Universite Paris-Saclay

Text-Mining

Named Entity Recognition

Named Entity Recognition on QUAERO corpus.

Pawel Budzynski

September 10, 2022

1 Introduction.

During this exercise implementation of Bi-directional LSTM-CNNs-CRF model for NER from ***NeuroNLP2*** (https://github.com/XuezheMax/NeuroNLP2) was used to perform Entity Recognition (NER). A number of neural networks was trained for NER task using QUAERO_FrenchPress corpus. Pre-trained word embeddings were used during the training. The experiments include training the network with different sets of hyperparameters to compare overall performance as well as more detailed results given a type of an entity. For experiments execution Google Colab was used to accelerate the computations thanks to the access to a GPU device. The network used for most of the experiments have one layer with 128 units. The experiments performed and their outcomes are presented in the following section.

2 Experiments and results.

2.1 Impact of the size of the training set

In the first experiments networks were trained using varying number of tokens for 10 and 30 epochs. The results are presented on Figures 1 and 2. It is clearly visible that the network is learning faster and obtain much better performance when more data is used for the training. For example, while usign $1M$ tokens the network gives similar performance as a network trained for 30 epochs when only $500k$ tokens were used fot the training. Furthermore, using a big number of tokens (≥ 500) allows to achieve acceptable scores already during the early epochs.

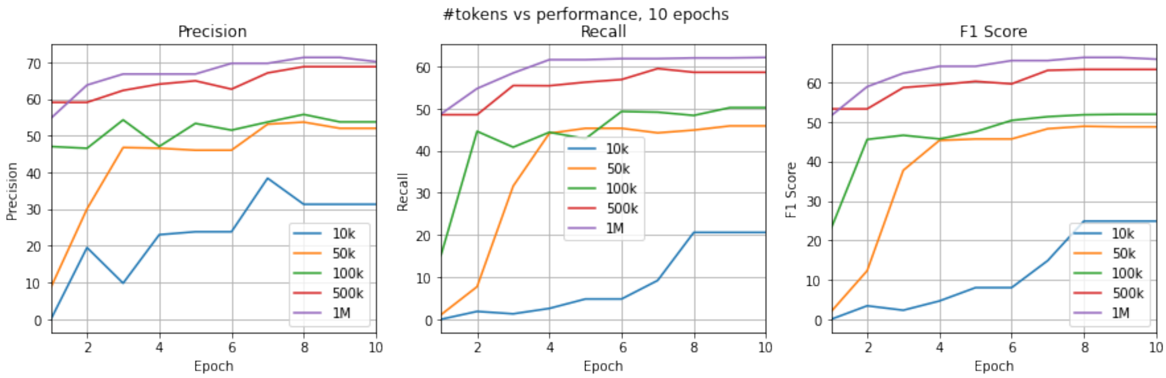

Fig. 1: Impact of the number of tokens on the performance. Network trained for 10 epochs.

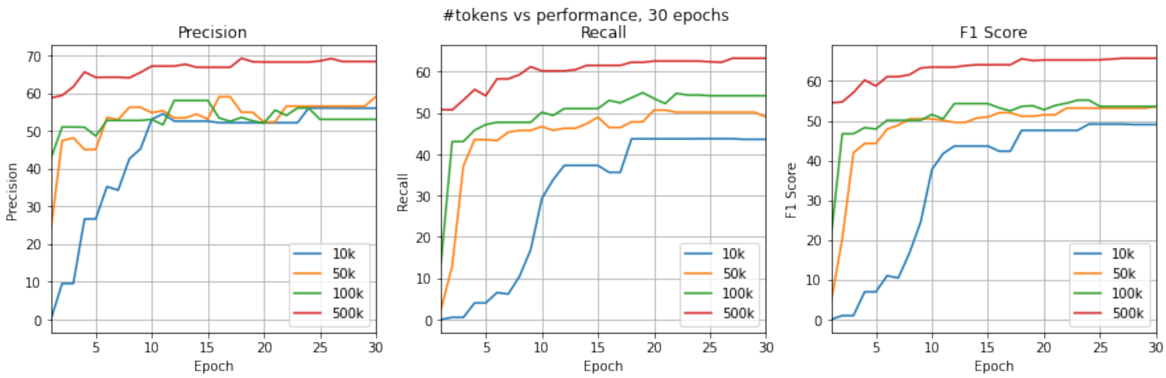

Fig. 2: Impact of the number of tokens on the performance. Network trained for 30 epochs.

2.2 Impact of the number of epochs

The impact of the number of epochs on the performanece was examined during the second experiment. On the Figure 3 the results are presented when the networks were trained for varying number of epochs on a training set of $10k$ tokens. Figure 4 shows the result when $100k$ tokes were used for trainig. There is no surprise that the networks trained for the greater number of epochs achieve better scores. However, the plots below show that the variance of the results is much less when a greater number of tokens was used for training. Precisely, when $100k$ tokens were used, a network manages to only slightly improve its performance between epochs 10 and 40.

