International Journal of Language and Linguistics
2013; 1(1) : 1-6
Published online May 2, 2013 (http://www.sciencepublishinggroup.com/j/ijll)
doi: 10.11648/j.ijll.20130101.11

SciencePG
Science Publishing Group

Rule-Based Sentence Detection Method (RBSDM) for Turkish

Özlem Aktaş*, Yalçın Çebi

Computer Engineering Department, Dokuz Eylul University, Izmir, Turkey

Email address:
ozlem@cs.deu.edu.tr (Ö. Aktaş), yalcin@cs.deu.edu.tr (Y. Çebi)

To cite this article:
Özlem Aktaş, Yalçın Çebi. Rule-Based Sentence Detection Method (RBSDM) for Turkish. *International Journal of Language and Linguistics*. Vol. 1, No. 1, 2013, pp. 1-6. doi: 10.11648/j.ijll.20130101.11

Abstract: The first process of generating a corpus, which is a representative of the language, is the determination of sentences, which is very complicated and hard to solve, but an important part of the corpus generation. Different approaches have been tried to find out sentence boundaries in some languages. In Turkish, the most known ways of determining sentence boundaries are using statistics and machine learning. In this study, to determine the sentence boundaries in contemporary Turkish, a rule-based method called "Rule-Based Sentence Detection Method for Turkish (RBSDM)" was developed by considering the agglutinative and rule based structure of Turkish. This method was tested on two different test sets generated by randomly selected columns from two Turkish newspapers. RBSDM determines end of sentences correctly and efficiently, about means of time and other costs, and provides success rate in a range of 99.60% and 99.80%.

Keywords: Linguistics, Natural Language Processing, Corpus, Turkish, Morphological Analysis, Sentence Boundary Detection

1. Introduction

"Natural Language" is the language naturally used by humans. Since 1940, researchers have worked for determining morphological specialties of natural languages. Because the computer technology had not developed at 1940 – 1950 yet, there were not enough data to be collected and processed in electronic environment. Since computer technology has been developed fast, more data has been collected and new technologies are developed using new researches.

Natural Language Processing (NLP) can be defined as the construction of a computing system that processes and understands natural language. The word "understand" in this definition can be clarified such as the following; "The observable behaviour of the system must make us assume that it is doing internally the same, or very similar, things that we do when we understand language" [1].

NLP processes work on a specialized database called "corpus" for any language. In NLP, there are two kinds of analyses used to generate and use a corpus: Morphological and Statistical Analysis [2]. Morphological analysis includes the investigation of the words' morphological status, such as determination of the sentence boundaries, investigation of the word types (verb, noun, adjective, etc.), and analyzing elements of the words (root, suffix or prefix). Statistical analysis can be done in two ways; on letters and words. The analyses applied on the letters are called "Letter Analysis"; for example, consonant and vowel positions, letter n-gram frequencies, relationship between letters such as letter positions according to each other. The analyses such as investigation of number of letters in a word, the order of the letters in a word, word n-gram frequencies, word orders in a sentence, are called "Word Analysis".

After the morphological analysis of a given text, a corpus can be created. The word "corpus" has different definitions such as:

- Corpus is a collection of linguistic data, either written texts or a transcription of recorded speech, which can be used as a starting-point of linguistic description or as a means of verifying hypotheses about a language [3].
- A collection of naturally occurring language text, chosen to characterize a state or variety of a language [4].
- But a corpus can be briefly defined as: "A special collection that is created from texts, used in Natural Language Processing area and allows all specialized processes, such as finding and separating the words quickly." [5]

The first step in the corpus generation, after the collecting texts process, is the "text segmentation". The main processes in the text segmentation are determining sentences and wordforms. Sentences generally end with known punctuations such as “.”, “…”, “!”, “?” in many languages. But, sometimes these punctuations are not used to indicate sentence boundaries, such as using “.” (dot) in e-mail addresses, web pages, abbreviations etc. Such ambiguities make the sentence boundary determination process very complex and hard to solve in all languages. Some ambiguities faced in English are as follows:

- She comes here by 5 p.m. on Saturday evening.
- www.tubitak.gov.tr is the web site of Scientific Research Supporting Association.
- My e-mail address is john@hacettepe.edu.tr.

As in all other languages, Turkish has such ambiguities as shown below:

Uluslar, bu ekonomik buhran sonucunda 2. Dünya Savaşı’nı yaşamıştır. (1)

(Nations faced with the 2.World War as a result of this economic crisis.)

