

---

## ANALIZADOR DE SENTIMIENTOS SOBRE MENSAJES EN REDES SOCIALES

---

202200135 – Joab Israel Ajsivinac Ajsivinac

### Resumen

La empresa “Tecnologías Chapinas, S.A” requiere el desarrollo de una nueva tecnología que permita el análisis de sentimientos de mensajes provenientes de la red social “Twitter” o actualmente renombrada como “X”, mediante la carga de un listado de palabras positivas y negativas, para lograr obtener una respuesta que se pueda cuantificar.

El programa carga datos de entrada mediante archivos XML para luego procesarlos. El sistema cuenta con dos partes principales las cuales son la parte del back-end, que es la que se encarga de procesar los datos y manejar la lógica detrás de los cálculos mediante una API (Interfaz de programación de aplicaciones) y el front-end que es la que se encarga de mostrar los datos al usuario de una forma intuitiva y fácil de entender, de forma que el usuario pueda interactuar, recibir o utilizar las capacidades del back-end.

El sistema busca simplificar el análisis de los mensajes en las redes sociales, para que los analistas sean más eficientes en las labores de procesar y analizar los diferentes mensajes.

### Palabras clave

Mensajes, emociones, back-end, front-end, analizar

### Abstract

*The company "Tecnologías Chapinas, S.A" requires the development of a new technology that allows for sentiment analysis of messages from the social network "Twitter," or currently known as "X," by loading a list of positive and negative words to obtain a quantifiable response.*

*The program loads input data through XML files and processes them. The system consists of two main parts: the back end, responsible for processing data and handling the logic behind calculations through an Application Programming Interface (API), and the front-end, which is responsible for presenting data to the user in an intuitive and easily understandable manner, allowing users to interact, receive, or utilize the capabilities of the back end.*

*The system aims to simplify the analysis of messages on social networks, making analysts more efficient in processing and analyzing different messages.*

### Keywords

*Messages, emotions, back-end, front-end, analyze.*

## Introducción

El sistema de analizado de mensajes de redes sociales es una herramienta dividida en dos partes, las cuales son el back-end que contiene toda la lógica detrás del análisis de mensajes, y el front-end que es el que se encarga de mostrar los datos a los usuarios. El análisis de mensajes resulta de gran importancia en la industria del análisis de datos para las diferentes empresas que trabajan sobre esta problemática, ya que facilita la comprensión del contenido de un mensaje dentro de una red social.

El software proporciona la forma de poder obtener y generar gráficas para poder llevar a cabo un análisis estadístico de las diferentes consultas que puede realizar el sistema, los reportes son generados en formato PDF para poder facilitar el uso de los datos obtenidos. Además de ser visualizada de la forma más amigable posible de cara al usuario.

## Desarrollo del tema

El sistema hace uso de listas, tuplas, diccionarios, sets nativos de Python, para un mejor control de la información. El software consta de dos partes, el back-end que se encarga de cargar y actualizar la base de datos, así como tener guardada la información de los mensajes y diccionarios de palabras, con el fin de que el procesamiento sea más efectivo esto mediante de endpoints los cuales son componentes esenciales en arquitecturas de red y sistemas de comunicación, son utilizados para transmitir datos, solicitudes y respuestas a través de una red. A su mismo, el uso de sets facilita el manejo de datos que pueden llegarse a repetir, ya que es un tipo de colección de datos diseñada para almacenar elementos distintos, y no permiten duplicados.

La base de datos es una implementación de archivos XML donde se guardan los datos para que permanezcan a lo largo del tiempo, simulando tablas y haciendo el trabajo que hace una base de datos. Siendo cargados al inicializar el sistema.

La segunda parte es el front-end el cual tiene la tarea de llamar a los endpoints respectivos para obtener la información deseada, para poder ser mostrada a los usuarios utilizando una interfaz amigable, donde los datos sean fáciles de entender, y de forma que el usuario solo tenga que interactuar con botones y paneles, para su visualización

### a. Carga de Datos

Este apartado se divide en dos, ya que se puede cargar dos tipos de archivos en formato XML, el primer tipo son archivos que contienen la configuración del diccionario de palabras positivas y negativas, y el segundo tipo son los mensajes, los cuales cuentan con la información necesaria para poder ser procesados adecuadamente.

