Extracting Features for Sentiment Classification: in the Perspective of Statistical Natural Language Processing

Pei Yin, Hongwei Wang, Wei Wang

School of Economics and Management, Tongji University, evawater@163.com
*Corresponding Author
School of Economics and Management, Tongji University, hwwang@tongji.edu.cn
School of Economics and Management, Tongji University, wayswang@gmail.com

Abstract

In order to improve the accuracy of sentiment classification especially for Chinese online reviews, this paper proposed an approach to extract features for sentiment classification effectively and efficiently. The proposed method is based on supervised machine learning and follows the basic procedure of statistical natural language processing (SNLP). Through feature selection, feature extraction and feature weighting, the significant features for sentiment classification are obtained while the insignificant ones are removed. Comparative experiments have been made on mobile phone online reviews in Chinese, and improvement of three extraction algorithms (DF, IG and CHI) have also been made to obtain more accurate result. This research enriches the studies on sentiment classification theoretically, and is helpful for future work.

Keywords: Sentiment Classification, Chinese Online Reviews, Statistical Natural Language Processing, Supervised Machine Learning

1. Introduction

With the rapid expanding of e-commerce, more and more people would like to share their personal feelings about certain products online. Enormous reviews are generated over the Internet, greatly influencing the decision making of potential customers, as well as the product design of manufacturers. Since it is impossible for customers and manufactures to manually read all online reviews, technology to automatically extract valuable information from these texts has become a hot topic of both academia and industry in recent years.

What readers most care about is authors’ sentiment orientation, thus a new research task called sentiment classification emerges recently. Unlike traditional text classification focused on subjects like “sports”, “movie” and so on, the purpose of sentiment classification is to determine the sentiment polarity of texts like “positive” or “negative”.

There are two approaches mainly utilized in previous sentiment classification studies: machine learning and semantic orientation. However, most studies are focused on reviews written in English, but Chinese has its unique way of emotional expression so that these research results of English reviews are unable to be directly applied to Chinese reviews. Therefore, this paper is motivated and focused on the improvement in sentiment classification for Chinese online reviews.

2. Related work

2.1. Machine learning approach

The machine learning approach mainly uses text classification algorithms to extract opinion and determine whether the opinion is positive or negative.

At first, text features with emotional information are selected to help distinguish sentiment polarity of the text. Words with different part of speech (e.g. adjective, adverb, verb and noun) and sentiment orientation are selected as features (e.g. Turney, 2002; Mullen et al., 2004; Xu et al., 2007; Kar and Mandal, 2011). Then, features are further selected by evaluating the value of
each feature and removing the feature with less value than a threshold. Information retrieval methods are mainly used in this process, such as Mutual Information (MI), Information Gain (IG), Chi-square Statistic (CHI) and Document Frequency (DF) (e.g. Zhou, et al., 2004; Tang, et al., 2007). After that, selected features are weighted and quantized to be converted into text vectors by Boolean Weight (e.g. Pang, et al (2002), Term Frequency (TF), Inverse Document Frequency (IDF), and Term Frequency-Inverse Document Frequency (TF-IDF). At last, sentiment classifier inputting text vectors is employed to predict the sentiment polarity of text. A large volume of literature and evidence show the dominant power that Support Vector Machine (SVM) has on text classification over other classifiers like Naïve Bayes (NB), Maximum Entropy (ME) and Neuron Networks, especially in the case of limited training samples (Pang et al., 2002; Ye, et al., 2005; Jiang, 2006; Ni, et al., 2007; Tang, 2007; Xia and Peng, 2009; Ye, et al., 2009; Zhang et al. (2009).

The machine learning approach is efficient and fully automatic, but the machine needs be trained to learn the pattern previously.

2.2. Semantic orientation approach

The semantic orientation approach is based on identifying and extracting sentiment words and phrases, as well as semantic rules and patterns contained in the evaluation text.

