Automatic Extraction for Product Feature Words from Comments on the Web*

Zhichao Li, Min Zhang, Shaoping Ma, Bo Zhou, and Yu Sun

State Key Laboratory of Intelligent Technology and Systems,
Tsinghua National Laboratory for Information Science and Technology,
Department of Computer Science and Technology, Tsinghua University,
Beijing, 100084, China P.R.
lizhichaoxyz@sohu.com, {z-m,msp}@tsinghua.edu.cn,
{zhoubo2000,sunorrain}@gmail.com

Abstract. Before deciding to buy a product, many people tend to consult others' opinions on it. Web provides a perfect platform which one can get information to find out the advantages and disadvantages of the product of his interest. How to automatically manage the numerous opinionated documents and then to give suggestions to the potential customers is becoming a research hotspot recently. Constructing a sentiment resource is one of the vital elements of opinion finding and polarity analysis tasks. For a specific domain, the sentiment resource can be regarded as a dictionary, which contains a list of product feature words and several opinion words with sentiment polarity for each feature word. This paper proposes an automatic algorithm to extraction feature words and opinion words for the sentiment resource. We mine the feature words and opinion words from the comments on the Web with both NLP technique and statistical method. Left context entropy is proposed to extract unknown feature words; Adjective rules and background corpus are taken into consideration in the algorithm. Experimental results show the effectiveness of the proposed automatic sentiment resource construction approach. The proposed method that combines NLP and statistical techniques is better than using only NLP-based technique. Although the experiment is built on mobile telephone comments in Chinese, the algorithm is domain independent.

Keywords: Resource constructing, product feature, opinion word.

1 Introduction

Before deciding to buy a product, many people tend to consult others' opinions on it. Web provides a perfect platform which one can get information. Many customers record their comments on products on the Websites, forums or blogs. Reading the comments, one concerned about a product can find out its main advantages and disadvantages. However, only few comments cannot give a convincing suggestion and

* Supported by the Chinese National Key Foundation Research & Development Plan (2004CB318108), Natural Science Foundation (60621062, 60503064, 60736044) and National 863 High Technology Project (2006AA01Z141).

G.G. Lee et al. (Eds.): AIRS 2009, LNCS 5839, pp. 112–123, 2009.
© Springer-Verlag Berlin Heidelberg 2009

persons do not have sufficient energy to browse more. Therefore, how to automatically manage the numerous comments and suggest the potential customers is becoming a research hotspot recently. The main target is mining the customers' opinions to the products. The opinions are classified into positive, negative and neutral. Naturally, customers prefer the products with more positive comments than those with more negative ones.

There are three levels for such opinion mining task: document level, product level and feature level. To the document level, a whole document only generates a single opinion, which is coarse and inaccurate. Because one document may contain not only one product, and have different opinion polarities to each, a general classification to the document is not appropriate. Then it comes to product level, which is generating an opinion polarity for a given product or a brand. Since nothing can be consummate, there are always some advantages and disadvantages for a product. So feature level opinion can be more exact to express the customers' attitude. It also helps potential customers understand the product more clearly and definitely. Feature stands for an *attribute* or a *component* of a product. For example: "待机时间(standby time)" is an attribute of a mobile telephone, while "显示屏(screen)" is a component, both of them can be named as "**feature words**". In the examples following, "The standby time of Nokia is long enough" expresses a positive opinion while "The screen is too small" expresses a negative one.

*The **standby time** of Nokia is long enough.* *Ex. 1*

*诺基亚的**待机时间**足够长*

*The **screen** is too small.* *Ex. 2*

显示屏太小了.

Therefore, to analyze a comment, we need to find the feature words the document contains and the opinion words that embellish the feature words. By confirming the polarity of the opinion words, the main viewpoint of the comment is discovered. With so many comments owning clear polarity of opinions for a given product, customers can easily get to know the product deeply.

