

INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO

Bases de Datos no Relacionales

Proyecto Final

Twitter Steaming con Spark

La Banda Gangrena

Integrantes:

Carlos Octavio Ordaz Bernal - 158525

Amanda Velasco Gallardo - 154415

Paola Mejia Domenzain - 157093

José Sánchez Aguilar - 156190

Fecha de entrega del proyecto:

20 de diciembre de 2018

Índice

1. Descripción general del proyecto	3
1.1. Resumen	3
1.2. Introducción	3
1.3. Contexto	3
2. Objetivos	5
2.1. Principal	5
2.2. Secundarios	5
3. Arquitectura del sistema	5
4. Características de las herramientas nuevas utilizadas	6
4.1. Spark Streaming	6
4.2. API de Twitter	7
5. Obtención y proceso de instalación de las herramientas	7
6. Fuentes de datos	13
6.1. Obtención de acceso a los datos	13
6.2. Obtención de Datos	16
7. Proceso detallado del tratamiento de la información	17
8. Resultados	18
9. Conclusiones	23

1. Descripción general del proyecto

1.1. Resumen

Respondiendo a la necesidad de procesar y entender el flujo de datos en la red, se realizó un sistema para analizar información en tiempo real proveniente de Twitter utilizando las herramientas de Streaming de Spark. De esta forma se logró extraer las palabras más usadas de un *trending topic* y graficar su frecuencia.

1.2. Introducción

En la actualidad, los conjuntos de datos que generan las empresas cumplen con ciertas características que imposibilitan o entorpecen el análisis mediante herramientas convencionales. Ya sea por su inmenso volumen, su gran complejidad o la rapidez de generación de los datos, se ha vuelto necesario desarrollar tecnologías que puedan procesar grandes flujos de información de forma rápida y eficaz. Tal es el caso al querer realizar un análisis sobre datos provenientes de redes sociales como Twitter, donde el análisis en tiempo real es de vital importancia para entender los contenidos generados por los usuarios. Para este propósito se requiere del *streaming*, palabra que en inglés se refiere a la distribución continua de contenido en la red [2]. En este marco, se han desarrollado herramientas como Spark Streaming para permitir a los desarrolladores aprovechar estos grandes flujos continuos y en tiempo real de información.

1.3. Contexto

Twitter es una de las aplicaciones con mayor tráfico en la red, donde las noticias y opiniones se distribuyen a gran escala en milisegundos. Los *trending topics* se refieren a los temas con mayor popularidad entre los usuarios. La figura 1 muestra la forma en la que un usuario puede visualizar los trending topics en la página principal de Twitter.

Tendencias: Global · Cambiar

#FelicesPosadas

📌 Promocionado por Gasolinera Pemex

Lula

314 mil Tweets

#15TemmuzUnutma

3.961 Tweets

#TürkünHürriyetAteşi

42 mil Tweets

#KurandakiCennetHayatı

1.890 Tweets

#ARSTOT

6.758 Tweets

#KadınGücü

55,1 mil Tweets

Hellboy

22,6 mil Tweets

Bale

51,9 mil Tweets

Marco Aurélio Mello

85,9 mil Tweets

Figura 1: Trending topics a nivel mundial

Sin embargo, el usuario solo visualiza una palabra o frase popular sin conocer otros temas relacionados que le puedan ayudar a conocer el contexto del trending topic. Por ejemplo, en el caso de "Lula", trending topic en la figura 1, palabras como 'Brasil', 'ex-presidente' y 'liberación' ayudan a entender más del tema que solo la palabra 'Lula'.

No obstante, utilizando herramientas de procesamiento de flujos continuos, se puede realizar un análisis mucho más rico que solamente conocer los trending topics. Se puede realizar un conteo de palabras para entender las opiniones de los usuarios sobre los trending topics. De igual manera, se puede observar a través del tiempo el cambio en las palabras utilizadas relacionadas con un tema y, por consiguiente, entender los cambios que puedan darse en la opinión pública.

2. Objetivos

2.1. Principal

En este trabajo se busca obtener *tweets* en tiempo real mediante la API de Twitter para analizar dicha información, también en tiempo real, utilizando las herramientas de streaming de Spark.

2.2. Secundarios

Se desea extraer las palabras más usadas en un trending topic y graficar su frecuencia así como obtener el número de palabras promedio utilizadas en un trending topic.

Asimismo, se desea encontrar los idiomas predominantes por tweets y los hastags asociados con los trending topics.

3. Arquitectura del sistema

Para analizar los datos de Twitter, fue necesario construir un sistema que primero extrajera tweets y posteriormente los enviara a Spark para que éste los limpiara, ajustara y realizara el análisis. En la figura 2 se muestra un diagrama de dicho sistema.

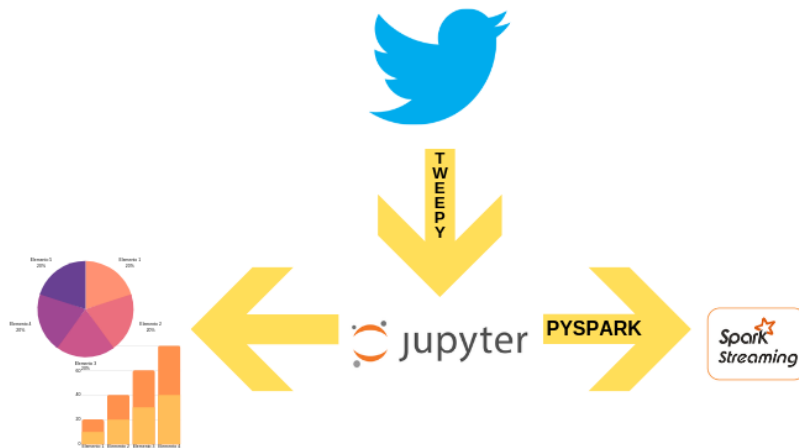


Figura 2: Arquitectura del sistema diseñado

Primero se utilizó Jupyter para conectarse a la API de Twitter y así extraer los tweets en formato JSON. La conexión entre Python y Twitter se

hizo mediante la librería `tweepy`. Los datos se enviaron a Spark para efectuar su procesamiento estableciendo una conexión gracias a la librería `pyspark`. Finalmente se utilizó la librería `Matplotlib` para elaborar las gráficas pertinentes.

4. Características de las herramientas nuevas utilizadas

4.1. Spark Streaming

Spark Streaming es una extensión al *framework* Apache Spark para el procesamiento de grandes volúmenes de datos mediante el manejo de clusters. Como se muestra en la figura 3, Spark Streaming permite efectuar funciones *MapReduce* para el procesamiento de datos entrantes de diversos tipos de *streams*. Asimismo, puede generar bases de datos y *dashboards* con la información procesada.



