Jonathan Mandl 211399175

Danielle Hodaya Shrem 208150433

Part 4

In this section, our model is a simple feed-forward neural network with a single hidden layer of 250 neurons.

We used the following hyperparameter configuration for the NER task:

- Learning rate: 0.001
- Epochs: 10
- Batch size: 64

For the POS task, we used the following hyperparameter configuration:

- Learning rate: 0.001
- Epochs: 5
- Batch size: 64

We lower-cased all the words in our vocabulary before extracting prefixes and suffixes for each word to ensure we can use the lower-case vocabulary of the pre-trained embeddings. All three embedding matrices—words, prefixes, and suffixes—were initialized randomly; if pretrained vectors were provided, they replace the corresponding rows in the word, prefix, or suffix matrix.

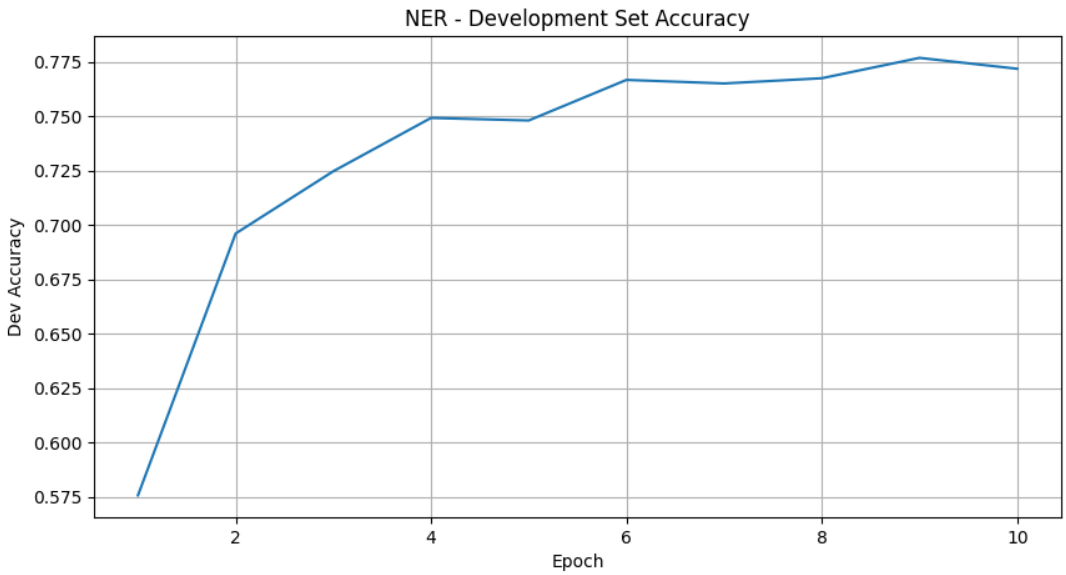
To handle prefixes, suffixes and words that appear in the development set but not in training, we added a special <UNK> token to our embedding matrices of prefixes, suffixes and words. During training, we randomly masked 15% of prefix/suffix/word tokens—replacing them with <UNK> token—so that the model learns a useful representation for unknown tokens.

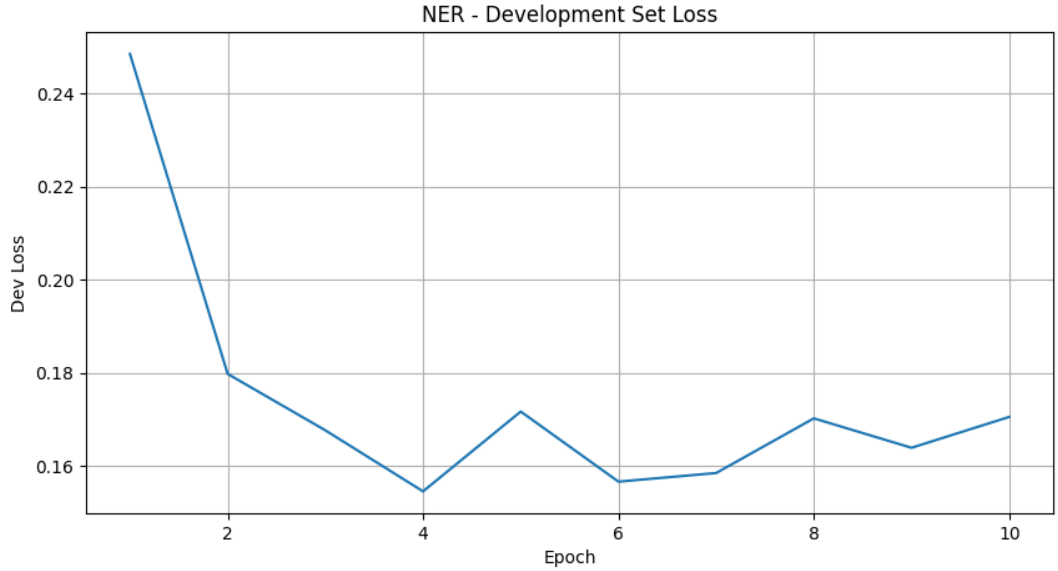
Combining pre-trained embeddings with subword units showed consistent improvements across both tasks. The improvement in NER accuracy from 0.77 to 0.80 was more pronounced than the increase in POS accuracy from 0.95 to 0.96. This suggests that the richer character-level signals provided by prefixes and suffixes are more useful in the NER task, where entity names often share morphological patterns (e.g., -berg, -man). While pre-trained embeddings alone did not significantly improve accuracy, they led to faster convergence in early epochs. The results indicate that the two types of embeddings are complementary and most effective when used together.

