AUTOMATIC EXTRACTION OF RULES FOR SENTENCE BOUNDARY DISAMBIGUATION

E. STAMATATOS, N. FAKOTAKIS, AND G. KOKKINAKIS

Dept. of Electrical & Computer Engineering
University of Patras
26500-Patras-Greece
stamatatos@wcl.ee.upatras.gr

ABSTRACT

Transformation-based learning (TBL) is the most important machine learning theory aiming at the automatic extraction of rules based on already tagged corpora. However, the application of this theory to a certain application without taking into account the features that characterize this application may cause problems regarding the training time cost as well as the accuracy of the extracted rules. In this paper we present a variation of the basic idea of the TBL and we apply it to the extraction of the sentence boundary disambiguation rules in real-world text, a prerequisite for the vast majority of the natural language processing applications. We show that our approach achieves considerably higher accuracy results and, moreover, requires minimal training time in comparison to the traditional TBL.

INTRODUCTION

The technological advances of the recent years have facilitated the collection and the automatic processing of large volumes of text. The development of large-scale applications in information extraction and information retrieval has paid special attention to the low-level text processing tasks such as proper name detection, sentence boundary detection, text chunking etc. as well as to the involvement of machine learning theories for acquiring automatically the appropriate knowledge.

In contrast to empirical approaches that try to represent linguistic knowledge based on subjective assessments, several corpus-based approaches have been proposed for the automated learning of linguistic knowledge. The most powerful one is the TBL theory (Brill, 1995) which combines the high degree of both robustness and accuracy of a corpus-based method with the representation of the captured information in a clearer and more direct fashion.

TBL has been applied successfully to a wide range of natural language processing tasks, including part-of-speech tagging (Brill, 1995), text chunking (Ramshaw & Marcus, 1995), spelling correction (Mangu & Brill, 1997) and dialog act tagging (Samuel, 1998). Although TBL is an application and language independent theory, it is obvious that when the features of the application are not taken into account it may be proved to be insufficient.

In this paper we present our approach to the automatic extraction of the disambiguation rules for sentence boundary detection in Modern Greek text. Sentence boundary identification is not a trivial problem since the punctuation marks are usually ambiguous. For example, a period may denote an abbreviation or a decimal point besides the end of a sentence. The ambiguity of the punctuation marks varies according to the text genre or the specific corpus. About 47% of the periods in the *Wall Street Journal* corpus denote abbreviations while the corresponding percentage for the *Brown* corpus is only 10% (Church & Liberman, 1991). This fact means that if no sentence boundary disambiguation rules would be taken into account we would be able to correctly identify about 53% of the sentences in the *Wall Street Journal* corpus and about 90% of the *Brown* corpus, considering any other ambiguity as negligible.

Our approach is a variation of the traditional TBL theory that takes into account the properties of this problem and simplifies the learning procedure. In particular, sentence boundary detection is characterized by a limited number of possible transformations. Moreover these transformations are unambiguously ranked according to their frequency of appearance and the triggering environments are not overlap. We show that our methodology performs better than TBL as concerns both training time cost as well as accuracy results based on experiments on a 200,829 word corpus.

The paper is organized as follows: Section 2 briefly describes the traditional TBL theory. Section 3 includes our approach while in Section 4 performance results are presented. Finally in Section 5 some conclusions are drawn and future work directions are given.

TRANSFORMATION-BASED LEARNING

TBL requires the existence of already tagged corpora in order to extract automatically the linguistic knowledge. Initially the training corpus is annotated based on an initial-state annotator and the annotated corpus is compared to the *truth* (i.e., the manually tagged corpus). An ordered list of transformations is, then, learned by performing the following procedure: Every possible rule of the following format is applied to the annotated corpus:

IF triggering environment THEN transformation

where the *transformation* changes the state of a tag if the condition described in the *triggering environment* is valid. Moreover, the degree in which the resulting corpus resembles to the truth according to an objective function is calculated. The rule with the lowest error rate is selected and it is applied to the annotated corpus. The learning continues by applying all the possible rules to the new annotated corpus for selecting the next rule. Thus, an extracted rule improves the accuracy of the annotated corpus. Learning stops when no rule manages to improve the accuracy of the annotated corpus beyond a predefined threshold.

For applying the acquired knowledge to a new text, that text has to be annotated by the initial-state annotator. The ordered list of learned rules is, then, applied. It has to be underlined that a rule is applied to the entire text before the next rule is examined.

TBL is an application-independent theory. In order to be adapted to a specific application the following have to be defined (Brill, 1995):

- The initial state annotator
- The space of allowable transformations (rules and the triggering environments)
- The objective function for comparing the corpus to the truth and choosing a transformation

Since the definition of every possible transformation is a hard task for certain applications, data-driven algorithms can be used for excluding cases that are not likely to be detected in a text. Typically, the TBL theory is independent of the complexity and the accuracy of the initial-state annotator. However, the more accurate the initial-state annotator, the less training time cost.

