Cálculo de Porcentaje de Pixeles de Comida en Imagenes basados en modelos CNN. Deep Learning

Carlos Martí González
Universidad Internacional Menendez Pelayo, Madrid, España
Máster Investigación Inteligencia Artificial

20 de febrero de 2024

Resumen

Para la resolución de este problema, se emplean arquitecturas preentrenadas basadas en CNNs para estimar el porcentaje de píxeles de comida, haciendo refinamiento para minimizar el RMSE. La adaptación focalizada de ResNet muestra el mejor desempeño.

1. Introducción

En el contexto de la visión artificial, la tarea de identificar y cuantificar con precisión los elementos visuales en las imágenes digitales es fundamental, con aplicaciones prácticas que van desde el análisis nutricional hasta la gestión de recursos. Este trabajo se centra en el desafío específico de estimar el porcentaje de píxeles de alimentos en imágenes, utilizando como recurso principal la UNIMIB2016 Food Database2. Este conjunto de datos comprende imágenes diversificadas de bandejas de comida, junto con información detallada sobre el porcentaje de los alimentos.

La esencia de este trabajo reside en el desarrollo y la evaluación de modelos basados en Redes Neuronales Convolucionales (CNNs), reconocidas por su capacidad para aprender características visuales complejas. A través del diseño de modelos CNN, se busca lograr predicciones precisas de las proporciones de alimentos en las imágenes, evaluando su rendimiento mediante métricas establecidas como el Error Cuadrático Medio (MSE) y la Raíz del Error Cuadrático Medio (RMSE).

El enfoque metodológico se orienta hacia la exploración de tres estrategias diferenciadas, anticipando la utilización de arquitecturas CNN consolidadas y preentrenadas, como posibles aproximaciones. Este enfoque permitirá examinar la efectividad del aprendizaje por *transfer learning* y el *fine-tuning* de parámetros en este ámbito específico. La selección de estas estrategias tiene como objetivo no solo mejorar la exactitud en la estimación de las proporciones de alimentos, sino también aportar a la optimización de las técnicas de entrenamiento en el procesamiento de imágenes de alimentos.

Para asegurar la robustez y la generalización de los modelos, se integrarán técnicas como la regularización, el dropout, el early stopping, y la ampliación de datos, buscando enriquecer el proceso de entrenamiento y evitar el sobreajuste. Además, se adoptarán algoritmos de optimización eficientes, como Adam y RMSprop, para mejorar la eficacia del entrenamiento.

Este trabajo, por lo tanto, no solo se propone avanzar en la precisión de la estimación del porcentaje de comida en imágenes, sino también contribuir al desarrollo de metodologías más eficientes y adaptables para el análisis de imágenes en aplicaciones prácticas, marcando una contribución significativa en la intersección de la tecnología y las aplicaciones cotidianas.

2. Metodología

2.1. Preparación de Datos

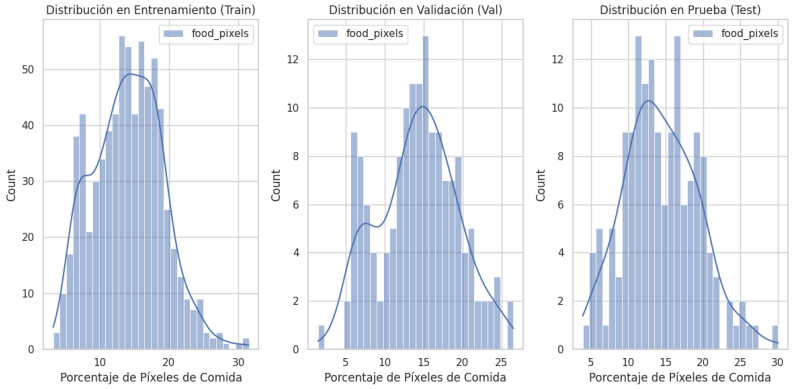
Para la preparación de datos, se ha utilizado como base, imágenes de comida con dimensiones espaciales de 640x480 píxeles, acompañadas de un archivo que contiene el nombre de cada imagen y el porcentaje correspondiente de alimentos que aparece en ellas. Estos datos proporcionan la base para entrenar, validar y probar los modelos de CNN desarrollados.

