



**Análisis de sentimientos en tiempo real para intervenciones de chat en videoconferencias  
por streaming**

Alexánder Uribe González

Trabajo de grado para optar al título de Ingeniero de Sistemas

Asesor:

Antonio Jesús Tamayo Herrera, Doctor en Ciencias de la Computación

Universidad de Antioquia

Facultad de Ingeniería

Pregrado UdeA

Medellín

2023

- Referencia** [1] Uribe González, “Análisis de sentimientos en tiempo real para intervenciones de chat en videoconferencias por streaming”, pregrado UdeA, Universidad de Antioquia, Medellín, 2023.
- Estilo IEEE (2020)



Centro de Documentación de Ingeniería (CENDOI)

**Repositorio Institucional:** <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - [www.udea.edu.co](http://www.udea.edu.co)

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

### **Dedicatoria**

Le dedico este éxito a mi familia, en especial a mi esposa Lina María Hernández Daza, a mis hijos José Manuel y Juan José, quienes, con toda bondad, cedieron sus espacios para que yo pudiera terminar mis estudios. A mis padres, Amparo González Bustamante y Gabriel Uribe Ardila, quienes asumieron parte de mis responsabilidades, para poder dedicarme a mis estudios y en general, a los que en medio de mis incertidumbres y temores me animaron a continuar.

### **Agradecimientos**

En primer lugar, le agradezco a mi Señor Jesús, dedico el fruto de este esfuerzo a mi familia y hago un reconocimiento muy especial a mi asesor, el Doctor Antonio Jesús Tamayo Herrera, quien puso a mi disposición todo su conocimiento y experiencia, para poder encontrar la mejor solución, desarrollarla y hacerla productiva.

TABLA DE CONTENIDO

RESUMEN	8
ABSTRACT	9
I. INTRODUCCIÓN	10
II. PLANTEAMIENTO DEL PROBLEMA	12
III. JUSTIFICACIÓN	13
IV. OBJETIVOS	14
V. HIPÓTESIS	15
VI. MARCO TEÓRICO	16
VII. METODOLOGÍA	20
VIII. RESULTADOS	38
IX. DISCUSIÓN	49
X. CONCLUSIONES	50
XI. RECOMENDACIONES	52
REFERENCIAS	53

LISTA DE TABLAS

TABLA I IMPACT MAPPING	28
TABLA II PRIORIZACIÓN	29
TABLA III REGISTRO DE LOS RELEASE	31
TABLA IV MENSAJES POSITIVOS	39
TABLA V MENSAJES NEUTRALES	40
TABLA VI MENSAJES NEGATIVOS	41

LISTA DE FIGURAS

Fig. 1. Herramientas de desarrollo de software	20
Fig. 2. Validación del modelo de análisis de sentimientos	21
Fig. 3. Front-End (Single Page Application)	33
Fig. 4. Back-End (Api-Rest)	34
Fig. 5. Diagrama de despliegue	34
Fig. 6. Registro de la URL	35
Fig. 7. Panel de resultados (procesamiento de datos)	35
Fig. 8. Validación del modelo frases positivas	39
Fig. 9. Validación del modelo frases neutrales	40
Fig. 10. Validación del modelo frases negativas	41
Fig. 11. Patrón de diseño MVC (Modelo-Vista-Controlador)	42
Fig. 12. Diagrama del sistema de AS para chats de transmisiones en vivo	42
Fig. 13. Text, ingreso URL del video en vivo	44
Fig. 14. Activación del botón Monitor	44
Fig. 15. Activación módulo de chats del en vivo	45
Fig. 16. Activación del modelo de análisis de sentimientos	46
Fig. 17. Indicador de actividad del proceso	47
Fig. 18. Detención del proceso	47

SIGLAS, ACRÓNIMOS Y ABREVIATURAS

<b>IA</b>	Inteligencia Artificial
<b>UdeA</b>	Universidad de Antioquia
<b>AS</b>	Análisis de Sentimientos
<b>MO</b>	Minería de Opiniones
<b>NLTK</b>	Natural Language Toolkit
<b>NLP</b>	Procesamiento Lenguaje Natural
<b>APP</b>	Aplicación
<b>URL</b>	Localizador Uniforme de Recursos

## RESUMEN

En el momento presente se vienen desarrollando procesos y herramientas de inteligencia artificial, que permiten aplicar modelos a datos, con el fin de obtener información que no es posible realizarla con herramientas de análisis convencionales, dada la cantidad de datos a procesar y su estructura, por lo que se hace necesario adoptar tecnologías de última generación, para obtener el mayor provecho posible en el análisis de la información. Para el caso del presente trabajo, se propuso aprovechar estas tecnologías para darle valor a las interacciones que se realizan en los chats que producen los usuarios al escribir los comentarios de transmisiones en vivo, con lo cual, el dueño de la información (orador, periodista, político, etc.) puede valorar las intervenciones de los asistentes en las transmisiones en vivo, que por lo general, son echadas en poco, pero que si se hace un proceso de inteligencia artificial se puede dar valor a dicha información y así permitirle al usuario la observación de tendencias, percepciones, positividad o no de los comentarios, todo esto a través de un análisis de sentimientos. La información podrá ser evaluada parcial o totalmente. Si se emiten resultados parciales, estos corresponderán al corte de la información extraída en el momento, para el caso del resultado total dependerá del final de la transmisión, permitiendo mostrar los resultados en la misma transmisión, a partir de los porcentajes de evaluación del modelo, clasificados como Bad (malos), Neutral (neutrales) y Good (buenos), con lo cual se obtiene la percepción de los usuarios con respecto al tema tratado en la transmisión.

***Palabras clave* — Inteligencia artificial, análisis de sentimientos, streaming, procesamiento de lenguaje natural.**



## ABSTRACT

In the present, artificial intelligence processes and tools are being developed, which allow models to be applied to data, to obtain information that is not possible with conventional analysis tools, given the amount of data to be processed and its structure, it is necessary to adopt cutting-edge technologies to obtain the greatest possible benefit in the analysis of information. In the case of this work, it was proposed to take advantage of these technologies to give value to the interactions that take place in the chats, which users produce when writing comments on live broadcasts, with which, the owner of the information (speaker, journalist, politician) can evaluate the interventions of the attendees in the live broadcasts; which are generally overlooked, but if an artificial intelligence process is carried out, value can be given to said information and thus allow the user to observe trends, perceptions, positivity or not of the comments, all this through of a sentiment analysis. The information may be evaluated partially or totally. If partial results are issued, these will correspond to the cut of the information extracted at the time, in the case of the total result it will depend on the end of the transmission, allowing the results to be shown in the same transmission, through the model evaluation percentages, classified as Bad, Neutral and Good, thereby obtaining a perception of the users regarding the transmission.

***Keywords*** — **Artificial intelligence, sentimental analysis, streaming, natural language processing.**

## I. INTRODUCCIÓN

En los últimos tiempos hemos observado cómo han evolucionado las herramientas tecnológicas y cómo estas se han aplicado a muchas disciplinas, proveyendo soluciones a situaciones del día a día de la sociedad. En este sentido, es posible aprovechar dichas herramientas, con el fin de obtener su mayor provecho y a la vez responder las necesidades de información y de análisis. Muchas de estas herramientas se encuentran con modelos creados y entrenados, para su explotación.

Para nuestro caso, la inteligencia artificial (IA), permite utilizar técnicas, como es el caso de la aplicación del tratamiento del lenguaje natural.

Dada la evolución de los medios de comunicación y más específicamente las redes sociales, en donde los individuos no sólo reciben información, sino que también interactúan con este medio, enviando mensajes, imágenes, mail entre otros, este tipo de interacciones se presentan de diversas formas. Este trabajo se centra en el análisis de opiniones emitidas como mensajes de chat en transmisiones en vivo, las cuales, en muchas ocasiones, no son valoradas, demeritando este tipo de información que puede representar mucho valor para el cliente objetivo. Un escenario que se ha considerado es la emisión de “en vivos”, en donde se pueden conectar más de 50 personas y todas interactuando con el expositor, lo que hace que controlar los mensajes sea muy difícil y se pierda información valiosa. En el presente proyecto se pretende ofrecer un proceso que permita recolectar los datos, almacenarlos, y enviarlos a un modelo de inteligencia artificial, para que se realice un análisis de sentimientos, que nos muestre la positividad, negatividad o neutralidad de la opinión de los espectadores con respecto al tema tratado. Esta información puede servir para reorientar el tema del expositor o para saber si la intención sobre su público está logrando el objetivo. (Chanchí, & Córdoba., 2019) [4], indican que: La computación afectiva ha sido ampliamente difundida en el contexto de las redes sociales, con el fin de analizar las emociones y/o sentimientos que expresan los usuarios a partir de sus publicaciones.

El análisis de sentimientos le permitirá al usuario tener un informe en varios momentos de la transmisión, para verificar la evolución de dicha positividad o negatividad.

Para el desarrollo del presente trabajo de grado, se utilizará la plataforma de videos en vivo: YouTube, de la cual se extraerán los mensajes que se emiten en el en vivo y de allí se

tomará la información, para llevarla al modelo de análisis de sentimientos, que permitirá obtener los resultados de positividad o negatividad.

## II. PLANTEAMIENTO DEL PROBLEMA

No existe un sistema que permita clasificar las opiniones emitidas en el chat de un “en vivo” en YouTube para darle valor a esta información.

De acuerdo con (Chachí et al., 2020) [9], actualmente una de las formas empleadas para determinar la percepción de un sujeto a partir de su opinión, es mediante la aplicación de técnicas de análisis de sentimientos, ya que estas corresponden a un área del procesamiento del lenguaje natural, el cual tiene como objetivo identificar y evaluar el valor emocional de un texto, mediante su estructura, llevándolo a un nivel de clasificación positiva, neutral o negativa.

Además, (Salazar et al., 2020) [10], enfatizan en la predicción del posible efecto de un texto escrito en los usuarios, basado en las palabras clave del texto, a su vez se define un proceso para una identificación automática de su polaridad, determinando una connotación positiva o negativa.

Es importante resaltar que, (Cardoso et al., 2019) [12], nos enseñan otras clasificaciones para la aplicación del análisis de sentimientos a las opiniones, entre ellas tenemos: miedo, ira, asco, sorpresa, tristeza y felicidad. Además, tienen en cuenta un aspecto que considero muy importante y es la que en el proceso de clasificación se pueden detectar textos agresivos y a la vez pueden ser eliminados en la red de donde fueron originados.

En el caso de (Hernández & Gómez, 2014) [13], recalca que, la tarea básica del AS, es la definición de la polaridad de sentimientos, por lo que, en el análisis de citas bibliográficas, también es posible hallar tendencias en forma negativa o positiva.

