

Guillermo Machín

ROBOTODIO



Índice

1. Introducción	2
2. Descripción de los datos.....	3
2.1. Dataset de tweets.....	4
2.2. Dataset de usuario	5
3. Metodología	6
3.1. Extracción	6
3.2. Limpieza y transformación.....	6
3.2.1. Limpieza.....	6
3.2.2. Traducción.....	6
3.2.3. Modelo Detoxify	7
3.3 Explotación	9
3.3.1. Datasets finales:.....	9
3.3.2. Visualización de los datos:.....	10
4. Frontend	12
4.1. Stack tecnológico	12
4.2. Manual.....	12
5. Bibliografía	13

1. Introducción

Es un hecho que en la actualidad, las redes sociales forman parte de nuestro día a día. Nos permiten mantener el contacto con amigos y familiares; o comunicarnos e interactuar con celebridades, marcas y personas que no son tan cercanas a nuestro círculo. Las redes sociales promueven el acercamiento y la relación entre las personas, aunque como siempre, también pueden provocar discusiones y desencadenar ataques e insultos entre los usuarios de estas aplicaciones.

Este es un grave problema al que se tienen que enfrentar las empresas, y que con el crecimiento constante de usuarios en estas aplicaciones y la aparición de nuevas redes sociales, es necesario abordar. Pero... ¿cómo?, ¿dónde está y quién establece el límite?, ¿cómo se monitoriza?

Está demostrado que con la cantidad de mensajes que se generan a diario en las redes sociales, es prácticamente imposible comprobar de forma manual cuáles cumplen con las normas comunitarias y cuáles no. Adicionalmente, existen estudios que muestran que aquellas personas cuyo trabajo es monitorizar y revisar mensajes agresivos u de odio en estas plataformas, desarrollan una probabilidad mayor de sufrir depresión o situaciones de estrés que acaben deteriorando su salud, generando además costes adicionales para las empresas.

Una de las soluciones es desarrollar sistemas y algoritmos que nos permitan identificar con precisión estos mensajes de odio, o hacer un cribado inicial y dejar los más ambiguos a la revisión del ojo humano, facilitando el trabajo de las personas y la experiencia del usuario dentro de estas redes sociales. Con este objetivo nació el proyecto Robotodio, identificar el discurso de odio en Twitter, mejorar las redes sociales y concienciar a la población de la polaridad de sus mensajes.

El alcance del proyecto es una aplicación que dada una cuenta de Twitter, devuelva una puntuación del nivel de odio de ese usuario en base al análisis de los últimos tweets. La aplicación es desarrollada totalmente en Python, y alojada en un servidor de la comunidad Streamlit, de modo que la interacción del usuario con el producto sea a través de un frontend de tipo página web.

El algoritmo utilizado en este proyecto ha sido extraído de GitHub y no ha sido objeto de desarrollo de este proyecto, solo su aplicación, análisis de resultados y desarrollo frontend. El modelo se llama Detoxify^[1], y ha sido entrenado con los dataset de las competiciones de Kaggle Jigsaw. En el apartado de metodología se entra más en detalles del proceso subyacente.

2. Descripción de los datos

Los datos utilizados en el proyecto Robotodio proceden exclusivamente de los tweets disponibles en aquellas cuentas de Twitter abiertas al público. Por lo que cualquier persona podría replicar estos análisis siempre y cuando la cuenta de Twitter objetivo siga siendo pública.

