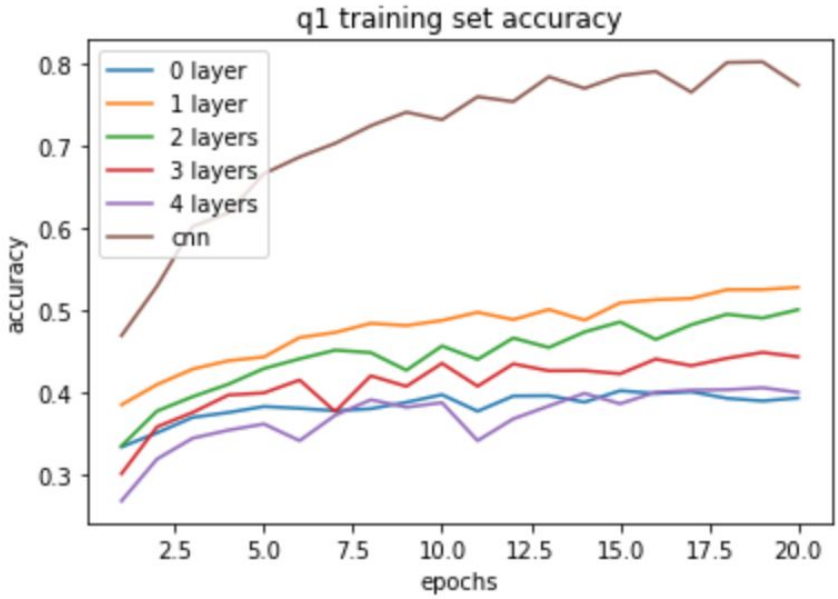
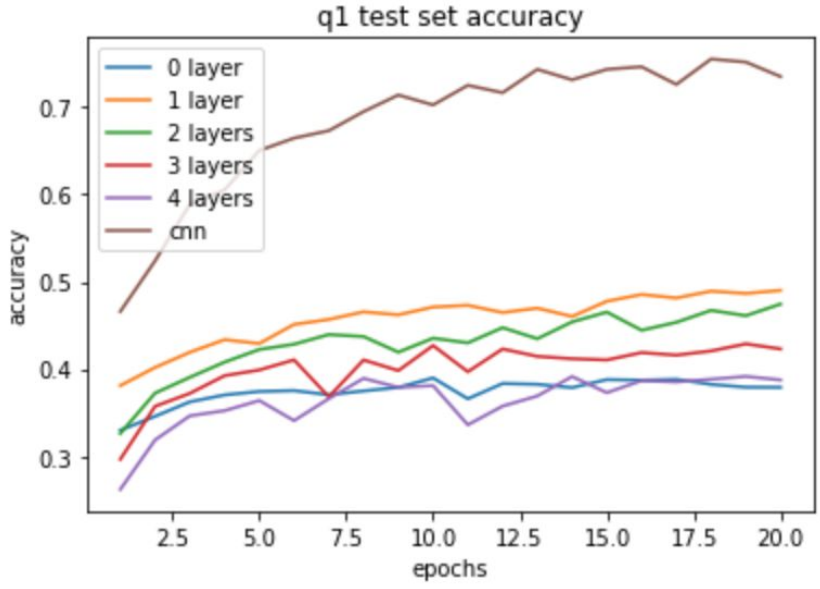
Bolun Yao