OUR APPROACH

Sentence Boundary Detection

A sentence boundary detector aims at the disambiguation of the potential sentence boundaries. In particular, there are certain punctuation marks that may denote the End Of a Sentence (EOS). In our study for Modern Greek we consider the following punctuation marks as potential sentence boundaries: period (.), exclamation point (!), question mark (; in Modern Greek), and ellipsis (…). Moreover, there are some cases where the colons (:) may also denote a sentence boundary but we consider those cases as negligible. Notice that the aforementioned punctuation marks are not located necessarily at the end of a token. A sequence of closing punctuation marks (e.g., *),], },* etc.) is likely to follow.

Each punctuation mark that may denote a sentence boundary has its own properties. As mentioned in the introduction, if every potential sentence boundary is considered as sentence boundary the resulting accuracy would be considerably high depending on the text-genre of the test corpus. This accuracy is equal to the *lower bound* of the corpus and every sentence boundary disambiguation algorithm has to perform better than that. Moreover, the space of allowable transformations includes two cases. A regular punctuation mark may be transformed to a sentence boundary and vice versa.

Several approaches have been proposed as regards the scope of the triggering environment in a sentence boundary disambiguation task. For example, (Reymar and Ratnaparkhi, 1997) propose a trainable model based on maximum entropy that requires no complicated information concerning the token that contains the candidate punctuation mark, one token before, and one after that. The system SATZ (Palmer & Hearst, 1997), makes use of a fully-connected feed-forward neural network for disambiguating sentence boundaries and requires prior POSP probabilities for each word acquired by the training corpus. It achieves 98.5% accuracy on a corpus of *Wall Street Journal* articles based on a 30,000 word lexicon using information about a 6-token context, that is 3 tokens preceding the candidate punctuation mark and 3 following. In this paper we use a more restricted triggering environment. In particular, we use information relevant to the token containing the candidate sentence boundary and the token that immediately follows. In more detail, the information we use consists of the following features:

1. Preceding word: the string that remains after the removal of any punctuation marks both in the beginning and in the end of the preceding-token.
2. Preceding punctuation marks: a sequence of punctuation marks that there may be to the left of the candidate sentence boundary in the preceding-token.

3. Following punctuation marks: a sequence of punctuation marks that there may be to the right of the candidate sentence boundary either in the preceding-token or the following-token.
4. Following word: the string that remains after the removal of any punctuation marks both in the beginning and in the end of the following-token.

For each of the above features simple measures are calculated, such as word length, first character type, last character type etc. The information used as triggering environment is independent of the state of the potential sentence boundary. Thus, the transformation of the state of this potential sentence boundary does not affect the triggering environments of its adjacent candidate sentence boundaries as well.

Methodology

Initially, all the candidate punctuation marks are considered to denote sentence boundaries. Then, a set of rules of the following format is applied to the entire text:

Rule set 1: IF *triggering environment* THEN *remove sentence boundary*

where *triggering environment* is the contextual information of a sentence boundary (i.e., the simple measures associated with the four measures as described in the previous subsection). After all the rules of set 1 have been applied, a second set of rules of the following format is applied to the entire text:

Rule set 2: IF *triggering environment* THEN *insert sentence boundary*

where *triggering environment* is the contextual information of a candidate sentence boundary as above. It has to be underlined that each punctuation mark that may denote the end of a sentence (i.e. period, exclamation point, question mark, and ellipsis) has its own rule sets 1 and 2.

In contrast to the traditional TBL, our methodology applies all the rules that perform the most likely transformation regardless the errors that may produce. Afterwards, all the rules that perform the next transformation are applied. This methodology certifies that the maximum number of transformations that may be applied to a certain triggering environment for the sentence boundary disambiguation problem is two.

Automatic Extraction of Rules

The rule sets 1 and 2 for each punctuation mark that may denote the end of a sentence are acquired automatically based on a training corpus. This corpus has to be tagged manually by inserting a special symbol after the end of each sentence. The procedure for selecting the rules is described below. For every possible triggering environment, a rule of the following form is considered (i.e., Prolog predicate):

rule(PUNC_MARK, N1, N2, TRIGGERING_ENVIRONMENT)

where

- *PUNC_MARK* is the specific punctuation mark we wish to extract disambiguation rules for,
- *N1* is an integer that indicates how many times the corresponding *TRIGGERING_ENVIRONMENT* of a potential EOS of the given *PUNC_MARK* that does not denote the end of a sentence has been detected in the training corpus, and
- *N2* is an integer that indicates how many times the corresponding *TRIGGERING_ENVIRONMENT* of a potential EOS of the given *PUNC_MARK* that denotes the end of a sentence has been detected in the training corpus.

