Feature Extraction and Opinion Summarization in Chinese Reviews

Wang Ge[1,*], Pu Pengbo[2] and Liang Yongquan[1]

[1]*College of Information Science and Engineering, Shandong University of Science and Technology, Shandong, Qingdao, P.R. China*

[2]*Department of Information Engineering, Shandong University of Science and Technology, Shandong, Taian, P.R. China*

Abstract: The paper describes the process of mining opinions from Chinese reviews of products sold online. The structure of Chinese reviews is free, which leads to a more complicated relationship between opinions and features. The paper introduces two main steps of opinion mining: feature extraction and opinion direction identification. The feature extraction function first extracts "hot" features that a lot of people have expressed their opinions in their reviews, and then finds those infrequent ones. In order to improve the accuracy of the experiment, redundant features are removed. The opinion direction identification function takes the generated features and summarizes the opinions into two categories: positive and negative. We extract adjectives and negative adverbs as opinion words and use the Naïve Bayes classifier to identify their direction. By direction, we mean whether an opinion is positive or negative.

Keywords: Opinion mining, Feature extraction, Opinion direction identification, Sentiment polarity.

1. INTRODUCTION

With the dramatic growth of web applications, more and more products are sold on the web. The number of freely available online reviews is increasing at a high speed. As customer feedback on the Web influences other customer's decision, these feedbacks have become an important source of information for businesses to take into account when developing marketing and product development plans. Some popular products can get hundreds of reviews or more, and some reviews are also long. So, it is difficult for a potential customer to read them to make an informed decision on whether to purchase the product. The large number of reviews also makes it difficult for product manufacturers or businesses to keep track of customer opinions and sentiments on their products and services. The situation is also notable in Chinese web services. It is thus highly desirable to produce a summary of reviews. As a result, the problem of "opinion mining" has seen increasing attention over the last years from [1, 2] and many others.

Opinion mining has become a significant subject of research in the field of data mining. The ultimate goal of opinion mining is to extract customer opinions on products and present the information in the most effective way. Many researchers have used different techniques to summarize the information and present it. For example, if we have a number of reviews about a given product, we want to know the number of negative and positive reviews. Classifying each review as negative or positive is the most important task.

And further, if we want to know a customer opinion on each of the different features of a product, it is necessary to extract product features and analyze the overall sentiment on each feature.

In this research, we study the problem of mining opinions from Chinese customer reviews of products sold online. Chinese reviews on the Internet lack standardization. People describe their opinions in Chinese using omission and free structure, which leads to a more complicated relationship between opinions and features. So it is necessary to achieve a deeper and more detailed understanding of the reviews, and analyze the underlying sentiment on each feature of a product. This paper involves three subtasks: (1) identifying features of the product that customers have expressed their opinions on (call product features); (2) identifying opinion words and deciding whether each opinion word is positive or negative; (3) summarizing the results.

The remainder of the paper is organized as follows: Section 2 introduces the related work of opinion mining. Section 3 describes the structure of opinion mining system. The first step in the opinion mining system is Chinese word segmentation. There is no sharp demarcation between words, which is different from English. In most cases Chinese word segmentation depends on the sentiment expression. So Chinese word segmentation is very complicated. Our work is not involved in this part. We use the existing word segmentation technique to finish this work. We introduce the work in Section 4. Section 5 describes the work of feature extraction and feature pruning. Section 6 introduces opinion word identification and sentiment orientation decision. In our work, we extract adjectives and negative adverbs as opinion words, and we use the Naïve Bayes classifier to decide sentiment

*Address correspondence to this author at the Department of Information Engineering, Shandong University of Science and Technology, Shandong, Taian, 271000, P.R. China; Tel: 13515389715; E-mail: wanggeg@163.com

