Sentence Compression for Target-Polarity Word Collocation Extraction

Yanyan Zhao[1], **Wanxiang Che**[2], **Honglei Guo**[3], **Bing Qin**[2], **Zhong Su**[3] **and Ting Liu**[2*]
1: Department of Media Technology and Art, Harbin Institute of Technology
2: Department of Computer Science and Technology, Harbin Institute of Technology
3: IBM Research-China
{yyzhao, bqin, tliu}@ir.hit.edu.cn, {guohl, suzhong}@cn.ibm.com

Abstract

Target-polarity word (T-P) collocation extraction, a basic sentiment analysis task, relies primarily on syntactic features to identify the relationships between targets and polarity words. A major problem of current research is that this task focuses on customer reviews, which are natural or spontaneous, thus posing a challenge to syntactic parsers. We address this problem by proposing a framework of adding a sentiment sentence compression (*Sent_Comp*) step before performing T-P collocation extraction. *Sent_Comp* seeks to remove the unnecessary information for sentiment analysis, thereby compressing a complicated sentence into one that is shorter and easier to parse. We apply a discriminative conditional random field model, with some special sentiment-related features, in order to automatically compress sentiment sentences. Experiments show that *Sent_Comp* significantly improves the performance of T-P collocation extraction.

1 Introduction

Sentiment analysis deals with the computational treatment of opinion, sentiment and subjectivity in text (Pang and Lee, 2008), and has received considerable attention in recent years (Liu, 2012). Target-Polarity word (T-P) collocation extraction, which aims to extract the collocation of a target and its corresponding polarity word in a sentiment sentence, is a basic task in sentiment analysis. For example, in a sentiment sentence "这款相机拥有新颖的外形" (*The camera has a novel appearance*), "外形" (*appearance*) is the target, and "新颖" (*novel*) is the polarity word that modifies "外形" (*appearance*). According, ⟨外形, 新颖⟩ (⟨*appearance, novel*⟩) is the T-P collocation. Generally, T-P collocation is a basic and complete sentiment unit, thus is very useful for many sentiment analysis applications.

Features derived from syntactic parse trees are particularly useful for T-P collocation extraction (Abbasi et al., 2008; Duric and Song, 2012). For example, the syntactic relation "Adj $\overset{\text{ATT}}{\curvearrowleft}$ Noun", where the ATT denotes an attributive syntactic relation, can be used as an important evidence to extract the T-P collocation ⟨外形, 新颖⟩ (⟨*appearance, novel*⟩) in the above sentiment sentence (Bloom et al., 2007; Qiu et al., 2011; Xu et al., 2013).

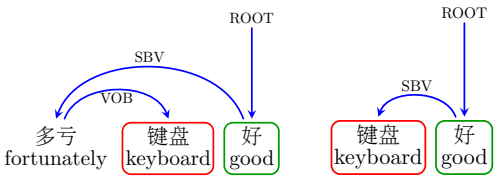
However, one major problem of these approaches is the "naturalness" of sentiment sentences, that is, such sentences are more natural or spontaneous compared with normal sentences, thus posing a challenge to syntactic parsers. Accordingly, many wrong syntactic features have been produced and these can further result in the poor performance of T-P collocation extraction. Taking the sentence in Figure 1(a) as an example, because the word "多亏" (*fortunately*) is so chatty,[1] the parsing result is wrong. Thus, are unable to extract the T-P collocation ⟨键盘, 好⟩ (⟨*keyboard, good*⟩).

To solve the "naturalness" problem, we can train a parser on sentiment sentences. Unfortunately, annotating such data will cost us a lot of time and effort. Instead, in this paper we produce a sentence compression model, *Sent_Comp*, which is designed especially to compress complicated sentiment sentences into formal and easier to parse ones, further improving T-P collocation extraction.

*Correspondence author: tliu@ir.hit.edu.cn

[1]Note that, in Figure 1, the Chinese word "多亏" is chatty, although its translated English word "fortunately" is not. In this paper, we focus on processing the Chinese data.

