International Journal of Hybrid Information Technology
Vol.8, No.3 (2015), pp.347-354
http://dx.doi.org/10.14257/ijhit.2015.8.3.31

Exploiting Machine Learning for Comparative Sentences Extraction

Wei Wang, TieJun Zhao, GuoDong Xin and YongDong Xu

School of Computer Science and Technology,
Harbin Institute of Technology,
Harbin, China
{wangwei}@hitwh.edu.cn, {tjzhao, gdxin }@hit.edu.cn,
{ydxu}@insun.hit.edu.cn

Abstract

This paper studies the problem of extracting Chinese comparative sentences from user reviews, which is a problem of text classification in the level of sentence. This paper first deals with the class skewed problem of review data, and then builds a SVM (support vector machine) model to classify comparative and non-comparative sentences into different groups on a balanced dataset. Various linguistic and statistical features are introduced to characterize a sentence. Experiments were conducted on user-generated product reviews. As a result, our experiments show significant performance, an overall F-score of 85.87%.

Keywords: *comparative sentence, machine learning, consumer review*

1. Introduction

The prevalence of web 2.0 provides people more opportunities to participate in the comments about commodities through Internet forums, shopping sites, and blog *etc*. These reviews, derived from customers, are able to reflect their real sentiment on how they feel products. Increasing researchers have been aware of the tremendous value of user-generated opinions for practical applications. They have used automated text analysis technology to extract such information. For example, information extraction method was used to acquire product attributes which are usually evaluated aspect. Sentiment analysis was applied to determine opinion direction of author towards different products. In summary, user opinion mining has drawn the attention of many researchers in the field of machine learning and data mining [1-5].

There are mainly two types of opinion in user reviews: comparison and direct opinion. Most of researches are concerned with direct opinions about a product [2, 3]. These systems try to provide parallel comparisons of different products by summarizing strengths and weaknesses of each product [6, 7]. Such systems may lead to an incorrect order between products because opinions about a product can be provided by someone who has not used any other products or be provided by people from different evaluation perspective. By contrast, comparative opinions express relationships of similarity or different between two or more products [8-11]. And comparative opinions are often given by someone who has used several products. Thus comparative opinions are more confident evaluation way than direct opinions.

Much of comparative opinions about products derived from consumer reviews. Users who experienced the products commonly prefer to compare several competitive products, expressing which one they think better and why. For instance,

ISSN: 1738-9968 IJHIT

Ex.1. “安全性和舒适性方面，车 x 好于车 y。(car x is better than car y in safety and comfort.)”

Ex.2. “所有同级别汽车中，x 性能最好。(The performance of X is the best in the same level of cars.)”

Extraction of comparative information from consumer reviews is a non-trivial task due to informal expression and unbounded size of blogs and forums. In this paper, our goal is to automatically extract Chinese comparative sentences from consumer reviews, which is a problem of text classification in the level of sentence. A SVM (support vector machine) model is built to classify each sentence into either comparative or non-comparative. The method first preprocesses the data by eliminating those regular opinion sentences, then a set of linguistic and statistical features that characterized a sentence are introduced. Finally those features are used to train a SVM classifier. Experimental results over 10-fold cross validation show the overall precision of 92.21% and the overall recall of 80.35%.

The rest of this paper is organized as follows: Section 2 discusses the related work. Section 3 states the method of balancing corpus. In section 4, we describe how to classify one sentence into comparative or non-comparative. Section 5 presents experiment results. Section 6 concludes our study and discusses future directions.

2. Related Work

Our research is related to linguistics and computational linguistics. In linguistics, Shang [12] explored Chinese comparative category and sub-category. Chen [13] discussed various language constructs of comparisons. Che [14] describes comparative gradable and comparative marker words. In summary, linguists have studied the grammar and semantic of comparative constructs, but almost no one studies on how to distinguish comparatives from non-comparative sentences.

In computational linguistics, nowadays there are two popular methods for identifying comparative information, machine learning and pattern match. Jindal and Liu [8] have investigated comparisons between products. In their research, comparative sentence candidates first were filtered based on some keywords. Then, a Naïve Bayesian model is trained using sequential rules mined from training samples. Park [15] identified comparison claims from full-text scientific articles based on a dependency tree representation with different classifiers. Huang [16] mined Chinese comparisons based on sequence pattern features. Song [17] use numerous patterns constructed manually to mine Chinese comparative sentences. Our study investigates various linguistic and statistic features, and verifies how those features are used in SVM classifier.

