Image Classification Optimization

Mengzhi Zhou
dept. Data Science and Analytics
Georgetown University
Washington, D.C. USA

Guanzhi Wang
dept. Data Science and Analytics
Georgetown University
Washington, D.C. USA

***Abstract*—How to improve image classification model accuracy while keeping fast-paced data processing speed? What need to do to solve overfitting problem caused by deep learning model without information sacrifice? In this project, we will compare 5 image classification models with different image processing technique and convolutional neural network structure to figure out tools for optimizing performance of image classification model.**

***Index Terms*—Image Classification, Convolutional Neural Network, Data Augmentation, Transfer Learning**

I. Introduction

Image classification is one of the most significant but highly complicated machine learning tasks in the world. Abundant companies are seeking their ways to employ image classification technology to their own product for better market competency, such as: iPhone face unlock technology and Instagram filters. On the other side, complexity of image classification technology becomes a large obstacle for lots of companies. The amount of data needs to be processed and transformed, the trade-off between accuracy and information loss, and how to eliminate noise from data, those problems need to be considered before employing image classification technology. Problems like how to get an optimized image classification model for products become a trouble for companies.

However, with the rapid development of deep learning in recent years, image classification performance has largely increase and various types of tools has been developed. Obstacles has been solved one by one. By using CNN, the performance records of image classification have been broken in the ImageNet Challenge 2012 [1]. Data augmentation method has been used as a tool to improve model performance and reduce overfitting problem. Moreover, traditional machine learning is also an effective and powerful way to classify images. For example, Bayesian methods, which is useful in solving various machine learning problems, have been applied in many image analysis and computer vision problems [2]. In this project, we try to optimize the image classification process by applying appropriate tools to make the process fast enough and as accurate as possible. Due to the limited time and hardware equipment, it is impractical to apply some huge models with Bayesian framework. According to Vailaya, Figueiredo, Jain and Zhang, they spent 12 days on a Sun Ultra 2 Model 2300 processor with 512 MB memory to select features and run the 700 samples training set for their models [2].

Thus, the remaining of paper is organized as follow: in section 2 we go through the related work of image classification. In section 3, we detailed look at the data of this paper. Then, we introduce the method and models used in the paper. Based on the method, we deliver the result, discussion of results and conclusion as section 5, 6 and 7.

II. Related Work

"The Complete Beginner's Guide to Deep Learning: Convolutional Neural Networks and Image Classification [3]", written by Anne Bonner, gives a basic overview for image classification and CNN. This paper talks about the foundation of CNN and image classification, such as: logics behind models, basic structure of CNN, etc. The idea of convolutional layers shown in paper give a potential approach to improve performance of image classification models. Another paper, named "The 4 Convolutional Neural Network Models That Can Classify Your Fashion Images [4]", directly presents a performance comparison between image classification models with different convolutional layer numbers on datasets: Fashion-MNIST. This paper shows that model's validation accuracy will decrease as convolutional layers increase, but after applying data augmentation method, overall accuracy of model will increase as convolutional layer increases. However, this paper only compared 3 models with different convolutional layer number and 1 model with transfer learning. To get better results, both of horizontal and vertical comparison will be included in our project.

Furthermore, there are several research studies that focus on data augmentation which is widely used in image classification tasks for performance improvement. "The Effectiveness of Data Augmentation in Image Classification Using Deep Learning [5]", written by Wang Jason and Luis Perz, illustrates an experiment to compare impact of different data augmentation techniques: traditional transformation, generative adversarial networks, and neural net augmentation. Not only different data augmentation tricks are used to compare, but also dataset. The first dataset is cat vs. dog, second one is dog vs. goldfish and last one is MNIST digit 0 vs. digit 8. By comparing performance of different data augmentation on different dataset, authors gave a comprehensive conclusion about data augmentation: data augmentation technique could highly reduce overfitting problem and increase accuracy of image classification model. Another paper related to data augmentation is "Occlusions for Effective Data Augmentation in Image Classification [6]" which is written by Fong Ruth and Andrea Vedaldi. This paper mentioned another common problem: "photographer bias", which can strongly affect neural network performance. "Photographer bias" means people will pay attention to the main objects in the photos instead of subtle areas.That will ignore important feature in subtle area of image. To solve this problem, authors illustrate occlusion augmentation techniques: batch augmentation, joint training and Dropout method, and set up an experiment to compare those occlusion augmentation effects.

