Sentiment Classification for Chinese Product Reviews* Using an Unsupervised Search Engine-based Method

Zhang Ziqiong [1,2] Li Yijun [1] Ye Qiang [1,2]
[1] Department of Management Information System, Harbin Institute of Technology, Harbin 150001
[2] Hong Kong Polytechnic University, Hong Kong
E-mail: {ziqiong, liyijun, yeqiang}@hit.edu.cn

Abstract: Sentiment classification aims at mining opinions of customers for a certain product by automatically classifying the reviews into positive or negative opinions. Though some pioneer researches explored the approaches for English review classification, few works have been done on sentiment classification for Chinese reviews. In this paper, we focus on a specific domain—cell phone review and propose an Intemet-based approach for Chinese product review mining. The experimental results show the effectiveness of the proposed approach in sentiment classification for Chinese product reviews.

Keywords: Chinese product review, search engine-based algorithm, PMI-IR, sentiment classification

1 Introduction

Sentiment classification, also known as polarity classification or opinion mining, attempts to address this problem by automatically determining whether a review is positive or negative. Sentiment classification has recently attracted much attention from the natural language processing community. There are relative studies at document level, at sentence level, and at expression level.

For the task of opinion mining from the Web, many researchers apply statistical approaches[1][4]. More specifically, they attempt to learn rules or statistical models from corpora in which sentences or documents are tagged with polarity labels (polarity-tagged corpus). Ordinarily, those polarity-tagged corpora are built in two ways: built by labeling polarity of reviews by hand, or built directly from labeled reviews in some review sites, such as AMAZON.COM. In the case of AMAZON. COM, the review's polarity is represented by using 5-star scale. However, the two approaches are not appropriate for building large polarity-tagged corpus. Manual construction of tagged corpus is time-consuming and expensive. It is difficult to build large corpora for various domains. The method that relies on review sites can not be applied to domains in which no large amount of labeled reviews is available.

To avoid these problems, Turney[2] first proposed using a search engine's corpus for review mining. Turney assumes that terms of similar orientation tend to co-occur. He chooses *excellent* and *poor* respectively as positive and negative seed words. Thus, terms that tend to co-occur with *excellent* in documents are more likely positive, and terms that tend to co-occur with *poor* in documents are more likely negative. The co-occurrence is measured by the number of hits returned by a search engine, with a query consisting of the term and a seed word. The sentiment orientation of the term is determined by the difference between its co-occurrence with *excellent* and *poor*. Then the average semantic orientation of all extracted terms in a review is calculated. If the average is positive, the review is predicted to recommend the item it discusses. Otherwise, the review is predicted not to recommend it. Turney used AltaVista NEAR operator[1] to get the number of hits, which constrained the search to documents that contain query words in a fixed neighborhood of another.

[3] showed for the task of sentiment classification to product reviews, Near operator is clearly superior to AND. It suggests that words that occur closer to each other are more likely to be semantically related. However, existing search engines only provide AND operator.

In this paper, we also present an Internet-based approach for review mining. It is similar to PMI-IR by issuing queries to a search engine, but it uses snippets to analyze the polarity of a term, which are small pieces of text snipped off from hit documents. A review is classified as positive if the average

*Supported by Research Funding of Hong Kong Polytechnic University (G-YX93)and National Natural Science Foundation of China (70771032).

[1] AltaVista no longer supports NEAR after it started using the Yahoo! 's database for its results from March 25, 2004.

semantic orientation of all extracted terms exceeds a given threshold; otherwise it is negative.

The remainder of this paper is organized as follows. Section 2 describes a typical Internet-based method for review mining. Section 3 gives an overview of our method. Section 4 presents the experiments and analyzes the results. Finally, the conclusion is presented in Section 5.

2 Internet-based methods for sentiment classification

The first Internet-based approach (PMI-IR) was proposed by Turney [2]. This method has been widely used by researchers. We summarize the process of typical PMI-IR Semantic Orientation(SO) approach [2][3][5] into 4 steps and then present related works on sentiment classification using PMI-IR or Internet-based method.

Step 1. Parse and tag the part-of-speech to the review documents using lexical analysis tools.

