Feature Engineering for Supervised Learning with Small Training Set

Yashar Mehdad [1], Scurtu Vitalie [2]

[1] University of Trento, FBK – Irst, Trento, Italy

[2] RGB s.r.l, Banzai Group, Milan, Italy

Abstract. The supervised Machine Learning(ML) approaches, has never disappointed the researchers, providing that, a reasonable training data in terms of amount and quality, is available. Since the data set plays an outstanding role in these algorithms, there are many attempts to decrease the size of annotated data needed for the certain task, or perform a fast and not expensive programming and annotation approach on the data. In this paper, we try to prove the effect of feature engineering when there is a limitation in the training data. We tired to tune a set of features for the Named Entity Recognition(NER) task. NER is one of the significantly important preliminary steps prior to main natural language processing tasks. Using supervised learning approach with the small training data set, we showed that our proposed set of features could prove to be very effective using a small training set of merely about 20000 tokens, while the results in the last experiences or the published papers, illustrate that learning curve in NER considerably dependents on the size of annotated data.

Keywords: Feature Engineering, Feature Selection, Supervised Learning, Named Entity Recognition.

1. Introduction

Working on supervised ML approaches, the main issue in one's mind, is the data set. Amongst the important characteristics which describe a reasonable data set for any task, size has been constantly an issue which always contrasts with the cost and time constraints. Preparing a large set of annotated data, which is not only expensive but also time consuming, motivating the idea of working with the small data set.

Approximately the entire results in previous experiences and papers, illustrate the dependency of the results and the size of data set in machine learning methods. Learning curves prove that having a large data set can increase the efficiency and results to some extent. This fact can be more observable in the Natural Language Processing (NLP) area. This idea encouraged us to select Named Entity Recognition task for our experimental data set.

Named Entity Recognition, in computational linguistics terms refers to the task of identifying the named entities which represent an instance of a name, location, person or organization. Since many tasks and applications in Natural Language Processing (NLP), such as Information Extraction and Summarization, Information Retrieval, Data Mining and Question Answering, are dependent on Named Entity Recognition, this task is considered as one of the main and important preliminary works in this field. The number of research papers which has been published during last few years is a clear evidence for the significant weight of this job for almost all attempts in the area of Human Language Technology. Amongst approaches which are applied to solving the problem of NER, Statistical methods proved to be effective, fast, and popular. Machine Learning algorithms used in this approach, appeared to be successful as well as reasonable, providing that a fairly large data set with high quality is available. Results gained using the ML methods, considering the features and data set as two important factors, never disappointed the attempts in this area.

This study is an attempt to describe a NER system using Machine Learning approach to improve the results exploiting a small set of data (only 20000 tokens) applying the selected set of features. The work tries to lead the idea of small training set to reality by selecting and tuning the features extracted from the data. The rest of this paper presents the system description and feature selection phase. The experiments carried out and the results are explained as well and finally the conclusion is made and future works are proposed.

2. System Architecture

Our Named Entity Recognition system is mainly based on the YAMCHA classifier machine which was initially created as generic, customizable and open source text chunker and can be adapted to the various tag-oriented NLP tasks. We used Support Vector Machine (SVM) algorithm as machine learning classifier provided by YAMCHA as well as other specifications such as window-size besides the built-in algorithm for multi-class problem (pair wise/one vs. rest).

For this research, we have carried out a fine tuning features study in order to improve the information given for the classifiers, removing noisy features and incorporating new meaningful features. The final set of the features is shown and described in the next section. A simple rule-based system was developed and attached to the system to be used as a feature which proved to be well improved the set of features.

The structure of the system is summarized in two main parts, Feature Extraction and Feature Selection. The Feature Extraction portion was made out of the YAMCHA boundary, the Feature Selection subsequently, selected the set of features based on the experiments and results. At each time, the features extracted, added or deducted to resolute the results and consequently compare the gained results to reach the optimal state. Figure 1 demonstrates the system structure and steps.

