**ANTEPROYECTO DEL**

**TRABAJO FIN DE MÁSTER**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **INFORMACIÓN GENERAL** | | | | | |
| **Alumno/a** | Azul María García | | | | |
| **Tutor/es:** | Antonio Benítez Hidalgo | | | | |
| **Título** | Representación y Análisis de la Opinión de los Clientes por medio de Búsqueda Vectorial | | | | |
| **Subtítulo** *(solo si en grupo)* |  | | | | |
| **Título en inglés** | Customer Opinion Representation and Analysis via Vector Search | | | | |
| **Subtítulo en inglés** *(solo si en grupo)* |  | | | | |
| **Trabajo en grupo:** | **Sí** |  | **No** | **X** |  |
| **Otros integrantes del grupo:** |  | | | | |

|  |
| --- |
| **INTRODUCCIÓN** |
| *Contextualización del problema a resolver. Describir claramente de dónde surge la necesidad de este TFM y el dominio de aplicación. En caso de que el TFM se base en trabajos previos, debe aclararse cuáles son las aportaciones del TFM.* |
| La búsqueda vectorial es un método avanzado que se utiliza para identificar y recuperar información en un conjunto de datos que es más similar a la consulta dada [1]. En esencia, transforma las consultas y los elementos en un conjunto de datos en representaciones vectoriales en un espacio multidimensional, lo que permite una búsqueda eficiente y precisa. La búsqueda vectorial es especialmente útil cuando se trata de grandes volúmenes de datos, como texto, imágenes o audio, donde las técnicas tradicionales de búsqueda pueden ser menos eficientes. Esta técnica se ha utilizado en una variedad de aplicaciones, desde motores de recomendación y análisis de sentimientos hasta reconocimiento de voz y visión por computadora [2].  En este Trabajo Fin de Máster, proponemos un sistema que permita analizar la percepción de los clientes de una empresa, como un hotel o un restaurante, usando técnicas de búsqueda vectorial sobre las opiniones de los clientes. Las reseñas se analizarán utilizando técnicas de análisis de sentimientos para generar etiquetas que representen el sentimiento expresado por los clientes y se almacenarán los vectores resultantes en una base de datos vectorial junto a metadatos para su posterior análisis.  El sistema debe ser capaz de realizar consultas eficientes en la base de datos para recuperar las reseñas más relevantes y poder representar el sentimiento generalizado de los clientes en diferentes áreas por medio de una interfaz gráfica intuitiva. |

|  |
| --- |
| **OBJETIVOS** |
| *Descripción detallada de en qué consistirá el TFM. Además de los objetivos generales deben describirse las funcionalidades del software que se desarrollará a alto nivel.* |
| El objetivo principal de este Trabajo Fin de Máster es desarrollar un sistema que permita a los empleados de una empresa obtener una visión general del sentimiento de los clientes en diferentes áreas para apoyar la toma de decisiones.  Para lograrlo, se propone implementar las siguientes funcionalidades principales:   1. **Recopilación de reseñas:** recopilar automáticamente y pre-procesar las reseñas de los clientes de diversas plataformas de reseñas en línea mediante técnicas de *web scraping* [3]. 2. **Análisis de sentimientos:** se aplicarán técnicas de análisis de sentimientos para procesar las reseñas y asignar etiquetas de sentimiento a cada una de ellas. Esto permitirá clasificar las opiniones de los clientes en positivas, negativas o neutrales. 3. **Almacenamiento de vectores y metadatos:** además de las etiquetas de sentimiento y otros metadatos, se almacenarán los vectores resultantes del análisis de sentimientos en una base de datos vectorial. Esto facilitará consultas eficientes y permitirá representar el sentimiento generalizado de los clientes en diferentes áreas. 4. **Consultas y representación de resultados:** la aplicación permitirá realizar consultas en la base de datos para recuperar las reseñas más relevantes en función de diferentes criterios, como la fecha, la puntuación, las palabras clave, entre otros. Los resultados se representarán de manera intuitiva y comprensible, proporcionando a los empleados una visión general del sentimiento de los clientes en distintas áreas. 5. **Detección de puntos de mejora y puntos fuertes:** la aplicación ofrecerá funcionalidades para identificar los aspectos en los que la empresa puede mejorar según las opiniones de los clientes, así como resaltar los puntos fuertes que los clientes aprecian. Esto ayudará a la empresa a tomar decisiones estratégicas para mejorar la experiencia del cliente.   En función de la marcha del proyecto, se contemplará incluir módulos adicionales alimentados por modelos de lenguaje en escala (LLM) [4]. Los LLM pueden analizar el tono, el contexto y el contenido de las reseñas de los clientes para, por ejemplo, proporcionar sugerencias de respuestas relevantes y apropiadas. Con estas capacidades, la herramienta podrá asistir a los empleados en el manejo eficiente de la comunicación con los clientes, especialmente cuando se enfrentan a un gran volumen de reseñas. |

