



ESCUELA POLITÉCNICA NACIONAL



FACULTAD DE INGENIERÍA DE SISTEMAS

**Sistema de Recuperación de Información basado en
Reuters-21578**

Autor:

Bruce Soto

Quito, Ecuador - 19 de junio de 2024



Contenido

1. INTRODUCCIÓN	2
1.1. Objetivo	2
2. MARCO TEÓRICO.....	2
2.1. Caltech101	2
2.2. Redimensionamiento de las imágenes	2
2.3. Normalización de las imágenes	3
2.4. Red Neuronal Convolucional	3
2.5. VGG16	3
2.6. K-nearest neighbors.....	4
2.7. Precisión, recall, F1-score	5
3. DESARROLLO	5
3.1. Preprocesamiento.....	5
3.2. Extracción de Características	6
3.3. Indexación.....	7
3.4. Diseño del Motor de Búsqueda	8
4. EVALUACIÓN DEL SISTEMA	10
6. CONCLUSIONES Y RECOMENDACIONES	13
7. BIBLIOGRAFÍA	13



1. INTRODUCCIÓN

1.1. Objetivo

El objetivo principal de este proyecto es crear un buscador innovador que permita a los usuarios hacer preguntas usando imágenes en vez de palabras. Este sistema estará diseñado para encontrar imágenes similares dentro de una colección específica de imágenes. El proyecto se dividirá en varias etapas, las cuales se detallarán a continuación.

2. MARCO TEÓRICO

2.1. Caltech101

Caltech101 es un conjunto de datos utilizado en el campo de la visión por computadora y el aprendizaje automático. Fue desarrollado por investigadores del Instituto de Tecnología de California (Caltech) para evaluar y comparar el rendimiento de algoritmos de reconocimiento de objetos.

El conjunto de datos Caltech101 incluye imágenes de 101 categorías de objetos diferentes, cada una con alrededor de 40 a 800 imágenes. Este conjunto de datos es útil para entrenar y probar modelos de aprendizaje automático en tareas de clasificación de imágenes y detección de objetos.

2.2. Redimensionamiento de las imágenes

Las imágenes en el dataset Caltech101 pueden tener diferentes resoluciones y proporciones. Redimensionar todas las imágenes a un tamaño estándar de 224 x 224 píxeles asegura que cada imagen tenga las mismas dimensiones. Esto es esencial porque las redes neuronales requieren entradas con dimensiones consistentes para procesar los datos de manera eficiente.

Durante el redimensionamiento, la imagen original se ajusta para que se adapte al nuevo tamaño. Dependiendo del método utilizado, esto puede implicar el escalado de la imagen manteniendo su relación de aspecto, o el recorte y escalado para ajustarse exactamente al tamaño requerido.



2.3. Normalización de las imágenes

Las imágenes digitales se representan mediante píxeles, y cada píxel tiene un valor que típicamente varía en una escala de 0 a 255 en una imagen en escala de grises o en cada canal de color en una imagen RGB. La normalización convierte estos valores a una escala de 0 a 1. Esto se hace dividiendo los valores de los píxeles entre 255, el valor máximo posible.

Beneficios de la Normalización: Normalizar los valores de los píxeles ayuda a que los datos sean más homogéneos y a que el modelo de aprendizaje automático funcione de manera más eficiente. La normalización evita problemas de escala entre características, lo que facilita la convergencia durante el entrenamiento y puede mejorar la precisión del modelo. Además, puede ayudar a estabilizar el proceso de entrenamiento al evitar que algunas características dominen el aprendizaje debido a su escala más alta.

2.4. Red Neuronal Convolutiva

Una Red Neuronal Convolutiva (CNN, por sus siglas en inglés: Convolutional Neural Network) es un tipo de red neuronal artificial especialmente diseñada para procesar datos que tienen una estructura en forma de cuadrícula, como las imágenes. Las CNN son muy eficaces en tareas de reconocimiento y clasificación de imágenes, detección de objetos y análisis de video.

Las capas Convolutivas aplican un conjunto de filtros (o núcleos) a la imagen de entrada para extraer características locales. Cada filtro realiza una operación de convolución sobre la imagen, produciendo un mapa de características que resalta patrones específicos, como bordes, texturas y formas. La salida de esta operación es un mapa de activación que representa la presencia de ciertos patrones en diferentes regiones de la imagen.