3. Balancing of Corpus

In real reviews, the ratio of comparative and non-comparative sentences is less than 1:5, which may produce a suboptimal result by existing classifiers. This paper uses keyword strategy to balance the corpus.

3.1. Sentence Types

A comparative sentence commonly contains some words that indicate comparisons, such as “比(than)”, “相似(similar)”,“不同(different)” and *etc*. These words can express comparative relationship between entities, which play an important role in discriminating comparative sentences. However, sentences that contain these words are not necessarily comparative sentences. Similarly, some sentences which don’t contain any indicator may be comparative sentences. We

divide all sentences into four types according to whether a sentence comprises indicators as well as the type it belongs:

Type 1: comparative sentences comprise indicators, *e.g.*, the sentence "她比他勤奋 (She is more diligent than him.)", is a typical comparative sentence. Such comparisons commonly contain some indicators, such as 'than', 'more' and etc.

Type 2: comparative sentences don't comprise indicators, *e.g.*, the sentence "X 相机有自拍功能，而 Y 相机没有。(Camera X has Self-timer function, but camera Y does not.)", implicitly compares two entities based on their shared attribute" Self-timer function". Although such sentences do not contain indicators, they are comparative sentences.

Type 3: non-comparative sentences comprise indicators, *e.g.*, the sentence "这款车外观比较炫 (The car looks more beautiful.)", is a non-comparative sentence though it contains "more" (It means 'the extent of beautiful').

Type 4: non-comparative sentences don't comprise indicators, *e.g.*, the sentence " 苹果手机质量不错"。(The quality of Apple phone is good.)", is a typical regular opinion sentence. The number of such sentences is the most in the corpus.

3.2. Data Balance Strategy

After sentences are classified into four types, we count the number of sentences for each type as follows:

Table 1. The Ratio of Four Types of Sentences

Type	Sentence Number	ratio
Type 1	1580	16.46%
Type 2	44	0.46%
Type 3	2211	23.03%
Type 4	5765	60.05%

Our final goal is to extract comparative sentences with type 1 and type 2. If the sentences of type 4 can be first ruled out, a relatively balance corpus will be obtained. We use keyword search technique to find out sentences of type1 and type 3. For extraction of sentences of type 2, some word-and-POS sequences formed as '<Subject Predicate, but Subject Predicate>' are constructed, which reflect the syntactic structure of type 2. For instance,

Ex.3. 手机 X 有蓝牙，而手机 Y 没有。(Phone X has bluetooth, but phone Y does not.)

For the example 3, "<NN, 有, 而, NN, 没有>(<NN has but NN does not>)" as a word-and-POS sequence is extracted, in which NN denotes the POS (Part of Speech) tag of noun. POS can adapt to different expression of comparative objects X and Y.

In order to compile a keyword lexicon, some comparative words were collected as seed-words from labeled corpus and linguistic literature, and then synonyms of seed-words were found by Tongyici cilin [1]. In addition, some word-and-POS sequences are added to the keyword lexicon. After artificial pruning, we produce a keyword lexicon that contains 102 words and 30 sequences.

We use $S_{original}$ to store the original set of sentences, S_k to store the set of keywords, and $S_{balance}$ to store the balanced set of sentences. The algorithm of keyword balance is given in Algorithm 1.

[1] (http://ir.hit.edu.cn)

Algorithm 1: keyword balance algorithm

Input: $S_{original}$, S_k

Output: $S_{balance}$

Method:

1. $S_{balance} = \phi$
2. for each $s_i \in S_{original}$ do
3. for each $k_j \in S_k$ do
4. if s_i contains k_j then
5. $S_{balance} \leftarrow s_i$
6. endfor
7. endfor
8. return $S_{balance}$

4. Extracting Comparative Sentences in Balanced Corpus

In order to filter out non-comparative sentences including keywords from the candidates, we employ machine learning technique (Support Vector Machine). Formally, let $S = \{s_1, s_2, ..., s_M\}$ be a set of sentences in a collection D. We convert each sentence into a vector $x = (x_{i1}, x_{i2}, ..., x_{in})$ in R^n. For this task, n equals to the number of features. Specifically, the feature $x_{ij} \in \{0,1\}$ for a sentence s_i corresponds to whether the j-th feature occurs in the sentence. Let $c_i \in \{0,1\}$ is a class variable such that $c_i = 1$ represent comparative and $c_i = 0$ represent non-comparative. The classifier will predict c_i for sentence s_i based on its feature vector x.

