“Quick, Draw!” Image Classification: A Hybrid Deep Learning Model

Mengci Duan, Wupeng Han, Yipin Lu
ANLY 590, 15 December 2018

Abstract

To predict the doodles from “Quick, Draw” dataset, a hybrid neural network of CNN and RNN with parallel structure was created to predict the doodles’ topics. The performance of this parallel model surpasses that of any single model. The eigenfaces and features extractions were also performed using Principal component analysis and convolutional neural network to explore the differences in perception of images between human and machine.

1. Introduction

Digital image processing is one of the most important fields of computer science, which uses computer algorithms to perform image processing on digital images. With the technology of digital image processing, machines are able to perform classification, feature extractions, pattern recognition and so on. This project is about an exploration in image classification as well as feature extractions.

“Quick, Draw” is an online game developed by Google. Players need to guess the topics of doodles drawn by other players. Then an AI was developed to predict the painting topics. The drawings were captured as vectors as well as the topics the player was asked to draw and in which country the player was located. A common network which people used a lot in image analyzation and classification is convolutional neural networks (CNN). However, single CNN does not yield expected outcomes. Thus, in the project, we decided to combine CNN and RNN (recurrent neural network) to perform image recognition.

2. Related Work

In the field of image classification, various CNN-based classification models have been created with different characteristics, for instance: VGGNet, MobileNet and ResNet and so on. People have used these networks to work with image datasets, for instance, the digital Mnist Dataset. Different from common pictures, which have complex patterns, doodles are simple and sometimes abstract. In this project, VGG and MobileNet will be employed for image classification.

Amongst the key aspects in machine learning are “Feature Extraction”. An APP called “Zepeto” goes viral recently. The user will upload a front-face picture into the app, and then a 3d dimension model will be generated and looks like the user. What’s more, these models are able to make poses like humans. The feature extraction, specifically in the field of facial recognition, has been applied in this application. This application shows a practical way of using feature extraction. Rather than carrying on such a complex model. The project puts more emphasis on the way machine process the world.

3. Data Description

The dataset is obtained from Kaggle. It is a collection of doodles painted by players across more than 300 categories. Kaggle provides two versions of this dataset: raw and simplified versions. The raw data is the exact input recorded from the user drawing, while the simplified version removes unnecessary points from the vector information. Simplified version dataset is employed in this project due to smaller sizes and the consideration of time efficiency. Also the project only utilizes the category of cats for training and feature extractions to save time. In the project, we began with using local installed python for processing. However, then we use Google Colab instead for efficiency.

4. Methodology

4.1 Image Classification

As Fig. 1 shows, a hybrid structure is applied to images in this project, which is a parallel neural network consisting of CNN and RNN. The pictures are processed through the CNN branch, yet the stroke sequences are processed by RNN branch. The project adopts MobileNets to implement CNN and GRU (Gated Recurrent Unit) to implement RNN. We employ the GRU rather than LSTM in this project since GRU has a less complex structure than LSTM, which results in higher computationally efficiency.

And in the end, these two branches would get concatenated together via these quantization encoding layers. Compared with CNN-based models, this hybrid model takes the inherent stroke-level temporal information of human sketches into account. Hence, we can expect that the hybrid model is able to beat CNN-based model in our project. For the purpose of comparison, we also created a single VGG16 model and a MobileNets Model to make classifications.

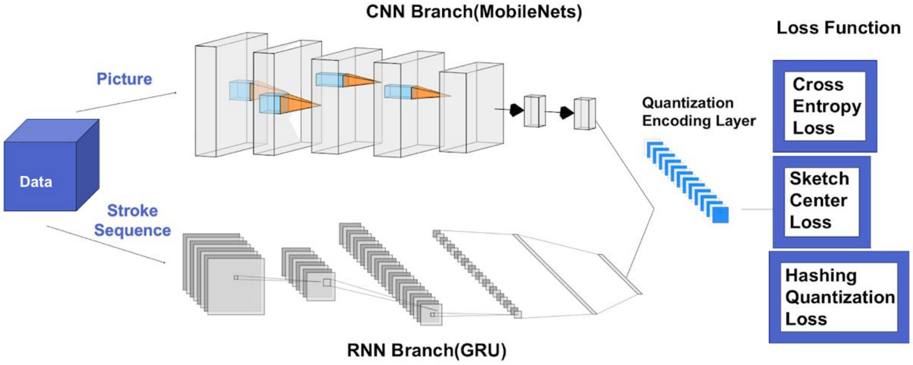


Figure 1. The Hybrid Model Used to Classify Quickdraws

The function of Quantization Encoding Layer is to transform the deep features (f_n) of CNN and RNN branches to the hashing code (b_n) by using one fully connected layer with sigmoid activation.

$$b_n = sgn(f_n - 0.5)$$

After that, we applied a hybrid loss function that consists of three parts: Cross Entropy Loss, Sketch Center Loss and Hashing Quantization Loss.

(1) The Cross Entropy Loss:

$$L_{cel} = \frac{1}{N}\sum_{n=1}^{N} -\log \frac{e^{\mathbf{W}_{\mathbf{yn}}^{\mathbf{T}}\mathbf{f}_{\mathbf{n}}+\hat{\mathbf{b}}_{\mathbf{yn}}}}{\sum_{j=1}^{L} e^{\mathbf{W}_{\mathbf{yj}}^{\mathbf{T}}\mathbf{f}_{\mathbf{n}}+\hat{\mathbf{b}}_{\mathbf{yj}}}}$$

This loss is widely used in multiclass classification. The W_j is the jth column of the Weights Matrix between the Quantization Encoding Layer and the softmax outputs.