Las CNN son capaces de aprender automáticamente las características importantes de las imágenes sin necesidad de ingeniería manual de características. Esto se debe a su capacidad para aprender representaciones jerárquicas a través de múltiples capas.

2.5. VGG16

Es un modelo de red neuronal convolutiva (CNN) que fue desarrollado por el Visual Geometry Group (VGG) de la Universidad de Oxford y presentado en el paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" en 2014. Este modelo es conocido por su arquitectura simple pero efectiva, y ha sido ampliamente utilizado en el campo de la visión por computadora.



VGG16 tiene una arquitectura profunda con 16 capas que tienen pesos entrenables. Estas capas incluyen 13 capas convolucionales y 3 capas completamente conectadas (fully connected). La red está diseñada para ser profunda pero relativamente simple en términos de arquitectura, utilizando solo dos tipos de capas: convolucionales y completamente conectadas.

A pesar de su arquitectura relativamente simple, VGG16 ha demostrado ser muy efectiva en tareas de clasificación de imágenes. Su profundidad permite que la red capture una gran variedad de características a diferentes niveles de abstracción.

2.6. K-nearest neighbors

El algoritmo K-Nearest Neighbors (KNN) es un método de aprendizaje supervisado utilizado para la clasificación y la regresión. Es un algoritmo sencillo pero muy eficaz que se basa en la idea de que los datos similares están cerca unos de otros en el espacio de características.

El parámetro K en KNN representa el número de vecinos más cercanos que se considerarán para tomar una decisión sobre la clasificación o la predicción de un nuevo dato. Por ejemplo, si K es 3, el algoritmo tomará en cuenta los 3 vecinos más cercanos al punto de datos para hacer la clasificación o la predicción.

Para encontrar los vecinos más cercanos, KNN calcula la distancia entre el nuevo punto de datos y todos los puntos en el conjunto de entrenamiento. Las distancias comunes utilizadas incluyen:

- Distancia Euclidiana: La distancia en línea recta entre dos puntos en un espacio multidimensional.
- Distancia de Manhattan: La suma de las diferencias absolutas de las coordenadas entre dos puntos.

Para la tarea de clasificación, el algoritmo asigna al nuevo punto de datos la clase que es más frecuente entre sus K vecinos más cercanos. Por ejemplo, si la mayoría de los vecinos más cercanos pertenecen a la clase "A", entonces el nuevo punto se clasificará como "A".

KNN es fácil de entender e implementar, ya que no requiere un proceso de entrenamiento explícito. En lugar de construir un modelo, KNN simplemente almacena los datos de entrenamiento y realiza cálculos en el momento de la predicción.



2.7. Precisión, recall, F1-score

La precisión, el recall y el F1-score son métricas comunes utilizadas para evaluar el rendimiento de modelos de clasificación en problemas de aprendizaje automático y procesamiento del lenguaje natural.

- Precisión: Se define como la proporción de casos positivos que el modelo identifica correctamente como positivos. Se calcula de la siguiente manera:
$$(\text{Verdaderos Positivos}) / (\text{Verdaderos Positivos} + \text{Falsos Positivos})$$
- Recall: Se define como la proporción de casos positivos reales que el modelo identifica correctamente. Se calcula de la siguiente manera:
$$(\text{Verdaderos Positivos}) / (\text{Verdaderos Positivos} + \text{Falsos Negativos})$$
- F1-score: Es una medida que combina la precisión y el recall en una única métrica. Se calcula de la siguiente manera:
$$2 * (\text{Precisión} * \text{Recall}) / (\text{Precisión} + \text{Recall})$$

3. DESARROLLO

3.1. Preprocesamiento

Este proceso involucra varias etapas, cada una con un propósito específico para limpiar y normalizar las imágenes. A continuación, se detallan los pasos necesarios:

Redimensionamiento de cada imagen

La imagen es ajustada a un tamaño específico. Este paso asegura que todas las imágenes tengan dimensiones uniformes, lo que es crucial para el procesamiento posterior y el entrenamiento del modelo. El tamaño de redimensionamiento se establece a un valor fijo, asegurando que todas las imágenes sean consistentes en sus dimensiones.