We construct a sentiment resource for a given domain to offer efficient and effective utilities to analyze the comments. The sentiment resource can be considered as a dictionary that contains a list of feature words and several opinion words with polarity tag for each feature. This paper mainly introduces the algorithm to extraction the feature words and opinion words. The polarity of opinion words determination is taken as future work. Although the product description by the company shows a lot of feature words of the product, the words are not abundant. Web users sometimes take informal words to describe a feature. Therefore, mining the feature words is necessary.

There are three benefits to construct such a dictionary beforehand. First, after holding such a dictionary, a new comment will be easily dealt with. We can extract the feature words and most opinion words from the new comment through matching the items in the dictionary quickly. Computing the polarity of the opinion words for every new comments offline will save much time. Second, processing a large set of comments to find the feature words and opinion words can not only use the NLP (Natural Language Processing) techniques but also the statistical characteristics. It gets a better

performance than processing comments one by one online. Supposing that we do not have the dictionary, when we get a new comment to process, we hardly take any statistical characteristics to make the performance better. Third, the dictionary is easy to maintain. When a new feature word or a new opinion word discovered, it can be easily added into the resource.

We use both NLP technique and statistical method to extract the feature words and the opinion words. Through our life experiences, the feature words are usually nouns or noun phrases, while the opinion words are usually adjectives. Tagging part-of-speech for all the words in the comments is helpful to the task. We can predefine some patterns to extract the target words with certain natural language characteristics. But this will bring a lot of noises. However, by using the different statistical characteristics of the words in the comments and in the whole Web background corpus, most noises can be removed. To get a higher recall, we also use unknown word finding techniques to extract more feature words. After filtering the noises and adding the new feature words, the performance is improved significantly in term of f-measure.

After describing about the related works in section 2, the data set and the main algorithm are introduced in section 3 and section 4 respectively, including feature words extraction, unknown word finding, and the usage of the background corpus, et al. Section 5 shows the evaluation of the performance. The conclusion and the future work are discussed in the last section.

2 Related Work

There are several researches to opinion mining for the product. The work in [2, 3] are based on document level. Feature level opinion mining also gains the interesting of researchers [4-11]. How to get feature words more exactly is the main problem in this task.

M. Hu and B. Liu [4, 5, 6] used association rule mining to find all frequent itemsets which are sets of words or phrases that occur together. CBA which was based on the Apriori algorithm is used. The words and the phrases extracted are considered as feature words. After two pruning phases (compactness pruning and redundancy pruning) to increase the precision and an infrequent feature identification phase to increase the recall, they got a good performance finally.

A. Popescu and O. Etzioni [7] built a system named OPINE. It is built on top of KnowItAll which was a Web-based domain-independent information extraction system. The system first extracts noun phrase and then filtered with point-wise mutual information value between the phrase and meronymy discriminators associated with the product class. They increased both precision and recall compared with M. Hu [6].

J. Yi and W. Niblack [8] extracted definite base noun phrases at the beginning of sentences followed by a verb phrase as the feature words. A definite base noun phrase is a noun phrase by specifically patterns preceded by the definite article “the”. The method was called bBNP (Beginning definite Base Noun Phrases) heuristic. To filter the noises, they used a document set that did not focus on the product. The terms appeared more in the documents focused on the product than in the ones did not were kept, while others were removed. They only used precision to evaluate their method and got a very high precision at the top 20 feature words.

C. Scaffidi, K. Bierhoff and et al. [9] considered noun and two nouns that occur successively as the feature word candidates. Comparing with the random section of English text, the feature words often occurred far more frequently in the comment text. Using the distribution of the words in the random section of English text to compute the probability that it occurred n times in the comment text, the less the probability was, the more possible it was a feature word. The Red Opal system can return quite high precision when few feature words returned.

