An Analysis of Sentence Boundary Detection Systems for English and Portuguese Documents*

Carlos N. Silla Jr. and Celso A. A. Kaestner

Pontifical Catholic University of Parana
Rua Imaculada Conceicao, 1155 - 80.215-901
Curitiba - Parana - BRAZIL
{silla, kaestner}@ppgia.pucpr.br

Abstract. In this paper we present a study comparing the performance of different systems found in the literature that perform the task of automatic text segmentation in sentences for English documents. We also show the difficulties found to adapt these systems to make them work with Portuguese documents and the results obtained after the adaptation. We analyzed two systems that use a machine learning approach: MxTerminator and Satz, and a customized system based on fixed rules expressed by Regular Expressions. The results achieved by the Satz system were surprisingly positive for Portuguese documents.

1 Introduction

When dealing with tasks related to the automatic processing of documents like summarization, translation, etc. one of the procedures that frequently occur is the segmentation of the text in sentences. This task is usually included in the pre-processing stage, and uses a simple criterion, tagged documents, or one of the approaches found in the literature.

The systems found in the literature can be grouped in two classes: the ones that use fixed rules to identify what is and what is not a sentence, and the ones that use a machine learning approach. In this work we evaluate the performance of one customized system that uses fixed rules, and two systems that use a machine learning approach: MxTerminator [1] and Satz [2]. The first system uses templates based on Regular Expressions, considering the context where a punctuation mark appears, and will be refered to as RE (Regular Expressions) [3]. The MxTerminator uses a Maximum Entropy Model to detect the sentence boundaries, while Satz considers the context where a possible punctuation mark appears and can be used with any machine learning algorithm; in this work, it was used with the C4.5 classifier [4].

The remaining part of the article is divided as follows: section 2 presents a general view of the systems used for comparison and how they were adapted to Brazilian Portuguese; section 3 describes the methodology used in the experiments and presents the corresponding results for two sets of documents, in

* This research was supported by the Brazilian PIBIC-CNPq Agency.

English and Portuguese; and finally, in section 4 we draw the conclusions and perspectives of this work.

2 The Different Approaches to Detect Sentence Boundaries

In this section we present an overview of the three different systems used in this work.

2.1 RE SYSTEM

As a representative of the fixed rules approach, we used a system that encode the rules as regular expressions, and considers the context where each possible end of sentence occurs within the document. This system was chosen because it has achieved results close to the MxTerminator system in a dataset of the TIPSTER collection in recent experiments [3].

The system uses a database of regular expressions which denote chains that contain punctuation marks but don't indicate the end of a sentence, like abbreviations and other sequences like e-mails, www addresses, etc. The database of regular expressions is kept on a text file, which allows easy manipulation of the existing rules.

To identify the sentences, the system scans the text until it finds the first period (.); after that, it analyzes the preceding string; if this string matches some regular expression, then the system concludes that this is not an end of sentence and advances to the next period. If the preceding string doesn't match any regular expression, the system verifies the string after the period, which might be one of the special cases and need a different treatment. If the system doesn't find any matching regular expression for the current string, it concludes that the period indicates an end of sentence, and tags the text with the appropriate marks, in this case: <S> and </S>. The procedure is repeated until the entire document has been analyzed.

The following special cases are treated by the system:

- Decimal Numbers: the system verifies if what comes before the period is a number, and if it is, it also verifies if the word after the dot is a number. That way it can distinguish between: “. 2003.” and “.... US$ 50.25”.
- Parenthesis at the end of a sentence: one characteristic of the English language is that sentences like “(that night.)” are correct, unlike the Portuguese language, in which the correct form is “(that night).”.
- Ellipsis: The last special case treated by the system is related to the occurrence of ellipsis (“. . . ”). In this case, the system verifies the occurrences of successive dots until it finds the last one of them, which indicates the end of the sentence.

To adapt the system to the Brazilian Portuguese it was necessary to add 240 new regular expressions that basically denote abbreviations of the language.

Since a text file describes the regular expressions, it was easy, although time consuming, to adapt the system.

2.2 MxTerminator

The system MxTerminator was developed by Reynar and Ratnaparkhi[1] in the Pennsylvania University and uses an approach which is independent of language or text genre. MxTerminator uses a machine learning algorithm named Maximum Entropy Model to identify the sentences of a document.

From a Corpus with the sentences already identified (training set) the model learns to classify each instance of period (.), exclamation mark (!) and question mark (?) as elements that identify what is a sentence end and what is not.

The training process is robust and doesn't need any type of fixed rules or some other linguistic information, like part-of-speech frequencies, or even specific information about the genre or domain of the texts, because during the training the system creates a list of induced abbreviations. This list is obtained considering an abbreviation as every word in the training set that has a white space before and after its occurrence and contains a possible end of sentence symbol (.,!,?), but doesn't indicate an end of sentence.

The possible sentences of the document are identified by scanning the text for sequences of characters separated by a blank space (token) containing one of the symbols that indicate a possible end of sentence (.,?,!).