Normalización de cada imagen

Después de redimensionar la imagen, sus valores de píxeles se convierten a un formato numérico de punto flotante y se escalan para que estén en el rango de 0 a 1. Esto se hace dividiendo los valores de los píxeles, que originalmente están en un rango de 0 a 255, por 255. Esta normalización facilita el aprendizaje del modelo al hacer que los datos sean más manejables y homogéneos.



Preparación del conjunto de pruebas

Similar al conjunto de entrenamiento, la función de preprocesamiento se aplica a cada imagen en el conjunto de datos de prueba para asegurarse de que todas las imágenes estén redimensionadas y normalizadas de manera uniforme.

Las imágenes se agrupan en lotes del mismo tamaño que en el conjunto de entrenamiento. Esto es esencial para mantener la consistencia en el manejo de datos durante el entrenamiento y la evaluación.

También se utiliza prefetching para optimizar el rendimiento, asegurando que el procesamiento de datos en el conjunto de prueba sea eficiente y no se convierta en un cuello de botella.

3.2. Extracción de Características

3.2.1. Carga y configuración del modelo preentrenado

Carga del modelo preentrenado

Se carga un modelo de red neuronal convolucional preentrenado, específico para tareas de reconocimiento de imágenes. Este modelo ya ha sido entrenado en un gran conjunto de datos de imágenes y se utiliza como base para extraer características.

Se configura el modelo para que no incluya la capa final de clasificación, que normalmente se utiliza para clasificar las imágenes en diferentes categorías. En lugar de esto, el modelo devuelve las características intermedias generadas por la red.

Creación de un modelo de extracción de características

Se define un nuevo modelo que toma las mismas entradas que el modelo preentrenado, pero su salida es la de la última capa convolucional. Esto permite obtener los mapas de características generados por el modelo sin la parte final de clasificación.

3.2.2. Extracción de Características

Preparación para la extracción

Se inicializan listas para almacenar los mapas de características, las imágenes originales y las etiquetas correspondientes.



Extracción de características en lotes

Se procesa cada lote de imágenes del conjunto de datos utilizando el modelo de extracción de características. Para cada lote de imágenes:

Se obtienen los mapas de características del modelo.

Se almacenan los mapas de características en una lista.

Se guardan las imágenes originales y las etiquetas en listas separadas.

Consolidación de datos

Después de procesar todos los lotes, se combinan los mapas de características, las imágenes y las etiquetas en una sola estructura de datos. Esto se hace para tener una representación completa y unificada de todos los datos procesados.

3.2.3. Guardado de datos

Almacenamiento de características y datos:

Los datos procesados, incluyendo las características aplanadas, las imágenes originales y las etiquetas, se guardan en archivos. Estos archivos se utilizan para almacenar de manera eficiente los datos para su uso posterior, como la capacitación de modelos adicionales o la realización de análisis.

3.3. Indexación

Configuración del algoritmo k-NN

Se inicializa un índice de búsqueda para encontrar los k vecinos más cercanos en un conjunto de datos. En este caso, se establece que el número de vecinos a considerar será 10. Esto significa que para cualquier punto de datos que se consulte, el algoritmo buscará los 10 puntos más cercanos en el espacio de características.

Elección del algoritmo de búsqueda

Se selecciona un algoritmo específico para realizar la búsqueda de los vecinos más cercanos. En este caso, se utiliza el algoritmo `ball_tree`. Este algoritmo es adecuado para manejar grandes conjuntos de datos y realizar búsquedas eficientes en espacios de alta dimensión. Construye un árbol de bolas para organizar los datos en una estructura que facilita la búsqueda rápida de vecinos cercanos.



Entrenamiento del índice

El índice k-NN se ajusta utilizando las características aplanadas del conjunto de entrenamiento. Durante este paso, el algoritmo construye una estructura de datos que permite realizar búsquedas rápidas para encontrar los vecinos más cercanos de un punto dado. El entrenamiento del índice implica procesar las características de todas las muestras en el conjunto de entrenamiento y organizar la información de manera que la búsqueda sea eficiente.

3.4. Diseño del Motor de Búsqueda

3.4.1. Carga y preprocesamiento de una imagen externa

Carga de la imagen

Se carga una imagen desde una ruta de archivo específica y se redimensiona a un tamaño estándar. Esto asegura que la imagen tenga las dimensiones adecuadas para el procesamiento posterior.