En este caso, los datos se han dividido en tres conjuntos:

- **Conjunto de Entrenamiento (69.91 %, 718 imágenes):** Utilizado para entrenar el modelo, permitiéndole aprender las características relevantes de las imágenes y cómo estas se correlacionan con los porcentajes de alimentos.
- **Conjunto de Validación (15.00 %, 154 imágenes):** Empleado para ajustar los hiperparámetros del modelo y realizar evaluaciones intermedias, sin influir directamente en el proceso de aprendizaje. Este conjunto ayuda a evitar el sobreajuste, asegurando que el modelo no solo memorice los datos de entrenamiento, sino que también sea capaz de generalizar a nuevos datos.
- **Conjunto de Prueba (15.09 %, 155 imágenes):** Reservado para evaluar el rendimiento del modelo después del entrenamiento y la validación. Este conjunto proporciona una medida imparcial del rendimiento del modelo en datos no vistos, reflejando su capacidad de generalización a situaciones reales.

Esta estrategia de división asegura que el modelo sea robusto y generalizable, evitando el sobreajuste y proporcionando una evaluación justa de su rendimiento, además de mantener la distribución del dataset original en cada uno de los conjuntos.

Figura 1: Dsitribución de los conjuntos de datos



Por otro lado, en cuanto al preprocesamiento de las imágenes, se han normalizado escalando los valores de los píxeles para que estén en el rango de 0 a 1. Esta normalización es importante porque ayuda a estabilizar el proceso de aprendizaje y a mejorar la convergencia del algoritmo de optimización al trabajar con valores de entrada más pequeños y manejables.

Además, se realiza un escalado de las imágenes desde su tamaño original de 640x480 píxeles a 224x224 píxeles. Este redimensionamiento reduce la carga computacional y asegura que las imágenes tengan un tamaño uniforme.

2.2. Diseño de la Arquitectura CNN

En cuanto al diseño de la arquitectura, se han aproximado tres configuraciones basadas en arquitecturas CNN preentrenadas, específicamente ResNet y EfficientNet, que son conocidas por su eficacia en tareas de clasificación y regresión en imágenes. A continuación, se justifica la elección de estas arquitecturas y se detalla cómo se adaptaron y personalizaron para el problema específico.

- **Aproximación 1 de ResNet50:** La primera configuración emplea ResNet50, una arquitectura que facilita el entrenamiento de redes con un número significativo de capas. Se seleccionó debido a su capacidad para preservar la información a través de las capas mediante conexiones residuales, lo que ayuda a evitar el problema de la desaparición del gradiente. En esta configuración, se congelaron todas las capas para mantener los pesos preentrenados en ImageNet, que contienen características visuales genéricas útiles para la tarea y, por tanto, realizar transfer learning para nuestra tarea específica. Se añadieron capas densas con activación 'relu' para la regresión, junto con capas de Dropout para evitar el sobreajuste, culminando en una capa de salida con activación lineal para predecir el porcentaje de píxeles de comida.
- **Aproximación 2 de ResNet50:** La segunda configuración de ResNet50 se diseñó para mejorar el rendimiento, ajustando la cantidad de capas entrenables al descongelar las últimas 10. Este enfoque permite un fine-tuning de las características visuales más específicas de las imágenes de comida. Se aumentó la tasa de Dropout y se ajustó la regularización para equilibrar la complejidad del modelo y el riesgo de sobreajuste, manteniendo una estructura similar de capas densas y activaciones 'relu', con la finalidad de mejorar la precisión en la predicción del porcentaje de alimentos.
- **Aproximación EfficientNet:** EfficientNet se eligió por su balance óptimo entre precisión y eficiencia computacional, adaptando la escala de la red (ancho, profundidad, resolución) de manera sistemática. En esta configuración, se congelaron las últimas 10 capas de la base preentrenada para preservar las características de nivel alto aprendidas en ImageNet, mientras se personalizaba el modelo para la tarea específica mediante la adición de capas densas con activación 'relu' y capas de Dropout, similar a las configuraciones de ResNet, pero aprovechando la eficiencia y efectividad de EfficientNet.

