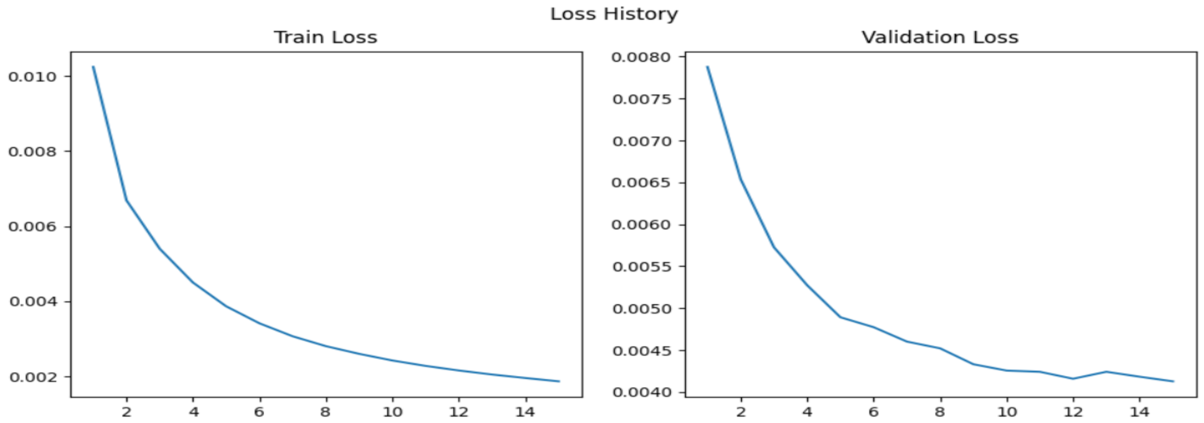
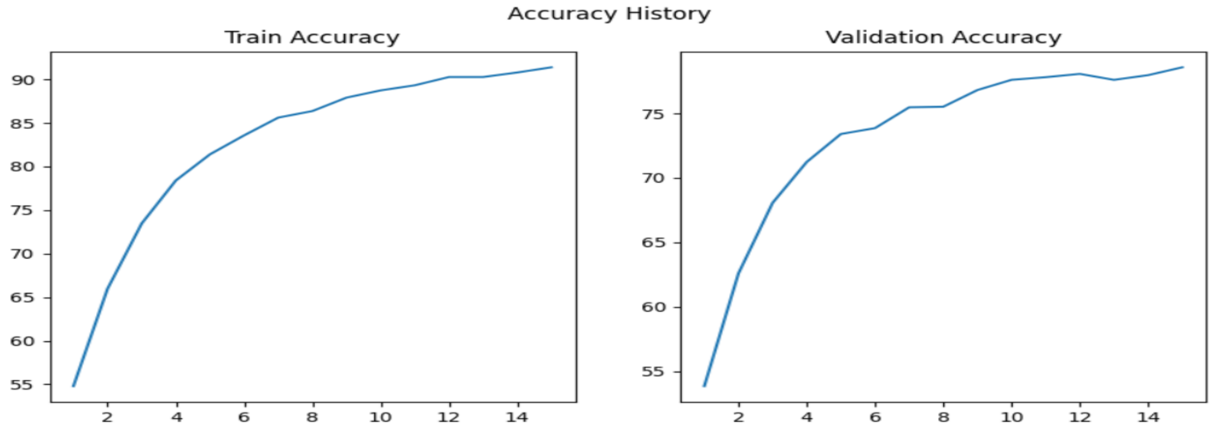
Students: Nadav Spitzer, 302228275. Lior Frenkel, 204728315

Part 5

NER task – With pretrained embeddings

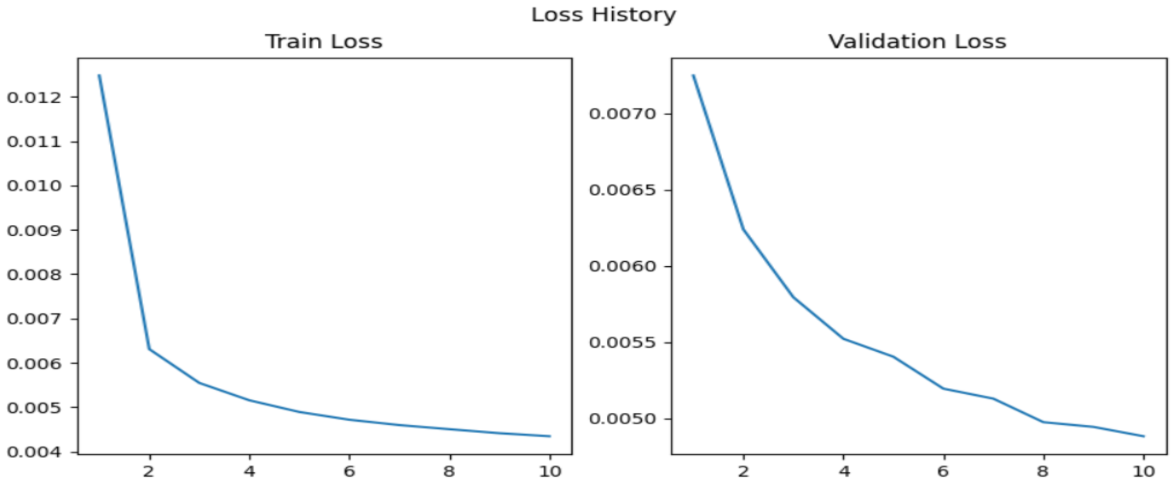
1. Parameters:
   a. Number of epochs: 15
   b. Learning rate: 1e-4
   c. Batch size: 32
   d. Hidden layer size: 150
   e. Optimizer: Adam
   f. Dropout with 0.5 probability
   g. Num of filters: 30
   h. Filter size: 3
2. Results:
   a. Train loss: 0.001865
   b. Train accuracy: 91.392%
   c. Validation loss: 0.004127
   d. Validation accuracy: 78.584%
3. Graphs:

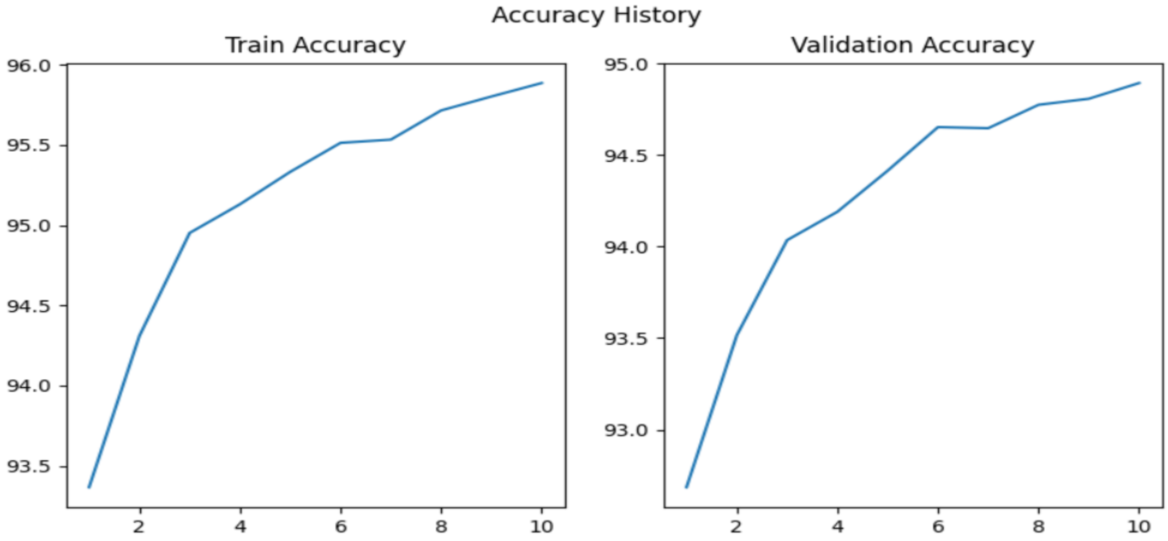




POS task – With pretrained embeddings

1. Parameters:
    a. Number of epochs: 15
    b. Learning rate: 1e-4
    c. Batch size: 32
    d. Hidden layer size: 150
    e. Optimizer: Adam
    f. Dropout with 0.5 probability
    g. Num of filters: 150
    h. Filter size: 3
2. Results:
    a. Train loss: 0.0043477
    b. Train accuracy: 95.887%
    c. Validation loss: 0.004881
    d. Validation accuracy: 94.895%
3. Graphs:





Different experiments:

1. We tried different number of filters for the POS task. The base case was with output dim of 30 filters. Reducing the filters to 15 didn't affect the scores in any way, but increasing it to 150 showed a little increase in the preformence (from 94.2% to 94.8% DEV accuracy).
2. We tried different number of filters for the NER task. The base case was with output dim of 30 filters. Reducing the filters to 15 and increasing them to 150 didn't affect the scores in any way.
3. Different filter size did not affect the scores in any way.

Meaningful Filters:

We found some meaningful filters in our tasks. The way to achieve that was by taking the model after training and run on the TEST dataset. When looking at the results of the Max-Pooling layer, we can tell which Convolution Filter was the most effective on the current word (the index of the highest score after the pooling). After achieveing the filter, we checked what is the best 'triplete' of letters in that word – that is, the index of the best result in the Conv filter.

Some of the results we achieved for the POS task:

For word: "influential" best characters are: ['i', 'n', 'f'] on filter #27

For word: "legislation" best characters are: ['l', 'e', 'g'] with filter #27

For word: "capital" best characters are: ['c', 'a', 'p'] with filter #31

For word: "bailout" best characters are: ['b', 'a', 'i'] with filter #31

From this we can conclude that these filters act very well on the prefix of the words.

Some of the results we achieved for the NER task:

For word: "defeat" best characters are: ['<s>', 'd', 'e'] with filter #89

For word: "friday" best characters are: ['<s>', 'f', 'r'] with filter #89

For word: "emirates" best characters are: ['e', 's', '</s>'] with filter #31

For word: "defence" best characters are: ['c', 'e', '</s>'] with filter #31

For word: "against" best characters are: ['s', 't', '</s>'] with filter #31

For word: "advancing" best characters are: ['i', 'n', 'g'] with filter #96

For word: "chinese" best characters are: ['s', 'e', '</s>'] with filter #96

For word: "despite" best characters are: ['p', 'i', 't'] with filter #76

For word: "outsiders" best characters are: ['i', 'd', 'e'] with filter #76

For word: "meeting" best characters are: ['t', 'i', 'n'] with filter #76

Here we have a not interesting result – Filter #31 and #96 seems to work very well with suffixes of words while filter #89 works with prefixes and Filter #76 works with combinations of words in the middle.