Un aspecto importante y que debe ser tenido en cuenta tiene que ver con el ser y que tan afectado o no puede estar, por una situación determinada, por ejemplo, la afectación del Covid-19, (Flores et al., 2023) [19], desarrollaron una aplicación de inteligencia artificial con redes neuronales, para identificar el estado emocional de los estudiantes en el aula virtual en tiempo real, para mostrarle al docente la percepción de sus estudiantes durante la clase virtual.

Por último, (López, 2015) [11], indica que los resultados de su investigación demuestran que el análisis de sentimientos es una disciplina, que puede ser útil para la toma de decisiones, tanto para las compañías, como para los individuos, con la observación de que es susceptible a ser mejorada para aprovechar las grandes cantidades de datos emitidas en opiniones en las redes sociales.

### III. JUSTIFICACIÓN

En la actualidad se cuentan con muchas herramientas que permiten aplicar nuevas tecnologías sobre los datos. Estas herramientas, como la inteligencia artificial, puede ayudar a analizar grandes cantidades de datos, que para una persona sería imposible; ante dicha imposibilidad, se pueden desarrollar procesos que ayuden a comprender las tendencias de un discurso, de una argumentación, de un producto, o de un tema específico. Estas tendencias pueden indicar en qué medida fueron comprendidos los contenidos por los oyentes, es decir, si el tema tratado generó positividad, neutralidad o negatividad. Llegar a estos resultados hace que se enriquezca el conocimiento acerca de lo que los oyentes piensan de lo que se dice y, además, que en medio de las transmisiones se pueda corregir el discurso para mejorar la percepción, hace que se le dé valor, importancia y reconocimiento a las opiniones de personas que sacan de su tiempo para escuchar los en vivo de YouTube.

## IV. OBJETIVOS

### *A. Objetivo general*

Construir y poner a disposición una aplicación web, para hacer análisis de sentimientos en tiempo real, a las intervenciones vía chat, para las transmisiones en la plataforma YouTube.

### *B. Objetivos específicos*

Seleccionar el mejor modelo de aprendizaje automático para análisis de sentimientos, basado en arquitecturas de redes neuronales modernas.

Desarrollar los pipelines para extracción, transformación y carga de los mensajes de chat generados en las transmisiones en tiempo real.

Implementar el backend de la aplicación para usar el modelo de análisis de sentimientos vía API.

Implementar el frontend de la aplicación para consumir el servicio de clasificación de mensajes de chat, para transmisiones en tiempo real, con una interfaz amigable y que presenta resultados en forma de estadísticas descriptivas.

Desplegar la aplicación web en un ambiente productivo usando Google Cloud Platform.

## V. HIPÓTESIS

Es posible darle valor a los mensajes que se emiten en los chats de los en vivo de YouTube a partir del análisis de sentimientos.

## VI. MARCO TEÓRICO

Saber qué se está hablando de nuestros productos, de nuestras opiniones o expresiones orales, de nuestros lanzamientos y en general de todo el material que se expone en las redes, se convierte en un reto y en una oportunidad para explotar esta información, ya que, dada su subjetividad y la cantidad de información emitida, se hace necesario aplicar tecnologías que nos permitan obtener su valor. Las opiniones dejan de ser un simple comentario y se convierten en una base de conocimiento para la toma de decisiones.

(Dubiau & Ale, 2013) [6], afirma que, "El objetivo principal del análisis computacional de sentimientos consiste en determinar la actitud de un escritor ante determinados productos, situaciones, personas u organizaciones (target); identificar los aspectos que generan opinión (features); quien las posee (holder); y cuál es el tipo de emoción (me gusta, me encanta, lo valoro, lo odio) o su orientación semántica (positiva, negativa, neutra)."

Además, (Bing, 2010) [7], indica que, "Este modelo primero descubre los objetivos sobre los cuales las opiniones tienen una expresión en una oración, y luego determina si las opiniones son positivas, negativas o neutral. Los objetivos son objetos y sus componentes, atributos y características. Un objeto puede ser un producto, servicio, individuo, organización, evento, tema, etc. Por ejemplo, en una oración de revisión de producto, identifica las características del producto que han sido comentadas por el revisor y determina si los comentarios son positivos o negativos. Por ejemplo, en la oración "La duración de la batería de esta cámara es demasiado corta", el comentario es sobre "la duración de la batería" del objeto de la cámara y la opinión es negativa. Muchas aplicaciones de la vida real requieren este nivel de análisis detallado porque para realizar mejoras en el producto uno necesita saber qué componentes y/o características del producto son del agrado y desagrado de los consumidores."

En este orden, las opiniones que se emiten nos pueden ayudar a determinar ciertos patrones, que los clientes observan en los productos o los oyentes expresan en alguna argumentación.

Estos patrones se pueden traducir como tendencias de aceptación, negación o neutralidad y estas a su vez nos brindan una información potencialmente valiosa, por ejemplo, para tomar acciones correctivas, identificar problemas de calidad en los productos, mal



entendimiento de un discurso argumentativo o por el contrario, una excelente aceptación de los productos y un entendimiento y agrado con un discurso.

(Becerra, 2017) [8], resalta que: Al analizar los grandes volúmenes de datos generados en redes sociales sobre diferentes temáticas, se pueden obtener descubrimientos valiosos. Advirtiendo que, si se realizan estas tareas manuales resultaría muy costoso, dada la cantidad de información, por lo tanto, utilizar los sistemas de análisis de sentimientos han resultado ser de gran utilidad, para obtener la percepción general de estos temas.

Con respecto a la recolección de datos, es importante precisar, que los datos corresponden a opiniones, motivadas por un tema específico, el cual genera opinión, por lo tanto, el público será quien defina si está o no de acuerdo con una postura, ideología o planteamiento.

Para hablar de temas de recolección de datos, es importante remitirnos a lo que afirmó: (Torres et al., 2019) [1]: “Una investigación es científicamente válida al estar sustentada en información verificable, que responda lo que se pretende demostrar con la hipótesis formulada. Para ello, es imprescindible realizar un proceso de recolección de datos en forma planificada y teniendo claros objetivos sobre el nivel y profundidad de la información a recolectar.”

En este orden, los datos crudos corresponden a la información que tendremos disponible en nuestra fuente, que en nuestro caso se encuentra en los datos no estructurados de los mensajes o “chats” que se están registrando en las transmisiones en vivo de YouTube y que son precisamente las interacciones de los usuarios con el expositor o los expositores, los cuales hablan de un tema determinado. Estos mensajes son generados de forma espontánea y corresponden a la opinión de los usuarios. Es importante que en los procesos de limpieza se consideren los datos que son emitidos desde “bots” y que pueden sesgar la positividad o negatividad de lo que se percibe en el chat.

Otro punto importante son las librerías de Python, como es el caso de pychat, para adquirir y preparar los datos, en este caso, dicha librería nos permite acceder al origen de los datos y a la captura de la información, específicamente de un video en vivo de YouTube y que tenga habilitados los chats. Es de aclarar que en el presente desarrollo no se aplicarán técnicas de calidad, por ejemplo, el tratamiento de mensajes repetidos, que, en muchos casos,

corresponden a bots que repiten mensajes, con la finalidad de sesgar la positividad o negatividad de los mensajes emitidos.

Otra técnica de recolección de datos y que puede ser aplicada a los videos finalizados de YouTube, es el concepto de “webscraping”, (López & Gonzales, 2021) [3], afirmaron que: Google Play Store representa diferentes tipos de aplicaciones de Android, la mayoría de las cuales son gratuitas. Así que toda la sección de comentarios es un reflejo de las opiniones de personas de diferentes gustos y mentalidades. Se construyó un scraper web (extraer información de páginas web de forma automatizada) para el propósito de recopilación de datos que proporcionó información crítica de revisión: nombre de usuario, comentario y calificación de una respectiva aplicación.

En cuanto a la preparación de la información, (Llorente, 2019) [2], afirma que: “Las expresiones informales del lenguaje natural como las que se dan en los tweets que componen el conjunto de datos del sistema no suelen ser correctas ortográficamente y es habitual que hagan uso de abreviaturas propias de los microtextos y palabras pertenecientes a una jerga o argot (variante del lenguaje estándar utilizada por un grupo social determinado). Es por ello que deben ser tratadas en un proceso de normalización”.

Además (López & Gonzales, 2021) [3], afirmaron que: “El siguiente paso es preparar los datos que se han obtenido, quitando las palabras innecesarias que generan ruido en los datos obtenidos 14, se omitió las palabras con una longitud menor a 2, errores ortográficos, variaciones léxicas, emoticones, además de filtrar algunos comentarios que estaban escritos en otros idiomas que no sea el español, este proceso se realizó de manera automatizada utilizando la librería NLTK (Natural language toolkit) de Python”.

Para nuestro caso, no es necesaria una estructuración de los datos, ya que el modelo de análisis de sentimientos tiene la capacidad de procesar la información, con oraciones completas, es decir, el texto de un chat escrito.

De otro lado, tenemos el pronóstico, el cual consiste en el grado de satisfacción o rechazo que tiene el asunto que se esté tratando en el momento. Para este caso se elige el modelo con mejor rendimiento y que será utilizado en el procesamiento de la información recolectada.

Un ejemplo de aplicabilidad de un modelo de análisis de sentimientos nos lo enseña: (Gil, 2018) [14], quien realizó un análisis sobre el impacto social de proyectos de vivienda, en el programa del gobierno: un techo para mi país, en donde se emplearon 1000 comentarios de

la red social Twitter. Lo cual permitió ver que la mayoría de los comentarios tienen una percepción positiva el 77%, mientras que los muy positivos tuvieron un 10%, los neutrales con un 8% y los negativos con el 5%. Concluyendo que el programa tiene un alto porcentaje de aceptación.

Además (Baviera, 2016) [16], afirma que, los mensajes que se emiten en redes sociales, como es el caso de Twitter, ofrecen posibilidades para evaluar las corrientes de opinión y puntualiza que, las grandes cantidades de información requieren de herramientas con la capacidad suficiente para procesarla, sin perder la fiabilidad

(Vélez, 2015) [15], nos muestra que también podemos aplicar análisis de sentimientos sobre posconflictos, para realizar una valoración de los diferentes puntos de vista, en el contexto de un posconflicto, con lo cual, sus opiniones, se pueden clasificar en sentimientos positivos, negativos o neutros. Muy importante resaltar que los perfiles que se analizaron corresponden a guerrilleros, políticos, medios de comunicación y generadores de opinión.

