

Proyecto 1: Técnicas Clasificación

1st Bryan Campos Castro
Alajuela, Costa Rica
bryancampos20@gmail.com

2nd Miguel David Sánchez Sánchez
Heredia, Costa Rica
miguelsanchez712000@gmail.com

Abstract—Este informe presenta los resultados obtenidos en dos proyectos: la aplicación de filtros a imágenes y la implementación de un chatbot para una pizzería basado en embeddings. Se detallan los pasos seguidos para aplicar los filtros de imagen y las técnicas utilizadas en el desarrollo del chatbot, incluyendo la similitud del coseno para seleccionar respuestas.

I. INTRODUCCIÓN

En este trabajo, se exploran dos áreas fundamentales: el procesamiento de imágenes y la inteligencia artificial aplicada a chatbots. La primera parte aplica diferentes filtros de imagen para mejorar o modificar las características visuales. La segunda parte desarrolla un chatbot para una pizzería, que utiliza la biblioteca Spacy para transformar preguntas y respuestas en embeddings y seleccionar la respuesta más adecuada mediante la similitud del coseno.

II. APLICACIÓN DE FILTROS EN IMÁGENES

Se aplicaron cuatro filtros a una imagen, cuyos resultados se muestran en la Figura 1. Los filtros son:

- Blur Filter
- Sobel Edge Detection
- Canny Edge Detection
- Color Map

La imagen original y las versiones filtradas se pueden ver en la Figura 1, donde cada una está etiquetada según el filtro aplicado.

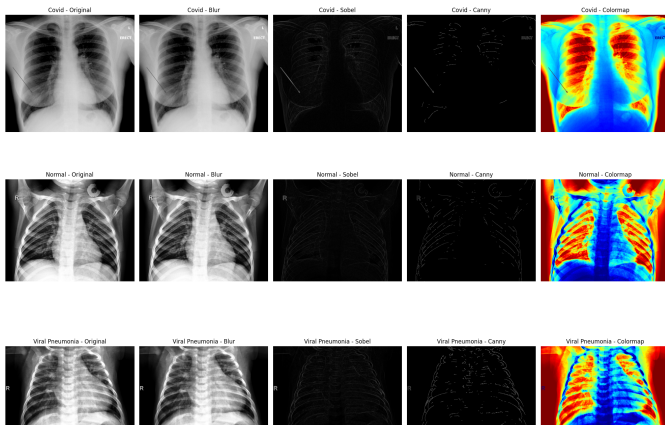


Fig. 1. Resultados de la aplicación de filtros a la imagen original.

A. Descripción de los Filtros Aplicados

Blur Filter: El filtro de desenfoque reduce el ruido en la imagen suavizando los detalles.

Sobel Edge Detection: Utiliza derivadas para destacar los bordes de la imagen, identificando los cambios abruptos en la intensidad.

Canny Edge Detection: Un algoritmo más sofisticado que también detecta bordes, pero incluye etapas de supresión no máxima y umbralización para obtener resultados más precisos.

Color Map: Técnica que asigna colores a valores en una imagen, facilitando la visualización de patrones y contrastes que no son evidentes en escala de grises.

B. Principales diferencias entre la imagen original y la imagen editada por cada filtro aplicado

- **Filtro Blur (Desenfoque Gaussiano):** La principal diferencia con la imagen original es que el filtro Blur suaviza la imagen, reduciendo el ruido y los detalles finos. Las estructuras y bordes en la imagen se ven más difusos, lo que ayuda a atenuar las imperfecciones menores. Como resultado, la imagen aparece menos nítida y más "borrosa".
- **Filtro Sobel (Detección de Bordes Sobel):** Este filtro resalta los bordes en la imagen, detectando cambios de intensidad en los ejes X e Y. Principalmente muestra los contornos y detalles donde ocurren transiciones abruptas de color o intensidad. En comparación con la imagen original, la imagen filtrada se ve oscura con líneas brillantes marcando los bordes principales. Este filtro es útil para identificar estructuras dentro de la imagen.
- **Filtro Canny (Detección de Bordes Canny):** Similar al filtro Sobel, el filtro Canny proporciona un resultado binario, donde los bordes son destacados de manera más precisa. En comparación con la imagen original, la imagen se convierte en una imagen en blanco y negro, donde los bordes son más delgados y mejor definidos que en Sobel. Esto produce un efecto de líneas sobre un fondo completamente negro.
- **Colormap (Mapeo de Color):** Este filtro aplica un mapeo de color sobre la escala de grises de la imagen, asignando diferentes colores a diferentes intensidades de píxeles. En comparación con la imagen original, la imagen se transforma en una representación en colores vivos, donde las intensidades más bajas se ven en colores fríos (azul) y las intensidades más altas en colores cálidos (rojo). Este mapeo de color es útil para visualizar la

distribución de intensidad en la imagen de una manera más intuitiva.

III. CHATBOT BASADO EN EMBEDDINGS

Para la implementación del chatbot de la pizzería, se utilizó la biblioteca Spacy, la cual ofrece un sistema eficiente de procesamiento de lenguaje natural sin costo por tokens. La motivación detrás del uso de Spacy es su capacidad de transformar tanto preguntas como respuestas en embeddings, permitiendo calcular la similitud de coseno para seleccionar la respuesta más adecuada a la consulta del cliente.