En todas las configuraciones, se ha utilizado la activación lineal en la capa de salida para la regresión del porcentaje de píxeles de comida, y se empleó la regularización L2 para controlar la complejidad del modelo. La elección del optimizador Adam se basó en su efectividad general en tareas de regresión. Tambhién se ha usado Global Average Pooling después del cada uno de los modelos preentrenados en lugar de aplanar los mapas de carcaterísticas completos, reduciendo así el nñumero de parámetros que llegan a las capas densas ayudando a minimizar el sobre ajuste al disminuir el número de parámetros.

En cuanto a la decisión de usar activaciones 'relu', han sido las que mejor rendimiento han dado, debido a su eficiencia computacional y su capacidad del gradiente. En cuanto al número de neuronas en capas densas, lo que se quería era que fueran poco a poco uniendo información y generalizando mejor para la tarea de regresión.

2.3. Estrategias de Entrenamiento

Para optimizar las estrategias de entrenamiento, se han adoptado técnicas de transfer-learning y fine-tuning, complementadas con métodos de regularización y augmentación de datos.

Inicialmente, en la primera aproximación, se ha aplicado transfer-learning, aprovechando las arquitecturas CNNs preentrenadas y congelando sus capas para utilizar las características visuales genéricas aprendidas de bases de datos extensas como ImageNet. Este enfoque permite que el modelo se beneficie de un conocimiento visual previo, facilitando una primera adaptación al nuevo conjunto de datos.

Posteriormente, se implementó fine-tuning, descongelando selectivamente las últimas capas de la red para permitir que el modelo se ajuste más finamente a las características específicas de las imágenes de alimentos. Este proceso de fine-tuning refina las representaciones de las características, mejorando la adaptabilidad del modelo a las particularidades del conjunto de datos objetivo y potenciando su rendimiento.

Además, para permitir que las capas añadidas aprendieran más rápido y hubiera una convergencia mejor, así cómo prevenir la desaparición o explosión del gradiente, se inicializaron los pesos de las capas densas con valores de una distribución normal con media 0 y varianza $2/n$, siendo n el número de neuronas de cada capa.

Para prevenir el sobreajuste y fomentar una mejor generalización, se han ido dando diversas técnicas de regularización y augmentación:

- **Dropout**: Integrado en las capas densas para disminuir la interdependencia entre los nodos, incrementando la robustez del modelo.
- **Regularización L2**: Empleada en las capas de salida para limitar la magnitud de los pesos, promoviendo la simplicidad del modelo y reduciendo la propensión al sobreajuste.
- **Early Stopping**: Utilizado para cesar el entrenamiento cuando no se detectan mejoras en la métrica de validación de RMSE tras un número determinado de épocas, evitando así el aprendizaje de ruido.
- **ReduceLROnPlateau**: Esta función ajusta la tasa de aprendizaje ante la ausencia de mejoras en la métrica de validación, permitiendo refinamientos más sutiles en los pesos del modelo sin cambios abruptos que puedan inducir sobreajuste.
- **Augmentación de datos**: La augmentación de los datos se llevó a cabo con 'ImageDataGenerator', aplicando transformaciones aleatorias como rotaciones, desplazamientos y cizallamientos, así como volteos horizontales y verticales. Se incorporó una función personalizada llamada 'add_noise' para inyectar ruido en las imágenes, incrementando la robustez del modelo frente a variaciones en los datos. Esta estrategia de augmentación enriquece el conjunto de entrenamiento, exponiendo el modelo a una gama más amplia de escenarios y reforzando su capacidad de generalización.

2.4. Optimización y Evaluación

En cuanto a la optimización y evaluación de las aproximaciones, nos enfocamos en varias áreas críticas para garantizar la precisión y eficacia del modelo.

En la elección del optimizador, se ha decidido usar Adam (Adaptive Moment Estimation) para estas aproximaciones debido a que se basa en su eficacia probada al combinar las fortalezas de AdaGrad y RMSProp. Esta sinergia lo hace ideal para manejar datos de alta dimensionalidad y conjuntos de datos extensos, características inherentes a las CNN en tareas de análisis de imágenes.