There are also some researchers on feature extraction in Chinese. B. Wang and H. Wang [10] created a bootstrapping method to extract feature words and opinion words in Chinese product comments. They used only a few manual tagged training data to generate a Naive Bayesian classifier. The features to train are only natural language characteristics, such as "is there an adverb in the right". The algorithm is iterative; the terms tagged by former round were added to training set to train the latter round classifier. The experiment indicated that few manual tagged data can also bring high performance on both feature words extraction and opinion words extracting.

Q. Su, X. Xu and et al. [11] also used noun and noun phrase (two or more adjacent nouns) as feature words candidates. Unlike English, there is no definite article "the" in Chinese comments to help to filter noises. But there are some Chinese own boundary indicators such as "的(of)". They clustered the feature words and the opinion words respectively to eliminate the problem that hard to mine the implicit feature words. The feature words embellished by the same opinion word may be clustered together to stand for a feature.

Our work is different at these points: first, we also employ verb and verb phrase to be feature word candidates, it can improve the recall; second, we use the statistical characteristic of candidates on background corpus to filter noises; third, we use unknown word finding techniques to extract more feature words.

3 Data Set

The experiment was built on mobile telephone domain in Chinese. We gathered comments from 2 web sites: http://www.bibifa.com/ and http://dp.cnmo.com/. There are totally 6405 comments (6.18 MB plain text data). The average length of the comments is 994 byte (nearly 500 Chinese characters). There are two types of the data in each comment. One is tagged by the customer with "advantage" or "disadvantage", while the sentences are short but have strong sentiment. The other is descriptive text.

Unlike English, Chinese does not make use of any white space characters between words. Therefore, we should segment the sentence into words by a tool ICTCLAS [12]. It can take the corpus and tag the words with part-of-speech. Following is an example. The tags "n" "d" "a" stand for noun, adverb and adjective respectively.

Original text: 外形太大，电池性能较差！ *Ex. 3*

Segmented text: 外形/n 太/d 大/a ，/wd 电池/n 性能/n 较/d 差/a ！/wt *Ex. 4*

Translation text: The profile is too big. The battery performance is poor.

4 Algorithm

4.1 Feature Words Extraction

We use both NLP techniques and statistic methods to extract the feature words. First, using NLP techniques to acquire the feature words candidates, and then filter the noises with statistic characteristics.

In most of the researches, only nouns and noun phrases are extracted as the feature word candidates. But taking the particularity of Chinese into account, sometimes a verb can also be regarded as a feature word. It is not so much far to find an example (Ex. 5).

*The **operating** is simple; the **reaction** is quick.* *Ex. 5*

***操作**/v 简单/a ，/wd **反应**/vi 快/a*

The words "操作(operating)" and "反应(reaction)" are both verbs, although they are nouns in English translation. However, postulating them as feature words is property. Therefore, for extracting feature words with more coverage, we use four patterns shown in Table 1.

Table 1. Feature words extraction patterns

Patterns	Examples
Noun	外形/n (figure)
Verb	反应/vi (reaction)
Noun + Noun	音乐/n 功能/n (music function)
Verb + Noun	拍摄/v 效果/n (screen effect)

We can establish such an assumption that the term has a higher probability to be a feature word when it occurs more times in the comments. It is fairly explicit, because the main features of the product must be the hotspots of the customer discussions. Obviously, not all the nouns, verbs and noun phrases are feature words. Extracting all such terms must bring lots of noises. We can simply remove the terms with low occurrence frequency to filter the noises, but it is not very effective. So we employ three other filter methods to remove the noises:

1) Import a rule that there should be an adjective on the right of the feature word.
2) Use the frequency of the terms in the background corpus.
3) Use the unknown word finding techniques.