Figures

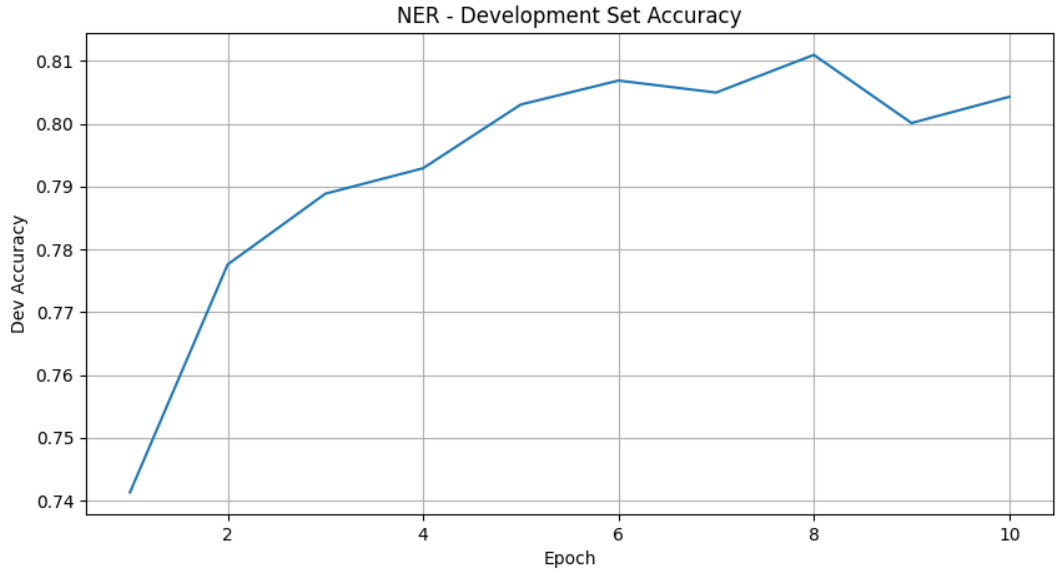
Task – NER

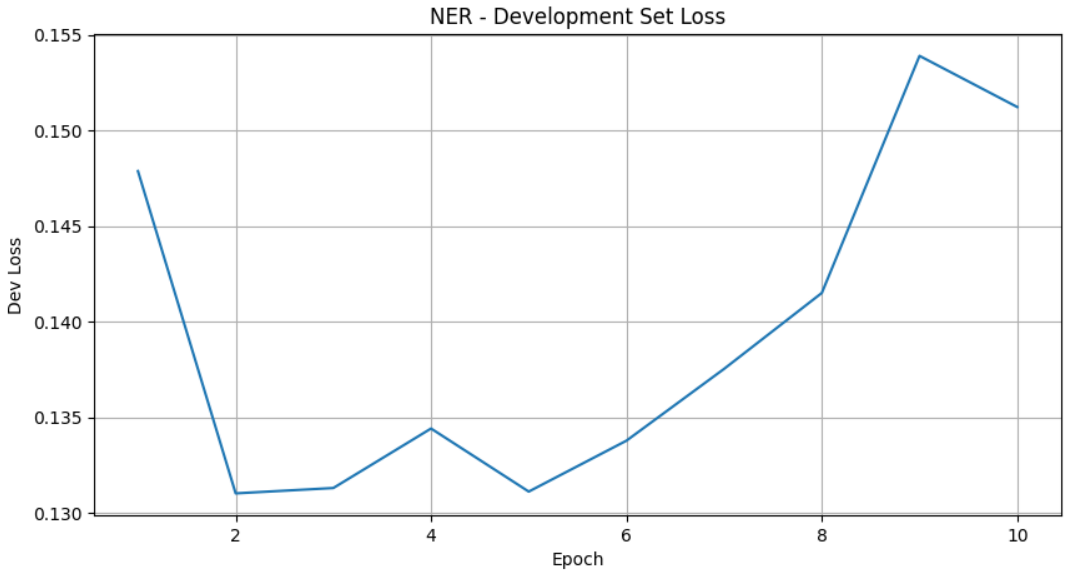
No pre-trained embeddings





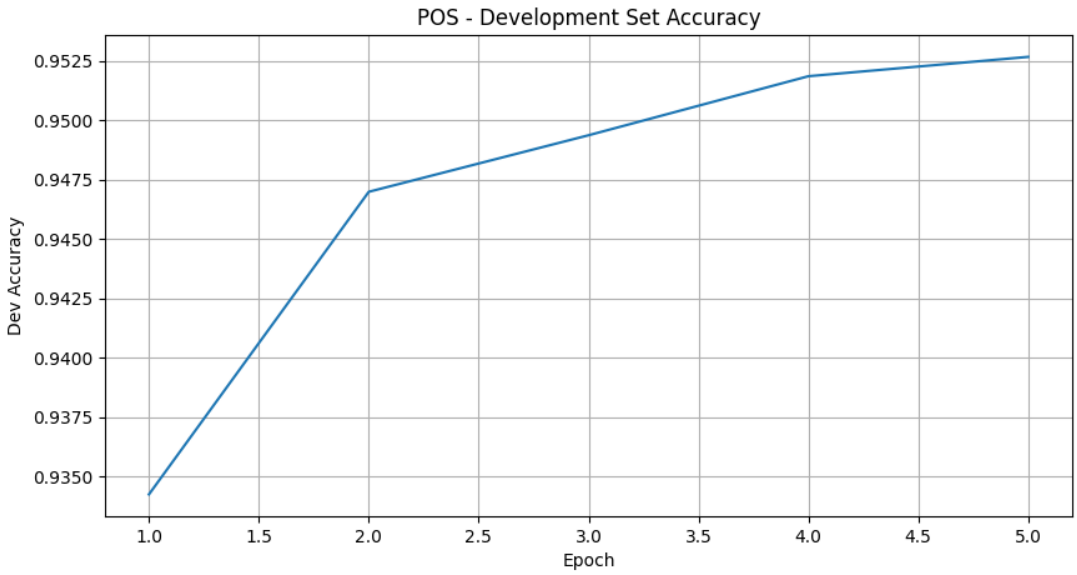