Figura 3: Funcionalidades de Spark Streaming

Internamente, Spark Streaming toma el stream de datos entrante y lo divide en lotes. Después cada lote es procesado individualmente para generar un stream de salida formado por los lotes.



Figura 4: Proceso interno de Spark Streaming

El stream que ingiere el sistema es una estructura llamada `DStream` o stream discretizado. Un `DStream` a su vez está conformado por `RDDs` o `Resilient Distributed Datasets`, los cuales son colecciones inmutables particionadas de objetos sobre los que se operará en paralelo para llevar a cabo las tareas MapReduce.

4.2. API de Twitter

Twitter cuenta con varias APIs para desarrolladores. En este proyecto se trabajó con la API de streaming. Ésta entrega *Tweet objects*, los cuales vienen en formato JSON e incluyen todos los atributos del tweet tales como:

- Fecha
- Texto
- Nombre de usuario
- Ubicación
- Hashtags
- Retweets

5. Obtención y proceso de instalación de las herramientas

A continuación se dará una descripción del proceso de instalación de las herramientas en Windows 10.

En primer lugar, se instaló la versión 5.3.1 de Anaconda con Python 3.7 de la siguiente liga: <https://www.anaconda.com/download/windows>

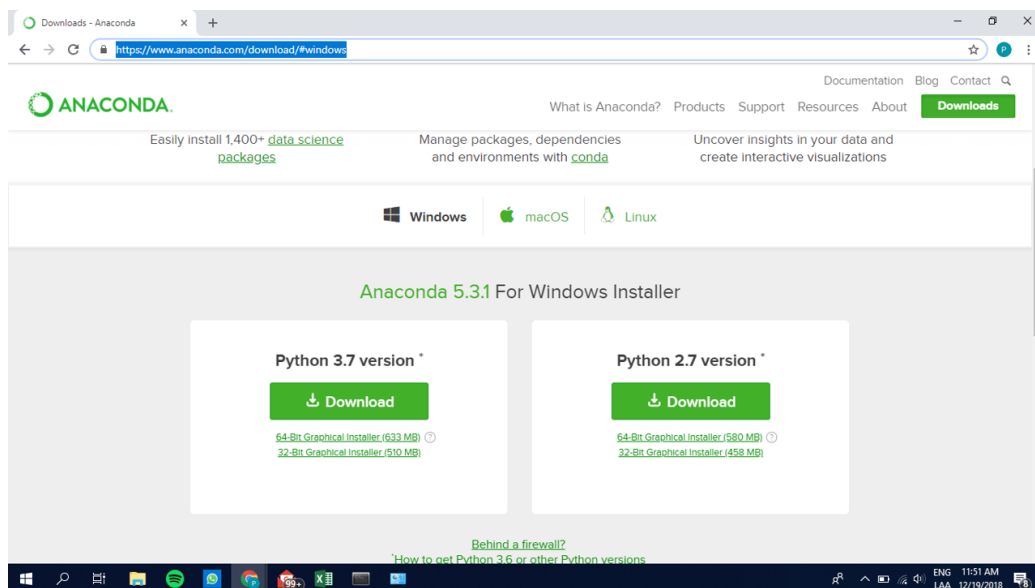


Figura 5:

Más adelante, desde la terminal de Anaconda (Anaconda Prompt) se instaló el paquete pyspark con el siguiente comando:

```
conda install -c conda-forge pyspark
```

En segundo lugar, se descargó la versión 2.3.2 con la versión de Hadoop 2.7 de la siguiente liga: <https://www.apache.org/dyn/closer.lua/spark/spark-2.3.2/spark-2.3.2-bin-hadoop2.7.tgz>

La figura 6 muestra la página para descargar spark.

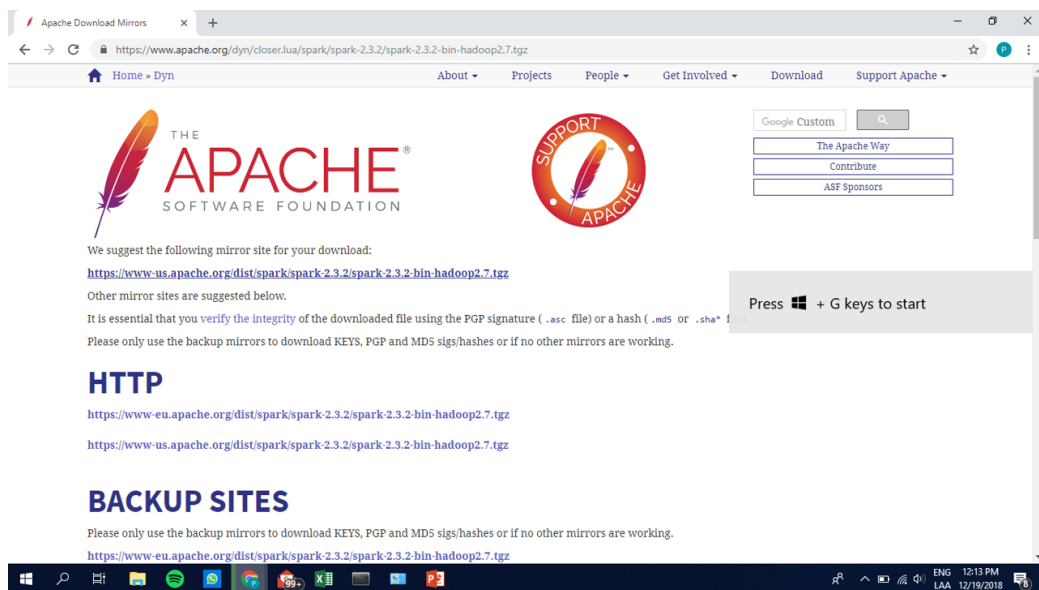


Figura 6:

La figura 7 muestra la extracción de los archivos usando 7-Zip.

