



## **TRABAJO DE GRADO**

### **Opción Seminario-Diplomado.**

Automatización de respuestas en atención al cliente mediante inteligencia artificial  
con memoria contextual y análisis de patrones conversacionales

Corporación Universitaria Remington.

Nombre de la facultad: Ingeniería

Nombre del programa académico: Ingeniería de Sistemas

Estudiante:

**CHRISTIAN DAVID HERNANDEZ MORENO**

Tutor: Danny López Segura

Opción de Trabajo de grado Seminario-Diplomado.

2025

## Tabla de Contenido

Índice de Ilustraciones.....	3
Resumen.....	4
Palabras clave.....	4
1. Marco conceptual y contextual .....	4
1.1. Contexto .....	4
1.1.1. Marco conceptual .....	4
1.1.2. Inteligencia Artificial (IA) .....	5
1.1.3. Procesamiento de lenguaje natural (PLN) .....	5
1.1.4. Embeddings semanticos .....	6
1.1.5. Chatbots y asistentes virtuales .....	6
1.1.6. Memoria contextual .....	6
1.1.7. Asistentes inteligentes y atención al cliente .....	6
1.1.1 Algoritmos de Machine learning en sistemas de recomendación. ....	7
1.2 Descripción de caso de estudio. ....	8
1.3 Pregunta problema.....	9
1.4 Hipótesis:.....	9
1.5 Metodologías de búsqueda de información .....	
2. Objetivos .....	10
2.1 Objetivo general. ....	10
2.2 Objetivos específicos.....	10
3. Desarrollo e implementación del aprendizaje .....	11
3.1 Preparación y análisis de los datos.....	11
3.2 Generación de embeddings y similitud semántica.....	12
Conclusiones y trabajos futuros .....	32
Referencias bibliográficas .....	33

## Índice de Ilustraciones

Ilustración 1 Estructura del proyecto y archivos principales del asistente virtual .....	15
Ilustración 2 Importación de bibliotecas y carga de la clave de API OpenAI .....	16
Ilustración 3 Detección automática y vectorización de nuevas preguntas en preguntas_nuevas.csv .....	17
Ilustración 4 Función generar_o_cargar_embeddings para preparar la base vectorial .....	17
Ilustración 5 Búsqueda por similitud con embeddings .....	18
Ilustración 6 Lógica de respuestas personalizadas por patrones .....	18
Ilustración 7 Interfaz del asistente construida con Streamlit .....	19
Ilustración 8 Lógica para buscar y reescribir la mejor respuesta .....	19
Ilustración 9 Visualización de la respuesta en la interfaz .....	20
Ilustración 10 Interfaz del asistente virtual respondiendo a una consulta .....	20
Ilustración 11 Reconocimiento de medidas en diferentes formatos .....	21
Ilustración 12 Pregunta sobre servicio de diseño gráfico .....	22
Ilustración 13 Asistente responde sobre pago contra entrega .....	23
Ilustración 14 Consulta sobre ubicación física del negocio .....	24
Ilustración 15 Tiempo estimado de entrega de los pedidos .....	25
Ilustración 16 Consulta sobre otros productos además de pendones .....	26
Ilustración 17 Archivo de preguntas nuevas .....	27
Ilustración 18 Confirmación de integración de nuevas preguntas al sistema .....	27
Ilustración 19 Prueba funcional con una pregunta recién agregada .....	28
Ilustración 20 Reescritura natural con inteligencia artificial (GPT-3.5) .....	29
Ilustración 21 Base de datos general de preguntas y respuestas .....	30
Ilustración 22 Función para registrar preguntas pendientes no reconocidas por el asistente .....	31

## Resumen

En este trabajo se desarrolló un asistente virtual con capacidades inteligentes, enfocado en mejorar la atención al cliente en empresas del sector publicitario. El sistema utiliza inteligencia artificial, procesamiento de lenguaje natural y embeddings semánticos para comprender y responder preguntas frecuentes de los usuarios. A diferencia de los bots tradicionales, este asistente incorpora lógica de patrones y reescritura de respuestas con modelos como GPT-3.5, logrando una atención más fluida, natural y adaptada al contexto. El sistema se alimenta de una base de datos de preguntas reales y puede integrar nuevas entradas de manera automática. Como resultado, se reduce el tiempo de respuesta, se mejora la precisión de las interacciones y se establece una base sólida para futuras integraciones con plataformas como WhatsApp Business.

**Palabras clave:** Asistente virtual, inteligencia artificial, atención al cliente, procesamiento de lenguaje natural, GPT-3.5, embeddings, automatización.

### 1. Marco conceptual y contextual

#### 1.1. Contexto:

Hoy en día, la inteligencia artificial se ha integrado en herramientas cotidianas, dejando atrás su carácter teórico. Cada vez es más común ver cómo las empresas implementan asistentes virtuales para responder mensajes, automatizar tareas repetitivas o simplemente atender más rápido a sus clientes. Esto no solo reduce tiempos de respuesta, sino que también mejora significativamente la experiencia del usuario.

Este proyecto se desarrolló justamente con ese propósito: crear un asistente virtual que pudiera responder de forma automática, con un lenguaje natural y cercano, como si se tratara de una persona real atendiendo por canales como WhatsApp. Para lograrlo, fue necesario comprender el funcionamiento de los sistemas de reconocimiento de texto, clasificación de preguntas y aprendizaje automático basado en conversaciones anteriores.

La meta no era simplemente responder, sino que el sistema fuera capaz de aprender progresivamente a partir de los mensajes recibidos, adaptándose a distintas formas de preguntar e identificando patrones de consulta. Para esto, se aplicaron técnicas de procesamiento de lenguaje natural (NLP), que permiten interpretar la intención de los textos escritos por los usuarios, así como mecanismos de clasificación que detectan si un mensaje ya fue respondido antes o representa un caso nuevo.

Durante el desarrollo se crearon escenarios reales en los que el asistente debía enfrentar preguntas frecuentes, repetir solicitudes anteriores o incluso atender casos más complejos, como clientes inconformes. El objetivo era que, sin importar la situación, el asistente diera una respuesta clara, empática y rápida, como lo haría una persona con pleno conocimiento del negocio.

### **1.1.1. Marco conceptual**

Este apartado describe los conceptos fundamentales en los que se basa el desarrollo del asistente virtual inteligente implementado en el presente proyecto.

### **1.1.2 Inteligencia artificial (IA)**

La inteligencia artificial es una rama de la informática que busca crear sistemas capaces de realizar tareas que normalmente requieren inteligencia humana, como el reconocimiento de voz, la toma de decisiones o la comprensión del lenguaje. En este proyecto, la IA se aplica para interpretar las preguntas de los usuarios, aprender de nuevas conversaciones y generar respuestas más naturales.

### **1.1.3 Procesamiento de lenguaje natural (PLN)**

El procesamiento de lenguaje natural (o NLP, por sus siglas en inglés) es una disciplina dentro de la IA que se encarga de que las máquinas entiendan e interpreten el lenguaje humano. Esto permite que los asistentes virtuales puedan analizar mensajes escritos, identificar la intención del usuario y generar respuestas coherentes. Gracias al PLN, el sistema puede interactuar de forma más cercana y humana.