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1360–1369, Dublin, Ireland, August 23-29 2014.



(a) before compression (b) after compression

Figure 1: Parse trees before and after compression.

This idea is motivated by the observation that, current syntactic parsers usually perform accurately for short, simple and formal sentences, whereas error rates increase for longer, more complex or more natural and spontaneous sentences (Finkel et al., 2008). Hence, the improvement in syntactic parsing performance would have a ripple effect over T-P collocation extraction. For example, we can compress the sentence in Figure 1(a) into a shortened sentence in Figure 1(b) by removing the chatty part "多亏" (*fortunately*). We can see that the shortened sentence is now well-formed (in Chinese) and its parse tree is correct, making it easier to accurately extract T-P collocation.

Traditional sentence compression aims to obtain a shorter grammatical sentence by retaining important information (usually important grammar structure) (Jing, 2000). 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" is a polarity word and very important for sentiment analysis. Therefore, *Sent_Comp* model for sentiment sentences is different from the traditional compression models, because it needs to retain the important sentiment information, such as the polarity word. Hence, using *Sent_Comp*, the above sentence can be compressed into "This is a great camera."

We regard *Sent_Comp* as a sequence labeling task, which can be solved by a conditional random fields (CRF) model. Instead of seeking the manual rules on parse trees for compression, as in other studies (Vickrey and Koller, 2008), this method is an automatic procedure. In this work, we introduce some sentiment-related features to retain the sentiment information for *Sent_Comp*.

We apply *Sent_Comp* as the first step in the T-P collocation extraction task. First, we compress the sentiment sentences into easier to parse ones using *Sent_Comp*, after which we employ the state-of-the-art T-P collocation extraction approach on the compressed sentences. Experimental results on a Chinese corpus of four product domains show the effectiveness of our approach.

The main contributions of this paper are as follows:

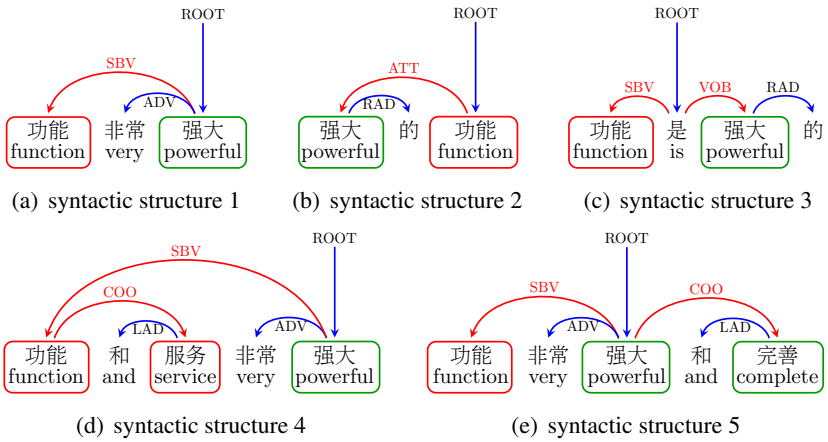
- We present a framework of using sentiment sentence compression preprocessing step to improve T-P collocation extraction. This framework can better solve the "over-natural" problem of sentiment sentences, which poses a challenge to syntactic parsers. More importantly, the idea of this framework can be applied to some 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.

2 Background

For our baseline system, we used the state-of-the-art method to extract T-P collocations introduced by Qiu et al. (2011), who proposed a double propagation method. This idea is based on the observation that there is a natural syntactic relationship between polarity words and targets owing to the fact that polarity words are used to modify targets. Furthermore, they also found that polarity words and targets themselves have relations in some sentiment sentences (Qiu et al., 2011).

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



(a) syntactic structure 1 (b) syntactic structure 2 (c) syntactic structure 3

(d) syntactic structure 4 (e) syntactic structure 5

Figure 2: Example of syntactic structure rules for T-P collocation extraction. We showed five examples from a total of nine syntactic structures. For each kind of syntactic structure (a) to (e), the target 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 targets and polarity words. Syntactic structure (d), which is extended from (a), describes the relation between two targets. 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 targets or two polarity words.