The token that contains the symbol which denotes a possible end of sentence is called Candidate. The system then uses the contextual information where each Candidate occurs. The contextual information is represented by a set of features like the prefix, the suffix, etc.

The main idea of the Maximum Entropy Model is that the probability of a certain class in this case - the sentence boundaries - in a given context, can be estimated by the joint probability distribution using a maximum entropy model.

To adapt the MxTerminator to the Brazilian Portuguese language the procedure was very simple, because the system uses, for training a text file of any size that must contain one sentence per line. Another interesting factor is that, besides the training files, no other type of information was needed. For this reason, the MxTerminator was considered the simplest of the analyzed methods to use and adapt and use in new language.

2.3 Satz

The Satz system was developed by Palmer and Hearst[2] and uses an approach that considers the context where each punctuation mark occurs; it can be used with any machine learning algorithm, and the original results were tested using Neural Networks [5] and the C4.5 Decision Tree Classifier [2].

The Satz system represents the context around a possible end of sentence symbol constructing a series of descriptor arrays, that represent an estimative of the part-of-speech distribution for each word.

Table 1. Mapped Classes

Grammatical Class	Mapped Tags
Miscellaneous	CUR; IN; OTHER; PDEN;
Noun	N; N/N; N/N/N; N/V
Verb	V; VAUX; V|PASS; V/V
Article	ART
Modifier	PCP; PCP/PCP; ADV; ADV/ADJ; ADV/ADV; ADV/KC; ADV/KS; ADV-KS-REL; ADV-KS-REL/ADV-KS-REL; ADV/PREP; ADV/PROADJ; ADV/PROSUB; ADJ; ADJ/ADJ; ADJ/V
Conjunction	KS; KS/ADJ; KC; CC
Pronoun	PROADJ; PROADJ/ADJ; PRO-KS; PRO-KS/PRO-KS; PROPESS; PROPESS/PROPESS; PROSUB
Preposition	PREP; PREP/ADJ; PREP/V
Proper Noun	PROP; NPROP; NPROP/NPROP
Number	NUM
Comma or Semicolon	, ;
Left Parentheses	QUOTEL; (
Right Parentheses	QUOTER;)
Non Punctuation Character	=; *; #;
Possessive	$
Colon or Dash	-; :; –;
Abbreviation	AB
Sentence Ending Punctuation	.; ._; !; ?;

The use of a part-of-speech estimative considers the context in which the word occurred rather than just the word itself. This is a unique aspect of the Satz system, and according to its authors is the main factor for the high efficiency of the system. The part-of-speech frequencies are stored in a lexicon. If a word is not present in the lexicon, a series of heuristics are used in order to define the corresponding frequency.

The context vector contains the descriptor arrays for each word surrounding the possible end of sentence, and is the input for the machine-learning algorithm. The output is used to indicate if a possible end of sentence mark corresponds to an end of sentence or not.

To adapt the Satz system to Portuguese it was necessary to re-implement the system, because the version available at the UCI Repository presents problems when dealing with accented characters, which are very common in Brazilian Portuguese. For example, a word like "agrícola" (agricultural) would be identified as two tokens: "agr" and "cola".

In order to re-implement the Satz system, we developed a Java-based version of the system that produces the descriptor arrays and integrated it with the Weka Data Mining Tool [6]. We used the J4.8 (which is a Java implementation of the C4.5 algorithm) in the tests. However, this procedure alone was not enough to adapt the system. We had to create a new lexicon using the part-of-speech information which is present in the Corpus. We also needed to map the Brazilian Portuguese Corpus tags to the 18 general categories of the system. Table 1 shows the tags mapped to each category.

3 Experiments and Obtained Results

In order to perform a comparison between the different systems using documents in English, we used one of the databases that contains news from the Wall Street Journal, which belongs to the TIPSTER document collection, from the Text Retrieval Conference (TREC - Reference number of the database: WSJ-910130).

The database contains 156 documents at different sizes, totalizing 3.554 sentences. To perform the experiments, each of the documents had their sentences detected and tagged manually. To evaluate the performance of the systems described in section 2, we also compared their results with the baseline proposed by Palmer[5]: where each sentence is obtained using the simple criterion period (.).

The results achieved by each system are presented in Table 2, where:

- "Precision" indicates the percentage of correctly classified sentences of the documents (Number of sentence endings correctly identified / Number of sentences identified);
- "Recall" indicates the percentage of correctly classified sentences of the documents regarding the number of sentences present in the original document (Number of sentence endings correctly identified / Number of sentences present in the original database);
- "F-measure" is a unique evaluation measurement, which combines precision and recall in a single metric: the harmonic mean.

The results achieved by the different systems show that although the RE system uses a fixed rule approach, its results are close to the other systems. This indicates that when the domain of the documents is well known, and no training Corpus is available, the use of a fixed rule system might be a good option.

In order to evaluate the performance of the systems with Portuguese Documents, we used a version of the Lacio-Web Corpus [7], that contains 21.822 sentences.