Los tweets se obtienen mediante la librería Tweepy, la cual permite interaccionar de forma sencilla con la API de Twitter. La API permite extraer para cada mensaje, multitud de metadatos en formato .json. A continuación se muestra un ejemplo de la información que se puede obtener por cada registro:

```
-----
{
  "created_at": "Sat May 01 13:01:11 +0000 2021",
  "id": 1388478608457539588,
  "id_str": "1388478608457539588",
  "full_text": "Dos escoltas de Iglesias son detenidos por agredir a los asistentes de un miti
n de vox \nY aqu\u00ed no pasa nada? \n\nVaya pa\u00eds de pandereta.",
  "truncated": false,
  "display_text_range": [
    0,
    134
  ],
  "entities": {
    "hashtags": [],
    "symbols": [],
    "user_mentions": [],
    "urls": []
  },
  "source": "<a href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Andr
oid</a>",
  "in_reply_to_status_id": null,
  "in_reply_to_status_id_str": null,
  "in_reply_to_user_id": null,
  "in_reply_to_user_id_str": null,
  "in_reply_to_screen_name": null,
  "user": {
    "id": 908629539609341952,
    "id_str": "908629539609341952",
    "name": "Espa\u00f1ol74 (david) \ud83c\uddea\ud83c\uddff8",
    "screen_name": "DFD_74",
    "location": "",
    "description": "Padre orgulloso, marido enamorado, currante, espa\u00f1ol y del atleti.
Nadie es perfecto, licenciado en Historia. Afiliado a Vox",
    "url": null,
    "entities": {
      "description": {
        "urls": []
      }
    }
  },
  "protected": false,
  "followers_count": 5809,
  "friends_count": 583,
  "listed_count": 13,
  "created_at": "Fri Sep 15 09:52:18 +0000 2017",
  "favourites_count": 142959,
  "utc_offset": null,
  "time_zone": null,
  "geo_enabled": false,
  "verified": false,
  "statuses_count": 62454,
  "lang": null,
  "contributors_enabled": false,
  "is_translator": false,
  "is_translation_enabled": false,
  "profile_background_color": "F5F8FA",
  "profile_background_image_url": null,
  "profile_background_image_url_https": null,
  "profile_background_tile": false,

```

```

    "profile_image_url": "http://pbs.twimg.com/profile\_images/1267955029748649991/-kSYQFm\_normal.jpg",
    "profile_image_url_https": "https://pbs.twimg.com/profile\_images/1267955029748649991/-kSYQFm\_normal.jpg",
    "profile_banner_url": "https://pbs.twimg.com/profile\_banners/908629539609341952/1594929970",
    "profile_link_color": "1DA1F2",
    "profile_sidebar_border_color": "C0DEED",
    "profile_sidebar_fill_color": "DDEEFF",
    "profile_text_color": "333333",
    "profile_use_background_image": true,
    "has_extended_profile": true,
    "default_profile": true,
    "default_profile_image": false,
    "following": false,
    "follow_request_sent": false,
    "notifications": false,
    "translator_type": "none",
    "withheld_in_countries": []
  },
  "geo": null,
  "coordinates": null,
  "place": null,
  "contributors": null,
  "is_quote_status": false,
  "retweet_count": 5,
  "favorite_count": 16,
  "favorited": false,
  "retweeted": false,
  "lang": "es"
}

```

Dentro del proyecto de Robotodio se trabaja con dos datasets diferenciados:

- Dataset de tweets: es un dataset de tipo hechos, donde se recogen todos los tweets que serán analizados posteriormente.
- Dataset de usuario: es un dataset de tipo dimensional, donde se recogen los datos principales del usuario objetivo.

Sobre estos ficheros, se añaden campos adicionales fruto de los modelos y de diferentes cálculos que se explican a lo largo de la memoria.

2.1. Dataset de tweets

Dentro de este DataFrame se han considerado de relevancia solo algunos de los campos del anterior formato. A continuación se muestra un listado de los atributos de interés, su significado, y su tipología:

- tweet.full_text (string): texto completo del tweet.
- tweet.user.screen_name (string): nombre de la cuenta de Twitter.
- tweet.id (string): identificador único para cada tweet.
- tweet.user.followers_count (int): número de seguidores de la cuenta de Twitter.
- tweet.source (string): desde qué dispositivo se ha escrito el tweet.
- tweet.created_at (timestamp): fecha de publicación del tweet.
- tweet.lang (string): idioma del tweet.
- len(tweet.full_text) (int): longitud del tweet.
- tweet.favorite_count (int): número de likes por cada tweet.
- tweet.retweet_count (int): número de retweets por cada tweet.
- re.findall(r"#(\w+)", tweet.full_text) (list): lista de hashtags utilizados en cada tweet, se extrae mediante expresiones regulares del campo tweet.full_text.

