

Tarea 02: Aplicación de filtros sobre imágenes y Chatbot

Objetivo:

El propósito de esta tarea es explorar y aplicar filtros sobre imágenes como técnica de preprocesamiento de datos para mejorar clasificadores utilizando CNN (Convolutional Neural Networks). Adicionalmente la tarea cuenta con un taller exploratorio de NLP, el cual pretende explorar las herramientas disponibles por la comunidad para aplicar embeddings sobre texto.

Filtros sobre imágenes:

Se espera que el estudiante sea capaz de investigar y comprender los diversos filtros para mejorar la extracción de características, reducción de ruido para mejorar el proceso de entrenamiento de modelos de aprendizaje automático basado en visión por computadora.

Pasos a seguir:

Realice la lectura del siguiente artículo:

<https://medium.com/@sumit-kr-sharma/image-filtering-in-computer-vision-ec60ec8a3e1>

Descargue el Dataset Covid-19 Image Dataset disponible en Kaggle:

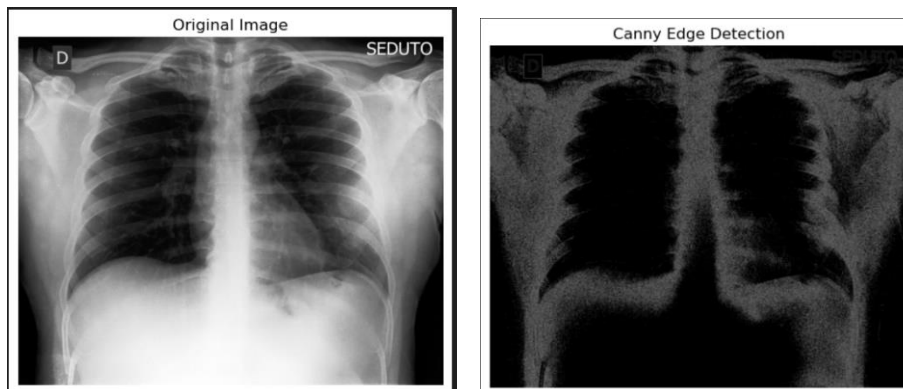
<https://www.kaggle.com/datasets/pranavraikokte/covid19-image-dataset>, a continuación tome 3 ejemplos del set de entrenamiento, uno por cada clase disponible.

A cada imagen aplique los siguientes filtros utilizando la biblioteca de OpenCV:

1. Blur Filter
2. Sobel Edge Detection
3. Canny Edge Detection
4. Seleccione uno de su preferencia.

Compare los resultados entre la imagen original y la imagen resultante de cada filtro, e indique con sus propias palabras las principales diferencias que usted logró observar.

Ejemplo de aplicar un filtro:

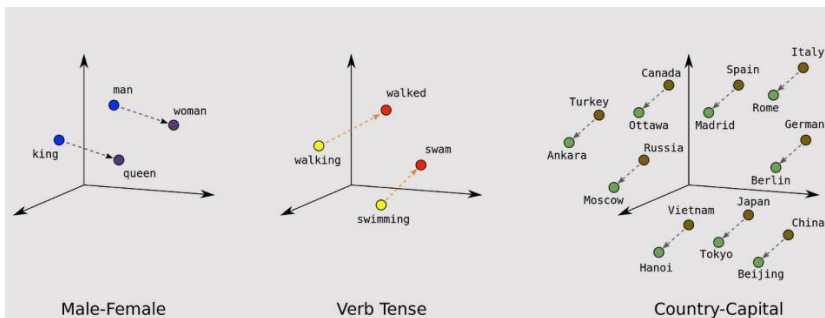


Chatbot:

Se espera que el estudiante sea capaz de investigar e implementar un chatbot basado en embeddings de texto, utilizando técnicas de comparación de texto como similitud de coseno.

Se recomienda realizar la siguiente lectura para comprender el funcionamiento de los embeddings:

<https://www.freecodecamp.org/news/understanding-word-embeddings-the-building-blocks-of-nlp-and-gpts/#:~:text=Word%20embeddings%20offer%20an%20effective,and%20sophistication%20of%20language%20models>



El estudiante debe definir el propósito y alcance de su chatbot. Puede ser un asistente virtual, un bot de atención al cliente, etc. Para cualquier caso va a contar un número limitado de respuestas (10). Y el chatbot debe recibir como entrada preguntas en Lenguaje Natural, basado en esta pregunta se debe seleccionar de las 10 respuesta la que tiene una mayor similitud con la pregunta realizada por el usuario. Las respuestas deben ser coherentes con las preguntas que indique el usuario.

Una vez diseñado el propósito del chatbot, piense en utilizar una biblioteca que le facilite la creación de embeddings (no se le pide entrenar ningún modelo), un ejemplo puede ser utilizar el API de OpenAI o cualquiera disponible en la comunidad de código abierto (e.g spaCy)

Pasos a seguir:

1. Defina el propósito y las respuestas de su modelo
2. Transforme sus posibles respuestas en embeddings y cree un dataset de conocimiento
3. Transforme la pregunta del usuario en un embedding
4. Compare el vector resultante del embedding de la pregunta con cada respuesta del dataset de conocimiento utilizando similitud de coseno y seleccione la que tenga mayor similitud.
5. Muestre la respuesta más similar al usuario.

Entrega:

Presentar los resultados obtenidos en un informe de Latex siguiendo la plantilla para artículos científicos de IEEE (usar Overleaf como recomendación) que incluya visualizaciones, snippets códigos utilizados (no es necesario incluir todo, solamente lo necesario en caso se requiera) y el análisis de los hallazgos, coloque la tabla de rúbrica que se encuentra en este documento en su informe.

Adjuntar el código implementado en un archivo separado para su revisión. El código debe ser presentado en un Jupyter notebook.

Coloque un zip en el espacio del TecDigital, que contenga el código fuente del Latex de su informe, el propio PDF del informe y el código fuente de su trabajo.

Fecha de Entrega: 24-09-2024

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 | |

NOTA: La revisión del chatbot se realizará de forma presencial para confirmar el adecuado comportamiento del NLP.

Referencias:

<https://learnopencv.com/edge-detection-using-opencv/>

<https://platform.openai.com/docs/guides/embeddings>

<https://ashutoshtripathi.com/2020/09/04/word2vec-and-semantic-similarity-using-spacy-nlp-spacy-series-part-7/>