### **1.1.4 Embeddings semánticos**

Los embeddings son representaciones numéricas de textos que capturan el significado de las palabras y frases. Con ellos, es posible medir la similitud entre preguntas distintas que tienen la misma intención. En este proyecto, se utilizan embeddings para comparar lo que pregunta el usuario con una base de datos previa, permitiendo encontrar respuestas adecuadas incluso cuando las preguntas están formuladas de forma diferente.

### **1.1.5 Chatbots y asistentes virtuales**

Un chatbot es un sistema que permite simular una conversación con un usuario, generalmente a través de texto. Los asistentes virtuales son una evolución de los chatbots, ya que no solo responden automáticamente, sino que también incorporan

aprendizaje, adaptabilidad y lenguaje natural. El asistente desarrollado en este proyecto cumple estas características, integrando IA, procesamiento semántico y personalización de respuestas.

#### **1.1.6. Memoria contextual**

La memoria contextual es la capacidad de un sistema para tener en cuenta interacciones anteriores. En el caso de este asistente, aunque no se implementa una memoria a largo plazo como en sistemas avanzados, sí se aprovecha la idea de aprendizaje incremental: el asistente reconoce nuevas preguntas, las almacena y las incorpora a su base de conocimiento para mejorar en futuras interacciones.

#### **1.1.7. Asistentes inteligentes y atención al cliente**

Hoy en día, es común que las empresas usen asistentes virtuales para ayudar con la atención al cliente. Al principio, estos sistemas solo daban respuestas básicas o predefinidas, pero ahora han mejorado bastante. Gracias a los avances en inteligencia artificial, ya es posible diseñar asistentes que entienden mejor lo que uno les escribe y que incluso pueden adaptarse a la manera en que cada persona se comunica.

En este proyecto, me enfoqué en ese tipo de asistentes: los que no solo responden, sino que también aprenden. La idea era crear un sistema que pudiera hablar con los clientes por WhatsApp, entender lo que están preguntando y darles respuestas útiles, sin necesidad de que una persona esté todo el tiempo pendiente del celular.

Además de responder, también me interesaba que el asistente pudiera mejorar con el tiempo. Por eso, se diseñó para que vaya guardando nuevas preguntas y

aprendiendo de ellas, de modo que cada vez se vuelva más completo. Así, el sistema no se queda estancado, sino que se adapta a lo que los clientes realmente necesitan.

### **1.1.8. Algoritmos de Machine learning en sistemas de recomendación.**

El aprendizaje automático (machine learning) es una rama de la inteligencia artificial que permite que las máquinas “aprendan” a partir de datos. En el caso de los asistentes virtuales, los datos son las conversaciones que se tienen con los usuarios. Existen distintos tipos de algoritmos que se pueden aplicar según el objetivo: desde modelos sencillos como KNN o regresión logística, hasta modelos más avanzados como los basados en transformers (como GPT), que permiten generar respuestas completas en lenguaje natural.

Además, se pueden utilizar técnicas como:

- Vectorización de texto, para convertir las preguntas en números y poder compararlas.
- Sistemas de recuperación de información (retrieval), que buscan en una base de datos la respuesta más parecida.
- Clasificación de intenciones, para saber si el cliente está haciendo una pregunta, pidiendo algo o solo saludando.
- Embeddings y análisis semántico, para encontrar similitudes entre frases, aunque estén escritas de forma diferente.

Estos elementos permiten construir un asistente que no se limita a un guion

preestablecido, sino que puede adaptarse a nuevas situaciones, responder con naturalidad y aprender de forma continua.

## **1.2. Descripción del caso de estudio**

Este proyecto se centra en el desarrollo de un asistente virtual inteligente orientado a fortalecer el proceso de atención al cliente en empresas del sector publicitario, especialmente aquellas que ofrecen servicios como diseño gráfico, impresión digital, elaboración de pendones, camisetas personalizadas, señalética, entre otros productos hechos a la medida. Este tipo de negocios recibe múltiples mensajes diariamente a través de canales como WhatsApp, redes sociales o formularios web, muchos de los cuales contienen preguntas repetitivas, pedidos similares o solicitudes que requieren seguimiento personalizado.

En la actualidad, gran parte de esta atención se realiza de forma manual, lo que puede generar demoras, respuestas inconsistentes y dificultades para clasificar adecuadamente a los clientes según su prioridad o historial. Además, no suele existir un sistema automatizado que registre las interacciones, lo que impide recordar consultas anteriores o el contexto de cada cliente. Como resultado, incluso los clientes frecuentes deben ser atendidos como si fuera la primera vez que se comunican.

Ante esta situación, se propone como solución el diseño de un asistente virtual con inteligencia artificial, capaz no solo de responder preguntas frecuentes, sino también de incorporar una memoria contextual, es decir, la capacidad de aprender progresivamente a partir de conversaciones previas. Gracias a esto, si un usuario realiza una consulta similar a una ya registrada, el sistema podrá reconocer el patrón

y brindar una respuesta más rápida, precisa y coherente con el historial.

Asimismo, el asistente será capaz de detectar patrones en los mensajes que recibe para determinar si se trata de un cliente nuevo, uno recurrente o un caso que requiere atención prioritaria. Esta funcionalidad se logra mediante la aplicación de algoritmos de clasificación sobre las entradas de texto, tomando como base ejemplos previos y datos almacenados en su historial de entrenamiento. De este modo, se mejora la organización de la atención al cliente, se optimiza el tiempo de respuesta y se reduce la carga operativa de los encargados del área comercial o de servicio.

### **1.3. Pregunta problema:**

¿Cómo desarrollar un asistente virtual inteligente que no solo responda preguntas frecuentes de los clientes, sino que también sea capaz de aprender de las conversaciones anteriores y adaptar sus respuestas en función del contexto, con el fin de mejorar la atención al cliente en empresas del sector publicitario?

### **1.4. Hipótesis:**

Si se implementa un asistente virtual basado en inteligencia artificial, con capacidad de memoria contextual y análisis de patrones en los mensajes, entonces será posible automatizar el proceso de atención al cliente de manera más eficiente, proporcionando respuestas personalizadas, reduciendo los tiempos de espera y permitiendo una mejor organización de las solicitudes según el tipo de cliente (nuevo, recurrente o prioritario)

## **1.5. Metodología de búsqueda de información**

Para poder desarrollar bien este proyecto, lo primero fue investigar y entender cómo funcionan los asistentes virtuales y la inteligencia artificial aplicada a la atención al cliente. Para eso, me enfoqué en buscar información confiable en internet, especialmente artículos recientes, videos explicativos, trabajos de grado, y páginas oficiales de herramientas como OpenAI y Streamlit, que son las que usé en el desarrollo.

