Yehoshua Stern 314963927
Eyal Cohen 207947086

NLP assignment 1 writeup

1. How we handled unknown words:
We defined various word signatures: words that start with a capital letter, all capital letters, acronyms, numbers, etc. additionally, we have signatures for common prefixes and suffixes.
In training, we count each word (that occurs less than 10 times in the training data) twice - once as itself, and once as a representative of its word signature (or as *UNK* if no signature is applicable.).
While decoding in test time, when we encounter a token that was not in the training text, we substitute the relevant signature for it and use the emission probabilities as calculated for that signature. Additionally, for any word that appears less than 10 times in the training data, the emission probabilities are calculated as a linear interpolation of the specific emission probabilities for that word and the emission probabilities for the relevant word signature.

2. viterbi hmm pruning strategy:
We note for each token (or signature) what tags it was tagged as in the training data, and in the viterbi algorithm we iterate only on the relevant tags for each word.

3.test scores (on dev set):

	MLE-greedy	hmm-viterbi	maxent-greedy	memm-viterbi
POS - accuracy	94.13%	96.16%	96.09%	96.73%
NER - token accuracy	94.88%	96.24%	96.53%	97.76%
NER - span precision	72.57%	83.33%	87.92%	89.13%
NER - span recall	73.64%	78.56%	79.88%	88.67%
NER - span f1 score	73.1%	80.87%	73.71%	88.90%

4.Is there a difference in behavior between the hmm and maxent taggers? Discuss.
Yes. as can be seen from the table, there is a substantial difference in performance between the greedy versions of the respective taggers. The maxent-greedy tagger is better than the MLE-greedy, because it uses features from the words before and after each word, unlike greedy-mle tagger which uses only the current word and last tags.

5.Is there a difference in behavior between the datasets? Discuss.
Yes, in POS tagging there are many hints in the text itself for the correct tagging - there are unique suffixes for verbs in different tenses, for plural nouns etc., and preceding tags are very informative for the current tag (for example, multiple verbs one after the other are not very likely). In NER tagging, the text itself has fewer hints for the correct tagging, therefore outside knowledge is often required to predict the correct tag - when the tagger encounters a new word it is harder to decide which entity it is if any. Even for words that were encountered, it is often impossible to decide which is the relevant tag from the immediate context, for example in the sentence:
"This political restructuring had broad consequences for **Liechtenstein**: the historical imperial, legal, and political institutions had been dissolved. (from https://en.wikipedia.org/wiki/Liechtenstein),
It is unclear whether 'Liechtenstein' refers to the microstate, or to a person with that name.

6. What will you change in the hmm tagger to improve accuracy on the named entities data?
Add word shapes signatures - i.e, each rare token will be assigned a signature based on it's word shape (as described in the Jurafski & Martin chapter 8 - I.M.F -> X.X.X, SQL -> XXX, etc.).

7. What will you change in the memm tagger to improve accuracy on the named entities data, on top of what you already did?
Add features from a lexicon, such as https://github.com/aritter/twitter_nlp/tree/master/data/annotated/wnut16/lexicon :
For each category, create a feature which is positive iff the word is part a span of words that is contained in that category.
Add word shape features, as in 6.

8. Why are span scores lower than accuracy scores?

- Most of the tags in the ner data are 'O' tags, which can be easily predicted by the tagger, and thus elevate the per token accuracy, while they are ignored by the span scores evaluation.
- Span scores evaluation is more stringent, in that it classifies a prediction as true only if all tokens in the span are predicted correctly, while per token accuracy gives partial

scores for partially correct span predictions. For example, if the prediction for the span “People’s/I-LOC Republic/I-LOC Of/I-LOC China/I-LOC” is instead:
“People’s/O Republic/I-LOC Of/I-LOC China/I-LOC”, the per token accuracy for the span is 0.75, but the span score is 0.

Description of additional features for MaxEnt model

We added the following features for the Maximum Entropy model:
For the current word, we added features from the signatures used for hmm tagging:
For each signature (other than the affix signatures, since suffixes and prefixes are already included as features) we add a feature if the signature describes the word.
For each of the preceding and succeeding two tokens, we add a feature if a word is described by a signature, including prefix and suffix signatures.