Sentence Boundary Detection for Social Media Text

Dwijen Rudrapal **Anupam Jamatia** **Kunal Chakma**
Dpt. of Computer Science and Engineering
National Institute of Technology
Agartala, Tripura, India
{dwijen.rudrapal, anupamjamatia, kchax4377}@gmail.com

Amitava Das
Indian Institute of Information Technology
Sri City
Andhra Pradesh, India
amitava.das@iiits.in

Björn Gambäck
Dpt. of Computer and Information Science
Norwegian University of Science and Technology
Trondheim, Norway
gamback@idi.ntnu.no

Abstract

The paper presents a study on automatic sentence boundary detection in social media texts such as Facebook messages and Twitter micro-blogs (tweets). We explore the limitations of using existing rule-based sentence boundary detection systems on social media text, and as an alternative investigate applying three machine learning algorithms (Conditional Random Fields, Naïve Bayes, and Sequential Minimal Optimization) to the task.

The systems were tested on three corpora annotated with sentence boundaries, one containing more formal English text, one consisting of tweets and Facebook posts in English, and one with tweets in code-mixed English-Hindi. The results show that Naïve Bayes and Sequential Minimal Optimization were clearly more successful than the other approaches.

1 Introduction

Sentences are basic units of the written language — just as words, phrases, and paragraphs — and detecting the beginning and end of sentences, or sentence boundary detection (SBD) is an essential prerequisite for many Natural Language Processing (NLP) applications, such as Information Retrieval, Machine Translation, Sentiment Analysis, and Document Summarization. For formal texts, sentence boundary detection has been considered a more or less solved problem since the 1990s, but the proliferation of social media has added new challenges to language processing and new difficulties for SBD, with state-of-the-art systems failing to perform well on social media, due to the coarse nature of the texts.

In spite of its important role for language processing, sentence boundary detection has so far not received enough attention. Previous research in the area has been confined to formal texts only, and either has not addressed the process of SBD directly (Brill, 1994; Collins, 1996), or not the performance related issues of sentence boundary detection (Cutting et al., 1992). In particular, no SBD research to date has addressed the problem in informal texts such as Twitter and Facebook posts.

The growth of social media is a global phenomenon where people are communicating both using single languages and using mixes of several languages. The social media texts are informal in nature, and posts on Twitter and Facebook tend to be full of misspelled words, show extensive use of home-made acronyms and abbreviations, and contain plenty of punctuation applied in creative and non-standard ways. The punctuation markers are also often ambiguous in these types of texts — in particular between actually being used as punctuation and being used for emphasis — creating great challenges for sentence boundary detection.

When analysing texts from Facebook and Twitter, we find a rich variety of practices to mark the end of a sentence, for example, with emoticons (such as ‘:)’, ‘:(’, ‘:D’, ‘♡’, etc.), with several consecutive punctuation markers (e.g., ‘!!!!’ and ‘????’) or several sequences of multiple periods (e.g., ‘... ...’ or ‘..... .....’), and by mixing multiple punctuations in different combinations (e.g., ‘!?’ or ‘...?’), as in the following three examples.

(1) *Rick Ross b-day party at K.O.D $200 a head ... .. Im in the building ... ...*

(2) *after you can walk again?! whaaat, your leg's broken????! LOL. makes no sense.*

(3) *@kirkfranklin happy birthday !!!! I'm so glad you were born !*

In this work we concentrate on resolving the ambiguity of sentence end markers for social media text, and have carried out several experiments to detect sentence boundaries, using both rule-based and machine learning-based strategies.

The rest of the paper is organized as follows: In Section 2, we discuss some previous work on sentence boundary detection, which has mostly been on formal text. Corpora collection and annotation is discussed in Section 3, while Section 4 describes the rule-based and machine learning-based approaches used. Evaluation of the experimental results and error analysis are discussed in Sections 5 and 6, respectively. Finally, Section 7 concludes and points out some directions for future research.