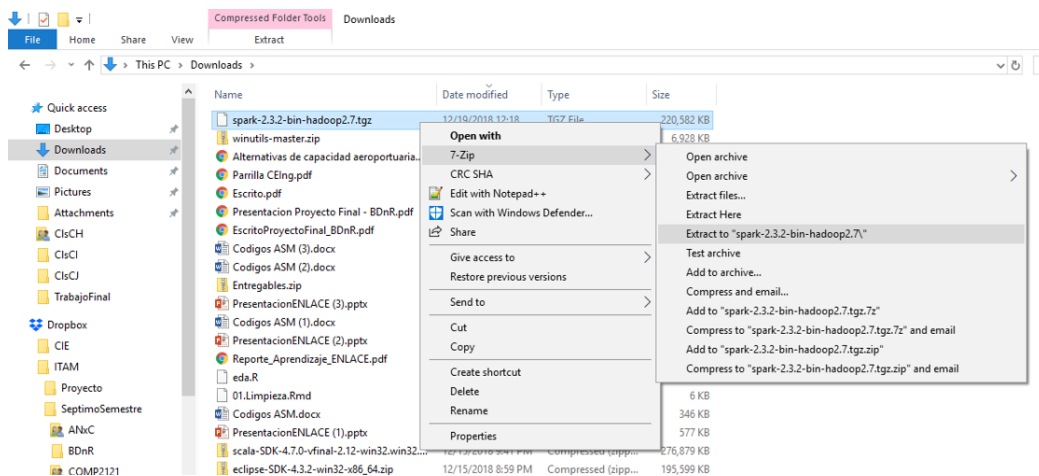


Figura 7:

Después de extraer el archivo .tgz queda un archivo .tar. El archivo .tar también se extrae usando 7-Zip. La figura 8 muestra esto.

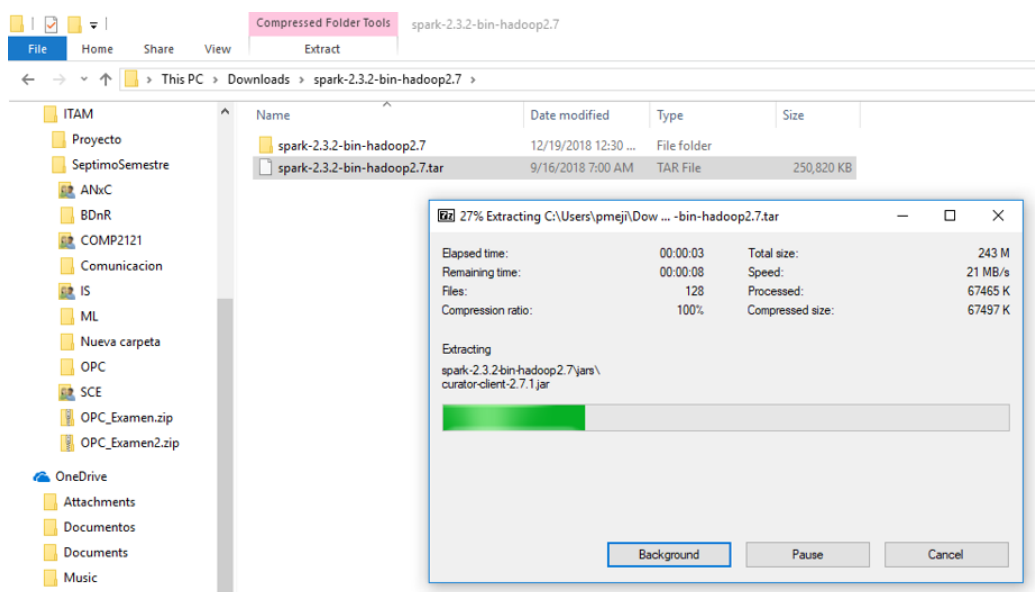


Figura 8:

Finalmente, se movieron los archivos a una nueva parte en C: con el nombre "spark" como se ve en la figura 9.

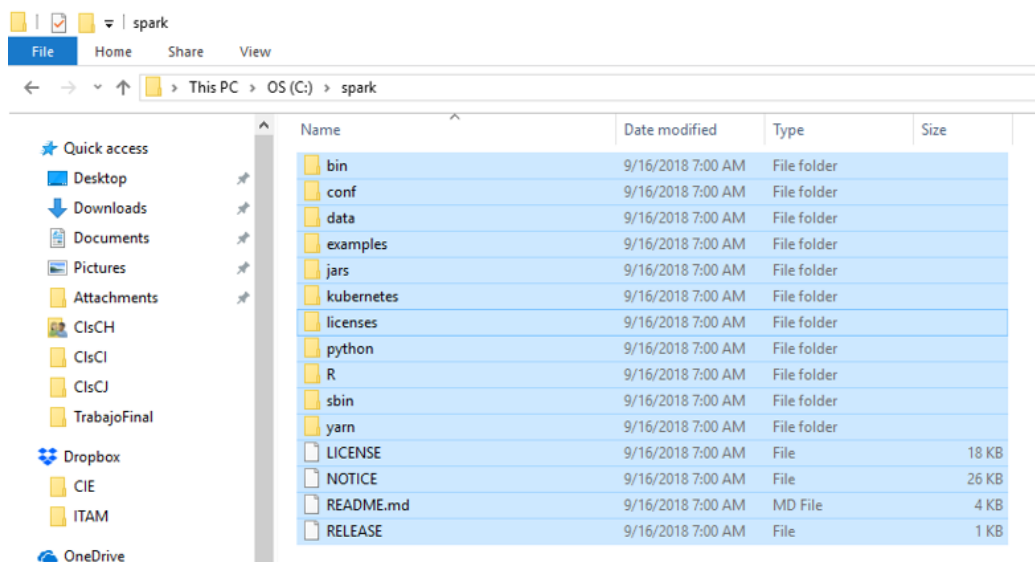


Figura 9:

Es importante acceder al Navegador de Anaconda a los entornos y comprobar que la versión de pyspark corresponda a la versión de spark instalada.

En caso de no ser la misma, al dar click derecho sobre el nombre 'pyspark' en el entorno, se puede escoger la versión adecuada como se muestra en la figura 10

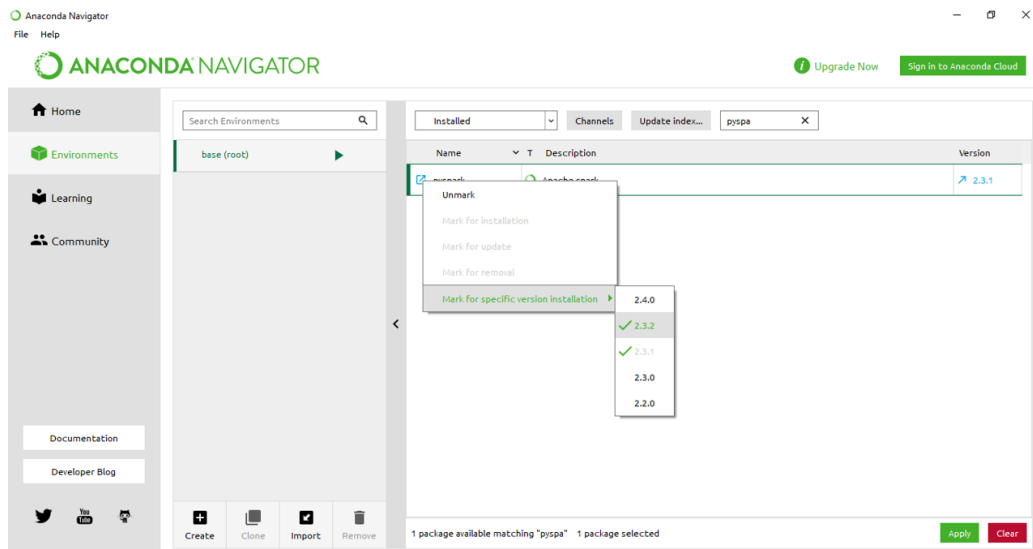


Figura 10:

En tercer lugar, se descargaron los winutils para la version 2.7 de Hadoop del el siguiente repositorio: <https://github.com/steveloughran/winutils>

La figura 11 muestra la página para descargar los winutils.