(2) The Sketch Center Loss:

$$L_{scl} = \frac{1}{N}\sum_{n=1}^{N} ||f_n - c_{yn}||_2^2$$

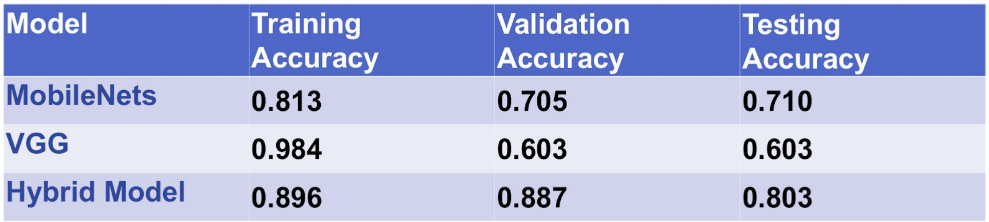
The C_{yn} is the mean of f_n after removing noise from sketches. The purpose of this loss is to find a fixed but representative center feature for each class, which help to eliminate the problem of similar appearances between strokes from different categories.

(3) The Hashing Quantization Loss helps to eliminate the error generated by the quantization-encoding. The function is:

$$L_{ql} = \frac{1}{N}\sum_{n=1}^{N} ||b_n - f_n||_2^2$$

5. Results

5.1 Image Classification



Model	Training Accuracy	Validation Accuracy	Testing Accuracy
MobileNets	0.813	0.705	0.710
VGG	0.984	0.603	0.603
Hybrid Model	0.896	0.887	0.803

Tables 1. Result of Different Models in Quickdraw Classification

Table 1 shows that the testing accuracy(0.803) of this hybrid structure outperforms a single MobileNets(0.71) or VGG(0.603). Notice one interesting finding is that the training accuracy of VGG is the highest among all three models possibly due to overfitting. And the MobileNets' testing accuracy is higher than the VGG, with a lower training accuracy. The reason for this higher testing accuracy might be that MobileNets model reduces the number of parameters with the same depth. Hence, it will be more robust to the problem of overfitting. Moreover, the hybrid model surpasses the MobileNets model. As we discussed earlier, the recurrent branch takes the steps of drawing strokes into consideration.

5.2 Feature Extraction

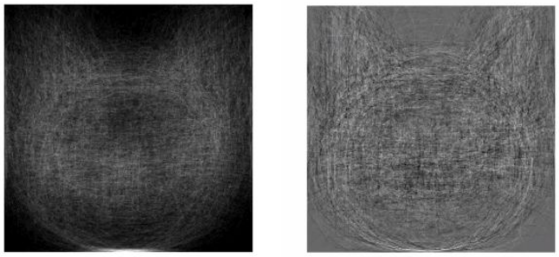


Figure 2. Mean Face (Left) and Eigenface (Right) of Cats

To begin with, the mean face of 1000 cat's pictures is generated using Principal Component Analysis (PCA) to create eigenfaces, from which a blurry shape of a cat's face can be observed when 20 eigenvectors were projected.

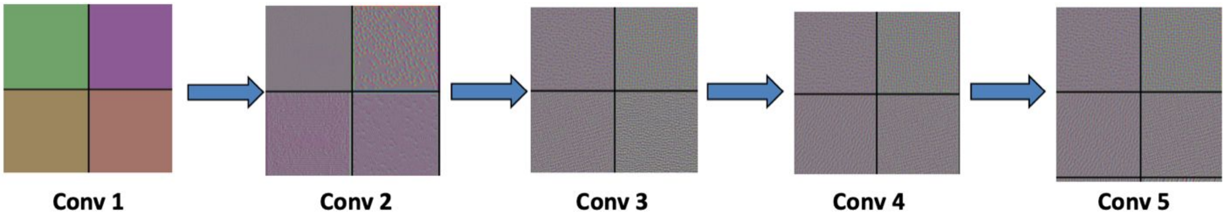


Figure 3. Outputs of CNN Filters in the First Five Layers

However, when the feature extraction with CNN is performed with the same data, the generated attributions are unperceivable for humans. The following pictures show the visualization of feature extraction with CNN using 5 CNN layers. Very few variances among visualizations which generated from Layer 2 to Layer 5 can be identified by human eyes. What's more, visualizations of features extracted by different layers of CNN are meaningless to human. People can only recognize these visualizations as colored squares with awkward patterns. However, AI can identify these "magpies," and classify these pictures correctly.

6. Conclusion

Different from common sequential network structure, the project presents a parallel structure of CNN and RNN with greater accuracy than a single transfer learning model. Also this project discusses that the non-linear transformation employed by CNN can detect complex features of these doodles. However, some improvements can be made in the project: In the project, only simplified dataset is employed. With the higher performance GPU and more time, more categories of the dataset and the original raw dataset should be used for achieving higher accuracy.

Reference

Author, Jana. R., &Author, Lovejoy. J., “Exploring and Visualizing an Open Global Dataset.” *Google AI Blog*, 25 Aug. 2017, ai.googleblog.com/2017/08/exploring-and-visualizing-open-global.html. Accessed 2 Dec 2018.

Beluga. “Greyscale MobileNet” *Kaggle*, 27 Nov. 2018, www.kaggle.com/gaborfodor/greyscale-mobilenet-lb-0-892. Accessed 2 Dec 2018.

Chollet, F. “How Convolutional Neural Networks See the World.” *The Keras Blog*, 30 Jan. 2016, blog.keras.io/how-convolutional-neural-networks-see-the-world.html. Accessed 2 Dec 2018.

Nguyen, H. “Combining CNN and RNN.” *Kaggle*, 28 Nov. 2018, www.kaggle.com/huyenvyvy/fork-of-combining-cnn-and-rnn. Accessed 2 Dec 2018.