Sentence Compression for Aspect-Based Sentiment Analysis

Wanxiang Che, Yanyan Zhao, Honglei Guo, Zhong Su, and Ting Liu

***Abstract*—Sentiment analysis, which addresses the computational treatment of opinion, sentiment, and subjectivity in text, has received considerable attention in recent years. In contrast to the traditional coarse-grained sentiment analysis tasks, such as document-level sentiment classification, we are interested in the fine-grained aspect-based sentiment analysis that aims to identify aspects that users comment on and these aspects' polarities. Aspect-based sentiment analysis relies heavily on syntactic features. However, the reviews that this task focuses on are natural and spontaneous, thus posing a challenge to syntactic parsers. In this paper, we address this problem by proposing a framework of adding a sentiment sentence compression (*Sent_Comp*) step before performing the aspect-based sentiment analysis. Different from the previous sentence compression model for common news sentences, *Sent_Comp* seeks to remove the sentiment-unnecessary information for sentiment analysis, thereby compressing a complicated sentiment sentence into one that is shorter and easier to parse. We apply a discriminative conditional random field model, with certain special features, to automatically compress sentiment sentences. Using the Chinese corpora of four product domains, *Sent_Comp* significantly improves the performance of the aspect-based sentiment analysis. The features proposed for *Sent_Comp*, especially the potential semantic features, are useful for sentiment sentence compression.**

***Index Terms*—Aspect-based sentiment analysis, potential semantic features, sentence compression, sentiment analysis.**

I. Introduction

THE internet holds a considerable amount of user-generated content describing the opinions of customers on products and services through blogs, tweets and other social media forms. These reviews are valuable for customers making purchasing decisions and companies guiding business activities. However, browsing the extensive collection of reviews to search for useful information is a time-consuming and tedious task. Consequently, sentiment analysis and opinion mining have attracted significant attention in recent years as they pave the way for the automatic analysis of user reviews and the extraction of information most relevant to users.

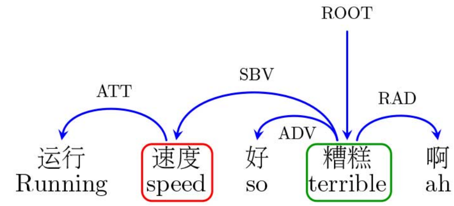


Fig. 1. A parse tree for a sentiment sentence.

Sentiment analysis entails several interesting and challenging tasks. One traditional and fundamental task is polarity classification, which determines the overall polarity (e.g., positive or negative) of a sentence or document [1], [2], [3]. However, these tasks are coarse-grained and cannot provide detailed information, such as the aspects on which the users comment. Recently, there has been a shift towards the fine-grained tasks, such as **aspect-based (or "feature-based") sentiment analysis**, which not only involves analyzing the opinionated text's polarity (e.g., positive, neutral, negative) and intensity (e.g., weak, medium, strong, extreme), but also identifying the aspect (or the topic, or target entity) of the opinion [4], [5], [6], [7].

⟨Aspect, Polarity word⟩ (A-P) collocation extraction and aspect polarity recognition can be considered as the basic tasks of the aspect-based sentiment analysis. For the sentiment sentence "运行速度好糟糕啊" (*The running speed is so terrible.*) provided in Fig. 1,[1] the A-P collocation extraction attempts to extract the collocation ⟨运行速度, 糟糕⟩ (⟨*running speed, terrible*⟩), while the aspect polarity recognition aims to identify the "negative" polarity tag through the polarity word "糟糕" (*terrible*) that modifies the aspect "运行速度" (*running speed*).

Features derived from syntactic parse trees have been proven to be particularly useful for the aspect-based sentiment analysis [8], [9]. For example, in Fig. 1, the syntactic relation "SBV" (SuBject and Verb) between the aspect and the polarity word[2] can be used as an important evidence to extract the A-P collocation ⟨运行速度, 糟糕⟩ (⟨*running speed, terrible*⟩) [11], [6],

Manuscript received November 09, 2014; revised March 19, 2015; accepted May 25, 2015. Date of publication June 11, 2015; date of current version September 04, 2015. This work was supported by the National Key Basic Research Program of China (973) under Grant 2014CB340503 and the National Natural Science Foundation of China (NSFC) under Grants 61300113 and 61370164. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Rodrigo Guido.

W. Che and T. Liu are with the Department of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: car@ir.hit.edu.cn; tliu@ir.hit.edu.cn).

Y. Zhao is with Media Technology and Art, Harbin Institute of Technology, Harbin 150001, China (e-mail: zyy@ir.hit.edu.cn).

H. Guo and Z. Su are with IBM Research-China, China, Beijing 100094, China (e-mail: guohl@cn.ibm.com; suzhong@cn.ibm.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2015.2443982

[1]In this paper, we focus on Chinese sentiment analysis task. The similar method can also be used into other languages.

[2]A Chinese natural language processing toolkit, Language Technology Platform (LTP) [10], was used as our dependency parser. More information about the syntactic relations can be found from their paper. The state-of-the-art graph-based dependency parsing model, in the toolkit, was trained on Chinese Dependency Treebank 1.0 (LDC2012T05).

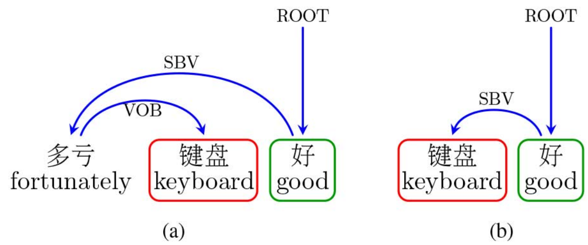


Fig. 2. Parse trees before and after compression. (a) before compression (b) after compression.

[12]. Additionally, several types of syntactic features, such as the syntactic path between two words, are useful for aspect polarity recognition. For example, the syntactic paths between the aspects and the polarity words can help us determine the correct polarity word "糟糕" (*terrible*) instead of the word "好" (*good, very*)[3] for the aspect "运行速度" (*running speed*). Thus, the final polarity for "运行速度" is negative, even though the sentence contains two polarity words, i.e., "好" (*good, very*) and "糟糕" (*terrible*), with opposite sentiment orientations.

However, for the aspect-based sentiment analysis, one major obstacle of the syntactic features-based approaches is the "naturalness" of the sentiment sentences, which are more natural or spontaneous and pose a challenge to syntactic parsers. Thus, several incorrect syntactic features have been produced, and these can further result in the poor performance of the aspect-based sentiment analysis. We can use the sentence in Fig. 2(a) as an example. Because the word "多亏" (*fortunately*) is so colloquial, the parsing result is wrong, which results in the wrong syntactic features. Thus, we are unable to correctly extract the A-P collocation ⟨键盘, 好⟩ (⟨*keyboard, good*⟩). Similarly, an inaccurate parser can also become an obstacle for the aspect polarity recognition.

In return, the improvement in the syntactic parsing performance would have a ripple effect on the aspect-based sentiment analysis. Therefore, to solve the "naturalness" problem, we can improve the performance of the aspect-based sentiment analysis by enhancing the syntactic parsing results. For example, we can train a parser on sentiment sentences to acquire a sentiment-specific parser. Unfortunately, annotating such data will cost us a lot of time and effort. Instead, we produce a sentence compression model, *Sent_Comp*, which is specifically designed to compress the complicated sentiment sentences into ones that are formal and easy-to-parse, which further improves the aspect-based sentiment analysis. Hence, the sentiment sentence compression can be considered as a preprocessing step for the aspect-based sentiment analysis.

This idea is motivated by the observation that current syntactic parsers generally perform accurately for simple and formal sentences; however, the error rates increase for more complex or more natural and spontaneous sentences. For example, the sentence in Fig. 2(a) is in a natural and spontaneous form, and its corresponding parsing result is wrong. However, if we use the *Sent_Comp* model to compress the sentence in Fig. 2(a) into a formal and shortened one in Fig. 2(b) by removing the colloquial part "多亏" (*fortunately*), we can observe that the shortened sentence is well-formed and its parse tree is correct. Thus, it is easier to accurately extract the A-P collocation and recognize the aspect polarity from the compressed sentence.

[3]In Chinese, "好" is a polysemous word. In most cases, it expresses the meaning of "good." But in the sentence of Fig. 1, it means "very."

Sentiment sentence compression is different from traditional sentence compression. Traditional sentence compression aims to obtain a shorter grammatical sentence by reserving important information (generally important grammar structure) [13], [14], [15]. For example, the sentence "Overall this is a great camera" can be compressed into "This is a camera" by removing the adverbial "overall" and the modifier "great." However, the modifier "great," which is also a polarity word, is extremely important for sentiment analysis. Therefore, the *Sent_Comp* model for sentiment sentences is required to reserve the important sentiment information, such as the polarity word. Accordingly, using *Sent_Comp*, the above sentence should be compressed into "this is a great camera."

We regard *Sent_Comp* for the aspect-based sentiment analysis as a sequence labeling task, which can be solved using the Conditional Random Fields (CRF) model. Instead of seeking the manual rules on parse trees for compression, as in other studies [16], the CRF-based method is an automatic procedure. In this study, we introduce certain sentiment-related features, such as perception and polarity features, and potential semantic features, such as word embedding features and word clustering features, for the *Sent_Comp* model.