Es importante resaltar que un modelo de análisis de sentimientos tiene aplicabilidad en muchos campos, uno de ellos es la entrevista para selección de personal, (Martínez & Mateus, 2019) [17], realizaron un análisis de sentimientos a las entrevistas laborales, adicionaron técnicas de fijación de mirada y luego seleccionaron el algoritmo de aprendizaje de máquina supervisado con redes neuronales artificiales, de tal forma que crearon una herramienta para apoyar el proceso de selección de personal, mediante la interpretación de los patrones oculares.

Por otro lado, (Arcila et al., 2016) [18], describen en su artículo, cómo se aplican técnicas de análisis de datos textuales a gran escala (Big-data), como es el caso del análisis automatizado de contenidos, la minería de datos, el aprendizaje automatizado, el modelamiento de temas y el análisis de sentimientos. Con lo cual puede ser utilizado para generar conocimiento en ciencias sociales.

Para finalizar, (Jiménez et al., 2014) [20], nos muestran que existen muchos trabajos aplicados a la minería de opiniones (MO), llamada análisis de sentimientos (AS), la cual se centra en detectar información subjetiva en un texto y su posterior clasificación, pero por otro lado se muestran como desafíos el tratamiento de la negociación, de la ironía y del sarcasmo, el análisis a nivel del aspecto, entre otros.

## VII. METODOLOGÍA

En esta sección se presenta el proceso llevado a cabo para el desarrollo de la herramienta que permite darle valor a las opiniones emitidas por los usuarios que interactúan con él en vivo de YouTube.

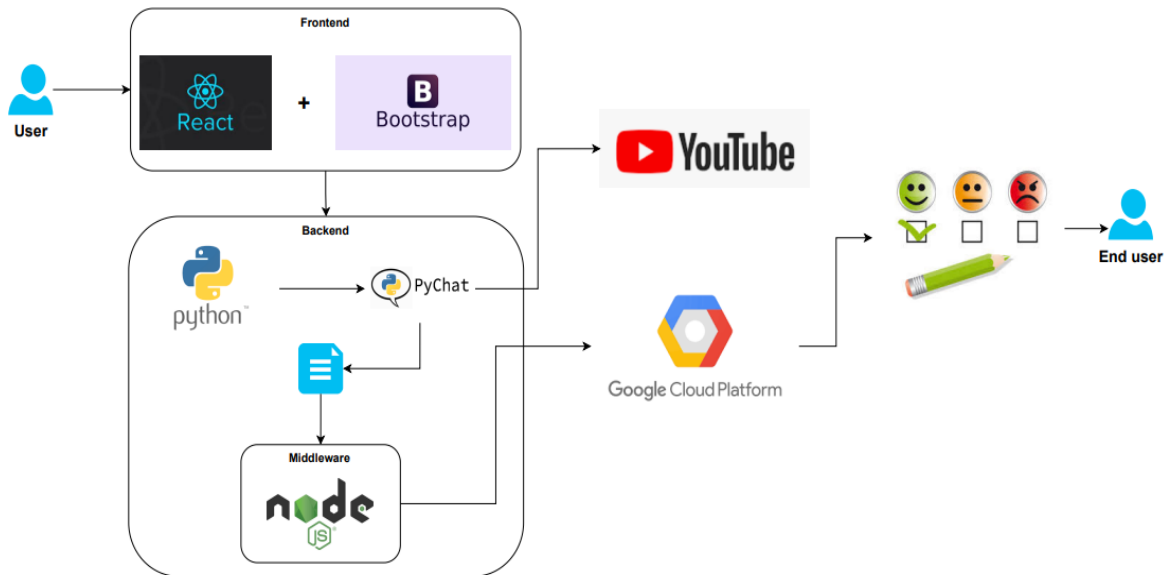


Fig. 1. Herramientas de desarrollo de software

Nota: fuente elaboración propia

En la construcción de la App y para darle valor a las opiniones, a través de un modelo de análisis de sentimientos, como primero, se establecieron todas las necesidades que requerían de herramientas, librerías, IDE, para, no solo desarrollar una aplicación, sino también para el despliegue, por lo cual se resolvieron preguntas como:

1. ¿Como se pueden recolectar los chats de un video en vivo de Youtube? Se tomó la librería Pychat de Python y a través de Google colab, se realizaron pruebas exitosas de su funcionamiento, en donde, al colocar el link del video en Youtube, se pudieron obtener los mensajes del en vivo, aunque llegaba más información de la que se necesitaba, por ejemplo, id\_mensaje, fecha, usuario, mensaje, de lo cual sólo se dejó el

campo del mensaje, ya que correspondía al dato necesario en el proceso del modelo de análisis de sentimientos.

2. ¿Qué modelo de análisis de sentimientos puedo utilizar para procesar los chats? En este caso se utilizó un modelo de Huggin Face: pysentimiento.

Huggin Face, desarrolla tareas de análisis de sentimientos y es un referente en la comunidad de NLP, ya que contribuye a la investigación y al desarrollo de modelos de lenguaje avanzado, ofreciendo una variedad de modelos pre-entrenados, al igual que herramientas y recursos para trabajar con NLP.

Pysentimiento es una biblioteca de Python, la cual fue desarrollada por Huggin Face, centrada en el análisis de sentimientos en texto, utilizando modelos pre-entrenados de Huggin Face. Con lo cual, utilizando pysentimiento, se puede determinar el tono emocional de un texto, es decir, si es positivo, negativo o neutral.

Por todo lo anterior y dada la capacidad de realizar procesos de análisis de sentimientos, fueron razones suficientes para seleccionar este modelo.

Se realizaron pruebas exitosas de la ejecución del modelo en Google colab:

```
from pysentimiento import create_analyzer
analyzer = create_analyzer(task="sentiment", lang="es")
msgIntro = ["Lo que estas diciendo es pura basura", "yo sí estoy de acuerdo con eso, me parece bien."]
msgBody = ["Texto aquí"]
msgConclusions = ["Texto aquí"]
#analyzer.predict("Qué gran jugador es Messi")
resultsIntro = analyzer.predict(msgIntro)
resultsBody = analyzer.predict(msgBody)
resultsConclusions = analyzer.predict(msgConclusions)
# returns AnalyzerOutput(output=POS, probas={POS: 0.998, NEG: 0.002, NEU: 0.000})
```

Fig. 2. Validación del modelo de análisis de sentimientos

Nota: fuente elaboración propia

3. ¿Teniendo resuelto el tema del modelo y la forma en que se recolectarían los mensajes, cómo puedo integrar esto y desarrollar la App? Luego de probar la librería Pychat de Python y el modelo: pysentimiento, se inició con el desarrollo de la aplicación en Node js y se comenzaron a integrar cada uno de los componentes del desarrollo, utilizando metodología Agile, para toda la gestión del proyecto:

Para la gestión del proyecto se utilizó la plataforma Trello y se definieron cada uno de los sprints, para el cumplimiento del desarrollo de la App:

**Sprint 0:**

US - Identificar a grandes rasgos las funcionalidades del sistema, priorizarlas por el valor que generan

Se puede usar cualquier tipo de estructura para su especificación: historia de usuario, casos de uso, escritura formal, by example, BDD

Se deben llevar a la columna (Product back log) del tablero y priorizarlas (mayor valor primero)

US - Configuración inicial del proyecto Back-End

Criterios de aceptación:

C1- Crear el ambiente de desarrollo, que incluya todas las herramientas y tecnologías necesarias para el desarrollo y prueba del software.

C2 - Configurar el servidor, donde se ejecutará el Back-End y los servicios necesarios para su correcta ejecución.

C3 - Especificar las dependencias y librerías que deben estar configuradas y disponibles en el proyecto Back-End, necesarias, para el correcto funcionamiento del sistema.

US-Configuración inicial del proyecto Front-End

Se debe subir al repositorio respectivo la primera versión en la rama master

Incluir las buenas prácticas y estándares visto en la curva de aprendizaje: I18N, configuración del data source, manejo logs, excepciones.

Criterios de aceptación:

C1 - Configurar el entorno de desarrollo, para este caso se utiliza el IDE visual studio code, configurar el control de versiones git y gestor de paquetes npm.

C2 - Configurar el marco de trabajo (React)

C3 - Crear archivo de estilos inicial CSS.

## US-Project Inception

### Criterios de aceptación

C1 - Se debe subir al repositorio respectivo, la primera versión en la rama master

C2 - Incluir las buenas prácticas y estándares visto en la curva de aprendizaje

C3 - Usar templates que ya estén desarrollados

C4 - La estructura básica del portal debe incluir: una página central donde se va a tomar la URL y un botón para ejecutar el proceso y una barra inferior (footer), para mostrar los resultados.

## **Sprint 1:**

US - Crear el servicio: `start_monitoring`, para iniciar el monitoreo y recolección de los chats de la UTL configurada.

### Criterios de aceptación

C1 - El servicio debe cumplir con los requisitos funcionales establecidos, es decir, que al iniciar el servicio de pychat, para tomar los mensajes del en vivo, se puedan capturar y llevar a un archivo.txt

C2 - El servicio debe ser capaz de manejar la carga esperada y mantener un rendimiento óptimo. Para este caso debe leer todos los chats emitidos en el en vivo, en tiempo real.

C3 - El servicio debe cumplir con los requisitos de seguridad establecidos para proteger los datos y prevenir posibles brechas de seguridad, es decir, garantizar tanto la lectura, como la escritura del dato en el archivo.txt

C4 - El servicio debe ser confiable y estar disponible en todo momento, se debe garantizar la capacidad de recuperación ante fallos y la implementación de mecanismos de respaldo y recuperación de datos.

C5 - El servicio debe poder integrarse correctamente con otros sistemas o servicios en el ecosistema de la aplicación, es decir, garantizar la capacidad de consumir y proporcionar datos en los formatos requeridos, la compatibilidad con los protocolos de comunicación establecidos, y la correcta interacción con otros servicios y APIs

C6 - El servicio debe ser fácil de mantener y permitir futuras expansiones o modificaciones

C7 - El servicio debe ser sometido a pruebas exhaustivas para garantizar su correcto funcionamiento, para este caso se realizarán únicamente las pruebas unitarias.

US - Crear el servicio: stop\_monitoring, para detener el servicio de monitoreo y recolección de datos

Criterios de aceptación

C1 - El servicio debe cumplir con los requisitos funcionales establecidos, es decir, debe detener el servicio de pychat, por lo tanto, debe detener la captura de los mensajes del en vivo.

C2 - El servicio debe ser confiable y estar disponible en todo momento, se debe garantizar la capacidad de recuperación ante fallos.

