5-9-2018



**LENGUAJES DE PROGRAMACIÓN**

**PROYECTO SEGUNDO PARCIAL**

**BUSCADOR DE EVENTOS Y CONGRESOS CIENTÍFICOS BASADO EN REDES SOCIALES**

Gabriel Andrés Del Pino Paredes

Alix Patricia Ferrín Alcívar

Juan Jose Flores Solorzano

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

SVSDV

**Contenido**

[**Introducción** 2](#_Toc523944196)

[**Objetivos** 2](#_Toc523944197)

[**Marco Teórico** 3](#_Toc523944198)

[**Alcance del proyecto** 5](#_Toc523944199)

[**Metodología** 5](#_Toc523944200)

[**Resultados esperados** 5](#_Toc523944201)

[**Resultados Obtenidos** 6](#_Toc523944202)

[**Preprocesamiento de la información** 8](#_Toc523944203)

[**Recomendaciones** 10](#_Toc523944204)

[**Referencias** 10](#_Toc523944205)

**Tabla de ilustraciones**

[**Figura 1:** Código para obtener tweets mediante el API tweepy 6](file:///C:\Users\alixi\Documents\GitHub\ProyectoLenguaje\Proyecto%202%20parcial.docx#_Toc523944649)

[**Figura 2:** CSV de tweets obtenidos mediante el API tweepy 7](file:///C:\Users\alixi\Documents\GitHub\ProyectoLenguaje\Proyecto%202%20parcial.docx#_Toc523944650)

[**Figura 3 :** Código para mostrar los tweets y la cantidad de veces que se han repetido 8](file:///C:\Users\alixi\Documents\GitHub\ProyectoLenguaje\Proyecto%202%20parcial.docx#_Toc523944651)

[**Figura 4:** Lista de tweets con su id y cantidad de veces que se han repetido 8](file:///C:\Users\alixi\Documents\GitHub\ProyectoLenguaje\Proyecto%202%20parcial.docx#_Toc523944652)

[**Figura 5:** Visualización de tweets con su frecuencia en aplicación realizada en Android Studio 9](#_Toc523944653)

[**Figura 6:** Visualización de tweets con su frecuencia en aplicación realizada en Android Studio 9](#_Toc523944654)

# **Introducción**

En la actualidad, existe un sin número de información digitalizada, información que está al alcance de cualquier persona que cuente con un dispositivo electrónico, sin embargo, mucha de esos datos que se encuentran en el internet no están clasificados u organizados adecuadamente, es decir, no pasa por un proceso que la convierta en información, de hecho, en redes sociales se puede encontrar cualquier tipo de información, la cual puede ser ingresada por cualquier usuario, sin necesidad de estar corroborada. Es por todo esto que se necesita de un filtro que permita al usuario encontrar ya no datos, sino más bien información que le ayude a la hora de buscar algo que le interese, en este caso, el enfoque está dirigido a eventos y congresos científicos, lo cual conlleva una responsabilidad a la hora de filtrar toda esa información que luego será mostrada al usuario.

# **Objetivos**

* Extraer información sobre eventos y congresos científicos desde la red social Twitter mediante el proceso llamado webscrapping.
* Pre-procesar esta información recolectada con java de una manera efectiva.
* Visualizar en una aplicación hecha en Android Studio toda la información pre-procesada mediante búsquedas, recomendaciones y votaciones del usuario.
* Visualizar en una gráfica cuantas veces se han repetido los tweets encontrados.

# **Marco Teórico**

**Web-Scrapping**

Web scraping es una técnica utilizada mediante programas de software para extraer información de sitios web. Usualmente, estos programas simulan la navegación de un humano en la World Wide Web ya sea utilizando el protocolo HTTP manualmente, o incrustando un navegador en una aplicación.

El web scraping está muy relacionado con la indexación de la web, la cual indexa la información de la web utilizando un robot y es una técnica universal adoptada por la mayoría de los motores de búsqueda. Sin embargo, el web scraping se enfoca más en la transformación de datos sin estructura en la web (como el formato HTML) en datos estructurados que pueden ser almacenados y analizados en una base de datos central, en una hoja de cálculo o en alguna otra fuente de almacenamiento. Alguno de los usos del web scraping son la comparación de precios en tiendas, la monitorización de datos relacionados con el clima de cierta región, la detección de cambios en sitios webs y la integración de datos en sitios webs. También es utilizado para obtener información relevante de un sitio a través de los rich snippets.