Estos atributos se extraen para cada tweet y se colocan en un DataFrame de Pandas de modo que cada tweet constituye un registro o fila, y los atributos extraídos del .json, las columnas. El objetivo es pasar de un fichero plano de texto como puede ser un .json, a un formato tabular que permita manipular y analizar los datos de forma sencilla. A continuación se muestra un ejemplo del DataFrame final:

	tweet	account	id	followers	source	date	language	length	likes	RTs	hashtags
0	Dos escoltas de Iglesias son detenidos por agr...	DFD_74	1388478608457539588	5809	Twitter for Android	2021-05-01 13:01:11	es	134	16	5	[]
1	He salido a andar un rato \n🐼 https://t.co/B45...	DFD_74	1388456562969219073	5809	Twitter for Android	2021-05-01 11:33:35	es	52	7	1	[]
2	@ludoxigen Llegas 76 años tarde gilipolles	DFD_74	1388415018870394883	5809	Twitter for Android	2021-05-01 08:48:30	es	42	1	0	[]
3	Con los datos que se están barajando para las ...	DFD_74	1388414587087822848	5809	Twitter for Android	2021-05-01 08:46:47	es	129	6	1	[]
4	Tengo un local alquilado y le ha llegado la úl...	DFD_74	1388403492474212353	5809	Twitter for Android	2021-05-01 08:02:42	es	134	23	11	[]

2.2. Dataset de usuario

Dentro de este DataFrame se han considerado de relevancia solo algunos de los campos del anterior formato. A continuación, se muestra un listado de los atributos de interés, su significado, y su tipología:

- tweet.user.screen_name (string): cuenta de Twitter
- tweet.user.name (string): nombre de la cuenta en Twitter
- tweet.user.description (string): Bio
- tweet.user.created_at (timestamp): fecha de ingreso en Twitter
- tweet.user.friends_count (int): número de cuenta seguidas
- tweet.user.followers_count (int): número de followers
- tweet.user.statuses_count (int): número de tweets

Ejemplo del DataFrame final:

	variable	value
0	account	DFD_74
1	account_name	Español74 (david) es
2	bio_description	Padre orgulloso, marido enamorado, currante, e...
3	creation_date	2017-09-15 09:52:18
4	friends	583
5	followers	5809
6	tweets	62454

3. Metodología

El esquema seguido en el proyecto es similar al de cualquier proceso ETL (Extract, Transform and Load). En primer lugar se obtienen los datos, luego se limpian y se aplica el modelo pre-entrenado de Detoxify, y por último se procede a la explotación de los mismos desde un frontend.

Como se podrá ver en el directorio notebooks, existen tres cuadernos diferentes. Sobre estos cuadernos se apoya la memoria, tratando de explicar de un modo más teórico, lo recogido de forma más práctica en los notebooks. Cada fichero contiene:

- 01-expanded-model.ipynb: modelo expandido, necesita como input los tweets en inglés, devuelve más datos por cada mensaje.
- 02-multilingual-model.ipynb: modelo multilingüe, admite como input mensaje en varios idiomas, devuelve un único indicador por mensaje.
- 03-performance-test.ipynb: prueba de rendimiento del modelo multilingüe para ver cuánto tiempo tarda en ejecutar el modelo completo, en función del número de tweets introducido como input.

3.1. Extracción

La extracción de los datos se realiza mediante la librería de Python Tweepy, la cual permite interactuar de forma sencilla con la API de Twitter. Como los datos vienen en formato .json, se ha utilizado también la librería re para parsear el mensaje.

A la hora de la extracción se ha definido una función que descarga exclusivamente aquellos mensajes escritos por el usuario objetivo, obviando para el análisis los retweets (de aquí en adelante RT's). Esta función toma como argumento el número de tweets a analizar y la cuenta objetivo.

3.2. Limpieza y transformación

3.2.1. Limpieza

Sobre el dataset resultante de la extracción se ha limpiado el campo tweet (véase 2.1. Dataset de tweets) para eliminar caracteres como saltos de línea, dobles espacios, retornos de carro...etc.

Esto permite obtener un mensaje más claro y legible, así como evitar errores en fases posteriores como la predicción del nivel de odio o la traducción en el modelo expandido.

3.2.2. Traducción

El modelo entrenado Detoxify puede trabajar tanto en formato multilingüe, como en formato expandido.

Para el modelo expandido, el cual es capaz de devolver un mayor número de datos, es necesario tener como input del proceso los datos en inglés. Para ello se ha utilizado la librería TextBlob, la cual se comunica con la API de Google Traductor para realizar la traducción. Este proceso se encuentra exclusivamente en el notebook *01-expanded-model.ipynb*.

Aunque esta librería funciona muy bien, tiene la limitación que al pasar por la API de Google tiene un uso limitado de tweets por día, por lo que sirve para hacer análisis muy puntuales. Si se supera el límite, la IP utilizada para hacer las request a la API se bloquea durante un periodo de 24 horas.

