Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 34 (2008) 2622–2629
Expert Systems with Applications
www.elsevier.com/locate/eswa

An empirical study of sentiment analysis for chinese documents

Songbo Tan *, Jin Zhang

Intelligent Software Department, Institute of Computing Technology, Chinese Academy of Sciences, P.O. Box 2704, Beijing 100080, PR China

Abstract

Up to now, there are very few researches conducted on sentiment classification for Chinese documents. In order to remedy this deficiency, this paper presents an empirical study of sentiment categorization on Chinese documents. Four feature selection methods (MI, IG, CHI and DF) and five learning methods (centroid classifier, K-nearest neighbor, winnow classifier, Naïve Bayes and SVM) are investigated on a Chinese sentiment corpus with a size of 1021 documents. The experimental results indicate that IG performs the best for sentimental terms selection and SVM exhibits the best performance for sentiment classification. Furthermore, we found that sentiment classifiers are severely dependent on domains or topics.

Keywords: Sentiment analysis; Information retrieval; Machine learning

1. Introduction

With the advent of the Web and the enormous growth of digital content in Internet, databases, and archives, text categorization has received more and more attention in information retrieval and natural language processing community. This kind of work has focused on topical categorization, attempting to sort documents according to their subjects (such as economics or politics) (Pang et al., 2002).

However, recent years have seen rapid growth in non-topical text analysis, in which characterizations are sought of the opinions, feelings, and attitudes expressed in a text, rather than just the subjects. A key problem in this area is sentiment classification, where a document is labelled as a positive ('thumbs up') or negative ('thumbs down') evaluation of a target object (film, book, product, etc.).

Up to now, many researches have been conducted on English document sentiment classification. These researches have fallen into two categories. The first (''machine learning techniques'') (Mullen & Collier, 2004; Pang et al., 2002) attempts to train a sentiment classifier based on occurrence frequencies of the various words in the documents. The other approach (''semantic orientation'') (Hatzivassiloglou & McKeown, 1997; Turney & Littman, 2002; Whitelaw, Garg, & Argamon, 2005) is to classify words into two classes, such as ''positive'' or ''negative'', and then count an overall positive/negative score for the text. If a document contains more positive than negative terms it is deemed as positive, and if the number of negative terms exceeds the number of positive terms it is assigned as negative.

On Chinese document, however, there is relatively little investigation conducted on sentiment classification. Through our energies on searching the Internet, only one people's work can be found, i.e., Ye et al. (2005, 2006).

In order to remedy this deficiency, we present an empirical study of sentiment classification on Chinese documents. Our attempt is to answer following questions:

Which feature selection method does perform best for sentiment classification on Chinese documents?

How many features are sufficient for this job?

Which learning method does perform best?

Can a sentiment classifier trained on one domain (e.g. education) perform well on another different domain (e.g. movie)?

* Corresponding author. Tel.: +86 10 62600928.
E-mail address: tansongbo@software.ict.ac.cn (S. Tan).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.05.028

In this work, we use four traditional feature selection methods (Yang & Pedersen, 1997), i.e., mutual information (MI), information gain (IG), CHI statistics (CHI), document frequency (DF), and five learning methods, i.e., K-nearest neighbor (KNN) (Yang & Lin, 1999), centroid classifier (Han & Karypis, 2000), Naïve Bayes (NB) (McCallum & Kamal, 1998), winnow (Zhang, 2001) and support vector machine (SVM) (Joachims, 1998). We conduct experiments on a Chinese sentiment corpus with a size of 1021 documents divided into three domains: education, movie, and house.

The rest of this paper is constructed as follows: next section presents related work on sentiment analysis. Feature selection and learning methods are described in Section 3. Experimental results are given in Section 4. Finally Section 5 concludes this paper.

2. Related work

In this section, we present the related work on sentiment categorization. Sentiment classification has been investigated in different domains such as movie reviews, product reviews, and customer feedback reviews (Gamon, 2004; Pang et al., 2002; Turney & Littman, 2003).

Most of these researches up to this point have focused on training machine learning algorithms to classify reviews (Pang et al., 2002).

