Natural Language Processing and Information Retrieval with Applications in Social Networks Final Project

Abdul Gafar Manuel Meque
Institute of Information Science / Address line 1
Nataional Chengchi University / Address line 2
Taiwan International Graduate Program / Address line 3
ameque@iis.sinica.edu.tw

Abstract

In the present project, I present alternative feature set selections to improve the performance of the Noun Phrase chunker featured in (Bird et al., 2009). Several feature set plus algorithms combinations where examined and the end system, has proven to greatly outperform the baseline system by more 3 points in F1-Measure.

1 Introduction

The task of Chunking, also know as shallow parsing, is the process of analyzing a sentence by its basic (most elementary) unit of information (constituents parts) for later perform the linkage to a higher order unit (such as Noun Phrases), has been made popular by its introduction as a shared task in CoNLL 2000. Various approaches at solving the Chunking task have been employed over the past 15 years or so, ranging for rule based approaches using regular expression rules to (more computational expensive and widely employed today) statistical approaches (Jurafsky and Martin, 2009). The current work will focus mainly in the statistical approaches, using the machine learning and natural language processing modules provided by the nltk package, the subject of (Bird et al., 2009). Three different algorithms will be explored with various feature sets combinations and the results will be evaluated and the best performing one will be chosen for further inspection.

2 Dataset, Task and Baseline

2.1 Data

The data set for this project comes from the CoNNL-2000 shared task, concerning chunk segmentation recognition in the test set using machine learning approach. Two sets where provided (preavailable in nltk), one for training and the other one for testing, making a total of 259104 word tokens, structured as follows:

Chunk Tag	Count
B-NP	55081
I-NP	63307
O	27902

Table 1: Chunk tags & their respective counts

- Training Set
  - 211727 word tokens
  - 8936 chunked sents (the actual target of the work) of NP type
- Test Set
  - 47377 word tokens
  - 2012 chunked sents (the acctual training set in this work)of NPtype

After a few analysis over the distribution of the NP chunk tags, POS tags and words in the training , it is clear that there is a substantial amount of words tagged as I-NP and B-NP as depitected in the table bellow:

2.2 Task and Goals

The main task in this project is to devise ways to improve the performance scores of the baseline system described in [2.3]

2.3 Baseline System

The baseline used for the current project is provided in the chapter 7.3 of (Bird et al., 2009), to run the system an external software is required, its described in section [3]. The following table shows the performance of the baseline system on the test set, using different algorithms. This step is necessary to provide backing to the intuition that

Score	MEwM	MEwG	NB	NBB	SVM
IOB	96.1	96.0	95.5	95.0	96.2
Preci.	88.8	88.3	85.9	86.1	88.7
Recall	91.2	91.2	90.0	90.0	91.5
F-Me.	90.0	89.8	87.9	88.0	90.1

Table 2: (Bird et al., 2009)'s ChunkParse score (baseline)

Features	IOB Accuracy	Precision	Recall	F1 Score
Set1	96.6	91.1	92.2	91.7
Set2	96.7	91.4	92.4	91.9
Set3	97	91.8	93.1	92.4
Set4	97.1	92.3	93.3	92.8

Table 3: System performance uses SVM

the algorithm also influences the performance, but it allows for a clear view on the degree of influence.

3 Proposed Approach

Gather all the statistical information from the train data and after a careful analysis of the baseline chunker, the new step is to redesign the features selection and test. This is a two step process: spacing

label choose candidates feature sets

label train and test using three best performing algorithms from the baseline.

To prevent over-fitting and over-tuning the model on the basis of the test data, a cross-validation testing scheme will be adopted, using only the training and development set, using 10 fold.

3.1 Feature Selection

Combination of significant features have been selected, following the proposed selection schemes from similar works such as (editor,) and (editor,)

- Feature Set 1:
  - Word Lemma, POS Tag with context, Word Shape
- Feature Set 2:
  - From Set 1 + word
- Feature Set 3:
  - Set 1+ subtree (parse tree)
-
  - POS,Word,Shape,tagsinceDT

3.2 Solving for unseen words

To enhance the feature vector, the wordnet thesaurus is employed, first to update the Word feature (this particularly true in Set3 and Set4) will not always be the exact input word, it will be either the POS of the word (if the word is not in the wordnet's synsets), and secondly the lemmatizer is employed to acquire the lemmas.

4 Test and Results

Applying the feature selection the following performance scores are achieved by the system with LinearSVC and different feature sets:

As we can see from the above table, SVM' SVC outperforms the other algorithms for this particular task, in this settings, so the final code is implemented as such.

5 Conclusion

The final result of all implementations and changes for improvements made in this project can be verified by the code implementation, that provides all required aspects from scoring and also allows for cross-validation.

References

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python Analyzing Text with the Natural Language Toolkit*. OReilly Media Inc.

editor, editor.

Daniel Jurafsky and James H. Martin. 2009. *Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics.* Prentice-Hall., 2nd edition edition.