Step 2. Selectively extract two-word phrases conforming certain patterns based on part-of-speech.

Step 3. Determine the semantic orientation(SO) of a phrase, *phrase*. Take *excellent* and *poor* as a reference word pair (RWP) for example, SO(*phrase*) is calculated as:

$$SO(phrase) = PMI(phrase, excellent) - PMI(phrase, poor) \quad (1)$$

where, Pointwise Mutual Information (PMI) between two words, $word_1$ and $word_2$, is defined as follows :

$$PMI(word_1, word_2) = \log_2\left(\frac{p(word_1 \,\&\, word_2)}{p(word_1)\, p(word_2)}\right) \quad (2)$$

$p(word_1 \,\&\, word_2)$ is the probability that $word_1$ and $word_2$ co-occur. If the words are statistically independent, the probability that they co-occur is given by the product $p(word_1)\, p(word_2)$. The ratio between $p(word_1 \,\&\, word_2)$ and $p(word_1)\, p(word_2)$ is a measure of the degree of statistical dependence between the words. The log of the ratio corresponds to a form of correlation, which is positive when the words tend to co-occur and negative when the presence of one word makes it likely that the other word is absent. PMI-IR[2] estimates PMI by issuing queries to a search engine (hence the IR in PMI-IR).

Let hits(*query*) be the number of hits returned from a search engine, given the query *query*. SO(*phrase*) is calculated from equations (1)and (2) as follows:

$$SO(phrase) = \log_2\left(\frac{hits(phrase\ \text{NEAR}\ excellent)\, hits(poor)}{hits(phrase\ \text{NEAR}\ poor)\, hits(excellent)}\right) \quad (3)$$

The NEAR operator constrains the search to documents that contain *phrase* and *excellent* (or *poor*) within a given window size.

Step 4. Calculate a review's semantic orientation by averaging the SO values of all the extracted phrases in a review. The opinion is positive if its average semantic orientation exceeds zero, otherwise is negative.

3 Sentiment classification using snippets

This section describes our approach of calculating the semantic orientation of phrases and the overall polarity of a review is predicted by the average SO of all extracted phrases. This process is composed of the following steps: 1) Word segmentation and POS tagging for Chinese reviews. 2) Extract sentiment phrases from a review. 3) Crawl snippets from the Web. 4) Infer sentiment orientation of a phrase. 5) Sentiment classification for a Chinese review.

3.1 Sentiment phrase extraction

There are some studies about English pattern extraction to identify sentiment phrases form reviews [2][7]. We adopted the two-word patterns in [2],

First a part-of-speech tagger ICTCLAS 3.0[2] was applied to the Chinese reviews. Two consecutive

[2] http://mtgroup.ict.ac.cn/~zhp/ICTCLAS/

words were extracted form the review if their tags conform to any of the pattern in Tab. 1. Adjective or adverb in the patterns provides subjectivity, while the other word provides context.

Tab.1 Patterns of tags for extracting two-word phrases from reviews

ID	First word	Second word	Third word (Not Extracted)
1	Adjective	Noun	anything
2	Adverb	Adjective	Not Noun
3	Adjective	Adjective	Not Noun
4	Noun	Adjective	Not Noun
5	Adverb	Verb	anything

3.2 Semantic orientation calculation of phrases

In this section, we present an Internet-based approach that will use snippets to analyze the polarity of a sentiment phrase.

Snippets indicate the small pieces of text snipped off from documents, which ordinarily occur below links returned by search engines. Snippets can be sentences, clauses or segments extracted from hit pages. Snippets usually contain all or part of the query words. They allow a user to preview where the query words occur in a document.

For example, we issued a query of *"low cost" excellent*[3] to Google, then nearly 1000 snippets were returned. We listed four typical snippets below (in their returned order and snippets between them were omitted):

(1) Storage units are great shelter for the homeless. Public storage lockers can make ***excellent low cost*** homes for people who want to live in public storage...

(2) ***Low Cost***, ***Excellent*** Quality, and Fast Delivery Time! QTC Sales. Our site is under construction. Please come back later! Contact Info. QTC Sales Corporation ...

