CARRERA DE ESPECIALIZACIÓN EN INTELIGENCIA ARTIFICIAL

MEMORIA DEL TRABAJO FINAL

Desarrollo de un *pipeline* de aprendizaje continuo para chatbots basados en PLN

Autor:
Ing. Porra Bustos, Matias Exequiel (UTN-FRLR)

Director:
Dr. Ing. Cárdenas Rodrigo (FIUBA)

Jurados:
Esp. Ing. Torrent Leandro (FIUBA)
Nombre del jurado 2 (pertenencia)
Nombre del jurado 3 (pertenencia)

Este trabajo fue realizado en la Ciudad Autónoma de Buenos Aires, entre mayo de 2024 y abril de 2025.

Resumen

El presente trabajo propone una solución avanzada para la automatización de la comunicación empresarial, orientada a emprendedores y pequeñas empresas. El sistema desarrollado consiste en un chatbot que integra técnicas de procesamiento del lenguaje natural (PLN) y gestión de bases de datos, articulado en un pipeline de aprendizaje continuo. Este chatbot está diseñado para optimizar la interacción con los clientes, ya que genera respuestas precisas y relevantes a sus consultas en lenguaje natural. Además, el sistema se retroalimenta de las interacciones previas para mejorar su rendimiento de manera constante y adaptarse mejor a las necesidades de los usuarios.

Agradecimientos

A mi novia por apoyo incondicional durante todo el proceso de desarrollo de este trabajo.

A mi familia, por estar siempre presente.

A mi tutor, por su guía y su conocimiento.

A Michael Schreiber por sus grandes aportes y guía en mi formación profesional en IA.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción general

En este capítulo se presentan las motivaciones que impulsaron el desarrollo del sistema, diseñado para automatizar la comunicación entre emprendedores y sus clientes. Se ofrece, además, una explicación detallada sobre qué son los chatbots y las razones por las que su uso resulta fundamental en este contexto. Finalmente, se describen los objetivos principales del trabajo, centrados en la facilidad de implementación y en la mejora de la eficiencia operativa, junto con el alcance definido durante la planificación.

1.1. Motivación

El sistema desarrollado en el presente trabajo surge de la necesidad de proporcionar a emprendedores una herramienta que les permita automatizar la comunicación con sus clientes de manera eficiente y personalizada. Esta herramienta les ofrece una ventaja competitiva al optimizar la gestión de sus interacciones con los usuarios y les permite mejorar la eficiencia operativa, además de reducir los costos asociados a la atención al cliente.

1.1.1. ¿Qué es un chatbot?

Un chatbot es un programa de software que emplea técnicas de Procesamiento del Lenguaje Natural (PLN) [1], para interpretar y contestar automáticamente las consultas de los usuarios de manera coherente y eficiente. Esto facilita la automatización de interacciones comunes y mejora la experiencia del cliente.

1.1.2. ¿Por qué un chatbot?

A continuación, se destacan algunas de las razones principales por las que un chatbot es la solución ideal para este trabajo:

- Disponibilidad 24/7: los chatbots están disponibles en todo momento, lo que les permite a las empresas atender consultas de sus clientes a cualquier hora del día.
- Reducción de costos: al automatizar la atención al cliente, los chatbots ayudan a las empresas a reducir los costos asociados al personal humano sin comprometer la calidad del servicio.
- Recolección de datos valiosos: los chatbots pueden recopilar y analizar información relevante sobre las interacciones con los clientes, lo que les permite a las empresas ajustar sus estrategias de manera eficiente.

- Optimización de tiempos de respuesta: la capacidad de generar respuestas inmediatas en función de consultas predefinidas optimiza los tiempos de respuesta.

1.2. Alcance y objetivos

1.2.1. Objetivos principales

El trabajo se enfoca en tres objetivos principales que se interrelacionan para ofrecer una solución integral a pequeños emprendedores. Cada uno de estos objetivos está orientado a asegurar que la herramienta sea accesible, fácilmente implementable y de bajo costo, para garantizar una experiencia eficiente y optimizada para las empresas que la adopten.