Bu sezon kaybedilen maç sayısı 2. Dünya Kupası’na katılma şansı azalıyor. (2)

(The game number lost in this season is 2. The chance of attending to World Cup is decreasing.)

The “.” (dot) character was used for enumeration in the Sentence 1, and to indicate end of sentence in the Sentence 2. After this character, both of the sentences have the same word that begins with uppercase (“Dünya”). So, this is hard to say that “.” is used for enumerating or end of sentence.

In order to determine sentence boundaries for Turkish language correctly and efficiently, a rule-based sentence determination method (RBSDM) is developed and implemented by considering the agglutinative nature and rule-based structure of Turkish. In this study, this method is explained briefly and the results of the tests are given.

2. Rule-Based Sentence Detection Method for Turkish

Many available natural language processing tools do not perform a reliable detection of sentence boundaries since ambiguities appeared. The Rule Based Sentence Detection Method (RBSDM) was developed to solve the ambiguities faced in sentence boundary detection problems in Turkish. In this method, the rules and abbreviations in Turkish were used by the developed program to minimize the ambiguities. The input is a plain text file and the output of the program is an XML [6] tagged file. The main scheme for the RBSDM algorithm is given in Fig. 1.

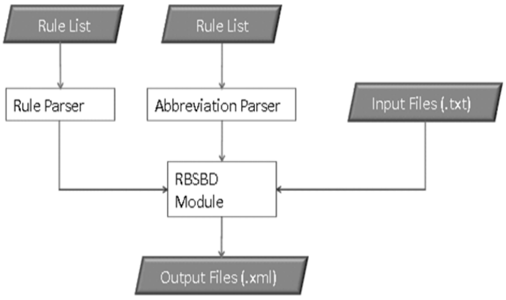

***Figure 1**. Main scheme for RBSDM*

The rules, which are essential for resolving the end of sentence (EOS), have been determined by the linguists and stored in an XML file, which is shown in Table 1.

***Table 1.** The rule list for sentence boundary detection in XML format*

Rules in XML format
<rule EOS=“True”> **L.U** </rule>
<rule EOS=“True”> **L.#** </rule>
<rule EOS=“True”> **?.'** </rule>
<rule EOS=“True”> **?.”** </rule>
<rule EOS=“True”> **?.(** </rule>
<rule EOS=“True”> **?.)** </rule>
<rule EOS=“True”> **?.-** </rule>
<rule EOS=“True”> **?./** </rule>
<rule EOS=“True”> **?./** </rule>
<rule EOS=“False”> **U.L** </rule>
<rule EOS=“False”> **L.L** </rule>
<rule EOS=“False”> **?.,** </rule>
<rule EOS=“False”> **#.L** </rule>
<rule EOS=“False”> **#.'** </rule>
<rule EOS=“False”> **#.”** </rule>
<rule EOS=“False”> **#.(** </rule>
<rule EOS=“False”> **#.)** </rule>
<rule EOS=“False”> **#.-** </rule>
<rule EOS=“False”> **#.,** </rule>
<rule EOS=“False”> **#.#** </rule>
<rule EOS=“False”> **#.U** </rule>

In the rule list, each rule consists of three characters. The first character indicates the first character of the word before punctuation mark that is used for end of sentence (“.”, “…”, “!”, “?”), second character is the punctuation mark itself, and the third character indicates the first character of the word after punctuation mark as shown in Fig. 2.

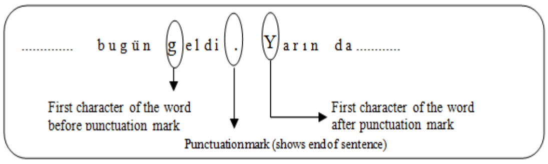
.... bugün geldi . Yarın da...........
First character of the word before punctuation mark
Punctuationmark (shows endof sentence)
First character of the word after punctuation mark

***Figure 2.** The characters used in the rules.*

The definitions of the characters used in the rule list are given in Table 2.