Una vez se tiene cargados los archivos respectivos, el programa manda el contenido al backend, dependiendo del tipo de archivo cargado se procesa de diferente manera:

#### a.1 Procesamiento de archivos de Configuración

Con respecto a este apartado, el programa lee las palabras, pasando las palabras a minúscula, quitándole las tildes, para poder manejar bien las palabras, estas palabras se guardan en un archivo XML, para que el almacenamiento del diccionario de palabras permanezca a lo largo del tiempo, a su vez, se tiene un análisis de palabras, ya que si una palabra ya está registrada como palabra positiva, y se vuelve a ingresar la misma palabra pero como negativa,

dicha palabra se almacenara en otro tag denominado negativo rechazada, de igual forma si el caso es contrario, esto para llevar un mejor manejo de las diferentes palabras. Finalmente, las palabras positivas y negativas solo se ingresan una sola vez, si vienen más palabras de un mismo tipo que son iguales no se consideran como un error, solo se omiten los valores.

## **a.2 Procesamiento de archivos de Mensajes**

Para el procesamiento de archivos que contienen los mensaje, se usan expresiones regulares para poder agrupar los hashtags, y los usuarios mencionados, junto con una expresión regular para poder obtener la fecha del mensaje, una vez se tiene esta información se procede a ver que tipo de sentimiento, si hay más palabras positivas, el mensaje se denomina como positivo, si hay más palabras negativas se denomina como negativo, si la cantidad de palabras positivas y negativas son iguales, el mensaje es neutro.

Luego de cada carga se genera un archivo de resumen, y se muestra en pantalla en un apartado diferente el resumen respectivo dependiendo del tipo de subida que se elija.

## **b. Peticiones**

Dentro de este apartado se puede seleccionar que tipo de petición se requiere, además los resultados podrán ser exportados en formato .PDF mediante un botón dentro de la interfaz facilitando el manejo de la información obtenida. Para todos los filtros si no existen mensajes cargados, se le indicará al usuario que no hay mensajes, por ende, que no se puede buscar información respecto a los mensajes, lo anterior se muestra mediante el uso de una ventana emergente que desglosa la información antes mencionada.

Para la petición de los rangos de fechas se hace uso de un calendario, donde se puede seleccionar un rango en específico, facilitando el uso de la interfaz, simplificando el trabajo del usuario.

Para cada petición el sistema agrupa por fechas la información según el tipo de petición elegida, para posteriormente aplicarle las condiciones para poder obtener la información requerida, para el rango de fechas se hace uso de la librería datetime, para poder comparar las fechas de una manera eficiente, aplicando el formato de fecha: DD/MM/YYYY, dos dígitos para el día, dos dígitos para el mes, y 4 dígitos para el año. Una vez se tenga la información, la API regresa los datos por medio de un JSON, el cual el front-end recibe para poder mostrar la información a los usuarios de forma amigable.

Las peticiones disponibles son:

### **b.1. Consultar Hashtags**

Una vez seleccionado el tipo de petición, el usuario elegirá un rango de fechas, para lo cual la API, regresará los datos de los hashtags para dicho rango, junto con información para la creación de una gráfica de barras, la cual estará desplegada en pantalla junto con la información de los hashtags, la información estará conformada por la fecha como un encabezado, creando un marco donde se encontrarán los hashtags encontrados y el número de veces que se encuentra dentro de esa fecha.

Al ser un rango esta información se repite según el número de coincidencias encontradas, si no se encuentra información para el rango indicado, se mostrar en el apartado respectivo un mensaje de que no se encontró información.

## **b.2. Consultar Menciones**

Similar a la consulta anterior, el usuario elegirá un rango de fechas, para lo cual la API, regresará los datos de las menciones para dicho rango, junto con información para la creación de una gráfica de barras, la cual estará desplegada en pantalla junto con la información de las menciones, la información estará conformada por la fecha como un encabezado creando un cuadro donde se despliega a los usuarios mencionados encontrados y el número de veces que se encuentra dentro de esa fecha.

Al ser un rango esta información se repite según el número de coincidencias encontradas, si no se encuentra información para el rango indicado, se mostrar en el apartado respectivo un mensaje de que no se encontró información.

## **b.3. Consultar Sentimientos en Mensajes**

Similar a las consultas anteriores, el usuario elegirá un rango de fechas, para lo cual la API, regresará los datos de las emociones por mensaje para dicho rango, junto con información para la creación de una gráfica de barras, la cual estará desplegada en pantalla junto con la información de las menciones, la información estará conformada por la fecha como un encabezado, creando un contenedor donde está el tipo de sentimiento y el número de veces que se encuentra dentro de esa fecha.

Al ser un rango esta información se repite según el número de coincidencias encontradas, si no se encuentra información para el rango indicado, se mostrar en el apartado respectivo un mensaje de que no se encontró información.