2 Related Work

The approaches that have been taken to sentence boundary detection can on a general level be categorized as either machine learning-based or rule-based. State-of-the-art machine learning approaches perform well with an accuracy of around 99% on formal texts such as news-wire and financial newspaper texts, whereas rule-based approaches on the same type of texts typically report around 93% accuracy. Our approach investigates the use of different punctuations and patterns for marking the end of sentence in social media text and is based on comparing three machine learning algorithms — Conditional Random Fields, Naïve Bayes and Sequential Minimal Optimization — to a rule-based system, also with the rule-based strategy showing lower accuracy than the machine learning approaches, as described in the next sections. First, however, we will discuss some relevant previous work.

Riley (1989) proposed one of the earliest feature-based machine learning approaches, mainly for investigating the occurrences of periods as sentence end markers, and reported 99.8% accuracy on 25 million words of AP news-wire and the one million words tagged Brown corpus (Kučera and Francis, 1967).

Reynar and Ratnaparkhi (1997) presented a maximum entropy model trained on an annotated corpus. The model classifies each occurrences of '.', '?' and '!' as a valid or invalid sentence boundary. They proposed a domain-specific system which uses knowledge about the structure of English financial newspaper text, and a domain-independent system which uses the structure of English text genres. They report accuracy of 98.8% and 98.0%, for the domain-dependent and domain-independent system, respectively.

The *SATZ* system (Palmer and Hearst, 1997) uses a lexicon with part-of-speech probabilities and a feed-forward neural network. The system represents the context surrounding a punctuation mark as a series of vectors of probabilities. This probability of a word is used as input to the neural network to disambiguate sentence boundaries. *SATZ* reached an accuracy of 98.9% on Wall Street Journal (WSJ) text. Integrating the classifier with a heuristics-based approach increased the accuracy to 99.5%.

Gillick (2009) described a statistical system which focuses on full stops as candidate boundaries only. The system employs Support Vector Machines as a learning framework and reported accuracy rates of 99.75% on WSJ and 99.64% on the Brown corpus.

The *Punkt* system (Kiss and Strunk, 2006) introduced an unsupervised approach based on abbreviation identification. *Punkt* identifies abbreviations using three traits such as if an abbreviated word preceding a period and the period itself form a close bond. Abbreviations have the tendency to be rather short and often contain word-internal periods. *Punkt* achieved an accuracy of 98.35% on WSJ data and 98.98% on Brown texts.

Wong and Chao (2010) presented an incremental algorithm for sentence boundary detection using different features like if a word is capitalized, word length, potential punctuation, and the status of sign such as '$' and numbers, extracted from the trigram contexts of a training corpus. They report 99.98% accuracy on the Brown corpus, and slightly lower on the Tycho Brahe and Hoje Macau corpora with 96.51% and 98.73%, respectively.

The *iSentenizer-*μ system (Wong et al., 2014) presented an incremental tree learning architecture to detect sentence boundaries in a mixture of different text genres and languages. This model also utilized features derived from the trigram con-

texts of a training corpus. The system performance showed accuracy of 99.8%, 99.81% and 99.78% on the WSJ, Brown and Tycho Brahe corpora, respectively.

Rule-based to automatic sentence boundary detection have not been as succesful and hence also not as common as the various machine-learning based strategies. Grefenstette and Tapanainen (1994) describe one of the earliest rule-based systems, using a set of rules in the form of regular expressions to distinguish periods used in abbreviations, numbers, email and web addresses from those used as end of sentence markers. The system addressed periods only as sentence boundaries and showed an accuracy of 93.78%.

A more recent rule-based approach by Mikheev (2002) included a set of rules to detect capitalized words, abbreviations and other sentence termination punctuation such as periods, question marks, exclamation marks and semicolons. The system reached an accuracy of 99.55% on WSJ text and 99.72% on the Brown corpus, respectively.

3 Data Collection

In order to investigate the performance of sentence boundary detection on social media text and compare it to performance on more formal text, we have used three different text collections.

