Volume 6, Issue 2, February 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Detecting Sentence End in Marathi Text

Nagmani Wanjari*, Gauri M. Dhopavkar
Department of Computer Science and Engineering, YCCE, Nagpur,
Maharashtra, India

Abstract— This paper reports the development of a system that is been designed with the aim of detecting the sentence end in Marathi text. For this task of detecting correct sentence end a rule based approach is used. This approach works on the basis of rules designed keeping in mind the pattern and structure followed by Marathi language. This Sentence boundary detection system which is under development at present is able to identify abbreviations and decimal numbers present in the text and it is also capable of identifying the position of every occurrence of the punctuation mark period '.' in the input text which is considered as most ambiguous punctuation mark.

Keywords— Sentence Boundary Detection, Natural Language Processing, Inflection rule, SOV, Rule based;

I. INTRODUCTION

Sentence Boundary Detection is an integral part for many natural language processing applications. A sentence in English language is generally denoted by punctuation marks such as '.', '!', '?'. The task is made difficult due to the ambiguity of these punctuation marks.

For example consider following sentences,

"Mr. Depp lived near St. Martin Block in Washington D.C."

"Surprise!!", shouted his friends .

The above mentioned sentences depict the ambiguities of the punctuations clearly. The first sentence can be considered a good example for depicting the ambiguities caused by the period. The period can either indicate a salutation or abbreviation or sentence end. The exclamation point in the second example indicates surprise element and not the sentence end.

A. Sentence boundary detection in Marathi

The task for detecting the sentence end is made more complicated due to various facts. For example in English language the pattern near the sentence end can be denoted as "word punctuation capital letter word", however in Marathi language uses Devanagari rather than the English alphabet so does not follow the concept of capital letters. It follows subject object verb (SOV) concept, i.e. the sentences in this language mostly ends with a verb.

For example:

"विजयने एम. बी. ऐ. झाल्यावर डॉ. सी. के. साळवे रूग्णालयात इंटर्नशीप घेतली."

" "मला आश्र्य वाटते ....", ती निराशेने वकत्वय पूर्ण करण्यासाठी प्रयत्न करीत म्हणाली."

Also the form of the verb changes with the gender, number of persons, or tense in the sentence. This too further complicates the task of detecting the sentence end for Marathi text.

II. RELATED WORK

A large amount of research has been done for detecting the sentence end in English and various other foreign languages. For English language various tools exist that is capable of identifying the sentence end, for example various tools like tokenizer, etc [9]. Some of these tools could also be implemented for various other foreign languages such as for Brazillian Portuguese, Spanish [3]. Few of these tools can also identify slangs present in online text. Some amount of work has also been done for few Indian languages such as Malayalam, Bengali, Kannada. This task of detecting a sentence end for Kannada language is tackled using rule based approach by Mona Parakh et al. in [1]. In [2] two techniques for detecting the sentence end are compared for Kannada language. Not much work has been done towards detecting the sentence end for Marathi language. For Bengali language Aniruddha Ghosh et al., presents a syntactic rule based model for identifying boundary of clause and uses Conditional Random Field based statistical model for identifying types of the clause in Bengali Language [10]. For Malayalam language a system to identify clause using machine learning approach had be developed by Sobha, Lalitha Devi et al. [11]. For Marathi language small amount of work is available. Such as S. B. Kulkarni et al. compares and states the differences in Marathi and English language encountered during translation [5].Charugatra Tidke et al. presents different inflection rules for English to Marathi translation [7].

III. PROPOSED SYSTEM

We propose a system that would be able to perform the task of sentence boundary detection for Marathi language. The system proposed is based on certain rules designed keeping in mind the grammar and the pattern followed by Marathi language. We have observed certain pattern and rule followed by the language. Marathi language allows a sentence to end with post position. For example-

त्याणे जेवण केले का?

मी आधी त्याना पाहिले आहेत का?“

ते अजूनही पुस्तक आणायला निघाले नाही का ?

As mentioned before Marathi is a verb final language i.e. there is a big probability that the sentence might end with a verb. So the pattern followed by Marathi language can be considered as “verb/postposition followed by punctuation followed by other part of speech” then the punctuation (‘.’,’?’,’!’) used indicates the sentence end. The rules for the system has been designed keeping in mind such occurrences and exception that might be encountered.