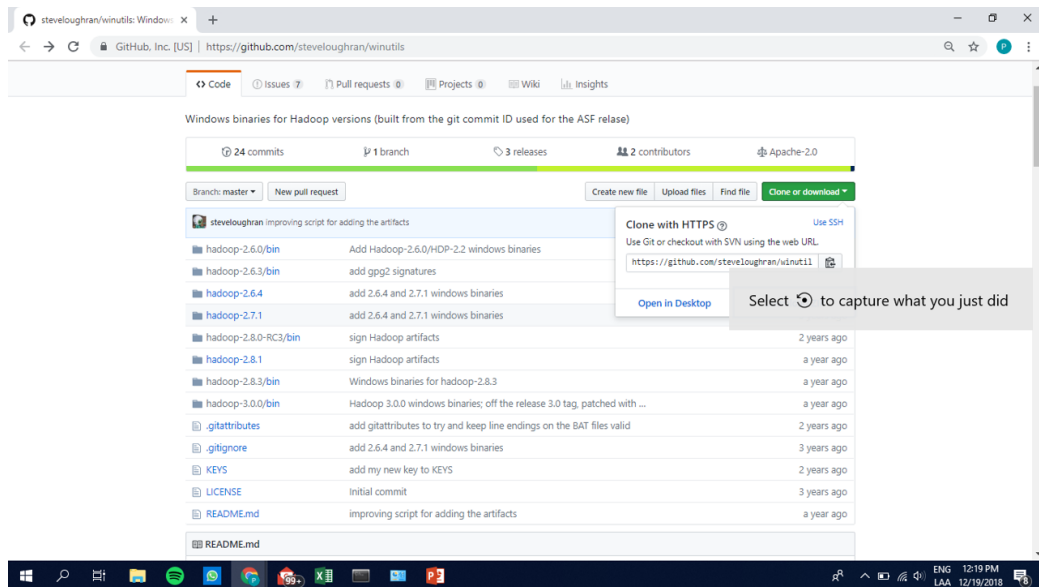


Figura 11:

Una vez descargados, se creo una carpeta en C: con el nombre de hadoop para guardar los winutils. Esto se puede ver el la figura 12

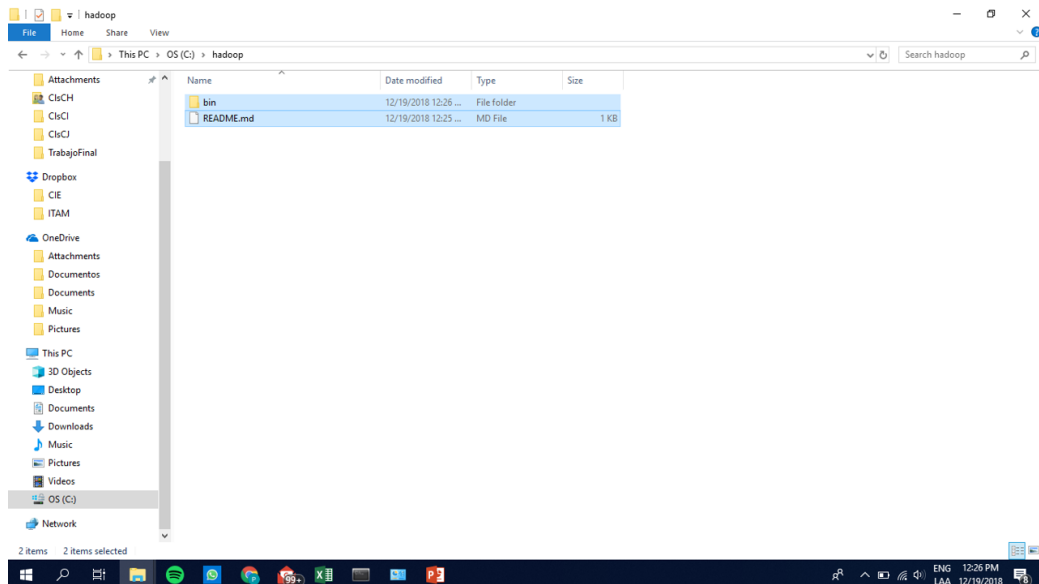


Figura 12:

En cuarto lugar, se crearon variables de entorno dentro del panel de control. Para crear las variables se ingreso al panel de control, después sistema

y seguridad, propiedades avanzadas y variables de entorno como se ve en la figura 13

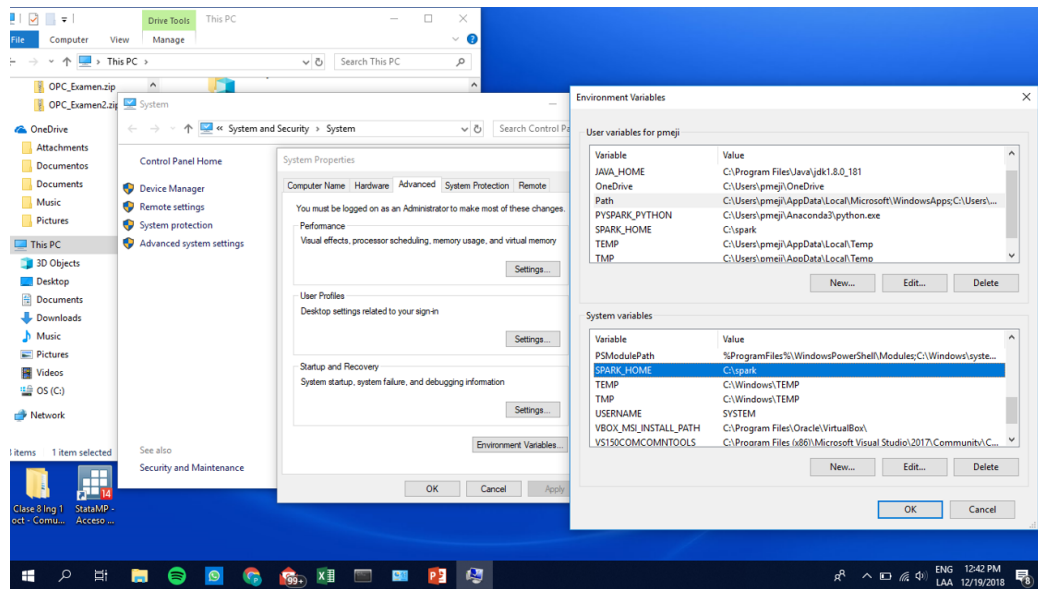


Figura 13:

En quinto lugar se consiguieron las llaves para acceder a los datos de Twitter. En la sección de "Fuentes de datos" trata esto a detalle.