Pang et al. (2002) conducted an extensive experiment on movie reviews using three traditional supervised machine learning methods (i.e., Naive Bayes (NB), maximum entropy classification (ME), and support vector machines (SVM)). His results indicate that standard machine learning techniques definitively outperform human-produced baselines. However, he found that machine learning methods could not perform as well on sentiment classification as on traditional topic-based categorization. It is worth noticing that he collected a movie-reviewing corpus that contains 700 positive reviews and 700 negative reviews. This corpus has been becoming the benchmark for sentiment categorization research.

Mullen and Collier (2004) employed support vector machines (SVMs) to bring together diverse sources of potentially pertinent information, including several favorability measures for phrases and adjectives and, where available, knowledge of the topic of the text. Models using the features introduced are further combined with unigram models and lemmatized versions of the unigram models. His experiment on Pang's dataset indicates that hybrid SVMs which combine unigram-style feature based SVMs with those based on real-valued favorability measures obtain superior performance.

Research has also been done by counting positive/negative terms and automatically determining whether a term is positive or negative (Turney & Littman, 2002).

Kennedy and Inkpen (2005) made use of a semantic lexicon for identifying positive and negative terms. This lexicon is taken from the General Inquirer (Stone, Dunphy, Smith, & Ogilvie, 1966) (GI for simplicity), which is a dictionary that contains information about English word senses. The method he used to classify a review is to count positive and negative terms in it, as well as take into account contextual valence shifters. Valence shifters are terms that can change the semantic orientation of another term, such as not, never, none, nobody, etc. In his experiment he used two datasets: one is collected by himself; the other is Pang's dataset.

Chaovalit and Zhou (2005) compared machine learning based methods and orientation based methods. He used n-gram model as supervised learning approach. The n-gram represents text documents by word tuples. For orientation-based methods, he first used Minipar[1] to tag and parse the review documents and then selectively extracted two-word phrases conforming to phrase patterns from Turney's study (Turney, 2002). His method to determine the semantic orientation is the same as Turney. For example, a phrase's semantic orientation would be positive if it is associated more strongly with "excellent" than "poor" and would be negative if it is associated more strongly with "poor" than "excellent".

3. Methodology

This section presents the methodology of sentiment classification system we used. First, we use Chinese text POS tool ICTCLAS (Zhang, 2003) to parse and tag Chinese review documents. Then feature selection method is used to pick out discriminating terms for training and classification. Finally we use machine learning method to learn a sentiment classifier.

3.1. Feature selection methods

In this sub-section, we give a brief introduction of four effective feature selection methods, i.e., DF, CHI, MI and IG. All these methods compute a score for each individual feature and then pick out a predefined size of feature set.

3.1.1. DF

Document Frequency is the number of documents in which a term occurs in a dataset. In Document Frequency Thresholding one computes the document frequency for each word in the training corpus and removes those words whose document frequency is less than some predefined small threshold or bigger than some predefined large threshold. The basic assumption is that both rare and common words are either non-informative for category prediction, or not influential in global performance.

It is the simplest criterion for term selection and can easily scales to a large dataset with linear computation complexity. It is a simple but effective feature selection method for text categorization.

[1] http://www.cs.ualberta.ca/~lindek/minipar.htm.

3.1.2. CHI

The CHI statistic measures the association between the term and the category (Galavotti, Sebastiani, & Simi, 2000). It is defined to be

$$\mathrm{CHI}(t,c_i)=\frac{N\times(AD-BE)^2}{(A+E)\times(B+D)\times(A+B)\times(E+D)}$$
$$\text{and}\quad \mathrm{CHI}_{\max}(t)=\max_i(\mathrm{CHI}(t,c_i))$$

where A is the number of times t and c_i co-occur; B is the number of times t occurs without c_i; E is the number of times c_i occurs without t; D is the number of times neither c_i nor t occurs; N is the total number of documents.

3.1.3. MI

Mutual information is a criterion commonly used in statistical language modeling of word associations and related applications (Yang & Pedersen, 1997). It can be defined as following,

$$\mathrm{MI}(t,c_i)=\log\left(\frac{A\times N}{(A+E)\times(A+B)}\right)\quad\text{and}$$
$$\mathrm{MI}_{\max}(t)=\max_i(\mathrm{MI}(t,c_i))$$

where the denotations of A, B, E, D and N are the same as the definitions in CHI.