Under the Bayesian framework, Vailaya, Figueiredo, Jain and Zhang modified the binary Bayesian classifiers that achieved classification accuracy that higher than 90% for all four classification problems with database of 6931 photos [2]. In "Local Naive Bayes Nearest Neighbor for Image Classification" written by McCann and Lowe, they create the Local Naive Bayes Nearest Neighbor which "gives a 100 times speed-up over the original NBNN on the Cal- tech 256 datasets [7]" . Also, Zhou, Cui, Li, Liang and Huang in their paper "Hierarchical gaussianization for image classification [8]" discussed about the hierarchical Gaussian mixture model whose parameters are learned in a Bayesian framework that is useful in image classification .

III. Data Collection

Dataset used in this project was collected from Kaggle [9], with 25,000 RGB images for dogs and cats. Each class has 12,500 images with different height and width length. Both of cats and dogs images are put in same folder but labeled with different file name: class name + image index (e.g, cat.1200.jpg). Since data processing method used in this project require different classes images to be sorted in separate directories, dogs and cats images need to be splitted and put into different directories. Based on special feature that different classes of image has different file name format, function find could distinguish those classes of images easily and move to different directories. At the same time, we walked through image samples and identify several irrelevant images which should be deleted for training data quality. Several samples of meaningless images in our training data are presented at the end of this section.

To facilitate processing time of model training, GPU based tensorflow is applied to model training instead of CPU based approach. Number of images that used for model training also decrease from 25,000 to 20,000 by randomly subset method. Train validation ratio is set as 7:3 for data balance, which means each class has 7,000 images for training and 3,000 images for validation to prevent overfitting and underfitting problem. The last step before image classification model training is transfering image to other format that neural network model could understand and read. In our project, function ImageDataGenerator is used to transfer those images into batches of data arrays for model training. This function could easily handle various images and make preparation for further model training.

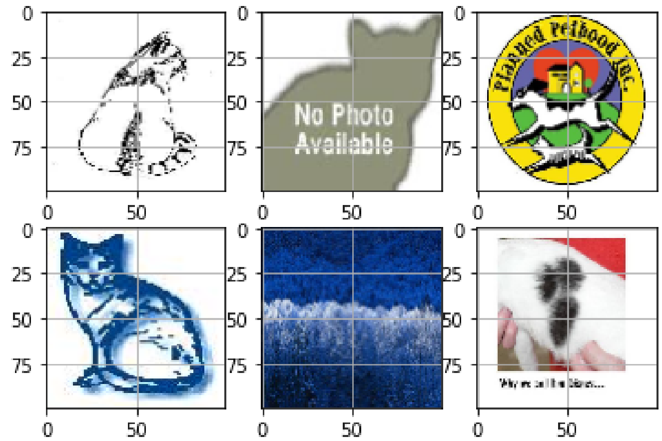


Fig. 1. Irrelevant Image Examples

IV. Methods

This project is aimed to improve image classification performance mainly from two parts: data augmentation and convolution neural network. Data augmentation includes various image transformation approaches, include: randomly cropping, horizontal flip, zoom, rescale, brightness adjust, etc. This method is widely used to improve generalisation properties and reduce overfitting problem. To some extent, data augmentation could eliminate several problems caused from small training data size since this technique could generalize "brand new" images from original dataset. It also could help deep learning model learn objects' important features better by image transformation. However, this technique will largely slow down model training process since abundant work from image transformation.

