CS583A: Course Project

Sean Trinh and Hariharan Vijayachandran

May 18, 2019

1 Summary

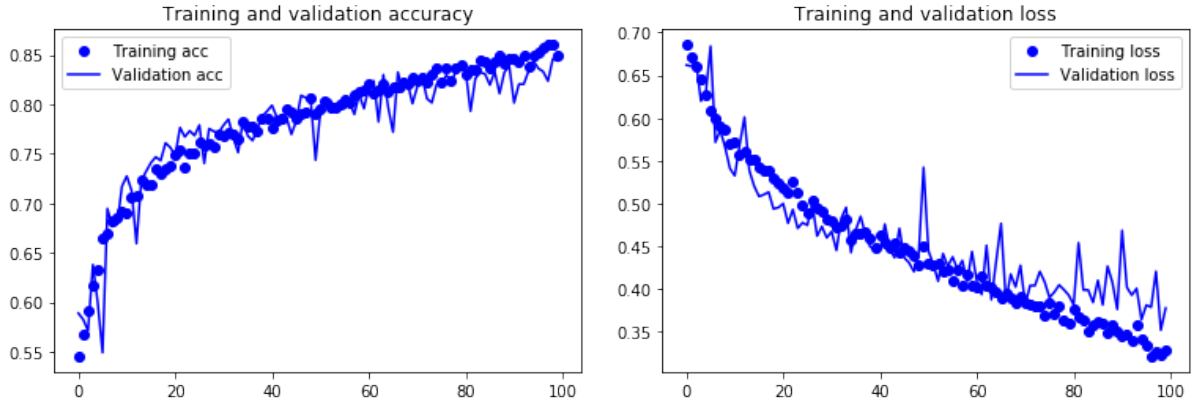
We participated in an active competition on Kaggle. The competition's name is "TMDB Box Office Prediction." Our objective was to, given data about movies, like title, cast, crew, budget, etc., predict the amount of revenue that the movies would make. The final model consisted of an Ensemble model built with three separate models, all of which took data and feed them into a network, differing slightly but each consisting of a few Dense layers. The first model took in budget, popularity, and runtime. The second model took in the number of cast members and the number of crew members. The last model took in the release month and release year of each movie. We then feed these models into the Ensemble model, which itself consisted of a network of a few Dense layers, and outputted revenue predictions. The models were primarily implemented using Keras, but numpy and pandas were used to perform minor, auxiliary functions. The models were trained and run on a MacBook Pro with one 2.2 GHz Intel Core i7 processor and 16 GB of memory. Performance of the model was evaluated on mean squared logarithmic error (MSLE). Our score on the public leaderboard is 2.28782. With this score, at the time of the writing of this report, we rank 631 out of 1179 teams. Since this is still an active competition, we currently do not have a score nor rank on the private leaderboard.

2 Problem Description

Problem. The problem is to classify cats and dogs based on photos. This is a binary classification and image recognition problem. The competition is at `https://www.kaggle.com/c/dogs-vs-cats`. [Use your own words to give a very short description. Do NOT copy the description on Kaggle.]

Data. The data are 256×256 JPEG images. The number of training samples is $n = 25,000$. The number of classes is 2. The training set is well-balanced: $n_{\text{dog}} = 12,500$ and $n_{\text{cat}} = 12,500$.

Challenges. [What makes the problem challenge? E.g., the training set is too small, the data is imbalanced, etc.]



(a) The classification accuracy on the training set and validation set.

(b) The loss on the training set and validation set.

Figure 1: The convergence curves.

3 Solution

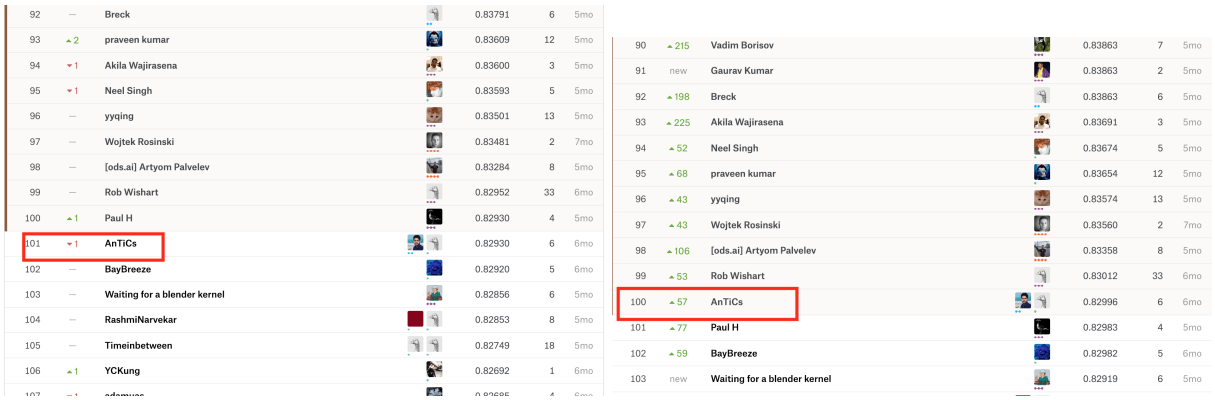
Model. The model we finally choose is the ResNet50 [**?**], a standard deep convolutional neural network. A description of ResNet is online: `https://en.wikipedia.org/wiki/Residential_network`. [Describe your model only if it is non-standard.]

Implementation. We implement the ResNet50 model using Keras with TensorFlow as the backend. [Or, we directly use the ResNet model provided by Keras.] Our code is available at `https://github.com/wangshusen/CS583A-2019Spring/`. We run the code on a MacBook Pro with one Intel i7 CPU and 32 GB memory (or a workstation with 4 NVIDIA GeForce XXX GPUs.) It takes 2.2 hours to train the model.

Settings. The loss function is categorical cross-entropy. The opitmizer is RMSprop. [Specify the other hyperparameters such as learning rate, regularization, epochs, batch size, etc.]

Advanced tricks. [If you used advanced tricks, e.g., pretrain the model on ImageNet and fine-tune it on the Dog-VS-Cat dataset, then write down your tricks and discuss their contribution to the performance, e.g., improve the validation error by 5.6%.]

Cross-validation. We tune the parameters using a 5-fold cross-validation. [If you do not have GPU, you can simply partition the training data to 80%-20% or 90%-10% for hyperparameter tuning.] [Just show the results of your final chosen model.] Figure 1 plots the the convergence curves on 80% training data and 20% validation data. [If you use 5-fold cross-validation, just plot one of the five folds.] [Analyze the convergence curves. E.g., if the training accuracy is much better than the validation accuracy, then the model is likely to overfit the data, and that is why you pretrain your model on ImageNet.]



(a) Private leaderboard.

(b) Public leaderboard.

Figure 2: Our rankings in the leaderboard.

4 Compared Methods

[Try different methods (with brief descriptions) and report their performance.]

5 Outcome

We participated in an active competition. Our score is 0.89311 in the public leaderboard and 0.90112 in the private leaderboard. We rank 100/312 in the public leaderboard and 101/312 in the private leaderboard. The screenshots are in Figure 2.