Table 2. Results achieved by the different systems in the TIPSTER (English) document collection

System	Precision	Recall	F-Measure
Baseline	30,29%	50,61%	37,89%
RE	92,39%	91,18%	91,78%
MxTerminator	91,19%	91,25%	91,22%
Satz	98,67%	85,98%	91,88%

Table 3. Robustness of the different systems using the Lacio-Web (Portuguese) document collection (uncustomized versions)

System	Precision	Recall	F-Measure
Baseline	85,40%	92,25%	88,69%
RE	91,80%	88,02%	89,87%
MxTerminator	94,29%	95,84%	95,05%
Satz	99,48%	98,81%	99,14%

To evaluate the robustness of each one of the systems, i.e. their performance using their original configuration, without any modification in the regular expressions nor any re-training, we used the Lacio-Web Corpus with unidentified sentences. The results achieved by each system are presented in Table 3. For the Satz system which is dependent on part-of-speech frequencies, the test was performed using only the original heuristics of the system.

Finally, in order to evaluate the performance of each of the systems when customized, they were adapted to Brazilian Portuguese by providing the needed information about the language. The MxTerminator was trained using the Lacio-Web Corpus using 10-fold cross-validation [8]. To the RE system we added 240 new regular expressions mostly containing abbreviations of Brazilian Portuguese words. The Satz system was also trained using 10-fold cross-validation, but a complete lexicon was created using the part-of-speech frequencies available within the Corpus. The results achieved by each system can be seen in Table 4.

Table 3 shows that the results achieved by the MxTerminator are surprisingly good, but not as impressive as the ones achieved by Satz. The results indicate that the MxTerminator and Satz are robust methods, although the results achieved by Satz are surprisingly positive. The results also indicate that the fixed rule approach, even in the form of regular expressions, is not well suited if the domain and genre of the texts are unknown.

Table 4 shows that after being adapted to Brazilian Portuguese, both machine learning methods improved their performance, except for the fixed rule approach. However, the results achieved by Satz even without the lexicon with part-of-speech information for all the words in the documents are outstanding.

Table 4. Results achieved by the different systems in the Lacio-Web (Portuguese) document collection (customized versions)

System	Precision	Recall	F-Measure
Baseline	85,40%	92,25%	88,69%
RE	91,80%	88,02%	89,87%
MxTerminator	96,31%	96,63%	96,46%
Satz	99,59%	98,74%	99,16%

4 Conclusions

In this work, we analyzed three different systems for the task of automatic text segmentation, in order to identify the sentence boundaries in a document. We performed experiments for both English and Portuguese documents, using a fixed regular expression rules system, the Satz decision tree approach and the MxTerminator maximum entropy approach. The best results were achieved by the Satz system: 91,88% of F-measure in the English document database; 99,14% in the Portuguese document database without retraining, using only heuristics and 99,16% in the same collection with retraining.

These results indicate that the part-of-speech frequencies, in the case of Brazilian Portuguese, are not as important as it is when working with English. This is explained by the fact that English sentence construction follows more restrictive construction patterns than the sentences in Portuguese, which is a Latin language. The adaptability and robustness of each system were also evaluated.

Although the RE system achieved results similar to the ones achieved by the other two systems, this was one specific case where the domain and genre of the text was well known. The MxTerminator achieved good results and was also the easiest system to adapt to the Brazilian Portuguese language. The Satz system had an outstanding performance showing that the part-of-speech information does not matter to identify the sentences in Brazilian Portuguese Documents.

Acknowledgments

We would like to thank Dr. Adwait Ratnaparkhi from the Pennsylvania University for sending us the MxTerminator system, and to Dr. Marti A. Hearst from the Berkeley University for kindly sending us the original files used by Satz.

References

1. Reynar, J., Ratnaparkhi, A.: A maximum entropy approach to identifying sentence boundaries. In: Proceedings of the Fifth Conference on Applied Natural Language Processing. (1997) 16–19
2. Palmer, D.D., Hearst, M.A.: Adaptive multilingual sentence boundary disambiguation. Computational Linguistics **23** (1997) 241–267
3. Silla Jr., C.N., Valle Jr., J.D., Kaestner, C.A.A.: Automatic sentence detection using regulares expressions (in portuguese). In: Proceedings of the 3rd Brazilian Computer Science Congress, Itajaí, SC, Brazil (2003) 548–560
4. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann (1993)
5. Palmer, D.D.: SATZ - an adaptive sentence segmentation system. Master's thesis (1994)
6. Witten, I.H., Frank, B.: Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Wiley-Interscience, San Francisco (1999)
7. Aluisio, S.M., Pinheiro, G.M., Finger, Nunes, M.G.V., Tagnin, S.E.: The lacio-web project: overview and issues in brazilian portuguese corpora creation. In: Proceedings of the Corpus Linguistics 2003. Volume 16. (2003) 14–21
8. Mitchell, T.M.: Machine Learning. McGraw-Hill (1997)