Fig. 2. Data Augmentation Example

Structure of convolutional neural network primitively decide the model's overall performance. Neural network components, like: number of convolutional layer, number of fully connected layer and drop out rate, activation function, has close relationship to precision and information loss. As the complexity of model increase, model accuracy normally will increase, at the same time, processing time also will largely increase. Sometimes, overfitting problem will appear when apply model to validation data.

Therefore, the optimization between model complexity and processing time is significant to image classification task. To figure out each component's effect, 5 image classification models with different data preprocessing methods and convolutional neural network structure will be trained and compared performance with each other in this project.

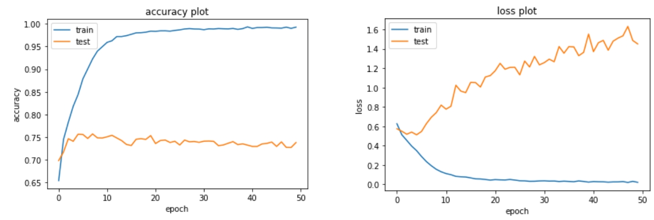
- The first one is baseline model without any data augmentation approach. This model's structure is relatively simple than other models. It only contain 1 convolution layer with 32 filters numbers, relu activation function, 0.3 dropout rate with image input shape 64*64 and 1 fully connected layer. Optimizer of this model is adam and loss function is binary cross entropy.
- Second model have similar network structure with baseline model. In contrast, this model have data augmentation parts with rotation, width-shift, height-shift, shearing, horizontal flip, and zoom for image transformation.
- The third model with 3 convolution layers has higher complexity structure than baseline model. The first convolution layer is same as baseline model. The second convolution layer has higher filters number: 64, same relu activation function as previous one, 0.3 dropout rate. The last convolution layer of this model has 128 filters to process data which is much larger than baseline model's complexity. However, the second model does not have data augmentation part.
- Next model is the combination of data augmentation model and 3 layer convolutional neural network model. It contains both of data augmentation parts that third model has and complicated neural network structure that second model has. This model has highest complexity level among previous 3 convolutional neural network models.
- The last model is pretrained model which represent transfer learning component. Transfer learning means machine learning model use knowledge learned from solving one kinds of problem and apply it to different problems [10]. VGG16 model which was trained on ImageNet dataset will be included in this project.

V. Results

To comprehensively compare model performance, 5 evaluation metrics will be assessed through each image classification model, included: AUC score, confusion matrix, classification report, running time, accuracy and loss on both of training and validation dataset. The combination of AUC score, confusion matrix and classification report could give precise analyzation on classification accuracy. Running time of model is a direct indicator of model processing speed. At the end, plots of accuracy and loss will be used as a tool to identify underfitting and overfitting problem in models.

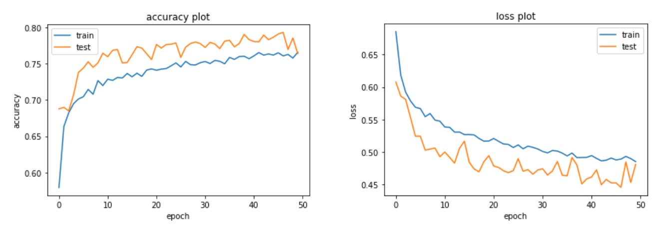
Since baseline model has the simplest neural network structure among those 5 models, accuracy of this model also is the lowest one, only has 0.7382 on test dataset. In contrast, low level structure complexity makes model process faster than other models, especially models involved with data augmentation parts. By investigating accuracy and loss plots of baseline model, overfitting problem can easily be identified. accuracy plot on training dataset is located above accuracy plot on test dataset.