We can see that the syntactic relations are important for this method, and Qiu et al. (2011) proposed eight rules to describe these relations. However, their work only focused on English sentences, whereas the relations for Chinese sentences are different. Thus, in accordance with Chinese grammar, we proposed nine syntactic structure rules between target t and polarity word p in a Chinese T-P collocation $\langle t, p \rangle$.[2] The three main rules are shown below and some example rules are illustrated in Figure 2.

Rule 1: $t \overset{\text{SBV}}{\curvearrowleft} p$, the "subject-verb" structure between t and p, such as the example in Figure 2(a).

Rule 2: $p \overset{\text{ATT}}{\curvearrowleft} t$, that p is an attribute for t, such as the example in Figure 2(b).

Rule 3: $t \overset{\text{SBV}}{\curvearrowleft} \circ \overset{\text{VOB}}{\curvearrowright} p$, the "subject-verb-object" structure between t and p, such as the example in Figure 2(c). The $\circ$ denotes any word.

The other six rules can be extended from the three main rules by obtaining the coordination (COO) relation of t or p, such as $t \overset{\text{SBV}}{\curvearrowleft} \circ \overset{\text{COO}}{\curvearrowright} p$ in Figure 2(e). Note that the POS for t should be noun and for p should be adjective.

As described above, the T-P collocation extraction relies heavily on syntactic parsers. Hence, if we can use the *Sent_Comp* model to improve the performance of parsers, the performance of T-P collocation extraction can also be improved accordingly.

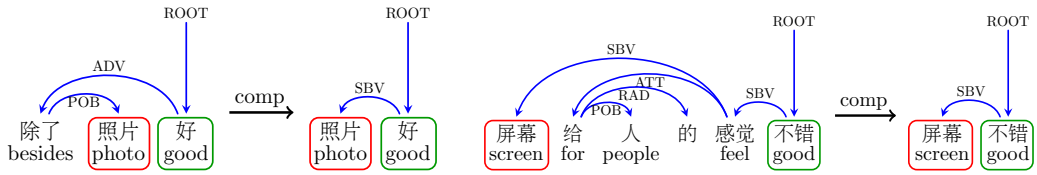
3 Sentiment Sentence Compression

3.1 Problem Analysis

First, we conducted an error analysis for the results of current T-P collocation extraction, from which we observed that the "naturalness" of sentiment sentences is one of the main problems. For examples:

- Chatty form: some sentiment sentences are so chatty, that they bring many difficulties to the parser. For example, in the sentence "多亏键盘好" (*fortunately the keyboard is good*) shown in Figure 1, the usage of the chatty word "多亏" (*fortunately*) affects the accuracy of the syntactic parser.

[2] A Chinese natural language processing toolkit, Language Technology Platform (LTP) (Che et al., 2010), was used as our dependency parser. More information about the syntactic relations can be found in their paper. The state-of-the-art graph-based dependency parsing model, in the toolkit, was trained on Chinese Dependency Treebank 1.0 (LDC2012T05).



(a) parse tree 1 before and after compression

(b) parse tree 2 before and after compression

Figure 3: "Naturalness" problem of sentiment sentences.

- Conjunction word usage: conjunction words are often used in sentiment sentences to show the discourse relations between two sentences. However, there are so many conjunction words in Chinese, some of which can cause errors among parsers. For example, in Figure 3(a), the parse tree of 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*).

- Feeling words/phrase usage: in sentiment sentences, people often use some feeling words/phrase, such as "给人的感觉" (*feel like*) in Figure 3(b) or "闻起来" (*smell like*). Given that the current syntactic parser cannot handle the feeling words/phrases very well, the T-P collocation ⟨屏幕, 不错⟩ (⟨$screen, good$⟩) in Figure 3(b) cannot be extracted correctly.

To address the "naturalness" problem, we compressed the sentiment sentences into one that are shorter and easier to parse. Similar to the examples in Figure 1 and 3, the compressed sentences can be easily and correctly parsed. The above analysis can be used as the criteria to guide us in compressing sentiment sentences when annotating, and can also help us exploit more useful features for automatic sentiment sentence compression.

