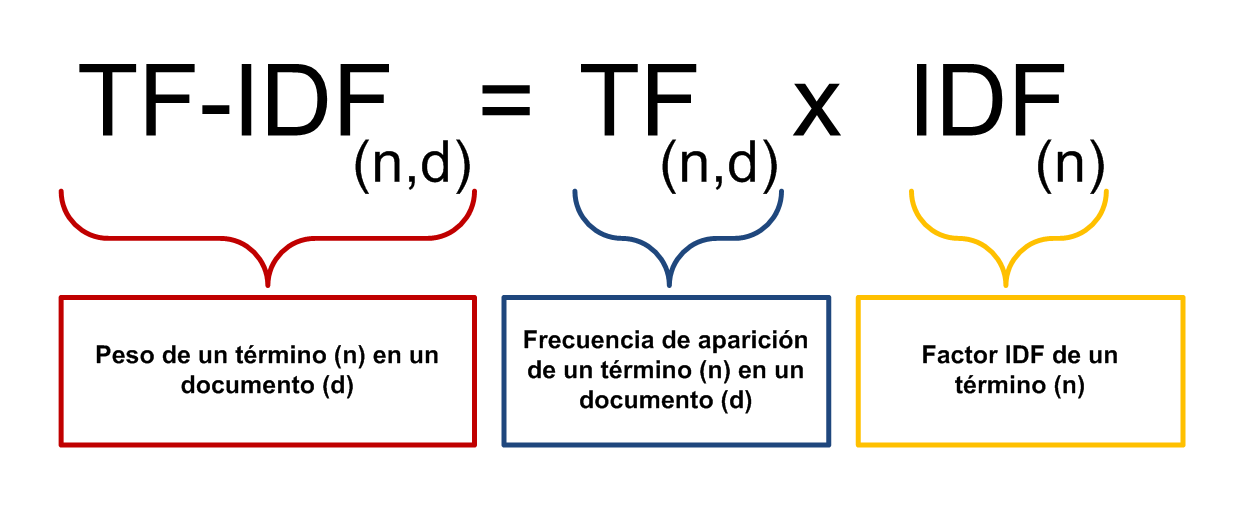
[](http://ccdoc-tecnicasrecuperacioninformacion.blogspot.com/2012/11/frecuencias-y-pesos-de-los-terminos-de.html)

Ilustración 19 Fórmula explicativa TF-IDF[14]

Esta técnica junto con ‘Word Count Vector’, donde cada celda representa la frecuencia de cada término en cada documento, son conocidas como técnicas de bolsas de palabras por no tener en cuenta su contexto, el orden en el que se presentan esas palabras.

## 5.2 División del dataset en el de entrenamiento y validación.

Este paso es el más sencillo de todos ya que gracias a la función solo se debe decidir la cantidad de datos que serán utilizados para el entrenamiento y cuantos para la validación. Debido al volumen de datos se destina el 20% a la validación del modelo creado.

*Referencia:* [04\_Feature\_Engineering](https://github.com/SpRdgz/TFM/tree/master/04_Feature_Engineering)

# 6. Estudio de modelos de Machine Learning

Una vez analizado el comportamiento de las técnicas presentes en el mercado se decide investigar sobre la construcción de modelos clásicos de machine learning en vez de modelos de Deep learning como las redes LSTM porque es necesario una mayor cantidad de datos

Se han realizado diferentes pruebas con distintos parámetros de entrada que se explicarán a continuación:

## 6.1 Multinomial Naive Bayes

También llamado clasificador de Bayes ingenuo, basado en el teorema de Bayes presupone que dada la independencia de las características la existencia o la ausencia de una de ellas no está asociada con la presencia o la ausencia de otra [15]. Es un clasificador probabilístico donde se calcula la probabilidad de cada categoría utilizando el teorema de Bayes y el resultado será la de mayor probabilidad.

Ha sido calculada mediante la librería sklearn y no es necesaria la elección de ningún parámetro más que unos buenos datos de entrada. En este caso hemos introducido para el entrenamiento los datos tokenizados con la extracción del lemma, procesados posteriormente con la representación de palabras TF-IDF y los resultados han sido:

precision recall f1-score support

0 0.84 0.90 0.87 71

1 0.89 0.83 0.86 70

micro avg 0.87 0.87 0.87 141

macro avg 0.87 0.86 0.87 141

weighted avg 0.87 0.87 0.87 141

Accuracy Score 86.52482269503547

## 6.2 SVM. Support Vector Machine

Utilizado principalmente en problemas donde el objetivo es la clasificación mediante un hiperplano que sirve de elemento diferenciador entre las clases. En caso de que la división no sea lineal se podrían modificar las coordenadas espaciales a través de la transformación del kernel.

