1 Named Entity Recognition

For the name entity recognition task, the code provided by the course was upgraded by adding features and then by modifying the model.

1.1 Constrained

Multiple features were designed. We first started by designing the features in the chapter 21 of Jurafsky and Martin's book:

- identity of wit
- identity of neighboring words
- part of speech of wi
- part of speech of neighboring words
- wi contains a particular prefix (from all prefixes of length less than 4)
- wi contains a particular suffix (from all suffixes of length less than 4)
- wi is all upper case
- word shape of wi
- word shape of neighboring words
- short word shape of wi
- short word shape of neighboring words
- presence of hyphen

The features "base-phrase syntactic chunk label of wi and neighboring words" was not implemented.

And the last features mentioned in the book is the use of a gazetteer. For this, we searched online had to search online and found one at http://download.geonames.org/, a gazetteer that was in the list of gazetteer from wikipedia. This gazetteer provided us with a list of names and aliases for places in the whole word, but also with categories for this places. The use of a gazetteer improved our score and we were surprised that the score didn't went even higher.

Other features computed where: The word in lowercase. How many time the word repeated in the previous 50 words or next 50 words in the corpus. This is because analyzing the corpus we saw that the proper names were repeated often. Also, we changed the number of neighbors to see the 3 last neighbors and the next three neighbors when computing all the previous features.

the adaboost classifier and the random forest classifiers were tried but both yield worse results.

1.2 Unconstrained

For the unconstrained results, a different model was tried. this model is CRF from sklearn_crfsuite. This model improved our score greatly. This is because in the perceptron model, we could not find a way to tell the model that after predicting for example B-ORG, we couldn't have I-ORG. Because everything was predicted in the .predict method of the classifier, we could not use the previous prediction as features. But the CRF has transitional probabilities and this allow for those errors to be prevented.

1.3 Conclusion

Overall we found that every features was really important, and that gazetteer, even if they improve the score, are not magical solutions that solve the problem automatically. After reading many papers, many interesting methods were found, like stacking models or using more complex features. this was not done because of a time constraint but it would have been really interesting to test them.