C3 - El servicio debe poder integrarse correctamente con otros sistemas o servicios en el ecosistema de la aplicación, es decir, la correcta interacción con otros servicios y APIs

C4 - El servicio debe ser fácil de mantener y permitir futuras expansiones o modificaciones

C5 - El servicio debe ser sometido a pruebas, para garantizar su correcto funcionamiento, para este caso se realizarán únicamente las pruebas unitarias.

US - Crear el archivo de requerimientos, para la gestión de dependencias

Criterios de aceptación

C1 - Crear el archivo: Requerimientos.txt en la carpeta API\_CHATS\_LIVE

C2 - Adicionar las siguientes dependencias, para garantizar su correcta actualización: fastapi==0.95.1, pandas psutil pychat==0.5.5, uvicorn[standar]==0.21.1, pysentimiento==0.4.2, tensorflow==2.11.0, scikit-learn==1.2.2

C3 - Ejecutar el siguiente código, para garantizar la correcta actualización de las dependencias: pip install -r api/requirements.txt

C4 - Tomar evidencia de actualización de las dependencias.



## Sprint 2

US - Desarrollar una función que valide los argumentos obligatorios en una url

Criterios de aceptación

C1 - Función creada con la recepción y validación de la url, anexo parte del código de creación: 

```
if __name__ == "__main__":  
    parser =  
    argparse.ArgumentParser("ChatMonitor")  
    parser.add_argument("--url", type=str,  
    required=True)  
    args = parser.parse_args()  
    monitor(url=args.url)
```

C2 - Se debe cumplir con los requisitos funcionales establecidos, es decir, la función de validar que la url cumpla con el estándar de una dirección url.

C3 - La función debe ser confiable y estar disponible en todo momento, se debe garantizar la capacidad de recuperación ante fallos.

C4 - La función debe poder integrarse correctamente con otros sistemas o servicios en el ecosistema de la aplicación, es decir, la correcta interacción con otros servicios y APIs

C5 - La función debe ser fácil de mantener y permitir futuras expansiones o modificaciones

C6 - La función debe ser sometida a pruebas, para garantizar su correcto funcionamiento, para este caso se realizarán únicamente las pruebas unitarias.

US - Crear la función: `monitor_chat`, la cual recibirá la url, para recibir los chats que se generan en vivo

Criterios de aceptación

C1 - Guardar los mensajes emitidos en el en vivo. en un archivo .txt

C2 - Utilizar la librería `pychat` de Python

US - Crear la función `save_chat`, para guardar los mensajes recibidos en un archivo: .txt

Criterios de aceptación

C1 - La función debe almacenar los mensajes que se emiten en el en vivo, en un archivo plano.

C2 - Verificar que al archivo creado se le concatene un id y fecha de ejecución

US - Crear la función: `open_f_data`, para abrir el archivo generado en la función `save_chat` y recuperar los chats guardados.

Criterios de aceptación

C1 - La función debe recuperar los mensajes que se guardaron, en el archivo plano.

C2 - Verificar que sean recuperados todos los chats que fueron emitidos en el en vivo.

### **Sprint 3**

US - Diseñar un prototipo del Front End

Criterios de aceptación

C1 - Diseñar una presentación inicial del front-end, dónde se realizará la captura de la URL

C2 - Incluir un botón para iniciar el proceso de captura, almacenamiento de mensajes y ejecución del modelo de análisis de sentimientos.

C3 - Diseñar en el área footer, el panel de resultados del modelo donde se muestre el porcentaje de mensajes malos, neutrales y buenos (Bad, Neutral and Good)

US - Crear el servicio: `analyze_chat`, para enviar los mensajes recolectados al modelo de análisis de sentimientos

Criterios de aceptación

C1 - Tomar los mensajes guardados en el archivo `.txt`

C2 - Enviar los mensajes al modelo de sentimientos, para realizar la predicción

C3 - Mostrar los resultados devueltos por el modelo de sentimientos

## **Sprint 4**

US - Realizar despliegue de la solución

Criterios de aceptación

C1 – Crear la imagen del proyecto en Docker, para realizar el despliegue en GCP

C2 – Instalar la máquina virtual en GCP

C3 - Crea una instancia de Google Compute Engine

C4 - Instalar Docker en la instancia y construir la imagen de Docker que se quiere desplegar en la instancia.

C5 - Ejecutar el contenedor utilizando el comando docker run

### **Inception:**

#### **1. Descripción de la necesidad**

En las transmisiones en vivo, se genera información en forma de chats, con las opiniones que expresan las personas que se conectan. Estas opiniones pueden ser exponenciales y no permiten ser revisadas o leídas en medio de la transmisión, motivo por el cual se hace necesario y es importante darle un valor a dicha información, con modelos de inteligencia artificial, de tal forma que le den una interpretación a lo que se está hablando.

TABLA 1  
IMPACT MAPPING

Meta	Actor	Impactos	Entregables
Gestionar el 100% de los chats emitidos en un en vivo de YouTube y la ejecución de un modelo de inteligencia artificial, a través de una aplicación web al 30/05/2024	Expositor	Recolección los chats emitidos en el en vivo	Acceso a la aplicación
			Registro de la url en la aplicación
			Iniciar el proceso de almacenamiento de los chats y ejecución del modelo
			Detener el proceso de almacenamiento y/o ejecución del modelo
		Aplicar un modelo de inteligencia artificial (análisis de sentimientos)	Visualizar los resultados obtenidos en la ejecución del modelo de análisis de sentimientos

Visual story mapping

Recolección de chats emitidos en vivo

- Acceso a la aplicación
- Ingresar la URL del video de YouTube en vivo
- Iniciar proceso de almacenamiento (txt) de los chats
- Ejecución del modelo
- Detener el proceso de almacenamiento y/o ejecución del modelo

Aplicación del modelo de inteligencia artificial (Python)

- Visualización de resultados (% Positivos, % Negativos, % Neutros)

## 2. Priorización

TABLA 2  
PRIORIZACIÓN

Must	Should	Could	W'ont
Acceso a la aplicación			
Ingresar la url del video de YouTube en vivo			
Iniciar proceso de almacenamiento (txt) de los chats			
Ejecución del modelo			
Detener el proceso de almacenamiento y/o ejecución del modelo			
Visualización de resultados (% Positivos, % Negativos, % Neutros)			

## 3. Descripción de la solución

Como solución se planteó desarrollar una aplicación web, utilizando el framework de Angular para el Font-End y para el Back-End, en Next y React. Se utilizó las capas: Vista-Lógica-Persistencia, para el Back-End.

El objetivo de la aplicación fue, permitir recolectar los chats de un video en vivo de YouTube y aplicar un modelo de inteligencia artificial de análisis de sentimientos, que este permitiera observar la positividad, neutralidad o negatividad de la información que se ingresa en los chats.

El Front-End, no tiene roles, pues, no se desarrolló un módulo para realizar dicha gestión, en una versión posterior se implementará, pero funcionalmente se desarrolló la solución para realizar una sola aplicación del modelo a la vez. En un futuro y con el refinamiento des esta versión inicial, se estará disponibilizando, para realizar peticiones múltiples.

Página de acceso y procesamiento de datos

Mediante esta página el usuario podrá ingresar el link del video en vivo de YouTube y dará inicio al proceso de recolección de los chats, posterior a este proceso, se realizarán ejecuciones por demanda, cada vez que el usuario quiera evaluar los chats recolectados.

Una vez finalice el en vivo, se realizará un procesamiento completo de chats, para aplicar el análisis de sentimientos con todos los chats emitidos en el en vivo.

El usuario contará con la opción de detener el procesamiento del modelo.

#### **4. Plan de trabajo y cronograma**

A continuación, se describen las diferentes actividades, las cuales fueron desarrolladas en el presente proyecto y el cronograma, priorizado en los diferentes releases (entregas funcionales).

La duración del proyecto fue de aproximadamente 8 semanas (2 meses y medio), dividido en las siguientes entregas funcionales:

- Release 0 - 2 semanas – sprint review 0 – Abril 25 - 2024
  - Configuración Trello
  - Especificación backlog inicial
  - Configuración inicial proyectos front
  - Configuración inicial proyectos back
  
- Release 1 (R1): 2 semanas - sprint review 1 – Mayo 9 – 2024
  - Funcionalidad 5: Archivo de requerimientos - gestión de dependencias
  - Funcionalidad 6: Servicio start\_monitoring
  - Funcionalidad 7: Servicio: stop\_monitoring
  
- Release 2 (R2): 2 semanas – sprint review 2 – Mayo 23 – 2024
  - Funcionalidad 1: monitoring\_chat
  - Funcionalidad 2: save\_chat
  - Funcionalidad 3: open\_f\_data
  - Funcionalidad 4: \_\_name\_\_
  
- Release 3 (R3): 2 semanas – sprint review 3 – Junio 6 - 2024
  - Funcionalidad: servicio analyze\_chat
  - Diseñar un prototipo del Front-End
  - Desarrollo del Front-End - captura de la url del en vivo de youtube – inicio y detención del proceso



---

	Entrega código fuente																			
	Entrega oficial																			

## 5. Kick-Off

Se establecieron las siguientes reglas de trabajo:

Marco de trabajo: Scrum

Ceremonias y tiempos:

- Daily Scrum (todos los días de 8 am a 8:15 am) máximo 15 minutos - (que se hizo ayer, que se va a hacer hoy y que impedimentos se tienen)
- Sprint review: Un día antes del vencimiento del sprint, se muestra el producto desarrollado a los PO, PM y a todo el equipo, según el cronograma de actividades. Una vez desarrolladas las funcionalidades del sprint (Sprint execution)
- Retrospective: Se realiza una reunión entre el PM y el equipo, para realizar una retroalimentación de lo que se realizó bien, los problemas que se presentaron y se establece un plan de mejoramiento. Esta reunión se realiza el último día del sprint.
- Sprint Plannig: Se actualizan las tareas planeadas del siguiente sprint. Esta reunión se realiza el último día del sprint.

## 6. Metodología de trabajo

Para el desarrollo del proyecto se usó el marco de trabajo ágil SCRUM, el cual nos permitió desarrollar el producto de forma incremental y evolutiva. Se realizaron iteraciones, llamadas Sprint. Este proyecto estimó realizarse en 3 sprints con una duración de 15 días cada uno.