ID: 20615120

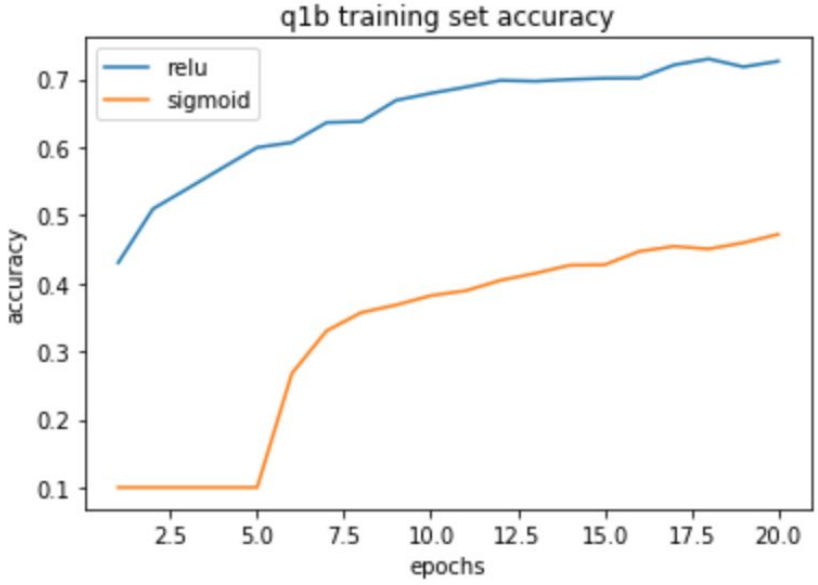
Q1(a)

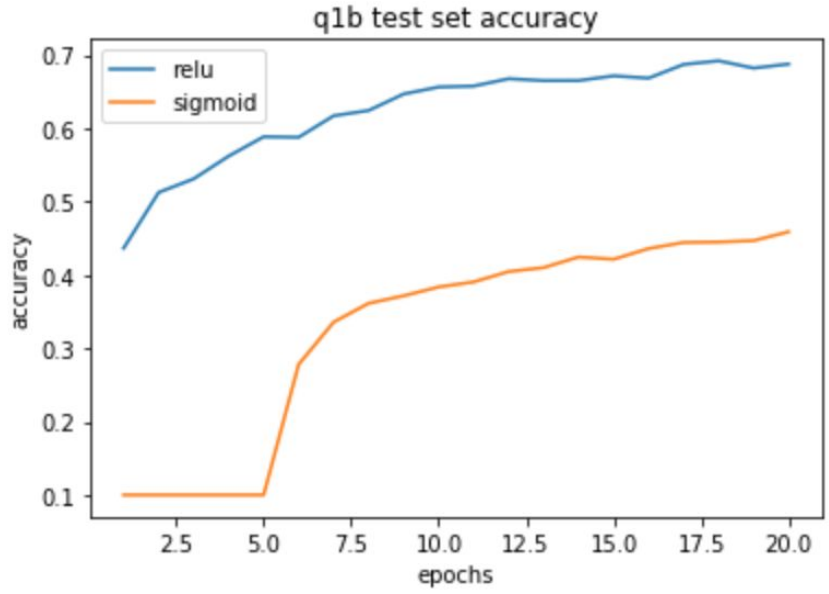




The convolutional NN is the best since the model is sparse using shared weights. It is easy to optimize. For the 5 different layer densed NN, 1 hidden layer is the best. In theory, deep neuron network can get the same result as the single hidden layer NN does. However, Since deep NN has more weights, the optimizer in this case fails to find the best solution.

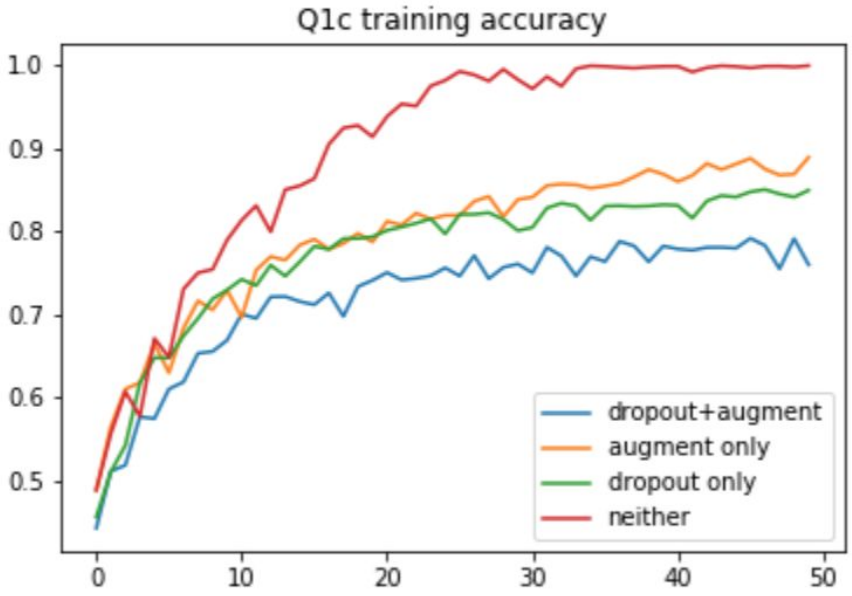
(b)

