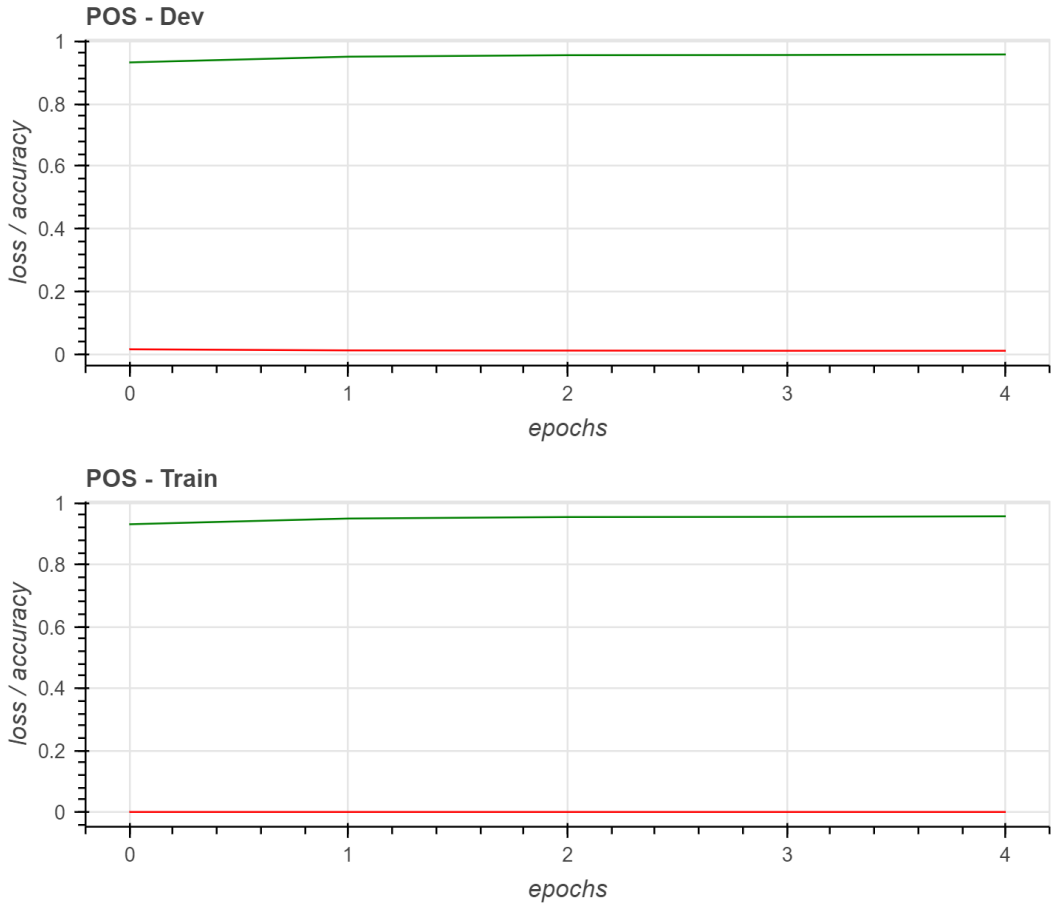
Dafna Magid 315635870 Oved Nagar 302824875

The parameters of the best model for each of the tasks (NER and POS):

	Model POS	**Model NER**
Optimizer	torch.optim.RMSprop	torch.optim.RMSprop
Batch size	512	512
Hidden dim	165	165
Learning rate	0.005	0.005
Number of epochs	10	10
Last epoch accuracy dev	no pre-trained- 96% pre-trained- 95%	no pre-trained – 98% pre-trained- 98%

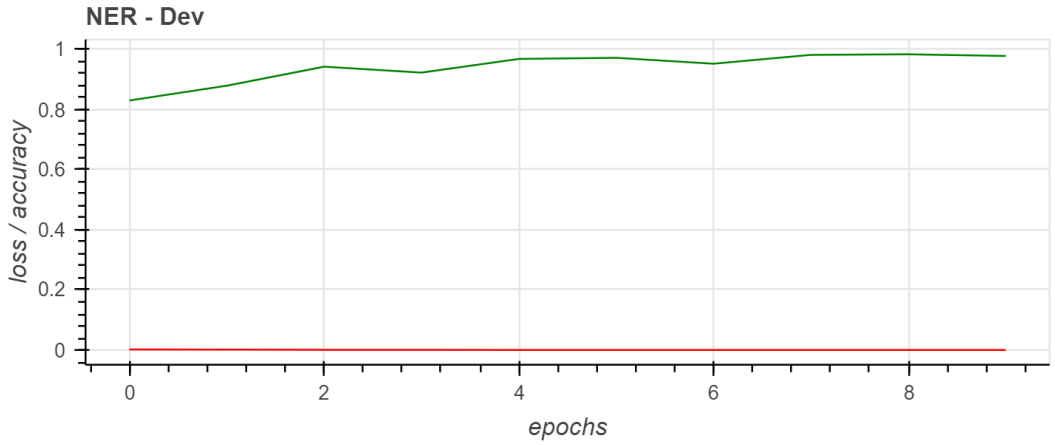
The graphs showing the **accuracy** of the dev set and the **loss** on the dev set as a function of the number of iterations:

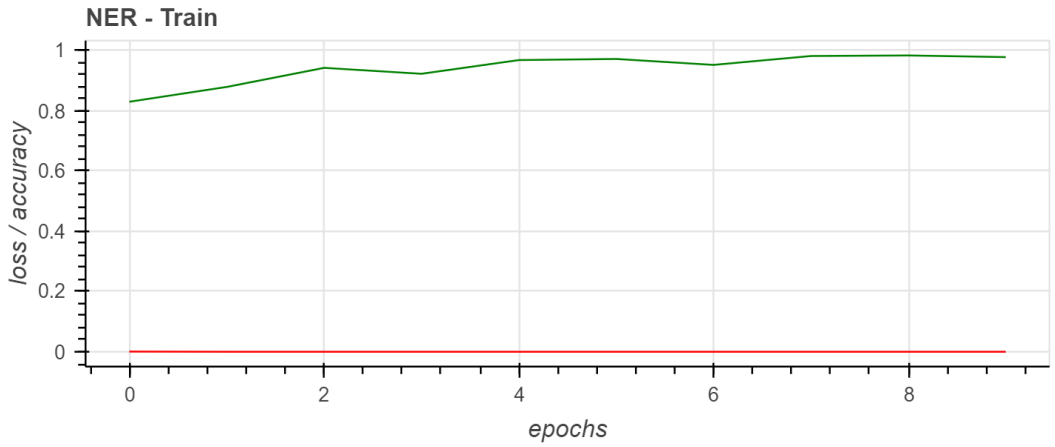
pre-trained POS-



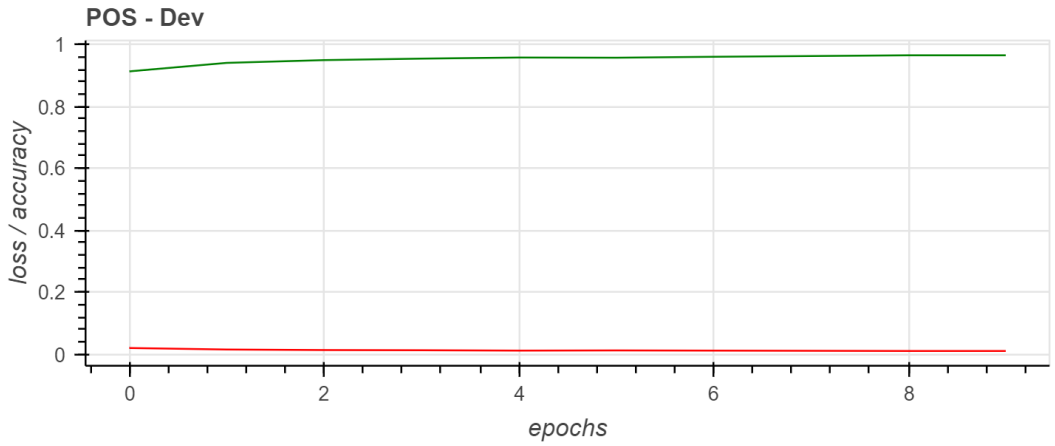
Dafna Magid 315635870 Oved Nagar 302824875