We apply *Sent_Comp* as the first step of the basic aspect-based sentiment analysis task: A-P collocation extraction. First, we use the *Sent_Comp* model to compress the sentiment sentences into ones that are easier to parse. Then, we use the state-of-the-art aspect-based sentiment analysis approaches on the compressed sentences. The experimental results of the Chinese corpora for four product domains indicate that the approaches using *Sent_Comp* can achieve significant improvements over the approaches without *Sent_Comp*, which indicates that the sentiment sentence compression is effective for the aspect-based sentiment analysis.

The primary contributions of this paper can be concluded as follows:

- We present a framework for using the sentiment sentence compression model to improve the aspect-based sentiment analysis. This framework can better solve the "over-natural" problem of sentiment sentences, which poses a challenge to the syntactic parsers used in the sentiment analysis. More importantly, the idea of this framework can be applied to other sentiment analysis tasks that rely heavily on syntactic results.
- We develop a simple yet effective compression model *Sent_Comp* for sentiment sentences. To the best of our knowledge, this is the first sentiment sentence compression model.
- We propose several features for *Sent_Comp*, in which potential semantic features are particularly effective.

This paper is organized as follows: Section II details the proposed sentiment sentence compression model *Sent_Comp* which combines the CRF and efficient features; Section III introduces a state-of-the-art algorithm for the aspect-based sentiment analysis; Section IV evaluates our models on both

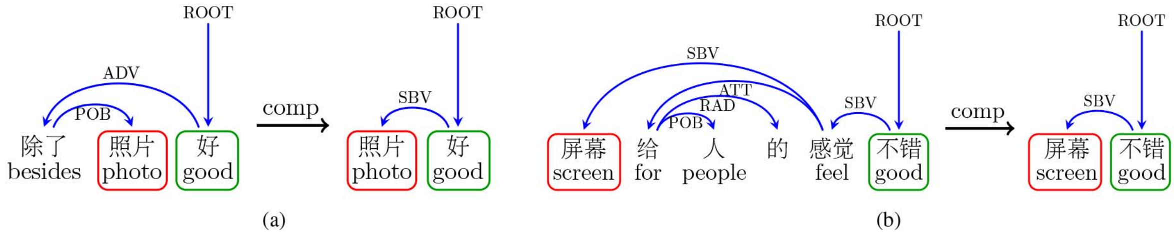


Fig. 3. "Naturalness" problem of sentiment sentences. (a) parse tree 1 before and after compression (b) parse tree 2 before and after compression.

aspect-based sentiment analysis tasks and other sentiment analysis tasks; Section V provides the related work on sentiment analysis and sentence compression; and lastly we conclude this paper in Section VI.

II. Sentiment Sentence Compression

A. Task Analysis

In this section, two questions need to be explained. The first question is that in the sentiment sentences, which are completely different from common sentences, what types of elements need to be compressed. The second question is how to compress the over-natural sentiment sentences into easy-to-parse sentences.

What to be compressed: Different from the common news sentences, the compression model for the sentiment sentences aims to not only compress the redundancy in sentences, but also retain the polarity-related information, such as the polarity words needed to maintain the sentence's original polarity.

We can make an observation of the sentiment sentences that have the wrong syntactic parsing results, from which we can learn what elements need to be compressed. A few examples are listed below.

- Colloquial form: certain sentiment sentences are so colloquial that they cause several difficulties to the parser. For example, in the sentence "多亏键盘好" (*fortunately the keyboard is good*), as indicated in Fig. 2, the usage of the colloquial word "多亏" (*fortunately*) affects the accuracy of the syntactic parser.
- Conjunction word usage: conjunction words are always used in sentiment sentences to indicate the discourse relations between two sentences. However, there are several conjunction words in Chinese, some of which can cause errors for parsers. For example, in Fig. 3(a), the parse tree of the sentence "除了相片较好" (*besides the photo is good*) is wrong because of the usage of the conjunction word "除了" (*besides*). Meanwhile, dropping the conjunction words from the sentiment sentences does not affect the meaning or the polarity orientation of the original sentence.
- Perception words/phrase usage: in sentiment sentences, people always use certain perception words/phrases, such as "给人的感觉" (*feel like*) in Fig. 3(b) or "闻起来" (*smell like*). Given that the current syntactic parser cannot appropriately handle the perception words/phrases, the A-P collocation $\langle$屏幕, 不错$\rangle$ ($\langle screen, good\rangle$) in Fig. 3(b) cannot be extracted correctly.

To address the "naturalness" problem, we propose compressing the sentiment sentences into ones that are shorter and easier-to-parse. Similar to the examples in Figs. 2 and 3, the compressed sentences can be easily and correctly parsed, which are further useful for the sentiment analysis tasks that are heavily dependent on syntactic parsers, such as the aspect-based sentiment analysis.

The above analysis can be used as a criteria to guide us in compressing the sentiment sentences when annotating and help us exploit the useful features for the automatic sentiment sentence compression.

How to Compress: Generally, there are two types of sentence compression methods that have been previously studied for common sentences: the extractive method and the abstractive method. The extractive method preserves the essential content of the sentence by dropping certain unimportant words. The abstractive method compresses an original sentence by reordering, substituting, inserting, and removing its words [18].

Clearly, the abstractive method requires more resources and is more complicated. More importantly, this type of method can easily change the original aspects or the polarity words, which are always treated as the important elements in the aspect-based sentiment analysis tasks. Therefore, in this paper, we focus on only the extractive approach to compress the sentiment sentences.

A traditional sentence compression model deletes the unnecessary words and reserves the basic content, thus the primary elements for the sentiment analysis, such as the polarity words and aspects, are more likely to be dropped. Based on the above analysis, the principle for the sentiment sentence compression model is to reserve the sentiment-related words, besides reserving the basic content.

B. Task Definition

Formally, an extractive sentence compression aims to shorten a sentence $\mathbf{x} = x_1 \cdots x_n$ into a substring $\mathbf{y} = y_1 \cdots y_m$, where $y_i \in \{x_1, \cdots, x_n\}$, $m \leq n$.

The sentiment sentence compression task can be converted into a classic classification problem, in which each word in a sentiment sentence is classified as "delete" or "reserve." Similar to the work of Nomoto *et al.* [19], in this paper we regard the sentiment sentence compression as a sequence labeling task, which can be solved using the Conditional Random Fields (CRF) model.

We assign a compression tag t_i to each word x_i in an original sentence $\mathbf{x}$, where

$$t_i = \begin{cases} \texttt{N} & \text{if } x_i \in \mathbf{y} \\ \texttt{Y} & \text{else.} \end{cases}$$

For instance, the sentence from Fig. 3(b)

屏幕/screen 给/for 人/people 的/ 感觉/feel 不错/good

can be tagged into:

TABLE I
FEATURES OF SENTIMENT SENTENCE COMPRESSION FOR ASPECT-BASED SENTIMENT ANALYSIS. w IS THE WORD, AND t IS THE POS TAGGING. perception$(\cdot)$ AND polarity$(\cdot)$ ARE BOTH BINARY FEATURES TO INDICATE WHETHER A WORD IS A PERCEPTION (POLARITY) WORD OR NOT. suffix$(\cdot)$ AND prefix$(\cdot)$ ARE THE LAST AND THE FIRST CHARACTER OF A CHINESE WORD. cluster$(\cdot)$ REPRESENTS THE BROWN WORD CLUSTERING FEATURE, AND wordembedding$(\cdot)$ IS THE WORD EMBEDDING FEATURE. BOTH cluster$(\cdot)$ AND wordembedding$(\cdot)$ HAVE MULTIPLE DIMENSIONS. dependency$(\cdot)$ IS THE DEPENDENCY RELATION BETWEEN A WORD AND ITS PARENT

Basic Features
01: $w_{i+k}, -1 \leq k \leq 1$
02: $w_{i+k-1} \circ w_{i+k}, 0 \leq k \leq 1$
03: $t_{i+k}, -2 \leq k \leq 2$
04: $t_{i+k-1} \circ t_{i+k}, -1 \leq k \leq 2$
Sentiment-related Features
05: perception(w_i)
06: polarity(w_i)
Potential Semantic Features
07: suffix(w_i) if $t_i == n$ else prefix(w_i)
08: cluster(w_i)
09: wordembedding(w_i)
Syntactic Features
10: dependency(w_i)

屏幕_N 给_Y 人_Y 的_Y 感觉_Y 不错_N

A first-order linear-chain CRF is used, which defines the conditional probability as follows:

$$P(\mathbf{t}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_i M_i(t_i, t_{i-1}|\mathbf{x}) \tag{1}$$

where $\mathbf{x}$ and $\mathbf{t}$ are the input and output sequence, respectively; $Z(\mathbf{x})$ is the partition function; and M_i is the clique potential for the edge clique i.

Based on previous studies, feature selection and representation are essential for this task.

C. Features

The features for the sentiment sentence compression model *Sent_Comp* are listed in Table I, which can be divided into two parts. The first part is the features used in common sentence compression, including word (w), POS tagging (t) and their combination context features (01–04). These are known as Basic Features. Because sentiment sentences are slightly different from common sentences, we need to introduce certain special features. We have designed the sentiment-related features (05–06) and potential semantic features (07–09) to better handle sentiment analysis data and generalize word features, which comprise the second part. Lastly, we add the syntactic parse features (10), which are commonly used in traditional sentence compression task. In this section, we primarily introduce several sets of special features for the sentiment sentences below.

The first set is **Sentiment-related Features**, which depict how to process the sentiment-related elements of a sentiment sentence during the compression. For example, we can delete certain perception words, such as "I think", which do not change the meaning and the sentiment orientation of the original sentence but are always wrongly parsed. Conversely, we cannot delete the polarity words, such as "perfect", that are essential to sentiment sentences. In this paper, we design two types of sentiment-related features: Perception Features and Polarity Features.

