Omer Zucker   200876548
Omer Wolf   307965988

We will answer the following questions:

how large were the training and test sets, did your network succeed in distinguishing the two languages (it should)? how long did it take (both wall-clock time (i.e., number of seconds), and number of iterations)? did it succeed only on the train and not on the test? what you did in order to make it work, etc.

Following are the global variables used by our model:

experiment.py
```
EMBEDDINGS_DIM = 180
LAYERS = 1
HIDDEN_DIM = 50
HIDDEN_MLP_DIM = 50
EPOCHS = 1    # number of iterations
BATCH = 360   # number of sentences
```

gen_examples.py
```
BATCH = 500
DATA_SIZE = 5000
```

At first we have created our examples by artificial language for both pos/ner using discrete number of characters and digits so we got our train and dev data sets. After some tests we chose our size of both train and dev to be 5000. Than we thought taking small batch might be good for this size of data but our assumption was wring until we were understanding we should take at least 10% of batch size (from our entire data set) so we set our batch to be 500. We were also using hidden dim of 50 and 180 for embeddings dim.

After our data sets were ready we start implementing our neural network using the Long-Short-Term-Memory (LSTM) followed by MLP with one hidden layer. By testing it we found that 1 epoch and 360 batch (iterations) is quite enough to get result of 100% accuracy. We run the model several times for both train and test data sets in order to check best running time. Our final result was around 20 seconds.