3.2 Task Definition

We focus on studying the methods for extractive sentence compression.[3] Formally, 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 $y_i \in \{x_1, \cdots, x_n\}$, $m \leq n$.

In this paper, similar to Nomoto (2007), we also treated the sentence compression as a sequence labeling task which can be solved by a CRF model. We assigned a compression tag t_i to each word x_i in an original sentence $\mathbf{x}$, where $t_i = \mathsf{N}$ if $x_i \in \mathbf{y}$, else $t_i = \mathsf{Y}$.

A first-order linear-chain CRF is used which defines the following conditional probability:

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

where $\mathbf{x}$ and $\mathbf{t}$ are the input and output sequences respectively, $Z(\mathbf{x})$ is the partition function, and M_i is the clique potential for edge clique i. Here, we used the CRFsuite toolkit to train the CRF model.[4]

3.3 Features

The features for *Sent_Comp* are listed in Table 1. Aside from the basic word (w), POS tag (t) and their combination context features (01 – 04), we introduced some sentiment-related features (05 – 06) and latent semantic features (07 – 08) to better handle sentiment analysis data and generalize word features. Then we added the syntactic parse features (09), which are commonly used in traditional sentence compression task.

One sentiment-related feature ($\mathsf{feeling}(\cdot)$) indicates whether a word is a feeling word, which is inspired by the naturalness problem in Figure 3(b). As discussed above, the current parser often produces wrong parse trees because of these feeling words. Therefore, the feeling words tend to be removed from a

[3]Generally, there are two kinds of sentence compression methods: extractive method and abstractive method. Because abstractive method needs more resource and is more complicated, in this paper, we only focus on extractive approach.

[4]www.chokkan.org/software/crfsuite/

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: feeling(w_i)
06: polarity(w_i)
Latent Semantic Features
07: suffix(w_i) if $t(w_i) == n$ else prefix(w_i)
08: cluster(w_i)
Syntactic Features
09: dependency(w_i)

Table 1: Features of sentiment sentence compression

sentiment sentence for *Sent_Comp*. We can obtain a feeling word lexicon from HowNet,[5] a popular Chinese sentiment thesaurus, where a feeling word is defined by DEF={perception|感知} tag. Finally, we collected 38 feeling words, such as 发觉 (*realize*), 发现 (*find*), and 认为 (*think*).

The other sentiment-related feature (polarity$(\cdot)$) indicates whether a word is a polarity word. One of the main differences between a sentiment sentence and a formal sentence is that the former often contains polarity words. In contrast to the features of feeling$(\cdot)$, polarity words (e.g., "great" in the sentence "Overall, this is a great camera") tend to be retained, because they are important and special to sentiment analysis. In this paper, we treat polarity words as important features, considering that they are often tagged as modifiers and are easily removed by common sentence compression methods. We can obtain the polarity feature (polarity$(\cdot)$) from a polarity lexicon, which can also be obtained from HowNet.

To generalize the words in sentiment sentences, we proposed two kinds of semantic features. The first one is a suffix or prefix character feature (prefix$(\cdot)$ or suffix$(\cdot)$). In contrast to English, the suffix (for noun) or prefix (for non noun) characters of a Chinese word often carry that word's core semantic information. For example, 自行**车** (*bicycle*), 汽**车** (*car*), and 火**车** (*train*) are all various kinds of 车 (*vehicle*), which is also the suffix of the three words. Given that all of them may become targets, they tend to be retained in compressed sentences. The verbs, **感**觉 and **感**到, can be denoted by their prefix *feel* (感), and can be removed from original sentences because they are feeling words.