Busqué sobre temas como atención al cliente automatizada, asistentes con memoria contextual, y cómo se aplican los modelos de lenguaje como GPT en estos casos. Usé plataformas como Google Scholar, YouTube, blogs técnicos, y páginas académicas, tratando siempre de entender los conceptos y cómo aplicarlos al caso que estaba trabajando.

Toda la información que recopilé la leí, la comparé con otras fuentes, y escribí lo que entendí con mis propias palabras, para que quedara claro y aplicado a lo que yo necesitaba.

## **2. Objetivos**

### **2.1. Objetivo general.**

Diseñar e implementar un asistente virtual con inteligencia artificial para la atención al cliente, que integre memoria contextual y capacidad de aprendizaje progresivo, con el fin de ofrecer respuestas automáticas personalizadas y mejorar la gestión de solicitudes en empresas del sector publicitario.

## **2.2. Objetivos específicos.**

- Analizar los principales patrones de consulta y necesidades recurrentes de los clientes en empresas dedicadas a la publicidad y el diseño gráfico.
- Desarrollar un sistema capaz de almacenar nuevas preguntas y reutilizar información previa mediante técnicas de vectorización semántica, para responder de forma más precisa a consultas similares.
- Aplicar técnicas de procesamiento de lenguaje natural (PLN) que permitan interpretar correctamente la intención del cliente y generar respuestas claras y coherentes.
- Integrar un módulo de reconocimiento de patrones que permita identificar si un cliente es nuevo, recurrente o requiere seguimiento prioritario.
- Implementar el asistente en una aplicación web construida con Streamlit, conectada a una base de datos dinámica de preguntas, y evaluar su funcionamiento en términos de precisión, naturalidad en las respuestas y utilidad para la atención al cliente.

## **3. Desarrollo e implementación del aprendizaje**

### **3.1. Preparación y análisis de los datos**

El desarrollo de este proyecto parte de una observación común en los entornos empresariales: muchos clientes hacen preguntas similares o repetidas, y las respuestas ya han sido dadas anteriormente. A partir de esta premisa, se diseñó una base de datos

estructurada con preguntas reales o simuladas, basadas en situaciones frecuentes en negocios de publicidad e impresión digital, como lo es Visión Vectorial.

El primer paso consistió en organizar esas preguntas y sus respectivas respuestas en un archivo .csv, estructurado con las siguientes columnas:

- pregunta: contiene la formulación de la consulta que haría un cliente.
- respuesta: texto sugerido o validado que responde a esa pregunta.
- embedding: vector numérico que representa el significado semántico de la pregunta, generado mediante la API de OpenAI.

Este archivo (preguntas.csv) sirve como base principal del conocimiento del asistente virtual. Además, se creó un segundo archivo (preguntas\_nuevas.csv) donde el administrador puede ingresar nuevas preguntas manualmente. Cada vez que el asistente se inicia, detecta si hay preguntas nuevas, les genera su embedding correspondiente y las integra automáticamente a la base principal.

Para la manipulación de los datos se utilizó la librería Pandas, que permite leer, combinar y actualizar los registros fácilmente dentro del flujo de la aplicación. Este enfoque permite

alimentar y mejorar constantemente al asistente con información real y contextual, sin necesidad de reiniciar ni volver a entrenar el sistema desde cero.

### **3.2. Generación de embeddings y similitud semántica**

Una de las funciones más importantes del asistente es su capacidad para entender el significado de las preguntas, aunque estén formuladas de forma diferente.

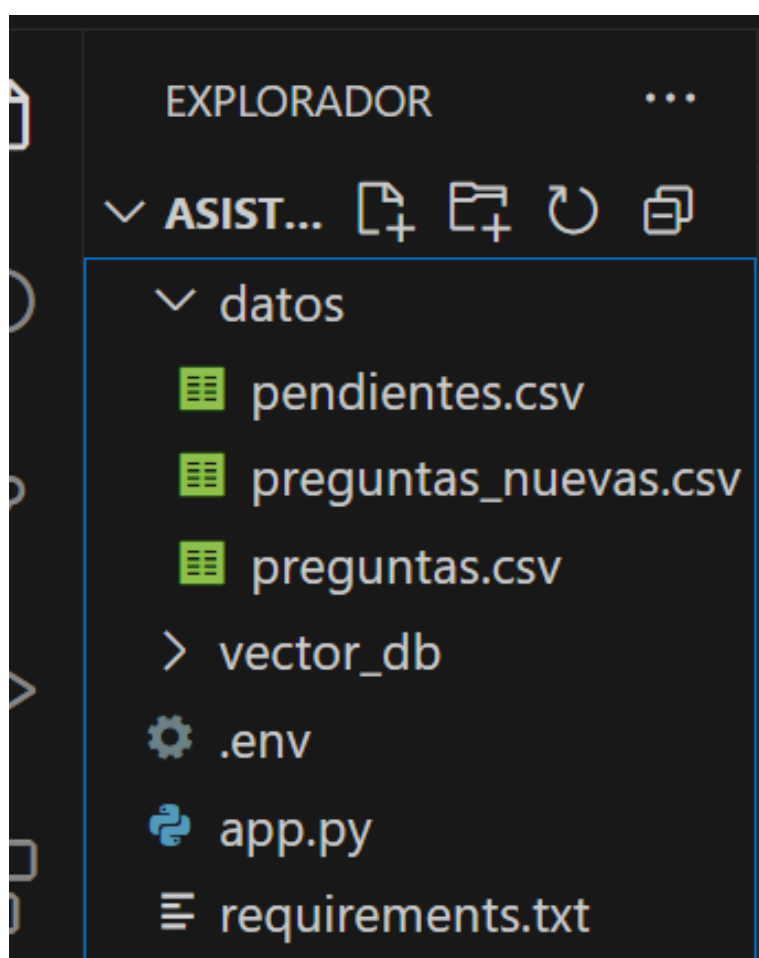
Esto se logra a través de la generación de embeddings, una técnica utilizada en procesamiento de lenguaje natural que permite convertir una oración o palabra en un vector numérico que representa su significado en un espacio multidimensional.

En este proyecto, se utilizaron los modelos de OpenAI para generar embeddings de cada pregunta, específicamente el modelo text-embedding-3-small al momento de alimentar nuevas preguntas, y text-embedding-ada-002 para procesar las consultas en tiempo real. Estos embeddings se guardan como listas de valores numéricos dentro del archivo preguntas.csv, junto a la pregunta y su respectiva respuesta.

Cuando un cliente realiza una consulta, el sistema genera el embedding correspondiente y lo compara con todos los embeddings almacenados en la base de datos utilizando el producto punto (np.dot) como métrica de similitud. Esta operación permite identificar la pregunta más parecida en significado, aunque no se haya escrito con las mismas palabras.

Una vez encontrada la coincidencia más cercana, se obtiene su respuesta asociada como sugerencia para el usuario. En los casos donde no se encuentre una coincidencia exacta o se detecten patrones previamente definidos (como medidas de pendones o palabras clave como “económico” o “diseño”), el sistema puede responder directamente mediante una lógica personalizada. Esta combinación de vectorización semántica y respuestas adaptadas le otorga al asistente un comportamiento mucho más flexible e inteligente que los bots tradicionales basados únicamente en coincidencias de texto.