1874-4443/15 2015 Bentham Open

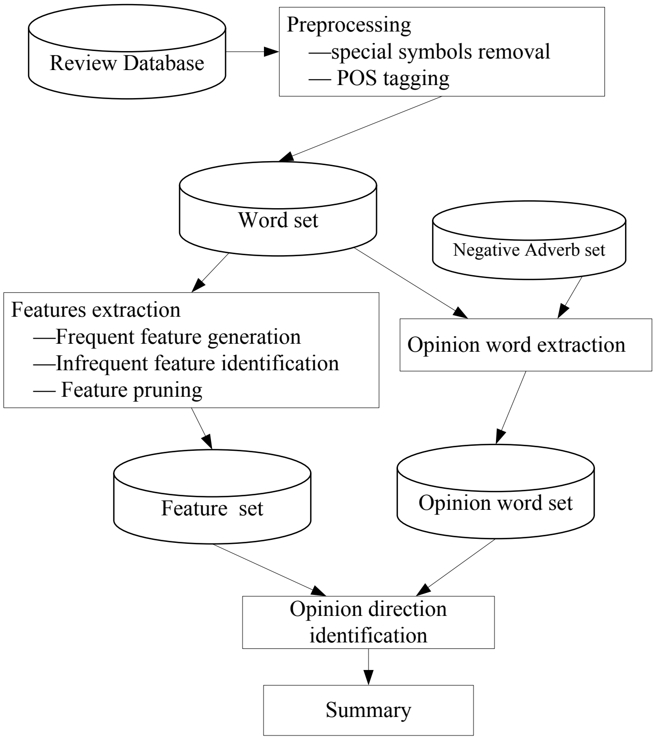

Fig. (1). Architecture of opinion summarization system.

polarity of each opinion word. Section 7 shows the results of the experiments. Finally, Section 8 draws the conclusion.

2. RELATED WORK

With the fast growing development of blogs, social networks and e-commerce, opinion mining and sentiment analysis became a field of interest for many researches. A very broad overview of the existing work was presented in [3-7]. In their survey, the authors describe existing techniques and approaches, applications, future research areas, and future challenges in opinion mining. Opinion mining and sentiment analysis actually focus on polarity detection and feature-based opinion mining. Both fields use data mining and natural language processing (NLP) techniques to discover, retrieve and distill information and opinions from vast textual information. Many researchers attempt different techniques to detect the polarity of reviews. Polarity detection is classifying each review document as positive, negative, or neutral. [8] presents an unsupervised learning algorithm, called PMI-IR, to measure the similarity of pairs of words. In order to find out whether any new adjectives are associated with the same opinion polarity as the seed words, [9] adapts the association rule algorithm to compute the similarity of two adjectives. The closer a given adjective is to a seed adjective, the greater the similarity in sentiment orientation between the two adjectives. [2] utilizes the adjective synonym set and antonym set in WordNet to predict the sentiment orientation of adjectives. [10] uses Naïve Bayes to classify the polarity.

Most research on sentiment focuses on text written in English. Comparing with previous researches for English review, opinion mining for Chinese product reviews is a more challenging task due to the complexity of Chinese sentiment expression and the limited resources of Chinese sentiment analysis. For Chinese online product reviews, [11] collects core concepts as fuzzy sets and utilize propagation rules to extract product features and sentiment words. They choose the candidate sentiments which are close (within a text window of 5 size) to the product features and find out which word has the max similarity with the determined word to determine the word belonging to certain type of sentiment. Their research don't consider the meaning of a word in the sentence. For example, "电池寿命长 (Battery life is long)" and "电脑启动需要很长时间 (The computer takes a long time to start up)", "long" in the two sentences has opposite meanings. [12] clusters product features and opinion words simultaneously to find out the sentiment association between the two groups of data objects. In their work, the extracted opinion word is adjectives, and don't consider negative adverbs. We think negative adverbs that modify adjectives can affect the polarity of opinion words. In our work, we extract negative adverbs and adjectives in the step of opinion extraction.

3. THE OPINION SUMMARIZATION SYSTEM

Fig. (**1**) gives an architectural overview for our opinion summarization system. The system performs the summarization in two main steps: feature extraction and opinion direction identification. We use the existing corpus built by Songbo Tan *et al.* and put it in the review database. Preprocessing cleans up the raw data such as special words (such as "@, #") removal, Part-of-Speech (POS) tagging. The feature extraction function, which is the focus of this paper, first extracts “hot” features that a lot of people have expressed their opinions on in their reviews, and then finds those infrequent ones. In order to improve the accuracy of the experiment, redundant features are removed. The opinion direction identification function takes the generated features and summarizes the opinions into two categories: positive and negative. We extract negative adverbs and adjectives as opinion words, and put them in opinion words set. We use the Naïve Bayes classifier to identify the direction of opinion words. By direction, we mean whether an opinion is positive or negative.