We used word clustering features (cluster$(\cdot)$) as the other latent semantic feature to further improve the generalization over common words. Word clustering features contain some semantic information and have been successfully used in several natural language processing tasks, including NER (Miller et al., 2004; Che et al., 2013) and dependency parsing (Koo et al., 2008). For instance, the words 外观 and 样子 (*appearance*) belong to the same word cluster, although they have a different suffix or prefix. Both words are important for T-P collocation extraction and should be retained. We used the Brown word clustering algorithm (Brown et al., 1992) to obtain the word clusters (Liang, 2005). Raw texts were obtained from the fifth edition of Chinese Gigaword (LDC2011T13).

Finally, similar to McDonald (2006), we also added the dependency relation between a word and its parent as the syntactic features. Intuitively, the dependency relations are helpful in carrying out 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.

4 Experiments

4.1 Experimental Setup

4.1.1 Corpus

We conducted the experiments on a Chinese corpus of four product domains, which came from the Task3 of the Chinese Opinion Analysis Evaluation (COAE) (Zhao et al., 2008).[6] Table 2 describes the corpus,

[5] www.keenage.com

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

Domain	# reviews	# sentences	# collocations
Camera	138	1,249	1,335
Car	161	1,172	1,312
Notebook	56	623	674
Phone	123	1,350	1,479
All	478	4,394	4,800

Table 2: Corpus statistics for the Chinese corpus of four product domains.

where 4,394 sentiment sentences containing 4,800 T-P collocations are manually found and annotated from 478 reviews.

We ask annotators to manually compress all the sentiment sentences. Specifically, the annotators removed some words from a sentiment sentence according to two criteria stated as follows: (1) removing the word should not change the essential content of the sentence, and (2) removing the word should not change the sentiment orientation of the sentence. In order to assess the quality of the annotation, we sampled 500 sentences from this corpus and asked two annotators to perform the annotation. The resulting word-based Cohen's kappa (Cohen, 1960) (i.e., a measure of inter-annotator agreement ranging from zero to one) of 0.7 indicated a good strength of agreement.

4.1.2 Evaluation

Generally, compressions are evaluated using three criteria (McDonald, 2006), namely, grammaticality, importance, and compression rate. Obviously, the former two are difficult to evaluate objectively. Previous works used human judgment, which entails a difficult and expensive process. In this paper, similar to a common sequence labeling task, we simply used the F-score metric of removed words to roughly evaluate the performance of sentiment sentence compression. Of course, the final effectiveness of sentence compression model can be reviewed by the derived T-P collocation extraction task.

For T-P collocation extraction, we applied the traditional *P*, *R* and *F-score* for the final evaluations. Specially, a fuzzy matching evaluation is adopted for the T-P collocation extraction. That is to say, given an extracted T-P collocation $\langle t, p \rangle$, whose standard result is $\langle t_s, p_s \rangle$, if t is the substring of t_s, and meanwhile p is the substring of p_s, we consider the extracted $\langle t, p \rangle$ is a correct T-P collocation.

4.2 Sentiment Sentence Compression Results

Features	P(%)	R(%)	F(%)
Basic (01 – 04)	76.4	57.4	65.5
+ feeling (05)	75.9	57.6	65.5
+ polarity (06)	76.6	57.6	65.7
+ suffix or prefix (07)	78.4	56.9	66.0
+ cluster (08)	74.9	58.9	65.9
+ dependency (09)	75.3	57.2	65.0
All (01 – 08)	**77.3**	**59.1**	**67.0**
All - feeling (05)	77.1	58.9	66.8

Table 3: The results of sentiment sentence compression with different features.

Results of *Sent_Comp* with different features are shown in Table 3. All results are reported using five-fold cross validation. We can see that the performance is improved when we added feeling[7] and polarity features (05 – 06) respectively, indicating that the sentiment-related features are useful for sentiment sentence compression. In addition, the latent semantic features (07 – 08) are also helpful, especially the suffix or prefix features, which show better performance than the four other kinds of features.

Nonetheless, the dependency features (09) have a negative on compression performance due to the specificity of compression for sentiment sentences. That is because the lower dependency parsing performance on sentiment sentences introduces many wrong dependency relations, which counteract the

[7]In Table 3, although the performance of adding feeling is comparative to the basic system (Basic (01-04)), the system without feeling (All - feeling (05), the last line) is worse than the system using all the features (All (01-08)). This can illustrate the effectiveness of the feeling feature.