4.1.1 Adjective Rule

As the purpose of extracting feature words and opinion words is to construct a dictionary, if no opinion words appear with the feature word, the feature word is not very valuable in our resource. And the most syntax that the customers use to express their opinion is "feature word + adjective". (Ex.1 Ex.3 Ex.4 and Ex. 5) This syntax is easy

to express an opinion and accords to oral language. While the comments on the Web are often not very official text, it occurs quite often. Usually, customers add some adverb between the feature word and the adjective. Therefore, before extracting the feature words, stopwords and adverbs should be removed first. The adjective rule can be defined as: if in all the comments there is not any adjective occurs on right of the candidate term, the term is not a feature word. As the rule, the feature word candidates without an adjective on the right are removed to aspire after a higher precision. The main reason to follow this rule is to remove the verbs that cannot be feature words.

4.1.2 Background Corpus

Because the comments we use to mining feature words are from the Web, it is ineluctable that many common words that occur everywhere on the Web are mistakenly extracted as feature words. For example: "网址(Website)", "人们(people)". If the feature word candidate occurs too often on the whole Web text, we can doubt it is a true feature word for the domain we are concerned about justly. We regard the whole Web text as a background corpus. To get the occurrence frequency of the terms, Sogou Lab Internet Vocabulary [13] is used. The Internet Vocabulary is from the statistic analysis of the Chinese Web corpus indexed by Sogou search engine (http://www.sogou.com/) in October 2006. It is related to over 100 million Web pages containing more than 150,000 words with high frequency. The vocabulary gives the POS tag and occurrence frequency for each word. The common words that may be noises are usually nouns or verbs, while the phrases are seldom noises. We use the background corpus to filter the single nouns or verbs.

We define a feature named **TFP** (*term frequency proportion*) as follow:

$$TFP(t) = n / \ln(N) \tag{1}$$

Where n is the occurrence frequency of the term t in the comments that we use to mine the feature words, N is the occurrence frequency in the background corpus given by the Internet Vocabulary.

With the higher TFP, the candidate has more probability to be a true feature word. For example, the word "生活(life)" occurs in the comments 168 times and 251,581,894 times in the background corpus, while "杂音(murmur)" occurs 151 times in the comments and 856,288 times in the background corpus. "杂音(murmur)" has a higher TFP value (11.05) than "生活(life)" (8.69). Although "杂音(murmur)" appears less than "生活(life)" in the comments, it has more probability to be a true feature word. Therefore we select a threshold to confine the terms that have low TFP value. If TFP is less than the threshold, the term will be removed from the candidate set.

4.1.3 Unknown Word Finding Technique

As we use nouns, verbs and noun phrase to be the feature word candidates, the potential feature words that are not tagged as these POS will be missed. Especially, the mistake of Chinese word segmentation will cause the missing more familiarly. The feature word that is quite relative to the product or quite technological may not be segmented correctly by the common segmentation tool. It can influence the

performance of feature word extraction. For example, the word "蓝牙(Bluetooth)" is segmented as "蓝(blue)/a 牙(tooth)/n", not "蓝牙(Bluetooth)/n". Although it is a feature word, it cannot be extracted from the comments. To solve this problem, we employ an unknown word finding technique.

Z. Luo and R. Song [14] use context-entropy to find the unknown words. According to their investigation, "significant terms in specific collection of texts can be used frequently and in different contexts. On the other hand sub-string of significant term almost locates in its corresponding upper string (that is, in fixed context) even through it occur frequently." We select the feature word "蓝牙(Bluetooth)" to explain this. Fig.1 shows the contexts of "蓝牙(Bluetooth)" and "牙(tooth)". We can see on the left of "蓝牙(Bluetooth)", the contexts are various, while on the left of "牙(tooth)" the contexts are almost only "蓝(blue)" which is dominating. It means that in the comments of mobile telephone the word "牙(tooth)" is just a substring of the word "蓝牙(Bluetooth)". "蓝牙(Bluetooth)" can stand for a significant term.

