Project Report 4

CS 445

Natural Language Processing

Fall 2020-2021

Instructor:

Reyyan Yeniterzi

Faculty of Engineering and Natural Sciences

Sabanci University

Efe Şencan

25083

Features:

I have utilized 10 features for my training process. 7 of them are morphological features such as Stem of the word, Part of Speech, Prop, Noun Case, Ortographic Case, All Inflectional Features, and Start of the Sentence. The rest of the 3 features are denoted as “Person”, “Organization”, and “Location”. The value of these 3 features are determined based on the occurrence of that vocabulary in my previous gazeteers. If that particular vocabulary in the ne.txt exists in my person_gazetter.txt, organizational_gazeteer.txt, location_gazeteer.txt files, the value of these features are updated. For instance, if the current vocabulary in ne.txt exist in my Person gazetter, then the value of the “Person” feature will be True and the rest of them will be False. If that word, does not exist in any of these gazeteers, then they will all be False.

Tags:

I used 6 different tags for labeling the words in the provided txt file. The word will be tagged as “B-PER” if the tag of the word is “PERSON” and it is the starting word of that particular tag. The word will be tagged as “I-PER” if it’s tag is “PERSON” but it not the at the starting word position. The rest of the tags are “B-LOC”, “I-LOC”, “B-ORG”, and “I-ORG”. The same logic also applies for these tags. If the word does not have any specific tag in ne.txt file, then that word will be labeled as ‘O’.

CRF Model:

I used sklearn crf suite library for the machine learning model. My model is as follows:

```
crf = sklearn_crfsuite.CRF(
    algorithm='lbfgs',
    c1=0.1,
    c2=0.1,
    max_iterations=100,
    all_possible_transitions=False,
)
```

Reference:

https://sklearn-crfsuite.readthedocs.io/en/latest/tutorial.html

Baseline Model:

In order to measure the performance improvement of our model, I first build a baseline model which consists of only the "Stem" feature of the vocabulary. I used 5-fold cross validation as described in the project document and calculate the average f1-score, recall and precision for the tags in each fold.

Removing the 'O' tag

When we take a look at our data, we can observe that most of the labels of the words are consisted of the 'O' tag. Therefore, our dataset is highly imbalanced. To better evaluate the performance of our model, we removed the 'O' tag from the dataset.The results of the baseline model after the removal is as follows:

Average f1-scores	Average Recall	Average Precision
B-PER: 0.72	B-PER: 0.84	B-PER: 0.62
I-PER: 0.62	I-PER: 0.70	I-PER: 0.56
B-LOC: 0.53	B-LOC: 0.61	B-LOC: 0.47
I-LOC: 0.44	I-LOC: 0.54	I-LOC: 0.37
B-ORG: 0.55	B-ORG: 0.71	B-ORG: 0.44
I-ORG: 0.48	I-ORG: 0.61	I-ORG: 0.40

The average of the micro-avg metrics are **0.61.**

Adding New Features to the Existing Features:

When we add the gazetteer features namely the "**Person:", "Location:", "Organization:**" in addition to our existing feature "Stem", we observed the performance improvement in our model. The results of the model is as follows:

Average f1-scores	Average Recall	Average Precision
B-PER: 0.83	B-PER: 0.88	B-PER: 0.78
I-PER: 0.76	I-PER: 0.80	I-PER: 0.73
B-LOC: 0.77	B-LOC: 0.81	B-LOC: 0.73
I-LOC: 0.54	I-LOC: 0.64	I-LOC: 0.47
B-ORG: 0.60	B-ORG: 0.74	B-ORG: 0.50
I-ORG: 0.51	I-ORG: 0.60	I-ORG: 0.45

The average of the micro-avg metrics are **0.73.**

From these results, we can say that our model generates better predictions than the baseline, when we provide the information of whether that particular word previously occurred in a particular gazeteer type. Hence, we provide a prior knowledge to the model which leads to assigning higher weight during the label prediction process.

Adding the POS feature:

When we add the Part of Speech feature to the existing features in our word dictionaries, the performance of our model is as follows:

Average f1-scores	**Average Recall**	**Average Precision**
B-PER: 0.84	B-PER: 0.90	B-PER: 0.79
I-PER: 0.77	I-PER: 0.82	I-PER: 0.73
B-LOC: 0.78	B-LOC: 0.82	B-LOC: 0.75
I-LOC: 0.59	I-LOC: 0.72	I-LOC: 0.50
B-ORG: 0.63	B-ORG: 0.76	B-ORG: 0.53
I-ORG: 0.57	I-ORG: 0.69	I-ORG: 0.48

The average of the micro-avg metrics are **0.75.**

Adding the PROP feature:

Average f1-scores	**Average Recall**	**Average Precision**
B-PER: 0.85	B-PER: 0.90	B-PER: 0.81
I-PER: 0.77	I-PER: 0.82	I-PER: 0.73
B-LOC: 0.79	B-LOC: 0.82	B-LOC: 0.76
I-LOC: 0.59	I-LOC: 0.72	I-LOC: 0.50
B-ORG: 0.63	B-ORG: 0.76	B-ORG: 0.54
I-ORG: 0.57	I-ORG: 0.68	I-ORG: 0.50

The average of the micro-avg metrics are **0.758.** Although we haven't achieve high performance improvement in terms of the micro-avg, we observe minor improvements in the average precision for the I-ORG, B-PER, and B-LOC.