4. POS

Chinese texts need to be segmented into words before extracting candidate terms. But there is no space or other symbols between words, which is different from English. So word segmentation of Chinese texts is very important. This is the first step in Chinese opinion mining. In our research, we use LTP-Cloud (Language Technology Platform Cloud) finishing word segmentation. LTP-Cloud [13], which is developed by the Research Center for Social Computing and Information Retrieval at Harbin Institute of Technology (HIT-SCIR). LTP-Cloud can show the segmentation result in XML form. It is easy for further applications. For example, “这款电脑符合我的要求（This notebook meets my requirements.）”, the segmentation result by LTP-Cloud is as follows:

```
<xml4nlp>
<note sent="y" word="y" pos="y" ne="y" parser="y" wsd="n" srl="y"/>
<doc>
<para id="0">
<sent id="0" cont="这款电脑符合我的要求。">
<word id="0" cont="这" pos="r" ne="O" parent="1" relate="ATT"/>
<word id="1" cont="款" pos="q" ne="O" parent="2" relate="ATT"/>
<word id="2" cont="电脑" pos="n" ne="O" parent="3" relate="SBV"/>
<word id="3" cont="符合" pos="v" ne="O" parent="-1" relate="HED">
<arg id="0" type="A0" beg="0" end="2"/>
<arg id="1" type="A1" beg="4" end="6"/>
</word>
<word id="4" cont="我" pos="r" ne="O" parent="6" relate="ATT"/>
<word id="5" cont="的" pos="u" ne="O" parent="4" relate="RAD"/>
<word id="6" cont="要求" pos="n" ne="O" parent="3" relate="VOB"/>
<word id="7" cont="。" pos="wp" ne="O" parent="3" relate="WP"/>
</sent>
</para>
</doc>
</xml4nlp>
```

Where *pos* is the part-of-speech tagging, *ne* is named entity. *parent* and *relate* are always in pairs, in which *parent* is the number of father node based on dependency parsing and *relate* is their relationship. As is shown in Fig. (**2**), the segmentation result by LTP-Cloud can be shown in a tree.

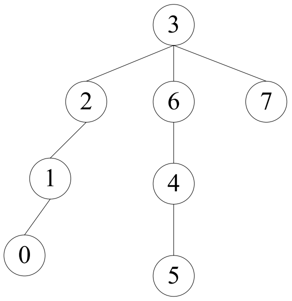

Fig. (2). Syntax trees.

5. FEATURE EXTRACTION

5.1. Frequent Feature Extraction

We select 1000 positive and negative reviews from the corpus to extract features and opinion words. Text preprocessing needs to be done before extracting candidate terms. This work can improve the effectiveness of further operations. Text pre-processing includes:

1. Remove the special words (such as “@, #”) in the corpus;
2. Remove URL links (e.g. http://abc.com) in the corpus;
3. Remove spaces, punctuation marks and stop words in the corpus.

Nouns/noun phrases from the reviews are the features that customer express their opinions on. We calculate the frequency of nouns/noun phrases in the selected reviews. In our work, if a noun/noun phrase appears in more than 0.5%(minimum support) of the review sentences, we call it the candidate frequent feature, and store it to the feature set for further processing. Table **1** shows the results of frequent feature extraction.

5.2. Infrequent Feature Identification

For infrequent feature, our approach is similar to the Double Propagation algorithm [14]. There are two steps: 1) using opinion words extracts features; 2) using existing features extracts features. We consider adjectives and negative adverbs as the candidate sentiments and nouns (n) or noun phrases (np) as the candidate features. Infrequent features

Table 1. Result of frequent feature extraction.

Product Name	Precision	Recall	F-score
Notebook	0.532	0.680	0.597
Hotel	0.674	0.766	0.717
Book	0.645	0.732	0.686

Table 2. Infrequent feature extraction rules.