(3) With its ***low cost*** of consumable, fast print speeds, and high print quality, the Samsung ML-3051N is an ***excellent*** mono laser printer for a small office ...

(4) Pay less for car hire in Cairo with our ***low cost*** deals and all-inclusive prices to suit ... The city of Cairo offers an ***excellent*** range of accommodation, ...

In (1) (2) and (3), both *Low lost* and *excellent* occur in one sentence. They occur near in (1) and (2), and far in (3). In (4) we can infer *low cost* and *excellent* occur more far in the hit document, as there is a sign "..." which means that *low cost* and *excellent* separately come from different segments of the document and text between them is omitted. In the proposed approach, we will split this kind of snippet into segments.

To calculate Semantic Orientation (SO) of a sentiment phrase, *phrase*, we present a variant PMI-IR algorithm.

As NEAR operator is no longer available, we estimate $p(word_1 \ \& \ word_2)$ not with the number of hits but using returned snippets. For example, to calculate $p(phrase \ \& \ excellent)$, i.e. co-occurrences of *phrase* and *excellent*, we issue a query of *"phrase" excellent* to Google and crawl returned snippets. This process ordinarily collects nearly 1000 snippets for a query. Given a widow size, ten words for example, we note the number of snippets in which both *phrase* and *excellent* occur in a ten-word window. Let snippets(*phrase* NEAR *excellent*) denote the number, then equation (3) can be rewritten as:

$$SO(phrase) = \log_2\left(\frac{\text{snippets}(phrase \text{ NEAR } excellent)\text{hits}(poor)}{\text{snippets}(phrase \text{ NEAR } poor)\text{hits}(excellent)}\right) \qquad (4)$$

For each phrase, both definitions of SO(*phrase*) in equation (3) and (4) include an item hits(*poor*)/hits(*excellent*). In fact, it equals to adding the same weight to all phrases' SO values. We think it may be more proper to remove it from the definition of SO(*phrase*) for two reasons. First,

[3] *Low cost* is put in quotation marks to constrain the search to documents that contain *Low cost* in one clause.

hits(*poor*) and hits(*excellent*) are approximate numbers. In the time of our experiments, hits(*poor*) and hits(*excellent*) by Google are respectively *about* 57,300,000 and 44,400,000 pages, so the ratio between them is also an approximate number. Second, Turney set zero as threshold value for judging sentiment orientation of *phrase* in equation (1). In other words, the polarity of *phrase* is classified as positive if PMI (*phrase*, *excellent*) is bigger than PMI (*phrase*, *poor*); otherwise is negative. However, this assumption could be inappropriate. [6] shows that when using equation (1) to calculating SO values of phrases, zero as threshold value to judge polarity of *phrase* is somewhat biased. Thus we remove hits(*poor*)/hits(*excellent*) from equation (4) and meanwhile set nonzero threshold for review polarity classification. A simplified estimator of SO(*phrase*) is as follows:

$$\mathrm{SO}(\textit{phrase}) = \log_2\left(\frac{\mathrm{snippets}(\textit{phrase}\ \mathrm{NEAR}\ \textit{excellent})}{\mathrm{snippets}(\textit{phrase}\ \mathrm{NEAR}\ \textit{poor})}\right) \qquad (5)$$

In the preliminary experiments, we also found if snippets(*phrase* NEAR *reference word*) is small, *phrase* is likely a meaningless or wrongly spelt word collocation. That is because review data from web are often noisy and fragmentary which would result in errors in Chinese word segmentation and POS tagging. We just skip *phrase* when snippets(*phrase* NEAR *excellent*) or snippets(*phrase* NEAR *poor*) equals zero.

Let S_p and S_n respectively be two sets of snippets returned by a search engine, with a query "*phrase*" *excellent* and a query "*phrase*" *poor*[4]. Fig. 2 describes our algorithm of calculating semantic orientation of *phrase*.