Adding the NCS feature:

Average f1-scores	Average Recall	Average Precision
B-PER: 0.87	B-PER: 0.90	B-PER: 0.84
I-PER: 0.83	I-PER: 0.86	I-PER: 0.80
B-LOC: 0.86	B-LOC: 0.88	B-LOC: 0.84
I-LOC: 0.61	I-LOC: 0.71	I-LOC: 0.53
B-ORG: 0.73	B-ORG: 0.87	B-ORG: 0.64
I-ORG: 0.63	I-ORG: 0.72	I-ORG: 0.56

The average of the micro-avg metrics are **0.81.** We observed significant amount of improvement when we provide NCS feature. The possible values of the NCS in our dictionary are “Nom”, “Acc”, “Dat”, “Abl”, “Loc”, “Gen”, “Ins” and “Equ”. If none of these values exist in the morphological analysis of that word, then we set the NCS feature as False.

Adding the INF feature:

Average f1-scores	Average Recall	Average Precision
B-PER: 0.87	B-PER: 0.91	B-PER: 0.84
I-PER: 0.83	I-PER: 0.85	I-PER: 0.82
B-LOC: 0.86	B-LOC: 0.88	B-LOC: 0.84
I-LOC: 0.62	I-LOC: 0.73	I-LOC: 0.54
B-ORG: 0.75	B-ORG: 0.88	B-ORG: 0.65
I-ORG: 0.65	I-ORG: 0.74	I-ORG: 0.57

The average of the micro-avg metrics are **0.81.** When we add the inflectional features to our word dictionary, we observed minor improvements in the B-ORG and I-LOC tags.

Adding the SS feature:

When we provide the information about whether that particular word is at the beginning position of the sentence to the model. The results are as follows:

Average f1-scores	Average Recall	Average Precision
B-PER: 0.87	B-PER: 0.91	B-PER: 0.84
I-PER: 0.84	I-PER: 0.85	I-PER: 0.83
B-LOC: 0.86	B-LOC: 0.89	B-LOC: 0.84
I-LOC: 0.61	I-LOC: 0.72	I-LOC: 0.53
B-ORG: 0.75	B-ORG: 0.88	B-ORG: 0.66
I-ORG: 0.65	I-ORG: 0.74	I-ORG: 0.59

Adding the OCS feature:

When provide the information of whether that particular is capitalized or not. The performance of our model is as follows.

Average f1-scores	Average Recall	Average Precision
B-PER: 0.89	B-PER: 0.91	B-PER: 0.87
I-PER: 0.87	I-PER: 0.86	I-PER: 0.88
B-LOC: 0.87	B-LOC: 0.89	B-LOC: 0.85
I-LOC: 0.66	I-LOC: 0.77	I-LOC: 0.58
B-ORG: 0.81	B-ORG: 0.88	B-ORG: 0.76
I-ORG: 0.75	I-ORG: 0.79	I-ORG: 0.71

The average of the micro-avg metrics are **0.85.** We observed a tangible performance improvement in terms f1-score especially for the I tags. This reason of this situation could be, the information of a word being capitalized important for determining the B and I tag. If a tag consist of multiple words, then there are common non-special words such as “ve” in Turkish, and the word “ve“ could not be capitalized. Therefore, the OCS feature is helpful for determining such cases.

Adding Additional Features

I also added additional features to my word dictionary such as the lowercased letter of the word, whether only the first letter of that word is capitalized or not, whether that word consist of digits, and also the last 2 and 3 letters of the word. After adding these, the performance of the model is as follows:

Average f1-scores	Average Recall	Average Precision
B-PER: 0.89	B-PER: 0.91	B-PER: 0.88
I-PER: 0.86	I-PER: 0.86	I-PER: 0.87
B-LOC: 0.88	B-LOC: 0.89	B-LOC: 0.86
I-LOC: 0.66	I-LOC: 0.77	I-LOC: 0.58
B-ORG: 0.81	B-ORG: 0.86	B-ORG: 0.77
I-ORG: 0.74	I-ORG: 0.77	I-ORG: 0.71

The average of the micro-avg metrics are **0.854.**

After adding all these features, the format of a single train instance is as follows:

```
{'INF': 'A3sg+Pnon+Nom',
 'Location': False,
 'NCS': 'Nom',
 'OCS': True,
 'Organization': False,
 'POS': 'Noun',
 'PROP': False,
 'Person': False,
 'SS': True,
 'Stem': 'müzik',
 'word.isdigit()': False,
 'word.istitle()': True,
 'word.isupper()': False,
 'word.lower()': 'müzik',
 'word[-2:]': 'ik',
 'word[-3:]': 'zik'}
```

Result:

As a result, we observed a drastic performance improvement compared with our baseline model when we add new features to our training set especially the ones related to the morphological analysis of the word. When there is no morphological features but only the stem of the word, we achieved an average micro-avg of 0.60 in our baseline model. We can increase that average micro-avg score to 0.72 by adding the additional information of whether that particular word exists in my previously tagged gazeeters. We

can further increase the average micro-avg scors up to 0.85 by adding morphological features of that word such as NCS, SS, OCS, INF, PROP, and POS properties. Overall, our results or compatible with the (Yeniterzi et al., 2011). We achieved high f1-score for predicting the named entity of a word using the described features. Moreover, If we had bigger dataset, we expect to further increase our prediction performance and reach even a closer performance with the paper.