Conversión y normalización

La imagen cargada se convierte en un arreglo numérico (un array) y se normaliza para que los valores de los píxeles estén en el rango de 0 a 1. Esto facilita el procesamiento de la imagen por el modelo.

Expansión de dimensiones

Se ajusta la dimensión del arreglo para que tenga una forma adecuada para ser ingresada en el modelo de extracción de características. Este paso es necesario para que la imagen pueda ser procesada por el modelo de manera correcta.

Retorno de resultados

La función devuelve la imagen preprocesada lista para el modelo y la imagen original para visualización.

3.4.2. Extracción de características de una imagen



Obtención de mapas de características

Se utiliza el modelo para predecir los mapas de características a partir de la imagen preprocesada. Esto extrae las representaciones de alto nivel que el modelo ha aprendido a identificar.

Aplanamiento y normalización

Los mapas de características se aplanan para convertirlos en un vector unidimensional. Este vector se normaliza para asegurar que todas las características estén en una escala comparable, facilitando la comparación con otros vectores de características.

3.4.3. Búsqueda de imágenes similares usando k-NN con la misma etiqueta

Normalización de la característica de consulta

Se normaliza el vector de características de la imagen de consulta para que esté en la misma escala que las características en el índice k-NN.

Búsqueda de vecinos cercanos

Se realiza una búsqueda en el índice k-NN para encontrar los vecinos más cercanos al vector de características de consulta. Se obtienen las distancias y los índices de estos vecinos en el conjunto de entrenamiento.

Filtrado por etiqueta

Se filtran los vecinos para seleccionar solo aquellos que tienen la misma etiqueta que el vecino más cercano. Se detiene el proceso cuando se han encontrado el número deseado de vecinos con la misma etiqueta.

3.4.4. Visualización de imágenes y vecinos más cercanos

Visualización de la imagen de consulta

Se muestra la imagen de consulta junto con su etiqueta para proporcionar contexto visual.



Visualización de vecinos más cercanos

Se muestra cada una de las imágenes de los vecinos más cercanos. Para cada imagen, se presenta su etiqueta y la distancia desde la imagen de consulta, proporcionando una visualización de cómo se relacionan las imágenes similares en función de sus características.

3.4.5. Ejecución del proceso

Extracción de una imagen y etiqueta del conjunto de entrenamiento

Se selecciona una imagen del conjunto de entrenamiento para usar como consulta. Se extrae también su etiqueta para la visualización.

Carga y preprocesamiento de la imagen de consulta

Se carga y preprocesa la imagen de consulta desde un archivo específico.

Extracción de características de la imagen de consulta

Se extraen las características de la imagen de consulta utilizando el modelo de extracción de características.

Búsqueda de vecinos más cercanos

Se buscan los vecinos más cercanos a la imagen de consulta utilizando el índice k-NN.

Visualización de resultados

Se muestran la imagen de consulta y las imágenes de los vecinos más cercanos, junto con información sobre sus etiquetas y distancias.

4. EVALUACIÓN DEL SISTEMA

Definición de métricas de evaluación

Se crean listas para almacenar los valores de precisión, recall y F1-score que se calcularán durante la evaluación del sistema.

Iteración sobre el conjunto de prueba

Se procesa cada imagen en el conjunto de datos de prueba de manera individual.



Extracción de características de la imagen de consulta

Para cada imagen del conjunto de prueba, se extraen sus características utilizando el modelo de extracción de características. La imagen se convierte en un formato adecuado y se obtiene un vector de características a partir del modelo.

Búsqueda de imágenes similares

Se realiza una búsqueda en el índice k-NN para encontrar las imágenes más similares en el conjunto de entrenamiento. Esto proporciona las distancias y los índices de las imágenes más cercanas.

Obtención de etiquetas de los vecinos

Se recopilan las etiquetas de las imágenes de entrenamiento que corresponden a los índices de los vecinos más cercanos encontrados en el paso anterior.

Cálculo de métricas:

Para cada imagen, se calculan las métricas de evaluación:

- Precisión: Proporción de etiquetas relevantes entre las etiquetas predichas.
- Recall: Proporción de etiquetas relevantes que fueron correctamente identificadas.
- F1-score: Media armónica entre precisión y recall, que proporciona una medida equilibrada.

Se utiliza la opción `zero_division=1` para manejar casos en los que no se predicen etiquetas relevantes, evitando errores en el cálculo.

