Kinds of Features for Chinese Opinionated Information Retrieval

Taras Zagibalov
Department of Informatics
University of Sussex
United Kingdom
T.Zagibalov@sussex.ac.uk

Abstract

This paper presents the results of experiments in which we tested different kinds of features for retrieval of Chinese opinionated texts. We assume that the task of retrieval of opinionated texts (OIR) can be regarded as a subtask of general IR, but with some distinct features. The experiments showed that the best results were obtained from the combination of character-based processing, dictionary look up (maximum matching) and a negation check.

1 Introduction

The extraction of opinionated information has recently become an important research topic. Business and governmental institutions often need to have information about how their products or actions are perceived by people. Individuals may be interested in other people's opinions on various topics ranging from political events to consumer products.

At the same time globalization has made the whole world smaller, and a notion of the world as a 'global village' does not surprise people nowadays. In this context we assume information in Chinese to be of particular interest as the Chinese world (the mainland China, Taiwan, Hong Kong, Singapore and numerous Chinese communities all over the world) is getting more and more influential over the world economy and politics.

We therefore believe that a system capable of providing access to opinionated information in other languages (especially in Chinese) might be of great use for individuals as well as for institutions involved in international trade or international relations.

The sentiment classification experiments presented in this paper were done in the context of Opinionated Information Retrieval which is planned to be a module in a Cross-Language Opinion Extraction system (CLOE). The main goal of this system is to provide access to opinionated information on any topic ad-hoc in a language different to the language of a query.

To implement the idea the CLOE system which is the context for the experiments described in the paper will consist of four main modules:

1. Query translation
2. Opinionated Information Retrieval
3. Opinionated Information Extraction
4. Results presentation

The OIR module will process complex queries consisting of a word sequence indicating a topic and sentiment information. An example of such a query is: ”Asus laptop + OPINIONS”, another, more detailed query, might be ”Asus laptop + POSITIVE OPINIONS”.

Another possible approach to the architecture of the CLOE system would be to implement the processing as a pipeline consisting, first, of using IR to retrieve certain articles relevant to the topic followed by second stage of classifying them according to sentiment polarity. But such an approach probably would be too inefficient, as the search will produce a lot of irrelevant results (containing no opinionated information).

Proceedings of the ACL 2007 Student Research Workshop, pages 37–42,
Prague, June 2007.

2 Chinese NLP and Feature Selection Problem

One of the central problems in Chinese NLP is what the basic unit[1] of processing should be. The problem is caused by a distinctive feature of the Chinese language - absence of explicit word boundaries, while it is widely assumed that a word is of extreme importance for any NLP task. This problem is also crucial for the present study as the basic unit definition affects the kinds of features to be used.

In this study we use a mixed approached, based both on words (tokens consisting of more than one character) and characters as basic units. It is also important to note, that we use notion of words in the sense of Vocabulary Word as it was stated by Li (2000). This means that we use only tokens that are listed in a dictionary, and do not look for all words (including grammar words).

3 Related Work

Processing of subjective texts and opinions has received a lot of interest recently. Most of the authors traditionally use a classification-based approach for sentiment extraction and sentiment polarity detection (for example, Pang et al. (2002), Turney (2002), Kim and Hovy (2004) and others), however, the research described in this paper uses the information retrieval (IR) paradigm which has also been used by some researchers.

Several sentiment information retrieval models were proposed in the framework of probabilistic language models by Eguchi and Lavrenko (2006). The setting for the study was a situation when a user's query specifies not only terms expressing a certain topic and also specifies a sentiment polarity of interest in some manner, which makes this research very similar to the present one. However, we use sentiment scores (not probabilistic language models) for sentiment retrieval (see Section 4.1). Dave et al. (Dave et al., 2003) described a tool for sifting through and synthesizing product reviews, automating the sort of work done by aggregation sites or clipping services. The authors of this paper used probability scores of arbitrary-length substrings that provide optimal classification. Unlike this approach we use a combination of sentiment weights of characters and words (see Section 4).

Recently several works on sentiment extraction from Chinese texts were published. In a paper by Ku et al. (2006a) a dictionary-based approach was used in the context of sentiment extraction and summarization. The same authors describe a corpus of opinionated texts in another paper (2006b). This paper also defines the annotations for opinionated materials. Although we use the same dictionary in our research, we do not use only word-based approach to sentiment detection, but we also use scores for characters obtained by processing the dictionary as a training corpus (see Section 4).

4 Experiments

In this paper we present the results of sentiment classification experiments in which we tested different kinds of features for retrieval of Chinese opinionated information.

