

ECE 228 PROJECT DOG BREED CLASSIFICATION

Zhenyan Wang, Dingqian Zhao, and Kexin Hong

University of California San Diego, La Jolla, CA 92093-0238,

ABSTRACT

With the development of machine learning, Convolutional Neural Networks (CNN) are widely used in various disciplines. It provides many tools that can perform different tasks on large data sets. The focus of this paper is on tools that classify the breed of a dog based on an input image. The data set is Stanford Dog data set. We pick three representative models, VGG16, Inception V3 and Xception to extract features in an image. Then we use logistic regression classifier to identify the breed of dogs in it. Xception and Inception both perform well. Their accuracy reach 99% on training set and 94% on test set. However, VGG16 performs poorly whose accuracy is just 85% on test sets. In the end, the reasoning of such results is discussed.

Index Terms—CNN, VGG16, Inception V3, Xception, Dog classification, Logistic regression,

1. INTRODUCTION

Dogs are adorable creatures. Despite the enormous variety in the way a dog acts and reacts to the world around them, at least we can tell what kind of dog it is with the help of Convolutional Neural Networks (CNN). The input to our algorithm is an image. We use transfer learning to extract different level features with pretrained models, such as VGG16, Inception, and Xception. The output is the breed of the dog in the image with the help of Logistic Regression Classifier.

Besides, this topic provides a great domain for classification problems. After all, dogs are possibly the most photographed creatures apart from humans. With the increase number of cameras, sources of dog images are abundant. These images show dogs of various shapes and sizes under different poses, which could provide sufficient data for categorization experiments. The results in this domain then might extend to broader domain of species categorization.

(Convolution Neural Networks has recently become prominent for image classification problems. Compared to other classification techniques, CNN is unique in that it starts from raw data, obviating the need to hand engineer features prior to applying the modeling technique. For image sets that are resistant to summarization via traditional features, Neural Networks are particularly suitable to deal with such problem. In our case, the images are graphs of dogs.)

2. RELATED WORK

Conventionally, computer vision problems have been solved using hand-engineered features like HOG, SURF and various type of features. However, features extraction task has been mostly shifted toward Convolutional Neural Networks since the success of AlexNet[1] at 2012 and VGG[2] at 2014 . After that, deeper models such as GoogleNet, ResNet[3] and DenseNet [4] are introduced.

Our project especially addresses fine-grained categorization, that is, we need to make discrimination between a large number of similar classes. Some of the previous work relies on segmentation to localize the object of interest before extraction. For example, determine plant species based on images of leaves. Their dataset is leaf photos with flattened leaves photographed with top and bottom lighting in plain white background. They achieved good performance using a color-based EM algorithm for segmentation.[5]

In another study, Dog Breed Classification Using Part Localization, a successful localization and classification pipeline designed for the problem are presented. Their breed identification algorithm relies on accurate facial detection as a first step, and yet produces impressive results via a pipeline backed by SVM. [6] Another previous work uses the multiple kernel framework with an SVM classifier. They compute four features for the flowers, each describing different aspects. The most discriminative kernels for each class are then combined. [7] Besides that, a variety of techniques and models have been used, including Inception, Xception, ResNet, etc. [8]

3. DATASET AND FEATURES

To train the model, we use the Stanford Dogs dataset [9]. This dataset has 20,580 images for 120 different breeds of dogs, and each dog breed has minimum 60 images. For the sake of quicker experiments, we picked 2247 pictures of 20 different types of dogs to model. We have divided dataset into train data and test where train data is 80% and test data is 20% respectively. Since the size of images varies, we set a 299*299*3 window to cut out the image from the center. Examples of image samples from our data set are shown in Figure 1.

For feature extraction, we applied pre-trained models as feature extractors to do the transfer learning, as learned filters on



Fig. 1. Some dog images from data set

large datasets are transferable into other image classification domains. The three pre-trained model we use for feature extraction are VGG16, Inception, Xception. Details about feature extraction are in Methods part.

4. METHODS

For feature extraction, we use VGG16, Inception, Xception models and compare their performance. After feature extraction, we implement a logistic regression classifier to do the classification.

