DIABETIC RETINOPATHY DETECTION

Zheming Yin
Electromobility
University of Stuttgart
Stuttgart, 70569
st178328@stud.uni-stuttgart.de

Ziheng Tong
Electromobility
University of Stuttgart
Stuttgart, 70569
st174258@stud.uni-stuttgart.de

February 13, 2023

ABSTRACT

In this paper, we preprocessed and did the data augmentation based on the given IDRID dataset, and built several models to predict binary-classification, multiple-classification and regression problem on the datasets, including simple CNN, VGG, ResNet and Inception-Resnet-V2 by transfer learning. We also used ensemble learning to combine multiple algorithms into a more powerful model. For the models, W&B is used to sweep the hyperparameters to find the best combination in order to improve the performance of model. In the evaluation, we plotted the confusion matrix, ROC-curve, Grad-CAM, dimensional reduction and so on to show the evaluation results intuitively. In addition, we also tried Kaggel EyePACS dataset.

1 Introduction

Using deep learning to detect diabetic retina is a hot topic in the field of medical image processing since traditional detection methods are time-consuming and rely on a large number of medical resources. For remote areas, once the chance of treatment is missed, it will lead to visual loss or even blindness. Therefore, we built some models based on deep neural networks to diagnose diseases and grade their severity.

2 Input pipeline

2.1 TFRecord (Ziheng Tong, Zheming Yin)

We used TFRecord files to build the input pipeline for the IDRID dataset, which includes 413 training images and 103 test images, to accelerate the data import process. In the case of binary classification, the images corresponding to the label 0 and 1 will be marked as label 0, which corresponding to non-proliferative diabetic retinopathy (NPDR), and the rest will be marked as 1, namely proliferative diabetic retinopathy (PDR). In the case of multi-classification, the corresponding label will not be changed.

2.2 Images Preprocessing and Data Augmentation (Zheming Yin, Ziheng Tong)

After reading images and their corresponding labels from the TFRecord files, a series of preprocessing operations were performed according to the problems existing in the dataset, in order to solve the black border and imbalanced data problem:

- Crop the black borders at the left and right ends of the image and reshape the image size to (256, 256, 3).
- In the problem of binary classification, oversample the images with label 1, and in the case of multi-classification, oversample the images with label 1, 3, 4.
- Randomly adjust the contrast, brightness, and angle of the images, and perform a left-right flip operation. We did this augmentation while training, namely on the fly.

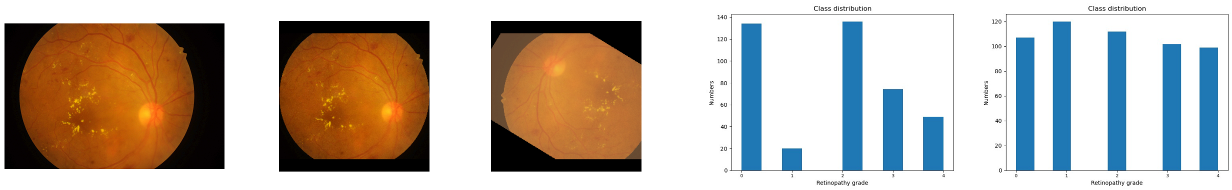

Figure 1: Image Preprocessing and Oversampling

2.3 Kaggle EyePACS (Ziheng Tong)

The EyePACS dataset contains over 88 thousand color fundus images, which is much larger than our dataset. They have also been resized to 1024×1024 and cropped to delete the black space in advance. The labeling is the same as our original dataset. For this EyePACS dataset, we read and combined the TFRecord files stored in the server directly. We then resized the image to 256×256 to fit our model. For this dataset, we only did the multiple classification task for this dataset.

3 Models and Hyperparameter Tuning

3.1 Models (Zheming Yin)

For image processing, we built three CNN-related models, namely the simplified CNN, VGG and ResNet101 models.By using VGG with deeper layers and more parameters, it can learn more complex dataset; one of the advantages of choosing ResNet is that, through the stack of residual blocks, it is used in the process of backward propagation to avoid gradient vanishing, even if the neuron network is very deep.