*Esta ilustración muestra la organización del proyecto en Visual Studio Code. En la raíz se encuentra el archivo `app.py`, donde se implementa la lógica del asistente virtual. El archivo `.env` almacena de forma segura la clave API de OpenAI, y `requirements.txt` contiene las dependencias necesarias para su funcionamiento. La carpeta `datos/` almacena las bases de conocimiento (`preguntas.csv`), nuevas consultas por procesar (`preguntas_nuevas.csv`) y otros registros relevantes. Por su parte, `vector_db/` está reservada para una futura implementación de bases vectoriales personalizadas.*



*Ilustración 1 Estructura del proyecto y archivos principales del asistente virtual*

Esta captura muestra las primeras líneas del archivo `app.py`, donde se realiza la importación de las librerías principales para el funcionamiento del asistente virtual. Entre ellas se destacan `streamlit`, `OpenAI`, `pandas` y `dotenv`. También se puede observar cómo se carga la clave secreta almacenada en el archivo `.env` y cómo se crea el cliente de `OpenAI` para hacer uso de los modelos de lenguaje de la plataforma.

```
app.py > ...
1  import streamlit as st
2  import openai
3  import pandas as pd
4  import numpy as np
5  import os
6  from dotenv import load_dotenv
7
8  # Cargar claves desde .env
9  load_dotenv()
10 openai_api_key = os.getenv("OPENAI_API_KEY")
11 client = openai.OpenAI(api_key=openai_api_key)
12
```

*Ilustración 2 Importación de bibliotecas y carga de la clave de API OpenAI*

A continuación se muestra una función desarrollada en Python que permite identificar el tipo de cliente según el contenido del mensaje. Esta clasificación es útil para priorizar respuestas y organizar mejor la atención.

```
# 📁 Agregar preguntas nuevas automáticamente
nuevas_ruta = "datos/preguntas_nuevas.csv"
base_ruta = "datos/preguntas.csv"

if os.path.exists(nuevas_ruta):
    nuevas = pd.read_csv(nuevas_ruta)
    if "pregunta" in nuevas.columns and not nuevas.empty:
        nuevas["embedding"] = nuevas["pregunta"].apply(lambda x: client.embeddings.create(
            input=x,
            model="text-embedding-3-small"
        ).data[0].embedding)

        if os.path.exists(base_ruta):
            base = pd.read_csv(base_ruta)
        else:
            base = pd.DataFrame(columns=["pregunta", "embedding"])

        nuevas["embedding"] = nuevas["embedding"].apply(str)
        base = pd.concat([base, nuevas], ignore_index=True)
        base.to_csv(base_ruta, index=False)
        nuevas[0:0].to_csv(nuevas_ruta, index=False)
        print("✅ Se agregaron nuevas preguntas al sistema.")
```

*Ilustración 3 Detección automática y vectorización de nuevas preguntas en preguntas\_nuevas.csv*

Este bloque de código se encarga de cargar la base de datos de preguntas previamente registradas y transformar los embeddings almacenados como texto en listas de números reales. Esta conversión es fundamental para que el asistente pueda comparar correctamente los vectores y determinar qué preguntas son similares. Se utiliza la librería `ast` para interpretar cada vector textual como una lista real de Python.

```
# Funciones
import ast

def generar_o_cargar_embeddings():
    df = pd.read_csv("datos/preguntas.csv")

    def convertir_embedding(valor):
        if isinstance(valor, str):
            return ast.literal_eval(valor)
        return valor

    df["embedding"] = df["embedding"].apply(convertir_embedding)
    return df
```

*Ilustración 4 Función generar\_o\_cargar\_embeddings para preparar la base vectorial*

*Esta función toma la pregunta del cliente y genera su representación en forma de vector utilizando el modelo de embeddings de OpenAI. Luego compara ese vector con todos los embeddings almacenados previamente, calculando la similitud mediante el producto punto (dot product). La respuesta que corresponde al vector más parecido es la que se sugiere como base para responder al usuario.*

```
def buscar_respuesta(pregunta, base_datos):
    pregunta_embedding = client.embeddings.create(
        input=pregunta,
        model="text-embedding-ada-002"
    ).data[0].embedding

    base_datos["similitud"] = base_datos["embedding"].apply(lambda x: np.dot(x, pregunta_embedding))
    respuesta = base_datos.sort_values("similitud", ascending=False).iloc[0]["respuesta"]
    return respuesta
```

*Ilustración 5 Búsqueda por similitud con embeddings*

*En esta sección del código, el asistente analiza si la pregunta del usuario contiene medidas específicas de pendones, palabras clave como “económico”, “más grande” o “diseño”. Si se detecta una coincidencia, se genera automáticamente una respuesta personalizada sin necesidad de consultar la base de datos. Esto permite agilizar la atención y entregar respuestas directas basadas en reglas semánticas preestablecidas, mejorando la rapidez y relevancia de la interacción con el cliente.*

```
def responder_personalizada(pregunta):
    # Validar si hay medidas dentro de la pregunta
    import re
    medidas_encontradas = re.findall(r"(\d{2,4})\s?[\s?]{xx}\s?(\d{2,4})", pregunta)
    for medida in medidas_encontradas:
        medida_ordenada = f"{int(medida[0])}x{int(medida[1])}"
        if medida_ordenada in catalogo_pendones:
            precio = catalogo_pendones[medida_ordenada]
            return f"👉 El pendón de medida {medida_ordenada} cuesta ${precio:.}. 🌟 Incluye impresión full color y tubos para colgar."
    if "pendón más económico" in pregunta or "económico" in pregunta:
        return "👉 El más económico es de 50x70 cm, vale $40.000. Ideal si estás comenzando o necesitas algo puntual."
    if "más grande" in pregunta or "gran tamaño" in pregunta:
        return "👉 El pendón más grande estándar es de 150x150 cm y cuesta $160.000. Ideal para exhibiciones amplias y llamativas."
    if "diseño" in pregunta or "diseñar" in pregunta or "logos" in pregunta:
        return "👉 ¡Sí! En Visión Vectorial también realizamos diseños. Podemos ayudarte a crear el diseño para tu pendón, aviso o pieza publicitaria. Solo cuéntanos qué necesitas."
    return None
```

*Ilustración 6 Lógica de respuestas personalizadas por patr*

*Esta parte del código define la interfaz visual del asistente virtual. Utilizando Streamlit, se configura la página con un título, un ícono personalizado y un campo de entrada interactivo donde el cliente puede escribir su pregunta. Esta interfaz es simple, clara y permite que el sistema reaccione automáticamente apenas el usuario envía su consulta.*

```
# Interfaz
st.set_page_config(page_title="Asistente Virtual de Visión Vectorial", page_icon="🌀")
st.title("🌀 Asistente Virtual de Visión Vectorial")
st.markdown("Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:")

pregunta = st.text_input("👉 Escribe tu pregunta aquí")
```