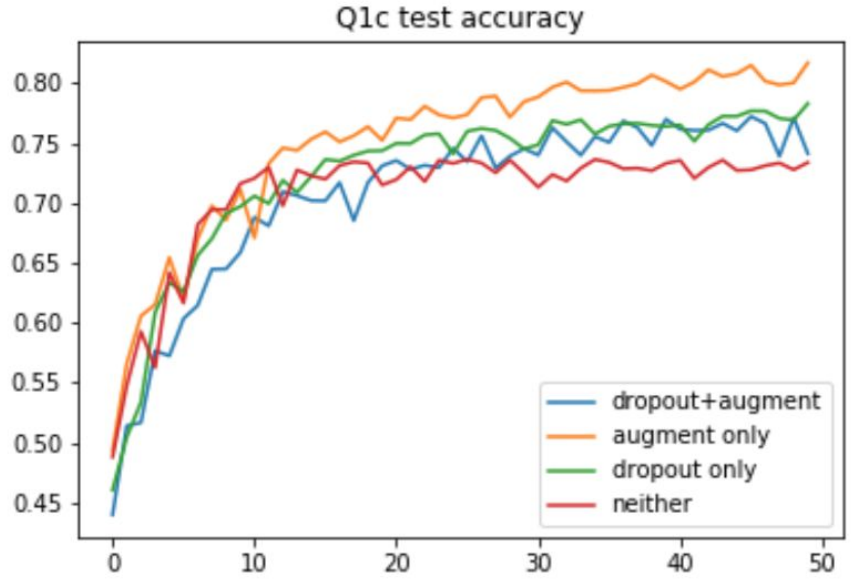




NN with Relu is much better than NN with sigmoid since sigmoid activation function causes the gradient vanishing problem.

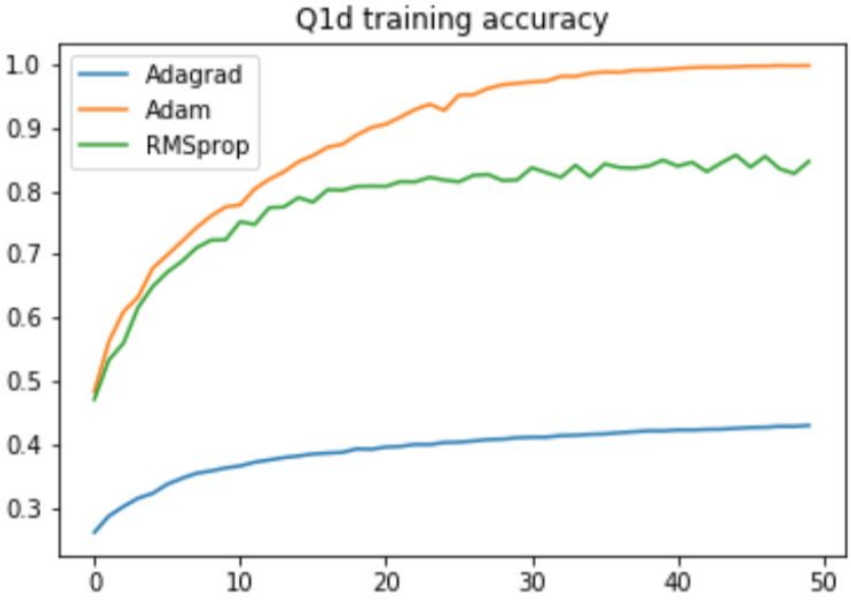
(c)