Domain	Method	P(%)	R(%)	F(%)
Camera	no_Comp	74.7	58.4	65.6
	manual_Comp	83.4	62.7	71.6
	auto_Comp	80.4	62.1	70.1
Car	no_Comp	68.2	53.1	59.7
	manual_Comp	76.3	57.7	65.7
	auto_Comp	72.3	56.1	63.2
Notebook	no_Comp	74.1	56.8	64.3
	manual_Comp	82.7	64.5	72.5
	auto_Comp	79.7	62.8	70.2
Phone	no_Comp	77.3	60.9	68.1
	manual_Comp	82.7	65.7	73.2
	auto_Comp	80.3	63.3	70.8
All	no_Comp	73.7	57.5	**64.6**
	manual_Comp	81.2	62.5	**70.6**
	auto_Comp	78.1	60.9	**68.4**

Table 4: Results on T-P collocation extraction for four product domains.

contribution of the dependency relation features. This is also the reason why we need to compress sentiment sentences as the first step for T-P collocation extraction. Finally, when we combine all of useful features (01 – 08), the performance achieves the highest score.

It is worth noting that 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. We will polish the *Sent_Comp* model in the future work.

4.3 *Sent_Comp* for T-P Collocation Extraction

We designed three comparative systems to demonstrate the effectiveness of *Sent_Comp* for T-P collocation extraction. Note that, *Sent_Comp* is the first step to process the corpus before T-P collocation extraction. The method for T-P collocation extraction was based on the state-of-the-art method proposed by Qiu et al. (2011) as described in Section 2.

no_Comp - This refers to the system that only uses the T-P collocation extraction method and does not perform sentence compression as the first step.

manual_Comp - This system **manually** compresses the corpus into a new one as the first step, and then applies the T-P collocation extraction method on the new compressed corpus.

auto_Comp - This system uses *Sent_Comp* as the first step to **automatically** compress the corpus into a new one, and then applies the T-P collocation extraction method on the new corpus.

From the descriptions above, we can draw a conclusion that the performance of **manual_Comp** can be considered as the upper bound for the sentiment sentence compression based T-P collocation extraction task.

Table 4 shows the experimental results of the three systems on T-P collocation extraction for four product domains. Here, **manual_Comp** can significantly ($p < 0.01$) improved the *F-score* by approximately 6%,[8] compared with **no_Comp**. This illustrates that the idea of sentiment sentence compression is useful for T-P collocation extraction. Specifically, the proposed method can transform some over-natural sentences into normal ones, further influencing their final syntactic parsers. Evidently, because the T-P collocation extraction relies heavily on syntactic features, the more correct syntactic parse trees derived from the compressed sentences can help to increase the performance of this task.

Compared with **no_Comp**, the **auto_Comp** system also yielded a significantly better results ($p < 0.01$) that indicated an improvement of 3.8% in the *F-score*, despite the fact that the automatic sentence compression model *Sent_Comp* may wrongly compress some sentences. This demonstrates the usefulness of sentiment sentence compression step in the T-P collocation extraction task and further proves the effectiveness of our proposed model.

[8]We use paired bootstrap resampling significance test (Efron and Tibshirani, 1993).

Moreover, we can observe that the idea of sentence compression and our *Sent_Comp* are useful for all the four product domains on T-P collocation extraction task, indicating that *Sent_Comp* is domain adaptive. However, we can find 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.

5 Related Works

5.1 Sentiment Analysis

T-P collocation extraction is a basic task in sentiment analysis. In order to solve this task, most methods focused on identifying relationships between targets and polarity words. In early studies, researchers recognized the target first, and then chose its polarity word within a window of size k (Hu and Liu, 2004). However, considering that this kind of method is too heuristic, the performance proved to be very limited. To tackle this problem, many researchers found syntactic patterns that can better describe the relationships between targets and polarity words. For example, Bloom et al. (2007) constructed a linkage specification lexicon containing 31 patterns, while Qiu et al. (2011) proposed a double propagation method that introduced eight heuristic syntactic patterns to extract the collocations. Xu et al. (2013) used the syntactic patterns to extract the collocation candidates in their two-stage framework.

