Bridging the information gap in environmental impact of grocery shopping: A Computer Vision Approach to measure CO2 emissions of grocery products

Cristobal Maturana
Stanford
cmaturan@stanford.edu

Abstract

In this paper, a study on the performance of several computer vision models for the task of grocery image classification is presented. The original dataset contains 5125 images classified in 43 different classes and split into training, validation and test sets. Data augmentation and data mining techniques were used to increase the volume of the training data. A transfer learning approach using a MobileNetV2 network pre-trained on Image Net resulted in the highest accuracy of 0.86 in the test set

1. Introduction

The food system accounts for 34% of the global GHG emissions, according to a study conducted by McKinsey. In the path towards achieving net zero emissions globally for 2050 and keeping global warming at noncritical levels, it is critical to quickly tackle these emissions. The main barrier that environmentally aware consumers face, which prevents them from making more sustainable grocery purchasing decisions, is the lack of information. Currently, understanding the environmental impact of groceries requires extensive and time-consuming online research.

A potential solution to this problem could be a mobile app that can provide the consumer with the estimated CO2 emissions, by taking a picture of a product in the supermarket. The goal of this project is exploring a first step towards this solution, by building a model that receives a grocery product as an input and outputs the name of the 'class' to which the product belongs. The task of classifying groceries is a particularly complex computer vision problem, largely due to the high variety and constant evolution of product formats and packages [1]. In this work, different convolutional neural network architectures are tested to categorize grocery products that belong to 43 different classes.

2. Related work

Several publications in the lasts 5-10 years have focused on the classification of grocery products. Jund et al [2] in 2016 built a grocery database with 25 classes of products and proposed a computer vision method to identify the class of a given picture using an adaptation of AlexNet, achieving an accuracy of 78.9%. This was one of the first works on this topic. Ciocca et al. [3] in 2021 proposed a multi-task learning network to classify groceries in 84 different classes, achieving an accuracy of 87.7%. In the same year, Filax et al [1] proposed a radically different approach, which treats grocery classification as an open set recognition, with the goal of designing a system that could work in a real environment with thousands or millions of SKUs. This method uses triplet mining and contrastive learning, similar techniques used for face recognition. Many authors misunderstand the concept of anonymizing for blind review. Blind review does not mean that one must remove citations to one's own work—in fact it is often impossible to review a paper unless the previous citations are known and available.

3. Dataset and features

The main grocery dataset used for this project [4] contains 5125 images from packaged goods, fruits and vegetables, divided into 81 fine-grained classes and 42 coarse-grained classes. The classification task performed in this project is focus on the coarse-grained classes. The images are in .jpg format and they come in two shapes: squared pictures of 348x348 pixels and rectangular images of 348x464 pixels. The data is also split into a training set folder (2,640), a validation set (296) and a test set (2,485). In the original dataset, the validation set contained images of only 37 of the 43 coarse-grained classes in the training and test sets.

In the data preprocessing stage, a redistribution of the training, validation and test sets was done to ensure that the validation set contained all 43 classes. Also, images were adjusted to a size of 160x160 pixels. A data augmentation step was also used, applying random flip and random rotation to images in the training data, generating 9 images for each original image.

Figure 1. Sample of the training data set used for this project milestone

After testing some initial models, as described in the following sections, additional data was incorporated with the goal of reducing variance. 50 images from Bing search were downloaded and added to every coarse-grained category. This was done by using a script that looped through all classes and downloaded and saved the images in different folders. After this, a data screening script was ran to ensure that all images were in adequate formats (.jpg, .png, .jpeg). Finally, the new images were visually checked to ensure that they were relevant to each of the categories. After these new images were added, the training set increased from 2,640 to 3,584 images. The same data pre-processing and augmentation steps used previously were applied to this new data.

4. Methods

Three different model architectures are tested in this classification task. The first network is a MobileNetV2, the second is a ResNet50, and the third is a VGG16. A transfer learning approach has been used on all occasions given the low volume of data available. Weights that have been pre-trained in ImageNet have been taken as a starting point, re-training some of the final layers for each model. Also, a sparse categorical cross entropy loss function has been used.

$$\text{Loss} = -\sum_{i=1}^{\text{output size}} y_i \cdot \log \hat{y}_i$$

The VGG16 network is a deep convolutional neural network with a very simple architecture. It is formed by consecutive steps of convolutional layers followed by max-pooling layers, which progressively reduce the height and width, but increase the depth of the input. The network has a total of 16 weighted layers and finishes with a few fully connected layers and a SoftMax layer.

