**Desarrollo de un Chatbot para Mesa de Servicio utilizando NLP y Machine Learning**

**1. Entendimiento del Negocio**

La mesa de servicio de Colfondos maneja un alto volumen de solicitudes de soporte técnico y consultas operativas. Actualmente, la resolución de estas solicitudes se basa en la experiencia de los agentes y en una base de conocimientos dispersa. Para optimizar la gestión de estos casos, se propone desarrollar un chatbot basado en Procesamiento de Lenguaje Natural (NLP) y técnicas de Machine Learning, capaz de clasificar incidentes y sugerir soluciones basadas en casos históricos.

**2. Comprensión de los Datos**

**2.1. Descripción de los Datos**

El dataset utilizado consta de 45,000 registros históricos de casos resueltos. Cada registro contiene:

* Descripción del problema (texto libre)
* Solución aplicada
* Área de resolución (cuando está disponible)

**2.2. Preprocesamiento de Datos**

Se realizaron los siguientes pasos:

* Eliminación de caracteres especiales, espacios innecesarios y stopwords
* Conversión de texto a minúsculas
* Tokenización y lematización

**2.3. Generación de Embeddings**

Para representar los casos en un espacio vectorial, se utilizaron embeddings generados con modelos de Ollama:

* mxbai-embed-large
* nomic-embed-text
* all-MiniLM-L6-v2

Se compararon estos modelos utilizando métricas de similitud y calidad de clustering.

**3. Preparación de los Datos**

**3.1. Almacenamiento en MongoDB**

Se optó por MongoDB debido a:

* Su flexibilidad para manejar datos no estructurados
* Escalabilidad y eficiencia en consultas de búsqueda de texto
* Facilidad de integración con modelos de NLP

Cada caso fue almacenado con su embedding y metadatos relevantes.

**4. Modelado**

**4.1. Técnicas de Clustering**

Se aplicaron algoritmos no supervisados para agrupar los casos en categorías latentes:

* **K-Means**: utilizado con análisis de silueta para determinar el número óptimo de clusters
* **DBSCAN**: efectivo para detectar outliers y agrupar casos similares
* **HDBSCAN**: mejora la estabilidad de DBSCAN y se ajusta mejor a estructuras de datos densas

**4.2. Modelado Supervisado con BERT**

Una vez obtenidos los clusters, se utilizaron como etiquetas para entrenar un modelo supervisado basado en BERT. Este modelo tiene la capacidad de:

* Clasificar nuevos casos en las categorías aprendidas
* Mejorar su precisión con nuevos datos etiquetados

**5. Evaluación**

Se compararon los resultados de los modelos de embeddings utilizando:

* **Análisis de silueta** para evaluar la calidad de los clusters
* **Precisión y recall** en la clasificación supervisada con BERT

**6. Implementación del Chatbot**

**6.1. Arquitectura del Chatbot**

El chatbot se diseñó para recibir descripciones de casos y seguir un flujo de procesamiento:

1. Convertir el texto ingresado en embeddings
2. Identificar la categoría más probable usando el modelo de clasificación
3. Buscar casos similares en la base de datos para sugerir una solución
4. Si no hay coincidencias claras, generar una respuesta utilizando modelos generativos como GPT o LLaMA

**6.2. Integración con MongoDB**

El chatbot consulta MongoDB en tiempo real para recuperar soluciones relevantes y mejorar sus respuestas con el tiempo.

**7. Conclusiones y Trabajos Futuros**

Este proyecto demuestra cómo los embeddings y técnicas de NLP pueden mejorar la eficiencia de una mesa de servicio. Para futuras mejoras se sugiere:

* Incluir más datos etiquetados para mejorar la precisión de clasificación
* Evaluar modelos de lenguaje más avanzados
* Integrar retroalimentación de los usuarios para mejorar el aprendizaje del chatbot

Esta estructura sigue la metodología CRISP-DM y cubre cada etapa de manera detallada. Si necesitas ajustes o mayor profundidad en algún punto, dime y lo afinamos.