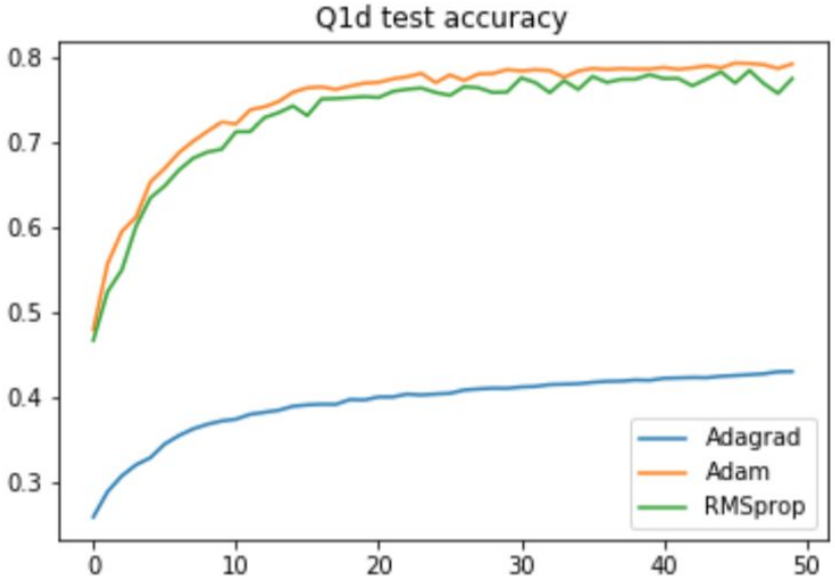




NN without dropout shows the overfitting(since the test accuracy is lower than training accuracy from the graphs above), it's obvious while using data augmentation. The data augmentation helps to improve the accuracy. However, the dropout negatively affect the accuracy when using data augmentation.

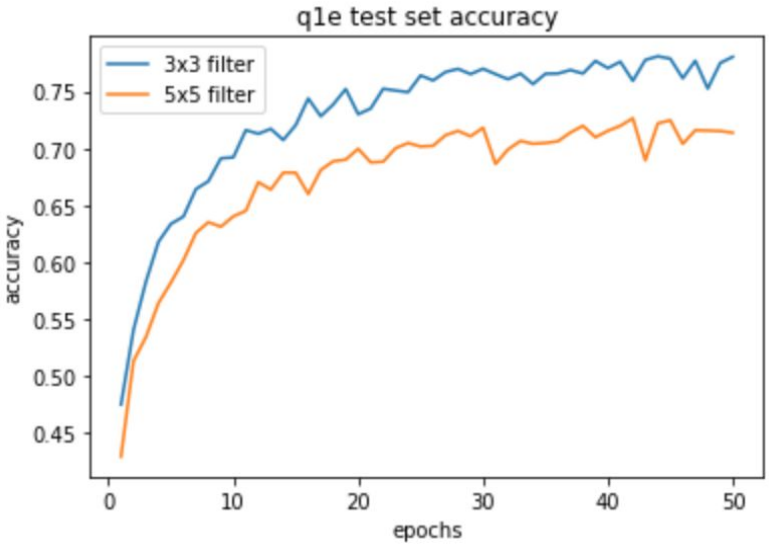
(d)





Adagrad is the worst optimizer in this case since adagrad has the small learning rate to better solve the large gradient, which makes it learning very slowly in the negative side. Adam learn and RMSprop have equally excellent performance in test. However, Adam learns faster. This is because Adam keeps an exponentially decaying average of past gradients mt, similar to momentum. This makes Adam have a very fast learning rate.

(e)



CNN with 3x3 filters is better than CNN with 5x5 filters, since a stack of small filters are better than a single larger filter.

Q2(a)

No, it's not translation invariant. Since it's asked to move the foreground of the image by at most 10 pixels, meanwhile, the "padding" of the required recognized image is also 10 pixels. That means the filters including object of interests in convolution layer are different between translation and no translation. Thus, the activated hidden layer units are different, which leads to different output for neural network.

(b)

No, it's not translation invariant. Similarly to (a), the filters in convolution layer are different. Besides, using the max pooling can only achieve local translation invariant since it highlights the important feature in each filter, but the question applies the global translation. That leads to different units being activated in pooling layer. As a result, the output of neural network are different between translation and no translation.