Part 5

Refael shaked Greenfeld 305030868

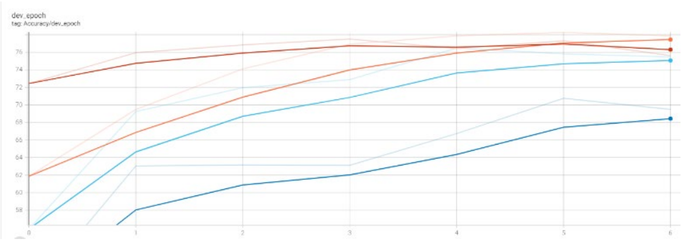
Danit Yshaayahu 312434269

First, in order to compare all the runs, we freeze the hyper parameters and the model seed in sake of comparability and only changed the model. Both between part 1,3,4 and 5 and both between the different window size and number of filters. See graphs below, when the - orange is the part 5 and the red is part 4 with the same hyper parameters.

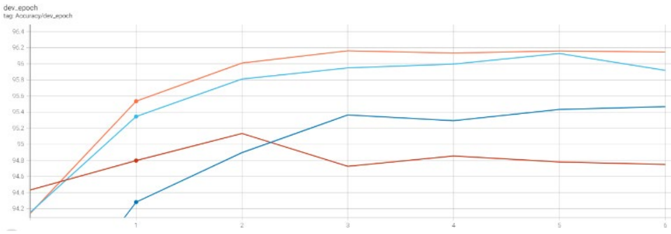
In the technical view, we chose to pad the words to the length of 20 taking, and if there is a word with more than 2 characters we would take only the first 20 and only their vectors would be an input to the Conv. We chose the length of 20 by the distribution of the length in the English language.

As you can see, the model in part 5 out preforms all the other model (at list in this specific setup – disclaimer, we didn't check the mean of various seeds due lack of computation power and lack of time). Although there is slight improvement in the NER, it is by 2 points, but you can see that if we continued more than 7 epochs we would reach an higher improvement. Regarding the POS, we believe 96.2 is close to the limit of the data. And it is really hard to tell if there is more room for improves although the CNN Char model does yield the best results.

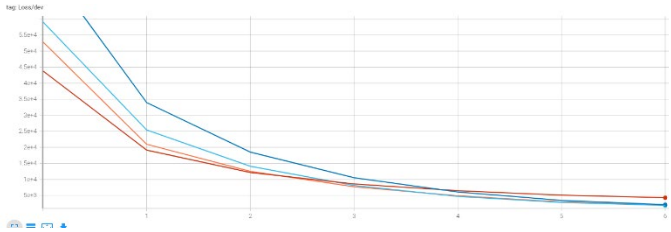
NER Accuracy – Part 5 is the Orange

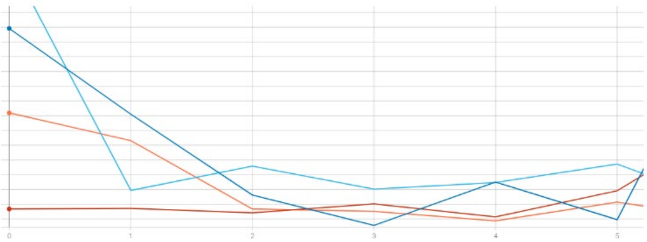


POS Accuracy – Part 5 is the Orange



Loss Dev all models NER and POS – part 5 is the orange





We examined a set of 3 types of filters: 20, 30 (the one used in the paper) and 40 and a set of 3 window sizes 3 (the one used in the paper), 4 and 5.

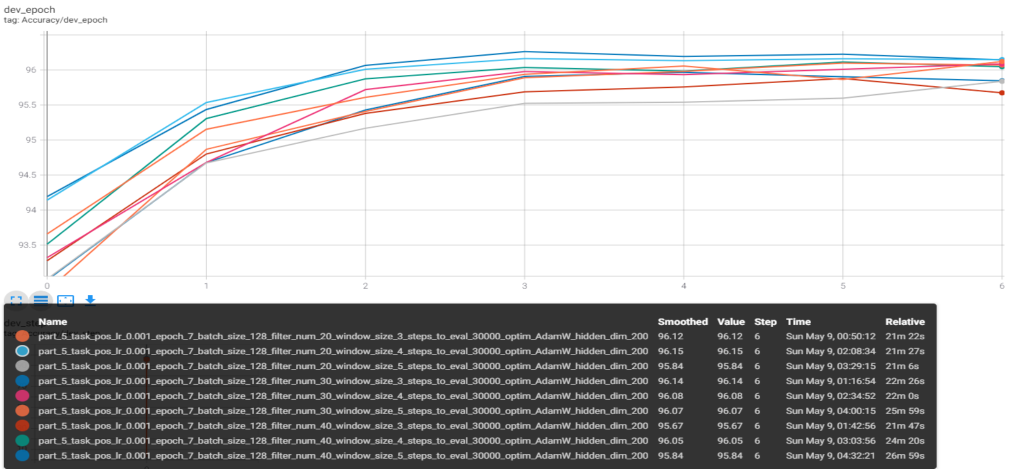
Pos:

Window Effect:

We can observe in the graph below that there is almost no difference between window size 4 and window size 3 where all the results were very similar to each other. In the other hand, size 5 was a bit more scattered and with a drop of half precent. We believe that it due to the fact that there is more signals is triplet or quadrants of words and a window of five is a bit too much in order to get signal for the POS tag. Also many of the words are shorter than 5 words what would result a vast use of padding and maybe more noise.

Filter Effect:

In the sense of filters 20 was the best with 96.14, but the difference was really minor between the 3 (96.07-96.14)
In the graph below you may see the various accuracy of the runs



	Name	Smoothed	Value	Step	Time	Relative
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_20_window_size_3_steps_to_eval_30000_optim_AdamW_hidden_dim_200	96.12	96.12	6	Sun May 9, 00:50:12	21m 22s
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_20_window_size_4_steps_to_eval_30000_optim_AdamW_hidden_dim_200	96.15	96.15	6	Sun May 9, 02:08:34	21m 27s
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_20_window_size_5_steps_to_eval_30000_optim_AdamW_hidden_dim_200	95.84	95.84	6	Sun May 9, 03:29:15	21m 6s
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_30_window_size_3_steps_to_eval_30000_optim_AdamW_hidden_dim_200	96.14	96.14	6	Sun May 9, 01:16:54	22m 26s
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_30_window_size_4_steps_to_eval_30000_optim_AdamW_hidden_dim_200	96.08	96.08	6	Sun May 9, 02:34:52	22m 0s
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_30_window_size_5_steps_to_eval_30000_optim_AdamW_hidden_dim_200	96.07	96.07	6	Sun May 9, 04:00:15	25m 59s
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_40_window_size_3_steps_to_eval_30000_optim_AdamW_hidden_dim_200	95.67	95.67	6	Sun May 9, 01:42:56	21m 47s
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_40_window_size_4_steps_to_eval_30000_optim_AdamW_hidden_dim_200	96.05	96.05	6	Sun May 9, 03:03:56	24m 20s
●	part_5_task_pos_lr_0.001_epoch_7_batch_size_128_filter_num_40_window_size_5_steps_to_eval_30000_optim_AdamW_hidden_dim_200	95.84	95.84	6	Sun May 9, 04:32:21	26m 59s

NER:

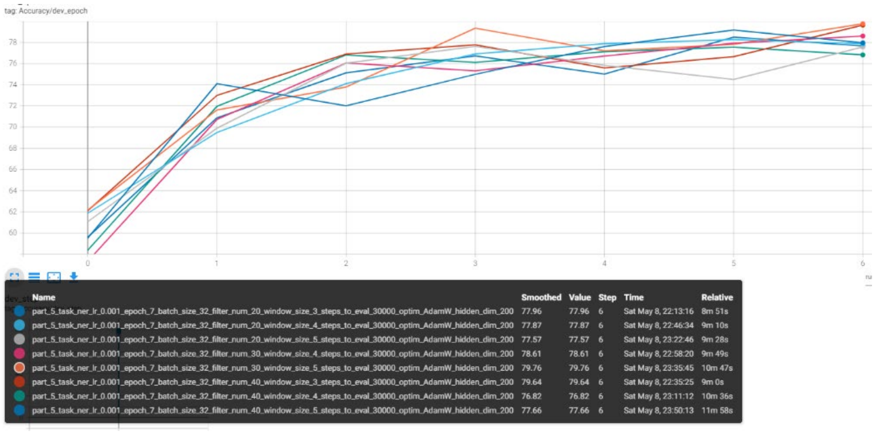
Window Effect:

Differently from the POS in the NER we noticed the opposite trend, when bigger window size yielded better results. 79.76 for window size 5, 78.61 for window size 4 and 76 for windows 3. We thought that maybe in NER all the characters in the word has a lot of meaning not only small patterns, and this may explain it.