En los últimos años el web scraping se ha convertido en una técnica muy utilizada dentro del sector del posicionamiento web gracias a su capacidad de generar grandes cantidades de datos para crear contenidos de calidad.

**Preprocesamiento de datos**

El preprocesamiento de datos es un paso preliminar durante el proceso de minería de datos. Se trata de cualquier tipo de procesamiento que se realiza con los datos brutos para transformarlos en datos que tengan formatos que sean más fáciles de utilizar.

En el mundo real, los datos frecuentemente no están limpios, faltan valores clave, contienen inconsistencias y suelen mostrar ruido, conteniendo errores y valores atípicos. Sin un preprocesamiento de datos, estos errores en los datos sobrevivirían y disminuirían la calidad de la minería de datos.

Algunos de las tareas de preprocesamiento de datos son las siguientes:

* Rellenar valores faltantes
* Identificar y eliminar datos que se pueden considerar un ruido
* Resolver redundancia
* Corregir inconsistencias

Los datos están disponibles en varios formatos, tales como formas estáticas, categóricas, numéricas y dinámicas. Algunos ejemplos incluyen metadatos, web data, texto, vídeo, audio e imágenes. Estas formas de datos tan variadas contribuyen a que el procesamiento de datos continuamente se encuentre con nuevos desafíos.

**Lenguaje R**

R es un entorno y lenguaje de programación con un enfoque al análisis estadístico.

R es una implementación de software libre del lenguaje S, pero con soporte de alcance estático. Se trata de uno de los lenguajes más utilizados en investigación por la comunidad estadística, siendo además muy popular en el campo de la minería de datos, la investigación biomédica, la bioinformática y las matemáticas financieras. A esto contribuye la posibilidad de cargar diferentes bibliotecas o paquetes con funcionalidades de cálculo y gráficas.

R es parte del sistema GNU y se distribuye bajo la licencia GNU GPL. Está disponible para los sistemas operativos Windows, Macintosh, Unix y GNU/Linux.

**Lenguaje Java**

Java es un lenguaje de programación orientado a objetos que se incorporó al ámbito de la informática en los años noventa. La idea de Java es que pueda realizarse programas con la posibilidad de ejecutarse en cualquier contexto, en cualquier ambiente, siendo así su portabilidad uno de sus principales logros. Fue desarrollado por Sun Microsystems, posteriormente adquirido por Oracle. En la actualidad puede utilizarse de modo gratuito, pudiéndose conseguir sin problemas un paquete para desarrolladores que oriente la actividad de programar en este lenguaje. Puede ser modificado por cualquiera, circunstancia que lo convierte en lo que comúnmente se denomina “código abierto”.

**Lenguaje Python**

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.

Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma.

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License,​ que es compatible con la Licencia pública general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores.

**Android Studio**

Android Studio es el entorno de desarrollo integrado oficial para la plataforma Android. Fue anunciado el 16 de mayo de 2013 en la conferencia Google I/O, y reemplazó a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. La primera versión estable fue publicada en diciembre de 2014.

Está basado en el software IntelliJ IDEA de JetBrains y ha sido publicado de forma gratuita a través de la Licencia Apache 2.0. Está disponible para las plataformas Microsoft Windows, macOS y GNU/Linux. Ha sido diseñado específicamente para el desarrollo de Android.

Estuvo en etapa de vista previa de acceso temprano a partir de la versión 0.1, en mayo de 2013, y luego entró en etapa beta a partir de la versión 0.8, lanzada en junio de 2014. La primera compilación estable, la versión 1.0, fue lanzada en diciembre de 2014.​ La última versión estable es la 3.0, y fue lanzada en octubre de 2017

# **Alcance del proyecto**

Este proyecto permitirá a un usuario recomendar y votar eventos o congresos científicos mediante distintos factores, ya sea fecha, ubicación, interés de este, etc. Todo esto a partir de una recolección de datos, la cual una vez obtenida de la red social Twitter será pre-procesada, es decir, se realizará un proceso de limpieza, selección y uniformidad a los datos obtenidos para convertirlos en información. Esta información luego será mostrada al usuario mediante una aplicación que le permita a él interactuar con ella.