With pre-trained embeddings

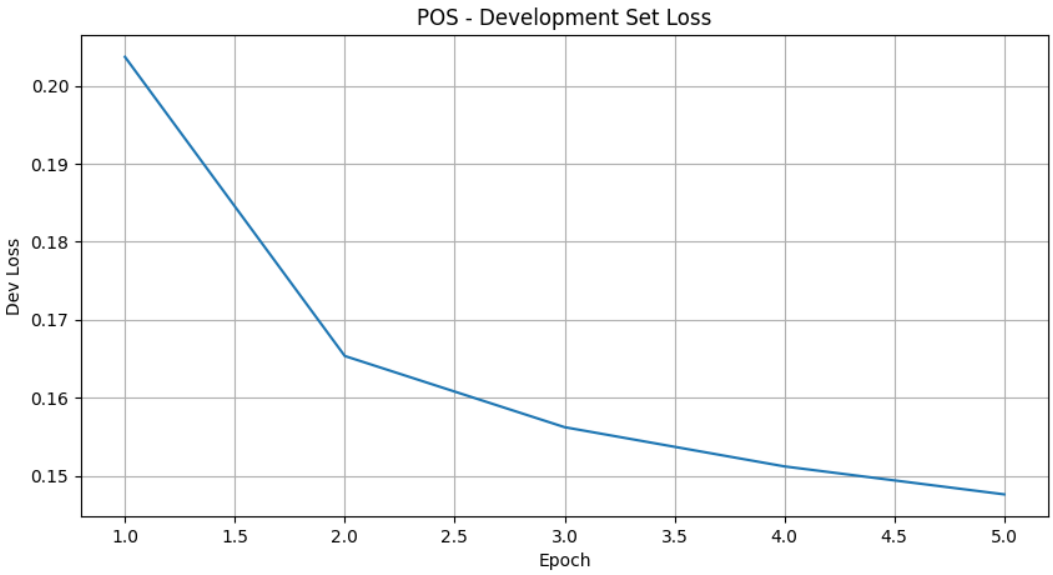




Task – POS

No pre-trained embeddings





With pre-trained embeddings