An Approach to Compress English Posts from Social Media Texts

Dwijen Rudrapal, Manaswita Datta, Ankita Datta, Puja Das and Subhankar Das

Abstract Compression of sentences in Facebook posts and Twitter is one of the important tasks for automatic summarization of social media text. The task can be formally defined as transformation of sentence into precise form by preserving the original meaning of the sentence. In this paper, we propose an approach for compressing sentences from Facebook English posts by dropping those words who contribute very less importance to the overall meaning of sentences. We develop one parallel corpus of Facebook English posts and corresponding compressed sentences for our research task. We also report evaluation result of our approach through experiments on develop dataset.

Keywords Sentence compression · Social media text · Machine learning · Facebook posts · Natural language processing

1 Introduction

Social media has received much attention to the researcher over time. Social network sites have become the major source for automatic dissimilation of important and current information about events taking place all over the world. User gets real-time

D. Rudrapal (✉) · M. Datta · A. Datta · P. Das · S. Das
National Institute of Technology, Agartala 799046, Tripura, India
e-mail: dwijen.rudrapal@gmail.com

M. Datta
e-mail: manaswitadatta@gmail.com

A. Datta
e-mail: ankitadatta@gmail.com

P. Das
e-mail: pujadas@gmail.com

S. Das
e-mail: sdasnitacs@gmail.com

H. S. Behera et al. (eds.), *Computational Intelligence in Data Mining*, Advances in Intelligent Systems and Computing 990,
https://doi.org/10.1007/978-981-13-8676-3_8

news, information and public opinion about every subject matter instantly. In short, social media is the information powerhouse to all kind of users.

When needed, a user can search for any particular topic and retrieve the relevant information as desired. But interpreting the information is quite challenging as there is no way that the most significant information can be comprehended by the users from such a sheer volume of data. To make it easy for the users to find out the information they are interested in, an automatic summarization system is required that can generate a condensed content.

Although various approaches are proposed for summarization of social media text, those methods use features such as term frequency [18], query relevance [13] and assign a score to each sentence. Then extract the sentences which maximize the score to compose the intended summary. Those approaches don't shorten the sentence which leads to redundant words, phrases and holds frequent non-significant words. Our work is motivated to solve this shortening problem.

Sentence compression is a sub-task for text summarization. Besides automatic summarization of text, sentence compression is required for various other Natural Language Processing (NLP) processes like semantic role labeling, factoid questioning. The state-of-the-art sentence compression methods are only for standardized texts. Standardized texts are those found in newspaper articles, documents. But social media text differs from the former. The coarse nature of social media text creates certain difficulties for the existing methods [10]. Variation of writing style among users create a lot of challenges like misspelled words, non-standard abbreviation, typological errors, chat slang, etc. Detection of sentence boundary is difficult because of inconsistent punctuation and unwanted capitalization. Emoticons, letter repetition "hiiii", lack of grammaticality are other concerns for any syntactic analysis of such kind of text. Henceforth, the existing methods fail to perform well on social media text.

According to Knight and Marcu [15], given an input source sentence of words $\mathrm{x} = \mathrm{x}_1, \mathrm{x}_2, \ldots \mathrm{x}_\mathrm{n}$ a compression is formed by dropping any subset of words. Good compression are those which use fewer words than the source sentence, retain the most important information from the source sentence and remain grammatically correct. In this paper, we propose an approach to obtain the compressed version of a sentence by deletion of non-significant words or phrases. To the best of our knowledge, this is the first work on sentence compression for social media text.

The rest of the details are divided into the following sections: In Sect. 2, we discuss the promising research relevant to sentence compression. Section 3, describes the dataset preparation method followed by Sect. 4, stating our proposed approach. The experiment setup and result, result analysis is detailed in Sects. 5 and 6, respectively. Section 7, concludes our research work and provides direction for future work.

2 Related Work

Several approaches for sentence compression of standardized text, both machine learning based and rule-based have been proposed to date, some of the promising research works are discussed below.

Sentence compression is one of the most experimented text-to-text generation methods. Most of the compression method proposed in various work [14, 22] detect data redundancy and apply reduction to generate summaries. Some works [4, 11] used compressed forms of texts to generate small screens subtitles. Knight and Marcu proposed model formulated the task as a word or phrase deletion problem. They proposed two models for sentence compression: a probabilistic noisy channel model and a decision-based model to generate abstractive summaries by dropping a subset of words. The deletion problem was further improvised to address global dependency in deleting local words [4, 11].