3.1.4. IG

Information gain is frequently employed as a term goodness criterion in the field of machine learning (Yang & Pedersen, 1997). It measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term in a document.

$$\mathrm{IG}(t)=-\sum_{i=1}^{|C|}P(c_i)\log P(c_i)+P(t)\sum_{i=1}^{|C|}P(c_i|t)\log P(c_i|t)+P(\bar{t})\sum_{i=1}^{|C|}P(c_i|\bar{t})\log P(c_i|\bar{t})$$

where $P(c_i)$ denotes the probability that class c_i occurs; $P(t)$ denotes the probability that word t occurs; $P(\bar{t})$denotes the probability that word t does not occurs.

3.2. Machine learning methods

3.2.1. Centroid classifier

The idea behind the centroid classification algorithm is extremely simple and straightforward. First we calculate the prototype vector or centroid vector for each training class; then compute the similarity between a testing document d to all centroids; finally, based on these similarities, we assign d to the class corresponding to the most similar centroid.

In training phase, we compute K centroids $\{C_1,C_2,\ldots,C_K\}$ for the K classes using following formula:

$$C_i=\frac{1}{|c_i|}\sum_{d\in c_i}d$$

where $|z|$ indicates the cardinality of set z, and d denotes the document in class c_i.

For each test document d, we calculate its similarity to each centroid C_i using cosine measure as follows:

$$\mathrm{sim}(d,C_i)=\frac{d\cdot C_i}{\|d\|_2\|C_i\|_2}$$

3.2.2. K-nearest neighbor classifier

The K-nearest neighbor (KNN) is a typical example-based classifier that does not build an explicit, declarative representation of the category c_i, but rely on the category labels attached to the training documents similar to the test document. As a result, KNN has been called lazy learners, since it defers the decision on how to generalize beyond the training data until each new query instance is encountered.

Given a test document d, the system finds the k nearest neighbors among training documents. The similarity score of each nearest neighbor document to the test document is used as the weight of the classes of the neighbor document. The weighted sum in KNN classification can be written as follows:

$$\mathrm{score}(d,c_i)=\sum_{d_j\in\mathrm{KNN}(d)}\mathrm{sim}(d,d_j)\delta(d_j,c_i)$$

where KNN(d) indicates the set of k nearest neighbors of document d. If d_j belongs to c_i, $\delta(d_j,c_i)$ equals 1, or otherwise 0. For test document d, it should belong to the class that has the highest resulting weighted sum.

3.2.3. Naïve Bayes

The Naïve Bayes algorithm is a widely used algorithm for document classification. Given a feature vector table, the algorithm computes the posterior probability that the document belongs to different classes and assigns it to the class with the highest posterior probability. There are two commonly used models (i.e., multinomial model and multi-variate Bernoulli model) for using Naïve Bayes approach for text categorization. In this paper, and without loss of generality, we run the multinomial model adopted by numerous authors (McCallum & Kamal, 1998).

Multinomial Naïve Bayes counts the probability of the word w_t given category c_j by following formula:

$$p(w_t|c_j)=\frac{\sum_{i=1}^{N_j}n_{it}}{\sum_{s=1}^{W}\sum_{i=1}^{N_j}n_{is}}$$

where n_{it} is the number of appearances of word t in document i, N_j refers to the number of training documents in category c_j and W refers to the vocabulary size.

The posterior probability can be calculated as follows:

$$p(c_j|d_i) = \frac{p(c_j)p(d_i|c_j)}{p(d_i)}$$

3.2.4. Winnow classifier

Winnow is a well-known online mistaken-driven method. It works by updating its weights in a sequence of trials. On each trial, it first makes a prediction for one document d and then receives feedback; if a mistake is made, it updates its weight vector using the document d. During the training phase, with a collection of training data, this process is repeated several times by iterating on the data. Up to now, there are many variants of winnow, such as positive winnow, balanced winnow, and large-margin winnow. In this work we only run balanced winnow because it consistently yields excellent performance (van Mun, http://citeseer.ist.psu.edu/cs).