For this purpose we tried to choose *n* number of candidate features for the initialization phase. We selected all possible relevant features for this task which possibly could be extracted easily from the available resources or directly from the data set. Having a large selection of features we tried to find the best features which could stabilize the learning curve.

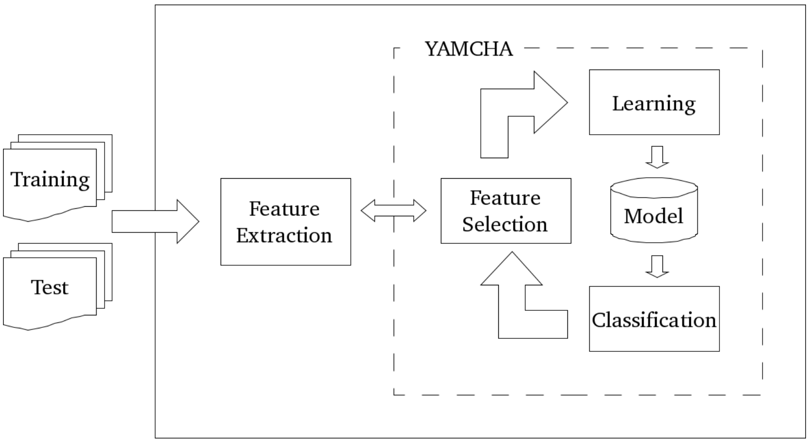


Fig. 1: Structure of the system

3. Experiments

A set of experiments was conducted on the English training set. A portion of CoNLL-2003 shared task English data set (45000 tokens) was used to evaluate the system and carry out the experiment as well as increasing the performance based on the models built. Since the main aim of the study is to optimize the NER performance with the small data set, only a portion (about 20%) of the original set was used and the learning curves is based on 45000 tokens data set for training.

Using the following 34 features, the window size for static features (all extracted feature) and dynamic feature (class labels) were tuned for each experiment. Different window sizes was used to candidate the best, and finally for static feature -2 to 2 (two feature before and after the current token) and -2 to 2 as well as -4 to 4 was chosen for dynamic feature window size. The dynamic feature window size would take into account a window of *-i* to *i* from the class labels as features. In another word, the class labels are used as features during the learning and classification.

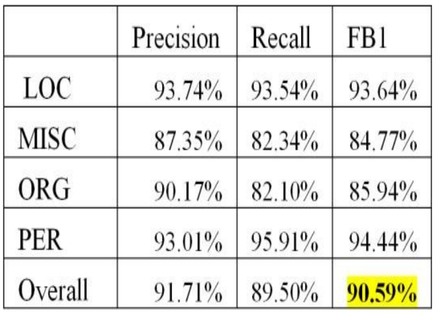
In the feature extraction phase a set of features, initially extracted and selected to be implemented in the training phase. In each step, number of features, which were estimated to be effective in the experiments, extracted and applied in the system, in addition amongst all features extracted, each iteration, a selection took place to approximate the optimal set of features.

Amongst 50 features which were extracted, the following is the final set of features which estimated to optimize the performance of the system with the small training data.

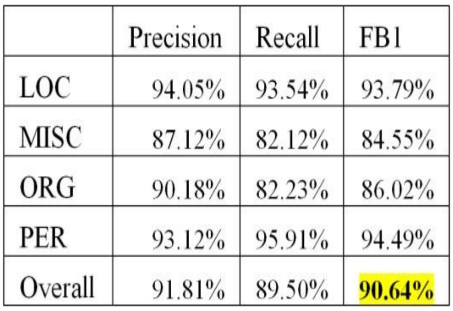
- Lower and upper case of each token.
- Part of Speech tags and Chunk-label of each token.
- Punctuation(being or containing).
- Being upper-case/lower-case.
- The token contains digits, the token is digit(number)
- The length of the token.
- Prefix and suffix of each token.
- Token's Number of senses in Word-Net.
- The token is in Word-Net.(Boolean)
- The token is the first word of a sentence.
- Frequency of the token in a corpus.
- The token exists in the list of locations, Organizations, People and Misc (different lists merged for this purpose).
- The token is inside the stop words list.
- Stem of each token.
- Lemma.
- A list of tokens which is more probable to co-occur with any organization or company was created and checked whether the token is in the list or not.(this feature was extracted to increase the F measure for the organizations)
- A list of collocations from a raw text corpus was extracted, filtered and used to check whether the token is inside the collocation list or not.