Based on the above, we can conclude that syntactic features are very important for T-P collocation extraction. However, the "naturalness" problem can still seriously affect the performance of syntactic parser. Once our sentiment sentence compression method can improve the quality of parsing, the performance of T-P collocation extraction task can be improved as well. Note that, to date, there is no previous work using a sentence compression model to improve this task.

5.2 Sentence Compression

Sentence compression is a paraphrasing task aimed at generating sentences shorter than the given ones, while preserving the essential content (Jing, 2000). There are many applications that can benefit from a robust compression system, such as summarization systems (Li et al., 2013), semantic role labeling (Vickrey and Koller, 2008), relation extraction (Miwa et al., 2010) and so on.

Commonly used to compress sentences, tree-based approaches (Knight and Marcu, 2002; Turner and Charniak, 2005; Galley and McKeown, 2007; Cohn and Lapata, 2009; Galanis and Androutsopoulos, 2010; Woodsend and Lapata, 2011; Thadani and McKeown, 2013) compress a sentence by editing 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 good compressed sentence. McDonald (2006), Nomoto (2007), and Clarke and Lapata (2008) tried to solve the problem by using discriminative models.

Aside from above *extractive* sentence compression approaches, there is another research line, namely, *abstractive* approach, which compresses an original sentence by reordering, substituting, and inserting, as well as removing (Cohn and Lapata, 2013). This method needs more resource and is more complicated. Therefore, in this paper, we only focus on *extractive* approach.

At present, the current sentence compression methods all focus on formal sentences, and few methods are being proposed to study sentiment sentences. As discussed in the above sections, the current compression models cannot be directly utilized to T-P collocation extraction owing to the specificity of sentiment sentences. Therefore, a new compression model for sentiment sentences should be established.

6 Conclusion and Future Work

In this work, we presented a framework that adopted a CRF based sentiment sentence compression model *Sent_Comp*, as a preprocessing step, to improve the T-P collocation extraction task. Different from the existing sentence compression models used for formal sentences, *Sent_Comp* incorporated some sentiment-related features to retain the sentiment information. Experimental results showed that the system with the sentence compression step performed better than that without this step, thus demonstrating the effectiveness of the framework and the compression model *Sent_Comp*.

Generally, the idea of this framework maybe useful for many sentiment analysis tasks that rely heavily on syntactic results. Thus in the future, we will try to apply the *Sent_Comp* model for these tasks. Besides, the simplicity and effectiveness of this framework motivates us to pursue the study further. For example, we will polish the *Sent_Comp* model by exploring more sentiment-related features and exploring other types of compression models.

Acknowledgments

We thank the anonymous reviewers for their helpful comments. This work was supported by National Natural Science Foundation of China (NSFC) via grant 61300113, 61133012 and 61273321, the Ministry of Education Research of Social Sciences Youth funded projects via grant 12YJCZH304, the Fundamental Research Funds for the Central Universities via grant No.HIT.NSRIF.2013090 and IBM Research-China Joint Research Project.

References

Ahmed Abbasi, Hsinchun Chen, and Arab Salem. 2008. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Trans. Inf. Syst.*, 26(3):12:1–12:34, June.

Kenneth Bloom, Navendu Garg, and Shlomo Argamon. 2007. Extracting appraisal expressions. In *HLT-NAACL 2007*, pages 308–315.

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Comput. Linguist.*, 18(4):467–479, December.

Wanxiang Che, Zhenghua Li, and Ting Liu. 2010. Ltp: A chinese language technology platform. In *Coling 2010: Demonstrations*, pages 13–16, Beijing, China, August. Coling 2010 Organizing Committee.

Wanxiang Che, Mengqiu Wang, Christopher D. Manning, and Ting Liu. 2013. Named entity recognition with bilingual constraints. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 52–62, Atlanta, Georgia, June. Association for Computational Linguistics.

