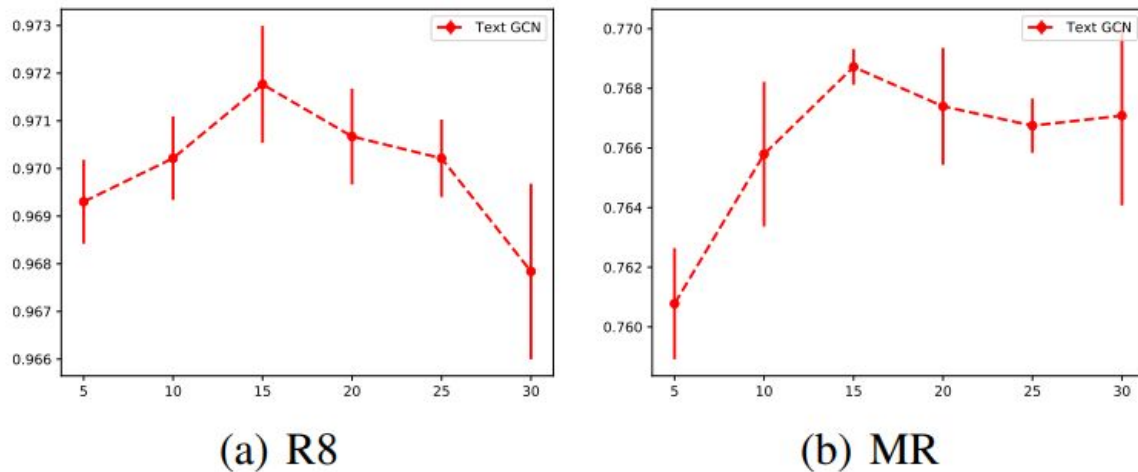


Figure 1

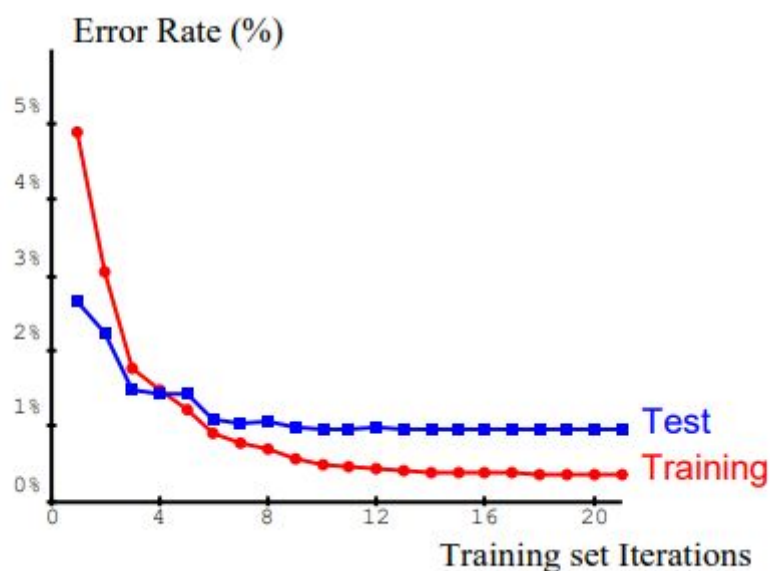


Yuan Luo, et al. Graph Convolutional Networks for Text Classification, Figure 2

For this figure, the input is R8 and MR training set and different sliding window sizes, and the output is test accuracy. And the desired function to learn is GCN, Graph Convolutional Network. The real-world problem in this project is text classification. The R8 dataset can be obtained at <https://www.cs.umb.edu/~smimarog/textmining/datasets/>. And The MR dataset can be obtained at <http://www.cs.cornell.edu/people/pabo/movie-review-data/>.

In this project, I will try to learn and implement GCN by using library or code based on the understanding of GCN. Then I can use two mentioned dataset to get test accuracy at different window sizes.

Figure2



Yann, et al. Gradient-Based Learning Applied to Document Recognition, Figure 5

For this figure, the input is MNIST dataset, and the output is test error rate and training error rate. And the desired function to learn is LeNet5

The real-world problem in this project is classification of different handwritten digits. And the dataset can be obtained at <http://yann.lecun.com/exdb/publis/index.html#lecun-97>. Or, it can be obtained at Tensorflow for Python.

In this project, I will try to learn and implement LeNet5. The pseudo-code shows the processing:

Input--Convolution-->C1--Subsampling-->S2--Convolution-->C3--Subsampling-->S4--Full connection-->C5--Full connection-->F6-->RBF-->output

After reconstructing LeNet5, it's possible to use training set to get training error rate, then use training set and test set to get test error rate.