Yehoshua Stern 314963927
Eyal Cohen 207947086

Part 3 (tagger 2):

Parameters: (POS)

Learning rate	Epochs	Batch size	Batches per epoch	Middle dimension
0.5 divided by 2 every 5 epochs	25	256	4096	`(DIM * 5 + out_dim) / 2` = 138

Parameters: (NER)

Learning rate	Epochs	Batch size	Batches per epoch	Middle dimension
0.5 divided by 2 every 5 epochs	25	256	4096	`(DIM * 5 + out_dim) / 2` = 138

The training process was with mini batches, every batch was taken randomly from the training data, we did not make a full pass for every epoch, we just took every epoch 4096 random batches of 256 samples.

Considerations:

Q: Did the accuracy improve over the tagger without the pre-trained embeddings? By how much?
A: not by too much, POS tagger improved by 0.1% accuracy and the NER tagger improved by 1%.

Q: How do you handle lower casing?
A: we used the lower-cased vectors to add new vectors for each of the words we have in the training data.
For example: if we had a pre-trained vector for the word "in", if we saw the word "In" we copied the vector for the lower-cased word and gave it to the new upper-cased word.

Q: What do you do with words that are not in the pre-trained vocabulary

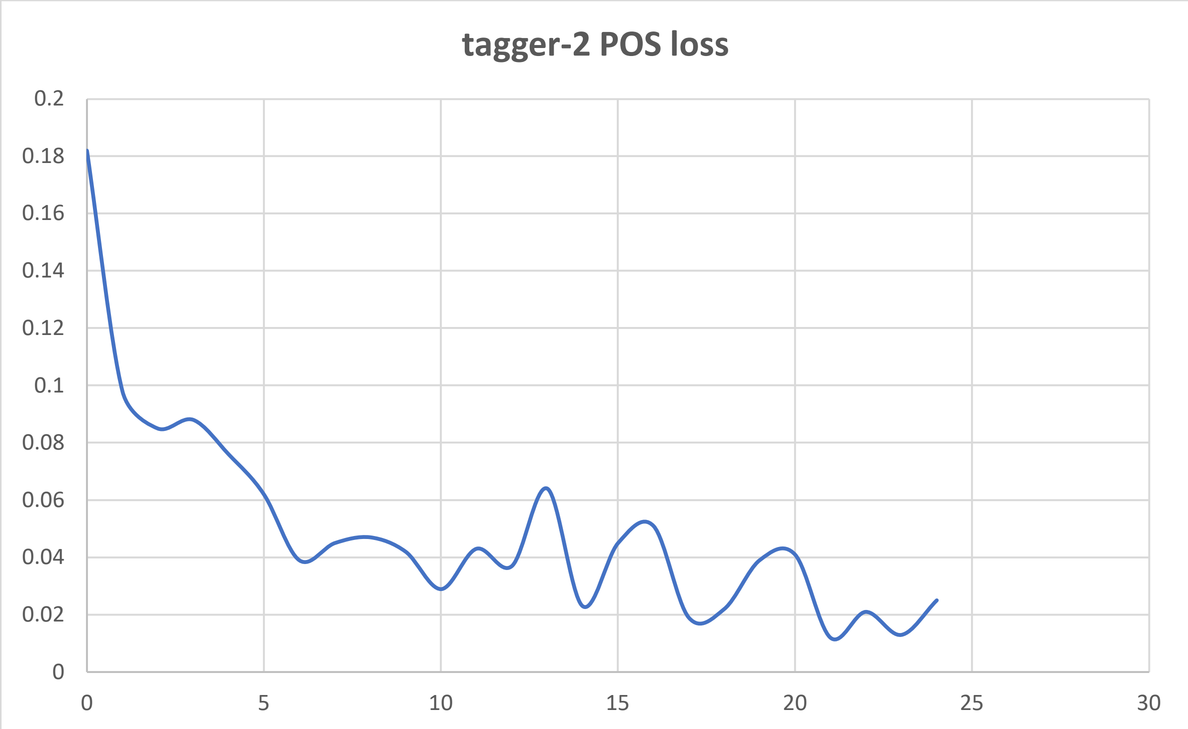
A: before starting the training process we updated our vocabulary by adding the words that do not appear in the vocabulary and added random (or copy of the lower-cased) vector for each of the words we added to the word vectors matrix we already have.