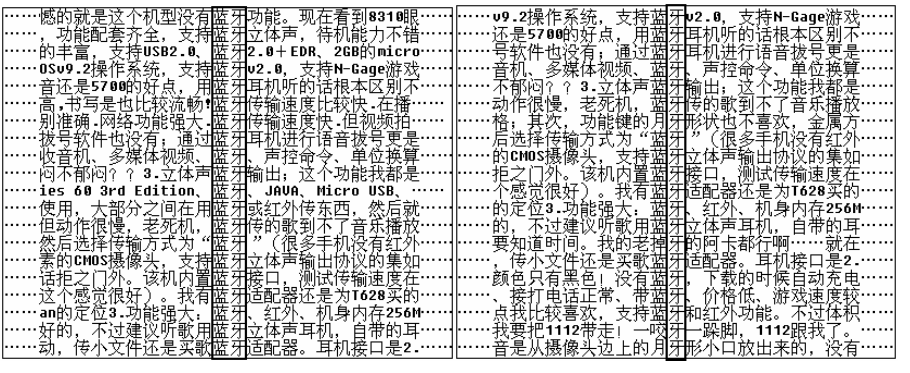

Fig. 1. The contexts of the word "蓝牙" and "牙"

To scale the chaos degree of the various contexts, left-context-entropy and right-context-entropy are defined. Assume ω as a term which appears n times in the corpus, $\alpha=\{a_1,a_2,\ldots,a_s\}$ and $\beta=\{b_1,b_2,\ldots,b_t\}$ as a set of left and right side contexts of ω in the corpus. Left-context-entropy and right-context-entropy of ω can be defined as:

$$LCE(\omega)=-\frac{1}{n}\sum_{i=1}^{s}C(a_i,\omega)\ln\frac{C(a_i,\omega)}{n} \tag{2}$$

$$RCE(\omega)=-\frac{1}{n}\sum_{i=1}^{t}C(\omega,b_i)\ln\frac{C(\omega,b_i)}{n} \tag{3}$$

Where $C(a_i, \omega)$ is count of co-occurrence of a_i and ω in the corpus and $C(\omega,b_i)$ is count of co-occurrence of ω and b_i.

We consider a feature word candidate joint with the word occurred on its left in the comments as a joint word. The difference of the left-context-entropy between original

feature word candidate and the joint word can used to judge if the candidate is only a substring of the joint word which can be a true feature word. If the LCE of the original candidate is low, and the LCE of the joint word is higher, we can regard that the joint word is a feature word more possibly than the original candidate. We can select an LCE threshold; if the LCE of the joint word is over threshold greater than the LCE of the original candidate, the original candidate is replaced by the joint word in the candidate set. The reason why we do not use the right-context-entropy is we have already employed the adjective rule on the right of the candidate. The candidate cannot joint the right term.

4.2 Opinion Words Extraction

In this task, we regarded only adjectives as opinion words. This step is simple. When extracting the feature words, the opinion words are also generated. The opinion words are extracted when we use adjective rule to filter the feature words noises. The candidate with an adjective on its right is considered as a feature word, while the adjective is an opinion word to this feature word. As describing in section 1, every opinion word must be along with a feature word to express a polarity. Therefore, an adjective without a feature word on the left is not a significant opinion word.

5 Evaluation

We manually pick up the feature words from the comments to organize the test data set to evaluate the performance of the feature words extraction algorithm. There are totally 1328 feature words in the test set TEST. The result feature word set that are created by the algorithm is signed as RESULT. Precision, recall and f-measure defined as follow are used.

$$precision = \frac{|TEST \cap RESULT|}{|RESULT|}, recall = \frac{|TEST \cap RESULT|}{|TEST|}, \\ f\_measure = \frac{2 * precision * recall}{precision + recall} \tag{4}$$

The performance of only using the patterns and adjective rule is shown in Fig. 2. Totally 2596 feature words have been extracted. Ranking the feature words by their occurrence frequency descending, the horizontal axis n stands for the set with top n results. From the illustration, we can find that at the top of the result list, there is a higher precision. It indicates that most of words with the high occurrence frequency are true feature words, which is in accordance with the assumption we take before in section 4.1. The recall curve in Fig. 2 is nearly linear, which indicates that the occurrence frequency of true feature words is nearly uniform distribution in the comments. Cutting the words whose occurrence frequency is below any threshold to maintain a high precision may depress the recall markedly. This can be proved by the f-measure curve; when all the results are kept, the f-measure can get the highest value.