4. Results

Based on the number of experiments, which was conducted on the training and test data, the set of 34 features with two different dynamic window sizes was selected to be demonstrated. The following tables and the graphs, illustrate the results of experiments with the small training set.



	Precision	Recall	FB1
LOC	93.74%	93.54%	93.64%
MISC	87.35%	82.34%	84.77%
ORG	90.17%	82.10%	85.94%
PER	93.01%	95.91%	94.44%
Overall	91.71%	89.50%	**90.59%**



	Precision	Recall	FB1
LOC	94.05%	93.54%	93.79%
MISC	87.12%	82.12%	84.55%
ORG	90.18%	82.23%	86.02%
PER	93.12%	95.91%	94.49%
Overall	91.81%	89.50%	**90.64%**

Table 1 & 2: Results based on -2 to 2 and -4 to 4 dynamic window size respectively.

The graphs below clearly demonstrate the results in each class (Location, Person, Organization and Misc.). it can be clearly noticed that the F measure for PER and LOC in both cases is higher than ORG and MISC which could be correspondence to their distinction and ambiguity. Moreover the existence of rich Gazetteers for PER and LOC could help increasing the performance in these two classes.

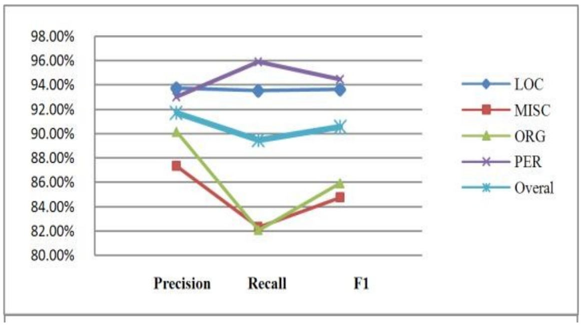


Fig. 2: Results based on -4 to 4 dynamic window size.

5. Conclusion and Feature Work

Based on the results, by tuning the features, we could obtain the good results with a small training set. Since annotation is one of the main issues of machine learning approach, we could reduce the time and cost of this task by reducing the training data, perform a reasonably good result with not large annotated data set. By having a glance at the following learning curves, we could simply observe that even by reducing the current 45000 tokens training set to 20000 we still could achieve a reasonable and fine outcome devoid of a remarkable declining in performance.

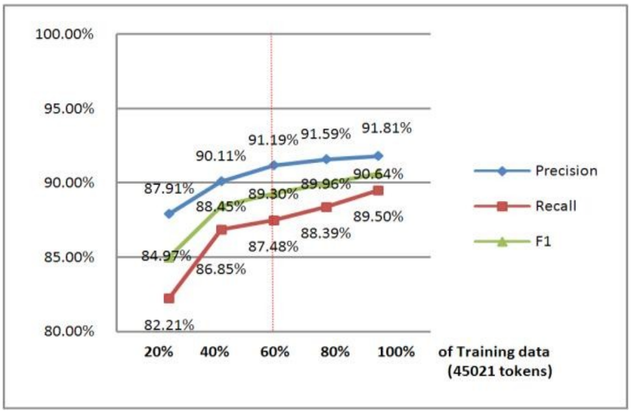


Fig. 3: Learning Curve based on 34 features (-4 to 4 dynamic feature window size)

Figure 3 shows the boundary of about 60% to achieve the reasonable results with only about 25000 tokens which is about only 10% of the original training size of CoNLL 2003 shared task. The results which was gained using only 10% of original training data, was about 90% which is a very good achievement with such a small training corpus.