Perception Features: The perception feature `perception`$(\cdot)$ indicates whether a word is a perception word. This type of feature is inspired by the naturalness problem in Fig. 2(b). As discussed above, the current parser produces wrong parse trees because of these perception words. Therefore, the perception words tend to be removed from a sentiment sentence for the sentiment sentence compression model *Sent_Comp*. We can obtain a perception word lexicon from HowNet,[4] a popular Chinese sentiment thesaurus, where a perception word is defined by `DEF` = {`perception`|感知} tag. Lastly, we collected 38 perception words, such as 发觉 (*realize*), 发现 (*find*) and 认为 (*think*). The perception features are represented as binary features; if the word w_i is a perception word, we tag it with "Perception", otherwise, we tag it with "no_Perception."

Polarity Features: The polarity feature `polarity`$(\cdot)$ indicates whether a word is a polarity word. One of the primary differences between the sentiment sentences and the common news sentences is that the former typically contain polarity words. In contrast to the `perception`$(\cdot)$, polarity words tend to be reserved because they are important and specific to the sentiment analysis. For example, if we delete "great" from the sentence "overall this is a great camera", the sentence turns into an objective sentence without a sentiment orientation. In this paper, we treat polarity words as important features, considering that they are always considered as modifiers and can be easily removed using common sentence compression methods.

We can obtain the polarity feature `polarity`$(\cdot)$ from a polarity lexicon, which can also be obtained from HowNet. Similar to the perception features, polarity features are also represented as binary features. If the word w_i is a polarity word, we tag it with "Polarity", otherwise, we tag it with "no_Polarity."

Aside from the basic features and the apparent sentiment-related features described above, we also explore a few deep and **Potential Semantic Features** to generalize the words in sentiment sentences, primarily including word suffix/prefix character features, Brown word clustering features and word embedding features.

Suffix and Prefix Character Features: The first type of potential semantic features is a suffix or prefix character feature (`prefix`$(\cdot)$ or `suffix`$(\cdot)$). In contrast to English, the suffix or prefix characters of a Chinese word often carry that word's core semantic information. For Chinese nouns, the suffix can carry this information. For example, 自行车 (*bicycle*), 汽车 (*car*) and 火车 (*train*) are all various types of 车 (*vehicle*), which is also the suffix of the three words. Given that all of them may become aspects, they tend to be reserved in compressed sentences. Thus, the suffix character features are important. Furthermore, for the words that are not nouns, such as verbs, the prefix can always carry the core information. For example, both verbs 感觉 and 感到, can be denoted by their prefix *feel* (感), and they can be removed from the original sentences because they are perception words. Similarly, the prefix character features are also useful.

Brown Word Clustering Features: We use a word clustering feature (`cluster`$(\cdot)$), which is a typical low-dimensional and

[4]www.keenage.com

generalized word representation, as another potential semantic feature to further improve the generalization over common words. The word clustering features contain certain semantic information and have been successfully used in several natural language processing tasks, including NER [20], [21] and dependency parsing [22]. For instance, the words 外观 and 样子 (*appearance*) belong to the same word cluster, even though they have different suffixes or prefixes. Both words are important for aspect-based sentiment analysis, and both of them should be reserved for sentiment sentence compression. Apparently, word clustering features can help us group and generalize the words.

The Brown word clustering algorithm [23] is one of the most effective word clustering algorithms. Liang *et al.* [24] presented an optimization method to significantly reduce the traditional clustering time to obtain the word clusters. Raw texts that have been used to train the Brown word clustering algorithm are obtained from the 5th edition of the Chinese Gigaword (LDC2011T13). The output of the Brown algorithm is a binary tree, where each word is uniquely identified by its path from the root. Thus, each word can be represented as a bit-string with a specific length. We can easily obtain the expected clusters by keeping only a certain length of the bit-string prefix. Lastly, we induce 1,000 Brown clusters of words, which is the same setting used in the prior study [25], [22]. These clusters are assigned separate cluster IDs.

Using the Brown word clustering features, we can better generalize similar-meaning but different-representation words, such as 外观 and 样子 (*appearance*).

Word Embedding Features: A word embedding is a function that maps words in a certain language to dense, continuous, and low-dimensional vectors (perhaps 50 to 500 dimensions) [26]. This type of 'map' of words ensures that similar words are distributed close together. Thus, word embedding can be treated as a type of soft word clustering. Consequently, word embeddings can be beneficial for a variety of NLP applications in different ways; the most simple and general way is to be fed as features to enhance existing supervised NLP systems. Previous studies have demonstrated the effectiveness of the continuous word embedding features in several tasks, such as chunking and NER using generalized linear models [25], [27].

Based on the above discussion, we consider using `wordembedding` features as one type of potential semantic features to represent each word of a sentiment sentence. If the word embedding features of two words are similar, it can indicate that they should have the same operation (delete or not) during the sentiment sentence compression. For instance, the word embeddings of the word 不错 and 给力 (*good*) are similar, and they should be reserved. Word embeddings can be learned from large-scale unlabeled texts through context-predicting models (e.g., neural network language models) or spectral methods (e.g., canonical correlation analysis) in an unsupervised setting. In this paper, an efficient open-source implementation of the Skip-gram model is adopted.[5] We apply a negative sampling method for optimization as well as an asynchronous stochastic gradient descent algorithm (Asynchronous SGD) for parallel weight updating. We set the dimension of the word embeddings to 50. A higher dimension is thought to bring more improvements in semi-supervised learning, but its comparison is beyond the scope of this paper.

[5]code.google.com/p/word2vec/

TABLE II
AN EXAMPLE OF THE FEATURES FOR THE WORD "感觉"

Basic Features	Example Values
01: $w_{i+k}, -1 \leq k \leq 1$	"的", "感觉", "不错"
02: $w_{i+k-1} \circ w_{i+k}, 0 \leq k \leq 1$	"的 ∘ 感觉", "感觉 ∘ 不错"
03: $t_{i+k}, -2 \leq k \leq 2$	"n", "u", "v", "a"
04: $t_{i+k-1} \circ t_{i+k}, -1 \leq k \leq 2$	"$n \circ u$", "$u \circ v$", "$v \circ a$"
Sentiment-related Features	**Example Values**
05: perception(w_i)	"Perception"
06: polarity(w_i)	"no_Polarity"
Potential Semantic Features	**Example Values**
07: suffix(w_i) if $t_i == n$ else prefix(w_i)	"感"
08: cluster(w_i)	a bit-string with a specific length
09: wordembedding(w_i)	50-dimensional vectors
Syntactic Features	**Example Values**
10: dependency(w_i)	"SBV"

Dependency Features: Lastly, similar to several previous sentence compression studies, such as the work of McDonald *et al.* [28], we also add the dependency relation between a word and its parent as the syntactic features. Intuitively, the `dependency` relations are beneficial in conducting sentence compression. For example, the `ROOT` relation typically indicates that the word should not be removed because it is the main verb of a sentence.

Even so, because the syntactic parsing results for sentiment sentences are not as perfect as those for common news sentences, the dependency features may be not so efficient as those in the common sentence compression model.

Compared with [17], this study explores more potential semantic features, such as word embedding features, which lead to improving the performance of the sentiment sentence compression model for the aspect-based sentiment analysis.

To better understand the above features, we provide an example to illustrate all the features in Table II. Based on the feature list in Table I, we acquire all the corresponding features for the word "感觉" (feel) in the example "屏幕/screen 给/for 人/people 的/ 感觉/feel 不错/good" in Section II-B as follows.

III. FRAMEWORK FOR ASPECT-BASED SENTIMENT ANALYSIS WITH SENTIMENT SENTENCE COMPRESSION

A. Framework

Sentiment sentence compression can be considered as a preprocessing step for the aspect-based sentiment analysis tasks that are heavily dependent on syntactic parsing results. Fig. 4 depicts the framework that uses sentiment sentence compression for the aspect-based sentiment analysis.

In this paper, we primarily focus on studying the compression model on sentiment sentences (Section II), and applying this model to the basic aspect-based sentiment analysis tasks (Section III) to demonstrate its effectiveness.

B. State-of-the-Art Algorithm for Aspect-based Sentiment Analysis

In this section, we introduce the state-of-the-art algorithm for the aspect-based sentiment analysis [6], which we used as our

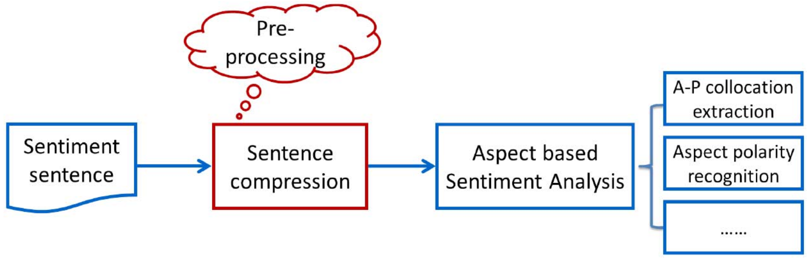


Fig. 4. The framework of using sentiment sentence compression for aspect-based sentiment analysis.