TABLE I
Accuracy and Loss Plot for Baseline Model



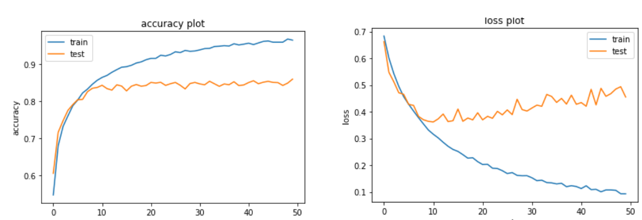
After data augmentation is introduced to second model, overfitting problem is solved. Accuracy plot on training dataset is located below accuracy plot on test dataset, and AUC score has improved to 0.7637. However, model processing time is rapidly increase from 13.52 mins to 22.59 mins. Running time is almost doubled. Instead of overfitting problem, underfitting problem appears since there is distinguishable separation between accuracy plot on training and test.

TABLE II
Accuracy and Loss Plot for Data Augmentation Model



Model with 3 convolution layers has improve AUC score from 0.7382 of baseline model to 0.8590, which is a impressive improvement for model training. Model processing time is not as long as previous model that involved with data augmentation, only has 14.03 mins. In contrast, it brings overfitting problem to model again. Training accuracy is larger than accuracy of test dataset, which will largely reduce model performance.

TABLE III
Accuracy and Loss Plot for 3 CNN Model



Except pretrained model, the combination of data augmentation and 3 convolution layers model has the best performance amongst those models. By sacrificing model processing speed, this model doesn't have overfitting problem as previous one, at the same time, accuracy of this model has reached to 0.8522, which is much higher than baseline model's accuracy: 0.7382.

	precision	recall	f1-score	support
cats	0.81	0.92	0.86	3000
dogs	0.91	0.79	0.84	3000
accuracy			0.85	6000
macro avg	0.86	0.85	0.85	6000
weighted avg	0.86	0.85	0.85	6000

Fig. 3. Classification Report for 3CNN+DA model

Compared to customized neural network models that has shown above, pretrained model VGG16 has the highest performance on accuracy since VGG16 model was trained from ImageNet dataset which containing over 1,000 categories. Also, 90 of the 1,000 classification categories are dog species which could largely improve model performance [11].

TABLE IV
ACCURACY AND LOSS PLOT FOR VGG16 MODEL

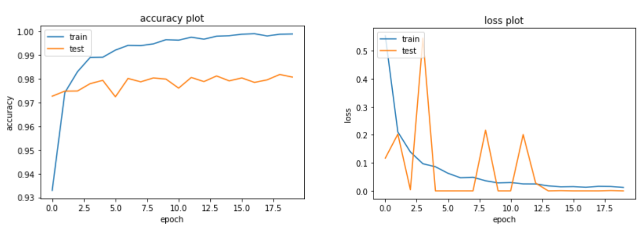


TABLE V
PERFORMANCE EVALUATION

	Convolutional Neural Network				
	Baseline	***DA***	***3CNN***	***3CNN+DA***	***VGG16***
Accuracy(val)	0.7382	0.7637	0.8590	0.8522	0.9818
Loss(val)	1.4504	0.4806	0.4552	0.3373	0.2631
AUC score	0.7382	0.7637	0.8590	0.8522	-
Time(mins)	13.52	22.59	14.03	22.54	23.47