Two types of techniques have been used in previous semantic orientation approach: corpus-based and dictionary-based techniques (Liu, 2007). The corpus-based techniques apply methods in information theory, such as Sentiment Orientation-Pointwise Mutual Information (SO-PMI) (Turney, 2002) and syntactic paths (Zhao et al., 2010), to find co-occurrence patterns of words and determine their sentiments. Dictionary-based techniques utilize synonyms, antonyms and hierarchies in WordNet (or other lexicons with sentiment information) to determine word sentiments (Andreevskaia and Bergler, 2006; Ebrahim, Fathian and Gholamian, 2012). Building upon WordNet, SentiWordNet is a lexical resource for sentiment analysis which has more sentiment related features than WordNet (Esuli and Sebastiani, 2006). SentiWordNet has been used as the lexicon in recent sentiment classification studies (Devitt and Ahmad 2007; Denecke 2008; Fahrni and Klenner 2008; Kang et al., 2012).

The semantic orientation approach needs no prior training, but is semi-automatic and based on external resources like corpus and lexicon. Moreover, this approach conducts opinion mining at a finer-grained level than the machine learning approach does.

3. Extracting features for sentiment classification

This paper follows the trend of statistical natural language processing and uses supervised machine learning techniques. Based on reviewing the related works above, it is obvious that more researches are focused on designing better classifier algorithms than improving the method of feature extraction. Therefore, the primary goal of this research is to improve the accuracy and the efficiency of extracting text features, and ultimately enhance the performance of sentiment classification for Chinese online reviews.

The main step statistical natural language processing is to represent the unstructured review in a structured form so that the review could be inputted to a sentiment classifier. Vector Space Model (VSM) is commonly used for text representation in most machine learning approach. Each piece of review is represented by a certain number of features. Supposing the feature sequence is \( \{f_1, f_2 ... f_n\} \), then the review can be presented as: \( r = \{t_1, t_2 ... t_n\} \), where \( f_i \), \( n \) and \( t_i \) respectively denotes the \( i^{th} \) feature, the quantity of extracted features, and the weight of feature \( f_i \) (also the value of \( i^{th} \) vector). Therefore, the proposed approach in this paper uses VSM to format online reviews. There are three major steps of building the VSM: feature selection, feature extraction and feature weighting.
3.1. Feature selection

Each review is segmented into words or terms at first. Some commonly used words like “的 (of)” are useless for sentiment classification, thus drop these words to avoid redundancy of feature set. It is assumed that adjectives, adverbs, verbs and nouns most likely indicate author’s opinions. In addition, some modal particles like “啊” (ah) and punctuations like “!” and “?” may also somewhat reflect the author’s emotion. As a result, six types of features are selected and listed in Table 1.

Table 1. Six types of selected features

<table>
<thead>
<tr>
<th>POS</th>
<th>Example</th>
<th>POS</th>
<th>Example</th>
<th>POS</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noun</td>
<td>极品(masterwork)</td>
<td>Adjective</td>
<td>不错(good)</td>
<td>Punctuations</td>
<td>!, ?</td>
</tr>
<tr>
<td>Verb</td>
<td>使用(use)</td>
<td>Adverb</td>
<td>容易地(easily)</td>
<td>Modal Particles</td>
<td>啊(ah)</td>
</tr>
</tbody>
</table>

Furthermore, it should be noted that negative words like “不” (not) is considered as an emotional modifier, which inverses the polarity of the following word, thus this paper combines the modifier with its target together as a feature. For example, if there is a “不” (not) before “漂亮” (pretty), the feature would be “不漂亮” (not pretty).

3.2. Feature extraction

After feature selection, there are still a great number of features. It is obvious that too many features containing a large amount of noise will not only add dimensions of vector space and increase the cost of classification, but also decrease the accuracy of classifier. On the contrary, too few features are not good either due to their incompetence of sufficiently reflecting the characteristics of reviews. So a certain number of features are extracted by three algorithms, which are DF (Document Frequency), CHI (Chi-square Statistic) and IG (Information Gain).

1) DF calculates the number of documents consisting of a particular feature.

\[
DF(t_i) = \sum_{j=1}^{M} N(C_j, t_i)
\]

Where \(C_1\) denotes the positive class and \(C_2\) denotes the negative class, \(N(C_j, t_i)\) denotes the number of documents containing feature \(t_i\) and belonging to class \(C_j\).