Las configuraciones que se han usado para el optimizador son:

- **Tasa de Aprendizaje (LR)**: Se ha optado por una LR inicial de 0.001, un estándar recomendado para Adam, equilibrando la rapidez de convergencia y la estabilidad del modelo, pero ajustándolo en las diferentes epochs.
- **Primer Momento**: Se establece en 0.9 para aprovechar la inercia de gradientes anteriores, suavizando las actualizaciones y favoreciendo una convergencia estable.
- **Segundo Momento**: Fijado en 0.999, este parámetro asegura una actualización estable de los pesos frente a cambios bruscos en los gradientes, permitiendo una adaptación más granular de la LR.

Para medir la eficacia de estos modelos, hemos utilizado el Error Cuadrático Medio (MSE) proporcionando una cuantificación del error promedio, elevando al cuadrado las diferencias entre predicciones y valores reales, lo que resulta útil para entender la magnitud del error y su raíz (RMSE) que ofrece una perspectiva más matizada del error, usando así métricas clave en la valoración de modelos de regresión y el baseline pedido es de 4.8.

3. Resultados

En cuanto a los resultados de las diferentes aproximaciones, se va a analizar detenidamente los resultados de las configuraciones basadas en ResNet y EfficientNet frente al umbral de RMSE.

En la aproximación 1 de ResNet, aplicando transfer learning y adaptaciones específicas, se alcanzó un RMSE de 5.2549 en el conjunto de test, mejorando respecto al baseline en la fase de validación. La aproximación 2 de ResNet, con fine-tuning en capas avanzadas, mostró una mejora adicional, registrando un RMSE de 4.6889 en el test, igualando su rendimiento en la validación. Por su parte, EfficientNet, a pesar de su excelente rendimiento en entrenamiento y validación con un

RMSE de 4.1315 y 3.6755 respectivamente, experimentó un incremento significativo en el conjunto de test a 12.7677, indicando posibles problemas de generalización.

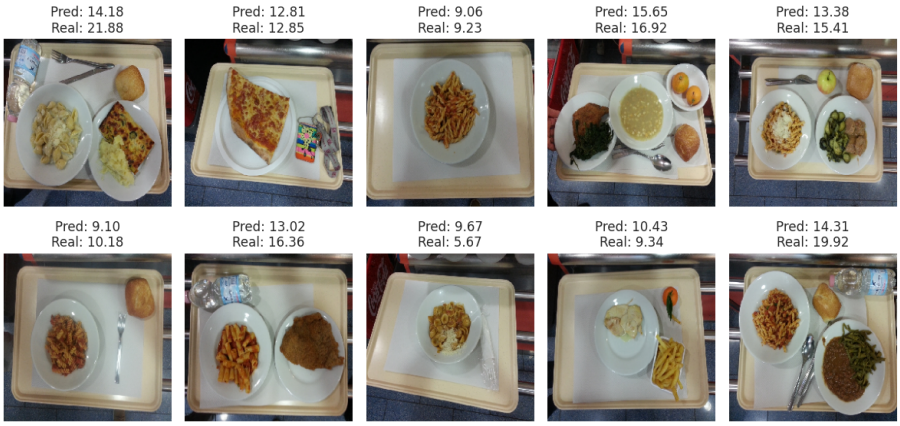
Aproximación	Entrenamiento	Validación	Test
ResNet 1	6.0899	4.7988	5.2549
ResNet 2	5.0122	4.6889	**4.6889**
EfficientNet	**4.1315**	**3.6755**	12.7677

Cuadro 1: Resultados de RMSE para las diferentes aproximaciones

Las modificaciones específicas y el fine-tuning implementados han permitido superar el valor de referencia de RMSE, demostrando la capacidad de estas aproximaciones para la tarea específica de cuantificar el porcentaje de píxeles de comida en imágenes.