3.2.3. Modelo Detoxify

El modelo Detoxify es un algoritmo entrenado por Unitary (<https://www.unitary.ai/>), empresa londinense dedicada al desarrollo de modelos de visión por ordenador para analizar el contenido en internet.

Entre sus desarrollos, han liberado Detoxify, una librería que permite obtener un scoring de odio en base a un mensaje de texto plano corto tipo Tweet.

Entrenamiento:

La librería Detoxify surgió como resultado de las competiciones de Kaggle Jigsaw. Estas competiciones fueron propuestas por la empresa Alphabet al comprobar que su API Alphabet's Perspective, desarrollada como solución de IA para detectar mensajes tóxicos en la red, devolvía mensajes sesgados.

Esto los llevó a crear 3 desafíos de Kaggle con el objetivo de construir mejores modelos de toxicidad:

- Toxic Comment Classification Challenge: el objetivo de este desafío era construir un modelo que pueda detectar diferentes tipos de toxicidad como amenazas, obscenidad, insultos u ataques identitarios en los comentarios de Wikipedia.
<https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge>
- Jigsaw Unintended Bias in Toxicity Classification: el segundo desafío trató de abordar el sesgo no intencionado observado en el desafío anterior mediante la introducción de etiquetas de identidad y una métrica de sesgo especial destinada a minimizar este sesgo en el dataset Civil Comments.
<https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification>
- Jigsaw Multilingual Toxic Comment Classification: el tercer desafío combinó datos de los dos desafíos anteriores y alentó a los desarrolladores a encontrar una manera efectiva de construir un modelo multilingüe a partir de datos de capacitación en inglés únicamente.
<https://www.kaggle.com/c/jigsaw-multilingual-toxic-comment-classification>

Fruto de estos tres concursos, dentro de Detoxify podemos encontrar tres modelos pre-entrenados:

Model name	Transformer type	Data from
original	bert-base-uncased	Toxic Comment Classification Challenge
unbiased	roberta-base	Unintended Bias in Toxicity Classification
multilingual	xlm-roberta-base	Multilingual Toxic Comment Classification

Original-model:

Basado en BERT. Introducido originalmente en 2018 por Google AI, BERT es un transformer bidireccional profundo previamente entrenado con texto sin etiquetar de Internet, que presenta resultados de vanguardia en una variedad de tareas de PLN, como la respuesta a preguntas y la inferencia de lenguaje natural. El enfoque bidireccional resultó en una comprensión más profunda del contexto en comparación con los enfoques unidireccionales anteriores (de izquierda a derecha o de derecha a izquierda).

Este modelo devuelve por cada tweet una puntuación para las siguientes etiquetas:

- toxic
- severe_toxic
- obscene
- threat
- insult
- identity_hate

Unbiased-model:

Basado en RoBERTa. Desarrollado por Facebook AI en julio de 2019, RoBERTa es una forma optimizada de pre-entrenar BERT. Facebook encontró que modificando algunos de los hiperparámetros de BERT, y con un mayor número de mini-batches, con tasas de aprendizaje mayores y con una cantidad de datos un orden de magnitud mayores, el rendimiento del modelo sufrió una mejora notable.

Este modelo devuelve por cada tweet una puntuación para las siguientes etiquetas:

- toxicity
- severe_toxicity
- obscene
- threat
- insult
- identity_attack
- sexual_explicit

Multilingual-model:

Basado en XLM-Roberta. Desarrollado a finales de 2019 por Facebook AI, XLM-Roberta es un modelo multilingüe construido sobre RoBERTa y pre-entrenado en 2.5TB de datos de Common Crawl. Aunque se entrenó en 100 idiomas diferentes, se consiguió no sacrificar el rendimiento por idioma, y ser competitivo en modelos monolingües.

El modelo multilingüe permite introducir como input mensajes en siete idiomas diferentes: inglés, francés, español, italiano portugués, turco y ruso.

