Food Item Recognition from Images

Jason Nader, Christopher Maalouf, Nour Kaddoura, Ameen Agha

May 6, 2024

GitHub Repository: https://github.com/MA4LOUF/261-FInal.git

Abstract

This report outlines our comprehensive approach to food item recognition from images using advanced deep learning techniques. We dive into the complexities of dataset preprocessing, model development, and optimization strategies. Our results exhibit substantial improvements in accuracy, validating the success of our methodology.

1 Introduction

Machine learning, specifically image recognition has witnessed significant advancements owing to the expansion of deep learning methodologies. In this project, we embark on the challenging task of food item recognition from images, a domain with far-reaching applications. Our objective is to accurately classify images of various food items into nine distinct categories using a deep learning model. To achieve this, we employ data cleaning and processing, training, diagnosis, hyper-parameter tuning and model selection.

2 Dataset

The dataset used in this project is stored in a folder called "data," which contains images grouped into nine different categories. Each category has a unique ID, detailed in the category.txt file. This file helps link each image to its respective category. The images were first converted into grayscale to simplify them, resized to a consistent size of 224x224 pixels, and converted back to color (RGB format). These steps ensured all images were standardized and neatly organized by category in the output folder.

To address the uneven number of images across different categories, we employed several techniques. We did undersampling, where we reduced the number of images in larger classes to balance the dataset. We also used oversampling to increase the number of images in smaller classes. Additionally, we applied class weighting based on the frequency of each class, giving more importance to underrepresented classes during our analysis. This comprehensive approach to data preprocessing and balancing was crucial for enhancing the performance of our image classification algorithm.

3 Models

Our journey towards developing an effective food item recognition model begins with the exploration of various deep learning architectures. Initially, we experimented using convolutional neural network (CNN) models available in popular frameworks such as Keras. However, recognizing the need for greater flexibility and control over model architecture, we transition to PyTorch, a powerful deep learning framework renowned for its flexibility and scalability. Leveraging the PyTorch framework, we carefully crafted a custom CNN architecture tailored to the difficulties of our dataset.

Model training constitutes a crucial phase in our methodology, wherein we employ a grid search strategy to optimize hyperparameters such as learning rate and batch size. The grid search encompasses a wide range of hyperparameter values, allowing us to identify the optimal configuration that maximizes model performance.

In the subsequent stages of the project, after preprocessing the dataset, the focus shifted towards model training and evaluation. Firstly, utilizing PyTorch's data handling capabilities, the dataset was transformed and organized for training and validation sets using the 'torch.utils. data' module. A transformation pipeline consisting of resizing, conversion to tensor, and normalization was applied to each image. The dataset, sourced from the "Cleaned data" directory which we created containing all the clean data was loaded into memory allowing access during training.

Subsequently, the dataset was split into training and validation sets using a split ratio of 80 percent for training and 20 percent for validation. Data loaders were then instantiated for both sets, facilitating efficient batch-wise processing during model training.

Moving forward, a modified ResNet architecture was employed for the classification task, leveraging transfer learning by initializing with pre-trained weights on the ImageNet dataset. The architecture was adapted to accommodate the specific number of classes in the project (nine classes) by replacing the classifier's last layer. This modification enabled the model to output predictions for the nine distinct classes present in the dataset. To train and fine-tune the model, a grid search approach was adopted to explore different combinations of hyperparameters, focusing initially on the learning rate. The model's performance was evaluated on the validation set using a custom evaluation function, computing both loss and accuracy metrics. The best-performing combination of hyperparameters was identified based on the highest validation accuracy. These hyperparameters, along with the corresponding accuracy, were reported, providing insights into the model's performance and parameter optimization process.

Finally, the weights of the best-performing model were saved to facilitate future utilization and deployment. The saved model weights serve as a valuable asset, encapsulating the learned representations and enabling seamless integration into production environments for real-world inference tasks. Overall, the combined efforts encompassed data preprocessing, model architecture selection, hyperparameter tuning, and model evaluation, culminating in a robust framework for image classification tailored to the project's requirements.

4 Results

Our efforts are manifested in the form of compelling results that underscore the efficacy of our approach having around 85 percent accuracy. Through meticulous experimentation and attentive evaluation, we achieve remarkable improvements in model accuracy and robustness. Our optimized model exhibits superior performance, accurately classifying food items with high precision. Notably, the model excels in generalization, demonstrating its ability to accurately classify unseen images with high confidence levels.

We present a comprehensive analysis of our results, including detailed accuracy metrics. Our model's performance is thoroughly evaluated on both training and validation datasets, showcasing its robustness and generalization capabilities. Moreover, we conducted extensive studies to clarify the impact of individual hyperparameters on model performance, providing valuable insights for future optimization endeavors.

5 Conclusion

In conclusion, our adventure to tackle the challenging task of food item recognition from images has resulted in promising results, underscoring the potential of deep learning methodologies in addressing real-world challenges. Through our strategies, we have successfully developed a robust and accurate food item recognition model. Looking ahead, the avenues for future exploration are plentiful. Further enhancements to model architecture, including the integration of attention mechanisms and ensemble learning techniques, hold promise for unlocking new frontiers in food item recognition. Additionally, the exploration of larger and more diverse datasets can potentially bolster model performance and foster greater generalization across varied domains and scenarios.

6 References

1. https://pytorch.org/docs/stable/index.html