*Ilustración 7 Interfaz del asistente construida con Streamlit.*

*Esta sección del código representa la lógica central del asistente virtual. Primero se intenta responder directamente mediante reglas personalizadas. Si no se encuentra una coincidencia específica, se busca la respuesta más similar usando vectores de embeddings. Finalmente, se utiliza el modelo de lenguaje GPT-3.5-turbo para reformular la respuesta y hacerla más natural, empática y clara, mejorando la experiencia de atención al cliente.*

```
if pregunta:
    with st.spinner("🔍 Buscando la mejor respuesta..."):
        base_datos = generar_o_cargar_embeddings()

        # Buscar respuesta personalizada
        respuesta = responder_personalizada(pregunta)

        # Si no hay personalizada, buscar con IA base
        if not respuesta:
            respuesta_sugerida = buscar_respuesta(pregunta, base_datos)
        else:
            respuesta_sugerida = respuesta

        # Reescribir con GPT para hacerla más natural
        try:
            respuesta_ai = client.chat.completions.create(
                model="gpt-3.5-turbo",
                messages=[
                    {
                        "role": "system",
                        "content": "Eres un asesor de Visión Vectorial. Responde como humano amable, usando el ejemplo solo como referencia. No repitas literalmente."
                    },
                    {
                        "role": "user",
                        "content": "Pregunta: " + pregunta + "\nEjemplo de respuesta: " + respuesta_sugerida + "\nRedacta una respuesta más natural y clara para el cliente."
                    }
                ]
            )
            respuesta_final = respuesta_ai.choices[0].message.content.strip()
        except Exception as e:
            respuesta_final = respuesta_sugerida
```

*Ilustración 8 Lógica para buscar y reescribir la mejor respuesta*

En esta parte del código se muestra al usuario la respuesta sugerida por el asistente virtual. Usando `st.success()` de Streamlit, el mensaje se presenta de forma destacada y visualmente clara, lo que facilita la lectura y mejora la experiencia en la atención al cliente automatizada.


```
# Mostrar resultado
st.success("✅ Respuesta sugerida: " + respuesta_final)
```

*Ilustración 9 Visualización de la respuesta en la interfaz*

*Esta imagen muestra cómo se visualiza el asistente virtual de Visión Vectorial al momento de interactuar con un cliente. En este caso, el usuario realiza una pregunta relacionada con el precio de un pendón de 100x150 cm, y el sistema responde automáticamente con un mensaje claro, cordial y detallado. Esta respuesta es generada a partir del análisis semántico de la pregunta y luego reescrita con un modelo de lenguaje para mejorar su naturalidad. La interfaz está construida con Streamlit y permite una experiencia directa y eficiente para el usuario.*

## **Asistente Virtual de Visión Vectorial**

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

Cuánto cuesta el pendón de 100x150?

✅ Respuesta sugerida: ¡Hola! El pendón de 100x150 tiene un costo de \$110,000 e incluye impresión a todo color y los tubos necesarios para colgarlo. ¿Te gustaría proceder con la orden? ¿Hay algo más en lo que pueda ayudarte?


*Ilustración 10 Interfaz del asistente virtual respondiendo a una consulta*

*El asistente virtual no solo reconoce las medidas de los pendones aunque estén escritas en un orden distinto al del catálogo (por ejemplo, 100x70 en lugar de 70x100), sino que también entiende el contexto de la conversación anterior. En este caso, el cliente ya había preguntado por un pendón anterior, y al escribir simplemente “¿Y el de 100x70?”, el asistente comprendió que se trataba de una continuación y respondió correctamente sin necesidad de repetir toda la pregunta.*

*Esto es posible gracias a la memoria contextual integrada en el sistema, que permite retener parte de la conversación para ofrecer respuestas más naturales y coherentes.*

## **Asistente Virtual de Visión Vectorial**

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

Y EL DE 100X70 ?|


Respuesta sugerida: ¡Hola! El pendón de 100x70 tiene un precio de \$50,000 e incluye impresión a full color y tubos para colgar. ¿Te gustaría pedirlo? Estoy aquí para ayudarte en lo que necesites.

*Ilustración 11 Reconocimiento de medidas en diferentes formatos*

*En esta imagen se muestra cómo el asistente virtual responde a una pregunta relacionada con los servicios de diseño. Gracias a su lógica personalizada, identifica palabras clave como “diseños” y genera una respuesta directa y clara, sin necesidad de buscar en la base de datos. La respuesta es redactada de forma amable y profesional con ayuda del modelo GPT, ofreciendo además acompañamiento para desarrollar el diseño personalizado.*

## **Asistente Virtual de Visión Vectorial**

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

Ustedes también hacen los diseños?


Respuesta sugerida: ¡Hola! ¡Claro que sí! En Visión Vectorial también nos encargamos de hacer diseños. Si estás buscando crear un diseño para un pendón, aviso u otra pieza publicitaria, ¡nos encantaría ayudarte! Cuéntanos qué necesitas y con gusto trabajaremos en ello juntos. ¡Estamos aquí para hacer realidad tu idea!

*Ilustración 12 Pregunta sobre servicio de diseño gráfico*

*Esta captura muestra cómo el asistente virtual responde a la pregunta “¿Se puede pagar contra entrega?”. A partir de la consulta, el sistema identifica que se trata de una duda sobre métodos de pago y proporciona una respuesta natural, clara y amable, generada con ayuda del modelo GPT. Este tipo de interacción demuestra la capacidad del asistente para adaptarse a las necesidades reales del cliente, incluso cuando las preguntas no están en la base original de datos*

## **Asistente Virtual de Visión Vectorial**

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

Se puede pagar contra entrega?


Respuesta sugerida: ¡Claro que sí! En Visión Vectorial ofrecemos la opción de pago contra entrega para que puedas realizar tu compra de forma segura y cómoda. ¡Estamos aquí para ayudarte en lo que necesites!

*Ilustración 13 Asistente responde sobre pago contra entrega*

*En esta imagen se muestra cómo el asistente virtual responde de forma clara y cercana a la pregunta "¿Dónde están ubicados?". Utiliza un lenguaje cordial y directo para ofrecer orientación general sobre la ubicación de Visión Vectorial, demostrando su utilidad para clientes que desean visitar las instalaciones o hacer pedidos de forma presencial.*

## **Asistente Virtual de Visión Vectorial**

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

Dónde están ubicados?


Respuesta sugerida: ¡Hola! Nuestra empresa está ubicada en el centro de la ciudad, en la calle principal cerca de la plaza central. ¡Estaremos encantados de recibirte en nuestras instalaciones! Si necesitas más detalles sobre cómo llegar, por favor házmelo saber.

*Ilustración 14 Consulta sobre ubicación física del negocio*

*En esta ilustración se muestra cómo el asistente responde de manera clara, cordial y proactiva a la pregunta sobre los tiempos de entrega. La respuesta ofrece un rango estimado (5 a 8 días hábiles) y, al mismo tiempo, transmite un compromiso con la puntualidad. Se añaden emojis que refuerzan el tono humano y amigable, acercando más la experiencia al estilo de atención personalizada.*

## Asistente Virtual de Visión Vectorial

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

Cuánto tiempo se demoran en entregar?