The balanced winnow algorithm keeps two weights for each feature, w_{kt}^+ and w_{kt}^-. For a given instance $(d_{k1}, d_{k2}, \ldots, d_{kW})$, the algorithm deems the document relevant iff

$$\sum_{t=1}^{W}(w_{kt}^+ - w_{kt}^-)d_{kt} \geqslant \tau$$

where τ denotes a given threshold and k indicates the class label.

The weights of the active features are updated only when a mistake is made. In the promotion step, following a mistake on a positive example, the positive part of the weight is promoted, $w_{kt}^+ = w_{kt}^+ \times \alpha(\alpha > 1)$ while the negative part of the weight is demoted, $w_{kt}^- = w_{kt}^- \times \beta(0 < \beta < 1)$. The coefficient of d_{kt}in the equation above increases after a promotion. In the demotion step, by contraries, the positive part of the weight is demoted, while the negative part of the weight is promoted.

3.2.5. SVM classifier

Support vector machines (SVM) is a relatively new class of machine learning techniques first introduced by Vapnik (1995). Based on the structural risk minimization principle from the computational learning theory, SVM seeks a decision surface to separate the training data points into two classes and makes decisions based on the support vectors that are selected as the only effective elements in the training set.

Multiple variants of SVM have been developed (Joachims, 1998). Here we limit our discussion to linear SVM due to its popularity and high performance in text categorization (Yang & Lin, 1999).

The optimization of SVM (dual form) is to minimize:

$$\vec{\alpha}^* = \arg\min\left\{-\sum_{i=1}^{n}\alpha_i + \sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \langle \vec{x_i}, \vec{x_j}\rangle\right\}$$

$$\text{Subject to: } \sum_{i=1}^{n}\alpha_i y_i = 0; \quad 0 \leqslant \alpha_i \leqslant C$$

4. Experiment results

4.1. Datasets

Because there is a lack of publicly available Chinese sentiment corpus for evaluating sentiment analysis systems, we collect a Chinese sentiment corpus by ourselves. For the sake of convenience, we call this corpus as "ChnSentiCorp". The total size is 1021 documents that consist of three domains: education, movie, and house. There are 507 education-related documents, 266 movie-related documents and 248 house-related documents. Each domain category contains positive and negative documents. The hierarchy is reported as Table 1. The total positive documents amount to 458; while the total negative documents amount to 563. As a result, the corpus can be regarded as four sentiment corpora: ChnSentiCorp, ChnSentiCorpEdu, ChnSentiCorpMov, and ChnSentiCorpHou.

4.2. The performance measure

To evaluate a semantic classification system, we use the F1 measure introduced by van Rijsbergen (1979). This measure combines recall and precision in the following way:

$$\text{Recall} = \frac{\text{number of correct positive predictions}}{\text{number of positive examples}}$$

$$\text{Precision} = \frac{\text{number of correct positive predictions}}{\text{number of positive predictions}}$$

$$\text{F1} = \frac{2 \times \text{Recall} \times \text{Precision}}{(\text{Recall} + \text{Precision})}$$

For ease of comparison, we summarize the F1 scores over the different categories using the micro- and macro-averages of F1 scores:

$$\text{Micro-F1} = \text{F1 over categories and documents}$$

$$\text{Macro-F1} = \text{average of within-category F1 values}$$

The MicroF1 and MacroF1 emphasize the performance of the system on common and rare categories, respectively. Using these averages, we can observe the effect of different kinds of data on a classification system.

Table 1
The size of collected four sentiment corpora

Domains	Sentiment	Documents
ChnSentiCorpEdu	Positive	204
	Negative	303
ChnSentiCorpMov	Positive	113
	Negative	153
ChnSentiCorpHou	Positive	141
	Negative	107
ChnSentiCorp	Positive	458
	Negative	563

4.3. Experimental design

We employ TFIDF as input features. The formula for calculating the TFIDF can be written as follows:

$$W(t,d) = tf(t,d) \times \log(N/n_t)$$

where N is the total number of training documents, and n_t is the number of documents containing the word t.

For Balanced Winnow, the initial weight value $w_{il}^{+}(w_{il}^{-})$ is set 2.0 (1.0), and the threshold was set to 1.0. The promotion parameter α and the demotion β were fixed as 1.2 and 0.8, respectively.

For KNN, we set the number k of neighbors to 13. It is worth noticing that we do not introduce any thresholds investigated by Yang (2001) because the adjusting of thresholds may incur significant computational costs.