Algunos ejemplos de predicción podrían ser:

Figura 2: Ejemplo de predicción de la mejor aproximación



Con todos los datos obtenidos y analizados, se ha visto que la segunda aproximación de ResNet parece generalizar mejor con respecto a las demás aproximaciones. Esto es debido a que aunque la aproximación de EfficientNet tiene un muy buen resultado en validación, en test empeora mucho, no siendo así en el otro caso, que siempre mantiene un buen resultado, superando al baseline, tanto en validación, como en test.

En cuanto a los desafios y limitaciones encontrados, se ha observado, por un lado, por un lado, las limitaciones en cuanto a los datos ya que los modelos dependen críticamente de la calidad y diversidad de los datos de entrenamiento. Si el conjunto de datos no es lo suficientemente variado o extenso, puede no capturar todas las variaciones posibles que el modelo encontrará en el mundo real, lo que limita su capacidad para generalizar. Este desafío es particularmente relevante en tareas como la cuantificación de píxeles de comida en imágenes, donde la variabilidad en la apariencia de los alimentos, la iluminación, y la perspectiva de las imágenes puede ser significativa.

Por otro lado, las dificultades computacionales, debido a que la implementación de fine-tuning en las capas avanzadas en modelos muy profundos, aunque efectiva en la mejora del rendimiento, implica un aumento en la complejidad computacional y los requisitos de memoria. Este aspecto puede limitar la escalabilidad del enfoque y su aplicabilidad en entornos con recursos limitados. Además, el fine-tuning es un proceso delicado que requiere una cuidadosa selección de capas a ajustar, tasas de aprendizaje y otros hiperparámetros. Una elección inadecuada puede llevar fácilmente a un rendimiento subóptimo o incluso al deterioro del modelo.

4. Conclusión

El trabajo presentado aborda el desafío de cuantificar el porcentaje de píxeles de alimentos en imágenes utilizando modelos basados en Redes Neuronales Convolucionales (CNNs), centrando el estudio en la UNIMIB2016 Food Database. Se han explorado tres configuraciones de arquitecturas CNN, incluyendo variantes de ResNet y EfficientNet, aplicando técnicas de transfer learning y fine-tuning, complementadas con estrategias de regularización y augmentación de datos para mejorar el rendimiento y la generalización de los modelos.

Los resultados obtenidos muestran un rendimiento variado entre las aproximaciones. La aproximación 2 de ResNet, que incorpora fine-tuning en las últimas capas, demostró ser la más efectiva, alcanzando un RMSE de 4.6889 en el conjunto de test y superando el valor de referencia establecido. Por otro lado, aunque EfficientNet mostró un rendimiento prometedor en las fases de entrenamiento y validación, experimentó una disminución significativa en el conjunto de test, lo que sugiere problemas de generalización.

La superioridad de la aproximación 2 de ResNet subraya la importancia del equilibrio entre la preservación de conocimientos visuales genéricos a través del transfer learning y la adaptación a características específicas del conjunto de datos mediante el fine-tuning. Esta estrategia no solo mejoró la precisión en la estimación del porcentaje de píxeles de alimentos sino que también destacó la relevancia de una cuidadosa selección de hiperparámetros y técnicas de regularización para evitar el sobreajuste y promover una robusta generalización.

Sin embargo, el trabajo también enfrenta desafíos inherentes a la naturaleza de la tarea y las limitaciones de los datos y la complejidad computacional. La variabilidad en la apariencia de los alimentos y las condiciones de las imágenes plantean dificultades significativas para la generalización del modelo, mientras que el fine-tuning en arquitecturas profundas eleva la demanda de recursos computacionales, lo que podría limitar la aplicabilidad de estos enfoques en entornos con recursos restringidos.

En conclusión, este trabajo demuestra la eficacia de las CNNs y las estrategias de entrenamiento avanzadas en la mejora de la precisión y la generalización del modelo. Sin dejar de lado, la necesidad de abordar los desafíos relacionados con la diversidad de los datos y las limitaciones computacionales para avanzar aún más en este campo.

5. Enlaces

https://colab.research.google.com/drive/1u5OuoFsmUOlNWlErrRGYb9p_-yuNss87?usp=sharing
https://drive.google.com/file/d/1_veNbXDYL-h7Fdcl0qo0w3UMjXYl0DAS/view?usp=sharing