The criterion, then, for a rule to be included in the Rule Set 1 of the given *PUNC_MARK* is:

$N1 > N2$, and $N2 < Total\_Candidate\_EOS * 0.01$

where *Total_Candidate_EOS* is the total number of the candidate EOS for the entire training corpus. The corresponding criterion for the Rule Set 2 is:

$N1 = 0$, and $N2 > 0$

These criteria have been acquired empirically. It has to be noted that initially we performed experiments using symmetric criteria for the two rule sets. However, it has been proved that the criterion for the rule set 2 has to be quite restricted in order to attain high accuracy results.

PERFORMANCE

The corpus we used for training and testing is composed by real-world text downloaded by the World Wide Web page of the Modern Greek weekly newspaper *TO BHMA* (the tribune) (Dolnet, 1998). Analytical data for this corpus are presented in table 1.

	Training corpus	Test corpus
Words	165,465	200,829
Sentences	7,274	8,736
Candidate EOS	9,136	10,977
Lower bound (%)	79.6	79.6

Table 1. The training and the test corpus.

The application of a certain sentence boundary disambiguation method to a text may produce two kinds of errors (Palmer and Hearst, 1997):

- **False Positive**: a punctuation mark the method erroneously labeled as a sentence boundary.
- **False Negative**: an actual sentence boundary that the method did not label appropriately.

The results of applying our method to the test corpus are presented in table 2. Analytical accuracy results for each punctuation mark as well as the number of acquired rules are given in table 3.

Accuracy (%)	99.4
False positives	40
False negatives	26

Table 2. Results on the test corpus.

Punctuation mark	Number of rules	Correct	False positives	False negatives
Period	190	9,796	29	17
Exclamation point	32	270	0	3
Question mark	46	522	0	5
Ellipsis	44	323	11	1
Total	312	10,911	40	26

Table 3. Analytical results for each punctuation mark.

In order to compare our method to the traditional TBL theory we applied it to the same corpus dealing with the periods only. The accuracy results are presented in table 4.

Learning algorithm	Total cases	False Positives	False Negatives	Accuracy
TBL	9,842	389	24	95.8
Our method	9,842	29	17	99.5

Table 4. Comparison to TBL theory.

Regarding the training time cost given that t is the time cost required by the TBL and n is the number of rules produced by TBL, our method requires:

$$training\ time\ cost = t / (n-1)$$

since all the rules are extracted by comparing only one time the corpus to the truth. It has to be underlined that TBL extracts one rule during each iteration.

CONCLUSIONS

We presented an approach to automatic extraction of rules for sentence boundary disambiguation. Our method is a variation of the TBL theory. Although TBL has been proved to be sufficient for a wide variety of natural language processing tasks its application to a certain problem without taking into account the intrinsic characteristics of this problem usually cause important losses as concerns both accuracy and training time cost. Hence, our approach takes full advantage of the features of the sentence boundary detection problem in order to improve the performance.

All the experiments were based on real-world text downloaded from the World Wide Web. It has been shown that the presented methodology achieves high accuracy results, comparable to other systems that are based on more complicated resources.

We believe that the same method can be successfully applied to resolve the sentence boundary disambiguation problem in other languages. Especially languages such as Spanish, or Italian, having similar characteristics to Modern Greek would mostly benefit.

REFERENCES

Brill E. (1995) "Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging", *Computational Linguistics*, 21(4), pp. 543-565.

Church K. and M. Liberman (1991) "A Status Report on the ACL/DCI". In *Proc. of the 7th Annual Conference of the UW Centre for the New OED and Text Research: Using Corpora*, pp. 84-91.

Dolnet, (1998) "*TO BHMA*". Lambrakis Publishing Corporation, http://tovima.dolnet.gr/

Mangu L. and E. Brill (1997) "Automatic Rule Acqisition for Spelling Correction". *In Proc. of the 14th International Conference on Machine Learning (ICML-97).*

Palmer D. and M. Hearst (1997) "Adaptive Multilingual Sentence Boundary Disambiguation", *Computational Linguistics*, pp. 241-267, 23(2).

Reynar J. and A. Ratnaparkhi (1997) "A Maximum Entropy Approach to Identifying Sentence Boundaries", In *Proc. of the 5th ANLP Conference*.

Ramshaw L. and M. Marcus (1995) "Text Chunking using Transformation-Based Learning". In *Proc. of ACL Third Workshop on Very Large Corpora*, pp. 82-94.

Samuel K. (1998) "Dialogue Act Tagging with Transformation-Based Learning". In *Proc. of the 17th International Conference on Computational Linguistics (COLING-ACL '98).*