# Your Name

남궁영수

Your Email address

ngys321@naver.com

# 1. Training Set Construction (5 pts)

Construct the training set for the amazon review dataset as instructed and report the following statistics.

| **Statistics** |  |
| --- | --- |
| the total number of unique words in T | 24473 |
| the total number of training examples in T | 2000 |
| the ratio of positive examples to negative examples in T | 1 |
| the average length of document in T | 187.677 |
| the max length of document in T | 3816 |

# 2. Performance of deep neural network for classification (20 pts)

Suggested hyperparameters:

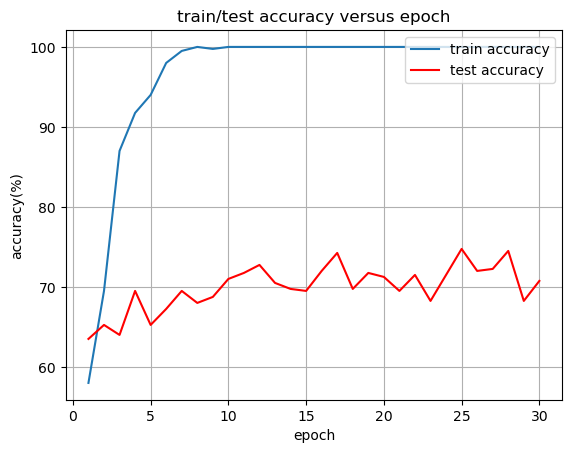
1. Data processing:
   1. Word embedding dimension: 100
   2. Word Index: keep the most frequent 10k words
2. CNN
   1. Network: Word embedding lookup layer -> 1D CNN layer -> fully connected layer -> output prediction
   2. Number of filters: 100
   3. Filter length: 3
   4. CNN Activation: Relu
   5. Fully connected layer dimension 100, activation: None (i.e. this layer is linear)
3. RNN:
   1. Network: Word embedding lookup layer -> LSTM layer -> fully connected layer(on the hidden state of the last LSTM cell) -> output prediction
   2. Hidden dimension for LSTM cell: 100
   3. Activation for LSTM cell: tanh
   4. Fully connected layer dimension 100, activation: None (i.e. this layer is linear)

|  | Accuracy | Training time(in seconds) |
| --- | --- | --- |
| RNN w/o pretrained embedding | 74.75 % | < 3198 s (30 epochs) |
| RNN w/ pretrained embedding | 88.0 % | < 4360 s (30 epochs) |
| CNN w/o pretrained embedding | 76.75 % | < 247 s (30 epochs) |
| CNN w/ pretrained embedding | 86.0 % | < 800 s (30 epochs) |

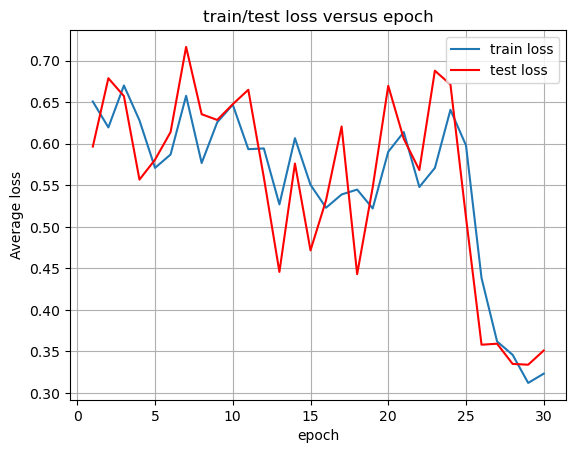
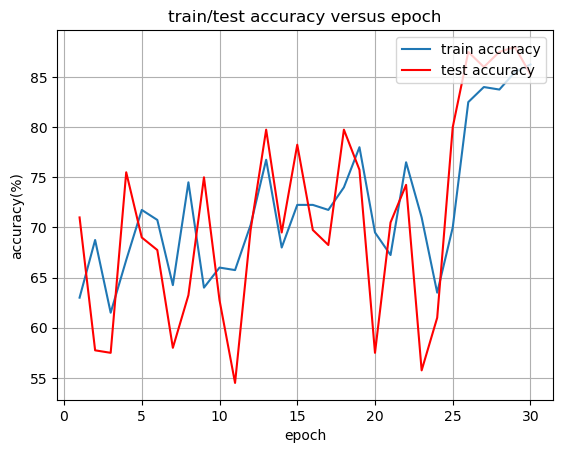
# 3. Training behavior (10 pts)

Plot the training/testing objective, training/testing accuracy over time for the 4 model combinations (correspond to 4 rows in the above table). In other word, there should be 2\*4=8 graphs in total, each of which contains two curves (training and testing).