4.1. Mining Sequence Pattern Features

From the above examples, we can see that comparative sentences have special language patterns different from non-comparatives, which can be used as the features of machine learning. We automatically extract frequent sequences by mining comparative sequential patterns. Some infrequent patterns are manually built.

A. Class Sequence Rule

Sequential pattern mining (SPM), which extracts all sequential patterns from a sequence database, is an important data mining task [18, 19]. A sequential pattern, also called frequent sequence, is a subsequence whose support exceeds a predefined minimal support threshold.

Class sequence rule (CSR): A **class sequence rule** (CSR) is an implication $X \rightarrow y$, where X is a sequence, $y \in \{comparative, non-comparative\}$. A data instance (s_i, y_i) is called to support a CSR if s_i contain X. A data instance (s_i, y_i) is called to satisfy a CSR if s_i contain X and $y_i = y$. The *support* of the rule is defined as the fraction of total instances that satisfies the rule.

$$Support(CSR) = \frac{\text{instance number of satisfy rule}}{\text{Total number of instances}} \quad (1)$$

The *confidence* of the rule is defined as the proportion of instances in the sequence database that supports the rule also satisfies the rule.

$$Confidence(CSR) = \frac{\text{instance number of satisfy rule}}{\text{instance number of support rule}} \quad (2)$$

B. Mining Indicative Pattern

In order to mine CSR, we firstly transform corpus into a set of sequences. Each sentence in training set is broken up into several clauses by punctuation. We find all clauses having keywords and perform Chinese word segmentation and POS tagging for them. For each clause that comprises at least one keyword, we use actual word of each keyword as an item, for other words, we use the POS of each word as an item to produce a sequence. In order to adapt to various expression of comparative, we use POS tags of some words to form sequences. Each sequence is attached a class tag according to whether the sentence is a comparative or non-comparative sentence.

Ex.4. “奇瑞/NN QQ/NN 比/P 它/PN 还/AD 窄/JJ ,/PU 胳膊/NN 也/AD 没有/AD 碰到/VV 门/NN 呀/SP !/PU (Chery QQ is narrower than it, the arm did not touch the door Yeah!)” .

Example 4 has keyword “比(than)” in the first clause. So the sequence produced for first clause is:

<{NN}{NN}{比}{PN}{AD}{JJ}> Comparative

This paper uses improved PrefixSpan algorithm to extract sequence patterns, which can be found in literature [1]. The algorithm need to meet the minimum confidence threshold (0.70 can work best in our experiment). In our context, the minimum support is set multiple values because some comparative keywords appear very frequently, while some others appear rarely. In this strategy, keywords with similar word frequency are set the same minimum support.

4.2. Manual Rule Features

Some patterns compiled manually are also added to pattern database, such as superlative sentences which are too flexible, so it is hard to find their patterns by existing algorithms. For instance, 是/vshi 中/f 最 (is ... the most) is a superlative patterns. We build 45 patterns for superlative sentences.

5. Experimental Results

5.1. Data Sets

The experimental data in this paper from the fourth Chinese Opinion Analysis Evaluation (COAE 2012) published Task 2 corpus, which consists of the corpus in the field of automobiles and electronic products, a total of 9600 sentences. The number of comparative and non-comparative sentences in each dataset is given in Table 2. We use LIBSVM package with the RBF kernel to perform classification [20], and apply ICTCLAS 2013 (Institute of Computing Technology, Chinese Lexical Analysis System) to execute Chinese word segmentation and POS tagging.

Table 2. Number of Sentences in Each Dataset

Data set	Comparative Sentences	Non-Comparative Sentences
electronic products	811	3989
automobile	813	3987
Total	1624	7976

We use 10-fold cross validations to measure the performance of classification since the size of corpus is limited, and report the experiment results on each dataset and average precision, recall as well as F-score for evaluation.