Fig. 3 shows the performance that uses the background corpus to filter the noises. The result feature word set is ranking the feature words by their TFP value descending. The horizontal axis *n* also stands for the set with top *n* results. Comparing with the performance without filtering (ref. Fig. 2), the precision curve and the recall curve

are both different. The precision curve is gently at the top, which indicates that most of the words with high TFP value are true feature words. They are most the phrases. The recall curve is gently at the bottom of the result list, which indicates most of the words with low TFP value are not true feature words, and cutting them will not influence the recall badly. There is a maximum point in the f-measure curve; keeping the top n results in the maximum point in the set can get the highest f-measure 0.6817. The words with low TFP value are removed.

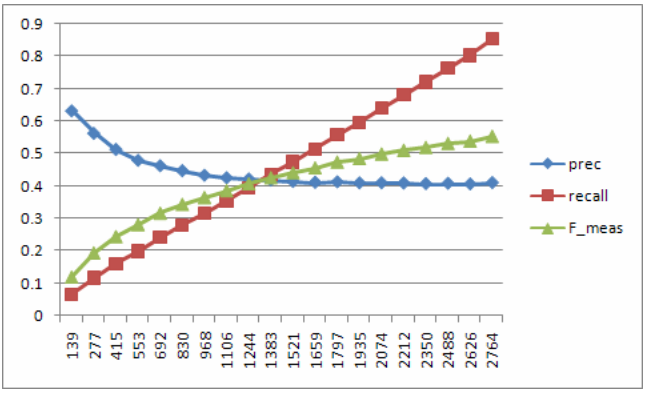

Fig. 2. The performance that only uses the patterns without any filtering

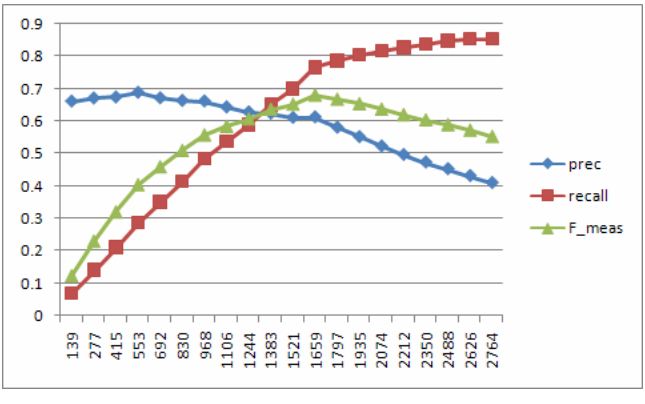

Fig. 3. The performance that uses the background corpus to filter the noises

To show the validity of using verbs and verb phrases as feature word candidates, we compare the performance between not using verbs and verb phrases, not using verbs and using all. The precision, recall and f-measure in Fig. 4 are the highest performance after filtering noises. The gray vertical bar stands for the performance of only using nouns and noun phrases (nouns + nouns) as candidates. It is the lowest among the three. When we added the verb phrases (verbs + nouns) pattern, the performance increases evidently. The white vertical bar stands for this performance. The black vertical bar for using all the