### Equipo de trabajo

- Product Owner: Conocedor del negocio, definición y priorización de los requerimientos. (Alexander Uribe G)
- Scrum Master: Encargado de alinear y establecer los canales de comunicación en todo el equipo, incluyendo los representantes del cliente, remover impedimentos y apoyar metodológicamente para lograr los resultados. (Alexander Uribe G)
- Team: Personal encargado de realizar cada una de las actividades especificada en cada reléase, tales como, desarrolladores, arquitecto, diseñadores, líder de infraestructura y personal de negocio.
  - 1 desarrollador
  - 1 tester
  - 1 scrum Master

### Diagrama de componentes

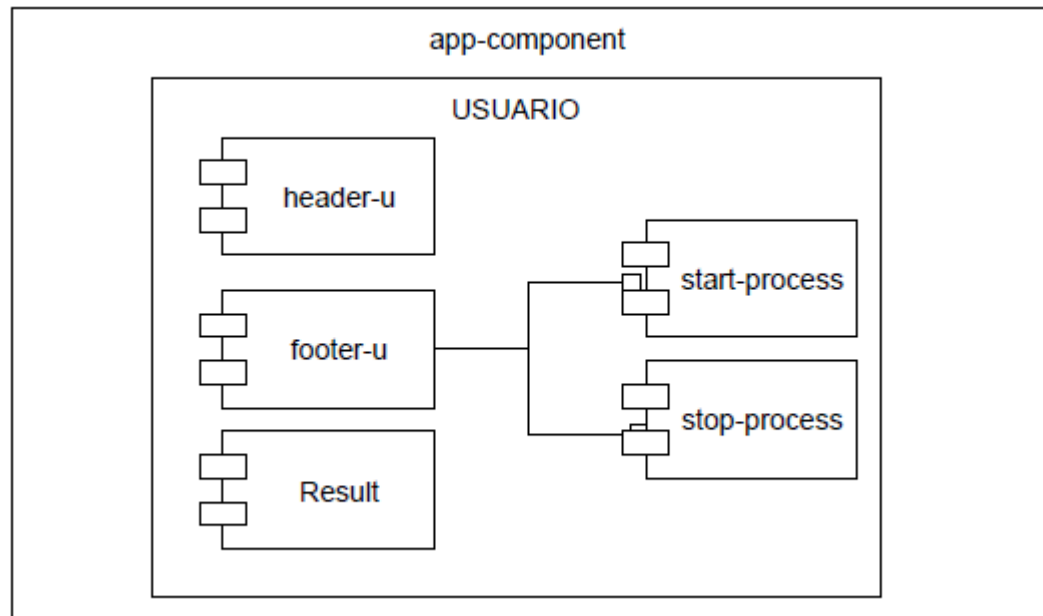


Fig. 3. Front-End (Single Page Aplication)

Nota: fuente elaboración propia

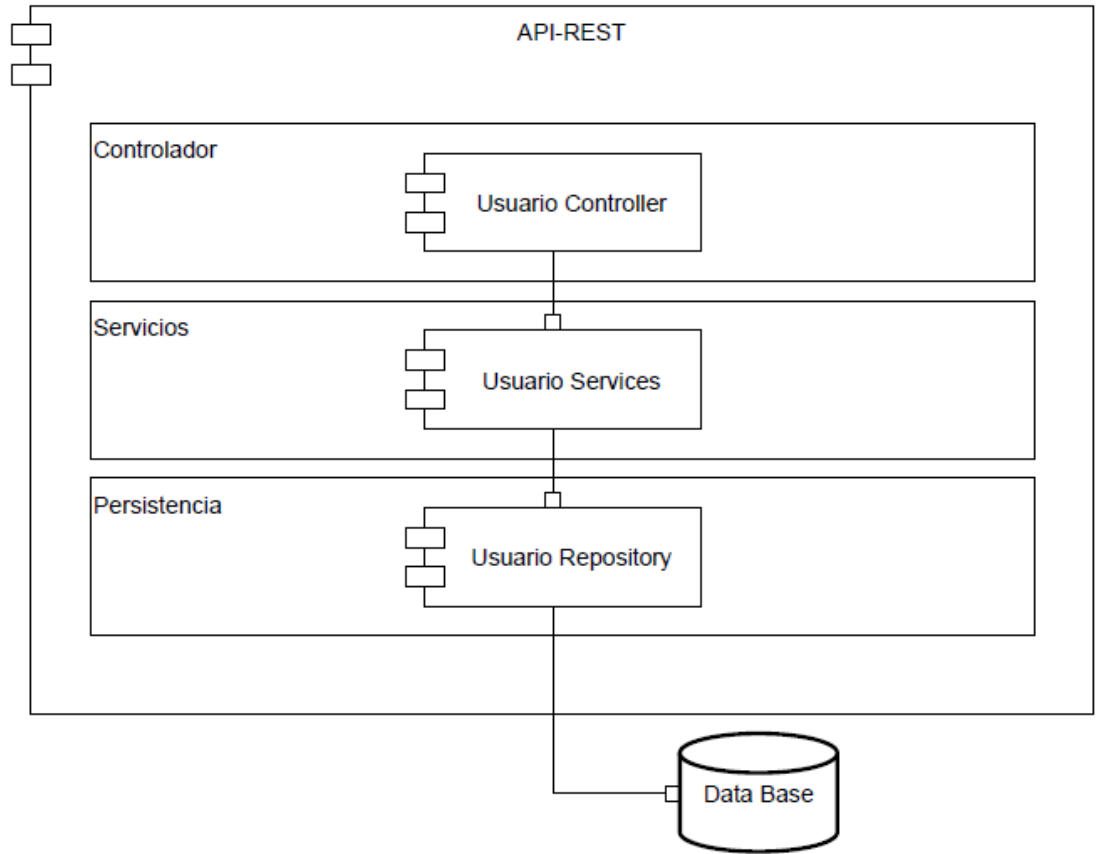


Fig. 4. Back-End (Api-Rest)

Nota: fuente elaboración propia

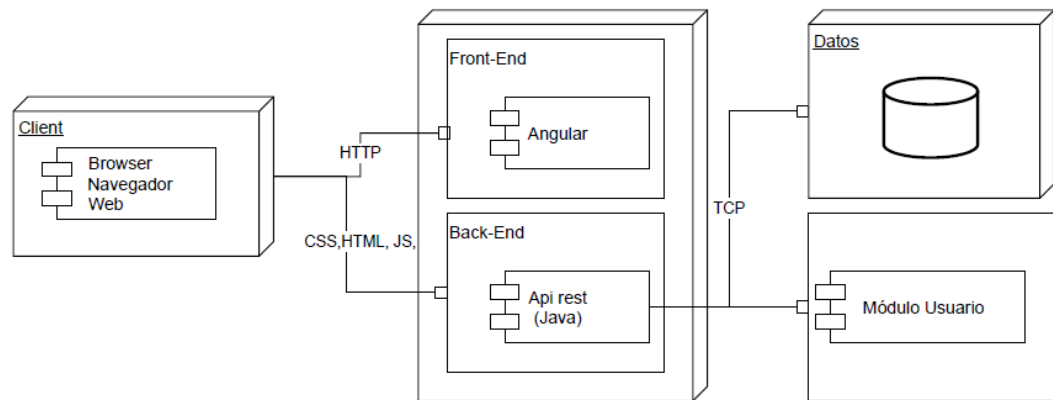


Fig. 5. Diagrama de despliegue

Nota: fuente elaboración propia

## Prototipo de la solución

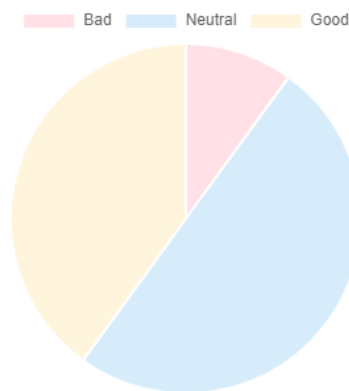
# Welcome to Live Chat Monitoring

Start monitoring by writing and YouTube Live URL

Fig. 6. Registro de la URL

Nota: fuente elaboración propia

## Chart



**Bad → 10%**

**Neutral → 50%**

**Good → 40%**

Fig. 7. Panel de resultados (procesamiento de datos)

Nota: fuente elaboración propia

4. ¿Cómo se recolectaron y se procesaron los datos? Los datos fueron recolectados y procesados de la siguiente manera:

### **1. Origen de la información**

La recolección de la información, que para nuestro caso correspondió a los mensajes emitidos por los usuarios que se conectan a las transmisiones en vivo de la plataforma Youtube, se tomaron los chats que se estaban escribiendo en el momento en que se inició el monitoreo del en vivo. De esta forma, fue extraída toda la información “chats”, que se generaron en el video. A través del lenguaje de programación Python se realizó esta tarea, de donde se puede tener acceso al link del video de YouTube y de allí; con algunas librerías, como es el caso de pychat, podemos descargar dicha información.

(Hernández, 2021) [5], dice: Es de buena práctica realizar una exploración de los datos antes de comenzar con el preprocesamiento y el entrenamiento de los mismos. Es por eso, que antes de elegir un dataset, se han estudiado diversos ejemplos que reunieran las características necesarias para poder extraer conocimientos y distribuciones. Finalmente se ha optado por el conjunto de datos de la Universidad de Stanford. Este conjunto de datos cuenta con 1.6 millones de tweets en inglés, con una buena limpieza, usado en diversos proyectos exitosos de este estilo y están categorizados en positivos y negativos.

### **2. Procesamiento de la base de datos**

Los datos que fueron recolectados desde el video de YouTube en el en vivo, fueron datos no estructurados y correspondieron a las opiniones emitidas por los usuarios que interactuaban a través del chat. Estas opiniones, fueron expresadas en oraciones que buscaban dejar constancia de un apoyo, un rechazo o una neutralidad. Dicha recolección fue realizada con la librería de Python “Pychat”, con la cual es posible y a través del link del video, tomar los mensajes registrados en el momento de establecer la conexión, estos mensajes fueron almacenados en un archivo de texto (txt), con la información digitada por el usuario. Dicho almacenamiento se va acumulando a medida que se recibían los mensajes.

### **3. Implementación del modelo clasificador de análisis de sentimientos**

Después de haber recolectado y almacenado los datos, continuamos con la selección de un modelo que nos permitió responder al objeto de nuestro estudio: análisis de sentimientos.

(Hernández, 2021) [5], dice que: Una vez analizados los datos, realizado el preprocesamiento y el procesamiento de los mismos, es el momento de introducirlos dentro de un modelo de red neuronal predictivo. En el apartado del Estado del Arte, se estudian los diferentes algoritmos que son aplicables a un problema de estas características y se comenta el por qué se ha elegido un modelo de memoria recurrente a corto plazo LSTM.