6. Fuentes de datos

6.1. Obtención de acceso a los datos

Los datos utilizados pertenecen a Twitter. Para obtener acceso a ellos primero se creó una cuenta de Twitter y a continuación se solicitó una cuenta de desarrollador en la siguiente liga:

<https://developer.twitter.com/en/apply-for-access.html>

La figura 14 muestra el portal para solicitar la cuenta.

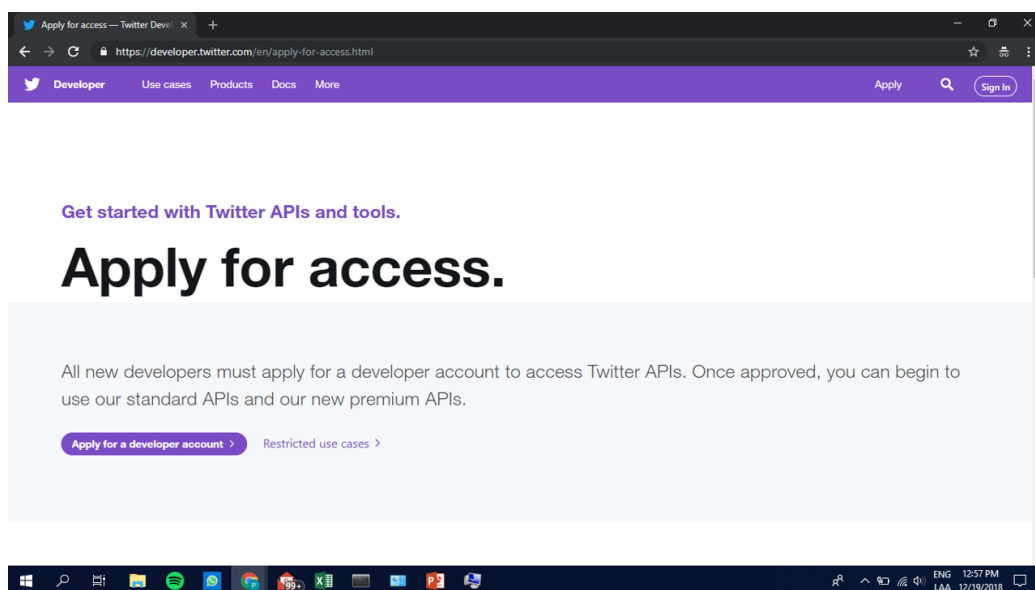


Figura 14:

Más adelante, se respondieron varios correos electrónicos con información relacionada al proyecto como se ve en la figura 15.

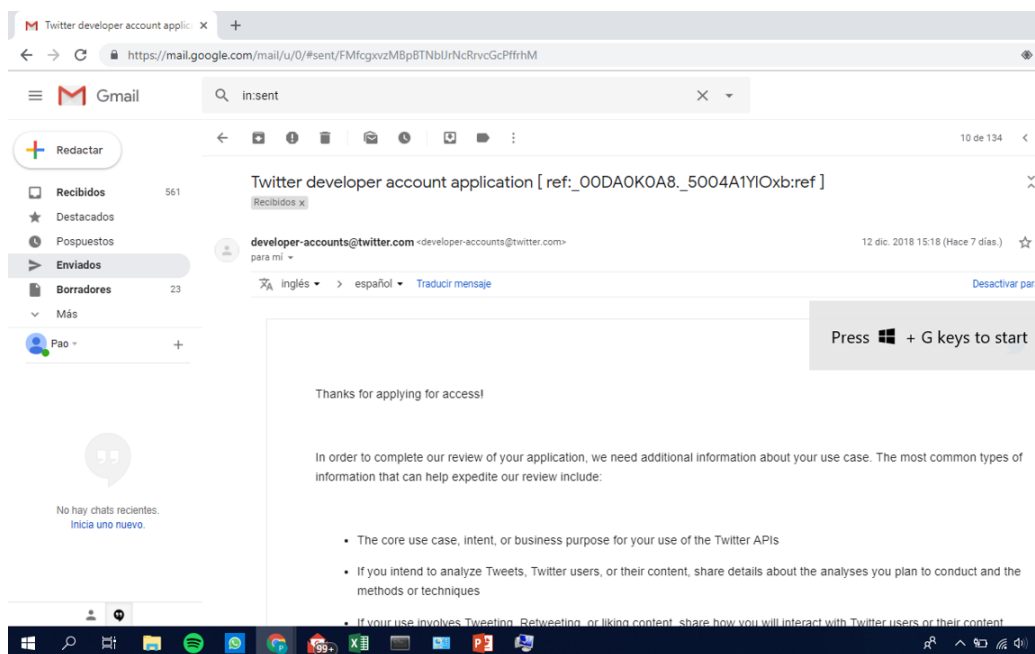
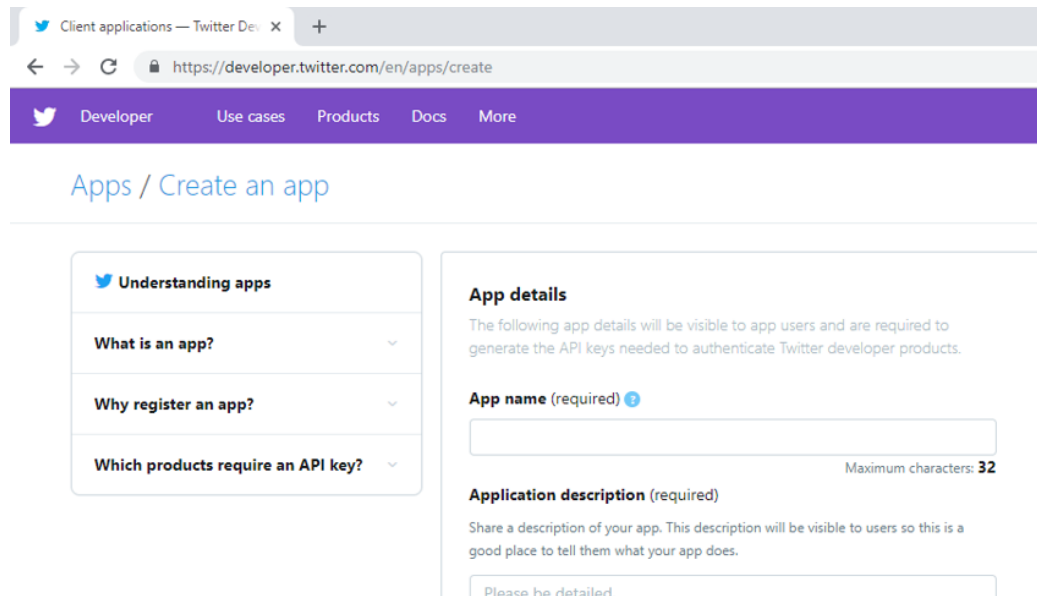