The first collection is a mixture of 3,000 tweets and Facebook posts in English, which we will call the"Social Media Corpus" (SMC). The tweets come from CMU's ARK Twitter data[1] (1,000 tweets) and Ritter[2] (1,000 tweets), while1,000 Facebook posts were randomly collected from campus-related billboard postings at different U.S. universities (CMU, Cornell, MIT, UCB, etc.).

The second collection is English-Hindi code-mixed twitter data (554 tweets) from the NITA corpus described by Jamatia et al. (2015), and the third collection consists of formal English text from the Brown corpus (1,125 messages), for comparison.

Utterance boundaries were manually inserted into the messages of the SMC by two human annotators. Due to the coarse nature of the corpus, also human annotators sometimes face difficulties in annotating sentence boundaries. Initially, the annotators agreed on only 61% of the utterance breaks. After discussions and corrections, the agreement between the annotators reached 80.06% on average over the entire corpus.

Corpus	Agreement
CMU ARK	92.3
Ritter	71.7
Facebook	76.2
NITA	92.5
Brown	99.8

Table 1: Inter annotator agreement (%) on the SMC, NITA and Brown corpora

Details of the inter-annotator agreement values for annotation on SMC are given in Table 1, where an agreement on an utterance implies that both annotators had segmented it into the same number of sentences, all with the same word lengths. The resulting corpus has in total 6,444 sentences, of which 3,522 come from Twitter and 2,922 from Facebook posts.

The NITA and Brown corpora excerpts were also annotated by two humans, manually inserting sentence boundaries into the messages. The inter-annotator agreement ratios were 92.5% and 99.8% for the NITA corpus and the Brown corpus, respectively. The resulting segmented NITA corpus has 1,225 sentences and the Brown corpus part has 1,000 sentences.

4 System Description

To detect the sentence boundaries of social media texts, we first need to resolve the issues of ambiguous punctuation used for sentence boundaries. Punctuation markers such as period (.) , semicolon(;), comma(,), vertical-bar (|), tilde(∼), etc., are not only used as sentence separators. The period (.) is the most ambiguous separator: In our Twitter/Facebook social media corpus there are 5,664 periods and out of those, manual inspection showed that only 3,168 (i.e., 55.93%) were actually used as sentence end markers.

To mitigate the ambiguous nature of the period, the entire corpus was tokenized using the CMU tokenizer[3] originally developed by O'Connor et al. (2010) and specially designed for Twitter-related

[1] www.ark.cs.cmu.edu/TweetNLP

[2] www.github.com/aritter/twitter_nlp

[3] www.ark.cs.cmu.edu/TweetNLP

1. **IF** the current token matches one of the patterns shown in Table 2
   AND the current token is in the last token position of the string
   THEN the current token is a sentence end marker.

2. **IF** the current token **AND** the previous token match any of the patterns shown in Table 2,
   AND the next token is any punctuation other those in Table 2
   THEN the next token is a sentence end marker.

3. **IF** the current token **AND** both the two tokens before it match any of the patterns shown in Table 2
   AND the next token is a word-token (other than a symbol)
   THEN the current token is a sentence end marker.

Figure 1: Rule-based sentence boundary detection algorithm

English text. This tokenizer is a sub-module of CMU Twitter POS tagger developed by Gimpel et al. (2011). It was observed that tokenization resolved a large portion of the ambiguous punctuation, e.g., 'Mr.'; 'U.S.A'; 'http://bit.ly/h1XIyQ'; '1019.0', etc.

After tokenization, we have developed two alternative approaches to automatically detect sentence boundaries, one rule-based approach and one approach based on applying machine learning. Three different machine learning algorithms were tested for automatic sentence boundary detection: Conditional Random Fields (CRF), Naïve Bayes (NB), and Sequential Minimal Optimization (SMO). The details of each approach will be discussed in following subsections.

4.1 Rule-Based Approach

A rule-based system was developed to automatically identify sentence boundaries in noisy social media text. The basic system splits Twitter and Facebook posts into sentences based on the punctuations that are normally used as sentence end markers, such as '.', '?', '!' and 'End-of-line'. 'End-of-line' will be used in this paper as a term refering to the cases where a social media post ends without any punctuation or any specific end marker; something which is quite common in informal texts, making the SBD task significantly more difficult.