### **4. Despliegue de la aplicación**

El despliegue de la aplicación corresponde a la forma en que se mostrarán los resultados de procesamiento de los datos en el modelo de sentimientos, para este caso se eligieron herramientas de desarrollo como Nodejs, para el tema de front end y Next para el backend. Se creó una imagen en Docker, para realizar el despliegue tanto de las dependencias, como de la aplicación. En Google Cloud Platform (GCP), las imágenes de Docker se pueden desplegar y administrar utilizando Google Kubernetes Engine (GKE) o Google Compute Engine (GCE). En nuestro caso la imagen de Docker se instaló en Google Compute Engine (GCE), para lo cual se siguieron los siguientes pasos: primero, se creó una instancia de Google Compute Engine; luego se conectó a la instancia a través de SSH; posteriormente se instaló Docker en la instancia y se descargó la imagen de Docker que se desplegó en la instancia, y por último, se ejecutó el contenedor utilizando el comando docker run. Con estas acciones se habilitó la aplicación en la nube con la URL: <http://34.16.130.26/>

## VIII RESULTADOS

En esta sección se presentan los resultados del sistema desarrollado, basado en microservicios reactivos, para llevar a cabo el análisis de sentimientos que valore la información que se itera en los mensajes de los vídeos emitidos en vivo de YouTube.

El sistema recolecta los mensajes, los guarda y los procesa en el modelo, en el momento en el que el usuario considere su evaluación, para ello cuenta con los servicios y funciones suficientes, que garantizan su funcionamiento. Además, fue desarrollado con los controles (botones) que le permiten capturar la URL del en vivo, iniciar el proceso de recolección de los mensajes, enviar los mensajes al modelo para su evaluación, mostrar los resultados del modelo, según su clasificación (Bad, Neutral ó Good), además de permitir detener el proceso de recolección de datos.

Dentro de la evaluación del modelo se probaron mensajes en los tres grupos de clasificación, para validar el nivel de asertividad en la clasificación de los mensajes según su categoría Bad, Neutral ó Good, para cada caso, se escribieron 5 mensajes para cada clasificación y se evaluaron en el modelo obteniendo los siguientes resultados:

TABLA 4  
MENSAJES POSITIVOS

MENSAJE	CLASIFICACIÓN		
	BAD	NEUTRAL	GOOD
Nos sentimos muy complacidos con ustedes			X
Estamos muy alegres con la información			X
Siempre les hemos tenido aprecio			X
Sus decisiones son la mejores			X
Siempre con ustedes para adelante			X

Nota: mensajes seleccionados como positivos y que deben ser evaluados por el modelo como GOOD, en el 100% de los casos

Como se puede observar en los mensajes de la **TABLA 4**, estos corresponden a frases que tienen una positividad en cada uno de sus argumentos, motivo por el cual el modelo debe responder satisfactoriamente por una tendencia positiva en el resultado de la evaluación.

Resultado de la evaluación del modelo, de las frases positivas, de la tabla 4:

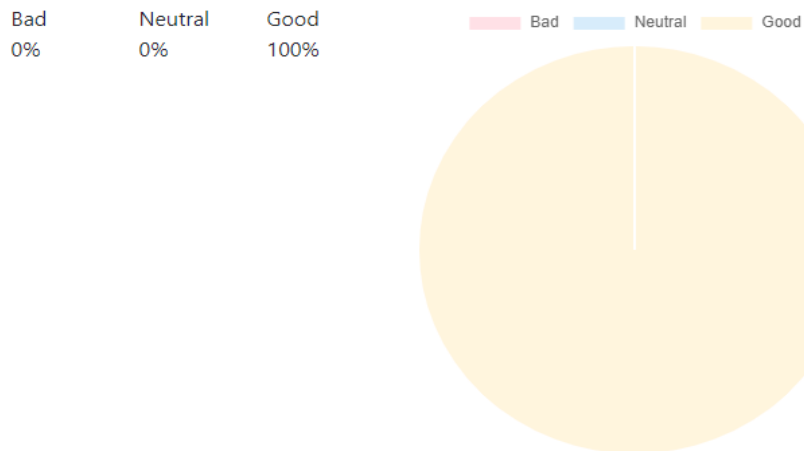


Fig. 8. Validación del modelo frases positivas

Nota: fuente elaboración propia

TABLA 5  
MENSAJES NEUTRALES

MENSAJE	CLASIFICACIÓN		
	BAD	NEUTRAL	GOOD
Puede ser		X	
quien tiene la razón		X	
podría comprometerme		X	
es probable		X	
podría pensarlo		X	

Nota: mensajes seleccionados como neutrales y que deben ser evaluados por el modelo como NEUTRAL, en el 100% de los casos.

Como se puede observar en los mensajes de la **TABLA 5**, estos corresponden a frases que tienen una neutralidad, en cada uno de sus argumentos, motivo por el cual el modelo debe responder satisfactoriamente por una tendencia neutra, en el resultado de la evaluación.

Resultado de la evaluación del modelo, de las frases neutras, de la tabla 5:

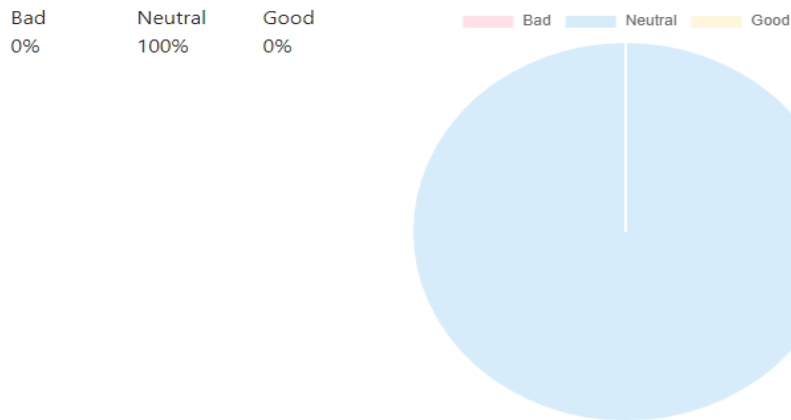


Fig. 9. Validación del modelo frases neutrales

Nota: fuente elaboración propia



TABLA 6  
MENSAJES NEGATIVOS

MENSAJE	CLASIFICACIÓN		
	BAD	NEUTRAL	GOOD
Nos sentimos muy tristes con la decisión que ustedes tomaron	X		
Ustedes se corrompieron recibiendo ese dinero	X		
Ustedes sobornaron al partido y le hicieron daño	X		
Como puede ser posible que hayan engañado a las personas	X		
Es increíble que esto pase	X		

Nota: mensajes seleccionados como negativos y que deben ser evaluados por el modelo como BAD, en el 100% de los casos.

Como se puede observar en los mensajes de la **TABLA 6**, estos corresponden a frases que tienen una negatividad, en cada uno de sus argumentos, motivo por el cual el modelo debe responder satisfactoriamente por una tendencia negativa, en el resultado de la evaluación.

Resultado de la evaluación del modelo, de las frases negativas, de la tabla 6:

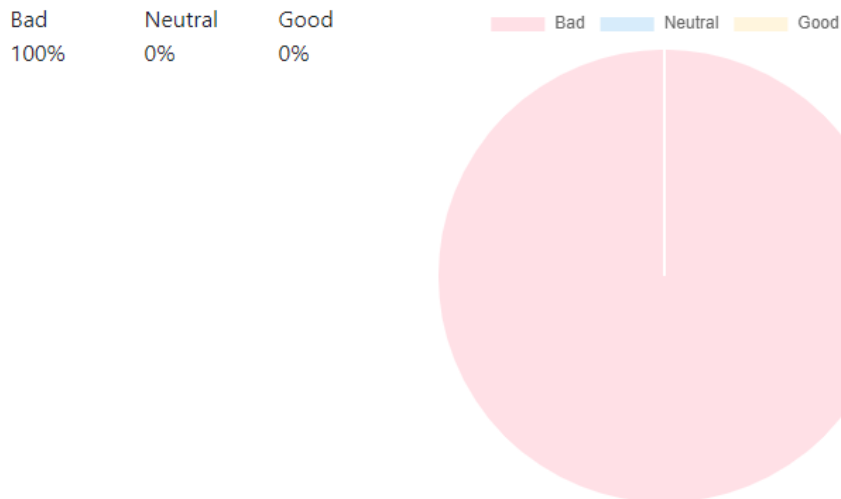


Fig. 10. Validación del modelo frases negativas

Nota: fuente elaboración propia

Se implementó un desarrollo Web, bajo el patrón de diseño MVC (Modelo-Vista-Controlador), en donde se utilizó Python para el backend, nodejs(nextjs) para el middleware y React para el frontend + bootstrap.

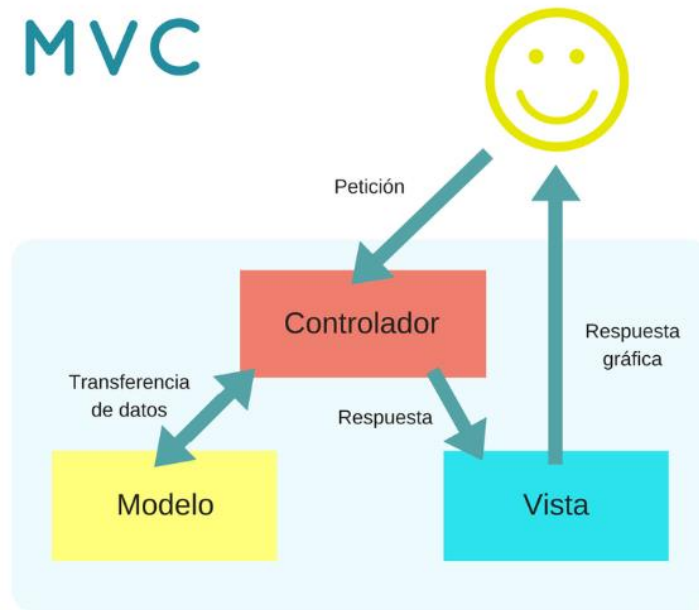


Fig. 11. Patrón de diseño MVC (Modelo-Vista-Controlador)

Nota: fuente <https://codingornot.com/mvc-modelo-vista-controlador-que-es-y-para-que-sirve>

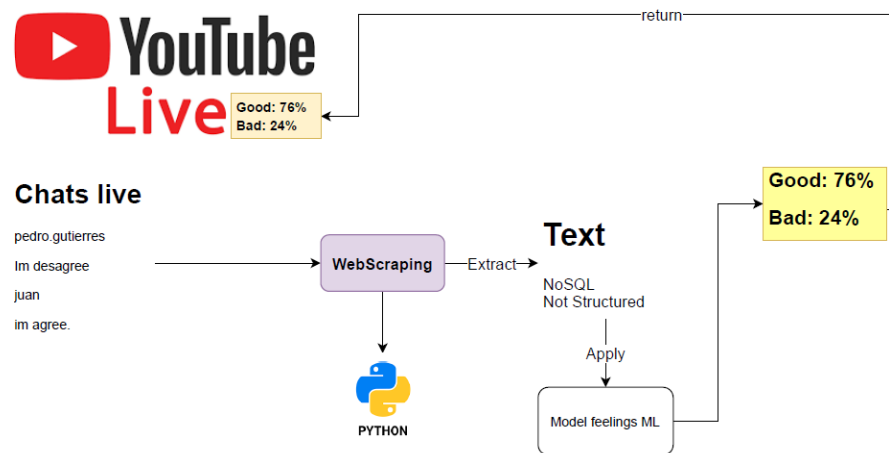


Fig. 12. Diagrama del sistema de AS para chats de transmisiones en vivo.