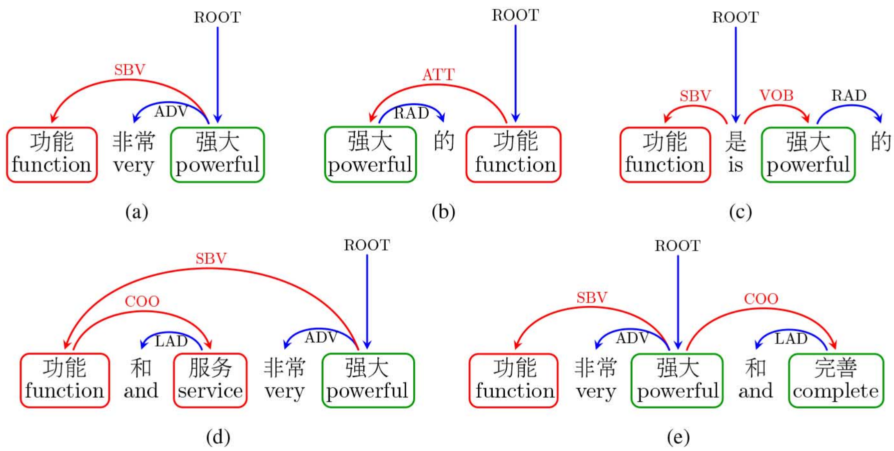


Fig. 5. Example of syntactic structure rules for A-P collocation extraction. We show five examples from a total of nine syntactic structures. For each kind of syntactic structure (a) to (e), the aspect is shown with a red box and the polarity word is shown with a green box. Syntactic structures (a) to (c) describe the relations between aspects and polarity words. Syntactic structure (d), which is extended from (a), describes the relation between two aspects. Syntactic structure (e), which is also extended from (a), describes the relation between two polarity words. Similarly, we can summarize the other four rules extended from (b) and (c) to describe the relations between two aspects or two polarity words. (a) syntactic structure 1 (b) syntactic structure 2 (c) syntactic structure 3 (d) syntactic structure 4 (e) syntactic structure 5.

baseline system. They proposed a double propagation method to extract the A-P collocations, aspects and polarity words. This idea is based on the observation that there are natural syntactic relations between polarity words and aspects owing to the fact that polarity words are used to modify the aspects. Furthermore, they also discovered that the polarity words and aspects themselves had relations in certain sentiment sentences.

Based on this idea, in the double propagation method, we first used an initial seed polarity word lexicon and syntactic relations to extract the aspects, which can fall into a new aspect lexicon. Then, we used the aspect lexicon and the same syntactic relations to extract the polarity words to expand the polarity word lexicon in return. This is an iterative procedure, i.e., this method can iteratively produce the new polarity words and the aspects back and forth using the syntactic relations.

We can observe that the syntactic relations are important to this method, and Qiu *et al.* [6] proposed eight rules to describe these relations. However, their study focused on only English sentences whereas the relations for Chinese sentences are different. Thus, in accordance with Chinese grammar, we propose nine syntactic structure rules between the aspect a and the polarity word p to extract the Chinese A-P collocation $\langle a, p\rangle$. The three primary rules are provided below, and certain example rules are illustrated in Fig. 5.

Rule 1: $a \overset{\text{SBV}}{\curvearrowleft} p$, expresses the "subject-verb" structure between a and p, such as the example in Fig. 5(a).

Rule 2: $p \overset{\text{ATT}}{\curvearrowleft} a$, expresses that p is a modifier for a, such as the example in Fig. 5(b).

Rule 3: $a \overset{\text{SBV}}{\curvearrowleft} \circ \overset{\text{VOB}}{\curvearrowright} p$, expresses the "subject-verb-object" structure between a and p, such as the example in Fig. 5(c). The symbol $\circ$ denotes any word.

The other six rules can be extended from the three primary rules by obtaining the coordination (COO) relation of a or p. For example, $a \overset{\text{COO}}{\curvearrowleft} \circ \overset{\text{SBV}}{\curvearrowleft} p$ in Fig. 5(d) and $a \overset{\text{SBV}}{\curvearrowleft} \circ \overset{\text{COO}}{\curvearrowright} p$ in Fig. 5(e) are extended from Fig. 5(a). It should be noted that the POS for a should be a noun, and the POS for p should be an adjective.

Apparently, the classic aspect-based sentiment analysis tasks, such as the A-P collocation extraction or aspect extraction, rely heavily on syntactic parsers, especially the syntactic relation features between two words. Meanwhile, as described in Section I, another aspect-based sentiment analysis task, i.e., aspect polarity recognition, relies on the polarity of the polarity word in the extracted A-P collocation, which indicates that this task indirectly relies on syntactic parsers. Hence, if we can use the *Sent_Comp* model to improve the performance of the parsers, then the performance of the aspect-based sentiment analysis can be improved accordingly.

However, for the other sentiment analysis tasks, such as the traditional sentence sentiment classification task [29], [30], [31], the sentiment sentence compression model may be of little use. The reason is that the state-of-the-art method for sentence senti-

TABLE III
STATISTICS FOR THE CHINESE CORPORA OF FOUR PRODUCT DOMAINS

Domain	# reviews	# sentences	# collocations
Camera	338	2,232	2,244
Car	161	1,172	1,204
Notebook	56	623	631
Phone	323	2,557	2,573
All	878	6,584	6,652

ment classification is the machine learning based method combined with certain features, in which syntactic features are not important. Moreover, machine learning based methods require rich features; however, the features are significantly reduced after using the sentence compression model. In this paper, we will provide a few experiments and discussions to demonstrate what types of tasks that the sentiment sentence compression model is suitable for.

IV. EXPERIMENTS

A. Experimental Setup

Corpora: The corpora that the experiments have been conducted on are obtained from two sources. The first source is the Task3 of the Chinese Opinion Analysis Evaluation (COAE) [32],[6] which includes four product domains, i.e., digital camera, car, phone and notebook. The other source is from the work of Zhao *et al.* [33], which includes two product domains, i.e., digital camera and phone. Table III describes the statistics of the corpora, where 6,584 sentiment sentences containing 6,652 A-P collocations are manually identified and annotated from 878 reviews.

Evidently, it is simple to evaluate the performances of the aspect-based sentiment analysis tasks, such as the A-P collocation extraction task, using the corpora in Table III. However, to evaluate the performance of the sentiment sentence compression model *Sent_Comp*, we request a few annotators to manually compress all the sentiment sentences in Table III into shorter ones to train and test the *Sent_Comp* model. Specifically, the annotators delete certain words from a sentiment sentence based on the eight types of annotation rules in Appendix A. For these sentences that cannot meet the rules, they annotate them based on the following two criteria: (1) deleting the word that cannot change the essential content of the sentence, and (2) deleting the word that cannot change the sentiment orientation of the sentence. To assess the quality of the annotation, we sample 500 sentences from these corpora and request two experts to perform the annotation. The resulting word-based Cohen's kappa [34], which is a measure of inter-annotator agreement ranging from zero to one, is approximately 0.7, indicating a good strength of agreement. Additionally, according to the corpora statistics, we observe that approximately 50% of the sentiment sentences can be manually compressed, indicating that the *Sent_Comp* model can be used in several sentiment sentences.

Evaluation: Generally, the compressions are evaluated using three criteria [28], i.e., grammaticality, importance, and compression rate. Clearly, the grammaticality and importance are difficult to evaluate objectively. Previous studies used human judgment, which is difficult and expensive. In this paper, we simply use the F-score metric of removed words to roughly evaluate the performance of the sentiment sentence compression. The evaluation functions are defined below. Evidently, the final effectiveness of the sentence compression model can also be reviewed by the final aspect-based sentiment analysis results.

[6]www.ir-china.org.cn/coae2008.html

TABLE IV
THE RESULTS OF SENTIMENT SENTENCE COMPRESSION WITH DIFFERENT FEATURE SETS

Feature sets	Features	P(%)	R(%)	F(%)
BF (01 – 04)	Basic (01 – 04)	76.58	62.23	68.66
+ SF (05 – 06)	BF + perception (05)	76.83	62.17	68.73
	BF + polarity (06)	75.91	62.80	68.73
+ PSF (07 – 09)	BF + suffix or prefix (07)	77.10	62.76	69.20
	BF + cluster (08)	76.55	63.47	69.40
	BF + word embedding (09)	75.45	63.73	69.09
+ SynF (10)	BF + dependency (10)	75.42	62.54	68.38
	All (01 – 09)	77.32	63.33	69.63

$$Precision = \frac{\#correct\ removed\ words}{\#model\ removed\ words}$$
$$Recall = \frac{\#correct\ removed\ words}{\#standard\ removed\ words}$$
$$F-score = \frac{2 * Precision * Recall}{Precision + Recall}$$

Additionally, we apply the traditional P, R, and F-*score* to evaluate the A-P collocation extraction task. Specifically, a fuzzy matching evaluation is used. Namely, given an extracted A-P collocation $\langle a, p\rangle$, whose standard result is $\langle a_s, p_s\rangle$, if $a \subseteq a_s$ and $p \subseteq p_s$, we can consider the extracted $\langle a, p\rangle$ as a correct A-P collocation.

B. Sentiment Sentence Compression Results

In this section, we present the experimental results of the sentiment sentence compression model *Sent_Comp* with different feature sets individually in Table I, i.e., Basic Features (**BF**), Sentiment-related Features (**SF**), Potential Semantic Features (**PSF**) and Syntactic Features (**SynF**). All the experiments are conducted using ten-fold cross validation.

The comparative results are provided in Table IV. We observe that the SF feature set, i.e., the perception feature (05) and the polarity feature (06), can improve the performance of the sentiment sentence compression with a small increase in the F-*score*. The reason is that both the perception and the polarity features are lexical features that have overlaps with the BF feature set.