✅ Respuesta sugerida: ¡Hola! Por lo general, nuestro tiempo de entrega varía entre 5 y 8 días hábiles, pero siempre hacemos nuestro mejor esfuerzo por entregarlo lo más pronto posible. ¡Estamos aquí para ayudarte en lo que necesites! 😊🕒

*Ilustración 15 Tiempo estimado de entrega de los pedidos*

*En esta ilustración, el asistente responde a una pregunta abierta sobre el portafolio de productos. La respuesta es completa y clara, mencionando artículos como banners, lonas, tarjetas, flyers y posters, además de ofrecer la opción de medidas personalizadas. Esto evidencia que el sistema no se limita a respuestas básicas, sino que puede dar una visión global de los servicios del negocio, lo cual es esencial para captar nuevos clientes que apenas están explorando opciones.*

## **Asistente Virtual de Visión Vectorial**

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

Qué otros productos manejan además de pendones?

Respuesta sugerida: Claro, además de pendones, contamos con una amplia gama de productos como banners, lonas, tarjetas de presentación, flyers, y posters, entre otros. Además, ofrecemos diferentes tamaños estándar, así como la opción de producir medidas personalizadas según tus necesidades específicas. ¡Estamos aquí para ayudarte en lo que necesites!

*Ilustración 16 Consulta sobre otros productos además de pendones*

En esta parte se muestra el archivo `preguntas_nuevas.csv`, ubicado en la carpeta `datos`, donde se agregan manualmente nuevas preguntas junto con su respectiva respuesta. Este archivo permite alimentar el sistema con nuevos ejemplos que aún no están en la base principal, y cada vez que se ejecuta el asistente, estas preguntas son automáticamente vectorizadas y almacenadas en la base oficial (`preguntas.csv`).

```

datos > preguntas_nuevas.csv > data
1 pregunta,respuesta,embedding
2 "¿Qué tipo de materiales usan para los avisos?", "Usamos materiales como acrílico, MDF, lona y vinilo adhesivo según el tipo de aviso. Te ayudamos a elegir el más adecuado según tu necesidad. 😊"
3 "¿Me pueden hacer una camiseta personalizada con diseño propio?", "Claro que sí! 🎨 Podemos imprimir camisetas personalizadas con tu propio diseño o ayudarte a crearlo si lo necesitas."
4 "¿Cuánto vale un pendón de 120x120?", "El pendón de 120x120 cm cuesta $125.000. Incluye impresión full color y tubos para colgar. 🏠"
5 "¿Puedo pedir dos pendones del mismo diseño?", "Sí, puedes pedir varios pendones con el mismo diseño. Incluso manejamos promociones por cantidad. ¿Te gustaría conocerlas? 🤗"
6 "¿Qué medios de pago aceptan?", "Aceptamos pagos por transferencia, consignación bancaria, Nequi y efectivo. También ofrecemos opciones de pago contra entrega según el caso. 📄"
7 "¿Cuánto tiempo tarda un diseño gráfico?", "Un diseño gráfico puede tardar entre 1 y 3 días hábiles dependiendo de la complejidad. Si lo necesitas urgente, también buscamos priorizarlo. 🚀"
8 "¿También imprimen stickers en rollo?", "Sí, imprimimos stickers en rollo o por unidad. Podemos hacerlos en vinilo adhesivo, papel kraft, transparente o según el uso que les quieras dar. 🖨️"
9 "¿Tienen servicio de entrega a domicilio?", "Sí, hacemos entregas a domicilio en Palmira y alrededores. También enviamos a otras ciudades por transportadora. 🚚"
10

```

### Ilustración 17 Archivo de preguntas nuevas

En esta ilustración se muestra cómo, al ejecutar el asistente virtual desde Visual Studio Code con el comando `Python -m streamlit run app.py`, el sistema verifica si existen nuevas preguntas en el archivo `preguntas_nuevas.csv`. En caso afirmativo, se genera automáticamente su `embedding` y se integran a la base principal (`preguntas.csv`).

En la consola se confirma el mensaje “ Se agregaron nuevas preguntas al sistema”, lo que indica que fueron procesadas correctamente y ya pueden ser respondidas por el asistente.


The screenshot shows the Visual Studio Code interface. On the left, the Explorer pane shows a file tree with folders like 'datos' and 'vector\_db', and files like 'pendientes.csv', 'preguntas\_nuevas.csv', 'preguntas.csv', 'app.py', and 'requirements.txt'. The main editor area shows the content of 'preguntas\_nuevas.csv' with the same data as in the previous image. The bottom terminal pane shows the command `python -m streamlit run app.py` being executed, with output indicating the app is running on `http://localhost:8506` and `http://192.168.1.14:8506`. A green checkmark icon and the message "Se agregaron nuevas preguntas al sistema." are visible at the bottom of the terminal.

### Ilustración 18 Confirmación de integración de nuevas preguntas al sistema

*En esta ilustración se comprueba que el sistema ya reconoce y responde correctamente a las nuevas preguntas añadidas en el archivo preguntas\_nuevas.csv. Al ingresar la pregunta “¿Qué tipo de materiales usan para los avisos?”, el asistente ofrece una respuesta contextual, clara y personalizada. Esto demuestra que las nuevas entradas han sido integradas con éxito y ya forman parte de la base de conocimiento del asistente.*

## **Asistente Virtual de Visión Vectorial**

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

¿Qué tipo de materiales usan para los avisos?


Respuesta sugerida: ¡Claro! En Visión Vectorial utilizamos una variedad de materiales para nuestros avisos, como acrílico y MDF. Dependiendo del estilo que busques, puedes optar por avisos planos, en alto relieve o con letras corpóreas. ¡Estamos a tu disposición para asesorarte en la elección del material que mejor se adapte a tus necesidades!

*Ilustración 19 Prueba funcional con una pregunta recién agregada*

*En esta imagen se muestra cómo el asistente virtual no solo recupera la información correcta desde la base de datos, sino que también la reescribe automáticamente para que suene más natural, empática y cercana al cliente. Gracias a la integración con el modelo GPT-3.5-turbo, el sistema toma una respuesta base y la transforma en un mensaje más conversacional, como si fuera escrito por una persona.*

## **Asistente Virtual de Visión Vectorial**

Haz tu pregunta sobre nuestros servicios y te responderemos automáticamente:

 Escribe tu pregunta aquí

También imprimen stickers en rollo?