As stated earlier (see Section 1), we assume that the task of retrieval of opinionated texts (OIR) can be regarded as a subtask of general IR with a query consisting of two parts: (1) words indicating topic and (2) a semantic class indicating sentiment (OPINIONS). The latter part of the query cannot be specified in terms that can be instantly used in the process of retrieval.

The sentiment part of the query can be further detailed into subcategories such as POSITIVE OPINIONS, NEGATIVE OPINIONS, NEUTRAL OPINIONS each of which can be split according to sentiment intensity (HIGHLY POSITIVE OPINIONS, SLIGHTLY NEGATIVE OPINIONS etc.). But whatever level of categorisation we use, the query is still too abstract and cannot be used in practice. It therefore needs to be put into words and most probably expanded. The texts should also be indexed with appropriate sentiment tags which in the context of sentiment processing implies classification of the texts according to presence / absence of a sentiment and, if the texts are opinionated, according to their sentiment polarity.

To test the proposed approach we designed two experiments.

The purpose of the first experiment was to find the most effective kind of features for sentiment polar-

[1] In the context of this study terms "feature" and "basic unit" are used interchangeably.

ity discrimination (detection) which can be used for OIR [2]. Nie et al. (2000) found that for Chinese IR the most effective kinds of features were a combination of dictionary look up (longest-match algorithm) together with unigrams (single characters). The approach was tested in the first experiment.

The second experiment was designed to test the found set of features for text classification (indexing) for an OIR query of the first level (finds opinionated information) and for an OIR query of the second level (finds opinionated information with sentiment direction detection), thus the classifier should 1) detect opinionated texts and 2) classify the found items either as positive or as negative.

As training corpus for the second experiment we use the NTU sentiment dictionary (NTUSD) (by Ku et al. (2006a))[3] as well as a list of sentiment scores of Chinese characters obtained from processing of the same dictionary. Dictionary look up used the longest-match algorithm. The dictionary has 2809 items in the “positive” part and 8273 items in the “negative”. The same dictionary was also used as a corpus for calculating the sentiment scores of Chinese characters. The use of the dictionary as a training corpus for obtaining the sentiment scores of characters is justified by two reasons: 1) it is domain-independent and 2) it contains only relevant (sentiment-related) information. The above mentioned parts of the dictionary used as the corpus comprised 24308 characters in the “negative” part and 7898 characters in the “positive” part.

4.1 Experiment 1

A corpus of E-Bay[4] customers’ reviews of products and services was used as a test corpus. The total number of reviews is 128, of which 37 are negative (average length 64 characters) and 91 are positive (average length 18 characters), all of the reviews were tagged as ‘positive’ or ‘negative’ by the reviewers[5].

We computed two scores for each item (a review): one for positive sentiment, another for negative sentiment. The decision about an item’s sentiment polarity was made every time by finding the biggest score of the two.

For every phrase (a chunk of characters between punctuation marks) a score was calculated as:

$$Sc_{phrase} = \sum(Sc_{dictionary}) + \sum(Sc_{character})$$

where $Sc_{dictionary}$ is a dictionary based score calculated using following formula:

$$Sc_{dictionary} = \frac{L_d}{L_s} * 100$$

where L_d - length of a dictionary item, L_s - length of a phrase. The constant value 100 is used to weight the score, obtained by a series of preliminary tests as a value that most significantly improved the accuracy.

The sentiment scores for characters were obtained by the formula:

$$Sc_i = F_i / F_{(i+j)}$$

where Sc_i is the sentiment score for a character for a given class *i*, F_i - the character’s relative frequency in a class *i*, $F_{(i+j)}$ - the character’s relative frequency in both classes *i* and *j* taken as one unit. The relative frequency of character *c* is calculated as

$$F_c = \frac{\sum N_c}{\sum N_{(1...n)}}$$

where $\sum N_c$ is a number of the character’s occurrences in the corpus, and $\sum N_{(1...n)}$ is the number of all characters in the same corpus.

Preliminary tests showed that inverting all the characters for which $Sc_i \leq 1$ improves accuracy. The inverting is calculated as follows:

$$Sc_{inverted} = Sc_i - 1$$

We compute scores rather than probabilities since we are combining information from two distinct sources (characters and words).

[2] For simplicity we used only binary polarity in both experiments: positive or negative. Thus terms “sentiment polarity” and “sentiment direction” are used interchangeably in this paper.

[3] Ku et al. (2006a) automatically generated the dictionary by enlarging an initial manually created seed vocabulary by consulting two thesauri, including tong2yi4ci2ci2lin2 and the Academia Sinica Bilingual Ontological Wordnet 3.