Figura 15:

A continuación, una vez que la cuenta de desarrollador fue aceptada, se creó una aplicación para el proyecto como se ve en la figura 16.



The screenshot shows the 'Create an app' page on the Twitter Developer portal. The browser address bar shows the URL <https://developer.twitter.com/en/apps/create>. The page has a purple header with navigation links: Developer, Use cases, Products, Docs, and More. Below the header, the breadcrumb 'Apps / Create an app' is visible. On the left, a sidebar titled 'Understanding apps' contains three expandable sections: 'What is an app?', 'Why register an app?', and 'Which products require an API key?'. The main content area is titled 'App details' and includes a note: 'The following app details will be visible to app users and are required to generate the API keys needed to authenticate Twitter developer products.' It features two required fields: 'App name (required)' with a text input box and a character limit of 32, and 'Application description (required)' with a text area and a note to 'Please be detailed.'

Figura 16:

La imagen 17 muestra las claves generadas para utilizar la aplicación.

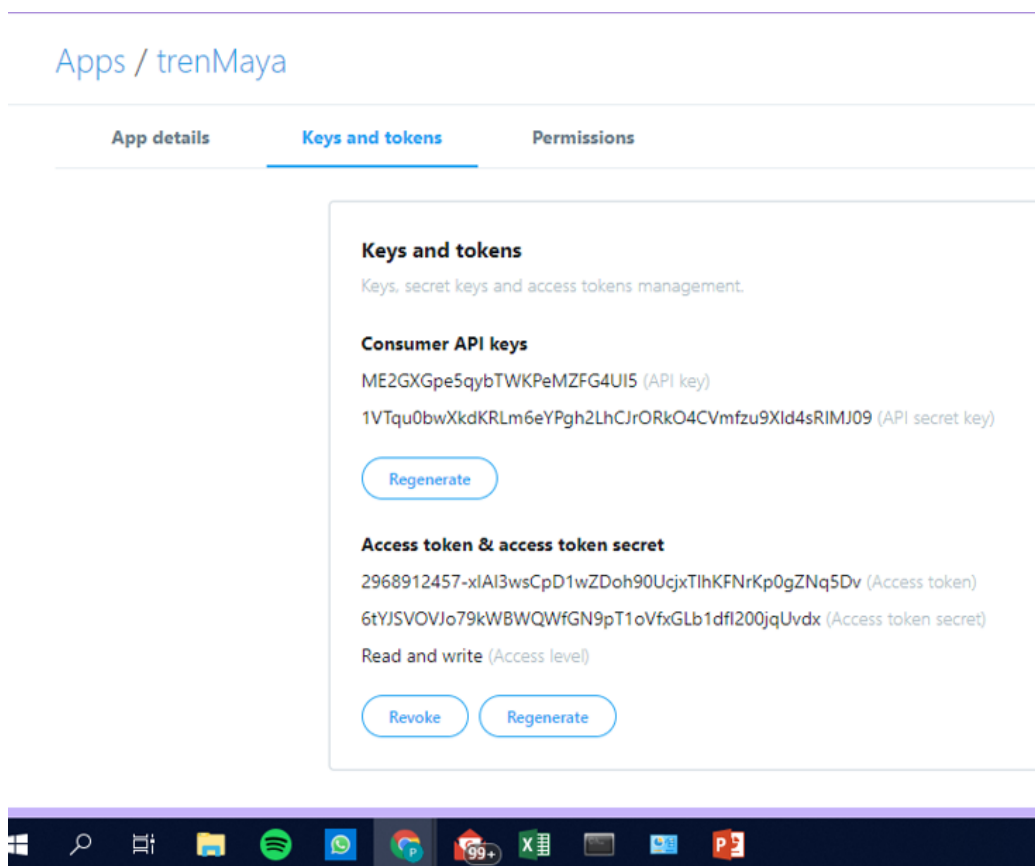


Figura 17:

6.2. Obtención de Datos

Utilizando las librerías `tweepy`, `socket` y `json` de python se creó un socket para conectarse al puerto 4040. El siguiente fragmento de código muestra esta funcionalidad:

```
s = socket.socket()
host = "localhost"
port = 4040
s.bind((host, port))
```

A continuación, se conectó a Twitter utilizando las claves de la figura 17. Utilizando la función `Stream` se solicitaron los datos que finalmente fueron filtrados con el método `filter` de `Stream`. El siguiente fragmento de código muestra esta funcionalidad para obtener tweets relacionados con el Tren Maya.


```

auth = OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_secret)

twitter_stream = Stream(auth, TweetsListener(c_socket))
twitter_stream.filter(track=['Tren Maya'])

```

7. Proceso detallado del tratamiento de la información

Utilizando la librería de Python pyspark se creó la conexión a Spark con el método "SparkContext". Esa conexión se utilizó para crear en punto de acceso a la funcionalidad de streaming de Spark, utilizando el método StreamingContext, y un punto de acceso a la base de datos, utilizando SQLContext.

A continuación, utilizando funciones lambda, se realizó un método map reduce para contar el número de ocurrencias de cada palabra.