# **Metodología**

Debido a nuestro poco conocimiento del tema, y que las destrezas a tomar en cuenta para el proyecto no abarcan este tema, la primera intención es buscar una librería que permita realizar una rápida y modesta limpieza de información, para así evitar problemas que pueden costar tiempo en su solución, así de esta manera el enfoque queda en la etapa de visualización de la información obtenida

Para la primera parte, obtención de datos, se utilizó como herramienta el api tweetpy

Para la sección de preprocesamiento de información, utilizamos el lenguaje java debido a que Android Studio soporta el mismo lenguaje, procesamos el csv recibido para obtener un arreglo de objetos, el tweet específico y la cantidad de veces que se ha repetido, enlazamos esta clase con la clase prediseñada en Android Studio, ingresarlas en una ListView y mostrarlas en el dispositivo

El proceso de visualización consiste en programar una aplicación en Android Studio que permita al usuario realizar búsqueda de eventos y congresos científicos, y a su vez, pueda recomendar y votar sobre ellos para una futura búsqueda.

# **Resultados esperados**

La finalidad principal del desarrollo de este proyecto es el de poder implementar un sistema capaz de poder identificar correctamente ciertos eventos científicos que resulten de interés al usuario, y así poder mantenerlo informado de temas relevantes que acontecen al panorama tecnológico actual de la sociedad, además de poder establecer una mejor forma para organizar futuros eventos y la asistencia a ellos.

De esta forma, esperamos poder mejorar el flujo de información entre los panelistas y la audiencia potencial que pueden llegar a obtener, pero que muchas veces se ve reducida debido a la falta de impacto al transmitir los datos sobre el evento a realizarse, mediante el internet y las redes sociales.

# **Resultados Obtenidos**

Figura 1: Código para obtener tweets mediante el API tweepy

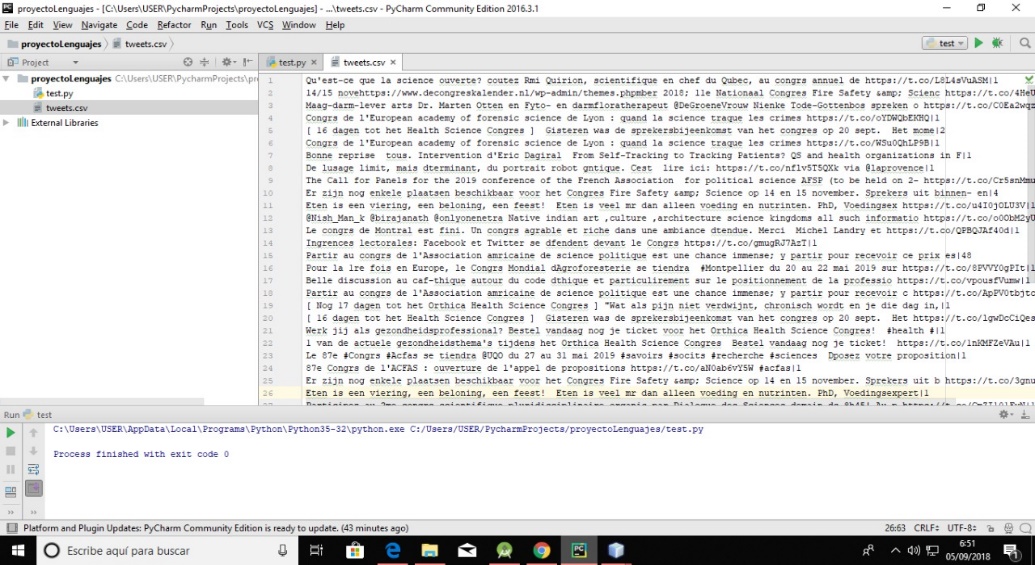
****

Figura 2: CSV de tweets obtenidos mediante el API tweepy

# **Preprocesamiento de la información**

Figura 3 : Código para mostrar los tweets y la cantidad de veces que se han repetido