## **c. Ayuda**

Esta opción tiene el propósito de ayudar al usuario a conocer a la persona que desarrollo el sistema, además de contar con la opción de poder abrir la documentación del proyecto que esta subida en la red.

## **Conclusiones**

Con base en el análisis de los datos y la comprensión del software para el análisis de sentimientos en mensajes proveniente de redes sociales, se puede resaltar la eficacia y las posibles áreas de mejor para esta solución innovadora.

El uso de estructuras nativas de Python para el almacenamiento de la información ha demostrado ser eficiente en términos de eficiencia y ahorro de líneas de código, lo que resulta importante para el rendimiento en general del programa. A sí mismo, el uso de archivos XML para el guardado de información para asegurar que los datos perduren a lo largo del tiempo, resulto bastante eficiente para este caso en cuestión, ya que se puede manejar la estructura del archivo a conveniencia

El uso de librerías de JavaScript para el despliegue de gráficas de barras permite la visualización de la información relevante del sistema, de forma que sea más fácil de comprender para los usuarios.

El software representa un avance prometedor en el campo del análisis del comportamiento humano. Los resultados obtenidos indican su potencial y eficacia, sin embargo, persisten desafíos y oportunidades para mejorar que invitan a futuras investigación y desarrollo.

## Referencias bibliográficas

Django (s.f). *Página oficial de Django*. The Web framework for perfectionists with deadlines.

<https://www.djangoproject.com/>

Flask (s.f). *Página oficial de Flask*. The Python Web Framework.

<https://flask.palletsprojects.com/en/3.0.x/>

Mozilla (s.f). *Learn Server-side Development with Django*. Mozilla Developer Network (MDN).

<https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django>

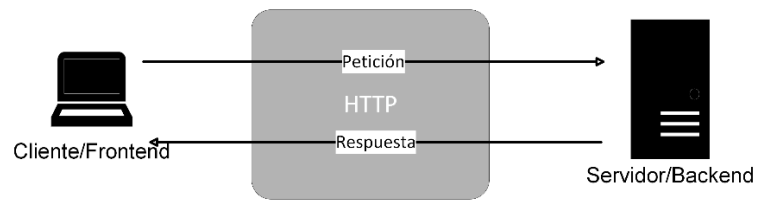


Figura 2. Diagrama cliente servidor

Fuente: Elaboración Propia

## Anexos

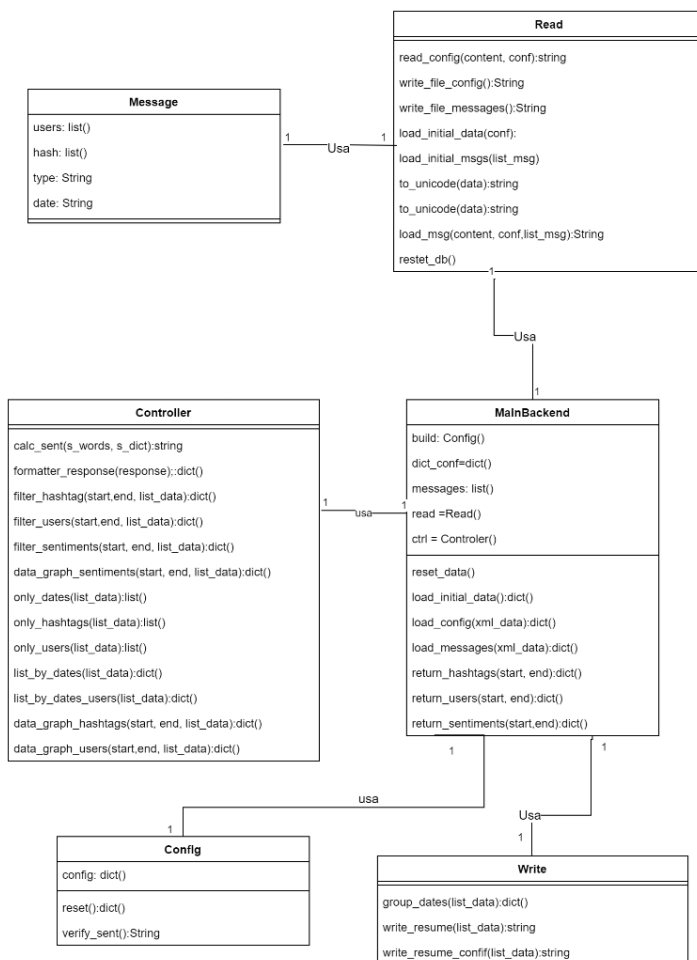


Figura 1. Diagrama de Clases, API

Fuente: Elaboración Propia