Table 2. *The definitions of the characters in the rule list*

Character	Meaning
.	End of Sentence punctuations (. … ! ?)
L	Lowercase
U	Uppercase
#	Number
?	Any character
-	Dash
,	Comma
(	Left parenthesis
)	Right parenthesis
/	Slash
‘	Single quote
“	Double quote

While processing text files, firstly the paragraphs are determined by “enter (‘\n’)” character at the end. After a paragraph is determined, characters are checked one by one if it is one of the punctuation marks of EOS defined in the rule file. In an ordinary situation, only punctuation marks might be good enough to determine the sentences. But, the structure of the Turkish is somehow complicated, and there are many ambiguities caused by the punctuation marks such as:

- Cumhuriyetimizin 75. yılı coşkuyla kutlandı.

(The 75th Anniversary of the Republic was celebrated with enthusiasm.)

- Tahta çıkan IV. Murat emirler yağdırdı.

(IVth Murat, who has gotten the throne, ordered commands.)

- Olimpiyatlar için uzun zamandır çalışan Ahmet koşuda 2. Uzun atlamada ise ancak 4. olabildi.(Ahmet, who had been working hard since a long time for the olympiads, has goı 2.place in running, but only 4. place in long jump.)
- Mehmet YILDIZ size uğradı.(A. Mehmet YILDIZ visited you.)
- Alfabenin ilk harfi A. Mehmet’e bunu öğretmeniz gerekiyor.

(The first letter of the alphabet is A. You have to teach this to Mehmet.)

In order to solve ambiguities, an additional rule file, in which abbreviations in Turkish was given, has been necessary. The abbreviation file has been taken from Turkish Linguistic Association [7] and accepted as is. This list has been also stored in XML format, as given in Table 3.

Users can easily add new rules and abbreviations to the XML files individually without knowing anything about the program structure. By using these rule lists, the obtained texts can be splitted into sentences and written in XML format to a file for further analysis.

In spoken texts, conversations are indicated by special character, “-”. This character causes an ambiguity, because of being used for bulleting. To solve this problem, a different control mechanism is developed. It was assumed that all bulleted texts belonged to one sentence and all lines were taken as one sentence after the punctuation marks “ : (colon)” and “ ; (semi-colon)”, which are used to indicate a bulleted list.

Table 3. *Example of abbrevation list in XML file*

Samples of Abbrevations in XML file		
<abbr>	A	</abbr>
<abbr>	AA	</abbr>
<abbr>	AAFSE	</abbr>
<abbr>	AAM	</abbr>
<abbr>	AB	</abbr>
<abbr>	ABD	</abbr>
<abbr>	ABS	</abbr>
<abbr>	ADSL	</abbr>
<abbr>	AET	</abbr>
<abbr>	HAVAŞ	</abbr>
<abbr>	HDD	</abbr>
<abbr>	zf	</abbr>
<abbr>	ZMO	</abbr>
<abbr>	zool	</abbr>
<abbr>	I	</abbr>
<abbr>	V	</abbr>
<abbr>	IX	</abbr>
<abbr>	X	</abbr>
<abbr>	XV	</abbr>
<abbr>	XXX	</abbr>

3. Application of RBSDM for Turkish

3.1. Test Sets

The test sets were generated by taking the columns from two Turkish newspapers. The real names of the newspapers and columnist were used during the tests but they were not written in this paper.

Two different test sets are generated to test the method. There are 10 different columnists and 20 columns of each from the Newspaper 1 (N1) in the Fist Test Set (TS1). In the Second Test Set (TS2), there are 10 different columnists and 20 columns of each from the Newspaper 2 (N2). The number of columns and sentences in the test sets are shown in Table 4.

3.2. Results

Developed algorithm was tested on two different test sets collected from the columns in two Turkish newspapers. Some paragraphs and splitted forms determined by the program are shown in Table 5.

The original texts were used in the tests without any corrections.

Conversation texts occurred in the articles were ignored as they belong to a spoken corpus and tagged as “spoken corpus sentence” (DLG: Dialog). This kind of sentences was asked to the user to determine their types. In the analysis, 78 undefined sentence blocks were asked to the user to determine that were either conversation or bulleted text, and tagged as DLG (dialog) or BL (bulleted list), to solve this kind of ambiguity situations.

The success rates were calculated by comparing the number of the sentences, which the program found, with the number of sentences in the original text, which were counted by linguists, and results were given in Table 6.

As given in Table 6, the program was tested on 17.412 sentences; 17.342 sentences were found correctly, only 64 sentences were resolved inaccurately. Some sentences that could not be resolved were shown in Table 7.