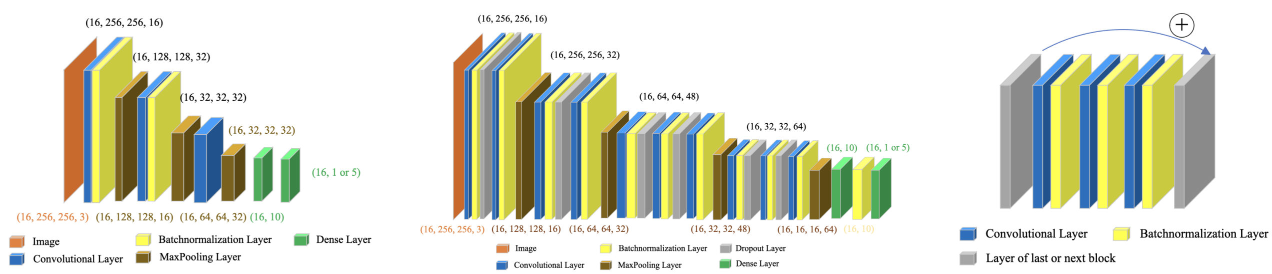

Figure 2: Simple CNN, VGG and Shortcut Connection of ResNet101

3.2 Training with Checkpoint and Early Stop (Ziheng Tong)

During our training process, we used checkpoints to store the trained parameters. Meanwhile, as the epoch increases, the model will tend to be overfitted. So we used early stop to store the best validation accuracy as the checkpoint during training. This technique is only applied after several epochs (e.g., 10 epochs) to avoid the random fluctuation of the validation accuracy. In the subsequent evaluation part, the model at this moment will be restored.
We also used TensorBoard and W&B to visualize and inspect the training process.

3.3 Ensemble Learning (Zheming Yin)

We ran the above-mentioned models separately, then stack the prediction lists along the first axis, and use the form of voting to select the label with the highest probability.

3.4 Transfer Learning (Zheming Yin)

In transfer learning, we selected Inception-Resnet-V2. The consideration is that this model can greatly reduce the number of parameters and the computational complexity, and have high non-linearity characteristics while ensuring the accuracy.

Figure 3: Early Stopping

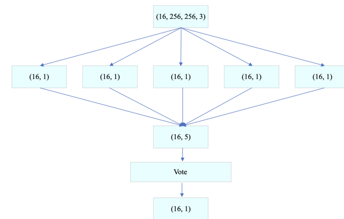

Figure 4: Ensemble Learning

3.5 Hyperparameter Optimization (Ziheng Tong)

The baseline models always cannot achieve the best performance without fine-tuned hyperparameters. In order to improve the performance of our models, we applied hyperparameter optimization on the VGG model. In this case, we set the batch size, learning rate and dropout rate as the sweeped parameters using Bayes method in W&B.

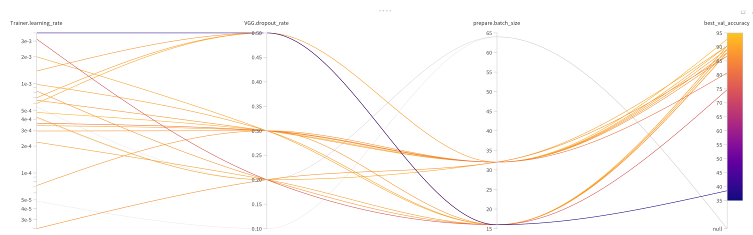

Figure 5: Hyperparameters Sweep with W&B

4 Evaluation and Visualization

4.1 Evaluation (Zheming Yin, Ziheng Tong)

To verify the training result of the model, we also need evaluate the trained model on the unseen data, namely test dataset. In our evaluation step, we use the parameters restored from checkpoint to predict images in test set and compare it with the ground truth.

- Binary-classification

Binary-classification	Accuracy	Precision	Recall	F1-score	Sensitivity	Specificity
Simple CNN	58.3%	54.7%	54.7%	54.7%	71.4%	46.3%
VGG	70.9%	93.8%	93.8%	93.8%	69.8%	76.5%
ResNet101	66.0%	84.4%	84.4%	84.4%	68.4%	58.3%
Inception-Resnet-V2	77.7%	81.2%	81.2%	81.2%	82.5%	70.0%
Ensemble Learning	68.3%	63.1%	63.1%	63.1%	72.3%	46.4%

- Multiple-classification

Multiple-classification	Accuracy	Precision	Recall	F1-score
Simple CNN	31.1%	19.1%	28.7%	21.8%
VGG	48.5%	53.2%	38.0%	38.9%
ResNet101	30.1%	24.8%	24.4%	24.0%
Inception-Resnet-V2	43.7%	35.2%	31.9%	31.9%
Ensemble Learning	37.2%	23.9%	26.8%	24.7%

- Evaluation with tuned VGG

Type	Accuracy	Precision	Recall	F1-score	Sensitivity	Specificity
Binary classification (Tuned)	81.6%	75.0%	75.0%	75.0%	94.1%	69.2%
Multiple classification (Tuned)	53.4%	39.4%	39.9%	39.2%	-	-

- Multiple classification of Kaggle EyePACS using tuned VGG model: The highest test accuracy is 81.5%.