	Extraction Rules	Constraints
$R1$	if $\exists(CW = s, R, PW) \rightarrow f = PW$	$R \in \{ADV, SBV\}$, $CW \in \{S\}$, $POS(PW) \in \{N\}$
	if $\exists(CW, R, PW = s) \rightarrow f = CW$	$R \in \{ADV, SBV\}$, $POS(CW) \in \{N\}$, $PW \in \{S\}$
$R2$	if $\exists(CW = f, R, PW) \rightarrow f = PW$	$R \in \{ATT, RAD, COO\}$, $POS(CW, PW) \in \{N\}$
	if $\exists(CW, R, PW = f) \rightarrow f = CW$	$R \in \{ATT, RAD, COO\}$, $POS(CW, PW) \in \{N\}$

Table 3. Result of infrequent feature identification

Product Name	Precision	Recall	F-score
Notebook	0.732	0.713	0.722
Hotel	0.745	0.794	0.769
Book	0.737	0.809	0.771

are extracted. Table **2** shows extraction rules based on dependency parsing of LTP-Cloud.

In Table **2**, CW (or PW) stands for the word of a child node (or parent node). s means the extracted sentiment word and f means the extracted feature. $\{S\}$ stands for the known sentiment words. $\{N\}$ are sets of POS tags of potential features. In this work, we consider sentiment words to be adjectives and negative adverbs, and features to be nouns or noun phrases. R means the relate attribute of a parent node and its child node in LTP tree. Precision and Recall for the infrequent feature identification are shown in Table **3**.

5.3. Feature Pruning

Not all extracted features are useful or genuine. There are also some redundant ones. Feature pruning aims to remove these duplicate features. Table **4** shows the result of feature pruning.

Our work in feature pruning is as follows:

(1) For two or more consecutive nouns (denoted by NS), we select the rightmost ones as features. For example, "电脑屏幕(computer screen)" or "电脑的屏幕(computer's screen)", we select "screen" as the feature and store it to feature set. NS has two types of sequence structure:

$$\gamma_1 : \ nn, \ nn = n_1 n_2 ... n_n \tag{1}$$

$$\gamma_2 : \ nn_i \text{的}(of) nn_j \tag{2}$$

The rule is explained as follows:

$$f = Right(NS) = \begin{cases} n_n & NS = nn \\ Right(nn_j) & otherwise \end{cases} \tag{3}$$

Table 4. Result of feature pruning.

Product Name	Precision	Recall	F-score
Notebook	0.824	0.769	0.796
Hotel	0.793	0.824	0.808
Book	0.778	0.813	0.795

Where nn stands for two or more consecutive nouns, that is $n_1n_2...n_n$, $POS(n_i) \in \{N\}$. $Right(NS)$ is selecting the rightmost noun in NS .

(2) It is very common that different words are used to describe the same object in Chinese reviews. We built a word set (e.g. 货, 宝贝, 东西). These words are frequently seen in reviews and they have the same meaning that is the product customers buy.

6. TRAINING THE CLASSIFIER

In review corpus, the words with sentiment polarity are mainly adjectives and adverbs. Adjectives are used more often for expressing emotions and opinions. Adverbs are mostly used in subjective texts to give an emotional color to an adjective or verb. In our work, adverbs are divided into two types. One is used to express the strength of feeling (e.g. very and quite). This type of adverbs have no effect on sentiment polarity. The other is negative adverbs (e.g. no and not). This type of adverbs affect the opinion direction. So we select adjectives and negative adverbs as opinion words in the paper.

We built a negative adverb set (denoted by NAs) which contains 11 negative adverbs by studying the review corpus, NAs ={不, 不要, 没, 没有, 无, 未, 白, 不必, 无从, 无需, 非 }. A negative adverb itself can not express a certain opinion except for combining with adjectives they modify. So we put negative adverbs together with adjectives they modify to study their sentiment polarity.

We find the sentiment orientation of a word w in the context of an associated feature f and sentence $sent$ .We restate this task as follows:

Given a set of sentiment orientation (SO) labels({ $positive$, $negative$ }), a set of reviews and a set of tuples T=($w, f, sent$), where w is a potential opinion word associated with feature f in sentence $sent$, assign a SO label to each tuple ($w, f, sent$).