2) CHI measures the associated degree of a particular feature with a certain class.

\[
CHI(t_i) = \frac{N(C_1, t_i) \times N(C_2, \bar{t}_i) - N(C_2, t_i) \times N(C_1, \bar{t}_i)}{N(C_2, t_i) \times N(C_1, \bar{t}_i) + N(C_1, t_i) \times N(C_2, \bar{t}_i)}
\]

Where \(N(C_j, \bar{t}_i)\) denotes the number of documents not containing feature \(t_i\) and belonging to class \(C_j\).

3) IG determines the uncertainty that a particular feature belongs to a certain class.

\[
IG(t_i) = \left[ - \sum_{i=1}^{M} P(C_i) \log P(C_i) - \sum_{j=1}^{N} P(C_j, t_i) \log P(C_j | t_i) \right] + P(\bar{t}_i) \left[ - \sum_{i=1}^{M} P(C_i | \bar{t}_i) \log P(C_i | \bar{t}_i) \right]
\]

Where \(P(t_i) = DF(t_i) / NC\) and \(NC\) denotes the total number of documents) denotes the possibility that a certain document contains feature \(t_i\). Similarly, \(P(\bar{t}_i)\) denotes the possibility that a certain document doesn’t contain feature \(t_i\). \(P(C_j | t_i) = N(C_j, t_i) / DF(t_i)\) denotes the conditional possibility that a certain document belongs to class \(C_j\) under the condition of containing feature \(t_i\), and \(P(C_j | \bar{t}_i) = N(C_j, \bar{t}_i) / DF(\bar{t}_i)\) denotes the conditional possibility that a certain document belongs to class \(C_j\) under the condition of not containing feature \(t_i\).
3.3. Feature weighting

The text feature should be weighted and converted into numeric vector, based on its impact on sentiment classification. Since the sentiment polarity (positive or negative) is basically determined by key emotional features, the influences that these features have on sentiment classification are determined by their appearance in certain class. Thus this paper utilizes Boolean method to set feature weights, as 1 or 0.

4. Experiments and evaluations

In the experiment, 2300 pieces of mobile phone reviews are collected from a popular e-commerce website (www.360buy.com) specialized in selling electronic products in China, and 1500 of which are training corpuses while the other 800 are testing corpuses. An ICTCLAS System developed by the Institute of Computing Technology of Chinese Academy (http://ictclas.org/) is utilized to do the word segmentation and POS tagging. Moreover, based on the research of Phienthrakul et al. (2009), the polynomial kernel function performs better than the monomial one in sentiment classification, thus a classifier LIBSVM with Radial Basis Function (RBF) that automatically works out favorite variables through 5-fold Cross-Validation training is introduced to this research.

4.1. Comparison of different extraction algorithms

Comparative experiments on the three extraction algorithms are conducted, and the results are shown in Table 2. Classification accuracy and CV-rate are utilized to evaluate the performance.

<table>
<thead>
<tr>
<th>Thresholds</th>
<th>DF</th>
<th>IG</th>
<th>CHI</th>
</tr>
</thead>
<tbody>
<tr>
<td>CV-rate</td>
<td>Accuracy</td>
<td>CV-rate</td>
<td>Accuracy</td>
</tr>
<tr>
<td>50</td>
<td>93.667</td>
<td>93.375</td>
<td>93.2</td>
</tr>
<tr>
<td>100</td>
<td>93.9333</td>
<td>94.4667</td>
<td>96.0667</td>
</tr>
<tr>
<td>200</td>
<td>95.1333</td>
<td>96.125</td>
<td>96.6333</td>
</tr>
<tr>
<td>300</td>
<td>95.6777</td>
<td>97.125</td>
<td>96.1333</td>
</tr>
<tr>
<td>400</td>
<td>95.6667</td>
<td>97.625</td>
<td>96.0667</td>
</tr>
<tr>
<td>500</td>
<td>95.7333</td>
<td>98</td>
<td>95.9333</td>
</tr>
<tr>
<td>600</td>
<td>95.8667</td>
<td>97.5</td>
<td>95.8667</td>
</tr>
</tbody>
</table>

From Table 2, we can see that when the number of features increases from 50 to 300, IG performs better than the other two algorithms. When the number of features increases from 400 to 600, DF performs better. The highest accuracy is achieved by DF with 500 features. From another point of view, however, IG reaches 97.5% only with rather fewer features compared to that when DF reaching 98% (200 vs 500). Thus it is concluded that IG performs better with relatively fewer features and is more efficient for extracting feature than DF.