4.1. VGG16

VGG is a convolutional neural network model for image recognition proposed by the Visual Geometry Group in the University of Oxford, where VGG16 refers to a VGG model with 16 weight layers [10]. In our case, the input layer takes an image in the size of 299 x 299 x 3. Figure 2 is the structure of VGG16: This image is passed through a stack of convolutional layers. In these layers, the filters were used with a very small receptive field: 3x3. The convolution stride is fixed to 1 pixel. Besides, the padding is one-pixel for 33 conv layers. Such setting of the spatial padding of conv layer input ensures that the spatial resolution is preserved after convolution. Spatial pooling is carried out by five max-pooling layers. These max-pooling layers follow some of the conv layers. Note that not all of the conv layers are followed by max-pooling. Max-pooling is performed over a 2x2 pixel window, with stride 2. The output layer is a softmax prediction on 20 classes. Softmax can be used to represent the probability distribution among these 20 classes. It is calculated in the following, where x is the input data into Softmax.

$$\text{softmax}(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

$$\text{softmax}(x) = \text{softmax}(x - \max x_i)$$

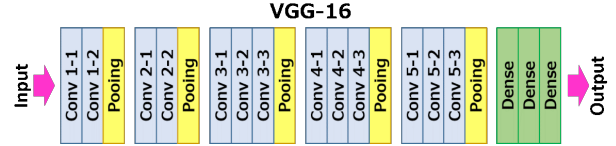


Fig. 2. Structure of VGG16

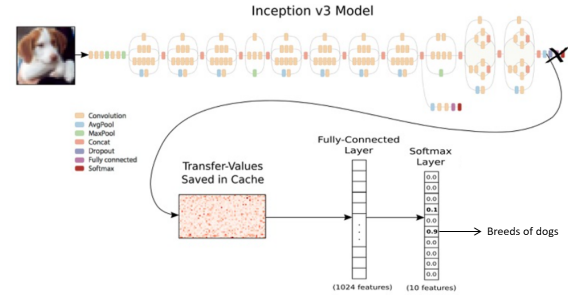


Fig. 3. Structure of Inception

4.2. Inception V3

Inception is a widely used image recognition model. It attains the new state of the art capacity for classification and detection on in the ImageNet Large-Scale Visual Recognition Challenge 2014.[11] It is a popular model which is made up of symmetric and asymmetric building blocks. It includes convolutions, average pooling, max pooling, concatenations, dropouts, etc. Batch norm is used extensively throughout the model, and applied to activation inputs. Loss is computed via Softmax function. In brief, it has a remarkable quality gain at a modest rise of computational requirements compared to shallower and narrower architectures. A high-level diagram of the model is shown in figure 3.

4.3. Xception

Xception uses depth-wise separable convolutions. The model performs one by one convolution first, and then moves into the channel wise spatial convolution. In addition, it does not have an intermediate activation, which also result in its higher accuracy compared to other methods, such as Inception.[12] Lastly, Xception performs better due to the better use of the model parameters.

The architecture of Xception is in figure 4 . It is developed from Inception v3. Just like Inception, Xception also has Entry, Middle and Exit flows. As shown in the picture, the data first goes through the entry flow, then through the middle flow, repeated for eight times, and finally goes through the exit flow. Of course, the core is the Middle flow. Entry flow is mainly used to downsample to reduce the spatial dimension. Middle flow learns and optimizes the features. The Exit flow summarizes and organizes the features, then it is handled to fully connected layer for expression.

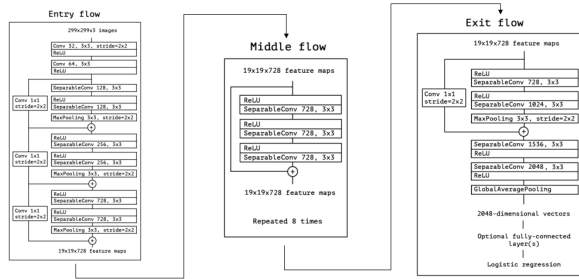


Fig. 4. Structure of Xception

4.4. Logistic Regression

Logistic Regression is a Statistical Learning technique categorized in Supervised Machine Learning methods. It has gained a tremendous reputation for last two decades due to its prominent ability of detecting defaulters. Compared to linear regression methods, logistic regression is useful when the data needs to be assigned to a discrete set of class. In our problem, we use Logistic Regression classifier after feature extractor to make classification. The LRC takes real-valued inputs and makes a prediction as to the probability of the input belonging to each of the dog breeds, which is comparing to the largest probability of each class and assign the image to it.

5. RESULTS AND DISCUSSION

We use mini-batch SGD and the size we pick is 32. If the size is too large, one drawback is learning is too slow, the other one is it may fallen into a local maximum. If the size is too small, it would not converge. We adjust the batch size from full-batch and find when the batch size is 32, the performance is the best. Our primary metrics is accuracy, precision and recall. Below are the corresponding result of three different models.