|  |
| --- |
| **MÉTODOS Y FASES DE TRABAJO** |
| **FASES DE TRABAJO:** |
| *Enumeración y breve descripción de las fases de trabajo en las que consistirá el TFM.* |
| 1. **Estudio del estado del arte en búsqueda vectorial:**   En esta etapa, se realizará una revisión exhaustiva del estado actual de la búsqueda vectorial. Se investigarán y comprenderán los conceptos clave, como vectores de búsqueda y bases de datos vectoriales.   * Investigar y comprender los conceptos clave relacionados con la búsqueda vectorial, como vectores de búsqueda y bases de datos vectoriales. * Explorar bibliotecas de procesamiento de lenguaje natural, como NLTK o spaCy, para comprender su funcionalidad y su aplicación en el procesamiento de texto. * Familiarizarse con las librerías de Python utilizadas para el análisis y visualización de datos, como NumPy, Pandas y Matplotlib, para entender cómo se pueden aplicar en el contexto de la búsqueda vectorial.  1. **Implementación de técnicas de representación vectorial:**   En esta etapa, se implementarán técnicas de representación vectorial utilizando frameworks de aprendizaje automático. Se utilizarán estos frameworks para crear y entrenar modelos de representación vectorial, que puedan capturar y representar la información semántica de los textos en forma de vectores. Se realizará el preprocesamiento de los datos y la transformación a vectores que puedan ser utilizados por los modelos de representación vectorial. Esto implica realizar tareas como limpieza de texto, tokenización, eliminación de stopwords y aplicar técnicas de vectorización como TF-IDF o word embeddings.   * Utilizar frameworks de aprendizaje automático, como TensorFlow o PyTorch, para crear y entrenar modelos de representación vectorial, como BERT. * Explorar librerías de procesamiento de texto y vectorización, como scikit-learn, para realizar el preprocesamiento de los datos de texto y transformarlos en vectores.  1. **Implementación de algoritmos para análisis de sentimiento:**   En esta etapa, se implementarán los algoritmos para el análisis de sentimiento utilizando librerías de procesamiento de lenguaje natural, para procesar los datos de texto y determinar la polaridad de cada texto (positivo, negativo, neutral).   * Investigar y seleccionar las librerías de procesamiento de lenguaje natural más adecuadas para el análisis de sentimiento, como NLTK o TextBlob. * Utilizar estas librerías para implementar algoritmos de análisis de sentimiento que puedan procesar los datos de texto y determinar la polaridad de cada texto.  1. **Despliegue de base de datos vectorial:**   En esta etapa, se seleccionará una base de datos que sea capaz de almacenar y consultar eficientemente los datos vectoriales generados en las etapas anteriores. Se configurará la base de datos seleccionada y se definirá el esquema de almacenamiento adecuado para los vectores de búsqueda. Además, se relizará web scraping para obtener los datos necesarios desde fuentes externas y almacenarlos en la base de datos vectorial.   * Seleccionar una base de datos escalable y de alto rendimiento, como MongoDB o Apache Cassandra, que sea capaz de almacenar y consultar eficientemente los datos vectoriales. * Utilizar librerías para realizar web scraping, como BeautifulSoup o Selenium, para obtener los datos necesarios y almacenarlos en la base de datos vectorial seleccionada.  1. **Desarrollo del backend:**   En esta etapa, se desarrollará el backend de la solución. Esto implica la implementación de la lógica de negocio, la gestión de las solicitudes del frontend y la comunicación con otras partes del sistema. Se utilizarán APIs, para permitir la comunicación y el intercambio de datos entre el frontend, el backend y otras partes del sistema.   * Utilizar Python como lenguaje de programación para desarrollar el backend de la solución. * Implementar APIs, como RESTful APIs o GraphQL, para permitir la comunicación y el intercambio de datos entre el frontend, el backend y otras partes del sistema.  1. **Desarrollo de la interfaz de usuario:**   Esta etapa se centrará en el desarrollo de la interfaz de usuario de la solución. El objetivo es crear una interfaz intuitiva y atractiva que permita a los usuarios interactuar de manera efectiva con la funcionalidad proporcionada por el sistema. A lo largo del desarrollo, se irán realizando pruebas con usuarios reales que permitan iterar en el diseño. También, se establecerán las conexiones necesarias para permitir la comunicación entre la interfaz de usuario y el backend. Se utilizarán APIs definidas en el backend para enviar y recibir datos, realizar consultas y procesar acciones en respuesta a las interacciones del usuario.   * Utilizar frameworks para el desarrollo de aplicaciones web, como Streamlit, para crear una interfaz de usuario intuitiva y fácil de usar. * Utilizar lenguajes de programación web, como HTML, CSS y JavaScript, para diseñar y desarrollar la interfaz de usuario y permitir una interacción efectiva con la solución.  1. **Documentación:**  * Generar documentación de la solución, que incluya descripciones detalladas de los pasos seguidos en cada etapa, así como información sobre las tecnologías utilizadas, las decisiones de diseño y cualquier otra consideración relevante. |
| **TEMPORIZACIÓN:** |
| *La siguiente tabla deberá contener una fila por cada una de las fases enumeradas en la sección anterior.* |
| |  |  | | --- | --- | | **FASE** | **HORAS** | |  | | Estudio del estado del arte en búsqueda vectorial | 45 | | Implementación de técnicas de representación vectorial | 65 | | Implementación de algoritmos para análisis de sentimiento | 80 | | Despliegue de base de datos vectorial | 55 | | Desarrollo del *backend* | 60 | | Desarrollo de la interfaz de usuario | 100 | | Documentación | 20 | |  | **425** | |