pre-trained NER-

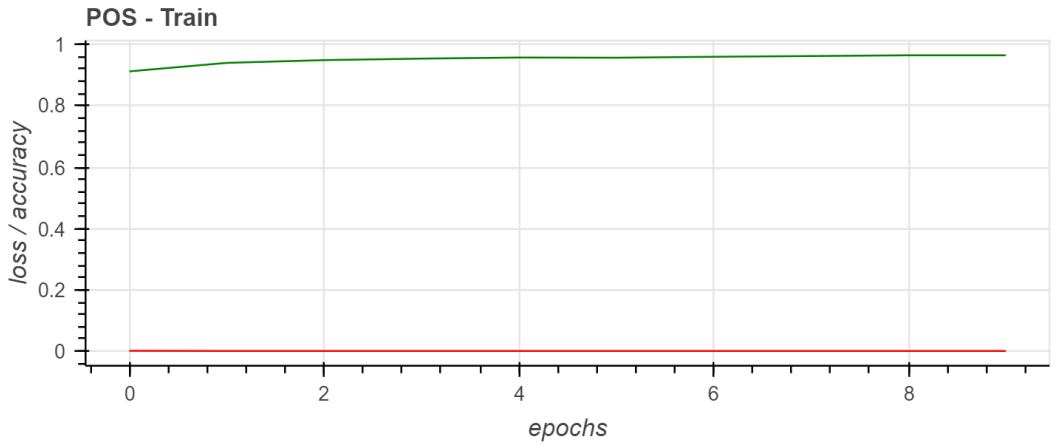




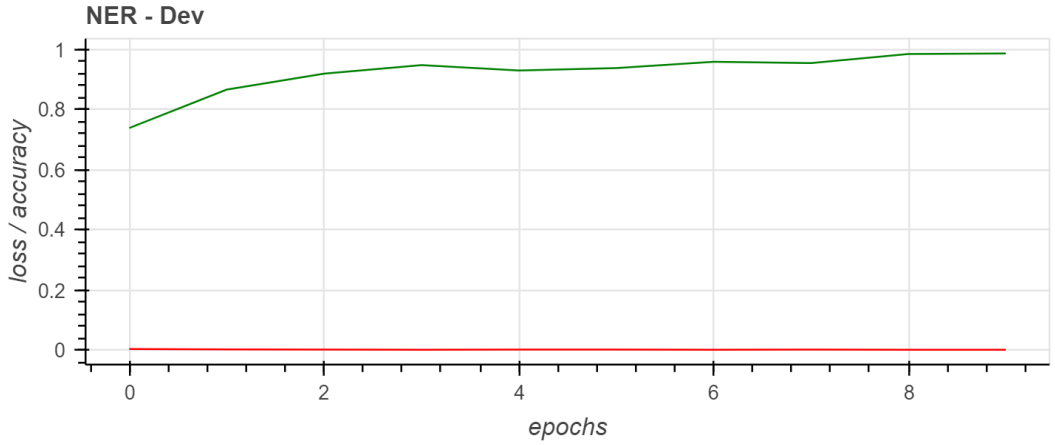
no pre-trained POS-

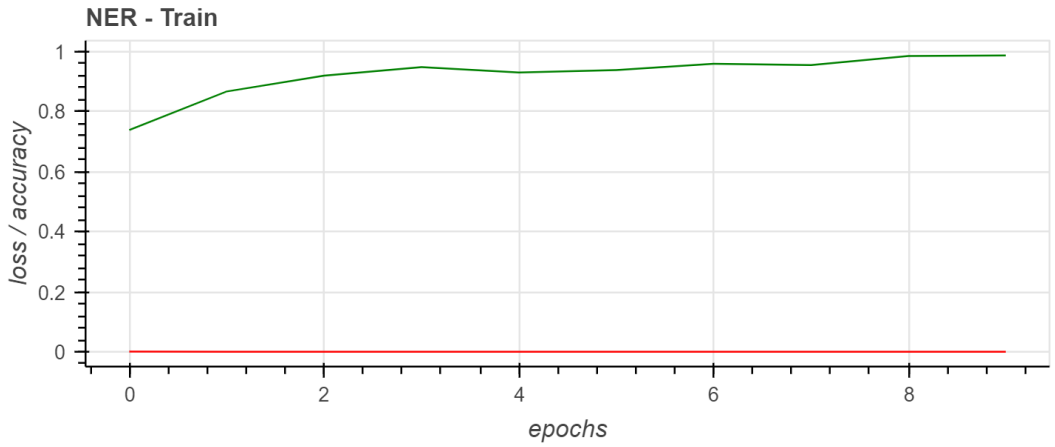


Dafna Magid 315635870 Oved Nagar 302824875



no pre-trained NER-





Dafna Magid 315635870 Oved Nagar 302824875

Brief analysis of the results:

Differences between the NER and POS tags:

In the POS tagging task, the labeling of a word depends in the context it appears in. For example, in the sentence: "the fruit flies like a banana" the word 'flies' can be a verb or a noun.
In the NER tagging task, the labeling of a word depends mostly on the word itself.

which of the pre-trained embedding and the subword units is more useful, and why?

In part 4, where for the representation of the words we made a combination of the word vector with the prefixes and the suffixes vectors, we got the best results.
This is because we have added significant information in this supplement, many words from the same family and type has a similar structure, so this is a great feature to take into account.
We have been able to better characterize many words that did not appear in Train set because we took into account their prefixes and the suffixes and not just a word vector belonging to a known word that has maximum imagination to it.

Are their contributions complementary?

This supplement complements the model because the model succeeded to represents more words in a better way and get higher accuracy percentage.

Are the trends consistent across the different tagging tasks? why?

Yes, but they are more significant in the NER model. This might be due to the fact that the NER model Relies more on the structure of a word than on the structure of the sentence, unlike the POS model. In the POS model, where the same word can have a different tags based on the context of her appearance.

Things we tried that didn't worked out:

- We tried representing words that didn't appear in the train set with embedded word vectors chosen according to the words surrounding that word. In the POS model it worsened the results because of the reasons we already discussed before.
- We tried using deeper networks.
- We tried using Adam, ADAgrad, SGD optimizers.
- In part 4: replace sum with concat.