The performance of VGG16 + LR is not as good, when the dataset got larger, the performance of the model got worse dramatically, when we used 2000 samples, the accuracy is around 95%, but if the whole dataset which has 20000 samples was used, the accuracy was only 84%. Below are the accuracy and some other scores with the confusion matrix. The performance of Inception/Xception + LR is much better and more stable, when the dataset got larger, the accuracy decreased not much, when we used 2000 samples, the accuracy is around 98%, when the whole dataset which has 20000 samples was used, the accuracy was still 94% which kept on a high level. Below are the accuracy and some other scores with the confusion matrix.

We can tell that inception/Xception model do a better job than VGG on dog identification.

VGG Train Accuracy: 0.9980464510527458
 VGG Test Accuracy: 0.8472596585804133
 VGG Recall: 0.8472596585804133
 VGG Precision : 0.859515706482539
 VGG F1: 0.8487010755492813

VGG Train Confusion Matrix
 [[103 0 0 ... 0 0 0]
 [0 94 0 ... 0 0 0]
 [0 0 87 ... 0 0 0]
 ...
 [0 0 0 0 ... 70 0 0]
 [0 0 0 0 ... 0 65 0]
 [0 0 0 0 ... 0 0 58]]

VGG Test Confusion Matrix
 [[19 0 0 ... 1 0 0]
 [0 21 0 ... 0 0 0]
 [0 0 27 ... 0 0 0]
 ...
 [0 0 0 0 ... 9 0 0]
 [0 0 0 0 ... 0 15 0]
 [0 0 0 0 ... 0 0 16]]

Fig. 5. Accuracy and confusion matrix of VGG16

Inception Train Accuracy: 0.9993488170175819
 Inception Test Accuracy: 0.9398023360287511
 Inception Recall: 0.9398023360287511
 Inception Precision : 0.9426408576937481
 Inception F1: 0.9394870651910917

Inception Train Confusion Matrix
 [[103 0 0 ... 0 0 0]
 [0 94 0 ... 0 0 0]
 [0 0 87 ... 0 0 0]
 ...
 [0 0 0 0 ... 70 0 0]
 [0 0 0 0 ... 0 65 0]
 [0 0 0 0 ... 0 0 59]]

Inception Test Confusion Matrix
 [[22 0 0 ... 0 0 0]
 [0 23 0 ... 0 0 0]
 [0 0 29 ... 0 0 0]
 ...
 [0 0 0 0 ... 12 0 0]
 [0 0 0 0 ... 0 16 0]
 [0 0 0 0 ... 0 0 17]]

Fig. 6. Accuracy and confusion matrix of Inception V3

Xception Train Accuracy: 0.9986976340351639
 Xception Test Accuracy: 0.9389038634321654
 Xception Recall: 0.9389038634321654
 Xception Precision : 0.9416252825698992
 Xception F1: 0.9388981058034154

Xception Train Confusion Matrix
 [[103 0 0 ... 0 0 0]
 [0 94 0 ... 0 0 0]
 [0 0 87 ... 0 0 0]
 ...
 [0 0 0 0 ... 69 0 0]
 [0 0 0 0 ... 0 65 0]
 [0 0 0 0 ... 0 0 59]]

Xception Test Confusion Matrix
 [[22 0 0 ... 0 0 0]
 [0 23 0 ... 0 0 0]
 [0 0 29 ... 0 0 0]
 ...
 [0 0 0 0 ... 11 0 0]
 [0 0 0 0 ... 0 16 0]
 [0 0 0 0 ... 0 0 20]]

Fig. 7. Accuracy and confusion matrix of Xception

6. CONCLUSION

We've tried three different models, VGG, Inception, Xception as feature extractors, cooperating with Logistic Regression Classifier to identify dog breeds.

Inception/Xception's performance is much better than VGG according to different kinds of metrics including accuracy, precision, recall and F1 score. Compared to VGG, inception/Xception has deeper layers, diverse kernel size, introduce inception module to keep sparse connection. VGG definitely has it's advantage, for example, it replace one large kernel by two or three 3 * 3 size kernel and make the layers deeper. But in this task, it's clear inception/Xception win.