Nota: fuente elaboración propia

El funcionamiento del sistema, como aparece en la Figura 12, se describe de la siguiente manera:

1. El navegador habilita un objeto text, para el registro de la url que se emite en vivo por youtube.
2. Desde el navegador se realiza una petición POST, para enviar la url a la API y guardar los mensajes que se están emitiendo en el en vivo.
3. Desde el navegador se realiza una petición GET, para enviar los mensajes guardados, hasta ese momento, al modelo de análisis de sentimientos.
4. Se realiza un método GET, para tomar los resultados del modelo y mostrarlos en el navegador.
5. Se realiza un método POST, para detener el proceso de almacenamiento de los chats y refrescar los objetos de resultados de la aplicación.

A continuación, se describe el paso a paso de la forma en que se utiliza la aplicación:

1. Se elige un video de Youtube que se esté transmitiendo en vivo y que tenga habilitados los mensajes. Se debe copiar la url del video y pegar en el “text” de la aplicación. Ver figura 6:

## Análisis de sentimientos en tiempo real para intervenciones de chat en videoconferencias por streaming

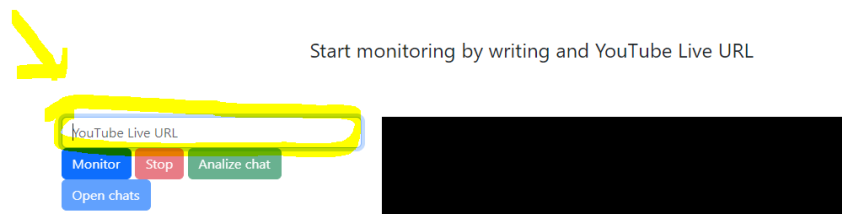


Fig. 13. Text, ingreso URL del video en vivo

Nota: fuente elaboración propia

2. Activar el botón: “Monitor”, para comenzar a recolectar los mensajes de los chats.

## Análisis de sentimientos en tiempo real para intervenciones de chat en videoconferencias por streaming

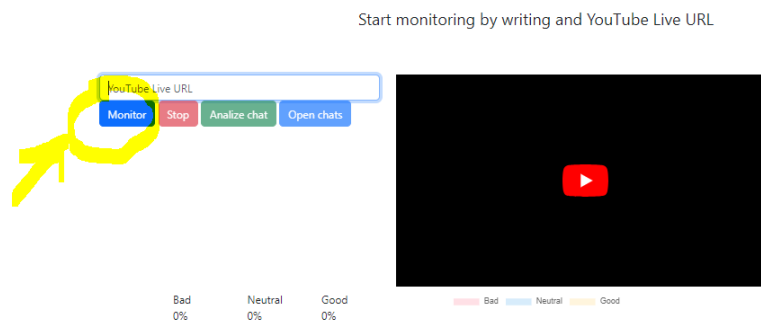


Fig. 14. Activación del botón Monitor

Nota: fuente elaboración propia

3. Presionar el botón: “Open Chats”, para abrir el módulo de los chats del video en vivo.

## Análisis de sentimientos en tiempo real para intervenciones de chat en videoconferencias por streaming

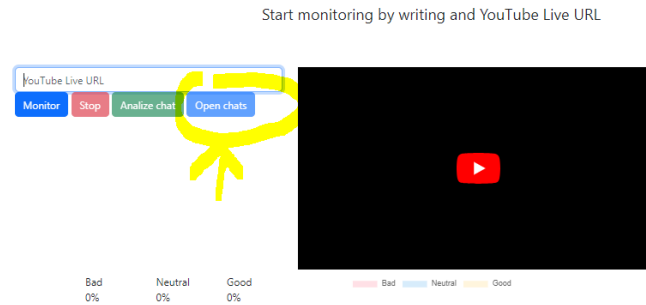


Fig. 15. Activación módulo de chats del en vivo

Nota: fuente elaboración propia

4. Presionar el botón: “Analyze chat”, para realizar el análisis en el modelo de sentimientos. Es de aclarar que los chats que serán enviados al modelo, corresponden a los que fueron almacenados en el momento en que se activó el botón: Monitor. Se observarán los porcentajes que fueron evaluados por el modelo para la clasificación correspondiente: BAD, NEUTRAL ó GOOD. y en un gráfico de tortas, se mostrará ese mismo resultado. Una vez se muestran los resultados el proceso continuará acumulando los chats, hasta que se detenga el proceso o finalice el en vivo.

### Análisis de sentimientos en tiempo real para intervenciones de chat en videoconferencias por streaming

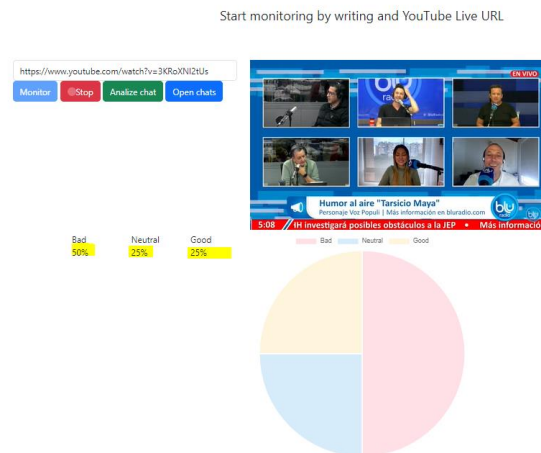


Fig. 16. Activación del modelo de análisis de sentimientos

Nota: fuente elaboración propia

5. El proceso continuará recolectando los mensajes, el indicador de activación se encuentra en el botón: Stop, el cual mostrará un círculo, que se encontrará en activación:

### Análisis de sentimientos en tiempo real para intervenciones de chat en videoconferencias por streaming

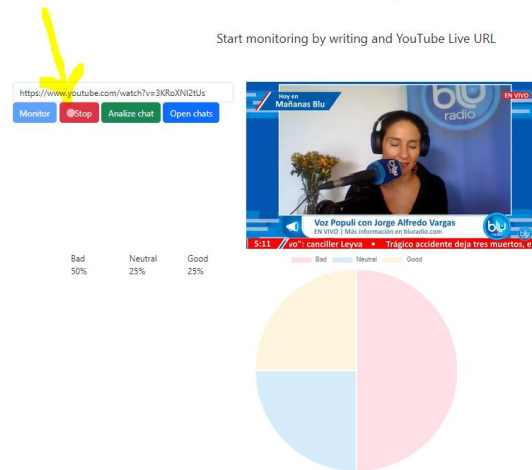


Fig. 17. Indicador de actividad del proceso

Nota: fuente elaboración propia

6. Para detener el proceso, se debe presionar el botón Stop, el cual refrescará todos los objetos como el “text”, la activación y los resultados del modelo.

### Análisis de sentimientos en tiempo real para intervenciones de chat en videoconferencias por streaming

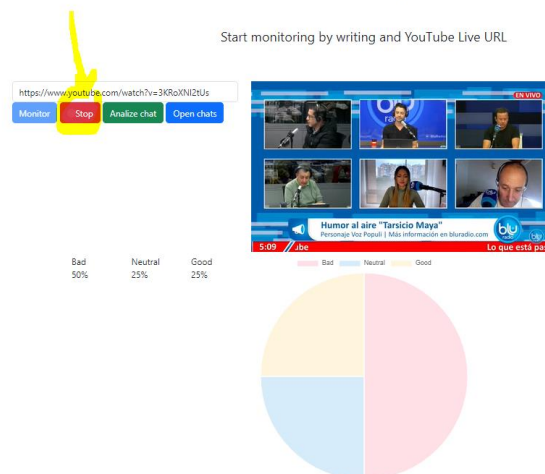


Fig. 18. Detención del proceso

Nota: fuente elaboración propia

Para acceder a la aplicación vía web, se realizó el despliegue de la aplicación en Google Cloud Platform, con Docker Compose. Para esto, se creó una máquina virtual en GCP, con toda la configuración necesaria para su correcta ejecución. El proyecto de desarrollo se encuentra en el repositorio git: [https://github.com/alexanderuribeids/chat\\_live\\_monitor.git](https://github.com/alexanderuribeids/chat_live_monitor.git) Después de realizar el despliegue de la solución e instalar las dependencias, se realizó la publicación en la nube, con el desarrollo del presente trabajo. El acceso a Chat\_live\_monitor, se realiza a través de la siguiente IP pública: <http://34.16.130.26/>



## IX. DISCUSIÓN

Con las pruebas realizadas al modelo, con los mensajes con tendencias positivas, devolvieron una evaluación del modelo, con el 100% de positividad, asegurando con esto, que la mayoría de los mensajes serán clasificados satisfactoriamente. Lo mismo ocurrió con las frases cargadas de neutralidad, donde se observó que marcaba dicha tendencia. En cuanto a la tendencia negativa, también fue testeado el modelo, arrojando resultados del 100% de acierto en su calificación negativa. Con todo esto se refleja el comportamiento reportado por los autores del modelo pre-entrenado seleccionado para llevar a cabo el análisis de sentimientos propuesto.

En la evaluación de los resultados a los en vivos de YouTube, se observó que a medida que se ejecutaba el modelo de análisis de sentimientos, en tiempos parciales y de acuerdo al tema de opinión del momento, se presentaba una distribución que correspondía a una tendencia marcada por la argumentación de la exposición y lo que está despertaba en la audiencia, esta percepción estaba determinada por una respuesta que estimulaba la sensación y el sentimiento de las personas.