Este modelo devuelve por cada tweet una puntuación para las siguientes etiquetas:

- toxicity

Aplicación:

Para aplicar el modelo basta con instalar la librería Detoxify, y utilizar una línea de código por cada modelo elegido.

```
>>> from detoxify import Detoxify
>>> Detoxify('original').predict("Shut up you idiot!")
>>> Detoxify('unbiased').predict("Shut up you idiot!")
>>> Detoxify('multilingual').predict("Shut up you idiot!")
```

El algoritmo devuelve para cada tweet un indicador de posible toxicidad:

```
>>> Detoxify('multilingual').predict('Desde Luego no se puede ser más imbécil')
>>> {'toxicity': 0.6107634}
```

De acuerdo con la bibliografía de la librería, un mensaje se puede considerar tóxico a partir de el umbral de 0.5. También se indica que este valor se debe tomar como indicador de potencialidad tóxica, ya que existen pequeños sesgos que podrían modificar esta puntuación distorsionando el resultado.

De forma adicional, se pueden inferir mensajes tóxicos relacionados con ataques particulares de tipo homófobo, racista...etc, utilizando lexicones específicos. Un lexicón no es más que una serie ordenada de palabras de una lengua, una persona, una región, una materia o una época determinadas. Por lo que elaborando una lista de palabras de una temática concreta, y buscando éstas en mensajes de elevada toxicidad, se pueden inferir mensajes de odio centrados en un tema de interés.

En el caso de uso de este proyecto, se ha utilizado para detectar aparte de mensajes de odio, mensajes enfocados en el racismo. Para ello se puede comprobar el lexicón ubicado en la carpeta `/project_robotodio/words_list/racist_words.txt`.

3.3 Explotación

Como se comentaba en puntos anteriores, existen dos notebooks, uno para aplicar el modelo original o extendido, y otro para el modelo multilingüe. Como para evitar las limitaciones del sistema de traducción de TextBlob se ha productivizado el modelo multilingüe, a partir de este punto, todos los datos, capturas de pantalla...etc harán referencia al mismo.

Una vez decidido el modelo, se ha analizado para cada tweet el nivel de toxicidad, y se ha anexo al DataFrame de tweets mostrado en el punto 2.1. *Dataset de tweets*.

	tweet	account	id	followers	source	date	language	length	likes	RTs	hashtags	toxicity	class	racist
0	Dos escoltas de Iglesias son detenidos por agr...	DFD_74	1388478608457539588	5809	Twitter for Android	2021-05-01 13:01:11	es	134	16	5	[]	0.153749	non-toxic	non-racist
1	He salido a andar un rato 🐶 https://t.co/B45tO...	DFD_74	1388456562969219073	5809	Twitter for Android	2021-05-01 11:33:35	es	52	7	1	[]	0.026182	non-toxic	non-racist
2	@ludoxigen Llegas 76 años tarde gilipolles	DFD_74	1388415018870394883	5809	Twitter for Android	2021-05-01 08:48:30	es	42	1	0	[]	0.144823	non-toxic	non-racist
3	Con los datos que se están barajando para las ...	DFD_74	1388414587087822848	5809	Twitter for Android	2021-05-01 08:46:47	es	129	6	1	[]	0.016933	non-toxic	non-racist
4	Tengo un local alquilado y le ha llegado la úl...	DFD_74	1388403492474212353	5809	Twitter for Android	2021-05-01 08:02:42	es	134	23	11	[]	0.019907	non-toxic	non-racist

Por último y de cara a la explotación de los datos podemos diferenciar dos apartados dentro de la explotación:

3.3.1. Datasets finales:

Top5:

De cara a mostrar en el frontend una muestra de los datos, se ha elaborado un dataset con el top 5 de tweets con mayor nivel de toxicidad.

	account	date	tweet	likes	RTs	toxicity	class	racist
54	DFD_74	2021-04-27 16:13:11	@DanielL83412758 Otro gilipollas. No se cansan...	1	0	0.992474	toxic	non-racist
25	DFD_74	2021-04-30 06:29:38	@Denna12_0 Son idiotas. No aprenden nada. Entr...	1	0	0.988248	toxic	non-racist
51	DFD_74	2021-04-28 04:54:22	Y estos gilipollas se creen toda esta mierda??...	3	2	0.961938	toxic	non-racist
57	DFD_74	2021-04-27 07:14:55	Que la mierda son la sexta, la ser, la primera...	69	37	0.961910	toxic	non-racist
31	DFD_74	2021-04-29 19:23:24	@europapress Menuda gilipollez. Siempre dando ...	3	0	0.934060	toxic	non-racist

Resumen:

En este dataset se muestran los datos principales de la cuenta objetivo, y adicionalmente, dos scorings, uno relativo al odio y otro al racismo.