5.2. Experimental Results

Precision, recall, and F-score are used to verify the effectiveness of the approach, their formulas are as follows:

$$\text{Precision} = \frac{number of\ extracted\ comparative\ sentences}{number\ of\ extracted\ sentences} \tag{3}$$

$$\text{R}ecall = \frac{number\ of\ extracted\ comparative\ sentences}{number\ of\ comparatives\ sentences} \tag{4}$$

$$F - score = 2 \times \frac{precision \times recall}{precision + recall} \tag{5}$$

Table 3. The identification Results on Automobile Balanced Corpus

Features	Precision	Recall	F-score
Keywords	0.8758	0.6964	0.7728
sequence patterns	0.9089	0.7104	0.7965
sequence patterns+ manual rules	0.9047	0.7270	0.8046
Keywords+ sequence patterns+ manual rules	0.9012	0.7624	0.8260

Table 3 and Table 4 present the experiment results on automobile and electronic product balanced corpus respectively. The results show all the precision values are quite good, while the recall values are low. For each dataset, the F-score of sequence pattern features is higher than that of keyword features, which indicates sequence pattern has a greater impact on the identification result of system. After manual rules are added, the recall of system is improved with a little loss of precision. When using both lexical and syntactic features, the system obtains the optimal recognition results, F-score of 82.6% and 89.10%. For each feature, the recognition results of sy stem in the car corpus are lower than those in electronic product corpus.

Table 4. The Identification Results on Electronic Balanced Corpus

Features	Precision	Recall	F1-score
Keywords	0.9064	0.8027	0.8502
sequence patterns	0.9246	0.7926	0.8531
sequence patterns+ manual rules	0.9399	0.8119	0.8706
Keywords+ sequence patterns+ manual rules	0.9429	0.8445	0.8910

Figure 1. Gives the Average Results that Include the Precision, Recall, and F-score Value of Different Methods

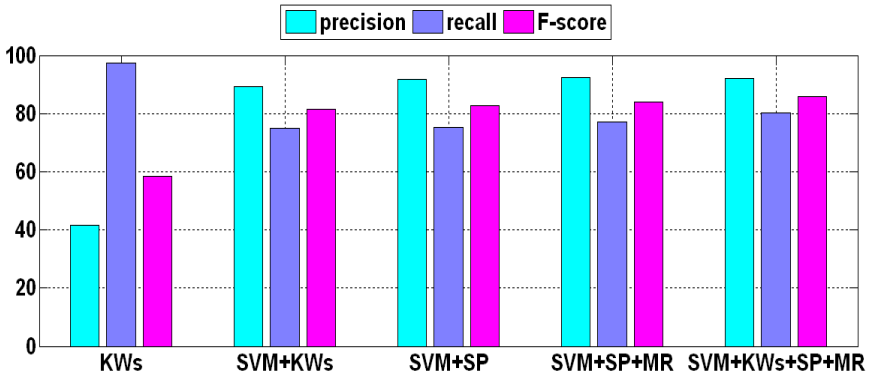

Figure 1. The Average Results on Balanced Corpus (%)

The Discussion of Result

(1) Keywords: We apply keyword technology to balance our corpus (Type1, 2, and 3). After corpus is balanced, we got a dataset including comparative sentences of 41.68%, non-comparatives of 58.32%, which indicates a relatively balance dataset is obtained. Recall of 97.29% for comparative sentences indicates that most comparative sentences are included in balanced corpus, *i.e.*, these keywords can cover almost all comparative sentences.

(2) SVM using keywords as features: After SVM model is applied, the F-score significant improved. We used the LIBSVM toolkit, kernel = RBF, gamma =0.0078 and C = 32 obtained the F-score of 81.42%.

(3) SVM using sequence patterns: Using alone sequence patterns as features to classify each sentence, we achieve the precision of 91.68%, the recall of 75.15%, and the F-score of 82.60%. This result shows that sequence patterns are effective features for distinguishing comparatives from non-comparatives.

(4) SVM using both sequence patterns and manual rules: All patterns that contain sequence patterns and manual rules are used to extract comparative sentences. The recall and F-score values are significantly improved. The F-score of 83.90% is achieved. This shows that manual rules are useful for our task.

(5) SVM using keywords, sequence patterns and manual rules: Using keywords and all patterns as the features, the recall is increased to 80.35%, and the F-score reaches 85.87%, which is the best result among these methods.