This study is an evidence of the significant role of feature selection and tuning, particularly when the training size is small. The results could generalize to other tasks of NLP when the data set or annotated data and annotation task is an issue. The same set of experiments could be conducted for other tasks and applications, to be used as the reference set of features while there is lack of data to be used for training the system.

Another interesting issue to be experimented and discussed is the effect of features set on language independent Named Entity Recognition. Since this study was focused merely on English NER, there would be a place to develop the feature set for other languages and language independent tasks. It is worthwhile to mention that, the choice of NER for this study was basically framed on the importance of this application as an element and obligation for other general or specific tasks in computational linguistics.

6. Acknowledgment

We would like to especially thank Bernardo Magnini and Roberto Zanoli for their kind support during the project.

7. References

- Magnini, B., Negri M., Prevete R., Tanev H., 2002. *The Theory of Parsing, Translation and Compiling*, Proceedings of SemaNet '02: Building and Using Semantic Networks Taipei, Taiwan(2002) 38.44.
- Chinchor, N., Robinson, P., Brown, E. 1998, *Hub-4 Named Entity Task Definition (version 4.8).* Technical Report, SAIC. http://www.nist.gov/speech/hub4_98 (1998).
- R. Florian, H. Hassan , A. Ittycheriah, H. Jing N. Kambhatla, X. Luo, N. Nicolov, and S. Roukos, 2004. *A Statistical Model for Multilingual Entity Detection and Tracking.* In NAACL/HLT.
- A. Mikheev, M. Moens, and C. Grover. 1999. *Named entity recognition without gazetteers.* In Proceedings of EACL'99.
- E. F. Tjong Kim Sang. 2002. *Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition.* In Proceedings of CoNLL-2002, pages 155–158.
- Zornitsa Kozareva, Oscar Ferrández, Andres Montoyo, Rafael Muñoz, Armando Suarez and Jaime Gomez, *Combining data-driven systems for improving Named Entity Recognition*. in the Journal Data and Knowledge Engineering, Volume 61, number 3, pp. 449-466.
- Oscar Ferrandez, Antonio Toral, and Rafael Munoz, 2006. *Fine Tuning Features and Post-processing Rules to Improve Named Entity Recognition.*
- James Mayfield, Paul McNamee, Christine Piatko. *Named Entity Recognition using Hundreds of Thousands of Features.* In the Proceedings of the 7th Conference on Natural Language Learning (CoNLL 2003), Edmonton, Candada, pp. 184-187, May 2003.
- Fredrick E. Kitoogo, V. Baryamureeba, 2007. *A Methodology for Feature Selection in Named Entity Recognition.* International Journal of Computing and ICT Research, Vol. 1, No. 1, June 2007.
- M.Ciaramita, Y. Altun. 2005. *Named-Entity Recognition in Novel Domains with External Lexical Knowledge*. In Workshop on Advances in Structured Learning for Text and Speech Processing (NIPS 2005).
- Zornitsa Kozareva, 2006. *Bootstrapping Named Entity Recognition with Automatically Generated Gazetteer Lists*. in Proceedings of EACL student session (EACL 2006) , Trento, Italy, April 2006
- Munro, Robert and Daren Ler. 2003. *Meta-Learning Orthographic and Contextual Models for Language Independent Named Entity Recognition*. Proceedings of CoNLL-2003, Canada,
- T. Kudo and Y. Matsumoto. 2001. Chunking with Support Vector Machines. In *Proc. of NAACL 2001*, pages 192–199.
- H.Schmid, "Probabilistic part-of-speech tagging using decision trees," in *Proceedings of International Conference on New Methods in Language Processing*, September 1994.
- C.J. van Rijsbergen, S.E. Robertson and M.F. Porter, 1980. *New models in probabilistic information retrieval.* London: British Library. (British Library Research and Development Report, no. 5587).