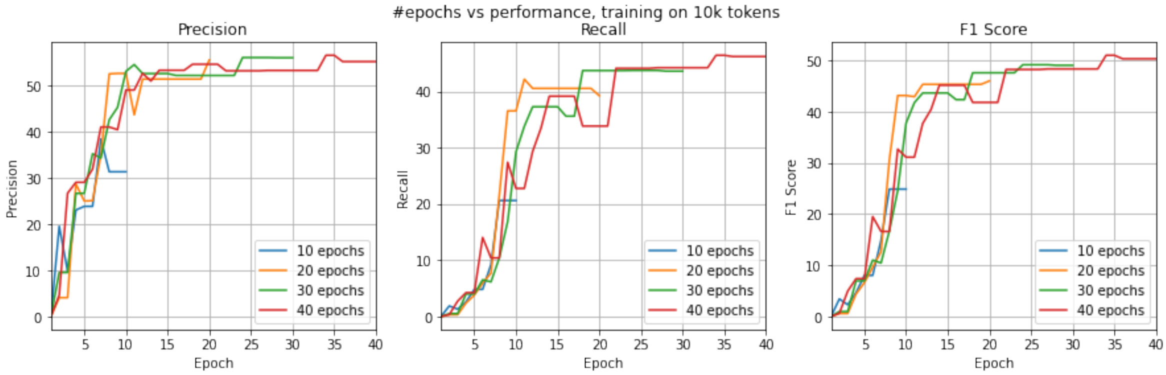

Fig. 3: Impact of the number of epochs on the performance. Network trained on 10k tokens.

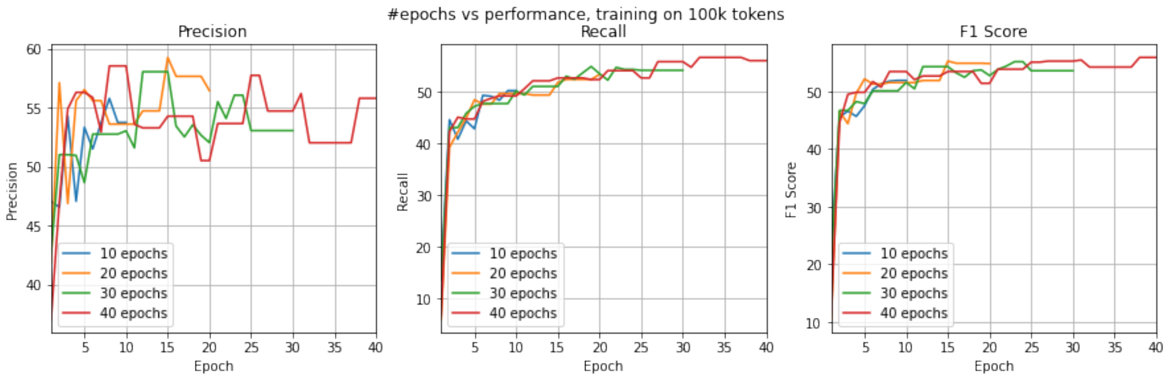

Fig. 4: Impact of the number of epochs on the performance. Network trained on 100k tokens.

2.3 Performance within entity types

The performance of the models for each type of entities were presented on the Figures 5 and 6. The results show similar trends in both cases. With increasing number of tokens or inclreasing number of epochs the performance is improving. However, it is visible that the scores obtained are much worse for particular types of entities. While performance on *person* and *location* types is comparable to the general scores obtained by the network the other categories achieved much worse scores. That seems to be directly correlated to the corpus specifics. The corpus contains majority of entities referring to persons while the other categories have much less examples.

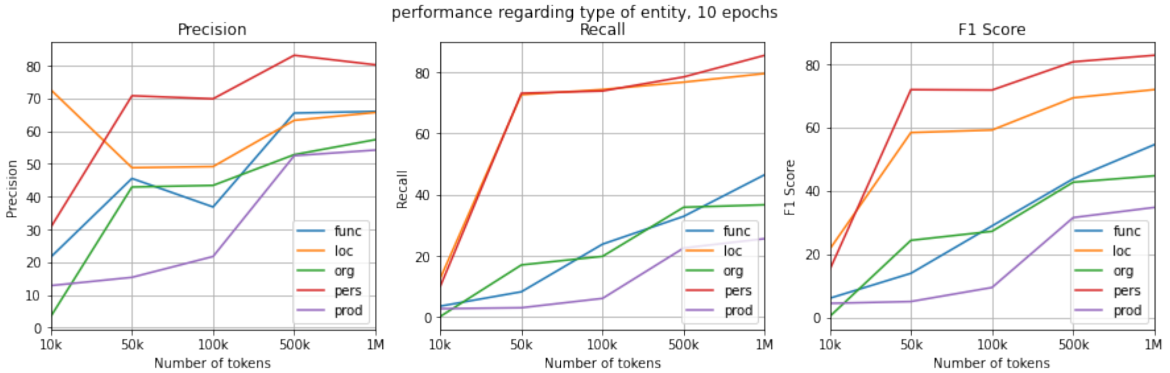

Fig. 5: Impact of the number of tokens on the performance given the entity type. Network trained for 10 epochs.

