Sooyeon Cho 5340075

Summary Report for the shared task GermEval2014

1. Motivation

Named entity recognition is one of the main tasks in natural language processing, and it can be used in a variety of fields like a search engine, content classification, and so on. While there are many researches and models for English named entity recognition, some language models are not developed enough to utilize them properly. GermEval2014 is an old shared task for named entity recognition in German language. Capitalization, which plays a big role in English entity recognition, is not a distinct characteristic in German language, thus I wanted to try German model and see the performances.

2. Data Description

The data, which is provided by GermEval2014, is sampled from German Wikipedia and News Corpora, annotated in two levels of named entities including nested entities. It uses NoSta-D annotation guideline with four main entity categories (person, location, organization and others) and sub-substructures (derivate and part). About 31,000 sentences are annotated in total and the data is provided in train, development and test set separately in tsv format. The task is to develop a classification model for two levels of named entities.

```
# http://de.wikipedia.org/wiki/Manfred_Korfmann [2009-10-17]
1 Aufgrund O O
2 seiner O O
3 Initiative O O
4 fand O O
5 2001/2002 O O
6 in O O
7 Stuttgart B-LOC O
8 , O O
9 Braunschweig B-LOC O
10 und O O
…
```

Table 1: An example sentence in the data

The lines start with # provide meta information of the sentence (Table 1). The first column is a token number, the second column is a token. The third and fourth columns are first and second level

(nested) entities, respectively.