James Clarke and Mirella Lapata. 2008. Global inference for sentence compression: An integer linear programming approach. *J. Artif. Intell. Res. (JAIR)*, 31:399–429.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37 – 46.

Trevor Cohn and Mirella Lapata. 2009. Sentence compression as tree transduction. *Journal of Artificial Intelligence Research*, 34:637–674.

Trevor Cohn and Mirella Lapata. 2013. An abstractive approach to sentence compression. *ACM Transactions on Intelligent Systems and Technology*, 4(3):1–35.

Adnan Duric and Fei Song. 2012. Feature selection for sentiment analysis based on content and syntax models. *Decis. Support Syst.*, 53(4):704–711, November.

B. Efron and R. J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman & Hall, New York.

Jenny Rose Finkel, Alex Kleeman, and Christopher D. Manning. 2008. Efficient, feature-based, conditional random field parsing. In *Proceedings of ACL-08: HLT*, pages 959–967, Columbus, Ohio, June. Association for Computational Linguistics.

Dimitrios Galanis and Ion Androutsopoulos. 2010. An extractive supervised two-stage method for sentence compression. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 885–893, Los Angeles, California, June. Association for Computational Linguistics.

Michel Galley and Kathleen McKeown. 2007. 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*, pages 180–187, Rochester, New York, April. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of KDD-2004*, pages 168–177.

Hongyan Jing. 2000. Sentence reduction for automatic text summarization. In *IN PROCEEDINGS OF THE 6TH APPLIED NATURAL LANGUAGE PROCESSING CONFERENCE*, pages 310–315.

Kevin Knight and Daniel Marcu. 2002. Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artif. Intell.*, 139(1):91–107, July.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing. In *Proceedings of ACL-08: HLT*, pages 595–603, Columbus, Ohio, June. Association for Computational Linguistics.

Chen Li, Fei Liu, Fuliang Weng, and Yang Liu. 2013. Document summarization via guided sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 490–500, Seattle, Washington, USA, October. Association for Computational Linguistics.

Percy Liang. 2005. Semi-supervised learning for natural language. Master's thesis, MIT.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Ryan McDonald. 2006. Discriminative sentence compression with soft syntactic evidence. In *In Proc. EACL*.

Scott Miller, Jethran Guinness, and Alex Zamanian. 2004. Name tagging with word clusters and discriminative training. In Daniel Marcu Susan Dumais and Salim Roukos, editors, *HLT-NAACL 2004: Main Proceedings*, pages 337–342, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Makoto Miwa, Rune Saetre, Yusuke Miyao, and Jun'ichi Tsujii. 2010. Entity-focused sentence simplification for relation extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 788–796.

Tadashi Nomoto. 2007. Discriminative sentence compression with conditional random fields. *Information Processing and Management*, 43(6):1571–1587, November.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, January.

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2011. Opinion word expansion and target extraction through double propagation. *Computational Linguistics*, 37(1):9–27.

Kapil Thadani and Kathleen McKeown. 2013. Sentence compression with joint structural inference. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 65–74, Sofia, Bulgaria, August. Association for Computational Linguistics.

Jenine Turner and Eugene Charniak. 2005. Supervised and unsupervised learning for sentence compression. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 290–297, Stroudsburg, PA, USA. Association for Computational Linguistics.

David Vickrey and Daphne Koller. 2008. Sentence simplification for semantic role labeling. In *ACL*, pages 344–352. The Association for Computer Linguistics.

Kristian Woodsend and Mirella Lapata. 2011. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 409–420, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Liheng Xu, Kang Liu, Siwei Lai, Yubo Chen, and Jun Zhao. 2013. Mining opinion words and opinion targets in a two-stage framework. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1764–1773, Sofia, Bulgaria, August. Association for Computational Linguistics.

Jun Zhao, Hongbo Xu, Xuanjing Huang, Songbo Tan, Kang Liu, and Qi Zhang. 2008. Overview of chinese pinion analysis evaluation 2008. In *The First Chinese Opinion Analysis Evaluation (COAE) 2008*.