Input: *phrase*, WINDOW
Output: SO(*phrase*)
Body:
- for each snippet $snip_p$ in S_p
  if ... exists in $snip_p$
    split $snip_p$ into $segments_p$ by ...; $S_p \leftarrow segments_p$
  for each snippet $snip_n$ in S_n
  if ... exists in $snip_n$
    split $snip_n$ into $segments_n$ by...; $S_n \leftarrow segments_n$
- for each snippet $snip_p$ in S_p
  if *phrase* and *excellent* exist and their distance<WINDOW
    snippets(*phrase* NEAR *excellent*) ++
  for each snippet $snip_n$ in S_n
  if both *phrase* and *poor* exist and their distance <WINDOW
    snippets(*phrase* NEAR *poor*) ++
- if snippets(*phrase* NEAR *excellent*) = 0 or snippets(*phrase* NEAR *poor*) = 0
  skip *phrase*
  else $\mathrm{SO}(\textit{phrase}) = \log_2\left(\frac{\mathrm{snippets}(\textit{phrase}\ \mathrm{NEAR}\ \textit{excellent})}{\mathrm{snippets}(\textit{phrase}\ \mathrm{NEAR}\ \textit{poor})}\right)$

Fig. 2 Semantic orientation calculation of phrases

For each review, we calculated the average SO value of all the extracted phrases to determine its overall semantic orientation. The opinion is judged as positive if its average semantic orientation exceeds the threshold value and is negative if otherwise.

[4] *phrase* is put in quotation marks to constrain the search to documents that contain *query* in one clause.

4 Experiments and results

4.1 Baseline dataset

In order to test the effectiveness of our algorithm, we used the polarity dataset of cell phone reviews created by Ye et al. [6]. It contains manually annotated 40 positive reviews and 40 negative reviews which were collected from the website "IT 168" (http://bj.it168.com/Newpinglun/cSpace_pl.asp?cType_code=0302). There are 6183 words in positive reviews and 4586 words in negative reviews. In the process of pattern extraction, totally 705 sentiment phrases are extracted from all the reviews.

4.2 Experiments

In the experiments, we chose Google over other search engines, because it has a huge database[5] and it always searches for pages containing all query words. We enclose the phrase in "double quotes" in a query, e.g. *"low cost" excellent*, so Google will detect phrase matches and usually ranks phrase matches higher.

We used the same reference word pairs (RWPs) as Ye et al. [6]. "经典" (Classical) and "精品" (High-quality Products) were chosen as positive words and "垃圾" (Garbage) and "失望" (Disappointing) as negative words. We carried out four separate experiments on different RWPs: "经典,垃圾", "经典,失望", "精品,垃圾", and "精品,失望".

We also tested on two window sizes, 10 characters and 20 characters. Fig. 3 and Fig. 4 show the sentiment classification accuracy for Chinese cell phone reviews at different threshold values, given the window sizes 10 and 20 characters.

Fig.3 shows that all RWPs can achieve 80% classification accuracy, given a window size of 10 characters. In Fig.4, the RWPs "经典,垃圾"and "精品,垃圾" can achieve accuracy above 80%, given a window size of 20 characters. The results are significantly better than the baseline accuracy 73.75%.