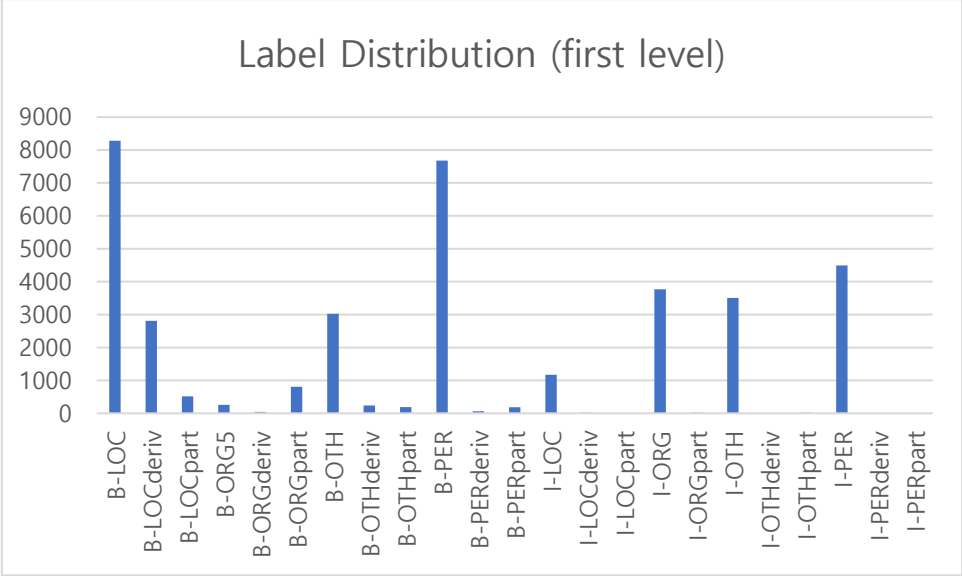

Figure 1: The distribution of first level entities in the train data

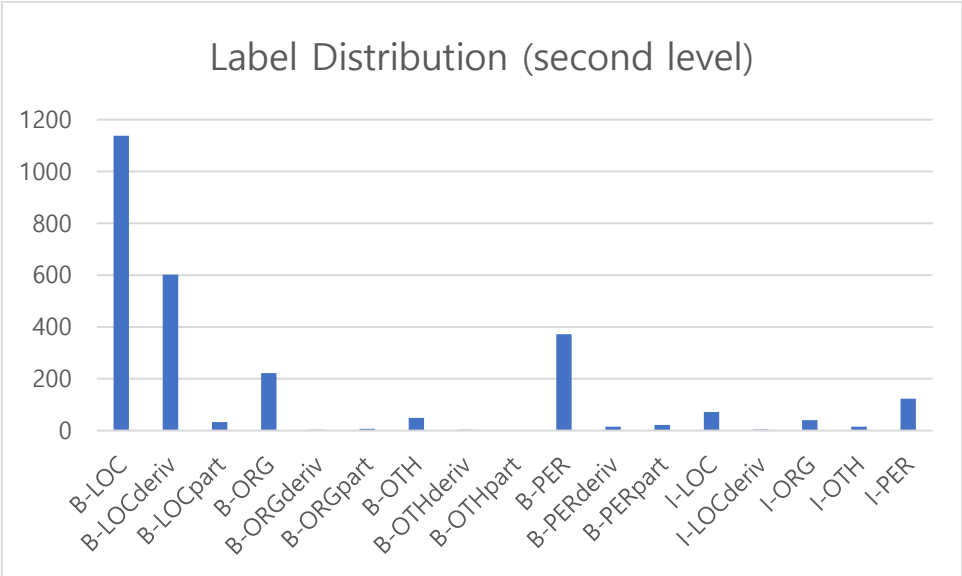

Figure 2: The distribution of second level entities in the train data

Figures 1 and 2 show that the labels are highly imbalanced. Labels like `I-LOCpart` and `I-PERderiv` even have only one occurrence each.

3. Models and Results

3-1. Model selection

The first attempt for developing a model was with spacy. Unfortunately, it did not go well because the data format and the setup are different from our lab task in the class, and the package documentation is not in detail. So, I used other models using scikit-learn python package. I tried various models like support vector machine, decision tree, naive bayes and MLP (MultiLayer Perceptron) classifier. The MLP classifier showed the best performance among many other classifiers. Default parameter was used for the final evaluation, because of issues with parameter tuning. It takes more than one day to run MLP classifier. I also tried grid search of the scikit-learn package, but it did not finish running after more than three days. In the Reddit lab project from the class, the default parameter showed the best performances in most cases, so I decided to try different models rather than tuning parameters in a single model.

3-2. Results

For the evaluation, the micro average f1-score was used instead of the macro average f1-score, because the number of classes is high. The label "O", which indicates that the token is not an entity, is excluded for evaluating performances. If the "O" label is included, the score is much higher (above 0.95). I used a dummy classifier for producing baseline.

Model	Entity Level	Precision	Recall	F1
SVM	Level 1	0.77	0.31	0.44
	Level 2	0.67	0.09	0.16
MLP	Level 1	0.71	0.37	0.49
	Level 2	0.62	0.09	0.15
Decision Tree	Level 1	0.15	0.39	0.22
	Level 2	0.02	0.44	0.04
Naive Bayes	Level 1	0.18	0.46	0.26
	Level 2	0.01	0.53	0.03

Table 2: Evaluation metrics of different models on the development set

Support vector machine and MLP classifier showed better results than decision tree and naive bayes

(Table 2). MLP showed slightly better performance than SVM, thus MLP was used for final evaluation. Interestingly, precision is higher than recall in SVM and MLP, whereas in decision tree and naive bayes recall is higher.

Model	Entity Level	Precision	Recall	F1
MLP	Level 1	0.71	0.36	0.48
	Level 2	0.29	0.00	0.01
Dummy (baseline)	Level 1	0.00	0.00	0.00
	Level 2	0.00	0.00	0.00

Table 3: Evaluation metrics on test set, MLP classifier and dummy classifier for baseline

The scores of level 1 entity performance on the final test set were similar to that on the development set, but level 2 showed poorer performance (Table 3). Dummy classifier gave 0.00 for every score.