Filter Effect:

Same as the window in the NER more == better when the best result was 79.64 for 40 filters but in the other hand 20 was better than 30 (77.9 vs 76). So it was hard for us to reach a conclusion.

In summary bigger windows were better for NER and smaller were better for POS without a clear effect of the filters on both of the tasks.



	Name	Smoothed	Value	Step	Time	Relative
●	part_5_task_ner_lr_0.001_epoch_7_batch_size_32_filter_num_20_window_size_3_steps_to_eval_30000_optim_AdamW_hidden_dim_200	77.96	77.96	6	Sat May 8, 22:13:16	8m 51s
●	part_5_task_ner_lr_0.001_epoch_7_batch_size_32_filter_num_20_window_size_4_steps_to_eval_30000_optim_AdamW_hidden_dim_200	77.87	77.87	6	Sat May 8, 22:46:34	9m 10s
●	part_5_task_ner_lr_0.001_epoch_7_batch_size_32_filter_num_20_window_size_5_steps_to_eval_30000_optim_AdamW_hidden_dim_200	77.57	77.57	6	Sat May 8, 23:22:46	9m 28s
●	part_5_task_ner_lr_0.001_epoch_7_batch_size_32_filter_num_30_window_size_4_steps_to_eval_30000_optim_AdamW_hidden_dim_200	78.61	78.61	6	Sat May 8, 22:58:20	9m 49s
●	part_5_task_ner_lr_0.001_epoch_7_batch_size_32_filter_num_30_window_size_5_steps_to_eval_30000_optim_AdamW_hidden_dim_200	79.76	79.76	6	Sat May 8, 23:35:45	10m 47s
●	part_5_task_ner_lr_0.001_epoch_7_batch_size_32_filter_num_40_window_size_3_steps_to_eval_30000_optim_AdamW_hidden_dim_200	79.64	79.64	6	Sat May 8, 22:35:25	9m 0s
●	part_5_task_ner_lr_0.001_epoch_7_batch_size_32_filter_num_40_window_size_4_steps_to_eval_30000_optim_AdamW_hidden_dim_200	76.82	76.82	6	Sat May 8, 23:11:12	10m 36s
●	part_5_task_ner_lr_0.001_epoch_7_batch_size_32_filter_num_40_window_size_5_steps_to_eval_30000_optim_AdamW_hidden_dim_200	77.66	77.66	6	Sat May 8, 23:50:13	11m 58s

Learned Filters

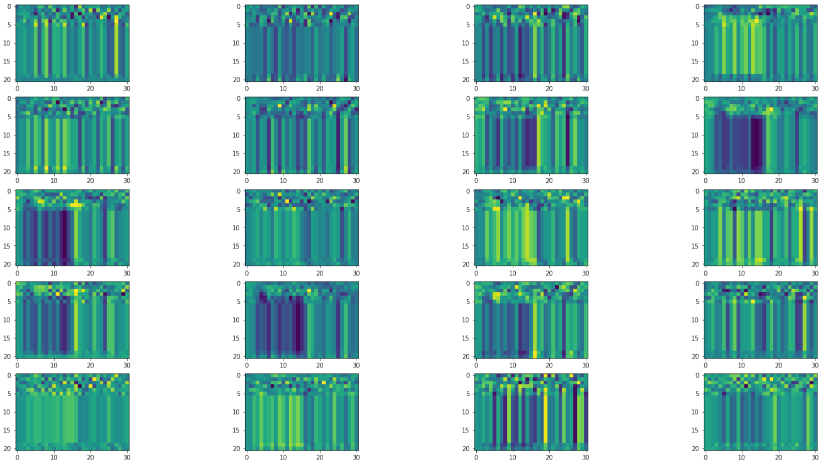
In order to analyze the filters we extracted the convolution activation maps of both the NER model and the POS model and applied them on different words (we of course compared graphs from the same word on different model) and we tried to identify a pattern in the image plot that is in fact a heat map created by the filters. The idea is that when a cell is “lighted” it means that the filter activate this cell, therefor the lighted area is the one that the filter learned to look at. For example if we see a filter that light only the first row, It means the filter is looking only at the first character in the word (at list in our set up the pad the end of the word up to 20 characters).

We than tried to identify “a pattern in the patterns” (to see if we can cluster different themes in the patterns of each model) and we tried to understand why these patterns specifically were learned by the model and how it can benefit from these specific filters.