El relativo al odio se calcula promediando el valor de toxicidad de cada tweet. El de racismo del mismo modo, pero solo con aquellos tweet cuyo atributo racist tiene el valor racist.

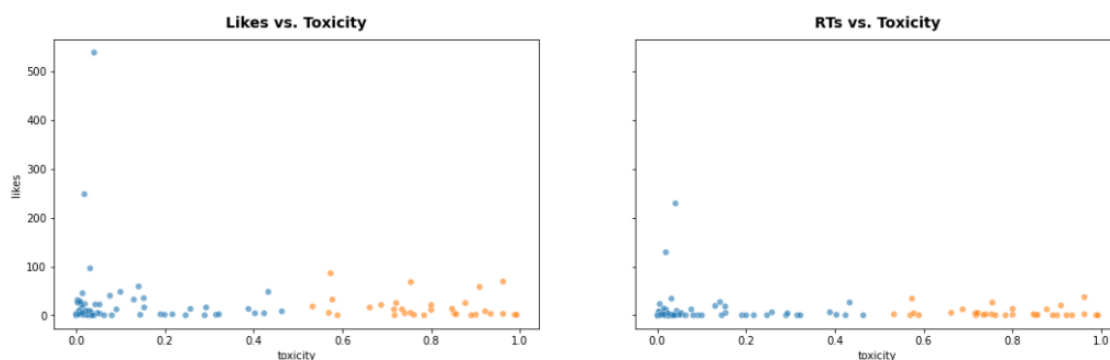
	variable	value
0	account	DFD_74
1	account_name	Español74 (david) es
2	bio_description	Padre orgulloso, marido enamorado, currante, e...
3	creation_date	2017-09-15 09:52:18
4	friends	583
5	followers	5809
6	tweets	62454
7	avg_toxicity	0.308469
8	racist_score	0.0

3.3.2. Visualización de los datos:

En este apartado se muestran los gráficos elaborados a partir de los datos expuestos en puntos anteriores. Los gráficos desarrollados son los siguientes:

Scatterplot toxicidad vs Likes y RTs:

Muestra la relación entre la toxicidad de los tweets y la repercusión que tienen.



4. Frontend

4.1. Stack tecnológico

El frontend de la aplicación de Robotodio se ha elaborado mediante la librería de Python Streamlit. Esta librería de código abierto permite crear aplicaciones web de forma fácil para mostrar resultados de análisis de datos o proyectos relacionados con la ciencia de datos.

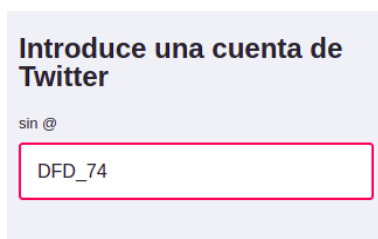
Es una librería relativamente nueva, muy sencilla de utilizar y que se integra con otras librerías de Python como pueden ser plotly, seaborn, pandas, scikit-learn o tensorflow. Además mediante el servicio Streamlit Sharing se puede alojar la aplicación en los servidores de la comunidad y poder compartir de forma sencilla la aplicación mediante una URL.

4.2. Manual

El Frontend desarrollado para el proyecto de Robotodio consiste en una interfaz web en la que el usuario introduce una cuenta de Twitter, y la aplicación muestra los resultados de los análisis descritos en el punto 3. *Metodología*.

A continuación, se muestra una guía de cómo utilizar e interactuar con el Frontend:

- 1. Acceder al servicio web. URL:
- 2. Introducir una cuenta de Twitter en la barra lateral (sin el símbolo “@”) y pulsar Intro.



- 3. Navegar por los resultados mostrados en la aplicación.

La interfaz es muy intuitiva y tiene una curva de aprendizaje muy rápida. Basta con un introducir una cuenta de twitter en la barra lateral y explorar los datos.



5. Bibliografía

Aunque se han utilizado numerosas webs, foros y contenido de internet para sacar adelante este proyecto, me gustaría resaltar aquellos enlaces que más me han servido de ayuda:

- [1] - <https://developer.twitter.com/en/docs/twitter-api>
- [2] - <https://github.com/unitaryai/detoxify>
- [3] - <https://medium.com/unitary/how-well-can-we-detoxify-comments-online-bfffe5f716d7>
- [4] - <https://docs.streamlit.io/en/stable/>