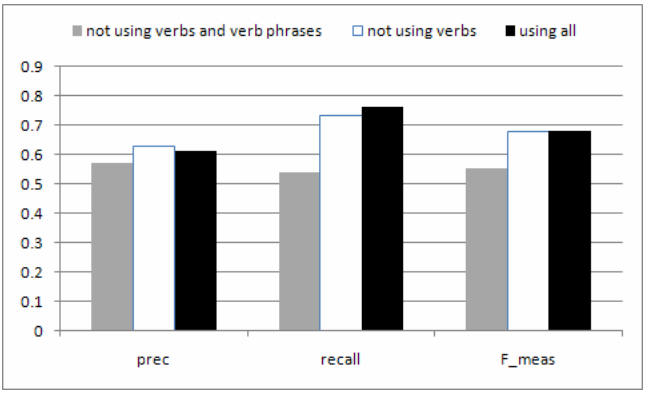

Fig. 4. Comparison of the performances between not using verbs and verb phrases, not using verbs and using all

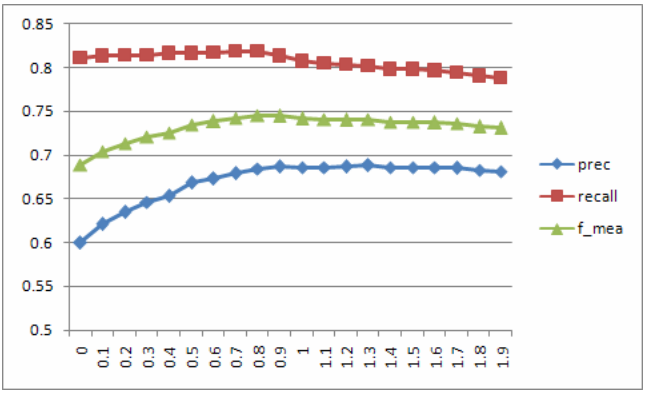

Fig. 5. The precision, recall and f-measure curves with the LCE threshold increasing

patterns is a little better than the white at f-measure and recall, which indicates that using verbs as feature words can extract more, but the noises are growing as well.

Next, we evaluate the validity of the unknown word finding technique. We control the threshold to observe the precision and the recall. If the LCE threshold is low, many terms will be replaced, and then the precision will become lower. Fig. 5 illustrates the curves with the LCE threshold increasing. The horizontal axis stands for the threshold. When the threshold is low, many joint words with a low LCE may replace the original candidate, which makes the precision low. However, when the threshold is high, very few joint words that are true feature words could replace the original candidates which are not. The recall is decreasing. When we select a proper LCE threshold 0.8, a higher f-measure 0.7407 is achieved.

Table 2 gives some examples for the unknown word finding technique using. The original candidates are not complete attribute or component of the product in the specific domain (Some of them do not have corresponding English translation in the domain context, which are marked as N/A in the table).

Table 2. Examples for the unknown word finding technique using

Original candidate	LCE of original candidate	Joint word	LCE of joint word
信功能 (N/A)	0.5069	短信功能(Short message function)	4.4261
性(N/A)	3.2736	兼容性(Compatibility)	4.1672
牙(tooth)	0.1392	蓝牙(Bluetooth)	6.5181

6 Conclusion and Future Work

We aim at automatically constructing the sentiment resource by mining customers' product comments. The sentiment resource can be considered as a dictionary for a given domain. The dictionary contains a list of feature words which stand for attributes or components of a product. For each feature word, there are several opinion words with polarity tag. The opinion words contain sentiment. The main contribution of this paper is extracting the feature words and the opinion words. Both NLP technique and statistical method are applied.

We use part-of-speech information and natural language patterns to extract the candidates of the feature words, and remove the noises through three filtering steps. The first step is based on the characteristic of natural language and the comments on the Web. The adjective restriction rule is shown to enhance the performance. The second and the third filtering steps are based on the statistical characteristic of the Web text. The common words that occur too often on the Web text are regarded with low probability to be a feature word. And the unknown word finding technique also brings a satisfying result. It concludes that using both NLP technique and statistical method is helpful in the task of extracting feature words. The use of statistical characteristic can be taken in the process of sentiment resource constructing. While dealing with a single comment, no statistical characteristic can be used. That is why we emphasize the importance and validity of resource construction.