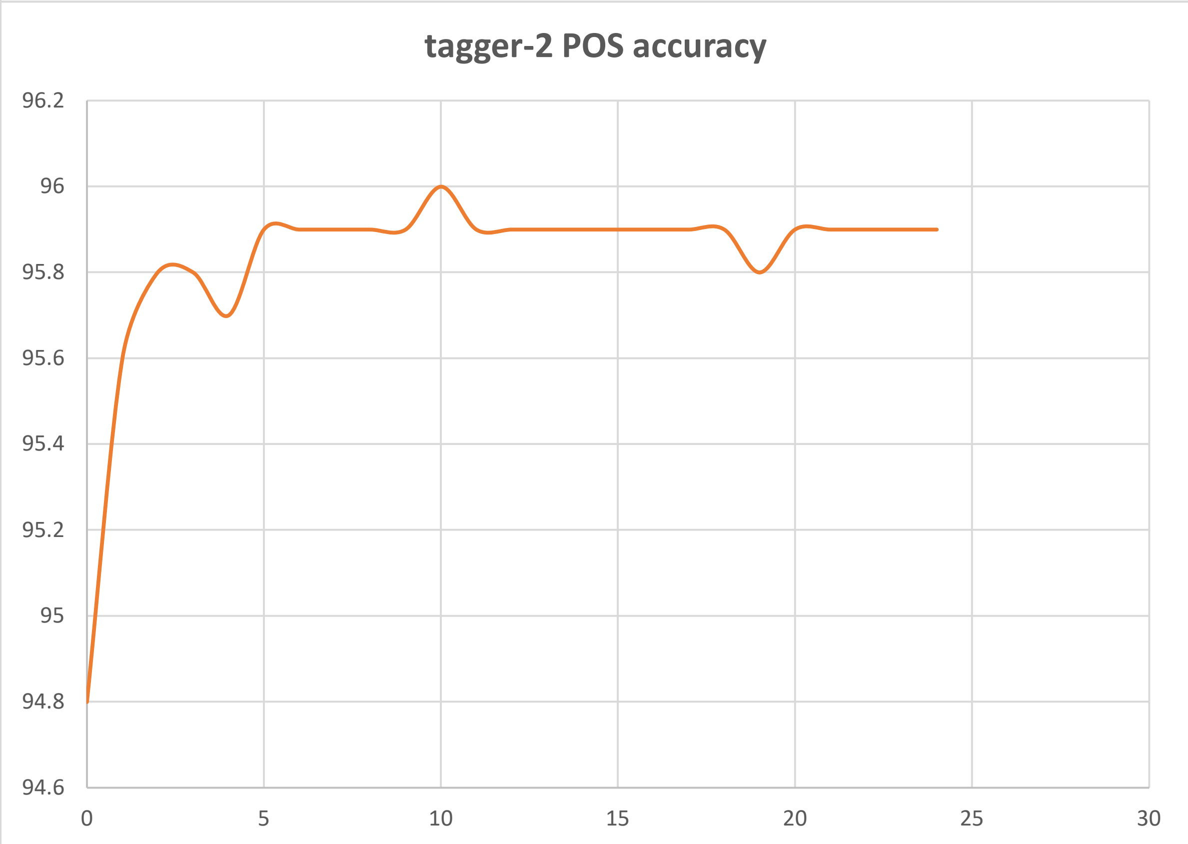
We also saw that the pretrained vectors have already special UNK, START, END symbols so we used them.

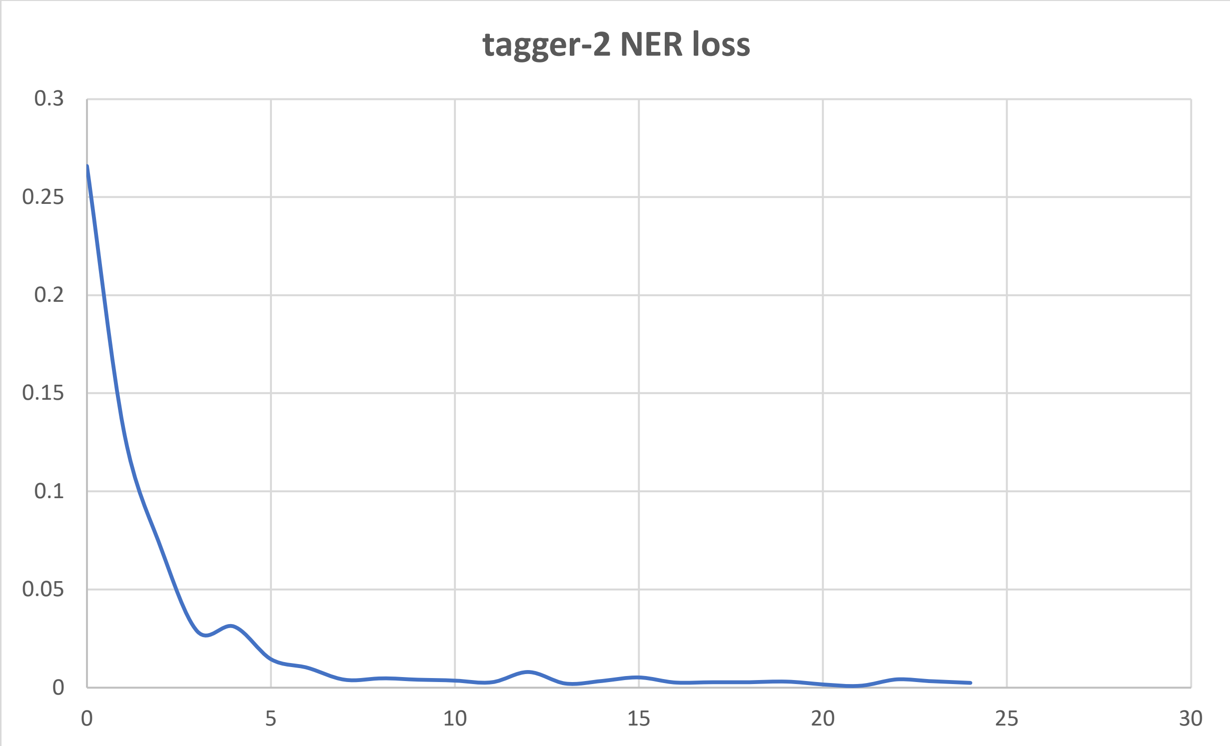
Another thing we noticed is that the given vocabulary has no numbers, instead each digit in number was switched to DG (100 turned to DGDGDG), so we did this transformation to the training data too. (which added 2% to the accuracy on the dev set)