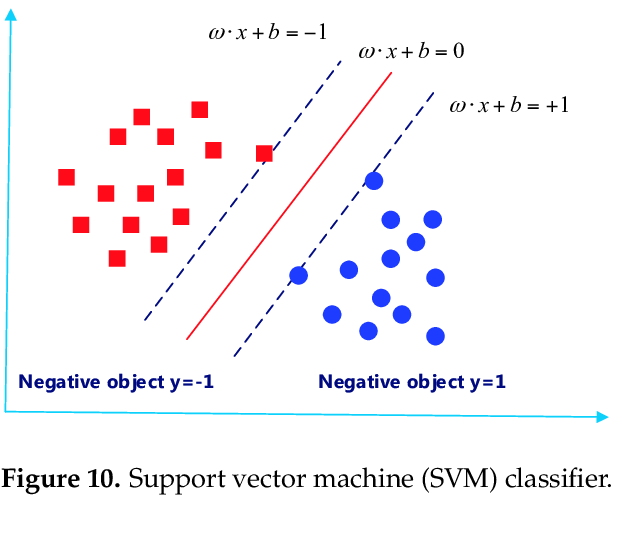


Ilustración 20 Clasificador SVM[15]

El entrenamiento ha sido realizado con los datos de entrada procesados de la misma forma que se hizo con el Multinomial Naive Bayes. Obteniendo los resultados siguientes:

precision recall f1-score support

0 0.86 0.94 0.90 71

1 0.94 0.84 0.89 70

micro avg 0.89 0.89 0.89 141

macro avg 0.90 0.89 0.89 141

weighted avg 0.90 0.89 0.89 141

con una precision del 89..36

# 7. Conclusiones

#### **Will Sentiment Analysis ever be 100% accurate, or close?**

Probably not, but that is not meant to be a bad thing. This will not be because people aren't smart enough to eventually make computers that really understand language. Instead, this is really just plain impossible, seeing as how it's rarely the case that 80% of people agree on the sentiment of text[16].

¿El Análisis de Sentimiento alguna vez será 100% exacto o cercano?

Probablemente no, pero eso no pretende ser algo malo. Esto no será porque las personas no son lo suficientemente inteligentes como para hacer computadoras que realmente entiendan el lenguaje. En cambio, esto es realmente imposible, ya que rara vez el 80% de las personas está de acuerdo con el sentimiento del texto [16].

Se concluye que los resultados han sido buenos ya que superan un valor del 89% de exactitud, con una precisión y un recall cercanos a 1 sin llegar a sobrepasar los valores del 100% que podrían indicar un overfitting.

Este proyecto no solo ha servido para un aprendizaje en profundidad sobre cada una de las técnicas de procesamiento de lenguaje natural sino que también nos ha mostrado la necesidad de desarrollar modelos a partir del propio lenguaje debido a las características únicas que tienen incluso los dialectos. Esto incluye desde el desarrollo de las librerías de stemming y lemmanization sino también los modelos de clasificación de palabras,

Hemos concluido que aunque las traducciones no sean de utilidad si se siguen los mismos pasos se puede llegar a importantes conclusiones como ha sido el recorrer los pasos utilizados por el equipo de Vader al utilizar técnicas similares en cuanto a la limpieza y clasificación de las características de los datos.

En cuanto a la planificación, es necesario no contar como productivo cada uno de los días por si surgen imprevistos en las planificaciones o en los estudios y se deben descartar vías. En este caso el Diagrama de Gannt inicial tenía muchas menos tareas que el que se ha presentado al finalizar.

En un futuro se quiere seguir ampliando la categorización de las palabras, revisar la librería de emoticonos, insertar listas de palabras con connotación negativa y positiva, incluir expresiones con intencionalidad tóxica como ‘me cago en tú’, listas de palabras que ensalzan un sentimiento como puede ser el diferenciar la palabra grande + adj o grandísima + adjetivo y por último ampliar el dataset de modo que se puedan implementar redes neuronales.