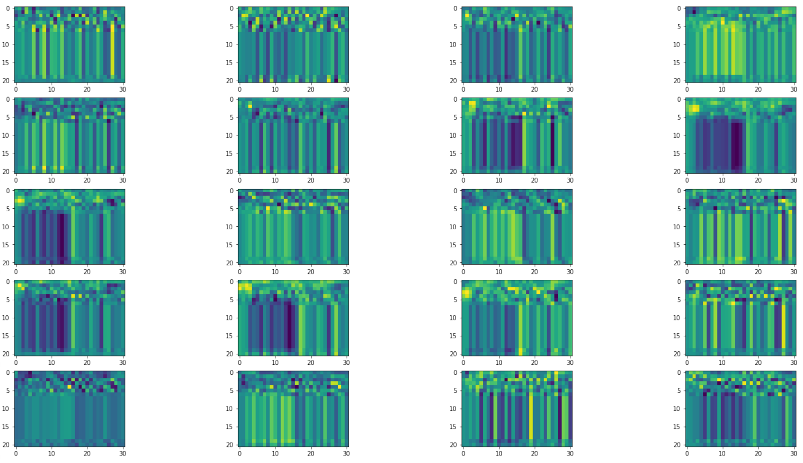
We pasted here four heat maps of the filters. Two of the NER for the words “head” and “Essex” and two for the POS model. The heatmap yielded different filter results, with different features that we will describe in the next paragraphs.

POS filters:

“head” -POS



“Essex-POS”

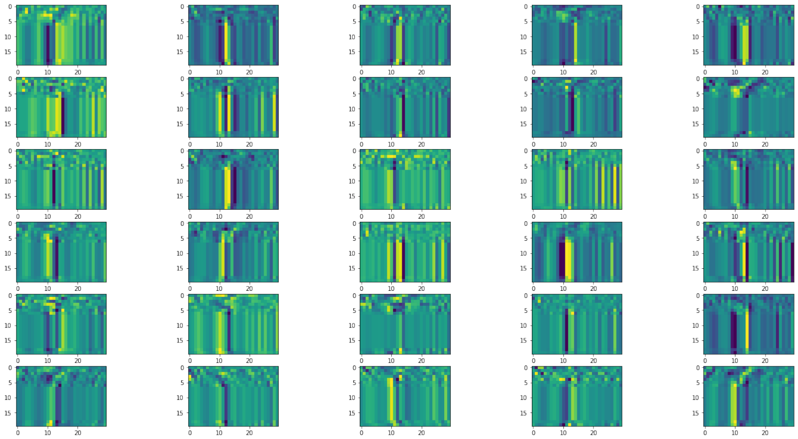


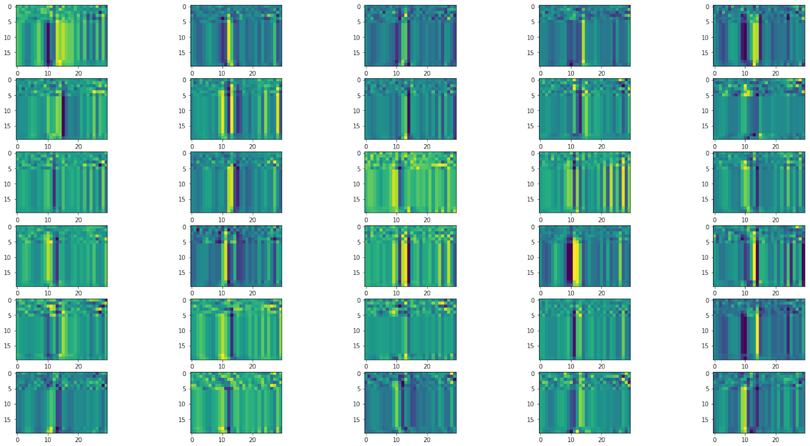
The POS filters, had much more filters that look at each character at a time (5,0) and (5,1) – in a sense they light the first 5 lines which correspond to the first 5 characters, we believe it is because first of all many of the words are about this size and second of all there interesting signal in this letters. Also, we saw some filters that look on half of the characters vector (splitting the vector in the middle) and we guess it means they   look at all the characters as whole.

NER filters:

Essex -Ner

head-Ner





In the other hand we saw that the NER filters are mode vertical. It means they look more at the word as a whole and less at a character at a time. We didn't manage to find a clear indication on focus on a specific word, although is could be a nice story to tell (:

s

THANKS Ya'll

MY god it was long!