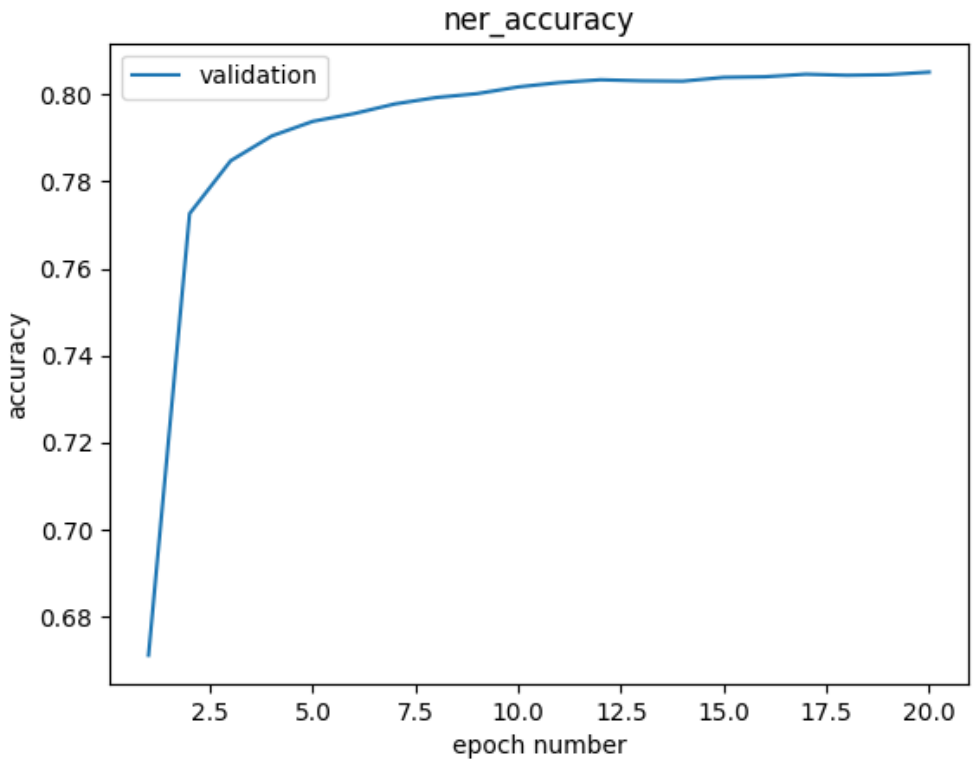
Assignment 2
Part 3
Course: Deep learning for texts and sequences.
Students: Lev Levin, id: 342480456
Lior Shimon, id: 341348498

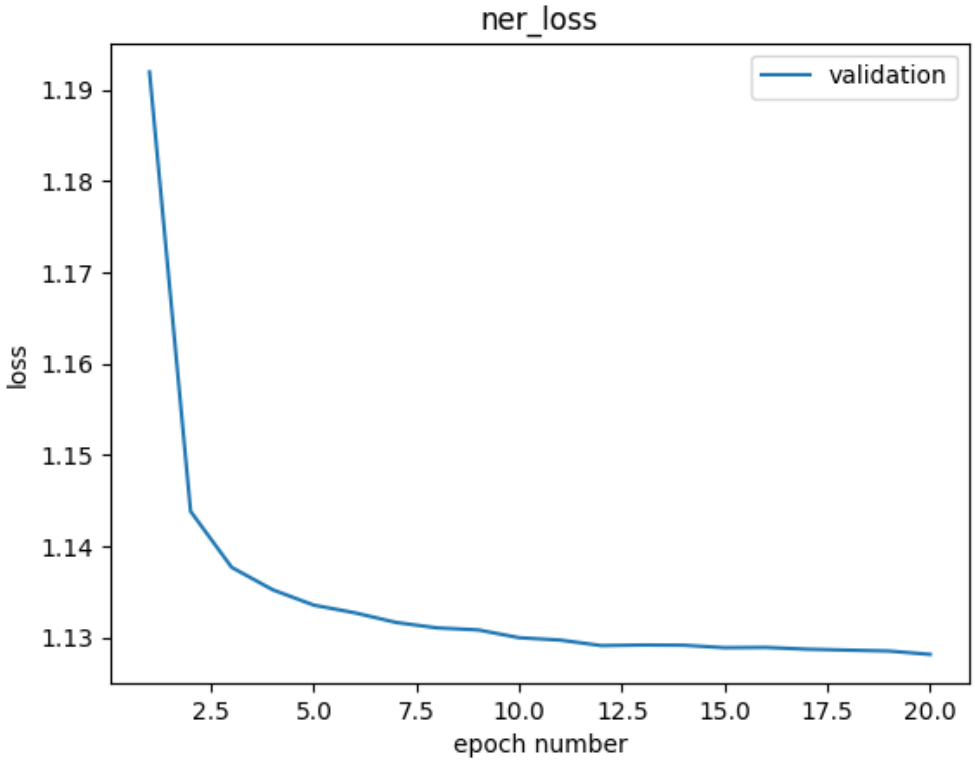
Parameters for ner model:

Description:

```
epochs = 20
hidden_size = 10
l_r = 0.01
torch.manual_seed(1)
batch_size = 1000
Optimizer: Adam
Initialization of embedding: uniform distribution(-1,1)
```

Results:



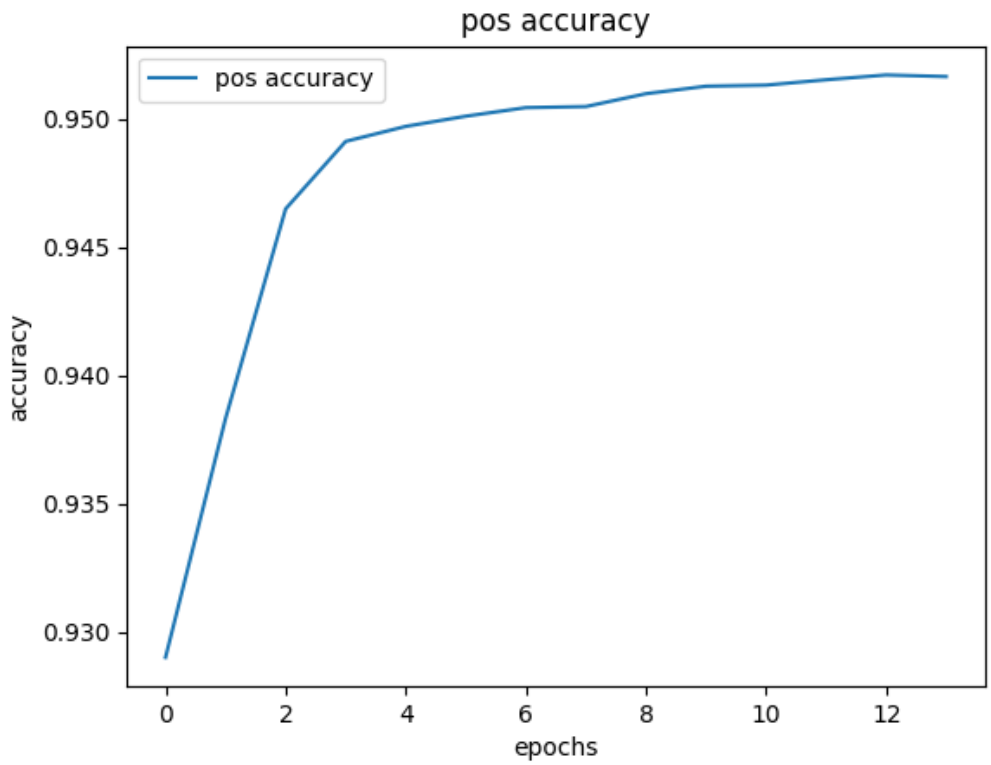


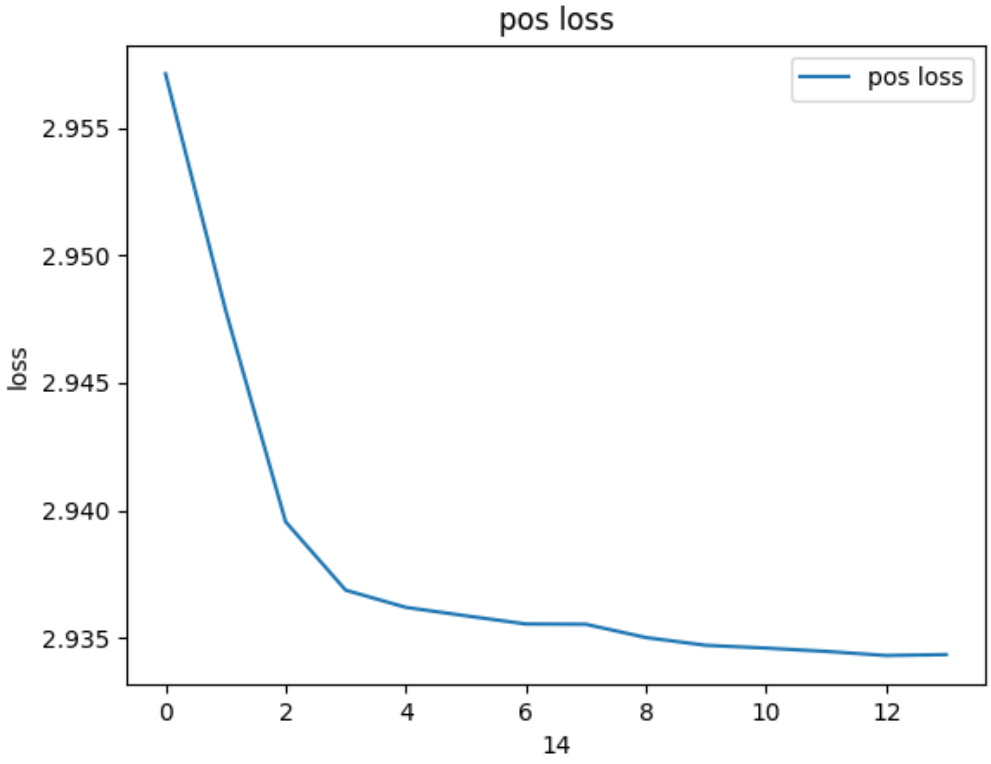

```
Epoch [20/20], Train_Loss: 1.0549, Train_Accuracy: 97.01%
Epoch [20/20] ,Dev_Loss:1.1282,  Dev_Accuracy:80.51%
```

Parameters for pos model:

```
epochs = 14
hidden_size = 60
l_r = 0.01
torch.manual_seed(1)
batch_size = 1000
Optimizer: Adam
Initialization of embedding: uniform distribution(-1,1)
```

Results :





```
Epoch [14/14], Train_Loss: 2.9059, Train_Accuracy: 98.00%
Epoch [14/14], Dev_Loss:2.9343,  Dev_Accuracy:95.16%
```

Description of the logic of using the pre-trained vectors:

Lower casing of pre-trained words:

We lower-cased all words in train set, dev set, test set. It guarantees the vocabulary doesn't contain the same words few times. In this way we avoid of creating unnecessary noise.

Words that doesn't appear in pre-trained vocab

We create a vocab which contains words from pre trained embedding and also words from train set. Also, we create a common embedding matrix, where vectors for pre trained words are corresponding vectors from pre trained embedding. And for words from train set that doesn't appear in pre vocab we initialize random vectors. Regarding dev and test sets, while reading sentences from a file, if we get a word that doesn't appear in our main vocabulary, we change the provided word to word '<Miss>' which means – any word that doesn't appear in the vocabulary. We add the '<Miss>' while constructing the vocabulary and initialize randomly a vector for it in embedding matrix. Thus, each word that appears in dev/test set and doesn't appear in train set or pre trained vocab, we map to the vector '<Miss>'.