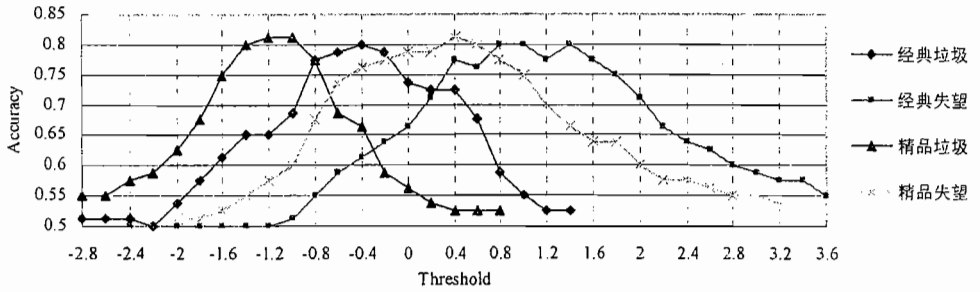

Fig.3 Accuracies at different threshold values given a window of 10 characters

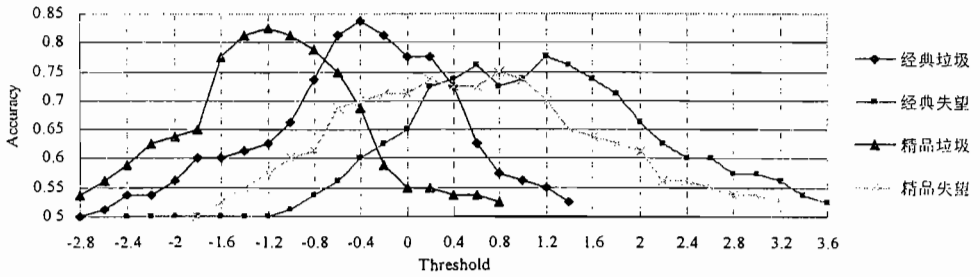

Fig.4 Accuracies at different threshold values given a window of 20 characters

[5] While no official claim is given, 20+ billion is once current estimate.

4.3 Discussion of results

From these experimental results three conclusions could be drawn.

First, four accuracy curves reach their peaks at different threshold values in Fig.3 and Fig.4. We checked the mining results manually and found a review obtained different sentiment orientation scores, using different RWPs. The reason is that given a window size, the co-occurrences between a phrase and different seed words are different, which result in different SO values of a phrase, and then different sentiment orientation scores of a review. Thus, for each RWP we should choose an optimal threshold value for the task of review sentiment classification.

Second, for each RWP, the two accuracy curves in Fig.3 and Fig.4 have the same general trend and they achieve higher accuracy at similar threshold value. Especially for the accuracy curves "经典,垃圾" and "精品,垃圾", the former reach its peak at -0.4, and the latter at -1.2 in two figures. That means that the optimal threshold value is not sensitive to window size. Thus in domain-dependant applications, we can first choose an optimal RWP for different products and then an optimal threshold value can be determined.

Finally, given a threshold value, the performance of each RWP can be affected by window size. The peaks of accuracy curves "经典,垃圾" and "精品,垃圾" in a window of 20 characters are higher than 10 characters, but reversely in the case of "经典,失望" and "精品,失望". Generally speaking, words that occur closer to each other are more likely to be semantically related, but there would be more co-occurrences of words in a larger window which tends to have higher statistical reliability. In Chinese language "垃圾" has a stronger negative sentiment than "失望", so a larger neighborhood would increase statistical reliability for "垃圾", but make noise for "失望".

Experimental results show that accuracy of review sentiment classification using Internet-based approach is influenced by combination of RWP, window size and threshold value. 1) With two window sizes, 10 and 20 characters, each RWP performs much alike at similar threshold value. 2) When a threshold value is fixed, for each RWP, window size would have an influence on classification performance, depending on the strength of RWP. An optimal window size will trade off between statistical reliability and noise making.

In addition, RWPs with stronger sentiment orientation perform slightly better as displayed in Fig.3 and Fig.4. Thus, in domain-related applications, RWP should be chosen first for a given product review. Correspondingly, optimal classification threshold value and window size can be determined.Tab.2 shows the results and parameters of out classifier when each RWP achieves relatively higher accuracy.

Tab.2 Average accuracy, precision and recall using Internet-based classifier

Parameters used	Acc(%)	Positive		Negative	
		Prec (%)	Recl(%)	Prec(%)	Recl(%)
经典垃圾 / 20 character window / threshold -0.4	83.75	88.57	77.50	80.00	90.00
经典失望 / 10 character window / threshold 1	80.00	77.27	85.00	83.33	75.00
精品垃圾 / 20 character window / threshold -1.2	82.50	84.21	80.00	80.95	85.00
精品失望 / 10 character window / threshold 0.4	81.25	82.05	80.00	80.49	82.50
Baseline	73.75	70.21	82.50	78.79	65.00

4.4 Results from supervised learning: using small sets of labeled data

Given infinite resources, we can always annotate enough data to train a classifier using a supervised algorithm that will outperform unsupervised or weakly-supervised methods, but acquisition of data can be costly and time-consuming. To make a comparison we trained Support Vector Machines(SVMs) using small amount of labeled data. In SVM experiment, data described in Section 4.1 was equally divided into training and testing sets, each set containing 20 positive reviews and 20 negative reviews. We applied ICTCLAS 3.0 for word segmentation and took Chinese word as feature with no stopword. We repeated the experiment for three times. Results are shown in Tab.3. As we expected, SVMs classifier performs not so steadily with a small size of training data. We were pleasantly surprised at that

the results are comparable to state-of-the-art supervised model SVMs.