The flow of the system is given in the fig 1–

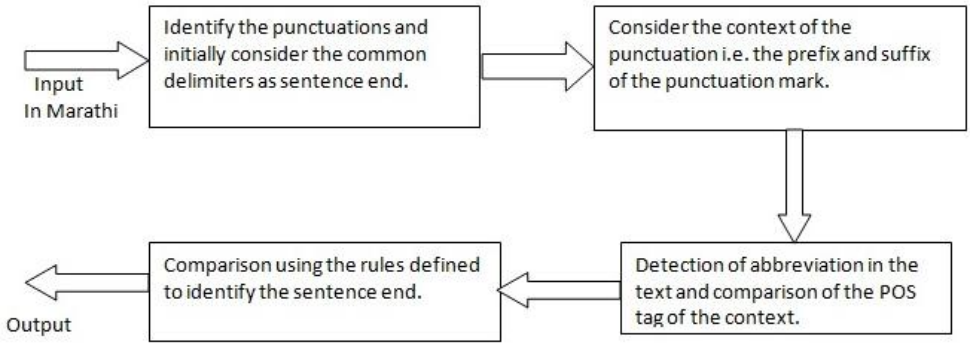

Fig 1: Flow of the system

IV. RESULT

Till now the system is capable of successfully differentiating between and maintain the count of ambiguous and unambiguous punctuation marks. After this the system then differentiates the ambiguous occurrence of the period from that occurrence of the periods ‘.’, which are unambiguous. As mentioned earlier that the system assumes that most ambiguities are caused by the ambiguous nature of the period. So we make it our priority to solve this problem and then procced further for other punctuation marks.

The screen shot below depicts the above mentioned procedure.

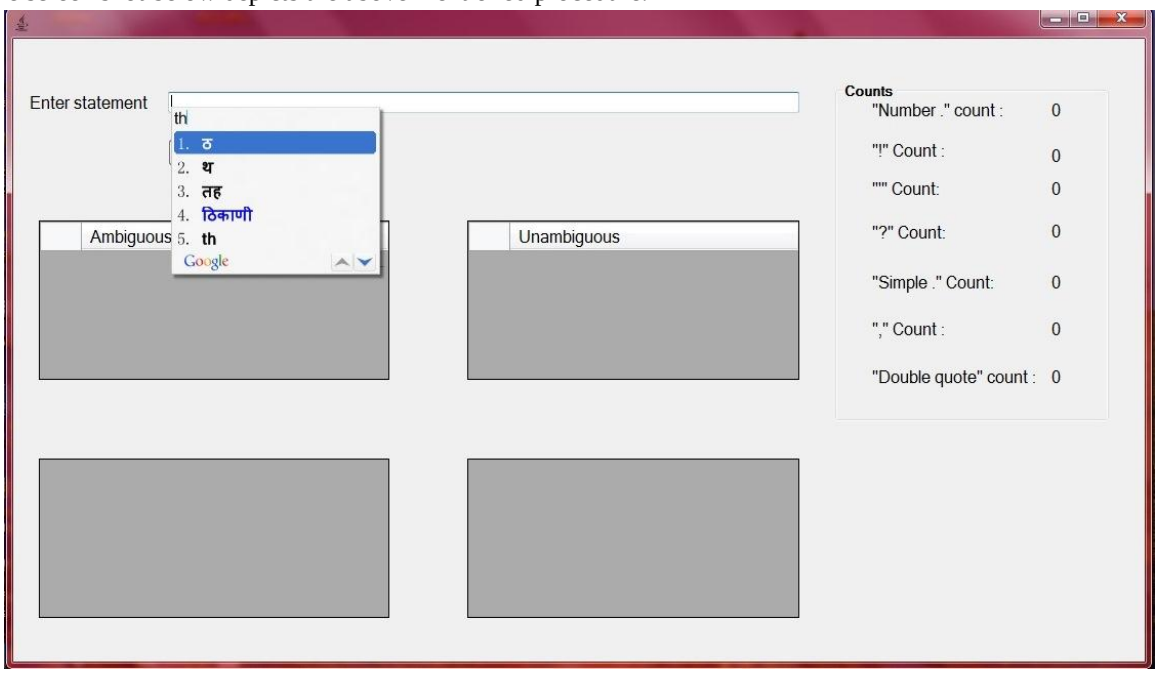

Fig 2: Input using google input tool.

