Named Entity Recognition with the Structured Perceptron

Vijeta Agrawal

$17^{th} March$ 2019

1 Aim

The main focus of this report is to show the results of a named entity recogniser (NER) using the structured perceptron. For each word in a sequence, the named entity recogniser predicts one of the following labels:

- O: not part of a named entity
- PER: part of a person's name
- LOC: part of a location's name
- ORG: part of an organisation's name
- MISC: part of a name of a different type (miscellaneous, e.g. not person, location or organisation)

2 Introduction and Background

Named-entity recognition (NER) (also known as entity identification, entity chunking and entity extraction) is a subtask of information extraction that seeks to locate and classify named entity mentions in unstructured text into pre-defined categories such as the person names, organizations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc. En.wikipedia.org. (2019)

Standard Perceptron is extended to handle structured prediction problems(Collins, 2002). Due to its simplicity, structured perceptron is very popular in practice. Structured prediction is similar to multiclass classification task. The difference is that in structured prediction the label set Y represents a set of structures that can be generated from a given structured input x. For example, in POS tagging task, the label set for a given sentence is the set of all possible tag annotations for each word. The label set is denoted as a function Y(x) of the given input x. Zhao 2014

In our case, the label set is consists of above mentioned elements.

3 Training & Test Sets

The training and test sets consists of sentences up to 5 words long, each from the data used in the shared task of CoNLL-2003, which concerns language-independent named entity recognition.

4 Feature Extraction

4.1 Phi_1

The feature extraction function for Phi_1 takes the training data as an input which is a list of tuples and extracts word_tag features for each tuple. After extracting the features, it calculates the frequency of each feature in the training data set.

4.2 Phi_2

The feature extraction function for Phi_2 also takes the training data as an input and extracts previous tagcurrent tag features for each tuple. After extracting the features, it calculates the frequency of this new feature in the training data set.

5 NER Models

This report will feature the results of two NER Models, Phi_1 and Phi_1 & Phi_2. Phi_1 Model performs training and testing using Structured Perceptron algorithm with Phi_1 features, while on the other hand, Phi_1 & Phi_2 model uses the merged feature set of Phi_1 and Phi_2.

The results of these models are calculated using f1_score metric.

6 Model Performance

The performance of the models can be determined by calculating its error on training data or the f1_score for test data. On training data, the Phi_1 and Phi_1 & Phi_2 models converge at 6th and 30th iterations respectively. The error count of both models on each iteration is shown below in the graph.

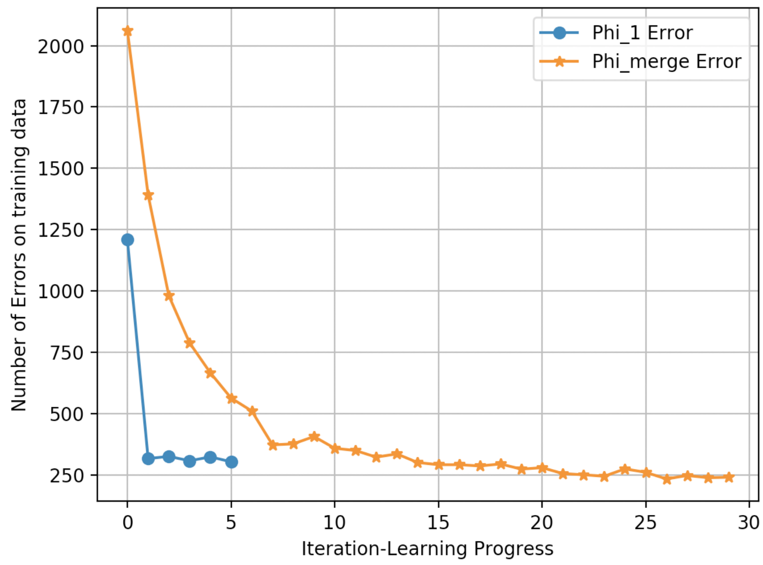

Figure 1: Learning Graph

7 F1_Score

The F1 score is the harmonic average of the precision and recall and it reaches its best value at 1 (perfect precision and recall) and worst at 0.
After running 6 iterations on Phi_1 and 30 iterations on Phi_1 & Phi_2 Models:

- The f1_score for Phi_1 Model is 0.7421
- The f1_score for Phi_1 & Phi_2 Model is 0.7335

The f1_score for both models are lower than expected because while testing, we are leaving out O, which is a majority class. Removing O was important also to fairly test our NER.

Label	**Phi_1**	**Phi_1 & Phi_2**
O	1996-08-22, . , BORROWER, LAST, AA+, REOFFER, =, NOTES, S, SHORT	ORG, 3, 1, 0, 2, 4, AT, Results, Attendance
PER	Peter, Colleen, Siegel, Hassan, Hafidh, Hillary, Gush,Steve, Stricker, O'Meeara	PER, McEwen, Fogart, Yoshikawa, Slight, Kocinski, SeguyYounis, Vialle, Koerts
LOC	BRUSSELS, LONDON, BEIJING, FRANKFURT, ATHENS, TUNIS, BAGHDAD, MANAMA, DUBAI, IRAQ	England, Finland, Italy, Calif., WASHINGTON, Bank, Mauritius, LONDON, SYDNEY, Pakistan
ORG	BAYERISCHE, VEREINSBANK, S&P, THAWRA, AN-NAHAR, AS-SAFIR, AL-ANWAR, AD-DIYAR, NIDA'A, AL-WATAN	Tokyo-Mitsubishi, ORG, POST, KANSAS, Newsroom, AD-DIYAR, Hampshire, Heerenveen, Albion, Blackburn
MISC	C$, Canadian, Open, Malaysian, Major, League, Baseball, AMERICAN, LEAGUE, EASTERN	League, German, English, Dutch, French, Democratic, Portuguese, GMT, MISC, Scottish

Table 1: Top 10 Features of each label for both the models

Above table represents the top 10 feature for each label in each model. The features makes sense because we get the names like Peter and Colleen in PER(people), places like Brussels, London in LOC(location) etc.

The differences between the f1_score of Phi_1 and Phi_1 & Phi_2 models are not what I was expecting, since increasing the features should have improved the performance but it didn't. So according to me, the features proposed were not very good, but using better features might have improved the results.

References

[Zha14] Kai Zhao. *Structured Prediction with Perceptron: Theory and Algorithms.* Tech. rep. Technical Report November, City University of New York. URL https://pdfs . . ., 2014.