

CS583A: Course Project

Xuting Tang

November 29, 2019

1 Summary

I participate in an active competition of Kannada digits recognition. The final model I choose is an ensemble of 3 convolutional neural network models, which takes 28×28 images as input and outputs the class labels, data augmentation and early stopping are used. I implement the ensemble using Keras, the competition requires participants to implement code in Kaggle Notebook and submit the prediction to get score. So I implemented code locally in my Jupyter Notebook and uploaded notebook to Kaggle to get test score. The code is scored on both the public test set, as well as a private (unseen) test set. Performance is evaluated on the classification accuracy. On the public leaderboard, my score is 0.97840; I rank 519 among the 900 teams. The private leaderboard will be available after 12/17/2019.

2 Problem Description

Problem. This problem is classifying MNIST like dataset for Kannada handwritten digits: images of handwritten Kannada digits are given and participants are required to return corresponding digits. The competition is at <https://www.kaggle.com/c/Kannada-MNIST>.

Data. The data has 785 columns, the first column is label, and the rest 784 columns are pixel values, these pixel values are integers between 0 and 255, inclusive. The number of training samples is $n = 60,000$. The number of classes is 10.

Challenges. The result of using existing pretrained models like Resnet and VGG16 is not very desired, these pretrained models are trained using ImageNet, which is completely different from this task. Besides, simple CNN can achieve pretty high accuracy, changing hyperparameters and model structure can only improve performance slightly.

3 Solution

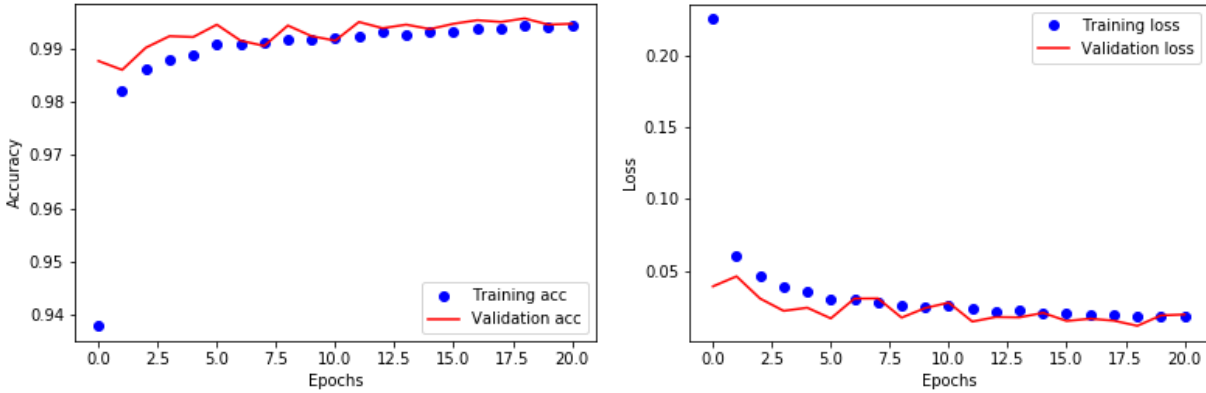
Model. My model is an ensemble of 3 convolutional neural networks.

Implementation. I implemented the model using Keras. The code is available at: .

Settings. The loss function is categorical cross-entropy, and the optimizer is Adam. The number of epochs is 50 but it ends early to prevent overfitting and the best model will be saved and reused for prediction. Data augmentation is used and its batch size is 64. Other settings are default values.

Advanced tricks. Ensemble models, data augmentation and batch normalization.

Validation. The training data is partitioned into training data and validation data. 80% is training data and the rest 20% is for validation. Figure 1 plots the convergence curves of one of the three models on 80% training data and 20% validation data. I specified 50 epochs in the code but it only took 20 epochs and it stopped earlier because of the EarlyStopping callback.



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

Figure 1: The convergence curves.

4 Compared Methods

Fully-connected neural network. I implemented a basic 2-layer fully connect neural network, the width of the first layer is 512 and the second layer's width is 10. This model serves as a baseline model, its training and validation accuracies are 99.84% and 98.65% respectively. Its score on public leaderboard is 0.9478.

Convolutional neural network. I implemented a basic convolutional neural network, this is also a baseline model. It contains 2 combinations of Conv2D and MaxPooling2D layers followed by a Flatten layer and 2 Dense layers. its training and validation accuracies are 99.74% and 99.02% respectively. Its score on public leaderboard is 0.9478.

Convolutional neural network with data augmentation. This is built upon previous CNN model, data augmentation is used to create more training data. Its training and validation accuracies are 99.34% and 99.50% respectively, its score on public leaderboard is 0.9726.

Ensemble of convolutional neural network. This is built upon the baseline CNN. Three models with the same structure are trained separately, their predictions are added up and the class with the maximum value will be the final predicted class. For a single model, its training and validation accuracies are 99.88% and 99.25% respectively. The score of this ensemble on public leaderboard is 0.9766.

Advanced tricks. My final method is an ensemble of 3 convolutional neural network models with data augmentation. For a single modal, the training and validation accuracies are 99.44% and 99.47% respectively. On the public leaderboard, this model's score is 0.9784.

5 Outcome

I participated in an active competition. My score is 0.9784 in the public leaderboard, I rank 519 among the 900 teams in the public leaderboard. The private leaderboard will be available after 12/17/2019. The screenshot of the rank is in Figure 2.












513	GreatGameDota	</> Kannada MNIST CNN		0.97840	1	2mo
514	Peter Xia	</> kaggle_day_1		0.97840	3	1mo
515	kevin joshua			0.97840	4	1mo
516	Keval			0.97840	1	1mo
517	DataDudes			0.97840	9	1mo
518	Sailesh			0.97840	10	3d
519	XT			0.97840	6	4h
Your Best Entry ↑ Your submission scored 0.94780, which is not an improvement of your best score. Keep trying!						
520	KonG	</> Solution with Resnet...		0.97820	7	1mo
521	Sergei Shutov			0.97820	3	1mo
522	竹亦然	</> Kannada-MNIST		0.97820	3	11d
523	Satish C Reddy			0.97820	3	3h

Figure 2: Rank in public leaderboard.