Candidate opinion words are denoted by a set of tuples $w=(a,r,n)$, where a is an adjective, n is a negative adverb, and r is the relationship between a and n .

$$r = \begin{cases} 1 & \textit{if n is the child node of a in the } \text{syntax tree} \\ 0 & \textit{otherwise} \end{cases} \tag{4}$$

If $r=1$, it shows adjective a is modified by negative adverb n in (a,r,n). a and n together express a certain opinion. We take a and n as a word denoted by w' , $w'=a\,\&\,n$. If $r=0$, it shows a is not modified by a negative adverb and $w'=a$. So the tuple T is changed, $T=(w', f, sent)$.

We built a sentiment classifier using the Naïve Bayes classifier. Naïve Bayes classifier is based on Bayes' theorem.

$$P(c\,|\,M) = \frac{P(c)\cdot P(M\,|\,c)}{P(M)} \tag{5}$$

Where c is a sentiment, M is a message. Because, we have equal sets of positive and negative reviews, we simplify the equation:

$$P(c\,|\,M) = \frac{P(M\,|\,c)}{P(M)} \tag{6}$$

$$P(c\,|\,M) \sim P(M\,|\,c) \tag{7}$$

We make an assumption of conditional independence in the Naïve Bayes classifier. Opinion words extracted in reviews are text features. The formula is as follows:

$$P(M\,|\,c) = \prod_{w'\in M} P(w'\,|\,c) \tag{8}$$

We calculate the probability of opinion word w_i' that belongs to sentiment c_j , that is $P(w_i'\,|\,c_j)$.

$$P(w_i'\,|\,c_j) = \frac{1 + fre(w_i', c_j)}{|W| + \sum_{k=1}^{|W|} fre(w_k', c_j)} \tag{9}$$

Where $|W|$ is the number of opinion words in text sets, $fre(w_i', c_j)$ is the frequency of opinion word w_i' in c_j . Smoothing treatment is used in the formula to avoid the item whose value is 0. If $w_i' = a\,\&\,n$, it is a compound word that is made up of an adjective and its negative adverb. In this case, we calculate the frequency that a and n co-occur. If $w_i' = a$, the opinion word is the adjective and we calculate the frequency of a . So, sentiment c_f that text M belongs to is a maximum we calculate based on the formula.

Table 5. Classification results of Chinese reviews in three fields.

Product Name	Polarity	Precision	Recall	F-score
Notebook	Positive	0.874	0.843	0.858
	Negative	0.857	0.806	0.831
Hotel	Positive	0.912	0.901	0.906
	Negative	0.887	0.823	0.854
Book	Positive	0.871	0.855	0.863
	Negative	0.890	0.847	0.868

$$c_f = \arg\max_{c_j \in C}\{\prod_{w' \in M} P(w' \mid c_j)\} \tag{10}$$

7. EXPERIMENTS

We carried out the experiments using customer reviews. Product review in the paper. Product reviews in the paper came from the corpus built by Tan *et al.* [15]. The corpus contains Chinese reviews of hotels, notebooks and books. There are 2000 positive and negative reviews after removing duplicates in each field.

In our experiment, we finished two tasks. The first is to find product features that have been commented on by customers. The experimental results are shown in Tables **1-4**. And the second is to decide whether the reviews are positive or negative. Table **5** gives the classification results of Chinese reviews in three fields. The performances are measured using the standard evaluation measures of precision (p), recall (r) and F-score (F).

$$p = \frac{a}{a+b} \tag{11}$$

$$r = \frac{a}{a+c} \tag{12}$$

$$F = \frac{2pr}{p+r} \tag{13}$$

With reference to a confusion matrix, a refers to the number of correctly classified positive (negative) reviews, b refers to the number of classified non-positive(negative) reviews, and c refers to the number of non-classified positive (negative) reviews.