The rule-based system correctly splits 2,468 posts of the in total 3,000 in SMC into 4,832 sentences with an F-measure of 72.34%, while the system was unable to split 532 of the posts. After observing the pattern of un-split posts in the corpus, we found that in addition to normal sentence end markers, a few other punctuation patterns have been used as sentence end, as shown in Table 2.

Considering the above classes of sentence end markers, the rules shown in Figure 1 were incorporated into the basic system, in order to increase its performance.

After applying the above rules to the corpus, the rule-base system splits 2,604 posts into 5172 sentences withan F-measure of 78.72%. The remaining non-split 396 posts (13.2%) were analysed in detail, revealing that all those posts are exceptional. These exceptional cases are further discussed in Section 6.

Pattern	Total Count	Count as EoS
... (three consecutive periods with one or more instances)	548	538
.. (two consecutive periods with one or more instances)	139	139
!!!! (Two or more consecutive occurrences of !)	175	175
??? (Two or more consecutive occurrences of ?)	59	59
!!?? OR .... (combination of one or more different punctuations)	35	35

Table 2: Pattern of punctuations as End-of-Sentence (EoS)

4.2 Machine learning-based approach

The second approach to sentence boundary detection for social media text is based on machine learning. We experimented with applying three machine learning-based classification algorithms to the SBD task, Conditional Random Fields, Sequential Minimal Optimization and Naïve Bayes. The CRF implementation used comes from Miralium[4], while the other two classification were implemented using Weka[5].

Conditional Random Fields are widely used for text sequence labelling tasks. The CRF-based implementation here was trained on basic textual features such as Anchor Word (the current word) which is followed by Next Word, the Previous Word and the word before that (so two words before the Anchor Word). The CRF model provides simultaneously correlated word-level features, which gives greater freedom for incorporating a variety of knowledge sources.

Sequential minimal optimization (Platt, 1998) is an iterative optimization approach to training Support Vector Machines, SVM (Vapnik, 1995). SVM in turn is a binary classification algorithm that aims to separate two classes in a high-dimensional space by inserting hyper-planes between the instances.

5 Experimental Evaluation

To fully evaluate our proposed approaches, we experimented with using some state-of-art SBD systems. We tested the *OpenNLP Sentence Detector*,[6] which is a part of Apache OpenNLP library, and the *splitta* sentence boundary detection tool.[7]

The performance of all the proposed systems was tested on the social media corpus, on a portion of Brown corpus and on the NITA corpus. Details of experiments with results are shown in Table 3. The machine learning approaches were evaluated using 5-Fold cross validation.

The CRF implementation reached 99.81% average accuracy with a weighted average F1-score of 67.0%. The F-measures of Naïve Bayes (NB) and Sequential Minimal Optimization (SMO) are as given in Table 3. It is noticable that the SMO classifier gave the best result for the social media corpus, whereas the Naïve Bayes classifier worked almost as well on SMC and was the best on the Brown corpus, with both those algorithms clearly out-performing CRF on all corpora.

SBD Approach	Corpus based F Measure (in %)		
	SMC	Brown	NITA
Rule-based	78.7	70.6	64.3
SMO	**87.0**	93.3	84.3
NB	86.0	**99.6**	**84.5**
CRF	67.0	57.2	56.9
OpenNLP	68.4	97.5	44.1
splitta: NB	56.1	99.5	54.4
splitta: SVM	54.9	**99.6**	56.0

Table 3: SBD system performance on the corpora

Similarly, of the standard systems tested, the SVM and NB implementations in *splitta* were also succesful on the Brown corpus, but the state-of-the-art systems performed substantially worse on the social media data (SMC and NITA).