Almacenamiento de métricas

Los valores de precisión, recall y F1-score se almacenan en listas para cada imagen evaluada.

Cálculo de promedios

Se calculan los promedios de precisión, recall y F1-score a partir de las listas acumuladas durante la evaluación del conjunto de prueba.



Resultados obtenidos

- Precisión promedio: 0.6845
- Recall promedio: 0.6845
- F1-score promedio: 0.6845

Estos resultados indican el rendimiento general del sistema de búsqueda de imágenes en términos de precisión, capacidad de recuperación y equilibrio entre ambos. La precisión, recall y F1-score promedio del sistema son aproximadamente 0.6845, lo que ofrece una medida general de la eficacia del sistema en la tarea de búsqueda de imágenes.

5. Interfaz de búsqueda de imágenes

La interfaz de búsqueda de imágenes es una herramienta básica diseñada para permitir a los usuarios consultar el sistema y obtener resultados visuales basados en la similitud con una imagen de referencia.

La interfaz es simple y permite al usuario cargar una imagen de consulta. El diseño está enfocado en la usabilidad y la funcionalidad, proporcionando una experiencia intuitiva para el usuario.

Cuando el usuario carga una imagen y envía la consulta, la imagen se envía al servidor backend para su procesamiento. El servidor está construido utilizando Flask, que maneja las solicitudes de la interfaz y coordina la búsqueda de imágenes similares.

Devolución de resultados

El servidor devuelve los resultados ordenados a la interfaz de usuario. Estos resultados incluyen las imágenes similares junto con información relevante, como las distancias de similitud.

Presentación de resultados

La interfaz muestra las imágenes similares al usuario de manera ordenada. Se presentan de mayor a menor similitud, facilitando la visualización de las imágenes más relevantes primero. La presentación es visualmente clara y permite al usuario identificar rápidamente las imágenes más similares a la consulta.



6. CONCLUSIONES Y RECOMENDACIONES

- Se ha La etapa de preprocesamiento, que incluye la redimensión y normalización de imágenes, es fundamental para asegurar la uniformidad de los datos de entrada y mejorar la precisión en la extracción de características.
- Utilizar el modelo VGG16 preentrenado ha sido efectivo para extraer características significativas de las imágenes, lo que contribuye positivamente a la precisión en la búsqueda de imágenes similares.
- El índice k-NN, implementado con el algoritmo de árbol de bolas, ha demostrado ser eficiente para encontrar vecinos cercanos en el espacio de características, proporcionando una búsqueda de imágenes efectivamente rápida.
- Los resultados de precisión, recall y F1-score, con un promedio de 0.6845, indican que el sistema de búsqueda de imágenes tiene un rendimiento aceptable, aunque hay margen para mejorar la precisión y la relevancia de los resultados.
- Se recomienda explorar técnicas adicionales de preprocesamiento, como la augmentación de imágenes, y evaluar otros modelos de redes neuronales para optimizar la calidad de la búsqueda. Además, ajustar el parámetro k en k-NN y mejorar la interfaz de usuario para incluir opciones de filtrado más detalladas pueden contribuir a una mejor experiencia de búsqueda.

7. BIBLIOGRAFÍA

- [1] DATASMARTS, 2020. *Conoce tus Datos #7: CALTECH-101*. [En línea] Available at: <https://datasmarts.net/es/conoce-tus-datos-7-caltech-101>
- [2] Cerotec estudios, 2022. *¿Qué es redimensionar una imagen, para qué te sirve y cómo hacerlo?*. [En línea] Available at: <https://www.cerotec.net/es/actualidad/que-es-redimensionar-una-imagen-para-que-te-s/n-545#:~:text=Cuando%20hablamos%20de%20redimensionar%20una,web%20o%20en%20redes%20sociales.>
- [3] KEEPCODING, 2024. *Preprocesamiento de imágenes de entrada*. [En línea] Available at: <https://keepcoding.io/blog/preprocesamiento-imagenes-entrada/>



[4] DataScientest, 2021. *Convolutional Neural Network : definición y funcionamiento*. [En línea]

Available at: <https://datascientest.com/es/convolutional-neural-network-es>

[5] DataScientest, 2021. *¿Qué es el algoritmo KNN?*. [En línea] Available at:

<https://datascientest.com/es/que-es-el-algoritmo-knn>