## X. CONCLUSIONES

En la construcción de la solución se pudo comprobar que existen muchos recursos gratuitos que permiten desarrollar procesos que generan valor en mensajes de chat. Para este trabajo, se utilizaron librerías como Pychat, de Python, la cual funciona perfectamente en la recolección de datos, más específicamente en la captura de los mensajes que se emiten en el en vivo.

Otro aspecto importante, es el modelo clasificador, ya que existen soluciones que se pueden reutilizar, para generar valor a la información, tal es el caso de los recursos de Hugging Face Hub, el cual consiste en una plataforma en línea que permite la carga, el intercambio y la descarga de modelos pre-entrenados de NLP (procesamiento del lenguaje natural), así como otros recursos relacionados con el aprendizaje automático.

Le corresponde a la ingeniería de software la innovación y la adecuación de herramientas y procesos, para dar valor a los datos, en el caso del presente trabajo, no se parte de una necesidad manifiesta por un cliente, es una idea que puede ser de utilidad para la persona que está transmitiendo un mensaje, preguntas como: ¿qué tanto se entiende el tema del que se habla? ¿Qué tan de acuerdo estoy o no con lo hablado? Son algunas de las preguntas que se busca resolver con el desarrollo. Este proyecto permite visibilizar las opiniones de los usuarios, para que no sean solo comentarios, sino que se conviertan en información valiosa para quien genera un debate.

Otro aspecto importante y aunque no forma parte del alcance del presente proyecto, es que se pueden implementar procesos de bloqueo de chats, en el caso de que se manifiesten amenazas, palabras agresivas, entre otros y con esto se controlará automáticamente las intervenciones en el chat; también se pueden identificar bots de respuestas automáticas, con el fin de depurar los chats y no permitir mensajes recurrentes con ideologías infundadas.

Otra aplicación del producto desarrollado, tiene que ver con la recuperación de mensajes importantes, para el caso de ventas de productos, clientes potenciales, entre otros.

Al verificar las diferentes opciones de los modelos de aprendizaje automático para el análisis de sentimientos, se seleccionó el recurso de Hugging Face Hub, con un modelo pre-entrenado de NLP (procesamiento del lenguaje natural), por su versatilidad, potencia y soporte.

En el caso del desarrollo del pipeline, y específicamente, para realizar el proceso de extracción de los mensajes de los en vivo de YouTube, se utilizó la librería Pychat, de Python.

Luego y a través de backend de la aplicación, se completó el pipeline, realizando los procesos de transformación y carga, en un archivo de texto, donde se almacenaron los mensajes.

En cuanto a la carga de los datos al modelo, se utilizó el backend de la aplicación, llevando los mensajes almacenados en el archivo de texto, al modelo de análisis de sentimientos. Es importante puntualizar que la carga de los datos hacia al modelo corresponden a cortes parciales, de los mensajes que se emiten en los en vivo de YouTube, con lo cual, cada vez que se ejecute la evaluación de los datos en el modelo corresponderá a los datos de los mensajes acumulados hasta ese momento

Se logró desarrollar una interfaz intuitiva y familiar con el usuario, de tal manera que su uso es de fácil operación, con carga del video de YouTube en la misma interfaz y con botones para accionar las opciones de recolección de los mensajes y ejecución del modelo. Una vez el modelo devuelve el resultado del análisis de sentimientos, este se muestra en los gráficos de resultados, presentando el porcentaje de positividad, neutralidad o negatividad.

En cuanto a la implementación y despliegue de la aplicación se utilizó la plataforma: Google Cloud Platform, donde se disponibilizaron los recursos, para dejar productiva la aplicación desarrollada.

Este proceso fue presentado en mi empresa Globant, donde tuvo gran aceptación y en particular en el cliente Jonson & Jonson, donde observaron muchas posibilidades de aplicación para la compañía.

Otro aspecto relevante es la oportunidad evolutiva que tiene la aplicación, ya que son muchas las opciones de mejora y complementarias, por ejemplo, la ampliación a otras redes como Twitter, FaceBook, entre otras. Y añadir funcionalidades como la detección de mensajes agresivos y que sean eliminados automáticamente en la red donde se encuentren.

## XI. RECOMENDACIONES

Se puede decir que la aplicación y toda la parte funcional del proyecto de grado quedó productivo y puede ser utilizado para realizar el análisis de sentimientos a los mensajes que se emiten en los en vivo de YouTube, arrojando información valiosa, para clasificar la positividad, negatividad o neutralidad de las opiniones, no obstante, pueden considerarse muchas mejoras y aplicaciones, como por ejemplo, la clasificación a partir del contexto, de tal forma que resuelva los casos en los que las emisiones cambian constantemente la temática. En este orden de ideas, se podrían establecer tiempos estimados y aproximados para la introducción, el cuerpo y las conclusiones, de tal forma que pueda identificarse el contexto y la temática que se está tratando. Estas mejoras podrían darle mayor precisión a la clasificación del modelo, evitando que se traslapen temas que generan emociones aceptación con temas que generan rechazo.

Otro aspecto importante es el evolutivo, es decir, que sea multi-redes sociales, para abarcar mucha más información y acercar al modelo a una clasificación más robusta y enriquecida. Esto debido a que existen generadores de opinión que realizan las transmisiones de los en vivo en varias redes sociales en el mismo momento.

Teniendo en cuenta las leyes de protección de datos, también podemos considerar en mejorar la aplicación, ofreciendo servicios de identificación y bloqueo de amenazas, insultos, sobornos, bots de repetición de mensajes y demás casos que representan una vulneración de los derechos humanos. Esta información podría ser de utilidad y podría ser material probatorio en procesos de acusación, para el caso de la fiscalía.

## REFERENCIAS

- [1] M. Torres, K. Paz, F. Salazar (2019) Métodos de recolección de datos para una investigación, Guadalajara, Jalisco, México, Universidad de Guadalajara.
- [2] P. Llorente (2019) Análisis de sentimientos aplicado a la opinión política en Twitter: un sistema de clasificación en tiempo real, Barcelona, España, Universitat Oberta de Catalunya.
- [3] J. López, F. Gonzales (2021) Análisis de sentimiento de comentarios en español en Google Play Store usando BERT, Arequipa, Lima, Perú, Universidad Nacional de San Agustín de Arequipa.
- [4] G. Chanchí, A. Córdoba (2019) Análisis de emociones y sentimientos sobre el discurso de firma del acuerdo de paz en Colombia, Popayán-Cauca, Colombia, Universidad de Cartagena, Universidad Nacional, Abierta y a Distancia.
- [5] P. Hernández (2021) Análisis de sentimientos en Twitter en castellano con redes neuronales recurrentes LSTM, Madrid, España, Universidad Carlos III de Madrid.
- [6] L. Dubiau, J. M. Ale (2013) Análisis de Sentimientos sobre un Corpus en Español: Experimentación con un Caso de Estudio, Buenos Aires, Argentina, Universidad de Buenos Aires.
- [7] L. Bing (2010) Sentiment Analysis and Subjectivity. Department of Computer Science, University of Illinois at Chicago.
- [8] M. Becerra (2017) Análisis de sentimientos en Twitter: El bueno, el malo y el >:(. Facultad de Matemática, Astronomía, Física y Computación, Universidad Nacional de Córdoba, Córdoba, Argentina.
- [9] G. Chachí, M. Ospina, M. Ospino (2020), “Análisis de sentimientos de la percepción de los estudiantes de Ingeniería de Sistemas de la Universidad de Cartagena (Colombia) sobre las actividades académicas desarrolladas durante el confinamiento debido al COVID-19” Facultad de ingeniería, Universidad de Cartagena, Cartagena, Colombia.
- [10] C. Salazar, J. Aguilar, J. Monsalve, E. Montoya (2020), “Análisis de Sentimientos/Polaridad en diferentes tipos de documentos” Universidad EAFIT, Medellín, Colombia, Universidad de Los Andes, Mérida, Venezuela.
- [11] R. López (2015), “Application of Sentiment Analysis to Data Generated in Social Media” Department of Computer Science, Polytechnic School, University of Alcalá, Alcalá de Henares, Spain.
- [12] A. Cardoso, L. Talame, M. Amor, C. Neil (2019), “Minería de Opiniones: Análisis de Sentimientos en una Red Social” Grupo de Análisis de Datos /Facultad de Ingeniería e

IESIING, Universidad Católica de Salta, Universidad Abierta Interamericana, Argentina.

[13] M. Hernández, J. Gómez (2014), “Análisis de Sentimientos Aplicado a Referencias Bibliográficas” Escuela Politécnica Nacional, Facultad de Ingeniería de Sistemas, Universidad de Alicante, Spain, Departamento de Lenguajes y Sistemas Informáticos.

[14] V. Gil (2018), “Análisis de sentimientos sobre el impacto social de proyectos de vivienda en América Latina: el caso un TECHO para mi país (Colombia)” Facultad de Ingenierías y Arquitectura. Programa del dpto. de Ingeniería de Sistemas. Universidad Católica Luis Amigó, Medellín, Colombia.

[15] I. Vélez (2015), “Análisis de sentimientos sobre el posconflicto colombiano utilizando herramientas de minería de texto” Escuela Colombiana de Ingeniería Julio Garavito. Bogotá Colombia.

[16] T. Baviera (2016), “Técnicas para el análisis del sentimiento en Twitter: Aprendizaje Automático Supervisado y SentiStrength” Universitat Politècnica de València, Valencia, España.

[17] J. Martínez, S. Mateus (2019), “Sentiment analysis using machine learning: Applied to job interviews” Revista Internacional de Tecnología, Ciencia y Sociedad, 8 (2), 2019, ISSN 2530-4895, Politécnico Colombiano Jaime Isaza Cadavid, Medellín, Colombia.

[18] C. Arcila, E. Barbosa, F. Cabezuelo (2016), “Técnicas big data: análisis de textos a gran escala para la investigación científica y periodística” Universidad de Salamanca, Facultad de Ciencias Sociales, Salamanca, España, Universidad del Norte, Barranquilla, Colombia, Universidad de Valladolid, Facultad de Ciencias Sociales, Jurídicas y de la Comunicación, Segovia, España.

[19] E. Flores, J. Livia, A. García, M. Dávila (2023), “Análisis de sentimientos con inteligencia artificial para mejorar el proceso enseñanza-aprendizaje en el aula virtual” Facultad de Educación y Humanidades del Campus de Melilla, Depósito legal: GR-94-2001 eISSN: 2530-9269 · pISSN: 1577-4147, Universidad Nacional Federico Villarreal, Lima, Perú.

[20] S. Jiménez, E. Martínez, M. Martina, L. Ureña (2014), “Desafíos del Análisis de Sentimientos” Departamento de Informática, Universidad de Jaén, , Cazalla de la Sierra, España.