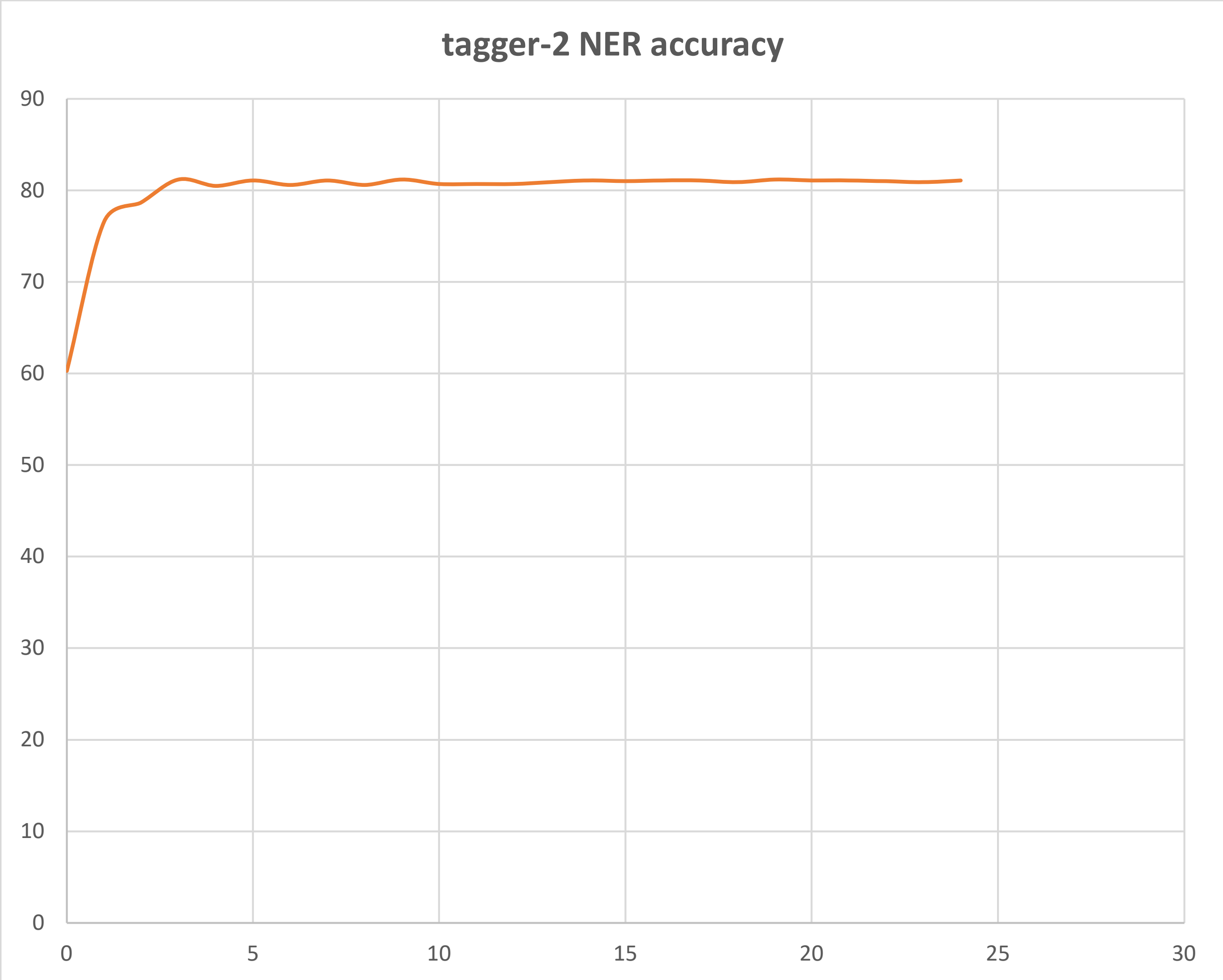
One more thing we did is to divide the vectors by 30, we noticed the model converging much faster when we do it.
We did it to match the standard deviation of the vectors we got in tagger 1.

Graphs: (x is epochs, y is value)






tagger-2 NER loss
0.3
0.25
0.2
0.15
0.1
0.05
0
0
5
10
15
20
25
30


tagger-2 NER accuracy
90
80
70
60
50
40
30
20
10
0
0
5
10
15
20
25
30