The ResNet50 model is a very deep network (50 layers) that is able to maintain a good performance largely due to the residual blocks that form it. These residual blocks allow activations from previous layers to skip intermediate layers. This allows the ResNet to have the flexibility to act as a shallower network if this more optimal for the problem at hand.

The MobileNetV2 is especially adequate for mobile applications because of the low computational cost compared to other deep networks [5]. The network has 154 layers, but uses depthwise separable convolution, which is around 10 times cheaper than regular convolution. It also uses a residual connection, like the ResNet.

Different parameters were tested on each model, with the goal of obtaining the best accuracy in the validation set. An error analysis was performed to understand the sources of variance, and finally an additional data augmentation step was used with the aim of reducing variance. This is explained in more detail in the following section

5. Experiments, results, and discussion

5.1. MobileNetV2

5.1.2 First iteration of MobileNetV2

The first model trained and tested was a MobileNetV2. The model was trained from layer 120 onwards, using base learning rate of 0.001, an Adam optimization algorithm and a dropout regularization of 0.2. The model was trained using batches of size 32 and a total of 15 epochs. Higher number of epochs were also tried but performance did not increase significantly. The evolution of the accuracy in the validation set is shown below.

Figure 2. Evolution of accuracy in training and validation sets for MobileNetV2 ‘a’

While the accuracy in the test set was high, showing almost non unavoidable bias, a high variance is seen. Accuracy in the validation set is around 0.77, much lower than the training set. This shows that the model is overfitting the training dataset, which could be solved by increasing regularization or by growing the training dataset. The first approach (increasing regularization) was chosen as a first step given the simplicity.

5.1.2. Second iteration of MobileNetV2

The second iteration used the same model as before, but with a dropout rate of 0.5 instead of 0.2. As the picture below shows, this yielded better results, as the variance was reduced and the accuracy on the validation set reached levels of 0.82. The only downside is that it took a slightly longer time to reach low bias levels compared to the previous model, which makes sense due given the high dropout rate used and the randomness this generates in the training process

Figure 3. Evolution of accuracy in training and validation sets for MobileNetV2 'b'

5.1.3. Third iteration of MobileNetV2

The third iteration also used the same model, but with a dropout rate of 0.7. In 15 epochs the model did not have time to converge to a high accuracy level in the training set, to the model was trained for a total of 20 epochs. The accuracy achieved in the validation set was 0.85, higher than in both previous cases.

5.2. ResNet50

The second model used was a ResNet50, trained for 45 epochs from layer 35 onwards. A dropout rate of 0.5 was used. As seen in the picture below, the performance of this model is significantly worse than performance of the MobileNetV2. The model not only converges more slowly but also has much higher variance, reaching accuracy levels below 0.5 in the validation set. A drop in performance is since in epoch 5 because the only the final layer was trained for the first 5 epochs.

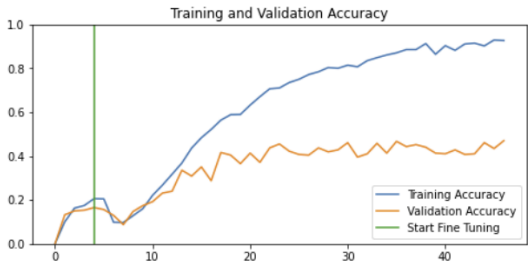

Figure 4. Evolution of accuracy in training and validation sets for ResNet50 'd'

5.3. VGG16

The third model tried was a VGG16 network. The model was trained from layer 12 onwards for 45 epochs, using a dropout of 0.5. The performance of this model was slightly below the MobileNetV2, reaching accuracy of 0.75 on the validation set after epoch 45.

5.4. Performance in the test set

After trying the three different models and a range of parameters, the best parameters that achieved higher validation accuracy (models '5.1.3', '5.2', '5.3') were selected for each model and tried on the test set. MobileNetV2 with 0.7 dropout described above in 'c' achieved the best performance, with 0.85 accuracy.