Figura 4: Lista de tweets con su id y cantidad de veces que se han repetido

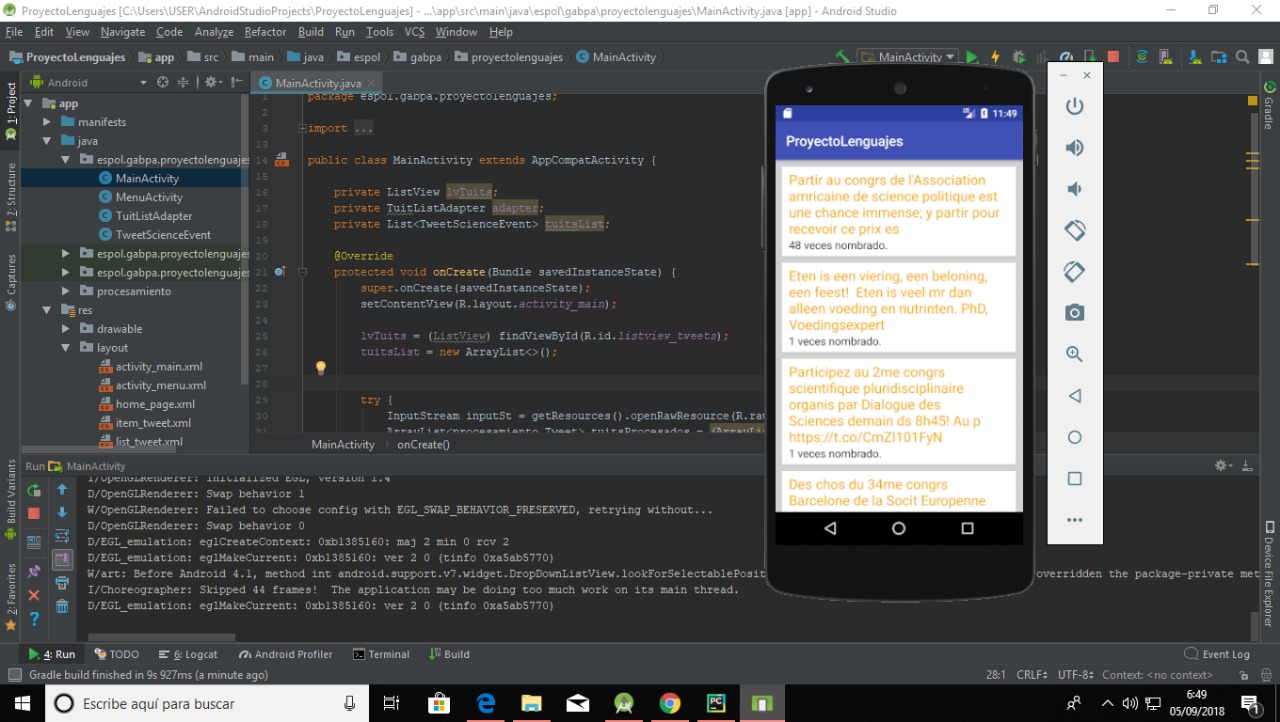
****

Figura 5: Visualización de tweets con su frecuencia en aplicación realizada en Android Studio