CONCLUSION

This paper proposed an effective method for identifying sentiment orientations of opinions expressed by reviewers on product features. It is able to deal with two major problems in opinion mining. One is feature extraction. The other is sentiment orientation identification. In our case, a domain corpus has a set of reviews reviewing the same product. And we find a review written by a single customer which is composed of a sequence of sentences. It is usually the case that the customer has the same sentiment or polarity on the same feature, although the feature may appear more than once in the review. We use similar Double Propagation algorithm in feature extraction. In this process, we extract product features of reviews and remove redundant features. Both explicit and implicit features are considered. We get higher p , r , F . In sentiment orientation identification, we use the Naïve Bayes classifier to classify opinion words. Opinion words we extracted are negative adverbs and adjectives. Our method can better distinguish different meanings of the same word in different sentences. We also achieve a higher value of p , r , F in sentiment identification.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This research is supported by the Specialized Research Fund for the Doctoral Program of Higher Education (No.20133718110014), the Nature Science Foundation of Shandong Province (No.ZR2012FM003) and the Nature Science Foundation of Shandong Province (No.ZR2013FM-023). The author would like to thank all the students and teachers for their efforts. We are also appreciating the reviewers and editors for their valuable suggestions and comments to improve this work.

REFERENCES

[1] P.D. Turney, “Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews ,” In: *Proceedings of the 40th annual meeting on association for computational linguistics*, Association for Computational Linguistics, Philadelphia, pp. 417-424, 2002.

[2] M. Hu, and B. Liu, “Mining and summarizing customer reviews,” In: *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, pp. 168-177, 2004.

[3] B. Liu, *Web Data Mining*, Springer-Verlag Berlin Heidelberg, pp. 411-447, 2007.

[4] D. Lee, O.R. Jeong, and S. Lee, "Opinion mining of customer feedback data on the web," In: *Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication*. ACM, New York, pp. 230-235, 2008.

[5] E. Cambria, B. Schuller, Y. Xia, and Havasi C, "New avenues in opinion mining and sentiment analysis," *IEEE Intelligent Systems*, vol. 28, pp. 15-21, 2013.

[6] H.D. Kim, K. Ganesan, P. Sondhi, and C. Zhai, "*Comprehensive Review of Opinion Summarization*," Survey, pp. 1-30, 2011.

[7] A. Rashid, N. Anwer, M. Iqbal, and M. Sher, "A Survey Paper: Areas, Techniques and Challenges of Opinion Mining," *International Journal of Computer Science Issues (IJCSI)*, vol. 10, pp. 18-31, 2013.

[8] P. Turney, "Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL," *Lecture Notes in Computer Science*, vol. 2167, pp. 491-502, 2001.

[9] A. Harb, M. Plantié, G. Dray, M. Roche, F. Trousset, and P. Poncelet, "Web Opinion Mining: How to extract opinions from blogs," In: *Proceedings of the 5th International Conference on Soft Computing as Transdisciplinary Science and Technology*, ACM, Cergy-Pontoise, pp. 211-217, 2008.

[10] A. Pak, and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," In: *International Conference on Language Resources and Evaluations*, Valletta, pp. 1320-1326, 2010.

[11] H. Wang, X. Nie, L. Liu, and J. Lu, "A fuzzy domain sentiment ontology based opinion mining approach for Chinese online product reviews," *Journal of Computers*, vol. 8, pp. 2225-2231, 2013.

[12] Q. Su, X. Xu, H. Guo, Z. Guo, X. Wu, X. Zhang, and B. Swen, "Hidden sentiment association in Chinese web opinion mining," In: *Proceedings of the 17th International Conference on World Wide Web*, ACM, New York pp. 959-968, 2008.

[13] W. Che, Z. Li, and T. Liu, "Ltp: A Chinese language technology platform," In: *Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations*, Association for Computational Linguistics, Beijing, pp. 13-16, 2010.

[14] G. Qiu, B. Liu, J. Bu, and C. Chen, "Expanding domain sentiment lexicon through double propagation," In: *International Joint Conference on Artificial Intelligence*, Pasadena, pp. 1199-1204, 2009.

[15] S. Tan, X. Cheng, M.M. Ghanem, B. Wang, and H. Xu, "A novel refinement approach for text categorization," In: *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, ACM, Bremen, pp. 469-476, 2005.

Received: September 16, 2014 Revised: February 23, 2015 Accepted: March 31, 2015