Progress Report

White Blood Cells Classification with Deep Learning

Shilong Yin sy2792 Diwei Xiong dx2183

Model Structure

In this project, we would like to recognize different type of white blood cells, including Neutrophil, Eosinophil, Lymphocyte and Monocyte, in a large number of images using deep neural networks. As we mentioned earlier in our project proposal, convolutional neural network is an optimal choice for image recognition due to its computational efficiency, high accuracy and easy-to-implement feature. Therefore, we will base our model on CNN.
We adopt the model structure from a kernel on the Kaggle website.[1] The structure of our model is shown below:

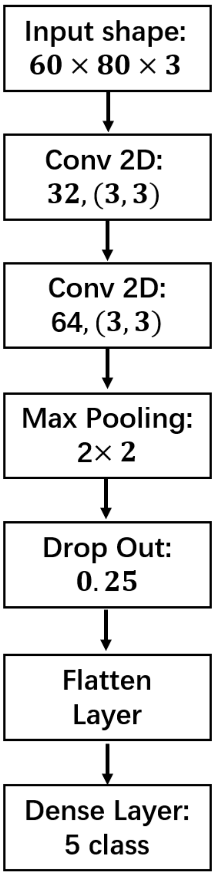

Fig. 1: Model structure

For our model, we will first resize our image to 60*80 to reduce the amount of computation; then, we use 2 conv2D layer, one with 32 units and 3X3 kernel size and the other with 64 unites. After that, we apply a Max-pooling layer with size of 2X2, then adding a dropout of 0.25 to reduce the amount of connections, followed by a flatten layer. Then, we apply dense layer and another dropout layer to further train the model. Finally, we apply a softmax function to pick the prediction with highest probability as our final prediction result.

Model evaluation

As for the error metric, we will use categorical cross entropy to calculate loss and backpropagate. Besides, since we have labels to the test data, we would compare the prediction results of our deep neural network model to the true labels, and use the test accuracy after training as the criterion to evaluate our model.

In the training process, we can also record the loss and accuracy of both our training data and test data and plot them afterwards, so that we can qualitatively assess the goodness of our model.

Finally, other criterions such as confusion matrix and f1 score are also appropriate metric we can consider in terms of evaluating our model. After getting the prediction results, we would like to plot a confusion matrix to visualize the accuracy of our model in terms of each class.

Initial result

Currently, we only set up the basic structure of the model according to the kernel and haven't tuned the parameters, so the result is not good. The test loss after training for 10 epochs is about 1.38 and the accuracy is 0.27. This is certainly not a satisfying result since the baseline is 0.25. So in the later stages, we will improve our model in the following aspects:[2]

- Increase the number of epochs.
- Change model structure: increase units of layer, add layers, change the convolution kernel size and so on.
- Split training data into train and validation dataset, and use validation in the training to early stop or adjust learning rate.
- Use batch normalization before apply input to the model.
- Use pretrained model as a base for our model
- Tuning hyperparameters according to model performance.
- Adjust the input size and preprocess methods.

Reference

[1] Paul Mooney. Identify Blood Cell Subtypes from Images. [Online] Available: https://www.kaggle.com/paultimothymooney/identify-blood-cell-subtypes-from-images

[2] Ian Goodfellow and Yoshua Bengio and Aaron Courville. Deep Learning. MIT Press. 2016.