A continuación, se detallan los objetivos principales:

- Proporcinar acceso a pequeños emprendedores: desarrollar un sistema que sea de fácil adopción y pueda ser utilizado por pequeñas empresas, independientemente de sus conocimientos técnicos.
- Garantizar una fácil implementación: diseñar una solución replicable, de modo que cualquier empresa pueda contar con el chatbot simplemente al proporcionar el contexto necesario para su entrenamiento. No será necesario disponer de infraestructura compleja ni personal especializado.
- Reducir el costo de mantenimiento: proporcionar una herramienta con un bajo costo de mantenimiento, tanto en términos económicos como de tiempo, para que los emprendedores puedan centrarse en su negocio principal.

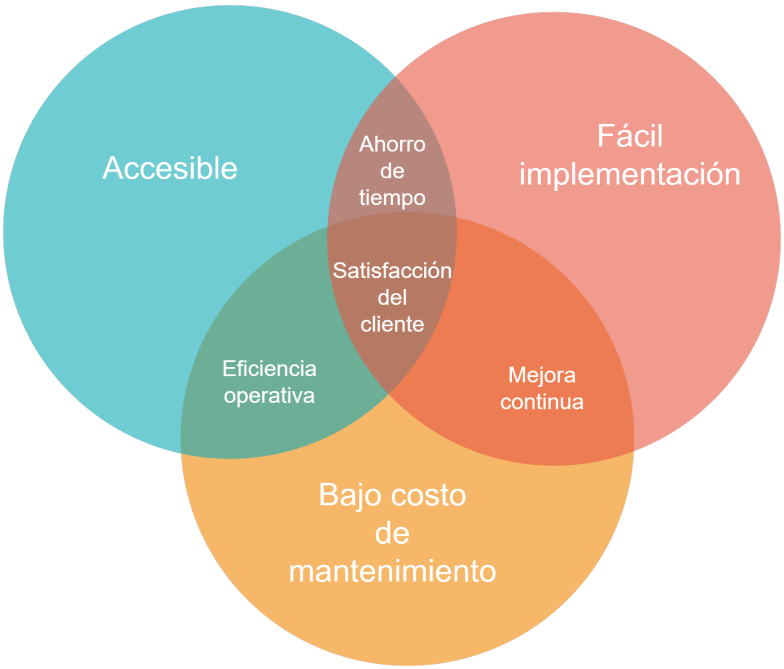

FIGURA 1.1. Relación entre los objetivos centrales.

1.2.2. Alcance del trabajo

Durante la planificación del trabajo se propusieron las siguientes actividades:

- Diseño, desarrollo e implementación de un *pipeline* completo para la creación y gestión de chatbots basados en PLN, con capacidad de aprendizaje continuo.
- Desarrollo de algoritmos para el preprocesamiento de información y generación de respuestas relevantes y coherentes.
- Establecimiento de métricas y procedimientos de evaluación para medir la calidad y eficacia de las respuestas del chatbot.
- Implementación de un sistema de retroalimentación que utilice los datos de interacciones con usuarios para mejorar el rendimiento del modelo.

El presente trabajo excluye:

- Desarrollo de interfaces de usuario avanzadas, por fuera de las básicas necesarias para probar el pipeline.
- Integración con sistemas externos, como bases de datos o plataformas de gestión de clientes.
- Implementación del sistema en un entorno productivo.
- Actividades de marketing o promoción del producto final.

1.3. Estado del arte

Los chatbots se han integrado en numerosos aspectos de la vida diaria y en diversos canales de comunicación. Se encuentran presentes en redes sociales como Facebook e Instagram, donde facilitan la interacción con empresas. Además, muchas organizaciones y entidades gubernamentales, han implementado chatbots para mejorar la atención al ciudadano. Plataformas como IBM Watson, Google Dialogflow y Microsoft Bot Framework permiten a las empresas crear chatbots personalizados con gran flexibilidad.

1.3.1. Recursos para chatbots

Servicios basados en APIs

Los servicios basados en API[1] simplifican la implementación de chatbots al ofrecer interfaces estandarizadas para acceder a funcionalidades avanzadas. Estos servicios permiten una mayor personalización y flexibilidad en el diseño de chatbots, que facilitan la integración con otros sistemas y plataformas. La utilización de APIs permite a las empresas adaptar las soluciones a sus necesidades específicas sin necesidad de desarrollar toda la infraestructura desde cero.