Table 4. *Numbers of columns and sentences in the test sets.*

Test Set I (TS1)		
Newspaper 1 (N1)		
Columnist	**Number of Columns**	**Number of Sentences**
C1	20	798
C2	20	1.746
C3	20	406
C4	20	834
C5	20	862
C6	20	697
C7	20	546
C8	20	1.252
C9	20	661
C10	20	532
Total	200	8.334
Test Set II (TS2)		
Newspaper II (N2)		
Columnist	**Number of Columns**	**Number of Sentences**
C1	20	582
C2	20	1.458
C3	20	546
C4	20	1.126
C5	20	1.316
C6	20	797
C7	20	972
C8	20	795
C9	20	634
C10	20	852
Total	200	9.078
Total Number of Sentences =	17.412	

Table 5 *Sample paragraphs and splitted forms*

Original Text	**Parsed Sentences**
Biliyor musunuz, geçenlerde 'Çırağan Palace Hotel Kempinski'nin Tuğra Restaurant'ı 'Dünyanın en iyi 10 mutfağı' arasına girdi.	<P I="**0**"> <S Index="**0**">Biliyor musunuz, geçenlerde 'Çırağan Palace Hotel Kempinski'nin Tuğra Restaurant'ı 'Dünyanın en iyi 10 mutfağı' arasına girdi. </S>
Düşünün 7 milyar insanın yaşadığı koca dünya, binlerce otel, lokanta ve...ilk on arasında bizim Tuğra Restaurant... Üstelik dünyanın en saygın uzmanlarından oluşan jüri tarafından seçildi.	<P I="**2**"> <S Index="**0**">Düşünün 7 milyar insanın yaşadığı koca dünya, binlerce otel, lokanta ve...ilk on arasında bizim Tuğra Restaurant.... </S> <S Index="**1**">Üstelik dünyanın en saygın uzmanlarından oluşan jüri tarafından seçildi. </S>
O yemekler, o müzik ve Boğaz... Kendinizi kesinlikle zaman tüneline sokar, en azından 150 yıl öncesine gidersiniz. Kendinizi 'sultan' sanabilirsiniz.	<P I="**4**"> <S Index="**0**">O yemekler, o müzik ve Boğaz.... </S> <S Index="**1**">Kendinizi kesinlikle zaman tüneline sokar, en azından 150 yıl öncesine gidersiniz. </S> <S Index="**2**">Kendinizi 'sultan' sanabilirsiniz. </S>
P: Paragraph S: Sentence	

Table 6. *Success rate*

Sample Text	**# of Sentences**	**# of Sentences Detected True**	**# of Sentences Detected False**	**Success Rates (%)**	**Success Rates – Except Misspellings (%)**
Columns in NP1	8.334	8.306	28	99.66	99.80
Columns in NP2	9.078	9.036	36	99.60	99.76
TOTAL	**17.412**	**17.342**	**64**	**99.63**	**99.78**

The reason of the false resolving sentence boundaries for the sentences 1, 2 and 4 in Table 7 was the punctuation mark “…” (three dots). Since this punctuation mark can be used in the middle of the sentence, it causes an ambiguity. The third sentence was resolved false because of the wrong usage of the punctuation mark “ ’ “ (apostrophe). The quotation mark is used for the words that belong to the sentence which is written in double quotes and needed to be quoted again, and also punctuation marks do not be used in the text that is written in the single quotes [7].

Table 7 Sample of false splitted sentences

Newspaper	Original Sentence	Parsed Sentences
NP1	Devamı şöyle: Millî Eğitim Bakanı’nın imzasıyla tüm okullara gönderilen genelgede... deniliyordu. *(It continues such that: It is said ... in the notice that was signed by the Head of the Department of Education and sent to all schools.)*	<Sentence Index=“1”>Devamı şöyle: Millî Eğitim Bakanının imzasıyla tüm okullara gönderilen genelgede.</Sentence> <Sentence Index=“2”> deniliyordu. </Sentence>
	Ama, düz yolda gitmeyi bilmeden, bir elinizde telefon, ağzınızda sigara... bu bir. *(But, there is a telephone in one of your hands; a cigarette in your mouth without knowing to go on the straight road… this is first.)*	<Sentence Index=“4”>Ama, düz yolda gitmeyi bilmeden, bir elinizde telefon, ağzınızda sigara.</Sentence> <Sentence Index=“5”> bu bir.</Sentence>
NP2	Telekom Genel Müdürü Mehmet Ekinalan her fırsatta Telekom'un 'muhteşem!' faaliyetlerini öve öve bitiremiyor. *(Mehmet Ekinalan, who is the Manager of the Telecommunication Department, praises the ‘magnificent!’ activities of the department all the time.)*	<Sentence Index=“0”>Telekom Genel Müdürü Mehmet Ekinalan her fırsatta Telekom'un 'muhteşem!</Sentence> <Sentence Index=“1”>' faaliyetlerini öve öve bitiremiyor.</Sentence>
	Tetikçileri var, devlet içinde devlet olmuşlar, devlet adına çalışıyorlar, devlet adamlarıyla ahbap çavuşlar.. şu, bu! *(They have triggermen, create a state in the state, work for the government, good friends with government… this, that!)*	<Sentence Index=“0”>Tetikçileri var, devlet içinde devlet olmuşlar, devlet adına çalışıyorlar, devlet adamlarıyla ahbap çavuşlar.</Sentence> <Sentence Index=“1”> şu, bu!</Sentence>