Tab.3 Average accuracy, precision and recall for SVMs with small amount of data

Round	Acc(%)	Positive		Negative	
		Prec (%)	Recl(%)	Prec(%)	Recl(%)
1	85.00	88.89	80.00	81.82	90.00
2	80.00	87.50	70.00	75.00	90.00
3	87.50	89.47	85.00	85.71	90.00

5 Conclusion

This paper presents an unsupervised Internet-based approach of classifying a Chinese product review as positive or negative using snippets from search engines. The core of the algorithm is to estimate the semantic orientation of a phrase by analyzing its association with reference words in the text of snippets. In the experiments with the Chinese cell phone reviews, the approach achieves accuracy over 80%. The results are comparable to state-of-the-art supervised model SVMs.

Reference

[1] B. Pang, L. Lee, S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. Proc. of the Conf. on Empirical Methods in Natural Language Processing, EMNLP'02, 2002:79-86.

[2] P. Turney. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. Proc. of the Meeting of the Association for Computational Linguistics, ACL'02, 2002: 417-424.

[3] P. Turney. Measuring praise and criticism: Inference of semantic orientation from association. ACM Transactions on Information Systems, 2003, 21(4):315-346.

[4] K. Dave, S. Lawrence, D. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. Proc. of the 12th Intl. World Wide Web Conference, WWW'03, 2003:519-528.

[5] P. Chaovalit, L. Zhou. Movie review mining: A comparison between supervised and unsupervised classification approaches. The 38th Hawaii International Conference on System Sciences, HICSS'05, 2005:1-9.

[6] Q. Ye, Y. Li, Y. Zhang. Semantic-Oriented Sentiment Classification for Chinese Product Reviews: an Experimental Study on the Reviews for Books and Cell Phones. Tsinghua Science and Technology (Special Issue), 2005, 10(S1):797-802.

[7] Z. Fei, J. Liu, G. Wu. Sentiment classification using phrase patterns. Proc. of the Fourth International Conf. on Computer and Information Technology, CIT'04, 2004:1-6.

Acknowledgements The authors wish to express sincere gratitude to Zheng-Hua Li for his constructive comments and technical support, and to Wen-Ying Zheng for her efforts on data preparation.

汉语商品评论情感分析——一种基于搜索引擎的无监督方法*

张紫琼[1,2] 李一军[1] 叶强[1,2]

[1]信息管理与信息系统系. 哈尔滨工业大学, 哈尔滨 150001 [2]香港理工大学, 香港

E-mail: {ziqiong, liyijun, yeqiang}@hit.edu.cn

摘 要: 作为非结构化信息挖掘的一个新兴领域，网络评论情感分析引起了人们的极大兴趣。利用对互联网上客户评论信息的挖掘与分析结果，消费者可以了解其他用户的态度倾向分布，做出更好的购买决策。本文提出了一种利用搜索引擎计算词语情感倾向的无监督方法，进而对商品评论的总体情感倾向进行褒贬分析。在对手机评论的情感分类试验中，分类精度达到80%以上，初步表明了该方法的可行性。

关键词: 汉语商品评论，搜索引擎，PMI-IR，情感分类

作者简介:张紫琼（1982－），女,黑龙江大庆人,博士生,主要研究领域为互联网情感分析;李一军(1957－),男,博士,教授,博士生导师,主要研究领域为管理信息系统，决策支持系统与电子商务;叶强(1972-),男,博士,教授,博士生导师,主要研究领域为电子商务，互联网情感分析.

*本项目受到国家自然科学基金 (70771032),香港理工大学研究基金(G-YX93)资助