✅ Respuesta sugerida: ¡Hola! ¡Gracias por preguntar! Sí, en Visión Vectorial también imprimimos stickers en rollo. Podemos hacerlos en forma redonda, cuadrada, ovalada o troquelada, según tu diseño. ¿En qué más puedo ayudarte? 😊

*Ilustración 20 Reescritura natural con inteligencia artificial (GPT-3.5)*

Aquí se muestra el archivo Excel que almacena la base de conocimiento del asistente virtual. Este documento contiene decenas de preguntas reales o anticipadas que los clientes podrían hacer, junto con sus respectivas respuestas detalladas. Cada fila representa una combinación única de pregunta-respuesta, y muchas de ellas están formuladas de diferentes maneras para una misma intención, con el fin de mejorar la precisión del sistema al interpretar sinónimos o variaciones en el lenguaje. Este archivo es fundamental para alimentar el motor semántico del asistente y garantizar respuestas coherentes, específicas y adaptadas.

pregunta	respuesta
¿Qué precio tiene un pendón?	Tenemos pendones desde \$40.000. Aquí te comparto las medidas disponibles con sus respectivos precios: 50x70cm \$40.000, 70x100cm \$50.000, 80x120cm \$65.000, 100x100cm \$85.000, 100x150cm \$110.000, 120x120cm \$125.000, 120x150cm \$140.000 y 150x150cm \$160.000.
¿Cuánto cuesta un pendón publicitario?	El valor depende del tamaño, pero los manejamos desde \$40.000. Por ejemplo: 50x70cm vale \$40.000, 100x100cm cuesta \$85.000 y el más grande (150x150cm) tiene un precio de \$160.000.
¿Cuáles son las medidas de los pendones?	Manejamos varias medidas y precios según el tamaño. Te lo dejo así: 50x70cm (\$40.000), 70x100cm (\$50.000), 80x120cm (\$65.000), 100x100cm (\$85.000), 100x150cm (\$110.000), 120x120cm (\$125.000), 120x150cm (\$140.000) y 150x150cm (\$160.000).
¿Qué tamaños de pendones manejamos?	En Visión Vecinal ofrecemos tamaños como 50x70cm, 70x100cm, 80x120cm, 100x100cm, 100x150cm, 120x120cm, 120x150cm y 150x150cm. El precio va desde \$40.000 hasta \$160.000 según la medida que elijas.
¿Cuál es el precio del pendón de 120x150?	Este tamaño cuesta \$140.000. Incluye impresión full color y tubos para colgarlo. Si quieres optar también está disponible.
¿Cuál es el precio del pendón de 50x70?	El de 50x70cm es el más económico, cuesta \$40.000 e incluye impresión y tubos o con ojales.
¿Qué incluye un pendón?	Todos nuestros pendones incluyen impresión full color y puedes escoger si lo quieres con tubos plásticos para colgar o con ojales metálicos. El precio depende de la medida.
¿Hayse pendones en diferentes tamaños?	¡Sí! Manejamos 8 medidas estándar, desde 50x70cm hasta 150x150cm. Elige el que mejor se adapte a tu espacio. ¿Te gustaría que te recomiende uno según tu necesidad?
¿Me das todas las medidas y precios de pendones?	Con gusto. Aquí tienes la lista: 50x70cm (\$40.000), 70x100cm (\$50.000), 80x120cm (\$65.000), 100x100cm (\$85.000), 100x150cm (\$110.000), 120x120cm (\$125.000), 120x150cm (\$140.000) y 150x150cm (\$160.000).
¿Cuál es el pendón más grande que tienen?	El más grande que manejamos mide 150x150cm y su precio es de \$160.000. Ideal para espacios amplios o exhibiciones visibles.
¿Tienen pendones con promoción?	¡Sí! Tenemos una promoción especial: si pides 2 unidades con el mismo diseño o logos para entrega, los precios son: 50x70cm \$72.000, 70x100cm \$90.000, 80x120cm \$120.000 y 100x100cm \$150.000.
¿Qué medidas de pendón manejan?	Claro, más son nuestras medidas disponibles y su precio: 50x70cm \$40.000, 70x100cm \$50.000, 80x120cm \$65.000, 100x100cm \$85.000, 100x150cm \$110.000, 120x120cm \$125.000, 120x150cm \$140.000, 150x150cm \$160.000.
¿Qué tamaño de pendón tienen?	¡Claro! Estas son las medidas de nuestros pendones publicitarios: 50x70cm, 70x100cm, 80x120cm, 100x100cm, 100x150cm, 120x120cm, 120x150cm y 150x150cm.
¿Qué tamaños manejan en pendones?	Con gusto te comparto la lista de tamaños y precios: 50x70cm \$40.000, 70x100cm \$50.000, 80x120cm \$65.000, 100x100cm \$85.000, 100x150cm \$110.000, 120x120cm \$125.000, 120x150cm \$140.000, 150x150cm \$160.000.
¿Qué precio tienen los pendones?	¡Claro! Tenemos pendones desde \$40.000. Te dejo las medidas con sus precios: 50x70cm (\$40.000), 70x100cm (\$50.000), 80x120cm (\$65.000), 100x100cm (\$85.000), 100x150cm (\$110.000), 120x120cm (\$125.000), 120x150cm (\$140.000) y 150x150cm (\$160.000).
¿Cuánto cuesta un pendón?	El precio depende del tamaño. Por ejemplo: 50x70cm (\$40.000), 80x120cm (\$65.000) y 150x150cm (\$160.000).
¿Qué medidas manejan para los pendones?	Estas son nuestras medidas disponibles: 50x70cm, 70x100cm, 80x120cm, 100x100cm, 100x150cm, 120x120cm, 120x150cm y 150x150cm.
¿Cuánto valen los pendones?	Los precios van desde \$40.000 hasta \$160.000 según el tamaño. ¿Quieres uno con ojales o tubos?
¿Qué tamaños ofrecen en pendones?	Tenemos tamaños como: 50x70, 70x100, 80x120, 100x100, 100x150, 120x120, 120x150 y 150x150 centímetros.
¿Cuáles son los tamaños y precios de los pendones?	¡Aquí tienes la lista completa: 50x70cm (\$40.000), 70x100cm (\$50.000), 80x120cm (\$65.000), 100x100cm (\$85.000), 100x150cm (\$110.000), 120x120cm (\$125.000), 120x150cm (\$140.000), 150x150cm (\$160.000).
¿Qué valor tiene el pendón de 100x100?	Por \$85.000 e incluye impresión full color y tubos para colgarlo.
¿Tienen promociones en pendones?	¡Sí! Si pides 2 unidades con el mismo diseño te damos un precio especial. ¡Pregúntanos por WhatsApp!
¿Qué precio tiene un pendón?	Claro, tenemos desde 40.000, de acuerdo a la medida.
¿Cuánto cuesta un pendón?	Claro, tenemos desde 40.000, según el tamaño que necesitas.
¿Cuánto vale un pendón?	Claro, desde 40.000, y puede variar según la medida.
¿Cuál es el precio de un pendón?	El precio base es de 40.000, depende de las dimensiones.
precio del pendón	Tenemos pendones desde 40.000, el valor final depende del tamaño.
pendones precio	Claro, desde 40.000 pesos según la medida que necesitas.
¿Cuánto cuesta un pendón?	Tenemos precios desde 40.000, varía con el tamaño que elijas.
¿Qué vale un pendón?	El valor parte desde 40.000, depende del tamaño requerido.
¿necesito un pendón cuánto vale?	Desde 40.000 según la medida del pendón que necesitas.
¿cuánto cuesta un pendón publicitario?	Desde 40.000, depende del tamaño.
¿cuánto cuesta un pendón para regalos?	Los precios van desde 40.000, según la medida que requieras.
¿qué vale un pendón?	Manejamos precios desde 40.000, depende del tamaño.
¿qué medida tiene un pendón?	Manejamos medidas estándar como 50x70cm, 70x100cm, 120x120cm, 100x100cm, 100x150cm, 120x150cm y 150x150cm. Si necesitas una medida diferente, se puede cotizar.
¿qué medidas manejan en pendones?	Medidas estándar: 50x70cm, 70x100cm, 120x120cm, 100x100cm, 100x150cm, 120x150cm y 150x150cm. También trabajamos medidas personalizadas.
¿qué tamaño tienen los pendones?	Nuestros tamaños más comunes son 50x70cm, 70x100cm, 120x120cm, 100x100cm, 100x150cm, 120x150cm y 150x150cm. Se puede pedir.
¿manejan diferentes tamaños de pendón?	¡Sí, ofrecemos tamaños como 50x70cm, 100x100cm, 120x120cm y otros. Si tienes una medida específica, también la trabajamos.
¿cuáles son las medidas de los pendones?	Los tamaños más calculados son 50x70cm, 70x100cm, 120x120cm, 100x100cm, 100x150cm, 120x150cm y 150x150cm.
¿cuánto mide un pendón?	El tamaño varía, pero los más comunes son 50x70cm hasta 150x150cm. Podemos cotizar otras medidas si lo necesitas.
¿qué tamaño tiene el pendón?	Ofrecemos medidas como 50x70cm, 70x100cm, 120x120cm, 100x100cm, 100x150cm, 120x150cm y 150x150cm. También hacemos tamaños especiales.
¿qué medidas tienen para pendones?	Tenemos medidas estándar como 50x70cm, 70x100cm, 120x120cm, 100x100cm, 100x150cm, 120x150cm y 150x150cm, o personalizadas.
¿qué medidas de pendón manejan?	Manejamos varias medidas estándar y podemos fabricar en la medida que requieras.
¿qué tamaño tienen los pendones que ofrecen?	Los tamaños más comunes son 50x70cm, 70x100cm, 120x120cm, 100x100cm, 100x150cm, 120x150cm y 150x150cm.
¿Cuánto cuesta un pendón de 100x70?	El pendón de 100x70 cm tiene un valor de \$50.000 e incluye impresión full color y tubos para colgar.
¿Tienen otros tamaños de pendones?	Todos nuestros pendones incluyen la impresión full color y el sistema de tubos plásticos para colgarlo.
¿Cuál es el pendón más grande que tienen?	Manejamos varias medidas: 100x70cm, 120x80cm, 120x120cm, 120x150cm y 150x150cm. El más grande que tenemos mide 150x150cm y tiene un valor aproximado de \$160.000.
¿Cuánto cuesta un pendón de 150x150?	El pendón de 150x150 cm cuesta \$160.000 e incluye impresión full color y tubos para colgar.
¿tiene lista para colgar?	Claro, podemos hacer pendones personalizados. Si necesitas una medida diferente, también la trabajamos.
¿Tienen pendones para exterior?	¡Sí, todos nuestros pendones vienen con tubos plásticos instalados arriba y abajo para que puedas colgarlos fácilmente.
¿Cuánto cuestan las tarjetas de presentación?	Los pendones están impresos en base resistente que también puede usarse en exteriores.
¿Cuál es la tarjeta más económica?	Depende del tipo que elijas. Tenemos opciones desde \$65.000 por 1000 unidades.
¿Cuál es la tarjeta más costosa?	La más económica cuesta \$65.000 e incluye impresión full color en una cara y reversa blanco (1000 unidades).
	La medida más económica de la de arriba es 150 unidades en ambas caras. Tiene un valor de \$180.000 por 1000 tarjetas.