4.2 Confusion Matrix and ROC-Curve (Zheming Yin)

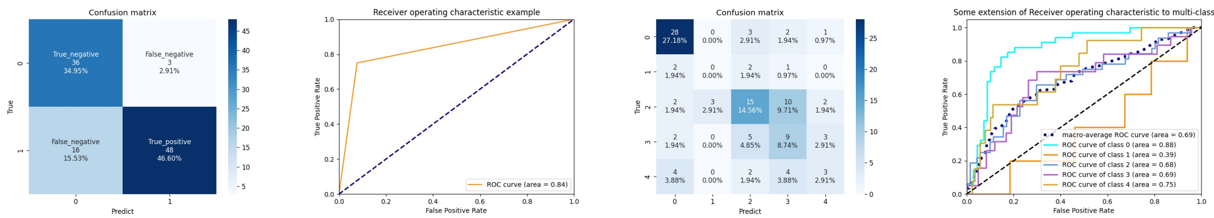

Figure 6: Confusion matrix and ROC for binary and multiple classification

4.3 Visualization (Zheming Yin)

For evaluation visibility, we used three methods, namely Grad-CAM, Guided Backpropagation and Guided Grad-CAM. It can be seen from the figure that the focus of the model is on the side of the eyeball, but there are still some images that focus on the bright spot of the pupil, which is also one of the reasons for the poor training effect of the model.

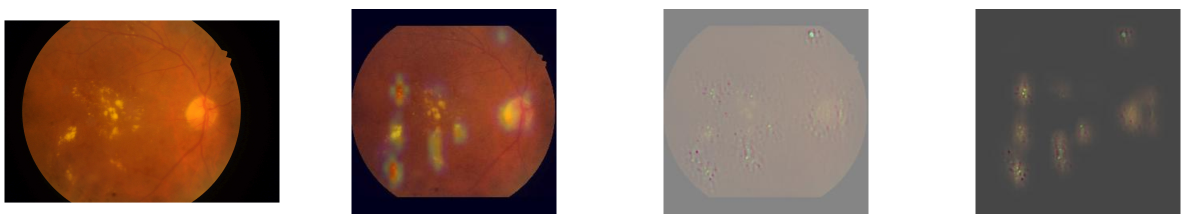

Figure 7: Deep Visualization (Original image, Grad CAM, Guided Backpropagation, Guided Grad CAM)

4.4 Dimensional Reduction (Zheming Yin)

Use the Umap method to do the dimensional reduction from the last convolutional layer to obtain its corresponding two-dimensional coordinates, which can be used to observe the categories of false predictions and improve the model.

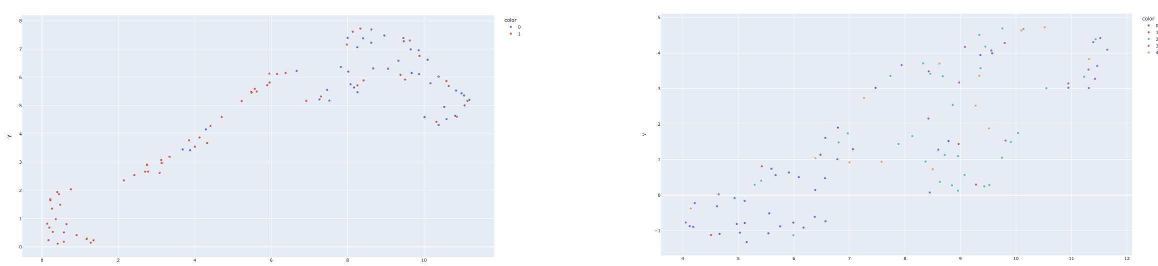

Figure 8: Dimensional Reduction for binary and multiple classification

5 Conclusion

To summarize, we can achieve good test result after the hyperparameter tuning of VGG model on binary classification for the task of diabetic retinopathy detection. And the corresponding deep visulization and dimensional reduction are also significant compared to other models.
From this project, we can also conclude that in the deep learning, the result can be improved by applying preprocessing and data augmentation as well as the modification of the model architectures and hyperparameters.
However, the 5-class classification is not so significant and thus need further research. For example, we can focus on a larger dataset and mining more information of the images.