[1] API (Interfaz de Programación de Aplicaciones) es un conjunto de reglas y protocolos que permite que diferentes aplicaciones se comuniquen entre sí.

Modelos locales

Los modelos locales como Llama 3.1, Phi 3, Gemma y Mistral son soluciones robustas que operan sin depender de servicios externos. Cada uno de estos ofrece características y ventajas específicas, como mayor control sobre los datos y la posibilidad de operar en entornos con restricciones de privacidad. Su uso es particularmente valioso cuando se requiere un alto nivel de personalización y seguridad en la gestión de datos.

1.3.2. Tendencias actuales

Cada vez más empresas optan por utilizar servicios basados en API, que han hecho más accesible y económico el despliegue de chatbots sofisticados. Estas herramientas simplifican la implementación y permiten la creación rápida de soluciones adaptadas a las necesidades específicas de cada empresa o entidad. Aunque los modelos locales son valiosos por su robustez y capacidad para operar sin depender de servicios externos, son útiles especialmente cuando se requiere un mayor control o confidencialidad.

Capítulo 2

Introducción específica

Este capítulo explora los requisitos fundamentales para el desarrollo de un sistema de chatbot, que abarcan aspectos funcionales, de documentación, pruebas, interfaz y rendimiento. Además, revisa los diferentes tipos de chatbots y modelos de inteligencia artificial, con énfasis en sus características y aplicaciones en el procesamiento de lenguaje natural.

2.1. Requerimientos

1. Requerimientos funcionales:
   a) El sistema debe ser capaz de procesar y comprender mensajes de texto entrantes.
   b) El chatbot debe poder proporcionar respuestas relevantes y precisas a las consultas de los usuarios.
   c) Los usuarios deben tener la posibilidad de interactuar con el chatbot a través de una interfaz de usuario.
   d) Se requiere una interfaz de usuario que facilite el proceso de reentrenamiento del chatbot mediante el uso de los historiales de conversación.
2. Requerimientos de documentación:
   a) Documentar detalladamente las tecnologías utilizadas y sus principales características, así como las particularidades de diseño de cada una.
   b) El sistema no almacenará datos personales de los usuarios, sino que se limitará a responder preguntas frecuentes. En caso de requerirse almacenamiento de datos, se utilizará una plataforma con medidas de seguridad integradas, como autenticación mediante logins con Google.
   c) Entregar una memoria técnica que contenga información detallada sobre la implementación del sistema, que incluya aspectos técnicos, arquitectura y decisiones de diseño.
   d) Proporcionar un registro de avance que documente los hitos alcanzados durante el desarrollo del proyecto, que contemple fechas de cumplimiento y descripciones de las tareas realizadas.

3. Requerimientos de Testing:
    a) Se llevarán a cabo pruebas en diferentes escenarios y con diferentes tipos de contexto de datos, para garantizar la fiabilidad y precisión del chatbot.
4. Requerimientos de la Interfaz:
    a) Se requiere una interfaz interactiva, que permita hacer preguntas y obtener respuestas en tiempo real dentro del mismo entorno de interacción.
5. Requerimientos de Rendimiento:
    a) Se establece un límite máximo de tiempo de respuesta de 1 minuto, para asegurar una experiencia satisfactoria para el usuario.

2.2. Modelos de inteligencia artificial para PLN

El procesamiento de lenguaje natural (PLN) ha avanzado significativamente gracias al desarrollo de diversas técnicas de inteligencia artificial. Estas se pueden clasificar en varias categorías, cada una con su propio enfoque y aplicación.

Modelos basados en reglas: estos modelos utilizan gramáticas y diccionarios para analizar el lenguaje. Aunque son efectivos en tareas específicas, su rigidez y la necesidad de una extensa programación manual limitan su aplicabilidad en contextos más amplios.

Modelos estadísticos: a medida que los datos comenzaron a acumularse, los modelos estadísticos, como los n-gramas, se convirtieron en populares. Estos modelos predicen la probabilidad de una palabra en función de las palabras anteriores. Sin embargo, su dependencia de datos limitados puede resultar en un contexto insuficiente, lo que afecta la calidad de las predicciones.