|  |
| --- |
| **ENTORNO TECNOLÓGICO** |
| **TECNOLOGÍAS EMPLEADAS:** |
| *Enumeración de las tecnologías utilizadas (lenguajes de programación, frameworks, sistemas gestores de bases de datos, etc.) en el desarrollo del TFM.* |
| Python |
| NumPy / Pandas / Matplotlib. |
| NLTK / spaCy / TextBlob / Scikit-learn |
| TensorFlow / PyTorch |
| MongoDB / Apache Cassandra |
| BeautifulSoap / Selenium |
| RESTful APIs / GraphQL |
| HTML / CSS / JavaScript / Streamlit |
| **RECURSOS SOFTWARE Y HARDWARE:** |
| *Listado de dispositivos (placas de desarrollo, microcontroladores, procesadores, sensores, robots, etc.) o software (IDE, editores, etc.) empleados en el desarrollo del TFM.* |
| PyCharm |
| WebStorm |
| Docker |

|  |
| --- |
| **REFERENCIAS** |
| *Listado de referencias (libros, páginas web, etc.)* |
| [1] <https://www.elastic.co/es/what-is/vector-search> |
| [2] <https://www.algolia.com/blog/ai/what-is-vector-search/> |
|  |
| [3] <https://www.techopedia.com/definition/34948/large-language-model-llm> |
| [4] <https://www.codingdojo.com/blog/what-is-web-scraping> |
|  |
|  |