A. Descripción del Proceso

El chatbot procesa las consultas del cliente mediante los siguientes pasos:

- 1) Las preguntas y respuestas se transforman en embeddings utilizando Spacy.
- 2) Se limpian las preguntas del usuario, eliminando palabras "basura" que no aportan significado, como "what", "how", "where", "do", "you", "can", "I", "with", "a", "your", "pizza", "pizzas", "have". Esto se realiza para mejorar la precisión en la comparación de embeddings.
- 3) Se calcula la similitud de coseno entre el embedding de la consulta del cliente y las posibles respuestas almacenadas.
- 4) La respuesta con la mayor similitud de coseno se devuelve al cliente.

El chatbot devuelve una lista de 10 respuestas posibles, eligiendo la más relevante con base en la similitud calculada.

Listing 1. Ejemplo de código en Python

```
import spacy
import numpy as np
from sklearn.metrics.pairwise import cosine_similarity

nlp = spacy.load('en_core_web_md')

# Definir respuestas (ejemplo)
respuestas = ["..."] # Respuestas del chatbot

# Función para limpiar la entrada del usuario
def clean_input(user_input):
    stop_words = ["..."] # Palabras vacías a eliminar
    return ' '.join([word for word in user_input.lower().split() if word not in stop_words])

def find_best_response(user_input):
    cleaned_input = clean_input(user_input)
    user_embedding = nlp(cleaned_input).vector
    response_embeddings = np.array([nlp(r).vector for r in respuestas])
    similarities = cosine_similarity([user_embedding], response_embeddings).flatten()
    return respuestas[np.argmax(similarities)]

# Ejemplo de uso
respuesta = find_best_response("What time do you close?")
print(respuesta)
```

```
(env) envbryan.campos@Bryan-Campos IATarea3 % python ParteB.py
Pregunta: What time do you close?
Respuesta seleccionada: We are open from 10 AM to 10 PM. (Similitud: 0.59)

Pregunta: Can I pay with a credit card?
Respuesta seleccionada: You can pay with credit card, cash, or mobile payment. (Similitud: 0.79)

Pregunta: What sizes do your pizzas come in?
Respuesta seleccionada: Our pizzas come in three sizes: small, medium, and large. (Similitud: 0.44)

Pregunta: Do you deliver?
Respuesta seleccionada: We offer delivery services from 11 AM to 9 PM. (Similitud: 0.53)

Pregunta: Do you offer gluten-free pizzas?
Respuesta seleccionada: Yes, we have gluten-free pizza options. (Similitud: 0.62)

Pregunta: How can I contact you?
Respuesta seleccionada: You can call us at 123-456-7890. (Similitud: 0.39)

Pregunta: What is your best-selling pizza?
Respuesta seleccionada: Our most popular pizza is the Margherita. (Similitud: 0.74)

Pregunta: Where are you located?
Respuesta seleccionada: We are located at 123 Main Street. (Similitud: 0.41)

Pregunta: Where can I see your menu?
Respuesta seleccionada: You can find our menu online on our website. (Similitud: 0.44)

Pregunta: Do you have vegan options?
Respuesta seleccionada: We do offer vegan cheese for our pizzas. (Similitud: 0.56)

(env) envbryan.campos@Bryan-Campos IATarea3 %
```

Fig. 2. Resultados

B. Resultados

El chatbot ofrece respuestas precisas y rápidas, siendo ideal para su implementación en sistemas de atención al cliente de restaurantes como pizzerías. Este enfoque simplifica el manejo de consultas comunes y mejora la eficiencia en la atención al cliente.

IV. CONCLUSIÓN

Este trabajo ha mostrado cómo aplicar diferentes filtros a imágenes para mejorar o modificar sus características visuales, así como el uso de embeddings y la similitud del coseno para implementar un chatbot eficiente en el contexto de un asistente virtual para pizzerías.

V. RÚBRICA

Criterios	Puntuación máxima	Puntuación obtenida
Descarga y mostrar imágenes del dataset original	10	
Aplicar y analizar filtro Blur	15	
Aplicar y analizar filtro Sobel Edge Detection	15	
Aplicar y analizar filtro Canny Edge Detection	15	
Aplicar y analizar filtro seleccionado	15	
Estructura y claridad del informe	30	
Rúbrica Tarea NLP		
Criterios	Puntuación máxima	Puntuación obtenida
Definición de propósito de chatbot y respuestas	20	
Convertir las respuestas en embeddings y almacenarlo en un dataset de conocimiento	15	
Convertir la pregunta del usuario en embeddings	15	
Comparar los embeddings con similitud de coseno y mostrar la respuesta más acorde.	50	

Fig. 3.

REFERENCES

- [1] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- [2] Dua, D., & Graff, C. (2019). Pima Indians Diabetes Database. UCI Machine Learning Repository. Retrieved from <https://archive.ics.uci.edu/ml/datasets/pima+indians+diabetes>
- [3] Hastie, T., Tibshirani, R., Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.