7. REFERENCES

- [1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, *Adv. Neural Inf. Process. Syst.*, pp. 19, 2012.
- [2] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, Return of the Devil in the Details: Delving Deep into Convolutional Nets, *arXiv Prepr. arXiv* , pp. 111, 2014.
- [3] K. He, X. Zhang, S. Ren, and J. Sun, Deep Residual Learning for Image Recognition, *Arxiv.Org*, vol. 7, no. 3, pp. 171180, 2015.
- [4] C. Szegedy et al., Going deeper with convolutions, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 0712June, pp. 19, 2015.
- [5] Belhumeur, P.N., Chen, D., Feiner, S.K., Jacobs, D.W., Kress, W.J., Ling, H., Lopez, I., Ramamoorthi, R., Sheorey, S., White, S., Zhang, L.: Searching the World's Herbaria: A System for Visual Identification of Plant Species. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV 2008, Part IV. LNCS*, vol. 5305, pp. 116–129. Springer, Heidelberg (2008)
- [6] Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: *Proc. 6th Indian Conf. on Computer Vision, Graphics and Image Processing*, pp. 722–729 (2008)
- [7] Liu J., Kanazawa A., Jacobs D., Belhumeur P. (2012) Dog Breed Classification Using Part Localization. In: Fitzgibbon A., Lazebnik S., Perona P., Sato Y., Schmid C. (eds) *Computer Vision – ECCV 2012. ECCV 2012. Lecture Notes in Computer Science*, vol 7572. Springer, Berlin, Heidelberg
- [8] Kaitlyn Mulligan¹ and Pablo Rivas. Dog Breed Identification with a Neural Network over Learned Representations from The Xception CNN Architecture, *Int'l Conf. Artificial Intelligence* (2019)
- [9] <http://vision.stanford.edu/aditya86/ImageNetDogs/>
- [10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. 2014.
- [11] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, Zbigniew Wojna; Rethinking the Inception Architecture for Computer Vision, *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016)
- [12] S.-H. Tsangm “Review: Xception - With Depthwise Separable Convolution, Better Than Inception-v3 (Image Classification),” *Towards Data Science*, 25-Sep-2018. [Online]. Available: <https://towardsdatascience.com/review-xception-with-depthwise-separable-convolution-better-than-inception-v3-image-dc967dd42568>. [Accessed: 14-May-2019].
- [13] <https://www.kaggle.com/gaborfodor/dog-breed-pretrained-keras-models-lb-0-3>

Contributions:

Zhenyang Wang: Model building of VGG16, Xception etc , results analysis

Dingqian Zhao: Data acquisition, pictures generation, paper writing

Kexin Hong: Parameter tuning, paper writing, collect related work

Answer to questions:

Group25:

Q1: Do you have some misclassification results? I think the model may meet some difficulties when classifying some kinds of dogs.

Yes, we have. We didn't demonstrate it because those misclassification doesn't show any regularity. The misclassification belong to different kinds of dogs across the dataset.

Q2: Why did you decide to use logistic regression? What about dense layers?

We know transfer learning as a methodology, use pretrained model mainly in two ways, as a feature extractors cooperating with a classical classifier, or delete the origin top layers and append a new dense layer to do a fine tuning.

Generally, the first method shows better performance on relatively small dataset which has around thousands of pictures like 17flowers dataset, and when the dataset get larger, fine tuning get better performance. In the dog identification task which has 10223 pictures to learn, both methods perform well.

We've looked through several Kaggle kernels which used fine tuning with inception/xception, no result is obviously better than our implementation. But your suggestion is great, it's always necessary to think about

Q3: Do you actually implement these three models? Or do you just use the libraries? There's nothing about model building in your code part.

No, we used the libraries, and actually we can't implement them because the most important thing in the pretrained-model is not 'model' but 'pretrained' parameters.

Q4: Why is your number of breeds only 20? As far as I know, Stanford's dog dataset is a large dataset. So I think it should be larger.

Sorry for the unclarity, as we said in the code demo part, we use 20 breeds to classify just aiming to keep the demo time in 5 minutes otherwise the training time is

too long, and that's why you can see the classification result is much better than what we demonstrate in the slides.

Group84:

Q1: Better demonstrate state-of-art result to compare.

Good point! That's true, it would be very good to compare our result to the state-of-art one and analysis the weakness of our implementation.

Group 87:

Q1: VGG16 seems overfitting. Maybe you can add some dropout layers to improve the result

Great point, but we can't append dropout layers to avoid overfitting because our classifier is a logistic regression classifier rather than a Neural network.

We did have some measures to avoid overfitting like L2 regularize the penalty and a further measure could be cross-validation.

Q2: What kind of picture may cause the model to misclassify?

Refer to answer in Group 25 Q1

Q3: There are too many words in some slides. (ex: p4)

Thanks for advice, we will improve next time.

Q4: What is effect of the transfer learning technique (using pretrained model on ImageNet)?

Transfer learning can largely save training time and improve the result. But there're some requirements to apply transfer learning like the pre-trained model need to be general enough, the number of layers we took from the model etc. If you have further interest, refer to this paper [Distant Domain Transfer Learning](#). It's not a state-of-art paper but gives clear description of transfer learning which still applies today.

Q5: The model used logistic regression after extracting features from the model. What will the results be if fully connected layers are used (since it is the most common way for object classification)?

Refer to answer in group 25 Q2