[4] http://www.ebay.com.cn/

[5] The corpus is available at http://www.informatics.sussex.ac.uk/users/tz21/corpSmall.zip.

In addition to the features specified (characters and dictionary items) we also used a simple negation check. The system checked two most widely used negations in Chinese: *bu* and *mei*. Every phrase was compared with the following pattern: *negation+ 0-2 characters+ phrase*. The scores of all the unigrams in the phrase that matched the pattern were multiplied by -1.

Finally, the score was calculated for an item as the sum of the phrases' scores modified by the negation check:

$$Sc_{item} = \sum (Sc_{phrase} * NegCheck)$$

For sentiment polarity detection the item scores for each of the two polarities were compared to each other: the polarity with bigger score was assigned to the item.

$$SentimentPolarity = argmax(Sc_i|Sc_j)$$

where Sc_i is an item score for one polarity and Sc_j is an item score for the other.

The main evaluation measure was accuracy of sentiment identification, expressed in percent.

4.1.1 Results of Experiment 1

To find out which kinds of features perform best for sentiment polarity detection the system was run several times with different settings.

Running without character scores (with dictionary longest-match only) gave the following results: almost 64% of positive and near 65% for negative reviews were detected correctly, which is 64% accuracy for the whole corpus (note that a baseline classifier tagging all items as positive achieves an accuracy of 71.1%). Characters with sentiment scores alone performed much better on negative reviews (84% accuracy) rather than on positive (65%), but overall performance was still better: 70%. Both methods combined gave a significant increase on positive reviews (73%) and no improvement on negative (84%), giving 77% overall. The last run was with the dictionary look up, the characters and the negation check. The results were: 77% for positive and 89% for negative, 80% corpus-wide (see Table 1).

Method	*Positive*	*Negative*	*All*
Dictionary	63.7	64.8	64.0
Characters	64.8	83.7	70.3
Characters+Dictionary	73.6	83.7	76.5
Char's+Dictionary+negation	76.9	89.1	80.4

Table 1: Results of Experiment 1 (accuracy in percent).

Judging from the results it is possible to suggest that both the word-based dictionary look up method and character-based method contributed to the final result. It also corresponds to the results obtained by Nie et al. (2000) for Chinese information retrieval, where the same combination of features (characters and words) also performed best.

The negation check increased the performance by 3% overall, up to 80%. Although the performance gain is not very high, the computational cost of this feature is very low.

As we used a non-balanced corpus (71% of the reviews are positive), it is quite difficult to compare the results with the results obtained by other authors. But the proposed classifier outperformed some standart classifiers on the same data set: a Naive Bayes (multinomial) classifier gained only 49.6 % of accuracy (63 items tagged correctly) while a Support vector machine classifier got 64.5 % of accuracy (82 items).[6]

4.2 Experiment 2

The second experiment included two parts: determining whether texts are opinionated which is a precondition for the processing of the OPINION part of the query; and tagging found texts with relevant sentiment for processing a more detailed form of this query POSITIVE/NEGATIVE OPINION.

For this experiment we used the features that showed the best performance as described in section 4.1: the dictionary items and the characters with the sentiment scores.

The test corpus for this experiment consisted of 282 items, where every item is a paragraph. We used paragraphs as basic items in this experiment because of two reasons: 1. opinionated texts (reviews) are usually quite short (in our corpus all of them are one paragraph), while texts of other genres are usually much longer; and 2. for IR tasks it is more usual to retrieve units longer then a sentence.

[6] We used WEKA 3.4.10 (http://www.cs.waikato.ac.nz/ ml/weka)

The test corpus has following structure: 128 items are opinionated, of which 91 are positive and 37 are negative (all the items are the reviews used in the first experiment, see 4.1). 154 items are not opinionated, of which 97 are paragraphs taken from a scientific book on Chinese linguistics and 57 items are from articles taken form a Chinese on-line encyclopedia Baidu Baike[7].

For the first task we used the following technique: every item was assigned a score (a sum of the characters' scores and dictionary scores described in 4.1). The score was divided by the number of characters in the item to obtain the average score:

$$averSc_{item} = \frac{Sc_{item}}{L_{item}}$$

where Sc_{item} is the item score, and L_{item} is the length of an item (number of characters in it).

A positive and a negative average score is computed for each item.

4.2.1 Results of Experiment 2

To determine whether an item is opinionated (for OPINION query), the maximum of the two scores was compared to a threshold value. The best performance was achieved with the threshold value of 1.6 - more than 85% of accuracy[8] (see Table 2).

Next task (NEGATIVE/POSITIVE OPINIONS) was processed by comparing the negative and positive scores for each found item (see Table 2).