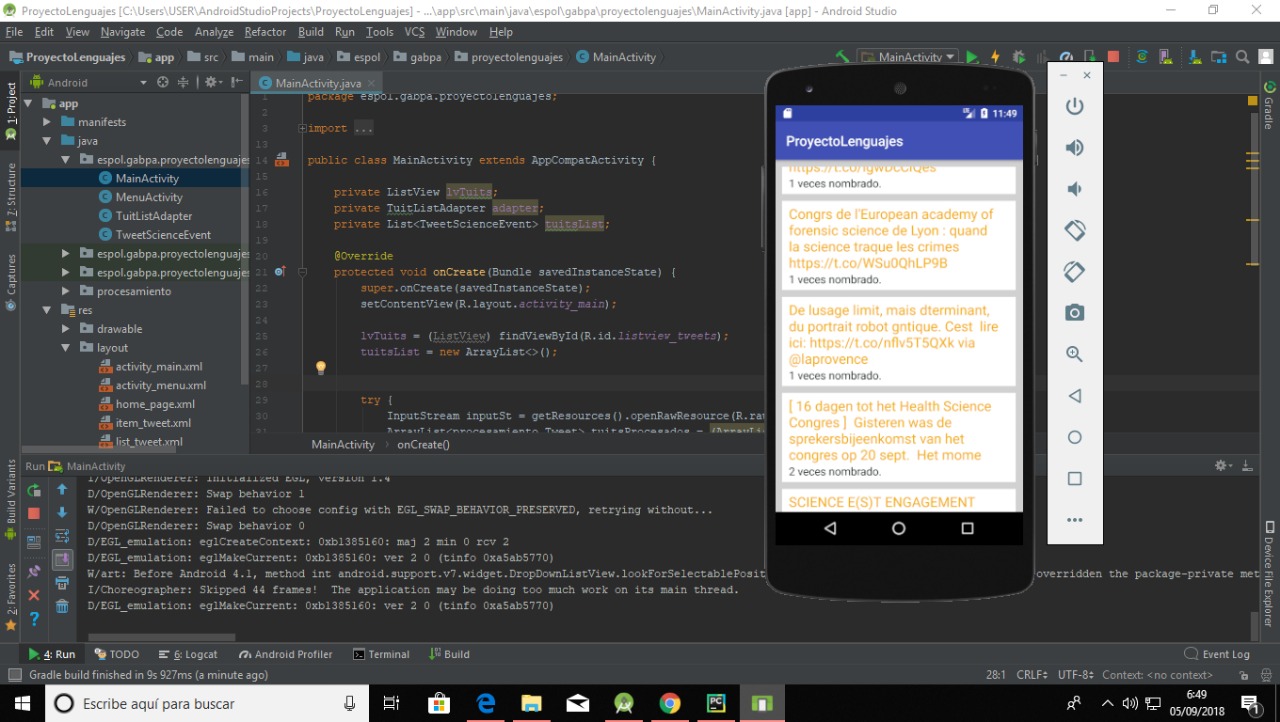
****

Figura 6: Visualización de tweets con su frecuencia en aplicación realizada en Android Studio

# **Recomendaciones**

Al involucrarse con la recolección de datos obtenidos de redes sociales, es de mayor utilidad y facilidad el uso de sus APIs oficiales, ya que ofrecen métodos de acceso y búsqueda muy útiles y con mucha flexibilidad. No utilizamos webscraping debido a que este depende mucho de la estructural html de la pagina y, en el caso de las búsquedas de Twitter, cada tweet puede tener una estructura distinta, lo cual complicaría mucho el desarrollo de este proyecto.

Además, es recomendable tener sólidos conocimientos acerca del Modelo Vista Controlador (MVC), para poder tener un mejor entendimiento de como esta estructurado un proyecto en Android Studio y desarrollar una aplicación con menos dificultades.

Por último, el entendimiento del Paradigma Orientado a Objetos es de mucha utilidad, ya que Android Studio soporta el lenguaje Java, el cual utiliza principalmente este paradigma y le resultara de gran ayuda al programador entender como distintas instancias de clases interactúan entre sí.

# **Referencias**

1. Martí, M. 08/04/2016. ¿Qué es el Web scraping? Introducción y herramientas. Lugar de publicación: [*https://sitelabs.es/web-scraping-introduccion-y-herramientas/*](https://sitelabs.es/web-scraping-introduccion-y-herramientas/)
2. (S.a). 30/12/2016. Calidad de datos en minería de datos a través del preprocesamiento. Lugar de publicación: [*https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/calidad-de-datos-en-mineria-de-datos-a-traves-del-preprocesamiento*](https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/calidad-de-datos-en-mineria-de-datos-a-traves-del-preprocesamiento)
3. (S.a). (s/f). Definición de Java. Lugar de publicación:[*https://definicion.mx/java/*](https://definicion.mx/java/)
4. Alvarez, M.A. 19/11/2003. ¿Qué es Python? Lugar de publicación: [*https://desarrolloweb.com/articulos/1325.php*](https://desarrolloweb.com/articulos/1325.php)
5. (S.a). (s/f). Android Studio v1.0: características y comparativa con Eclipse. Lugar de publicación: [*https://academiaandroid.com/android-studio-v1-caracteristicas-comparativa-eclipse/*](https://academiaandroid.com/android-studio-v1-caracteristicas-comparativa-eclipse/)