6 Error Discussion

The proposed approaches failed to recognize a few sentences in the SMC 3,000 Twitter and Facebook corpus. We analyzed those sentences and found usage of exceptional character sequencies as sentence end markers. Patterns such as 'I', 'II' and '*' are used as sentence end markers, although they are not normally considered as any punctuation symbols. The presence of these patterns is perhaps due to the user's (the one who has posted) expression of the emphasis in the posts; to stress the meaning of the posts by splitting them into sentences.

For this reason, the tested and developed systems could not mark sentence boundaries in some of the exceptional cases. Some examples from the corpus are as follows.

(4) *@RockThatBieber We R Who We R - Keeeeshaa | Only Girl - Rihanna | Shake - Jesse McCartney .... :)*

(5) *@_shintarona_ Op corz gots eet , so now head is spinning with haaawt || Well the YT*

[4] https://code.google.com/p/miralium/

[5] http://www.cs.waikato.ac.nz/ml/weka/downloading.html

[6] https://opennlp.apache.org/documentation/1.5.3/manual/opennlp.html

[7] https://code.google.com/p/splitta/

version I heard sounds exactly like that old one ...

(6) *Me * I love you DjHim * I love you too steph * AlwaysHim and forever :)*

In a few posts, periods have been either wrongly placed or intentionally inserted in between the texts, even though those periods are not actually marking the sentence end. For instance, in the following post

(7) *@syabillaedward honestly ? I have no idea . I just am ... all . the . time . :(*

there are periods (.) occurring at "all.the.time." In this example, the period (.) is inserted between texts to emphasize "all the time".

The tested machine learning algoriths performed well on the Brown corpus, with an F-measure of 93.3% with SMO and 99.6% for NB (Table 3). The reason for such a high score is that the nature of the Brown corpus is formal text where the significance of symbols such as '.', '?' and '!' are well defined. The machine learners are capable of identifying not only the regular meaning of these punctuation symbols, but also their occurrences in different situations. Therefore, it is most likely that our system would perform well on formal texts.

On the other hand, we observed that there is a slight drop in the F-measure of the SMO system from 87% to 84.3% when applied to the NITA English-Hindi code-mixed corpus as compared to the SMC social media corpus. SMC is purely based on English texts. Therefore, the boundaies of sentences within a post are easier to identify. In contrast, in the NITA code-mixed corpus, due to its complex nature, the difficulty of the SBD task increases in two differnt ways; one due to the code-switching at the sentence level and the other due to the code-mixing at the word level. In such a setting, identifying the beginning and end of a Hindi and an English sentence is comparatively difficult. For example:

(8) *Life Ok Dream Girl Ek Ladki Deewani Si Upcoming Tv show Story | Star Cast | Timing | Promo Wiki | Ne http://t.co/X9AhZRRPiD*

(9) *1 Ladke Ne ek Ladki Ko Call Ki Boy : I LOVE U Jaan Girl-Sacchi Boy-Mucchi Girl : ek 100ka Recharge Krwa Do Plz Boy-Sorry Didi Rong Number .*

The above examples are from the NITA English-Hindi code-mixed corpus, and show cases where the tested systems could not recognize the sentence boundaries. Example 8 is a sentence level code-mixed tweet; the following four sentences are there, but no sentence boundary markers were used.

(10) *Life Ok Dream Girl*

(11) *Ek Ladki Deewani Si*

(12) *Upcoming Tv show Story |*

(13) *Star Cast | Timing | Promo Wiki | Ne http://t.co/X9AhZRRPiD*

In Example 9, the English-Hindi code-mixing took place at the word level. This tweet can be split as follows into six sentences, even though there are no proper sentence end markers to define the sentence boundaries.

(14) *1 Ladke Ne ek Ladki Ko Call Ki*

(15) *Boy : I LOVE U Jaan*

(16) *Girl-Sacchi*

(17) *Boy-Mucchi*

(18) *Girl : ek 100ka Recharge Krwa Do Plz*

(19) *Boy-Sorry Didi Rong Number .*

7 Conclusion and Future Work

In this paper we have presented two different approaches to automatic sentence boundary detection in social media text. First, a rule-based system for the SBD task which achieved an F-measure of 78.7% in experiments on social media text. Second, a system based on machine learning approaches to detecting sentence boundaries. We adopted three different machine learning algorithms: Conditional Random Fields, Naïve Bayes, and Sequential Minimal Optimization, with SMO achieving the highest F-score on the social media corpus. For comparison, we also experimented with applying state-of-the art SBD systems to social media text, and with using the systems trained on social media on the more formal Brown corpus, as well as on an English-Hindi code-mixed corpus. This work is the first attempt towards SBD for social media text. Our next target is to make this SBD system more well-built and to make this system adequate for multilingual social media text.