4.2. Comparison of different number of features

As mentioned in Section 3.2, too many and too few features both decrease the performance of classification. Fig.1 shows the evidence to support this argument.
The result demonstrates that the accuracy of all three algorithms rises to the peak firstly and then declines. CHI shares the same growing rate with IG, but is outperformed by both IG and DF. And other conclusions drawn from Table 2 are also shown in Fig.1.

4.3. Comparison of different POS

As mentioned in Section 3.1, six types of POS are used to select features: noun, adjective, adverb, verb, modal particle and punctuation. Each type supposedly has different contribution to the performance of classification. IG is used to extract 50, 100, and 200 features and evaluate the effect of each type and its combinations. The results are shown in Table 3.

Table 3. Comparison of different POS and its combinations (%)

<table>
<thead>
<tr>
<th>Threshold</th>
<th>Adjective</th>
<th>Noun</th>
<th>Adverb</th>
<th>Verb</th>
<th>Comb.1</th>
<th>Comb.2</th>
<th>Comb.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>94.125</td>
<td>75.25</td>
<td>80.5</td>
<td>81.75</td>
<td>87</td>
<td>94.5</td>
<td>94.375</td>
</tr>
<tr>
<td>100</td>
<td>94.5</td>
<td>77.625</td>
<td>84</td>
<td>84.125</td>
<td>91</td>
<td>95.25</td>
<td>94.375</td>
</tr>
<tr>
<td>200</td>
<td>95.625</td>
<td>76.25</td>
<td>86.75</td>
<td>86.625</td>
<td>92.25</td>
<td>96.625</td>
<td>97</td>
</tr>
</tbody>
</table>

Caption

Comb.1: combination of noun, adverb and verb;
Comb.2: all features except from particles of speech and punctuation;
Comb.3: all features

Table.3 shows that comparing the single POS (noun, adjective, adverb and verb), adjective gets the best performance and even performs better than the combination of noun, adverb and verb, adverb and verb get similar performance, and noun performs the worst. This indicates that adjective contains more sentiment information and makes more contribution to the sentiment classification than other POS. Besides that, Comparing comb.2 and comb.3 indicates that modal particle and punctuation are not as important as it is assumed.

5. Modified feature extraction algorithms

5.1. Factors affecting feature extraction

In the three extraction algorithms, the key variables are the number of positive and negative corpuses either including the feature $t_i$ or not, which are denoted as $N(C_j, t_i)$ and $N(C_j, \overline{t}_i)$ ($j=1$ or 2). And these variables are greatly influenced by two factors. One is the number of positive corpuses and negative corpuses. If there are more negative corpuses than positive ones, the number of negative features would exceed that of positive features. In other words, $N(C_j, t_i)$ ($t_i$ is a negative feature) and $N(C_j, \overline{t}_i)$ ($t_i$ is a positive feature) would be magnified, thus negative features would be more likely to be extracted than positive features. The other factor is the length of corpus. If negative corpuses are longer than positive corpuses, there are more features in negative corpuses than in positive ones, thus it would also increase the possibility of extracting negative features.
5.2. Modified algorithms

\( N(C_j, t_i) \) is converted into \( N'(C_j, t_i) \) in order to eliminate the influence of above two factors.

\[
N'(C_j, t_i) = \frac{N(C_j, t_i)}{N_j} \quad (4)
\]

\[
N'(C_j, t_i) = \sum_{j=1}^{N_j} N(C_j, t_i) - N'(C_j, t_i) \quad (5)
\]

Where \( N_j \) and \( NC_j \) respectively denotes the number of features in corpuses belonging to class \( C_j \) and the number of corpuses belonging to class \( C_j \). Thus the modification of the three algorithms DF, CHI and IG are as follows:

(1) Modified DF

\[
DF'(t_i) = \sum_{j=1}^{N_j} N'(C_j, t_i) \quad (6)
\]

(2) Modified CHI

\[
CHI'(t_i) = \frac{N'(C_j, t_i) \times N'(C_j, \bar{t}_i) - N'(C_j, t_i) \times N'(C_j, \bar{t}_i)}{[N'(C_j, t_i) + N'(C_j, \bar{t}_i)] [N'(C_j, \bar{t}_i) + N'(C_j, t_i)]} \quad (7)
\]