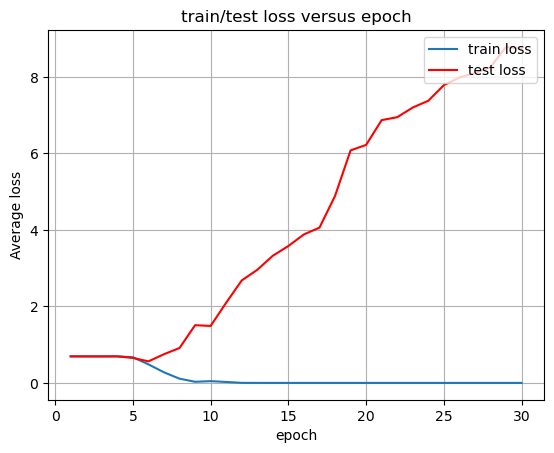
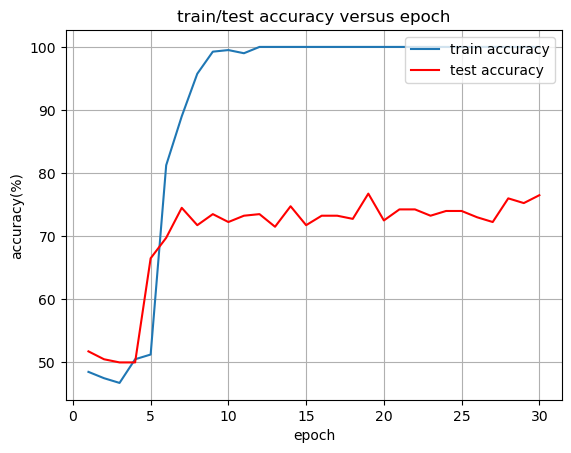
RNN w/o pretrained embedding

* training/testing objective over time
* 
* training/testing accuracy over time
* 

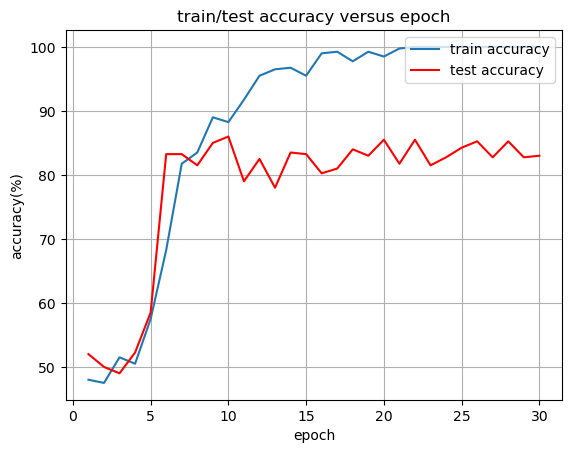
RNN w/ pretrained embedding

* training/testing objective over time
* 
* training/testing accuracy over time
* 

CNN w/o pretrained embedding

* training/testing objective over time
* 
* training/testing accuracy over time
* 

CNN w/ pretrained embedding

* training/testing objective over time
* 
* training/testing accuracy over time
* 

# 4. Analysis of results (10 pts)

Discuss the complete set of experimental results, comparing the algorithms to each other. Discuss your observations about the various algorithms, i.e., differences in how they performed, different parameters, what worked well and didn't, patterns/trends you observed across the set of experiments, etc. Try to explain why certain algorithms or approaches behaved the way they did.

Using pre-trained embedding makes results better. It gives more high test accuracy, less overfitting, which means that the gap between train accuracy and test accuracy are smaller than that of non-pre-trained one.

RNN based models showed higher test accuracy than CNN based one. But RNN based model needs more training time, almost like 4 times more than a CNN based one.

Pattern that I found here is that the training accuracy and test accuracy are going higher at first several epochs, like 5 - 7 epochs, and converge with some gap between training accuracy and test accuracy.

# 5. The software implementation (5 pts)

# Add detailed descriptions about software implementation & data preprocessing, including:

1. A description of what you did to preprocess the dataset to make your implementations easier or more efficient.

2. A description of major data structures (if any); any programming tools or libraries that you used;

3. Strengths and weaknesses of your design, and any problems that your system encountered;