The second type of feature set, the PSF feature set can significantly improve the performance over the BF feature set. It is reasonable that the PSF feature set explores the deep semantic representation of each word that is hidden behind the literal representation in the BF features. Therefore, this potential semantic feature set is complemental to the literal basic feature set. Three types of methods, i.e., the suffix/prefix character (07), the Brown word clustering (08) and the word embedding (09), are proposed to represent the potential semantic features of each word in the sentiment sentence. The detailed performances of these three potential semantic features are presented in Table IV. We can observe that all three types of features are effective. Here, we can also observe that the performance of adding word embedding features is a little lower than the other two kinds of features. The reason is that the word embedding features are just one type of the potential semantic features for sentiment sentence compression; they are supplementary to the other potential semantic features, i.e., the suffix/prefix character (07), the Brown word clustering (08), but not antagonistic to them. That is to say, although word embedding features failed to achieve better results than the other potential semantic features, once combining them together, the final system can outperform each of separate features. The similar conclusion was reached on other NLP tasks, such as the NER and Chunking tasks in [25] and the NER task in [27].

Nonetheless, from Table IV, it can be observed that the `dependency` features (10) have a negative effect on the sentiment sentence compression performance. This is completely different from the compression model for common news sentence, in which the syntactic features are the most necessary features [19], [28]. The reason for this fact is easily explained. The lower dependency parsing performance for the sentiment sentences introduces several wrong dependency relations, which counteract the contribution of the dependency relation features. This is also the reason why we need to compress the sentiment sentences as the first step of the aspect-based sentiment analysis. Lastly, when we combine all of the useful features (01–09), the performance achieves the highest score.

It is worth noting that the sentiment sentence compression is a new task proposed in this paper. For simplicity, this paper aims to attempt a simple yet effective sentiment sentence compression model. Several studies, such as selecting more useful features or polishing the model, can be performed on the *Sent_Comp* model in the future.

C. Effectiveness of Sentiment Sentence Compression Model for Aspect-based Sentiment Analysis

We select a traditional aspect-based sentiment analysis task, i.e., the A-P collocation extraction task, as a case study for two purposes. The first purpose is to demonstrate whether the *Sent_Comp* is reasonable for the aspect-based sentiment analysis. The second purpose is to demonstrate whether the approach proposed for the sentiment sentence compression in Section II is effective.

We design three comparative systems for the A-P collocation extraction below. It should be noted that *Sent_Comp* is the first step in correcting the corpora before the aspect-based sentiment analysis. Furthermore, the method for the A-P collocation extraction is the state-of-the-art method proposed by Qiu *et al.* [6], which has been described in Section III in detail.

- **no_Comp** This refers to the system that uses only the A-P collocation extraction method and does not perform *Sent_Comp* as the first step.
- **manual_Comp** This system **manually** compresses the corpora into new ones as the first step, and then applies the A-P collocation extraction method on the new compressed corpus.
- **auto_Comp** This system uses *Sent_Comp* as the first step to **automatically** compress the corpora into new ones, and then applies the A-P collocation extraction method on the new corpora.

From the descriptions above, we can observe that the first system does not use the compression model, and the other two systems use the *Sent_Comp* model as the preprocessing step. Furthermore, we can draw a conclusion that the performance of **manual_Comp** can be considered as the upper bound for the sentence compression based A-P collocation extraction task.

TABLE V
RESULTS ON A-P COLLOCATION EXTRACTION FOR FOUR PRODUCT DOMAINS

Domain	Method	P(%)	R(%)	F(%)
Camera	no_Comp	72.42	57.22	63.93
	manual_Comp	79.06	59.05	67.60
	auto_Comp	75.99	57.98	65.77
Car	no_Comp	67.71	55.54	61.03
	manual_Comp	74.38	57.49	64.85
	auto_Comp	72.17	56.49	63.37
Notebook	no_Comp	74.49	59.35	66.06
	manual_Comp	83.21	66.17	73.72
	auto_Comp	79.67	64.54	71.31
Phone	no_Comp	73.03	59.89	65.81
	manual_Comp	79.65	63.89	70.91
	auto_Comp	77.91	61.83	68.95
All	no_Comp	71.99	58.75	**64.70**
	manual_Comp	78.86	61.96	**69.39**
	auto_Comp	76.42	60.43	**67.49**

Table V presents the experimental results of the three systems using the A-P collocation extraction method for the four product domains. Here, the **manual_Comp** can significantly ($p < 0.01$) improve the *F-score* by approximately 5%[7] compared with that of the **no_Comp**. This result illustrates that the idea of sentiment sentence compression is useful for A-P collocation extraction. Specifically, the proposed method can transform certain over-natural sentences into normal ones, further influencing their final syntactic parsers. Evidently, because the A-P collocation extraction relies heavily on the syntactic features, the more correct syntactic parse trees derived from the compressed sentences can help increase the performance of the basic task of aspect-based sentiment analysis.

Compared with the **no_Comp**, the **auto_Comp** system also yields a significantly better result ($p < 0.01$) that indicates an improvement of 3% in the *F-score*, despite the fact that the automatic sentence compression model *Sent_Comp* may wrongly compress certain sentences. We observe that the *F-score* of the *Sent_Comp* is approximately 70%, which is not perfect. However, the *Sent_Comp* model is still effective for A-P collocation extraction. These results demonstrate that the idea of using sentiment sentence compression for the aspect-based sentiment analysis is reasonable and further prove that our CRF model combined with several feature sets used in the *Sent_Comp* is effective.

Moreover, we can observe that the idea of sentence compression and our *Sent_Comp* model are useful for all four product domains in the A-P collocation extraction task, which indicates that the *Sent_Comp* model is domain-independent. However, we can observe a small gap between **auto_Comp** and **manual_Comp**, which indicates that the *Sent_Comp* model can still be improved further. In the future, we will explore more effective sentence compression algorithms to bridge the gap between the two systems.

Further, Table VI lists several actual examples that are rescued using the compressed sentences. Here, the first column is

[7]We use paired bootstrap resampling significance test [35].

TABLE VI
ACTUAL EXAMPLES THAT ARE RESCUED USING THE COMPRESSED SENTENCES

No.	Before compression (original)	After compression	Gold A-P collocation
1	多亏/fortunately 键盘/keyboard 好/good	键盘/keyboard 好/good	⟨键盘/keyboard,好/good⟩
2	联想/Lenovo 服务/service 真的/really 很/very 棒/good	联想/Lenovo 服务/service 很/very 棒/good	⟨服务/service,棒/good⟩
3	使得/make 百公里/per 100km 油耗/fuel consumption 更/much 低/lower	百公里/per 100km 油耗/fuel consumption 更/much 低/lower	⟨油耗/fuel consumption,低/lower ⟩
4	除了/besides 内饰/interior 精细度/quality 比较/more 突出/outstanding 之外/	内饰/interior 精细度/quality 比较/more 突出/outstanding 之外/	⟨精细度/quality,突出/outstanding ⟩
5	中控台/console 上/ 的/ 控制/control 按钮/button 看起来/look 很/very 繁琐/complicated	中控台/console 上/ 的/ 控制/control 按钮/button 很/very 繁琐/complicated	⟨按钮/button, 繁琐/complicated⟩
6	在/ 便携性/convenience 上/ 并/ 不/not 太/very 好/good	便携性/convenience 并/ 不/not 太/very 好/good	⟨ 便携性/convenience,好/good ⟩
7	但/but 总体/in all 看/ 显示屏/screen 还/ 可以/good	显示屏/screen 还/ 可以/good	⟨显示屏/screen,可以/good⟩
8	画面/picture 中/inside 透着/show 一些/some 柔和/soft	画面/picture 透着/show 一些/some 柔和/soft	⟨ 画面/picture,柔和/soft⟩
9	但/but 听起来/sounds 效果/effect 还/ 不是/not 那么/ 理想/ideal	效果/effect 还/ 不是/not 那么/ 理想/ideal	⟨效果/effect,理想/ideal ⟩
10	做工/quality 确实/indeed 很/very 不错/good 了/	做工/quality 很/very 不错/good 了/	⟨做工/quality,不错/good ⟩

the example number. We have listed 10 examples. The second column shows the original sentences before compression; in each example, we extract each A-P collocation using the state-of-the-art algorithm, in which the aspect is labeled in red and the polarity word is labeled in green. If no aspect or polarity word is labeled, this indicates that no A-P collocation is extracted. The third column shows the sentences after compression; in each example, we use the same algorithm to extract the A-P collocation and use the same color to label the aspects and the polarity words. The gold A-P collocation for each sentence is shown in the fourth column.

From Table VI, we can observe that:

- After compression, we can correctly extract all the A-P collocations from the compressed sentences. In contrast, before compression, we can just extract the A-P collocations from two example sentences (No.2 and No.10), and both of them are wrongly extracted. We cannot extract the A-P collocations from the other 8 examples due to the imperfect syntactic results.
- These examples can illustrate that our proposed sentiment sentence compression method can compress the over-natural sentiment sentences into easy-to-parse sentences. And further, on the more correct syntactic parsing results, the performance of the A-P collocation extraction task is much better.

According to the definition from SemEval-2014 Task 4: Aspect Based Sentiment Analysis [30], the aspect-based sentiment analysis aims to identify the aspects of the entities being reviewed and determine the sentiment that the reviewers express for each aspect. The aspect and its polarity are the cores of the aspect-based sentiment analysis; therefore, the A-P collocation extraction mentioned in this paper is the most basic and classic task of the aspect-based sentiment analysis.