El siguiente fragmento de código muestra esta funcionalidad:

```

( lines.flatMap( lambda text: text.split( " " ) )
  .filter( lambda word: word.lower().startswith("#") )
  .map( lambda word: ( word.lower(), 1 ) )
  .reduceByKey( lambda a, b: a + b )
  .map( lambda rec: Tweet( rec[0], rec[1] ) )
  .foreachRDD( lambda rdd: rdd.toDF().sort( desc("count") )
    .limit(10).registerTempTable("tweets") ) )

```

Asimismo, en la consola de Spark (localhost:4040) es posible visualizar la división de trabajos, los trabajos activos y los trabajos completados junto con una descripción y estadísticas de duración y etapas. La figura 18 muestra la consola de Spark.

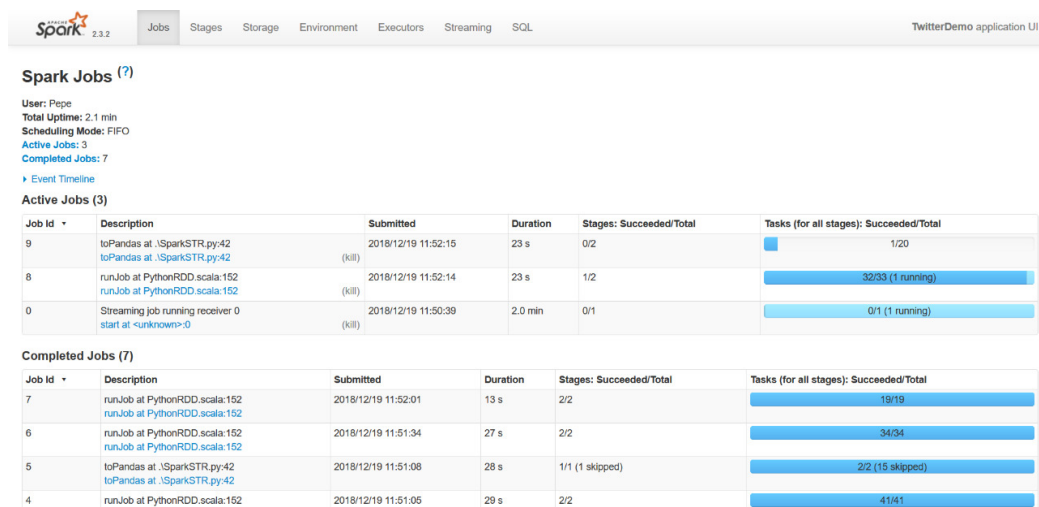


Figura 18:

8. Resultados

Se analizó el Trending Topic del presidente de Estados Unidos, Trump. La figura 19 muestra el número de tweets por idioma. El idioma que predomina es el inglés, seguido por un idioma indefinido que incluye los tweets sin texto, sólo con imágenes y finalmente los tweets en turco (tr).

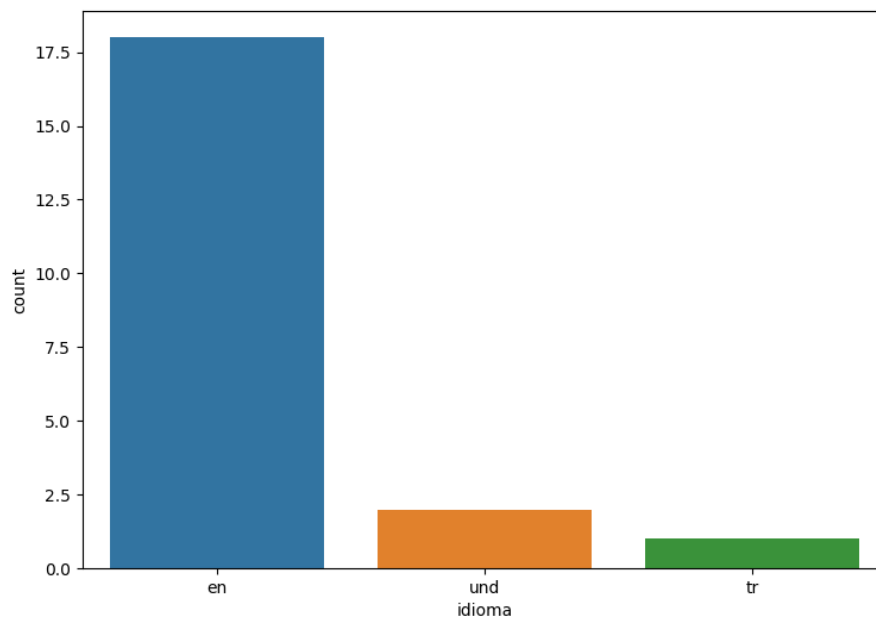


Figura 19: Número de Tweets por idioma relacionados con Trump

El sistema analiza la información en tiempo real, por eso mismo, las gráficas se actualizan cada 40 segundos. La figura 20 muestra la actualización de la figura 19. A diferencia de la figura 19, se incorporó el idioma portugués como cuarto idioma de Tweets sobre Trump.

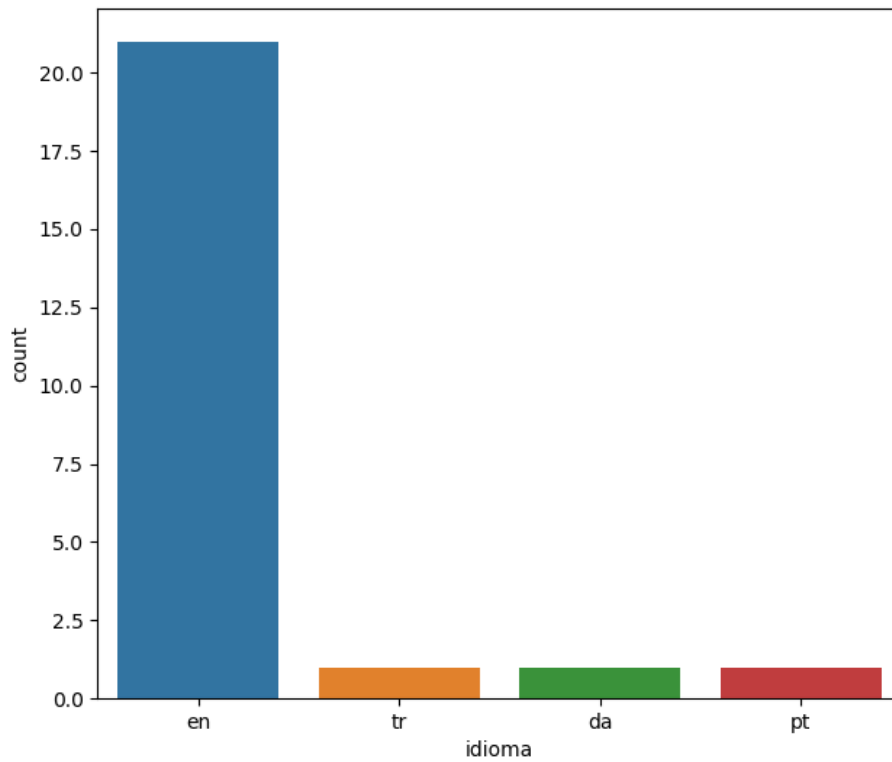


Figura 20: Número de Tweets por idioma relacionados con Trump después de 40 segundos

La figura 21 muestra el número de Tweets por localidades. El lugar con mayor número de tweets es el país Estados Unidos y dentro del empate de segundo lugar están algunas ciudades específicas de Estados Unidos como Georgia y Houston al igual que países como Francia y Canada.