Some sophisticated models were introduced to improve the accuracy further. Nguyen et al. [19] proposed two sentence reduction algorithms. One, the template-based learning model which learns lexical translation rules and another, a Hidden Markov model which learns sequence of lexical rules. Both the model performed better than the earlier research in terms of grammaticality and importance measure. Filippova and Strube [11] used rules for syntactic and semantic modification of parse tree to generate the compressed form. Cohn and Lapata [5] gave a compression method that extracts tree transduction rules from aligned, parsed texts. Then using a max-margin learning algorithm learns weights on transformations. Later, they [6] presented tree-to-tree transducer that is capable of transforming an input parse tree to a compressed parse tree based on Synchronous Tree Substitution Grammar. Lexicalized Markov Grammar was used by Galley and Mckeown [12] for sentence compression. Clarke and Lapata [3] used Integer Programming approach and encoded various decision variables and constraints in the model to guarantee the grammatical intactness of the reduced sentence. The unsupervised model proposed by Cordeiro et al. [8] used Inductive Logic programming (ILP). Banerjee et al. [2] used ILP for the generation of abstractive summaries by sentence compression. Most recently, Lu et al. [16] present an approach based on a Re-read Mechanism and Bayesian combination model for sentences compression.

All of them are designed for traditional texts. Our work is focused on social media texts, more precisely, Facebook English posts.

3 Dataset Preparation

In this section, we describe the methodology adopted for collecting posts, selecting the required posts, tokenization of the corpus, preparing the parallel corpus and tagging the tokens.

3.1 Facebook English Post Collection

We have collected a corpus of 1884 English posts from Facebook through Facebook API[1] related to UN election 2017 (CNN Facebook page[2]).

After a manual inspection, few posts were removed, being code-mixed or being non-English, as we restrict to English posts only. From these remaining posts, the first sentence of each was extracted since our task is related to sentence compression. Soon it was discovered that a proportion of the posts were identical. Those sentences were removed to make the posts unique in nature. Some sentences were also dropped because of the following reasons:

- Sentences contain unwanted symbols or noise in an intermediate position.
- Sentences didn't convey any meaning.
- Sentences seemed like a single sentence due to the lack of proper end marker.

Finally, 1200 unique and consistent sentences were manually selected for further processing.

3.2 Annotation Process and Challenges

The collected data were annotated using a semi-automatic technique to speed up the manual tagging. Tokenization is difficult for social media texts because of the noisy and informal nature. Ritter's tagger[3] was employed to tokenize and tag the data in the first place. From this 1200 sentences, a total of 16445 tokens are formed i.e. approximately 13.70 tokens for each sentence. The tagger also provided the POS tag, the chunk, and the named entity for each token.

However, the accuracy of Ritter's tagger for our corpus is only 85.3%. To avoid erroneous tagging, manually inspection of the whole corpus was done to correct wrong tags.

The reason behind wrong tokenization and tagging was mainly due to the fact that:

1. Lack of white space between words. For example, "Because he legitimately hasnt passed 1piece of legislation....", "1piece" was treated as a single entity even though two entities "1", "piece" are there.
2. Use of multiple periods between words. For example, "Yeah liberals love a nationalist military state and a narcissistic leader with a bad haircut....oh wait!", "haircut....oh" was considered a single token.
3. Sometimes word was tokenized into two due improper use of punctuation. For example, "Leftists should like N Korea, Gov't runs everything", "Gov't" was divided into two tokens; viz. "Gov" and "t".

[1] https://developers.facebook.com/.

[2] https://www.facebook.com/cnn/.

[3] https://github.com/aritter/twitternlp.

Table 1 Examples of sentence compression

Sentence 01:	Hahahah Todd just contradicted his own argument
Compressed form:	Todd contradicted his own argument.
Sentence 02:	Yeah liberals love a nationalist military state and a narcissistic leader with a bad haircut....oh wait!
Compressed form:	Liberals love a military state and a narcissistic leader!

For preparing the parallel compressed corpus, two annotators are involved in our work. Annotators are native English speakers. Both the annotators have annotated the whole corpus as per our defined annotation guidelines. To prepare gold standard dataset for our experiment, disagreed compression is resolved through discussion among the annotators. Following instructions are provided to the annotators for annotation task. The instructions are:

1. All the posts are related to the US election 2017.
2. Read the sentence and compress it by preserving its original meaning.
3. Restricted to delete words or phrase from the source sentence.
4. Restricted to insert, substitute or reorder the words of the sentence.
5. Not allowed to recreate two sentences from a single sentence when sentences are connected through conjunctions.
6. Sentences that cannot be shortened, keep unaltered.

Table 1 shows two examples of annotated sentences.