Theoretically, the sentiment sentence compression could be effective for other aspect-based sentiment analysis tasks, such as aspect polarity recognition and aspect extraction, because these tasks are dependent on the syntactic parsing results. For example, when extracting the A-P collocations, we can extract the aspects and generate the polarity word lexicon simultaneously [6]. Moreover, based on the statistics of the Chinese sentiment analysis corpora [33], more than 60% of sentiment sentences have A-P collocations. Clearly, the polarity of the aspect a can be correctly assigned by the polarity of its modifying polarity word p. Therefore, based on the A-P collocations, we can easily recognize more than 60% of the aspects' polarities.

D. Error Analysis

In this part, we make an error analysis on the results of Section IV-C. We randomly sample 500 sentences including 508 A-P collocations from the four domains, and compare the error distributions on the aspect-based sentiment analysis task when the compression results are correct and not. The results are shown in Table VII. The second and the third column describe the numbers of the A-P collocations, the ratios of the three kinds of errors, and the ratios of the correct results, when applying the aspect-based sentiment analysis algorithm on the correctly and incorrectly compressed sentences. It should be noted that the compressed sentences are automatically obtained using the *Sent_Comp* model in Section II.

By analyzing the extracted A-P collocations, we observed three kinds of errors.

- **Algorithm Error**: it is caused by the algorithm of the aspect-based sentiment analysis that is introduced in Section III-B.
- **Syntactic Parsing Error**: it is caused by the incorrect syntactic parsing results, such as the examples in Fig. 2 and Fig. 3.
- **Compression Error**: it is caused by the incorrect sentiment sentence compression results. For example, using the compression model *Sent_Comp*, an actual sentiment sentence of our corpora "充电器比较差" ("the charger is rather poor" in English) has been compressed into "比较差" ("rather poor" in English) by deleting the word "充电器" ("the charger" in English) which is also the aspect of this sentence, thus leading that we cannot extract the A-P collocation from the compressed sentence.

TABLE VII
ERROR DISTRIBUTIONS ON SENTIMENT ANALYSIS WHEN THE COMPRESSION RESULTS ARE CORRECT AND NOT

Type	Correct Comp	Incorrect Comp
# of A-P collocations	390	118
Error 1: Algorithm error	15.64%	17.80%
Error 2: Syntactic parsing error	12.31%	27.12%
Error 3: Compression error	0.00%	20.34%
Correct results	72.05%	34.75%

TABLE VIII
RESULTS OF USING *SENT_COMP* ON SENTENCE SENTIMENT CLASSIFICATION

Method	System	Acc(%)
Pang et al's	no_Comp_SSC	88.78
	manual_Comp_SSC	88.04
	auto_Comp_SSC	87.95
Mohammad et al's	no_Comp_SSC	91.43
	manual_Comp_SSC	91.06
	auto_Comp_SSC	90.67

From Table VII, it can be observed that the "Algorithm Error" and "Syntactic Parsing Error" can appear when applying the sentiment analysis algorithm on both the correctly compressed sentences and incorrectly compressed sentences. However, the third kind "Compression Error" can just appear when conducting on the incorrectly compressed sentences. Further, we can reach the following conclusions.

- The ratios of the "Algorithm Error" are comparative when applying the sentiment analysis algorithm on the correctly compressed sentences (15.64%) and the incorrectly compressed sentences (17.80%). It is reasonable, because we use the same aspect-based sentiment analysis algorithm on both the correctly and incorrectly compressed sentences.
- Comparing the ratios of the "Syntactic Parsing Error" on the correctly and incorrectly compressed sentences, we can find the ratio is much lower on the correctly compressed sentences (12.31%) than that on the incorrectly compressed sentences (27.12%). This is obvious, because correctly compressed sentences can obtain more correct syntactic parsing results than the incorrectly compressed sentences. From the results on the correct compressions, we can also observe that although we have correctly compressed the sentences into the short and easy-to-parse ones, 12.31% of the sentences are still wrongly parsed. We think these errors could be solved by increasing the performance of syntactic parsing.
- When applying the sentiment analysis algorithm on the incorrectly compressed sentences, we observe a special kind of error "Compression Error", which accounts for 20.34%. From Table IV, we observe that the compression model *Sent_Comp* in our paper is not perfect, thus leading to some incorrectly compressed sentences that affect the final sentiment analysis performance. But fortunately, more sentences can be compressed correctly, thus the gain of the sentiment analysis performance from the correctly compressed sentences can recover the loss from the incorrectly compressed sentences. This can be verified in Table V, that is, the sentiment analysis on the compressed sentences performs better than that on the sentences without compression.

E. Impact of Sentiment Sentence Compression Model for other Sentiment Analysis Tasks

In addition to the aspect-based sentiment analysis tasks, we also want to determine whether the compression model *Sent_Comp* is effective for other sentiment analysis tasks. We consider a basic and primary sentiment analysis task, sentence sentiment classification, as a case study, which aims to classify a sentence into positive, negative and neutral.

We perform two classic methods on this task. The first method is obtained from Pang *et al.* [29], who selected the unigram as the feature and used machine learning tools. This method is the first model for sentence sentiment classification, and several studies have proven that the unigram feature used in this method was the most important feature. The second method is obtained from Mohammad *et al.* [36], who built a state-of-the-art system in sentiment analysis of tweets and achieved first place in the subtasks of the SemEval-2013 competition "Detecting Sentiment in Twitter."

To evaluate the effectiveness of the *Sent_Comp* model, three comparative systems are designed for sentence sentiment classification (**SSC**).

- **no_Comp_SSC** This refers to the system using just the classic sentiment sentence classification method without using *Sent_Comp* as the first step.
- **manual_Comp_SSC** This system **manually** compresses the corpora into new ones as the first step and then applies the classic sentiment sentence classification methods on the new corpora.
- **auto_Comp_SSC** This system uses *Sent_Comp* as the first step to **automatically** compress the corpora and then applies the classic sentiment sentence classification methods on the compressed corpora.

Table VIII presents the experimental results of the three systems combined with the two classic methods, i.e., Pang *et al.*'s method and Mohammad *et al.*'s method, on a sentence sentiment classification task. Furthermore, we use $Accuracy$ to evaluate this task. Compared with the results of the aspect-based sentiment analysis, we unfortunately acquire a completely different conclusion. Namely, despite the method (Pang *et al.*'s or Mohammad *et al.*'s) we use, the system **no_Comp_SSC** that does not use the compression model *Sent_Comp* performs better than both the **manual_Comp_SSC** and **auto_Comp_SSC** systems that use *Sent_Comp*, even if we manually compress the corpora in the **manual_Comp_SSC**.

We make a thorough analysis to explain the results.

- Comparing the results that apply Pang *et al.*'s method and Mohammad *et al.*'s method, we find that although the features and resources used in Mohammad *et al.*'s method are complex and rich, the performances increase by only approximately 3%, which is not large. This result can illustrate that the unigram feature used in Pang *et al.*'s method is the most effective feature for sentence sentiment classification. Moreover, this can indicate that the syntactic features are not necessary for this task. For example, for the sentence "屏幕/screen 给/for 人/people 的/ 感觉/feel 不错/good" in Fig. 3(b), we can recognize its polarity just by unigram, especially by the inside polarity word "不错

/good." In comparison, syntactic features are almost useless. Furthermore, the compression model in our paper is not perfect. So if we mistakenly delete the polarity word, such as "不错/good" in this example when using the compression model, it is hard to recognize the polarity just by the features from the compressed sentence.

- For any machine learning tools, rich features are required. If we use the compression model *Sent_Comp* to compress a long sentence into a short one, the unigram features are correspondingly reduced and several other useful features are lost. Therefore, the performances of the systems using *Sent_Comp*, i.e., the **manual_Comp_SSC** and the **auto_Comp_SSC**, are slightly lower than that of the **no_Comp_SSC** system that does not use it. For example, for the sentence "多亏/fortunately 键盘/keyboard 好/good" in Fig. 2, the compressed sentence is "键盘/keyboard 好/good." However, the deleted word "多亏/fortunately" also has a "positive" sentiment orientation. Maybe it is helpful for recognizing the sentence's polarity. Thus after compression, the features are not as rich as before.

Based on the above analysis, the reason that the compression model is not useful for sentence sentiment classification task is clear. Namely, the sentiment sentence compression model is fit for tasks heavily dependent on the syntactic parsing results, such as aspect extraction, A-P collocation extraction, etc. Thus, the aspect-based sentiment analysis can benefit from this compression model, as discussed in Part C of Section IV. Conversely, other sentiment analysis tasks, represented by the task of sentence sentiment classification, cannot benefit from this compression model because they are not heavily dependent on the syntactic features.

V. Related Work

A. Sentiment Analysis

Earlier research on sentiment analysis primarily focused on polarity classification, i.e., determining the sentiment orientation of a sentence or a document [1], [2], [3]. However, these tasks are all coarse-grained and cannot provide more detailed information. Recently, there has been a shift towards fine-grained aspect-based tasks that can identify both the text expressing the opinion and the aspect of the opinion as well as analyzing its polarity (e.g., positive, neutral or negative) [37], [5], [6]. The A-P collocation extraction is the basic task of aspect-based sentiment analysis.

