Hope Tsai and An Nguyen
Deep Learning
Professor Bálint Gyires-Tóth
12 May 2022

Classifying Indian Dishes Using Deep Learning

Introduction (Hope)

Food is an integral part of everyday life, and yet, the current digital environment does not encourage much exploration in the world of cuisine. If one wants to research an unfamiliar dish, one needs to know its name. (If the dish belongs to a culture with a different writing system, then the difficulties in researching only increase.) Thus, if one becomes intrigued by a photograph in a menu or by a friend's dinner choice, one may be at a loss.

Deep learning can alleviate this issue. To explore this problem, we focused on Indian cuisine, because it has many image datasets online. We aimed to create a model that could classify images of Indian dishes, and experimented with random models, pre-trained models, and transfer learning. Ultimately, our best model, when run on the second dataset, was approximately 80% accurate.

Previous Solutions (An)

https://dl.acm.org/doi/pdf/10.1145/2986035.2986042?casa_token=ORzYOcGvouQAAAAA:vXBsxaz0pcUoiZRzlOhmpONJs-TtSYzOi1VUIo5TZosmFZuaWNHQS0ruo9fWRJXhR5_dmFxZk0Q
Hassannejad, Hamid, et al. "Food image recognition using very deep convolutional networks." *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management*. 2016.

This study finetuned Google's Inception V3 network and evaluated its effectiveness in classifying food images from three dataset. It is the most similar to our approach because we also used Inception V3 for our pretrained model.

https://d1wqtxts1xzle7.cloudfront.net/62937416/I080301495220200413-58760-ga87xi-with-cover-page-v2.pdf?Expires=1652626784&Signature=L34DZgvlHyKbqWzT5rT64npjEbXfCEsecCJNluiQeFBKq4boSUlC23AO-uV7R0aUCBspEOzi4RYRxfDfsDYqAKKJOya4dnDTrvToxyXVmtlN68ZLg~FmFeT5jz0088akb3dolZafeRzz-pYDMGwSyM3wvGcN70Fo9cKCwygjTuYnJGQUc3lKAPnX8eOd72XD9E2fM0hTM7jpFQEcnjhP2kvQR3erDYb6XSRpOo5VJ5Lk3xdZpNtwSL~q3aATBd1UFQkGvZxM8ZvSrXZS5wBzUlDW6hyB5L7NuIr7aUcu58Iz3gNJDmHXYs5PAPUS48jD4SQbpWN353wfoMd1G7s-Sg__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA
Burkapalli, Vishwanath C., and Priyadarshini C. Patil. "Segmentation and Identification of Indian food items from Images."

This study is the closest to our topic, classifying Indian food images. However, their system is based on segmenting the image to place them into categories. The study only consists of five different types of food, idli, vada, dosa, chapati and rice, rather than 91 category of Indian food.

https://www.sciencedirect.com/science/article/pii/S2214785322014444?casa_token=bxnAXfdSWeUAAAAA:lnq7xTvGyjyvSTvJ2OxGVyKDdNaeHScUmITq1lkn1-WQvsQ_JSVEf2--PksCjiSuzEBHa4Ly
Sathish, Suriyakrishnan, et al. "Analysis of Convolutional Neural Networks on Indian food detection and estimation of calories." *Materials Today: Proceedings* (2022).

This study evaluates the accuracy of InceptionV3, VGGNet16, and a basic CNN in detecting Indian cuisine and their calories. 11 classes were used to train these models. Out of the three models, InceptionV3 performs the best.

Data (Hope)

We found datasets by searching through Kaggle, an online platform that hosts competitions and stores code and datasets related to deep learning. The first dataset is titled "Indian Food Images Dataset" (by creator Sourav Banerjee) and contains 4000 images spread out across 80 classes (distinct Indian dishes). It also contains a text file of the dishes' names. The images are perfectly distributed; each class has 50 images.

The second dataset is titled "The-massive-Indian-Food-Dataset" (by creator Anshul Mehta), and contains 4770 images spread out across 15 classes. Unlike in the first dataset, the images in the second dataset are not well distributed. For example, there are 667 files in the Kofta-Resized class, but only 170 images in the dosa-resized class. Thus, there is a disadvantage to each dataset. The first dataset has roughly the same amount of information (number of images), but many more classes to choose from. On the other hand, the second dataset is skewed towards a few classes.