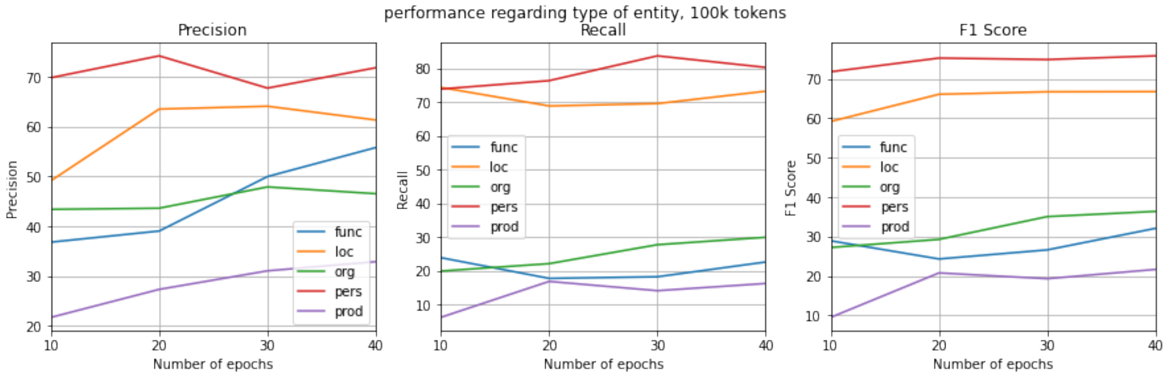

Fig. 6: Impact of the number of epochs on the performance given the entity type. Network trained on 100k tokens.

2.4 Automated annotation results

The detailed results of automated annotations on the dev set are presented in the Table 1. It is clearly visible that the networks performs way better on marking starting tokens (*"b"*) of NERs while inner tokens remains virtually unrecognised.

type	total	10 / 10k	10 / 100k	40 / 100k
b-func	617	58	213	179
i-func	599	30	245	256
b-loc	709	94	553	540
i-loc	324	0	51	45
b-org	503	1	124	168
i-org	370	2	20	88
b-pers	1381	332	1029	1092
i-pers	939	59	772	803
b-prod	329	12	22	79
i-prod	525	12	40	47

Table 1: Automated annotation results on the dev set for various settings.

2.5 Improved network architecture

As a last experiment a performance of bigger network was examined. The results are presented on the Figure 7. In this try a 2 layers with 256 neurons each was used. Interestingly, the performance of precision seems to be almost the same as in the case of the previous network which was much smaller (one layer with 128 units). However, the recall and F1 score imrpoved signicivantly (60% vs 70%).

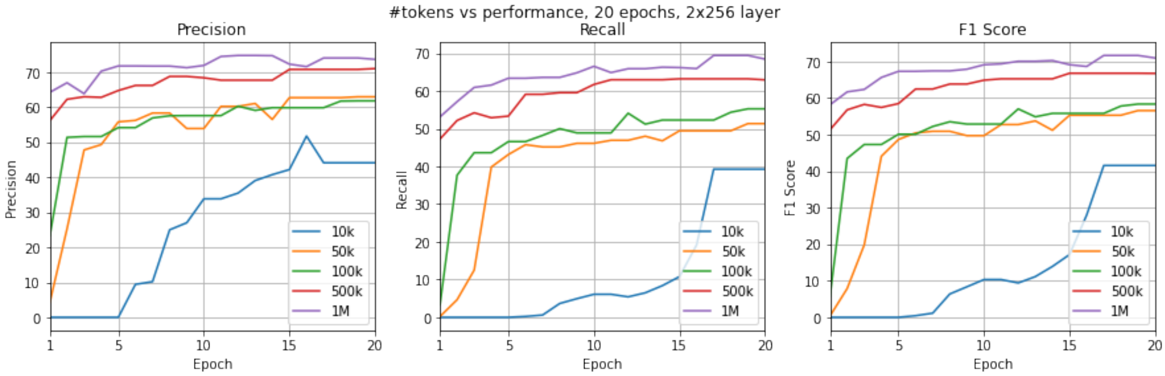

Fig. 7: Performance of the bigger network.

3 Conclusions.

In this exercise a number of network was trained to examine the performance and bavaviour of NER model. The experiments show that the performance of NER model is highly dependent on the data used for training. The performance improves when a bigger number of tokes is used for training. Furthermore, models success to recognise entity types that have more representatives in the dataset. Further experiments with network architecture and hyperparameters may lead to refinement of the model however the biggest impact is associated to the quality of the data used for training.