
SISTEMA AUTÓNOMO DE “CHAPIN WARRIORS S.A.”

201908355 – Danny Hugo Bryan Tejaxún Pichiyá

Resumen

A medida que crece la empresa y el avance de la tecnología continua, se requiere que las empresas sepan el nivel de aceptación o rechazo que tienen sobre sus clientes para poder mejorar en ciertos aspectos. Por lo que se necesita de la implementación de algoritmos de búsqueda de palabras clave en comentarios en las redes sociales para determinar el impacto sobre los clientes.

El programa carga la información mediante archivos de entrada, además de contar con funcionalidades de análisis estadístico mediante gráficas.

Los comentarios se clasifican en tres tipos distintos los cuales determinan si el impacto de los servicios de la empresa es positivo, negativo o neutral. Los comentarios son clasificados por fecha, empresa y servicios que prestan.

Es posible afirmar que el sistema cumple de manera eficiente con su función, y que demuestra que tan aceptada es la empresa y asimismo qué tan bien brindan sus servicios.

Palabras clave

Palabras Clave, Sentimientos, Empresa, Servicio, Archivos de Entrada.

Abstract

The company and technology are changing so it is need that the companies know the acceptance or rejection it has about its clients for in order to improve. Then it is need implement of search algorithms of keywords in comments of social networks for know impact on clients.

The program load data through input files also it has statistical analysis function through graphics.

The comments are classified in three types that make it know if the impact of services of the company are positive, negative or neutral. The comments are classified for date, company and services.

Is possible say that the system is doing its function and prove if a company is accepted by clients and that how good give its services.

Keywords

Keyword, Feelings, Company, Service, Input File.

Introducción

Para las empresas es importante conocer qué tan aceptadas son para sus clientes y potenciales clientes, además de obtener retroalimentación sobre la forma en la que sus trabajadores brindan cierto servicio.

Tecnologías Chapinas S.A. ha implementado un sistema que encuentra palabras claves en los comentarios de las redes sociales de las empresas que determina, mediante palabras clave, si la empresa o servicio o ambos son aceptados.

Se utiliza un diccionario de palabras que determinan sentimientos positivos y negativos en los comentarios, que servirán como parámetro de evaluación de los comentarios y así clasificarlos como comentarios positivos, negativos o neutrales.

Desarrollo del tema

El sistema funciona mediante dos programas que son Frontend y Backend. El frontend es el programa que está construido para que los usuarios interactúen de forma cómoda con el backend, programa que ejecuta los procesos lógicos y matemáticos. Ambos programas se comunican mediante métodos http y peticiones que actúan como parámetros de búsqueda. Es posible almacenar información en el backend mediante estructuras de control y programación orientada a objetos, pero para lograr persistencia de datos también es posible conectar una base de datos.

El sistema es manejado mediante interfaz gráfica construida con html y el framework Django, el backend. Para la persistencia de datos se utilizó la base de datos que incorpora Django.

El sistema tiene una funcionalidad para generar gráficos que den a entender de manera más clara y estadística el análisis realizado a los comentarios de determinada empresa en cierta fecha o de forma

general (todas las empresas a la vez). También se cuenta con una funcionalidad para generar gráficos que den a entender de forma estadística y legible la distribución de mensajes de cada tipo en un determinado intervalo de fechas. Al consultar los datos es posible ver el contenido del archivo de entrada al mismo tiempo que el de salida, ambos datos son mostrados con formato XML.

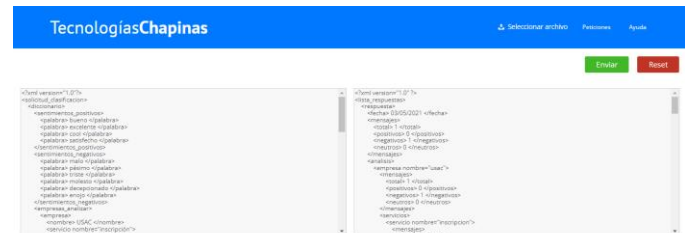


Figura 1. Ejemplo de pantalla principal.

Fuente: Elaboración propia.

a) Carga de Datos

Los datos se cargan en el sistema seleccionando el archivo XML en el sistema, conteniendo la información de las empresas con sus respectivos servicios, además de que son mostrados los comentarios que contiene la fecha y hora en que se escribió, el usuario que lo escribió, la red social en la que se escribió y el cuerpo del comentario.



Figura 2. Ejemplo de visualización de entrada.

Fuente: Elaboración propia.

b) Análisis de Datos

Del archivo de entrada se obtienen diccionarios de palabras que denotan sentimientos positivos de los clientes y otro diccionario que denota sentimientos negativos de los usuarios. Mediante el los diccionarios se busca en el cuerpo de los mensajes si se encuentran palabras que se encuentran en alguno de los diccionarios, al finalizar se hace un conteo de palabras positivas y negativas para cada mensaje y se determina la positividad o negatividad del mensaje mediante el conteo de palabras, si hay más palabras positivas que negativas es comentario positivo, de lo contrario es comentario negativo y si hay la misma cantidad de palabras positivas y negativas entonces se clasifica el mensaje como neutral.

```
<?xml version="1.0" ?>
<lista_respuestas>
  <respuesta>
    <fecha> 03/05/2021 </fecha>
    <mensajes>
      <total> 1 </total>
      <positivos> 0 </positivos>
      <negativos> 1 </negativos>
      <neutros> 0 </neutros>
    </mensajes>
    <analisis>
      <empresa nombre="usac">
        <mensajes>
          <total> 1 </total>
          <positivos> 0 </positivos>
          <negativos> 1 </negativos>
          <neutros> 0 </neutros>
        </mensajes>
      </empresa>
      <servicios>
        <servicio nombre="inscripcion">
          <mensajes>
            <total> 1 </total>
            <positivos> 0 </positivos>
            <negativos> 1 </negativos>
            <neutros> 0 </neutros>
          </mensajes>
        </servicio>
      </servicios>
    </analisis>
  </respuesta>
</lista_respuestas>
```

Figura 3. Ejemplo de visualización de salida.