Furthermore, although the experiment is built on the comments of mobile telephone domain in Chinese, the algorithm does not refer to any information of the domain. So it is a domain independent.

The future work can be expanded as the following:

1) For the aspect of feature word extraction, the more agile background corpus might be used to improve the timeliness of the vocabulary.

2) Polarity determination of opinion word will be studied in the future. For example, using the existing nature language tools such as HowNet [1] and considering the context of the opinion words, we can tag the polarity of the opinion words.

3) For the aspect of application of the sentiment resource, how to use the resource to analyze the comments is also an interesting task.

References

1. HowNet, http://www.keenage.com/
2. Ye, Q., Shi, W., Li, Y.: Sentiment Classification for Movie Reviews in Chinese by Improved Semantic Oriented Approach. In: Proceedings of the 39th Annual Hawaii international Conference on System Sciences, HICSS, January 04 - 07, vol. 03, p. 53.2. IEEE Computer Society, Washington (2006)
3. Li, J., Sun, M.: Experimental Study on Sentiment Classification of Chinese Review using Machine Learning Techniques. In: Proceedings of IEEE International Conference on Natural Language Processing and Knowledge Engineering 2007, pp. 393–400 (2007)
4. Hu, M., Liu, B.: Mining Opinion Features in Customer Reviews. In: Proceedings of Nineteenth National Conference on Artificial Intelligence, San Jose, California, USA, July 2-29, pp. 755–760. AAAI Press, Menlo Park (2004)
5. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the Tenth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining, KDD 2004, Seattle, WA, USA, August 22 - 25, pp. 168–177. ACM, New York (2004)
6. Liu, B., Hu, M., Cheng, J.: Opinion observer: analyzing and comparing opinions on the Web. In: Proceedings of the 14th international Conference on World Wide Web, WWW 2005, Chiba, Japan, May 10-14, pp. 342–351. ACM, New York (2005)
7. Popescu, A., Etzioni, O.: Extracting product features and opinions from reviews. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, Human Language Technology Conference. Association for Computational Linguistics, Vancouver, British Columbia, Canada, October 06-08, pp. 339–346. Morristown, NJ (2005)
8. Yi, J., Niblack, W.: Sentiment Mining in WebFountain. In: Proceedings of the 21st international Conference on Data Engineering (Icde 2005), ICDE, April 05-08, vol. 00, pp. 1073–1083. IEEE Computer Society, Washington (2005)
9. Scaffidi, C., Bierhoff, K., Chang, E., Felker, M., Ng, H., Jin, C.: Red Opal: product-feature scoring from reviews. In: Proceedings of the 8th ACM Conference on Electronic Commerce, EC 2007, San Diego, California, USA, June 11-15, pp. 182–191. ACM, New York (2007)
10. Wang, B., Wang, H.: Bootstrapping both Product Properties and Opinion Words from Chinese Reviews with Cross-Training. In: Proceedings of the IEEE/WIC/ACM international Conference on Web intelligence, Web Intelligence, November 02 - 05, pp. 259–262. IEEE Computer Society, Washington (2007)
11. Su, Q., Xu, X., Guo, H., Guo, Z., Wu, X., Zhang, X., Swen, B., Su, Z.: Hidden sentiment association in chinese Web opinion mining. In: Proceeding of the 17th international Conference on World Wide Web, WWW 2008, Beijing, China, April 21 - 25, pp. 959–968. ACM, New York (2008)
12. ICTCLAS, http://www.nlp.org.cn/
13. Sogou Lab Internet Vocabulary, http://www.sogou.com/labs/dl/w.html
14. Luo, Z., Song, R.: An Integrated Method for Chinese Unknown Word Extraction. In: Proceedings of 3rd ACL SIGHAN Workshop on Chinese Language Processing, Barcelona, Spain, pp. 148–155 (2004)