To tackle this task, most methods focused on identifying relationships between the aspects and the polarity words. In earlier studies, researchers recognized the aspect first and then chose its polarity word within a window of size k [7]. However, considering that this type of method is too heuristic, the performances proved to be extremely limited. To solve this problem, several researchers found that a syntactic pattern can better describe the relationship between the aspects and the polarity words. For example, Bloom *et al.* [11] constructed a linkage specification lexicon containing 31 patterns. Qiu *et al.* [6] proposed a double propagation method that introduced eight heuristic syntactic patterns to extract the collocations. Xu *et al.* [12] used the syntactic patterns to extract the collocation candidates in their two-stage framework.

Based on the above discussion, we can conclude that the syntactic features are extremely important in executing aspect-based sentiment analysis tasks. However, the "naturalness" problem can still seriously affect the performance of the syntactic parser. Once our sentiment sentence compression method can improve the quality of parsing, the performance of several aspect-based sentiment analysis tasks can be improved as well. It should be noted that to date, there is no previous study on using sentence compression models to improve aspect-based sentiment analysis.

B. Sentence Compression

Sentence compression is a paraphrasing task that aims at generating sentences that are shorter than the given ones, while preserving the essential content [13]. There are many applications that can benefit from a robust compression system. For example, we can use the system to reduce the redundancy in sentences and generate informative summarization systems [38]. Additionally, we can use it to compress the complicated sentiment sentences into easy-to-parse ones to get more accurate syntactic features, and further improve the tasks that primarily rely on the syntactic features, such as semantic role labeling [16], relation extraction [39], etc.

Tree-based approaches were commonly used to compress sentences [13], [14], [15], which created a compressed sentence by making edits to the syntactic tree of the original sentence. However, the automatic parsing results may not be correct; thus, the compressed tree (after removing constituents from a bad parse) may not produce a suitable compressed sentence. McDonald *et al.* [28] and Nomoto *et al.* [19] attempted to solve this problem using discriminative models. They studied classifiers to determine which words could be dropped by including features of the words themselves as well as part of the speech tags and parser trees. Here, the parser trees were soft evidence to determine whether to remove a word. Accordingly, the influencers of the parsing errors were reduced. Moreover, to improve the efficiency of the compression model, recent studies have been performed on polynomial time inference algorithms [40] and approximate inference algorithms [41] for sentence compression.

Currently, the existing sentence compression methods all focus on formal sentences, and few methods have been studied for sentiment sentences. As discussed in the above sections, the current compression models cannot be directly transplanted to sentiment sentences due to the specificity of the aspect-based sentiment analysis. Therefore, a new compression model for sentiment sentences should be established.

VI. Conclusion

We present a framework for using a sentiment sentence compression model *Sent_Comp* for aspect-based sentiment analysis. Different from the common sentence compression model, *Sent_Comp* not only compresses the redundancy in the sentiment sentences, but also needs to retain the polarity-related information to maintain the sentences' original polarities. Thus, the over-natural and spontaneous sentiment sentences can be compressed into more formal and easier-to-parse sentences after using the *Sent_Comp* model. Accordingly, the most important features for the aspect-based sentiment analysis, i.e.,

APPENDIX A
ANNOTATION RULES WITH COMPRESSION EXAMPLES

#	Rule Type	Examples	
		Original	Compressed
1	Conjunction word/phrase usage	1.除了/besides 相片/photo较好/good 2.虽然/although 没有/no 漂亮/beautiful 的/ 机身/body 设计/design 3.而且/and 还/also 带有/with 诱人/attractive 的/ 28/28 广角/wide-angle lens	1.相片/photo较好/good 2.没有/no 漂亮/beautiful 的/ 机身/body 设计/design 3.带有/with 诱人/attractive 的/ 28/28 广角/wide-angle lens
2	Colloquial word/phrase usage	1.多亏/fortunately 键盘/keyboard 好/good 2.再 加上/along with 按键/key 数目/number 太/too 多/many 3. 好在/luckily 奥林巴斯/Olympus 的/ 售后/service非常/very 好/good	1.键盘/keyboard 好/good 2.按键/key 数目/number 太/too 多/many 3. 奥林巴斯/Olympus 的/ 售后/service非常/very 好/good
3	Adverbial modifier before polarity word	1.屏幕/screen 看/look 起来/ 不错/good 2.快门/shutter 时滞/lag 真的/really 让人/for people 很/very 不爽/not good 3.自动/automatic 白平衡/white balance 相对/relatively 来说/speaking 比较/very 准确/accurate	1.屏幕/screen 不错/good 2.快门/shutter 时滞/lag很/very 不爽/not good 3.自动/automatic 白平衡/white balance比较/very 准确/accurate
4	Interjection	1.非常/very 人性化/humanity 的/ 设计/design 啊/a 2.方向盘/steering wheel 很/very 小巧/small and exquisite 喔/oh	1.非常/very 人性化/humanity 的/ 设计/design 2.方向盘/steering wheel 很/very 小巧/small and exquisite
5	Default subject	1.使得/make 夜景/night scene 拍摄/shooting 能力/ability 一般/ordinary 2.令/make 取景/viewfinding角度/point 更加/very 灵活/flexible 3.给人/for people 不错/good 的/ 操作/operation 感受/feel	1.夜景/night scene 拍摄/shooting 能力/ability 一般/ordinary 2.取景/viewfinding角度/point 更加/very 灵活/flexible 3.不错/good 的/ 操作/operation 感受/feel
6	Special adverbial collocation	1.在/ 速度/speed 处理/processing 方面/perspective 非常/very 迅速/quick 2.在/ 噪点/noise 控制/controlling 上/ 还是/still 不错/good 的/	1.速度/speed 处理/processing非常/very 迅速/quick 2.噪点/noise 控制/controlling 还是/still 不错/good 的/
7	Adverbial modifier in front of a sentence	1.在/under 阳光/sunshine 下/ 色彩/color 还原/rendition 非常/very 好/good 2.阴天/cloudy day 里/ 的/ 长焦/Telephoto 表现/Presentation 一般/ordinary 3.相对/relative to 于/ 同/ homologous系列/series 产品/product 来说/speaking 高/high 性价比/cost performance 是/is 三星/Samsung 的/ 优势/advantage	1.色彩/color 还原/rendition 非常/very 好/good 2.长焦/Telephoto 表现/Presentation 一般/ordinary 3. 高/high 性价比/cost performance 是/is 三星/Samsung 的/ 优势/advantage
8	Comparative part	1.PRO1/PRO1 的/ 画质/photo 比/than 828/828 更加/more 丰富/rich 2.声音/voice 部分/part 比/than 我/my 预期/expectation 的/ 好/better	1.PRO1/PRO1 的/ 画质/photo 更加/more 丰富/rich 2.声音/voice 部分/part好/better

syntactic features, can be more correctly acquired to enhance the performance of this task.

The sentiment sentence compression can be converted to determine each word in a sentiment sentence to be classified as "delete" or "reserve." In this paper, we establish a CRF-based compression model with rich features, including sentiment-related features and potential semantic features.

We conduct several experiments on the corpora of four product domains to evaluate the effectiveness of the feature sets used for *Sent_Comp* and the effectiveness of the *Sent_Comp* model applied in the aspect-based sentiment analysis. Our experimental results can validate the following points:

- Comparing the feature sets sentiment-related feature (SF) and the potential semantic feature (PSF) used in modeling *Sent_Comp*, the PSF set, which uses three types of methods, i.e., suffix/prefix character features, Brown word clustering features, and word embedding features, is more effective in generalizing the words in sentiment sentences.
- *Sent_Comp* is proven to be effective for the aspect-based sentiment analysis, which can also demonstrate that the CRF based method in Section II is effective for the sentiment sentence compression task.
- *Sent_Comp* is useful for the sentiment analysis tasks that rely heavily on syntactic features, such as the aspect-based sentiment analysis. However, for the tasks in which the syntactic features are not necessary, they cannot benefit from using the *Sent_Comp* model.
- *Sent_Comp* is domain-independent.

The idea of using sentiment sentence compression for aspect-based sentiment analysis can be considered as a basic framework. We believe that more sentiment analysis tasks that rely heavily on syntactic features will benefit from the sentiment sentence compression model.

APPENDIX A

Please see table at the top of the page.

REFERENCES

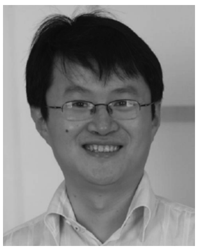
[1] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, Jan. 2008.