The two datasets share four dishes: biryani, jalebi, kofta, and naan. The combined dataset contains 8770 images and 91 classes.

Proposed Methods (Hope and An)

We ran two experiments on dataset 1, dataset 2, and the combined dataset. Within each experiment, we tested the same layer composition on a random model and a pre-trained model. With the pre-trained models, we used transfer learning and only trained on our added layers. In total, we ran six experiments.

In the first experiment, our layer composition consisted of a GlobalAveragePooling2D layer and two Conv2D layers. We used teh Adam optimizer and set the learning rate to 0.1. The patience for the early stopping callback was 10. The number of epochs was 100.

In the second experiment, our layer composition consisted of two Conv2D layers, three BatchNormalization layers, two MaxPooling2D layers, one GlobalAveragePooling2D layer, and one Dense layer. We used the SGD optimizer, and set the learning rate to 0.3. The patience for the early stopping callback was 20. The number of epochs was 200.

Evaluation Method (An)

We evaluated the results using the classification report in sklearn which consists of precision, recall, and f1-score, and the confusion matrix.

In the first experiment, the both models performed the best on for class number 51 when both datasets are used and it performed the best on class number three when dataset two is used. The second experiment also achieved the best results for those two classes. This might be due to their overrepresentation in the dataset.

In both experiment, the two models have low precision and recall scores (<0.5) which means our classifier has a high number of False negatives. This might be due to imbalanced class. The only exception was the results of the pretrained model with the second dataset, which has a recall score of 0.78 (>0.5). Overall, the pretrained model has higher recall scores than the random model.

The f1-score share a similar trend as it makes use of the combination from precision and recall scores. Since our f1-score is low, with the exception of the pretrained model for the second dataset, this indicates that our models are not good at classifying images. Hence, to improve the performances, we would need to fine tune our models or try different pre-trained ones.

Results (Hope and An)

In the first experiment, with the pre-trained model's predictions, the first dataset generally had a categorical accuracy of 0.25. The second dataset had 0.78, and the combined dataset had 0.33. With the random model's predictions, the first dataset had a categorical accuracy of 0.0056, the second dataset had 0.20, and the combined dataset had 0.11.

In the second experiment, with the pre-trained model's predictions, the first dataset generally had a categorical accuracy of 0.30. The second dataset had 0.81, and the combined dataset had 0.56. With the random model's predictions, the first dataset had a categorical accuracy of 0.58, the second dataset had 0.50, and the combined dataset had 0.29. More details can be found in the notebook. For both experiments, the highest scores in the classification occurred when the model ran on the second dataset.

Discussion (Hope and An)

In the first experiment, the pre-trained model had the highest accuracy for the second dataset. But for the random model, the second dataset only has a slightly higher accuracy than the other. The worst accuracy belonged to the random model when run on the first dataset. Overall, the pre-trained model, which fine-tunes Inception v3, performs better than the random model.

In the second experiment, with the pre-trained model, the second dataset had the highest accuracy out of the three datasets. As it had a high number of data and fewer classes to distinguish between, the model likely had fewer "choices" to make and more "background knowledge" when it classified images. Surprisingly, this did not help the model when it ran on the random model; all of the datasets had similar accuracies. The worst accuracy belonged to the pre-trained model when run on the first dataset. The first dataset actually experienced higher accuracies under the random model. Perhaps the weights of Xception interfered with classification.

One limitation of the project was the quality of the datasets. If we had more time, we would decrease the number of classes in the first dataset to discover if having fewer choices truly does increase model accuracy. Moreover, doing so would result in a combined dataset that is less skewed towards the first dataset. We would also search Kaggle and other online resources for more datasets, so that our model could be trained on more diverse images and dishes. Regarding the training portion of the project, we would investigate more pre-trained models to see if Xception is truly the most accurate. As we did not have much GPU, we had to choose which models to run and keep in the Google Colab. Ultimately, we hope to create a deep learning algorithm that can be generalized to classify dishes from datasets of any cuisine.