Fuente: Elaboración propia.

c) Graficación

La graficación es posible realizarla de dos formas.

La primera forma es mediante una fecha específica, se selecciona la fecha y posteriormente se selecciona una empresa en específico de lo contrario se toman todas las empresas para realizar el conteo.

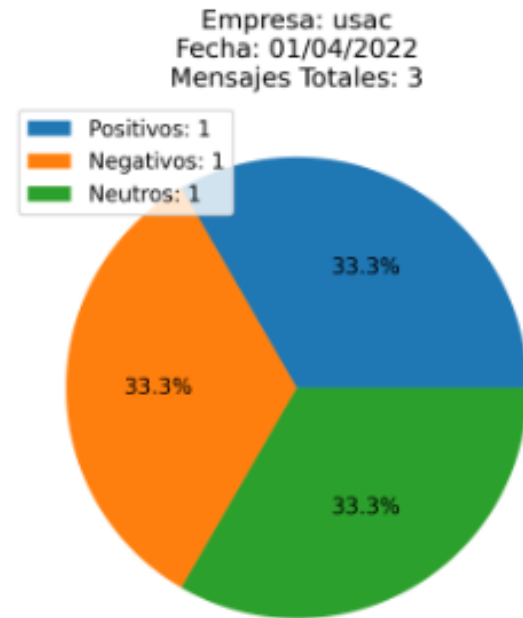


Figura 4. Ejemplo de gráfica de mensajes de una empresa en una fecha específica.

Fuente: Elaboración propia.

La segunda forma es mediante un intervalo de fechas específicas, se selecciona la fecha y posteriormente se selecciona una empresa en específico de lo contrario se toman todas las empresas para realizar el conteo. En este caso se genera más de una gráfica ya que puede haber mensajes de distintas fechas.

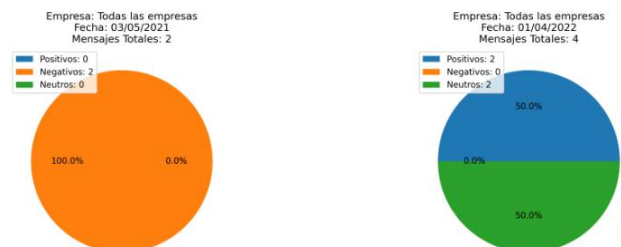


Figura 5. Ejemplo de gráfica de mensajes de una empresa en un rango de fechas específico.

Fuente: Elaboración propia.

Lógica del Sistema

Toda la información es almacenada en la base de datos que incorpora Django para tener persistencia en la información.

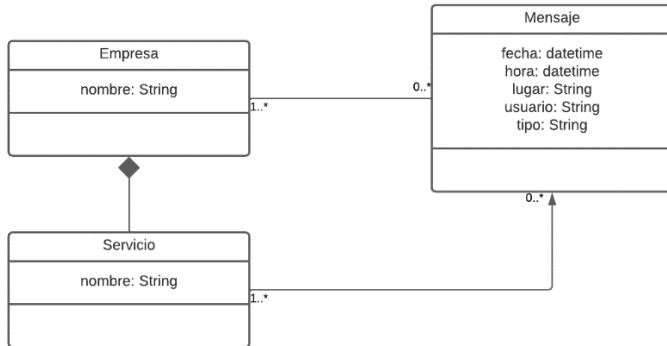


Figura 6. Diagrama de Clases.

Fuente: Elaboración propia.

En el diagrama se puede observar de manera más completa la organización lógica de la codificación para almacenar temporalmente mientras se ejecutan las operaciones necesarias para guardarlas en la base de datos.

La API fue desarrollada usando el framework Flask, sus endpoints permiten al frontend interactuar con el backend mediante métodos http siendo los más importantes:

1. **POST**
2. **GET**
3. **DELETE**

Conclusiones

Luego de comprender las funciones del sistema y de las clases implementadas se puede afirmar que el sistema si cumple de forma eficaz con el análisis de mensajes.

Para la clasificación de los mensajes es posible determinar su tipo según sean el cuerpo del mensaje, sin embargo aún existe probables errores al momento de analizar ya que únicamente se buscan las palabras pertenecientes a los diccionarios.

Se usó la librería matplotlib para la generación de gráficas de modo que la información se vuelve más compacta y entendible.

Bibliografía

Flask. (s.f.). Obtenido de
<https://flask.palletsprojects.com/en/2.1.x/>
Foundation, D. S. (s.f.). *Django*. Obtenido de
<https://docs.djangoproject.com/en/4.0/>
Hunter, J., Dale, D., Firing, E., & Droettboom, M.
(s.f.). *Mathplotlib*. Obtenido de
https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.pie.html