Query	*Recall*	*Precision*	*F-measure*
OPINION	71.8	85.1	77.9
POS/NEG OPINION	64.0	75.9	69.4

Table 2: Results of Experiment 2 (in percent).

Although the unopinionated texts are very different from the opinionated ones in terms of genre and topic, the standard classifiers (Naive Bayes (multinomial) and SVM) failed to identify any non-opinionated texts. The most probable explanation for this is that there were no items tagged 'unopinionated' in the training corpus (the sentiment dictionary) and there were only words and phrases with predominant sentiment meaning rather then topic-related.

It is worth noting that we observed the same relation between subjectivity detection and polarity classification accuracy as described by Pang and Lee (2004) and Eriksson (2006). The accuracy of the sentiment detection of opinionated texts (excluding erroneously detected unopinionated texts) in Experiment 2 has increased by 13% for positive reviews and by 6% for negative reviews (see Table 3).

Query	*Positive*	*Negative*
Experiment 1	76.9	89.1
Experiment 2	89.9	95.6

Table 3: Accuracy of sentiment polarity detection of opinionated texts (in percent).

5 Conclusion and Future Work

These preliminary experiments showed that using single characters and dictionary items modified by the negation check can produce reasonable results: about 78% F-measure for sentiment detection (see 4.1.1) and almost 70% F-measure for sentiment polarity identification (see 4.2.1) in the context of domain-independent opinionated information retrieval. However, since the test corpus is very small the results obtained need further validation on bigger corpora.

The use of the dictionary as a training corpus helped to avoid domain-dependency, however, using a dictionary as a training corpus makes it impossible to obtain grammar information by means of analysis of punctuation marks and grammar word frequencies.

More intensive use of context information could improve the accuracy. The dictionary-based processing may benefit from the use of word relations information: some words have sentiment information only when used with others. For example, a noun *dongxi* ('a thing') does not seem to have any sentiment information on its own, although it is tagged as 'negative' in the dictionary.

Some manual filtering of the dictionary may improve the output. It might also be promising to test the influence on performance of the different classes of words in the dictionary, for example, to use only adjectives or adjectives and nouns together (excluding adverbials).

Another technique to be tested is computing the

[7] http://baike.baidu.com/

[8] A random choice could have approximately 55% of accuracy if tagged all items as negative.

positive and negative scores for the characters used only in one class, but absent in another. In the current system, characters are assigned only one score (for the class they are present in). It might improve accuracy if such characters have an appropriate negative score for the other class.

Finally, the average sentiment score may be used for sentiment scaling. For example, if in our experiments items with a score less than 1.6 were considered not to be opinionated, then ones with score more than 1.6 can be put on a scale where higher scores are interpreted as evidence for higher sentiment intensity (the highest score was 52). The "scaling" approach could help to avoid the problem of assigning documents to more than one sentiment category as the approach uses a continuous scale rather than a predefined number of rigid classes. The scale (or the scores directly) may be used as a means of indexing for a search engine comprising OIR functionality.

References

Kushal Dave, Steve Lawrence, and David M. Pennock. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the International World Wide Web Conference*, pages 519 – 528, Budapest, Hungary. ACM Press.

Koji Eguchi and Victor Lavrenko. 2006. Sentiment retrieval using generative models. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, pages 345–354, Sydney, July.

Brian Eriksson. 2006. Sentiment classification of movie reviews using linguistic parsing. http://www.cs.wisc.edu/∼apirak/cs/cs838/eriksson_final.pdf.

Soo-Min Kim and Eduard H. Hovy. 2004. Determining the sentiment of opinions. In *Proceedings of COLING-04*, pages 1367–1373, Geneva, Switzerland, August 23-27.

Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. 2006a. Opinion extraction, summarization and tracking in news and blog corpora. In *Proceedings of AAAI-2006 Spring Symposium on Computational Approaches to Analyzing Weblogs*, volume AAAI Technical Report, pages 100–107, March.

Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. 2006b. Tagging heterogeneous evaluation corpora for opinionated tasks. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, pages 667–670, Genoa, Italy, May.

Wei Li. 2000. On Chinese parsing without using a separate word segmenter. *Communication of COLIPS*, 10:17–67.

Jian-Yun Nie, Jiangfeng Gao, Jian Zhang, and Ming Zhou. 2000. On the use of words and n-grams for Chinese information retrieval. In *Proceedings of the 5th International Workshop Information Retrieval with Asian Languages*, pages 141–148. ACM Press, November.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 271–278, Barcelona, Spain.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 79–86, University of Pennsylvania.

Peter D. Turney. 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, pages 417–424, Philadelphia, Pennsylvania.