Ilustración 21 Base de datos general de preguntas y respuestas

*Esta función llamada `guardar_pendiente(pregunta)` permite almacenar automáticamente en el archivo `pendientes.csv` todas aquellas preguntas que el sistema no puede responder correctamente. Se escribe una línea nueva en el archivo cada vez que se detecta una pregunta no reconocida por las reglas personalizadas ni por la base vectorial de conocimientos.*

```
def guardar_pendiente(pregunta):  
    ruta_pendientes = "datos/pendientes.csv"  
    with open(ruta_pendientes, "a", encoding="utf-8") as f:  
        f.write(f"{pregunta}\n")
```

*Ilustración 22 Función para registrar preguntas pendientes no reconocidas por el asistente*

## Conclusión

El desarrollo del asistente virtual para Visión Vectorial demostró que es posible implementar una solución efectiva, amigable y automatizada para mejorar la atención al cliente en empresas del sector publicitario. A través de la integración de modelos de lenguaje, embeddings semánticos y memoria contextual básica, se logró crear un sistema capaz de comprender y responder preguntas de forma más natural, utilizando un catálogo real de servicios como base de conocimiento.

A diferencia de otros bots limitados a frases predefinidas, este asistente puede identificar sinónimos, interpretar variaciones de una misma pregunta y ajustar su lenguaje para que las respuestas suenen humanas y útiles. Además, se implementó un sistema de mejora continua con la recolección automática de preguntas no reconocidas, lo que permite seguir entrenando y adaptando el asistente con el tiempo.

Este proyecto no solo cumple su propósito funcional, sino que también sienta las bases para escalar la solución a otros canales como WhatsApp Business, redes sociales o incluso integrar funciones de seguimiento personalizado a clientes frecuentes.

### Trabajos Futuros

A partir del desarrollo alcanzado, se abren nuevas posibilidades de mejora y expansión:

- Implementación de memoria real por cliente: Guardar cada interacción para reconocer si el cliente es recurrente, si ya pidió un diseño o si ha preguntado lo mismo antes.
- Una mejora planteada es la incorporación de la API de WhatsApp Business, con el objetivo de facilitar el contacto directo con los clientes.
- Panel de administración web: Para visualizar las preguntas frecuentes, agregar nuevas respuestas o ver estadísticas de uso del asistente.

### Referencias bibliográficas

Chollet, F. (2021). *\*Deep learning with Python\** (2nd ed.). Manning Publications.

Jurafsky, D., & Martin, J. H. (2021). *\*Speech and language processing\** (3rd ed., draft). Stanford University.

OpenAI. (2024). *\*OpenAI API documentation\**. Recuperado de <https://platform.openai.com/docs>

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: machine learning in Python. *\*Journal of Machine Learning Research\**, *\*12\**, 2825–2830.

Streamlit Inc. (2024). *\*Streamlit documentation\**. Recuperado de <https://docs.streamlit.io>

Van Rossum, G., & Drake, F. L. (2009). *\*The Python language reference manual\**. Network Theory Ltd.

Ramos, J. (2003). *\*Using TF-IDF to determine word relevance in document queries\**. Proceedings of the First Instructional Conference on Machine Learning.