# 4. Glosario

* **NLP:** Natural Language Processing, campo de las ciencias de la computación, inteligencia artificial y lingüística que estudia las interacciones entre las computadoras y el lenguaje humano.
* **Corpus:** Recopilación de datos que se utiliza para una investigación.
* **Léxico:** Las palabras contenidas dentro de un corpus (a menudo se hace referencia a ellas como "tokenizadas").
* **Dataset:** Conjunto de datos que sirven para analizar una problemática.
* **Machine Learning**: Método de análisis de datos que automatiza la construcción de modelos analíticos. Es una rama de la inteligencia artificial basada en la idea de que los sistemas pueden aprender de datos, identificar patrones y tomar decisiones con mínima intervención humana [17].
* **Diccionario de sentimientos**: Conjunto de palabras (posiblemente incluyendo tallos de palabras) con calificaciones.
* **Label:** Etiqueta que se pone a una parte de los datos

# 5. Bibliografía

«Jigsaw Unintended Bias in Toxicity Classification». [En línea]. Disponible en: https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification [Accedido: 10-abr-2019].

[1] «Lista: Insultos en castellano que deberías conocer y su significado». [En línea]. Disponible en: https://listas.20minutos.es/lista/insultos-en-castellano-que-deberias-conocer-y-su-significado-393340/. [Accedido: 10-abr-2019].

[2] D. Davidov, O. Tsur, y A. Rappoport, «Enhanced Sentiment Learning Using Twitter Hashtags and Smileys», 2010.

[3] Y. Gong, C. Poellabauer, y K. Shin, «Improving LIWC Using Soft Word Matching Speech and Language Processing for Health View project Improving LIWC Using Soft Word Matching», 2018.

[4] C. J. Hutto y E. Gilbert, «VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text», 2014.

[5] «Changing the way business decisions are made». [En línea]. Disponible en: https://blog.twitter.com/official/en\_us/a/ibm.html. [Accedido: 26-may-2019].

[6] Prasad Patil, «What is Exploratory Data Analysis? – Towards Data Science». [En línea]. Disponible en: https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15. [Accedido: 28-may-2019].

[7] «The Times is Partnering with Jigsaw to Expand Comment Capabilities», *New York Times Co.*, 2016.

[8] N. V. Chawla, «Data Mining for Imbalanced Datasets: An Overview», *Data Mining and Knowledge Discovery Handbook*, 2009. [En línea]. Disponible en: http://link.springer.com/10.1007/978-0-387-09823-4\_45. [Accedido: 29-may-2019].

[9] «Emoji Sentiment Ranking v1.0». [En línea]. Disponible en: http://kt.ijs.si/data/Emoji\_sentiment\_ranking/. [Accedido: 05-jun-2019].

[10] Ruslana Dalinina, «Introduction to Correlation», *Oracle + DataScience.com*, 2017.

[11] A. Ng, «Machine Learning and AI via Brain simulations».

[12] «Why is removing stop words not always a good idea - Wilame Lima Vallantin - Medium». [En línea]. Disponible en: https://medium.com/@wilamelima/why-is-removing-stop-words-not-always-a-good-idea-c8d35bd77214. [Accedido: 25-jun-2019].

[13] «Python Programming Tutorials». [En línea]. Disponible en: https://pythonprogramming.net/python-pickle-module-save-objects-serialization/. [Accedido: 27-jun-2019].

[14] «Técnicas avanzadas de recuperación de información: Frecuencias y pesos de los términos de un documento». [En línea]. Disponible en: http://ccdoc-tecnicasrecuperacioninformacion.blogspot.com/2012/11/frecuencias-y-pesos-de-los-terminos-de.html. [Accedido: 25-jun-2019].

[15] T. de: y T. de:, *Naive Bayes classifier*. .

[16] «Sentdex Analysis». [En línea]. Disponible en: http://sentdex.com/sentiment-analysis/. [Accedido: 28-jun-2019].

[17] «Aprendizaje automático: Qué es y por qué es importante | SAS». [En línea]. Disponible en: https://www.sas.com/es\_es/insights/analytics/machine-learning.html. [Accedido: 25-jun-2019].

# 6. Anexos

Todos los anexos pueden ser accedidos a través del repositorio de Github: <https://github.com/SpRdgz/TFM/blob/master/>