(3) Modified IG

\[
E'(t_i) = \left[ -\sum_{j=1}^{N_j} P(C_j) \times P(C_j) - \sum_{j=1}^{N_j} P(C_j|t_i) \times P(C_j|t_i) + P(C_j|t_i) \times \sum_{j=1}^{N_j} P(C_j|t_i) \times P(C_j|t_i) \right] \quad (8)
\]

Where \( P'(t_i) = DF'(t_i)/N(j), P'(C_j|t_i) = N'(C_j, t_i)/DF'(t_i), P'(C_j|\bar{t}_i) = N'(C_j, \bar{t}_i)/DF'(\bar{t}_i) \).

5.3. Experimental results and analysis

In order to test and verify the effectiveness that the modified algorithms have in obtaining more accurate results, comparative experiments of original and modified algorithms are conducted. The results are shown in Table 4 and Fig.2.

<table>
<thead>
<tr>
<th>Original/Modified</th>
<th>CHI</th>
<th>IG</th>
<th>DF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100</td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td>Original</td>
<td>93.75</td>
<td>96</td>
<td>97</td>
</tr>
<tr>
<td>Modified</td>
<td>94.375</td>
<td>97.25</td>
<td>97.5</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td>Original</td>
<td>94.375</td>
<td>96</td>
<td>97.375</td>
</tr>
<tr>
<td>Modified</td>
<td>96</td>
<td>97.375</td>
<td>97.5</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td>Original</td>
<td>95</td>
<td>96.125</td>
<td>97.125</td>
</tr>
<tr>
<td>Modified</td>
<td>94.5</td>
<td>96.125</td>
<td>97.25</td>
</tr>
</tbody>
</table>

Figure 2. Comparison of Original and Modified Algorithms
Table 4 and Fig.2 illustrate the improvements in accuracy achieved by the three modified algorithms. The result indicates that the number and the length of positive or negative corpuses indeed have an influence on feature extraction and sentiment classification, and the modified algorithms perform better than the original ones with different number of features, especially in the case of CHI and IG.

However, the modified algorithms do not always outperform the original ones, and for two cases in Table 4 and Fig.2, the modified algorithms gain worse performances. The modified CHI obtains less accurate result than the original CHI when the number of features is 300, due to the modification of CHI changes the growth curve of this algorithm. In other words, the original CHI achieves its highest accuracy with 300 features, while the modified CHI achieves its highest accuracy with fewer features (200). As discussed before, the smaller the number of features is, the more efficient the algorithm will be. So the worse performance in this case is not a setback of the modification but an evidence of the improvement in efficiency. The modified DF is outperformed by the original one when the number of features is 100. This is because the DF values of features are greatly reduced by dividing the number of features and corpuses (positive or negative), making it difficult to extract valid features with higher threshold. Moreover, the results of DF with 200 and 300 features also show the insignificant improvement made by the modification of DF.

6. Conclusion

The method proposed in this paper is a supervised machine learning approach, in the perspective of statistical natural language processing (SNLP). Several comparative experiments are conducted based on mobile phone online reviews. The experimental results indicate the following conclusions:

(1) Comparing feature extraction algorithms, the performances vary with different number of features. IG performs the best with fewer features, while DF achieves better result with more features.

(2) Too many and too few features will both decrease the performance of classification, thus an appropriate threshold is essential for sentiment classification.

(3) Adjective contains more sentiment information and makes more contribution to the sentiment classification than other POS, and best performance will be achieved when combining adjectives, adverbs, and verbs together.

(4) The feature extraction algorithms are modified and better accuracies of sentiment classification are achieved by considering the number and the length of positive or negative corpuses.

Future research will be conducted in the following aspects: 1) reviews with a neutral sentiment (neither positive nor negative) and its contribution to sentiment classification will be discussed; 2) the impact that the number of training corpuses has on sentiment classification will be analyzed; 3) the influence of selecting kernel function and parameters for SVM classifier on sentiment classification will be researched; 4) the performances of sentiment classification on corpuses with different evaluation domains like movie online reviews and hotel online reviews will be studied.

Acknowledgments

This work is partially supported by the NSFC Grant 70971099 and the Fundamental Research Funds for the Central Universities.

References