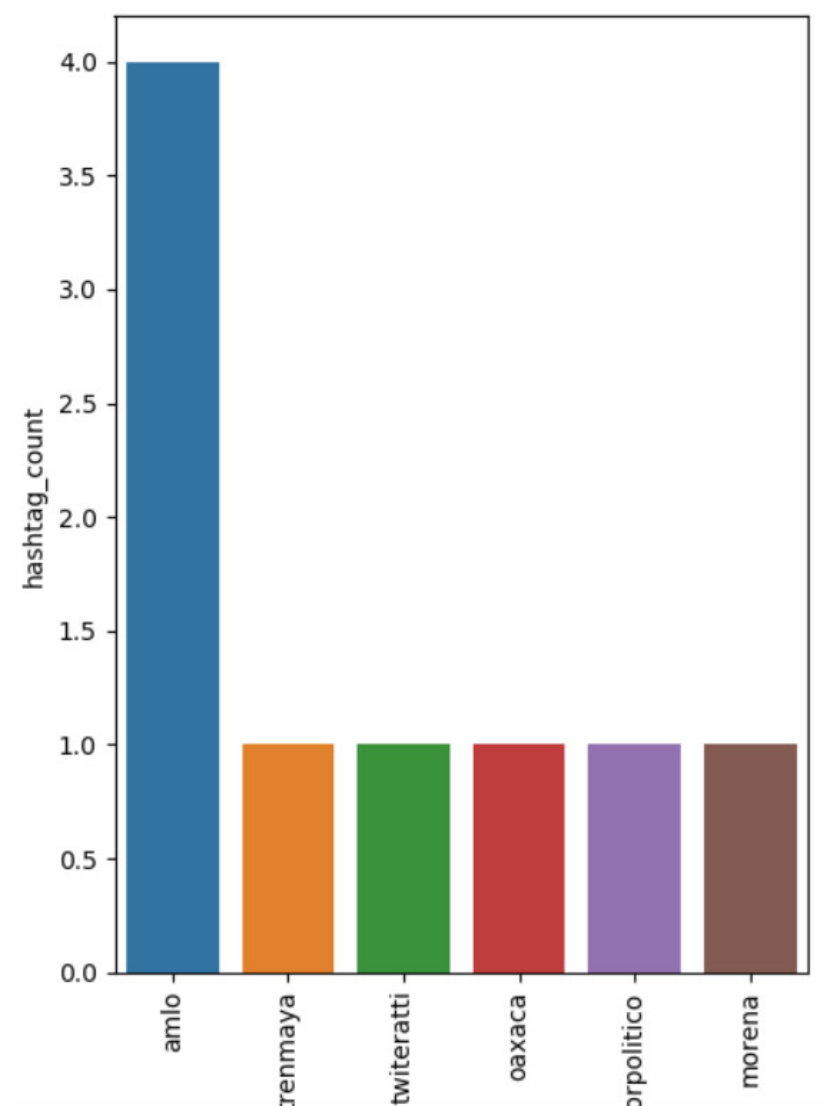


Figura 22: Hashtags relacionados con AMLO

Finalmente, la figura 23 muestra la frecuencia de las palabras más populares relacionadas con Trump omitiendo las preposiciones y artículos. Cabe resaltar que entre las palabras con mayor frecuencia se encuentra 'Syria' y 'wall' (muro).

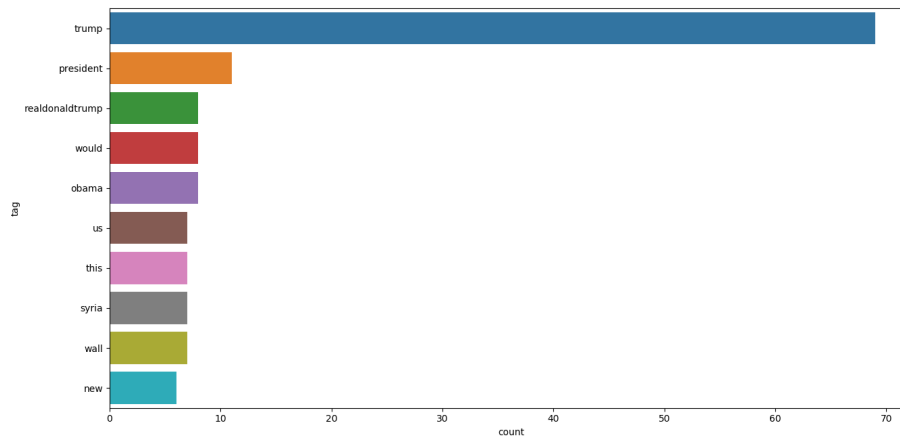


Figura 23: Frecuencia de palabras más populares relacionadas con Trump

9. Conclusiones

En conclusión, se logró el objetivo de analizar información en tiempo real proveniente de Twitter utilizando las herramientas de Spark Streaming, específicamente para extraer las palabras más usadas de un trending topic y graficar su frecuencia así como graficar las principales localidades de donde provienen los Tweets y el idioma en que están escritos.

Si bien este proyecto proporcionó ya un acercamiento al análisis de sentimiento, para trabajo futuro se podría realizar un análisis mucho más completo de los trending topics para entender no solo las palabras relacionadas sino también las emociones ligadas a dichos temas.

Referencias

- [1] Spark Streaming Programming Guide. Visitado el: 12/16/2018. Disponible en: <https://spark.apache.org/docs/latest/streaming-programming-guide.html>
- [2] ¿Qué es el streaming? (2016). Visitado el: 12/18/2018. Disponible en: <https://cehis.net/sitio/ayuda-video-streaming/asistencia-y-soporte/base-de-conocimiento-faq/ayuda-video-streaming/que-es-y-para-que-sirve-el-streaming>

- [3] Mazdakh (2017).The Infrastructure Behind Twitter: Scale. Visitado el: 12/19/2018. Disponible en: https://blog.twitter.com/engineering/en_us/topics/infrastructure/2017/the-infrastructure-behind-twitter-scale.html