Fig 2 shows the initial GUI used and how the input is taken using google input tool. Google input tool is a transliteration tool that is based on dictionary based phonetic transliteration approach.[]

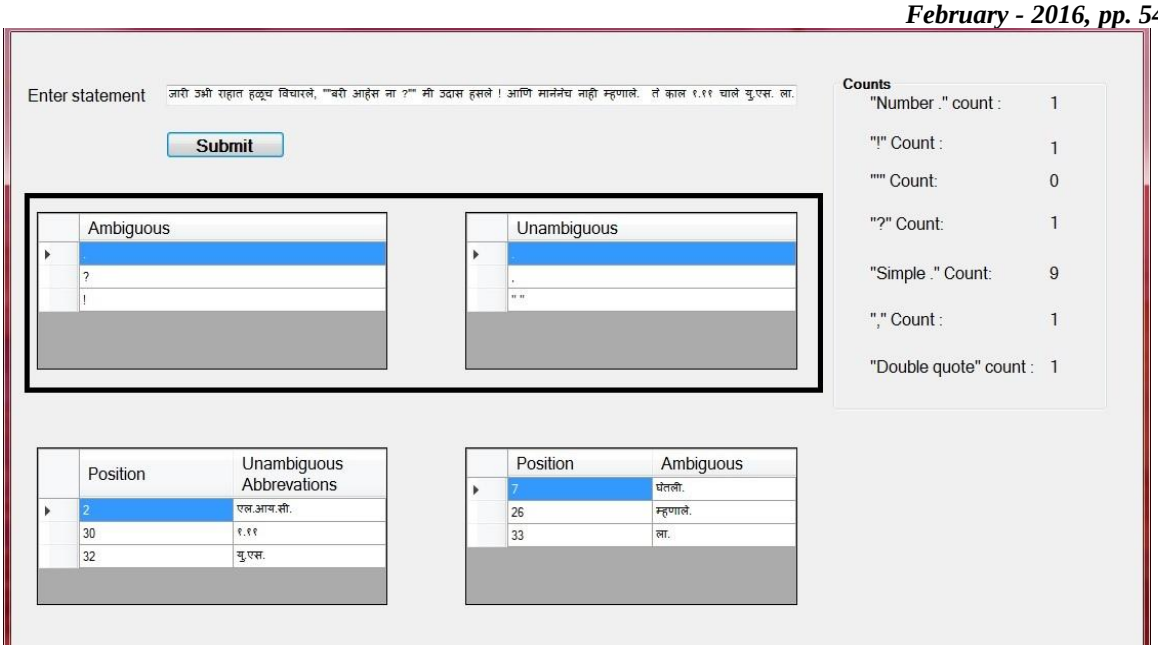

Fig 3: Screen shot differentiating between ambiguous and unambiguous punctuation mark.

Fig 3 is the screen obtained after processing of the text. After processing the system is able to differentiate between ambiguous and unambiguous punctuation mark (as seen in the marked up area) and is able to maintain the count of the every punctuation mark encountered in the input text and also the numbers encountered in the text. The system is also able to identify the abbreviations and the decimal numbers that are encountered in the input text. And also able to correctly report the position at which these are present. This can be seen in Fig 4. Priority has been given to disambiguate the punctuation mark period '.' First as it used to indicate abbreviations, decimal numbers, initials, ellipses and also sentence end. So, the system designed at this stage also pinpoint the other occurrences of the punctuation mark period '.'.