4 Proposed Approach

This section discusses the proposed approach to train a machine learning algorithm for compressing English sentences. We formulate some rules to drop words or phrases and accordingly features are derived to train classifiers.

4.1 Rules Formation and Features Selection

Since the parallel corpus is formed by deletion of a subset of words from source sentence, we frame a set of rules to find the viable pattern by which deletion was carried out. The rules are:

- Delete adjective occurring to the left of a noun. For example, in table I, sentence 02, the term “nationalist” is removed.
- Delete adverb occurring to the left of a verb. For example, in table I, sentence 01, the term “just” removed from the sentence.
- Delete the determiner.
- Name entity can be reduced to a single word specifically, reduced to last word of the noun phrase (e.g. the United States of America can be reduced to America).

Considering these set of rules, we select various features based on the lexical and syntactic nature of the sentences to generate the model for automatic compression. The features are:

- The current word (W).
- The previous word (PW).
- Previous to previous word (PPW).
- Next word (NW).
- Next to next word (NNW).
- The chunk (P): While analyzing the parallel corpus, it is observed that either the whole chunk was removed or reduced to its corresponding head word to obtain the compressed sentence.
- POS tag of W, PW, PPW, NW, NNW, P. POS tags resolve the ambiguity that exists between words [9].
- Named entity (NE): Named entities are nouns and hence kept intact for most of the sentences. These tokens are mostly not removed for compression.
- The headword of the chunk (H): The headword in a chunk is that word which is essential to the core meaning of the chunk. For e.g. in case of a noun phrase, the headword is likely to be the subject. The headword of each chunk was identified by following Collins work [7].

The former two lexical features are inspired from the work by Jurafsky et al. [21] where the relationship existing between the current word and its surrounding words are examined.

4.2 Classifier Selection

Choosing the right classifier is crucial for the performance of the experiment. For this experiment, a classifier that can detect whether a given word should be deleted or retained is needed. Since we are dealing with sentences extracted from Facebook, classifiers that work firmly for sequential tagging of data is required. The four classifiers used are namely, Sequential Minimal Optimization (SMO), Naive Bayes (NB), Logistic Regression (LR) and Random Forest (RF).

A NB classifier is based on Bayes theorem and considers all the features to be independent. Empirical study shows that NB [17] is quite effective for sequential tagging of data. SMO is related to both Support Vector Machines (SVM) and optimization

Table 2 Performance analysis of approaches

Evaluation metrics	Approaches			
	SMO	NB	LR	RF
Weighted average F-measure	**0.789**	0.770	0.700	0.759

algorithms and shown comparable performance in previous relevant works [20]. LR is selected because it performs well for correlated features. RF classifier averages multiple decision trees based on random samples from the dataset and proves to be effective for sequential data tagging.

5 Experiment Setup and Result

We used Weka[4] for implementing the machine learning algorithms. On the dataset, tenfold cross validation is performed. Table 2 describes the evaluation metric obtained after applying the algorithms on the dataset. Often, accuracy is used to measure how correct the prediction of a classifier is. However since we have an imbalanced dataset, it will not be helpful. We, therefore use F-measure (harmonic mean of precision and recall) to evaluate the performance of our proposed model. It can be clearly observed that SMO gives the highest weighted average F-measure (0.789) while LR (0.700) performed least.

6 Result Analysis

Table 3 lists two compression examples resulted by experimented algorithms. Although in the most cases the grammatical structure is not retained, still the intended meaning can be conveyed easily. SMO generated compressed sentence is very close to the manually compressed one. While other algorithms output is deviated in terms of compactness as well as readability. Further analysis showed that for most of the sentences, the length of the compressed version and machine-generated compressed version didn't differ much.

7 Conclusion and Future Work

We present a machine learning approach to train social media text corpus for obtaining compressed sentences. We detail on collecting, annotating the corpus using a semi-automatic technique and incorporate various features to determine whether a given

[4] https://www.cs.waikato.ac.nz/ml/weka/.

Table 3 Compressed output by trained models

Sentence 03:	I personally think that it's disrespectful to the n korean people to go there.
Gold form:	It's disrespectful to the n korean to go there.
SMO output:	It's disrespectful to the n korean people to go there.
NB output:	That it's disrespectful to the n korean people to go there.
LR output:	Think that it's is to the n korean people to go there.
RF output:	Personally think it's disrespectful to the n korean people to go there.
Sentence 04:	Normally i wouldn't be too happy about the govt telling me where i can and i can't go.
Gold form:	I wouldn't be happy about govt telling me where i can and i can't go.
SMO output:	I wouldn't be too happy about the govt telling me where i can and i can't go.
NB output:	I wouldn't be too happy about the govt telling me i can and i can't go.
LR output:	I wouldn't be too happy about the telling i can and i can't go.
RF output:	Normally wouldn't be too happy about the govt telling where i can and i can't go.

word should be deleted or retained. Drawing from the result obtained after applying four different machine learning algorithm (SMO, NB, LR, RF), SMO performed best with an F-measure of 0.789.