4. Conclusion

Proposed rule-based method (RBSDM) determines boundaries of sentences in Turkish with pre-determined rules and abbreviation lists in an efficient way, and the results are successive. The well-known highest success rate for Turkish sentence boundary method was denoted by Kiss and Strunk [8] about multilingual sentence boundary detection including Turkish, and it was measured as 98.74% mean value of all languages’ test results. It was tested on the METU Turkish Corpus [9], which only included Turkish newspaper Milliyet. Also, the success rate of the study by Dinçer and Karaoğlan [10], which was developed for only Turkish language, was measured as 96.02%.

The RBSDM was tested on two different test sets generated by randomly selected columns from two Turkish newspapers, which included misspellings and ambiguities.

The success rates were determined as 99.60% (99.76% without misspellings) and 99.66% (99.80% without misspellings) in these test sets. The average success rate of the algorithm was 99.78% if misspellings were discarded. If the sentences are written in formal way and with no spelling faults, the rule-based sentence boundary detection method would be more efficient and accurate.

Some ambiguities such as abbreviations and enumerations were solved by this rule-based method. The ambiguities that could not be solved by this method may be solved by using machine learning and statistical analyses for Turkish. The other parts in generating corpus, such as finding word types, determining root and suffixes of the lemmas can be attached into this structure easily because of its readability, flexibility and understandability, and an efficient corpus can be created.

Since the language structure is commonly the same, this algorithm can be easily adapted and used for other Turkic languages such as Uzbek, Kazak, Turkmen, Azeri and Kirgiz Turkish by only changing the rule and abbreviation lists.

References

[1] Z. Güngördü, “A lexical-functional grammar for Turkish”, MSc Thesis, Computer Engineering Department, Bilkent University, Ankara-Turkey, 1993.

[2] C. E. Shannon, “Prediction and Entropy of Printed English”, The Bell System Technical Journal, vol. 30:1, pp. 50-64, 1951.

[3] D. Crystal, A Dictionary of Linguistics and Phonetics, 3rd Edition, Blackwell, 1991.

[4] J. Sinclair, “Corpus Concordance”, Collocation, OUP, 1991.

[5] Varliklar, O. Developing a Method to Determine Root and Suffixes for Turkish Words to Generate Large Scale Turkish Corpus. M.Sc. Thesis, Dokuz Eylul University Graduate School of Natural and Applied Sciences Computer Engi-

neering Department, Izmir - Turkey, 2005.

[6] Boye, J. “XML, What’s in it for us?”, article published in www.irt.org, 1998.

[7] Ş. H. Akalın, R. Toparlı, Yazım Kılavuzu, Türk Dil Kurumu Yayınları, 24th Edition, Ankara, 2005.

[8] T. Kiss, J. Strunk, “Unsupervised Multilingual Sentence B oundary Detection”, Computational Linguistics vol. 32:4 pp,. 485-525, 2006.

[9] B. Say, D. Zeyrek, K. Oflazer, U. Ozge, “Development of a Corpus and a Treebank for Present-day Written Turkish”, Proceedings of the Eleventh International Conference of Turkish Linguistics, ICTL, Ankara, Turkey, 2002.

[10] T. Dinçer, B. Karaoğlan, “Sentence Boundary Detection in Turkish”, Advances in Information Systems Proceedings: Third International Conference, Izmir-Turkey, pp. 255, 2004.