Modelos de aprendizaje profundo: con el auge del aprendizaje profundo, arquitecturas como RNN (Redes Neuronales Recurrentes), LSTM (Memoria a Largo Plazo) y transformers han revolucionado el PLN. Estos modelos pueden captar patrones complejos en grandes volúmenes de datos, lo que les permite realizar tareas como la traducción automática y la generación de texto de manera más efectiva.

2.2.1. Modelos de Lenguaje modernos

Modelos como BERT (Bidirectional Encoder Representations from Transformers) y GPT (Generative Pre-trained Transformer) han transformado el campo del procesamiento de lenguaje natural (PLN), al ofrecer enfoques innovadores para comprender y generar texto.

- **GPT**: este modelo utiliza técnicas de aprendizaje profundo para generar texto coherente y contextualmente relevante. Su capacidad para crear respuestas naturales ha revolucionado la interacción de los chatbots con los usuarios, lo que permite conversaciones más fluidas y precisas.
- **BERT**: a diferencia de otros modelos, BERT se centra en el análisis bidireccional del texto, lo que le permite entender el contexto de manera más

profunda. Esta característica mejora la identificación de intenciones y la respuesta a preguntas complejas, lo que ayuda a crear chatbots más competentes en diálogos matizados.

2.3. Modelos de chatbots: RAG vs Fine-Tuning

Los chatbots se pueden clasificar según sus arquitecturas y métodos de entrenamiento. Entre las más destacadas se encuentran Retrieval Augmented Generation (RAG) y fine-tuning. Ambas arquitecturas ofrecen ventajas complementarias, lo que las convierten en opciones populares en el desarrollo de chatbots efectivos [2].

2.3.1. Retrieval Augmented Generation (RAG)

RAG combina la generación de lenguaje natural con la recuperación de información, enfocándose principalmente en una *knowledge base*[1]. Cuando un usuario formula una pregunta, RAG utiliza *embeddings*[2] para identificar y recuperar las partes del *relevant knowledge*[3] que son pertinentes a esa pregunta. Este conocimiento relevante se inyecta en el *prompt*[4] enviado al modelo de lenguaje (LLM). El *prompt* incluye tanto la pregunta como el contexto relacionado, lo que permite a la LLM generar respuestas más precisas y contextualizadas. La incorporación de datos externos en este proceso enriquece el conocimiento del modelo en tiempo real y disminuye las posibilidades de *hallucinations*[5]—respuestas que, aunque parecen plausibles, son incorrectas. Además, RAG utiliza bases de datos vectoriales y la búsqueda semántica para recuperar información relevante, lo que mejora la precisión y relevancia de las respuestas del chatbot. Este enfoque es especialmente útil en situaciones donde se requiere información actualizada o especializada, como en consultas de productos o en el soporte técnico.

[1] knowledge base [base de conocimiento].

[2] embeddings [representaciones vectoriales].

[3] relevant knowledge [conocimiento relevante].

[4] prompt [entrada o solicitud al modelo de lenguaje].

[5] hallucinations [alucinaciones].

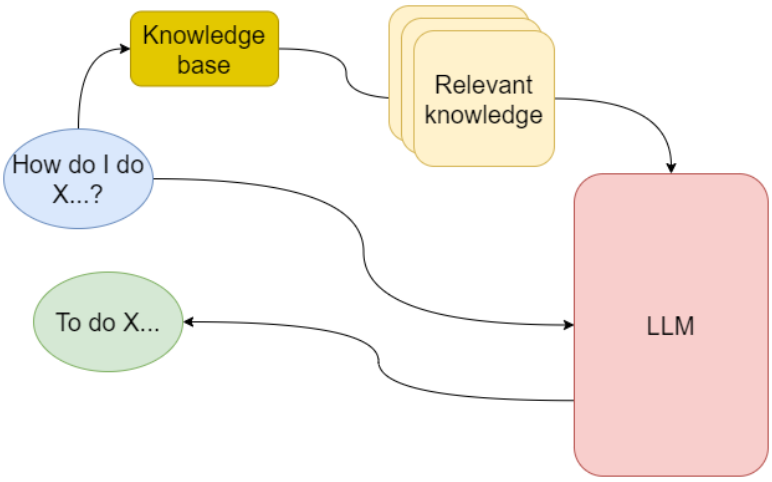

FIGURA 2.1. Diagrama de flujo de una consulta en un chatbot con arquitectura RAG.