Level 1	Precision	Recall	F1
B-LOC	0.81	0.51	0.63
B-LOCderiv	0.84	0.69	0.76
B-LOCpart	0.68	0.12	0.2
B-ORG	0.72	0.42	0.53
B-ORGderiv	0	0	0
B-ORGpart	0.81	0.1	0.18
B-OTH	0.76	0.34	0.47
B-OTHderiv	0.52	0.31	0.39
B-OTHpart	0.45	0.12	0.19
B-PER	0.78	0.48	0.59
B-PERderiv	0.25	0.18	0.21
B-PERpart	0.75	0.07	0.12
I-LOC	0.51	0.15	0.23
I-LOCderiv	1	0.75	0.86
I-LOCpart	0	0	0
I-ORG	0.45	0.18	0.26
I-ORGpart	0	0	0
I-OTH	0.42	0.09	0.15
I-OTHderiv	0	0	0
I-OTHpart	0	0	0

Level 2	Precision	Recall	F1
B-LOC	0.37	0.01	0.01
B-LOCderiv	0.26	0.03	0.05
B-LOCpart	0	0	0
B-ORG	0	0	0
B-ORGderiv	0	0	0
B-ORGpart	0	0	0
B-OTH	0	0	0
B-OTHderiv	0	0	0
B-OTHpart	0	0	0
B-PER	0	0	0
B-PERderiv	0	0	0
B-PERpart	0	0	0
I-LOC	0	0	0
I-LOCderiv	0	0	0
I-ORG	0	0	0
I-OTH	0	0	0
I-PER	0	0	0

I-PER	0.43	0.18	0.25
I-PERderiv	0	0	0
I-PERpart	0	0	0

Table 4: Classification report of MLP classifier on the test set

4. Conclusion and Discussion

This shared task provided good hands-on experience on named entity recognition. The task was challenging due to its long-running time and not satisfying performances, but I could get an good insight into the BIO-scheme NER project.

The first level entity f1-score of the final model on the test set is about 0.48, which is lower than the lab session from the class. The reason is that the task is much more complicated. Including BIO-scheme, three different entities and their substructures, the model has to classify into up to 24 categories.

Moreover, the experience shows that class imbalance is one of the important factors to consider in developing models. I tried giving different weights to classes in some models, but the performance did not improve. Especially, the number of entities in level 2 is too small and the classes are more imbalanced, so the result was much worse than in level 1. Unfortunately, I could not find ways to handle highly imbalanced classes in a classification model, this should be the next step in my future project.

5. References

- GermEval 2014 Named Entity Recognition Shared Task: Companion Paper - Darina Benikova , Chris Biemann , Max Kisselew , Sebastian Padó
- Modular Classifier Ensemble Architecture for Named Entity Recognition on Low Resource Systems - Christian Hänig, Stefan Bordag, Stefan Thomas
- GermEval-2014: Nested Named Entity Recognition with Neural Networks - Nils Reimers, Judith Eckle-Kohler, Carsten Schnober, Jungi Kim, Iryna Gurevych
- Morphology-aware Split-Tag German NER with Factorie - Peter Schüller
- HATNER: Nested Named Entitiy Recognition for German - Yulia Bobkova, Andreas Scholz, Tetiana Teplynska, Desislava Zhekova
- DRIM: Named Entity Recognition for German using Support Vector Machines - Roman Capsamun, Daria Palchik, Iryna Gontar, Marina Sedinkina, Desislava Zhekova

- BECREATIVE :Annotation of German Named Entities - Fabian Dreer, Eduard Saller, Patrick Elsa ̈sser, Ulrike Handelshauser, Desislava Zhekova
- Nessy: A Hybrid Approach to Named Entity Recognition for German - Martin Hermann, Michael Hochleitner, Sarah Kellner, Simon Preissner, Desislava Zhekova
- Semi-Supervised Neural Networks for Nested Named Entity Recognition - Jinseok Nam
- Adapting Data Mining for German Named Entity Recognition - Damien Nouvel and Jean-Yves Antoine
- Named Entity Recognition for German Using Conditional Random Fields and Linguistic Resources - Patrick Watrin, Louis de Viron, Denis Lebailly, Matthieu Constant, Stéphanie Weiser
- NERU: Named Entity Recognition for German - Daniel Weber and Josef Plötzl