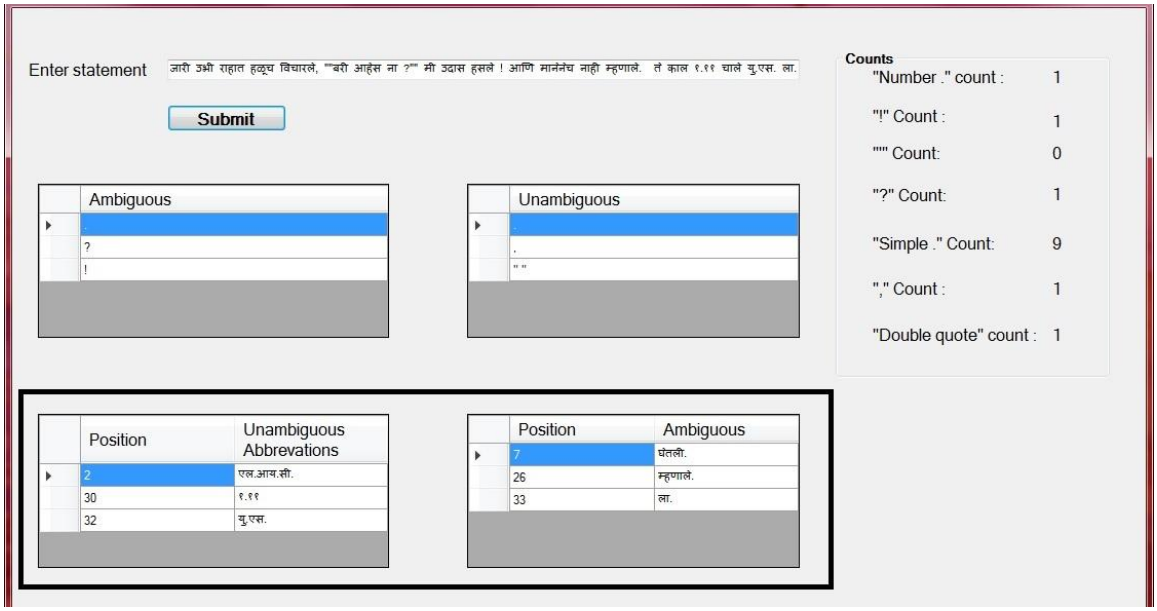

Fig 4: Screen shot indicating the identification of the abbreviations and decimal numbers.

Priority has been given to disambiguate the punctuation mark period '.' First as it used to indicate abbreviations, decimal numbers, initials, ellipses and also sentence end. So, the system designed at this stage also pinpoint the other occurrences of the punctuation mark period '.' (as seen in Fig 4).

V. CONCLUSION

By first tackling the symbol that is considered to be more ambiguous than the other punctuation mark, the task of detecting the sentence end in the Marathi text becomes much easier. Identifying the abbreviations and the decimal numbers present in the input text makes the punctuation mark period '.' much less ambiguous. The further task involves fully disambiguating the punctuation mark period '.', and resolving the ambiguity of the period '.', helps to an extent in disambiguating the ambiguities related to the punctuation marks exclamation mark '!', and question mark '?'.

REFERENCES

[1] Mona Parakh, Rajesha N., Ramaya M. "Sentence Boundary Disambiguation in Kannada Texts", Special Volume: *Problems of Parsing in Indian Languages* , 11 May 2011.

[2] Deepamala.N , Dr. Ramakanth Kumar.P , Surendra.H, ""Kannada Sentence Boundary Detection using Rule based and Maximum Entropy Methods", searchdl.org/public/book_series/AETS/7/173.pdf

[3] David D. Palmer, Marti A. Hearst, "Adaptive Multilingual Sentence Boundary Disambiguation", *Association for Computational Linguistics, ACL*; 23(2): p.241-269; 1997

[4] Tibor Kiss, Jan Strunk, "Unsupervised Multilingual Sentence Boundary Detection", *Association for Computational Linguistics*; 32; 2006

[5] S. B. Kulkarni, P. D. Deshmukh, K.V. Kale, "Syntactic and Structural Divergence in English-to-Marathi Machine Translation", International Symposium on Computational and Business Intelligence, 2013.

[6] Jeffery C. Reynar, Adwait Ratnaparkhi, "A Maximum Entropy Approach to Identifying Sentence Boundaries.", In: *Proceedings of the fifth conference on Applied NLP*; Washington D.C.; 1997

[7] Charugatra Tidke, Shital Binajakya, Shivani Patil, Rkha Sugandhi, "Inflectional rules for English to Marathi translation", *International Journal of Computer Science and Mobile Computing, ISSN 2320–088X, IJCSMC, Vol. 2* Issue, pg.7 – 18; 4, April 2013.

[8] Available: https://wordnet.princeton.edu/

[9] Jonathan Read, Rebecca Dridan, Stephan Oepen, Lars Solberg., "Sentence Boundary Detection: A long Solved Problem?" *In Proceedings of COLING , Posters,* pages 985–994, Mumbai, India; 2012.

[10] Aniruddha Ghosh, Amitava Das, Sivaji Bandyopadhyay, "Clause Identification and Classification in Bengali", *Proceedings of the 1st Workshop on South and Southeast Asian Natural Language Processing (WSSANLP),* the 23rd International Conference on Computational Linguistics (COLING), Beijing, pages 17–25, August 2010.

[11] Sobha Lalitha Devi, Lakshmi S, "Malayalam Clause Boundary Identifier: Annotation and Evaluation", *The 4th Workshop on South and Southeast Asian NLP (WSSANLP), International Joint Conference on Natural Language Processing, Nagoya*; pages 83–90; Japan, 14-18 October 2013.

[12] Available: https://en.wikipedia.org/wiki/Google_transliteration.