6. Conclusion

This paper studies how to automatically extract Chinese comparative sentences from consumer reviews, which is a problem of text classification in the level of sentence. Our work is partitioned into two subtasks: (1) Balancing data is to solve the class imbalance problem. (2) Extracting comparative sentences from balanced datasets by building the SVM classifier. Experimental results show the overall precision of 92.21% and the overall recall of 80.35%.

Acknowledgement

The work of this paper is funded by the project of National Natural Science Foundation of China (No.61173073, No.61172099, No.61073130, No. 61100093, No. 61272384), the Special Project of International Science and Technology Cooperation of China (No. 2014DFA11350).

References

[1] B. Liu, "Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Second Edition)", Springer, **(2011)**.
[2] S. M. Kim and E. Hovy, "Automatic identification of pro and con reasons in online reviews", In Proceedings of the COLING/ACL on Main Conference Poster Sessions, Association for Computational Linguistics, **(2006)**, pp. 483-490.
[3] B. Pang and L. Lee, "Opinion mining and sentiment analysis", Foundations and Trends in Information Retrieval, vol. 2, no. 1-2, **(2008)**, pp. 1-135.
[4] K. Xu, S. Liao, J. Li and Y. Song, "Mining Comparative Opinions from Customer Reviews for Competitive Intelligence", Decision Support Systems, vol. 50, no 4, **(2011)**, pp. 743-754.
[5] B. Liu, "Sentiment Analysis and Opinion Mining", Morgan & Claypool Publishers, **(2012)** May.
[6] B. Liu, M. Hu and J. Cheng, "Opinion observer: analyzing and comparing opinions on the web", In Proceedings of World-Wide Web Conference (WWW' 05), **(2005)**.
[7] J. Yi and W. Niblack, "Sentiment mining in Web Fountain", In Proceedings of the 21st International Conference on Data Engineering, **(2005)**, pp. 1073-1083.
[8] N. Jindal and B. Liu, "Identifying Comparative Sentences in Text Documents", In Proceedings of SIGIR'06, **(2006)**, pp. 244-251.
[9] N. Jindal and B. Liu, "Mining Comparative Sentences and Relations". In Proceedings of AAAI'06, **(2006)**, pp. 1331-1336.
[10] S. Yang and Y. Ko, "Finding Relevant Features for Korean Comparative Sentence Extraction", Pattern Recognition Letters, vol. 32, no 2, **(2011)**, pp. 293-296.
[11] S. Li, C. Y. Lin, Y. I. Song and Z. Li, "Comparable Entity Mining from Comparative Questions", In Proceedings of ACL'10, **(2010)**, pp. 650-658.
[12] P. Shang, "A Review on the System of Comparative Sentence". Applied Linguistics, **(2006)**, pp.77-80.
[13] J. Chen and X. Zhou, "The Selection and Arrangement of Grammatical Items concerning Comparative Sentences", Language Teaching and Research, no. 2, **(2005)**, pp. 22-33.
[14] J. Che, "A Brief Analysis of Comparative Sentences in Modern Chinese", Journal of Hubei Normal University (Philosophy and Social Science), vol. 25, no. 3, **(2005)**, pp. 60-63.
[15] D. Park and C. Blake, "Identifying Comparative Claim Sentences in Full-Text Scientific Articles", In Proceedings of ACL'12, **(2012)**, pp. 1-9.
[16] X. Huang, X. Wan, J. Yang and J. Xiao, "Learning to Identify Comparative Sentences in Chinese Text", In Proceedings of PRICAI'08, **(2008)**, pp. 187-198.
[17] R. Song, H. F. Lin and F. Chang, "Chinese Comparative Sentences Identification and Comparative Relations Extraction", Journal of Chinese Information Processing, vol. 23, no. 2, **(2009)**, pp.102-107.
[18] J. Pei, H. Pinto, Q. Chen, J. Han, B. Mortazavi-Asl, U. Dayal and M. Hsu. "PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth", In Proceedings of IEEE International Conference on Data Engingeering (ICDE-2001), **(2001)**.
[19] E. Riloff and J. Wiebe, "Learning Extraction Patterns for Subjective Expressions". In Proceedings of EMNLP'03, **(2003)**.
[20] C. C. Chang and C. J. Lin, "Libsvm: a library for support vector machines", ACM Transactions on Intelligent Systems and Technology, vol. 2, no. 3, **(2011)**, pp. 1-39.