*DA:Data Augmentation

VI. DISCUSSION OF RESULTS

From previous section, all five image classification models have achieved acceptable accuracy scores that higher than 0.73. By comparing baseline model and second one, data augmentation improves model accuracy score slightly. In expectation, data augmentation should increase validation accuracy much larger. It is also worth noting that, by looking at the run times, applying data augmentation is really time consuming and seems not worth to do that. CNN Model with 3 convolution layers has highest accuracy score among those customized model, at the same time, it also has relatively fast processing speed. The only problem within this model is overfitting which is not acceptable in model training. Unlike third model, the combination of data augmentation and 3 convolution layers image classification model has relatively high accuracy score but slow processing speed. Due to data augmentation method, this model does not have overfitting problem that happened in image classification model with 3 convolution layer. In this case, this model has better overall performance than others with high accuracy and no overfitting problem. The last one is pretrained model with peak accuracy performance 0.9818: VGG16 which is utilized large training data and fine tuning. This model also shows us the potential development space for previous 4 customized CNN model. After vertical and horizontal comparison among each individual model, it is obvious that the CNN framework with more hidden layers could facilitate model performance better than data augmentation, and the pre-requirement is that we only consider each component effect individually. On the other hand, it also demonstrate a significant point: the inseparable relationship between CNN model and data augmentation. To facilitate model performance, neural network structure should be complicated and number of hidden layers should increase. When complexity of network structure increase, overfitting problem will appear, then data augmentation should be introduced to model to reduce overfitting. In general, complicated network structure could eliminate underfitting problem and data augmentation could reduce overfitting problem. The optimization point between those two components will decide ovrall performance of deep learning model. Compared to previous related work, this project contains more evaluation metrics as performance indicators for both of horizontal and vertical comparison. Not only this point, pretrained model VGG16, which represent new technology : transfer learning, also is included to model comparison. It could provide a indicator as model peak performance.

VII. CONCLUSION

To conclude, we have discussed 5 different deep learning models to introduce the two major approaches used in the image classification project: data augmentation and structure of convolutional neural network. For our dataset, data augmentation is not always effective on accuracy improvement but could eliminate overfitting problem from model. Neural network structure plays an siginificant role in model optimization that could largely increase model performance without slow down processing speed. In the future, we may test more CNN models to determine the Optimal layer number considering both accuracy and run time. We would also apply the methods under Bayesian framework if we have enough capability.

REFERENCES

[1] Buyya, R., Calheiros, R. N., Dastjerdi, A. V. (Eds.). (2016). Big data: principles and paradigms. Morgan Kaufmann.
[2] Vailaya, A., Figueiredo, M. A., Jain, A. K., Zhang, H. J. (2001). Image classification for content-based indexing. IEEE transactions on image processing, 10(1), 117-130.
[3] Bonner, Anne. "The Complete Beginner's Guide to Deep Learning: Convolutional Neural Networks." Medium, Towards Data Science, 1 June 2019, towardsdatascience.com/wtf-is-image-classification-8e78a8235acb.

[4] Le, James. “The 4 Convolutional Neural Network Models That Can Classify Your Fashion Images.” Medium, Towards Data Science, 7 Oct. 2018, towardsdatascience.com/the-4-convolutional-neural-network-models-that-can-classify-your-fashion-images-9fe7f3e5399d.

[5] Wang, Jason, and Luis Perez. “The Effectiveness of Data Augmentation in Image Classification Using Deep Learning.” The Effectiveness of Data Augmentation in Image Classification Using Deep Learning, 13 Dec. 2017.

[6] Fong, Ruth, and Andrea Vedaldi. “Occlusions for Effective Data Augmentation in Image Classification.” 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), 23 Oct. 2019, doi:10.1109/iccvw.2019.00511.

[7] McCann, S., Lowe, D. G. (2012, June). Local naive bayes nearest neighbor for image classification. In 2012 IEEE Conference on Computer Vision and Pattern Recognition (pp. 3650-3656). IEEE.

[8] Zhou, X., Cui, N., Li, Z., Liang, F., Huang, T. S. (2009, September). Hierarchical gaussianization for image classification. In 2009 IEEE 12th International Conference on Computer Vision (pp. 1971-1977). IEEE.

[9] ”Dogs vs. Cats Kaggle”, Kaggle.com, 2020. [Online]. Available: https://www.kaggle.com/c/dogs-vs-cats/overview.

[10] Sarkar. (2018, November 17). A Comprehensive Hands-on Guide to Transfer Learning with Real-World Applications in Deep Learning. Retrieved from https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a

[11] Cats vs Dogs - Part 2 - 98.6 Accuracy - Binary Image Classification with Keras and Transfer Learning. (2019, May 12). Retrieved from https://wtfleming.github.io/2019/05/12/keras-cats-vs-dogs-part-2/