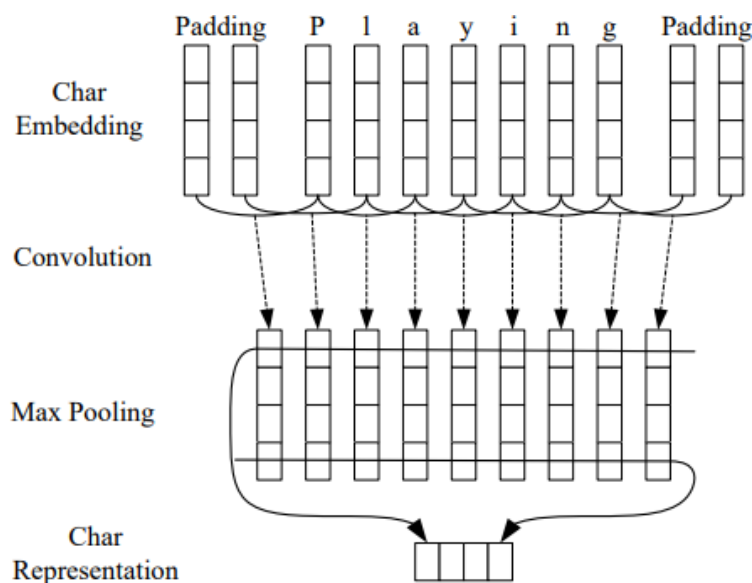


Deep Learning - 89687

Ass2 – Part 5

The model:

We use CNN to extract character-level representation of a given word. We separate each character of each word and use padding. We use only character embeddings as the inputs to CNN. Then we use DropOut and MaxPooling and send this to our model.



POS

The parameters we use are:

- Adam Optimizer
- One hidden layer with 200 neurons
- Learning rate = 0.01
- Number of epochs = 25 (we stop when it converges).
- Batch size = 32
- Loss = Cross Entropy
- Activation function = tanh
- Window_size = 3
- Filter_size = 30

Accuracy = 96.23%

NER

The parameters we use are:

Lea Setruk
Yoel Benabou

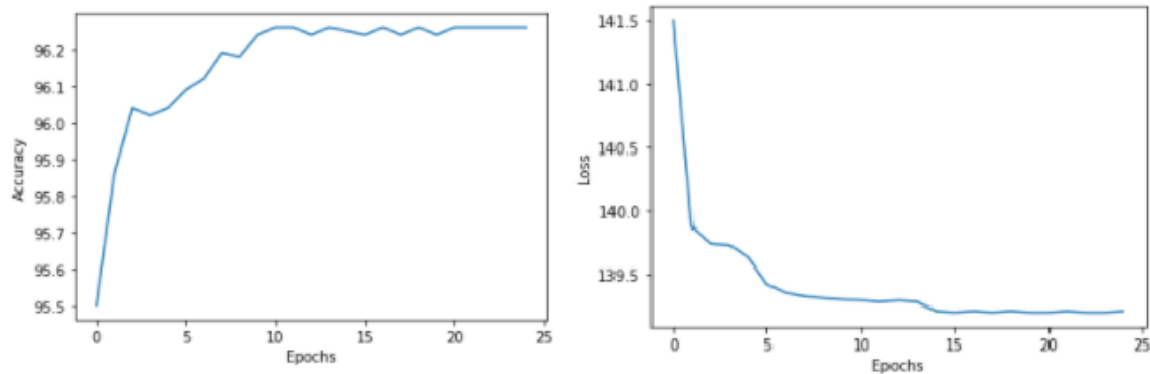
- Adam Optimizer.
- One hidden layer, with 200 neurons.
- Learning rate = 0.01
- Number of epochs = 25 (we stop when it converges).
- Batch size = 32
- Loss = Cross Entropy
- Activation function = tanh
- Window_size = 5
- Filter_size = 40

Accuracy = 76.6%

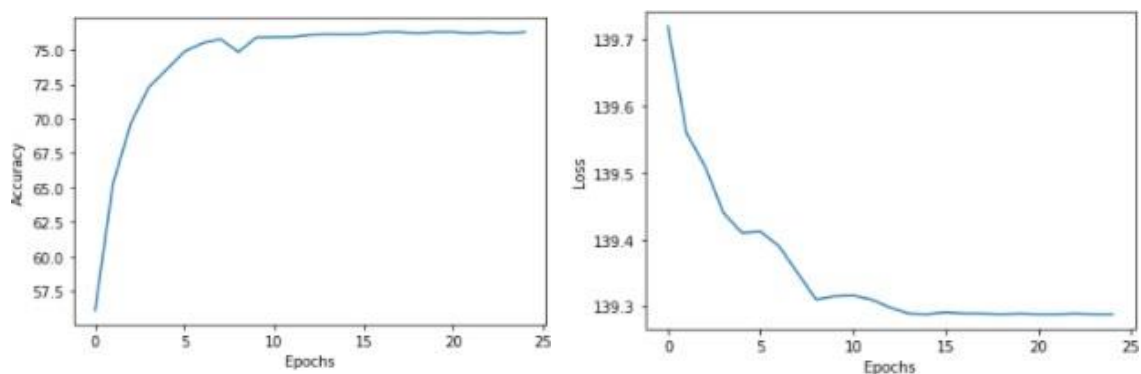
We notice that this method really increases our results. Especially for POS, it gives us a great accuracy.

Graphs

Accuracy / Loss POS:



Accuracy / Loss NER:



Analysis of filters and windows

We analyze the behavior with filters of different sizes: 20, 30 and 40 and windows of different sizes: 3, 4 and 5.

POS:

We notice that there is almost no difference between windows of size 3 and of size 4, but 3 gives us slightly better results (76.2%). With a window size of 5 words, the accuracy is a bit lower (75.9%). The reason can be that there is “too much not needed information”, it might also produce more noise. We didn’t notice differences between the different sizes of filters.

NER:

We notice that with a window size of 5, the accuracy is slightly better. The reason to that would be that, in NER, all the characters have an important meaning.

For the filters’ sizes, it is the same. The model learns better with a larger filter (for 40 filters, the accuracy is 76.2 and with filter 20, it is 75.3%).

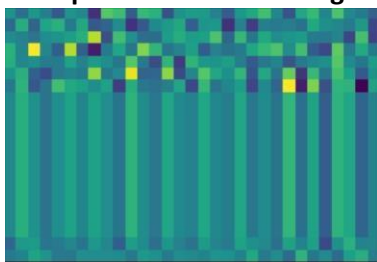
Learned Filters

We extracted the convolution activation maps of our models. We then applied them on some words in order to find patterns.

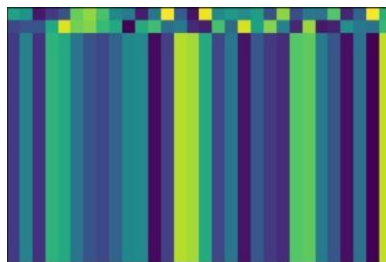
The filters created heat maps. When it’s light and clear, it means that the filter looks at the specific character.

It seems that in POS, the filter focuses on groups of characters at the start of the word or in the middle. In NER, it seems that the filter focuses on specific characters in the word.

Example with the word ‘eager’:



POS



NER

(Due to the situation, and problems of wifi in the bunker room for colab, this analysis is shorter and not complete as we would have wished. Thank you for your understanding.)