Being the first attempt at social media text compression, we want to further modify the model to improve the grammaticality of the compressed sentence. We want to increase the size of the corpus substantially to decrease the number of unknown and non-standard words. Owing to the extensive use of non-standard words, we want to employ lexical normalization [1].

References

1. Baldwin, T., Li, Y.: An in-depth analysis of the effect of text normalization in social media. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 420–429 (2015)
2. Banerjee, S., Mitra, P., Sugiyama, K.: Multi-document abstractive summarization using ilp based multi-sentence compression. In: IJCAI, pp. 1208–1214 (2015)
3. Clarke, J., Lapata, M.: Constraint-based sentence compression an integer programming approach. In: Proceedings of the COLING/ACL on Main conference poster sessions, pp. 144–151. Association for Computational Linguistics (2006)
4. Clarke, J., Lapata, M.: Global inference for sentence compression: An integer linear programming approach. J. Artif. Intell. Res. **31**, 399–429 (2008)
5. Cohn, T., Lapata, M.: Sentence compression beyond word deletion. In: Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1, pp. 137–144. Association for Computational Linguistics (2008)
6. Cohn, T., Lapata, M.: An abstractive approach to sentence compression. ACM Trans. Intell. Syst. Technol. (TIST) **4**(3), 41 (2013)
7. Collins, M.: Head-driven statistical models for natural language parsing. Comput. Linguist. **29**(4), 589–637 (2003)

8. Cordeiro, J., Dias, G., Brazdil, P.: Unsupervised induction of sentence compression rules. In: Proceedings of the 2009 Workshop on Language Generation and Summarisation, pp. 15–22. Association for Computational Linguistics (2009)
9. Jurafsky, D., Martin, J.H.: Part-of-speech tagging. In: Speech and Language Processing, 3rd edn, pp. 142–167 (2017). draft(ch. 10)
10. Eisenstein, J.: What to do about bad language on the internet. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies, pp. 359–369 (2013)
11. Filippova, K., Strube, M.: Dependency tree based sentence compression. In: Proceedings of the Fifth International Natural Language Generation Conference, pp. 25–32. Association for Computational Linguistics (2008)
12. Galley, M., McKeown, K.: Lexicalized markov grammars for sentence compression. In: Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, pp. 180–187 (2007)
13. Gupta, S., Nenkova, A., Jurafsky, D.: Measuring importance and query relevance in topic-focused multi-document summarization. In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, pp. 193–196. Association for Computational Linguistics (2007)
14. Jing, H.: Sentence reduction for automatic text summarization. In: Proceedings of the Sixth Conference on Applied Natural Language Processing, pp. 310–315. Association for Computational Linguistics (2000)
15. Knight, K., Marcu, D.: Summarization beyond sentence extraction: A probabilistic approach to sentence compression. Artif. Intell. **139**(1), 91–107 (2002)
16. Lu, Z., Liu, W., Zhou, Y., Hu, X., Wang, B.: An effective approach of sentence compression based on re-read mechanism and bayesian combination model. In: Chinese National Conference on Social Media Processing, pp. 129–140. Springer, Berlin (2017)
17. Najibullah, A.: Indonesian text summarization based on naïve bayes method. In: Proceeding of the International Seminar and Conference on Global Issues, vol. 1 (2015)
18. Nenkova, A., Vanderwende, L., McKeown, K.: A compositional context sensitive multi-document summarizer: exploring the factors that influence summarization. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 573–580. ACM (2006)
19. Nguyen, M.L., Horiguchi, S., Shimazu, A., Ho, B.T.: Example-based sentence reduction using the hidden markov model. ACM Trans. Asian Lang. Inf. Process. (TALIP) **3**(2), 146–158 (2004)
20. Platt, J.: Sequential minimal optimization: A fast algorithm for training support vector machines (1998)
21. Tseng, H., Jurafsky, D., Manning, C.: Morphological features help pos tagging of unknown words across language varieties. In: Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing (2005)
22. Unno, Y., Ninomiya, T., Miyao, Y., Tsujii, J.: Trimming cfg parse trees for sentence compression using machine learning approaches. In: Proceedings of the COLING/ACL on Main Conference Poster Sessions, pp. 850–857. Association for Computational Linguistics (2006)