[2] B. Liu, *Sentiment Analysis and Opinion Mining*, ser. Synthesis Lectures on Human Language Technologies. San Rafael, CA, USA: Morgan & Claypool, 2012.
[3] E. Cambria, B. Schuller, Y. Xia, and C. Havasi, “New avenues in opinion mining and sentiment analysis,” *IEEE Intell. Syst.*, vol. 28, no. 2, pp. 15–21, 2013.
[4] T. Wilson, “Fine-grained subjectivity and sentiment analysis: Recognizing the intensity, polarity, and attitudes of private states,” Ph.D. dissertation, Intell. Syst. Program, Univ. of Pittsburgh, Pittsburgh, PA, USA, 2007.
[5] A. Mukherjee and B. Liu, “Aspect extraction through semi-supervised modeling,” in *Proc. 50th Annu. Meeting Assoc. Comput. Linguist. (Vol. 1: Long Papers)*, Jeju Island, Korea, Jul. 2012, pp. 339–348, Assoc. for Comput. Linguist..
[6] G. Qiu, B. Liu, J. Bu, and C. Chen, “Opinion word expansion and target extraction through double propagation,” *Comput. Linguist.*, vol. 37, no. 1, pp. 9–27, 2011.
[7] M. Hu and B. Liu, “Mining and summarizing customer reviews,” in *Proc. KDD’04*, 2004, pp. 168–177.
[8] A. Abbasi, H. Chen, and A. Salem, “Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums,” *ACM Trans. Inf. Syst.*, vol. 26, no. 3, pp. 12:1–12:34, Jun. 2008.
[9] A. Duric and F. Song, “Feature selection for sentiment analysis based on content and syntax models,” *Decis. Support Syst.*, vol. 53, no. 4, pp. 704–711, Nov. 2012.
[10] W. Che, Z. Li, and T. Liu, “Ltp: A Chinese language technology platform,” in *Coling 2010: Demonstrations*, Beijing, China, Aug. 2010, pp. 13–16, Coling 2010 Organizing Committee.
[11] K. Bloom, N. Garg, and S. Argamon, “Extracting appraisal expressions,” in *Proc. HLT-NAACL’07*, 2007, pp. 308–315.
[12] L. Xu, K. Liu, S. Lai, Y. Chen, and J. Zhao, “Mining opinion words and opinion targets in a two-stage framework,” in *Proc. 51st Annu. Meeting Assoc. Comput. Linguist. (Vol. 1: Long Papers)*, Sofia, Bulgaria, Aug. 2013, pp. 1764–1773, Assoc. for Comput. Linguist..
[13] K. Knight and D. Marcu, “Summarization beyond sentence extraction: A probabilistic approach to sentence compression,” *Artif. Intell.*, vol. 139, no. 1, pp. 91–107, Jul. 2002.
[14] J. Turner and E. Charniak, “Supervised and unsupervised learning for sentence compression,” in *Proc. 43rd Annu. Meeting Assoc. for Comput. Linguist.*, Stroudsburg, PA, USA, 2005, pp. 290–297, ser. ACL ’05, Assoc. for Comput. Linguist..
[15] M. Galley and K. McKeown, “Lexicalized Markov grammars for sentence compression,” in *Proc. Human Lang. Technol.: Conf. North Amer. Chap. Assoc. for Comput. Linguist.; Proc. Main Conf.*, Rochester, NY, USA, Apr. 2007, pp. 180–187, Assoc. for Comput. Linguist..
[16] D. Vickrey and D. Koller, “Sentence simplification for semantic role labeling,” in *Proc. ACL*, 2008, pp. 344–352, Assoc. for Comput. Linguist..
[17] Y. Zhao, W. Che, H. Guo, B. Qin, Z. Su, and T. Liu, “Sentence compression for target-polarity word collocation extraction,” in *Proc. 25th Int. Conf. Comput. Linguist.: Technical Papers (COLING)*, Dublin, Ireland, Aug. 2014, pp. 1360–1369, Dublin City Univ. and Assoc. for Comput. Linguist..
[18] T. Cohn and M. Lapata, “An abstractive approach to sentence compression,” *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 3, pp. 1–35, 2013.
[19] T. Nomoto, “Discriminative sentence compression with conditional random fields,” *Inf. Process. Manage.*, vol. 43, no. 6, pp. 1571–1587, Nov. 2007.
[20] S. Miller, J. Guinness, and A. Zamanian, “Name tagging with word clusters and discriminative training,” in *HLT-NAACL’04: Main Proc.*, D. M. Susan Dumais and S. Roukos, Eds., Boston, MA, USA, May 2–7, 2004, pp. 337–342, Assoc. for Comput. Linguist..
[21] W. Che, M. Wang, C. D. Manning, and T. Liu, “Named entity recognition with bilingual constraints,” in *Proc. Conf. North Amer. Chap. Assoc. for Comput. Linguist.: Human Lang. Technol.*, Atlanta, GA, USA, Jun. 2013, pp. 52–62, Assoc. for Comput. Linguist..
[22] T. Koo, X. Carreras, and M. Collins, “Simple semi-supervised dependency parsing,” in *Proc. ACL-08: HLT*, Columbus, OH, USA, Jun. 2008, pp. 595–603, Assoc. for Comput. Linguist..
[23] P. F. Brown, P. V. deSouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai, “Class-based n-gram models of natural language,” *Comput. Linguist.*, vol. 18, no. 4, pp. 467–479, Dec. 1992.
[24] P. Liang, “Semi-supervised learning for natural language,” M.S. thesis, Mass. Inst. Technol., Cambridge, MA, USA, 2005.
[25] J. Turian, L. Ratinov, and Y. Bengio, “Word representations: A simple and general method for semi-supervised learning,” in *Proc. 48th Annu. Meeting Assoc. for Comput. Linguist.*, Stroudsburg, PA, USA, 2010, pp. 384–394, ser. ACL’10, Assoc. for Comput. Linguist..
[26] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, “A neural probabilistic language model,” *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, Mar. 2003.
[27] J. Guo, W. Che, H. Wang, and T. Liu, “Revisiting embedding features for simple semi-supervised learning,” in *Proc. EMNLP’14*, 2014.
[28] R. McDonald, “Discriminative sentence compression with soft syntactic evidence,” in *Proc. EACL*, 2006.
[29] B. Pang, L. Lee, and S. Vaithyanathan, “Thumbs up? sentiment classification using machine learning techniques,” in *Proc. EMNLP’02*, 2002, pp. 79–86.
[30] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, “Semeval-2014 task 4: Aspect based sentiment analysis,” in *Proc. 8th Int. Workshop Semantic Eval. (SemEval’14)*, Dublin, Ireland, Aug. 2014, pp. 27–35, Assoc. for Comput. Linguist. and Dublin City Univ..
[31] S. Rosenthal, A. Ritter, P. Nakov, and V. Stoyanov, “Semeval-2014 task 9: Sentiment analysis in twitter,” in *Proc. 8th Int. Workshop Semantic Eval. (SemEval 2014)*, Dublin, Ireland, Aug. 2014, pp. 73–80, Assoc. for Comput. Linguist. and Dublin City Univ..
[32] J. Zhao, H. Xu, X. Huang, S. Tan, K. Liu, and Q. Zhang, “Overview of Chinese pinion analysis evaluation 2008,” in *Proc. 1st Chinese Opinion Anal. Eval. (COAE’08)*, 2008.
[33] Y. Zhao, B. Qin, and T. Liu, “Creating a fine-grained corpus for Chinese sentiment analysis,” *IEEE Intell. Syst.*, vol. 30, no. 1, pp. 36–43, Jan.-Feb. 2015.
[34] J. Cohen, “A coefficient of agreement for nominal scales,” *Educat. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, 1960.
[35] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*. New York, NY, USA: Chapman & Hall, 1993.
[36] S. M. Mohammad, S. Kiritchenko, and X. Zhu, “NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets,” *CoRR*, vol. abs/1308.6242, 2013.
[37] T. A. Wilson, “Fine-grained subjectivity and sentiment analysis: Recognizing the intensity, polarity, and attitudes of private states,” Ph.D. dissertation, Univ. of Pittsburgh, Pittsburgh, PA, USA, 2008.
[38] C. Li, F. Liu, F. Weng, and Y. Liu, “Document summarization via guided sentence compression,” in *Proc. Conf. Empir. Meth. Nat. Lang. Process.*, Seattle, WA, USA, Oct. 2013, pp. 490–500, Assoc. for Comput. Linguist..
[39] M. Miwa, R. Saetre, Y. Miyao, and J. Tsujii, “Entity-focused sentence simplification for relation extraction,” in *Proc. 23rd Int. Conf. Comput. Linguist. (COLING’10)*, 2010, pp. 788–796.
[40] X. Qian and Y. Liu, “Polynomial time joint structural inference for sentence compression,” in *Proc. 52nd Annu. Meeting Assoc. for Comput. Linguist. (Vol. 2: Short Papers)*, Baltimore, MD, USA, Jun. 2014, pp. 327–332, Assoc. for Comput. Linguist..
[41] K. Thadani, “Approximation strategies for multi-structure sentence compression,” in *Proc. 52nd Annu. Meeting Assoc. for Comput. Linguist. (Vol. 1: Long Papers)*, Baltimore, MD, USA, Jun. 2014, pp. 1241–1251, Assoc. for Comput. Linguist..

Wanxiang Che received his Ph.D. in computer science from Harbin Institute of Technology (HIT), China, in 2008. He is a full-time Associate Professor in the School of Computer Science and Technology, HIT. His current research interests include natural language processing and information retrieval.

Yanyan Zhao received her Ph.D. in computer science from Harbin Institute of Technology (HIT), China, in 2011. She is a full-time Assistant Professor in the Department of Media Technology and Art, HIT. Her current research interests include sentiment analysis and social computing.

Honglei Guo received her Ph.D. degree in computer science from Northeastern University in 1997. She is a Research Staff Member at the Cognitive Understanding & Analytics department of IBM Research-China. Her research interests include natural language processing, text mining, sentiment analysis, machine translation, and machine learning.



Zhong Su received his Ph.D. degree in computer science from Tsinghua University in 2002. He is a Senior Technical Staff Member at the IBM Research-China (CRL) and Senior Manager of the Cognitive Understanding & Analytics department. He has been involved in many projects in CRL including text analytics, NLP, rich media analysis, and information integration.

Ting Liu received his Ph.D. in computer science from Harbin Institute of Technology (HIT), China, in 1998. He is a Full Professor in the School of Computer Science and Technology, HIT. His current research interests include natural language processing, information retrieval, and social computing.