5.5. Error analysis

Given that there is still a significant variance with the best model found, an error analysis was performed on 50 errors in the validation set to understand the sources of error and identify opportunities to augment the data and improve performance. Some examples of misclassification from the validation set are shown below:

Figure 5. Examples of misclassification in validation set for MobileNetv2

One conclusion from the analysis is that most of the errors occur between classes that are similar to the human eye. For example, a Honeydew Melon is very similar to a Lemon to a human's eye, and the model also often mixes these categories. This also occurs with Apples/Pears, Satsumas/Grapefruits and others. A particularly bad performance is observed in Limes, with an F1 score of 6% in the test set (the lowest among all classes). Other categories with bad performance are Mango (45% F1 score) and Sour-Milk (22% F1 score).

5.6. Increasing data volume

From the error analysis, the conclusion is that increasing the volume of training data is the most adequate way to meaningfully decrease variance. Hence, as described previously, additional images from Bing search was leveraged to increase the volume of training data. Fifty images were downloaded for every category using a script, but around 50% of them were not relevant and had to be eliminated. The result was an increase in 944 pictures (35% increase). However, the improvement in accuracy observed in the test set was minimal, going from 0.85 to 0.86 for the model presented in '5.1.3'.

All graphics should be centered. Please ensure that any point you wish to make is resolvable in a printed copy of the paper. Resize fonts in figures to match the font in the body text, and choose line widths which render effectively in print. Readers (and reviewers), even of an electronic copy, may choose to print your paper in order to read it. You cannot insist that they do otherwise, and therefore must not assume that they can zoom in to see tiny details on a graphic.

6. Conclusions and future work

There is still significant room for improvement in reducing variance in the classification task presented in this paper. The best accuracy achieved in the test set was 0.85 via a MovileNetV2, which means that the model incorrectly classifies a grocery in 15% of occasions. It is evident that more data or new approaches need to be pursued to meaningfully increase performance. More concretely, two lines of work are proposed to keep increasing performance:

6.1. Growing the volume of training data:

Currently, the training data contains an average of only 60 images for each class. We have demonstrated that this volume is not high enough to achieve a best-in-class performance. Hence, a line of work is increasing the volume of training data either by manually taking pictures in the supermarket or by performing a more exhaustive online search.

6.2. Testing contrastive learning models.

One potential alternative could be to use contrastive learning techniques with a network architecture and loss function like the one used for FaceNet [6]. A transfer learning approach would be used, by utilizing a network that has already been trained for a different type of object. This approach would be novel because it would enable the deployment of this solution in a real supermarket environment where SKUs are numerous and are constantly varying. Instead of having to train the model with a large volume of data every time a new product needs to be added, just one picture added to the database can enable contrastive learning.

Please refer to the author guidelines on the CVPR 2022 web page for a discussion of the use of color in your document.

If you use color in your plots, please keep in mind that a significant subset of reviewers and readers may have a color vision deficiency; red-green blindness is the most frequent kind. Hence avoid relying only on color as the discriminative feature in plots (such as red *vs.* green lines), but add a second discriminative feature to ease disambiguation.

7. Team member contribution

This project has been developed individually by Cristobal Maturana

References

[1] Filax, Marco, Tim Gonschorek, and Frank Ortmeier. "Grocery Recognition in the Wild: A New Mining Strategy for Metric Learning." VISIGRAPP (4: VISAPP). 2021.

[2] Jund, P., Abdo, N., Eitel, A., Burgard, W.: The freiburg groceries dataset. arXiv preprint

[3] Ciocca, Gianluigi, Paolo Napoletano, and Simone Giuseppe Locatelli. "Multi-task learning for supervised and unsupervised classification of grocery images." International Conference on Pattern Recognition. Springer, Cham, 2021.

[4] M. Klasson, C. Zhang and H. Kjellström, "A Hierarchical Grocery Store Image Dataset With Visual and Semantic Labels," 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), 2019, pp. 491-500, doi: 10.1109/WACV.2019.00058.

[5] Sandler, Mark, et al. "Mobilenetv2: Inverted residuals and linear bottlenecks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[6] Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.