2.3.2. Fine-Tuning

El fine-tuning implica ajustar un modelo de lenguaje preentrenado en un conjunto de datos específico para mejorar su comportamiento en contextos particulares. Este proceso optimiza la efectividad del modelo en aplicaciones industriales específicas, como en medicina, derecho o ingeniería. Sin embargo, una desventaja del fine-tuning es que el modelo resultante puede quedar congelado en un estado particular, lo que limita su adaptabilidad a nuevos contextos sin un nuevo ciclo de entrenamiento. Aunque el fine-tuning proporciona respuestas más precisas en contextos específicos, también puede carecer de la flexibilidad de RAG, que permite incorporar datos contextuales en tiempo real.

2.3.3. Ventajas y desventajas de los modelos RAG y Fine-Tuning

Esta subsección presenta una comparación entre ambos modelos, que muestra sus ventajas y desventajas.

TABLA 2.1. Ventajas y desventajas de los modelos RAG y Fine-Tuning.

Modelo	Ventaja	Desventaja
RAG	▪ Permite incorporar información contextual actualizada en tiempo real. ▪ Reduce la probabilidad de alucinaciones al usar datos relevantes. ▪ Mejora la precisión y relevancia de las respuestas del chatbot.	▪ Requiere una infraestructura adecuada para la recuperación de información. ▪ Dependencia de la calidad y disponibilidad de la base de datos. ▪ Puede ser más complejo de implementar y mantener.
Fine-Tuning	▪ Mejora la calidad de las respuestas al adaptar el modelo a un dominio específico. ▪ Proporciona un rendimiento más consistente en contextos bien definidos. ▪ Permite la optimización de la latencia al trabajar con un modelo entrenado.	▪ El modelo se congela en el tiempo y no se adapta a cambios contextuales. ▪ Requiere un conjunto de datos específico para el entrenamiento, lo que puede ser costoso. ▪ Menor flexibilidad para abordar consultas inesperadas o no entrenadas.

Capítulo 3

Diseño e implementación

3.1. Diseño de la arquitectura

...

3.1.1. Propuestas de implementación

... Descripción de los prototipos evaluados durante el diseño de la arquitectura y sus ventajas/desventajas.

3.1.2. Problemas y mitigaciones

... Problemas con los que me he encontrado mientras realizaba cada prototipo (para dar contexto a la arquitectura elegida en el punto siguiente)

3.1.3. Elección de la arquitectura de chatbot

... Razones por las cuales se eligió el tipo de chatbot (Fine-tuning o RAG definidos en el capitulo de Introducción Específica)

3.2. Arquitectura final propuesta

... Descripción de la arquitectura del sistema desarrollado, incluyendo una explicación detallada de las decisiones de diseño, las distintas componentes que lo conforman, y las razones detrás de la elección de esta arquitectura.

- - Diagrama de bloques de la arquitectura del chatbot propuesta.

3.2.1. Tecnologías suplementarias

... Herramientas y tecnologías adicionales necesarias para el trabajo.

3.2.2. Funcionamiento general

... Descripción completa del funcionamiento del pipeline, describiendo el camino de la pregunta desde que ingresa hasta que se genera la respuesta del chatbot.

3.3. Implementación

3.3.1. Preparación de los datos de contexto

... Especificación de los formatos requeridos y preprocesamiento de los datos para el entrenamiento del modelo

3.3.2. Sistema de reentrenamiento con historiales

... Implementación del sistema para reentrenar modelos usando historiales.

- - Diagrama de flujo del sistema de reentrenamiento.

3.3.3. Integración y ajustes finales del sistema

... Combinación de componentes y ajustes finales del sistema.

3.4. Despliegue de la interfaz de pruebas

... Implementación de la interfaz de usuario, utilizada para realizar las pruebas en el sistema durante la etapa de desarrollo.

- - Capturas de pantalla de la interfaz de pruebas.

Bibliografía

[1] IBM. *Natural Language Processing (PLN)*. https://www.ibm.com/mx-es/topics/natural-language-processing. Accedido el 17 de septiembre de 2024. 2023.

[2] Cobus Greyling. *RAG vs Fine-Tuning*. https://cobusgreyling.medium.com/rag-fine-tuning-e541512e9601. Accedido el 20 de septiembre de 2024. 2023.