Acknowledgements

Thanks to all the members of TweetNLP team (Carnegie Mellon University), Alan Ritter (Ohio State University), Dan Gillick (Google Research, Mountain View, CA), Robert Bley-Vroman (College of Languages, Linguistics, and Literature of the University of Hawai'i), and the Apache Software Foundation for making their datasets and tools available. Special thanks to the anonymous reviewers for their comments and suggestions that have helped to improve the paper.

References

Eric Brill. 1994. Some advances in transformation-based part of speech tagging. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (Vol. 1)*, AAAI '94, pages 722–727, Menlo Park, CA, USA. American Association for Artificial Intelligence.

Michael John Collins. 1996. A new statistical parser based on bigram lexical dependencies. In *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*, ACL '96, pages 184–191, Stroudsburg, PA, USA. Association for Computational Linguistics.

Doug Cutting, Julian Kupiec, Jan Pedersen, and Penelope Sibun. 1992. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, ANLC '92, pages 133–140, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dan Gillick. 2009. Sentence boundary detection and the problem with the u.s. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, NAACL-Short '09, pages 241–244, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for Twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, volume 2: short papers, pages 42–47, Portland, Oregon, June. ACL.

Gregory Grefenstette and Pasi Tapanainen. 1994. What is a word, what is a sentence? problems of tokenization. In *Proceedings of the 3rd Conference on Computational Lexicography and Text Research*, Budapest, Hungary, July.

Anupam Jamatia, Björn Gambäck, and Amitava Das. 2015. Part-of-speech tagging for code-mixed English-Hindi Twitter and Facebook chat messages. In *Proceedings of 10th International Conference on Recent Advances in Natural Language Processing*, pages 239–248, in Hissar, Bulgaria, 7-9 September, September.

Tibor Kiss and Jan Strunk. 2006. Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, 32(4):485–525, December.

Henry Kučera and W. Nelson Francis. 1967. *Computational Analysis of Present-Day American English*. Brown University Press, Providence, Rhode Island. `http://www.sls.hawaii.edu/bley-vroman/brown_corpus.html`.

Andrei Mikheev. 2002. Periods, capitalized words, etc. *Computational Linguistics*, 28(3):289–318, September.

Brendan O'Connor, Michel Krieger, and David Ahn. 2010. Tweetmotif: Exploratory search and topic summarization for twitter. In *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23-26, 2010*.

David D. Palmer and Marti A. Hearst. 1997. Adaptive multilingual sentence boundary disambiguation. *Computational Linguistics*, 23:241–267.

John C. Platt. 1998. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods — Support Vector Learning*. MIT Press, January.

Jeffrey C. Reynar and Adwait Ratnaparkhi. 1997. A maximum entropy approach to identifying sentence boundaries. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 803–806, Washington, DC, April. ACL.

Michael D. Riley. 1989. Some applications of tree-based modelling to speech and language. In *Proceedings of the Workshop on Speech and Natural Language*, HLT '89, pages 339–352, Stroudsburg, PA, USA. Association for Computational Linguistics.

Vladimir N. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, New York.

Fai Wong and Sam Chao. 2010. iSentenizer: An incremental sentence boundary classifier. In *Natural Language Processing and Knowledge Engineering (NLP-KE), 2010 International Conference on*, pages 1–7, Aug.

Derek F. Wong, Lidia S. Chao, and Xiaodong Zeng. 2014. iSentenizer-μ: Multilingual sentence boundary detection model. *The Scientific World Journal*.