For experiments involving SVM we employed SVMTorch, which uses one-versus-the-rest decomposition and can directly deal with multi-class classification problems. (http://www.idiap.ch/~bengio/projects/SVMTorch.html).

4.4. Comparison and analysis

Tables 2 and 3 report the best performance of four feature selection methods combined with five learning methods. The corpus we used is ChnSentiCorp as introduced in Section 4.1. From the two tables we can draw following conclusions:

Table 2
The best MicroF1 of four feature selection methods on five learning methods

	Centroid	KNN	Winnow	NB	SVM	Average
MI	0.8129	0.7943	0.8090	0.8012	0.8257	0.8086
IG	0.8736	0.8756	0.8981	0.8883	0.9060	**0.8883**
CHI	0.8658	0.8433	0.8776	0.8913	0.8903	0.8737
DF	0.8668	0.8511	0.8805	0.8717	0.8521	0.8644
Average	0.8548	0.8411	0.8663	0.8631	**0.8685**	

Table 3
The best MacroF1 of four feature selection methods on five learning methods

	Centroid	KNN	Winnow	NB	SVM	Average
MI	0.8084	0.7841	0.8049	0.7866	0.8244	0.8017
IG	0.8681	0.8730	0.8996	0.8840	0.9043	**0.8858**
CHI	0.8602	0.8404	0.8739	0.8882	0.8888	0.8703
DF	0.8612	0.8468	0.8777	0.8644	0.8480	0.8596
Average	0.8495	0.8361	0.8640	0.8558	**0.8664**	

First, with respect to feature selection methods, IG performs the best across almost learning methods. Its average MicroF1 is 0.8883, which is one percent larger that CHI (0.8737), two percents larger than DF (0.8644), and eight percents larger than MI (0.8086). The answer for the first question discussed in the introduction section is that IG is the best choice for semantic terms selection.

As such, with respect to learning methods, SVM produces the best average MicroF1 (0.8685) which is slightly higher than Winnow (0.8663) or NB (0.8631). This observation indicates that SVM, Winnow, and NB are all suitable for sentiment analysis.

Figs. 1–5 displays the performance curves of five learning methods using IG vs. feature number. The corpus we used is ChnSentiCorp as introduced in Section 4.1. From these figures we can observe that when the number of feature exceeds 6000, all learning methods produce desirable and reasonable performance. For example, using a feature set larger than 6000, the performance curves of SVM combined with IG, CHI, and DF keeps nearly unchanged. Consequently, our answer for the second question is that 6000 or larger size of features is sufficient for sentiment categorization.

The second observation is that three feature selection methods (i.e., IG, CHI, and DF) combined with four learning methods (i.e., Centroid, KNN, Winnow, and NB) exhibit similar and robust performance. Under all conditions, MI does not have comparable performance with any of the other methods.

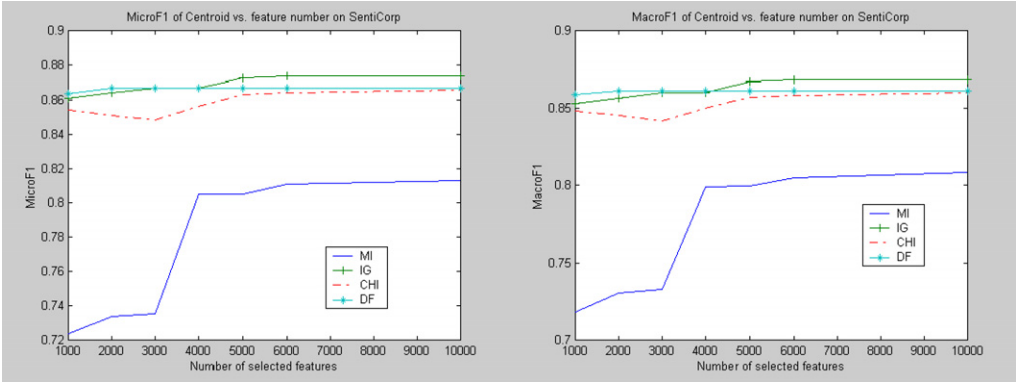

Fig. 1. The performance curves of centroid using IG vs. feature number.

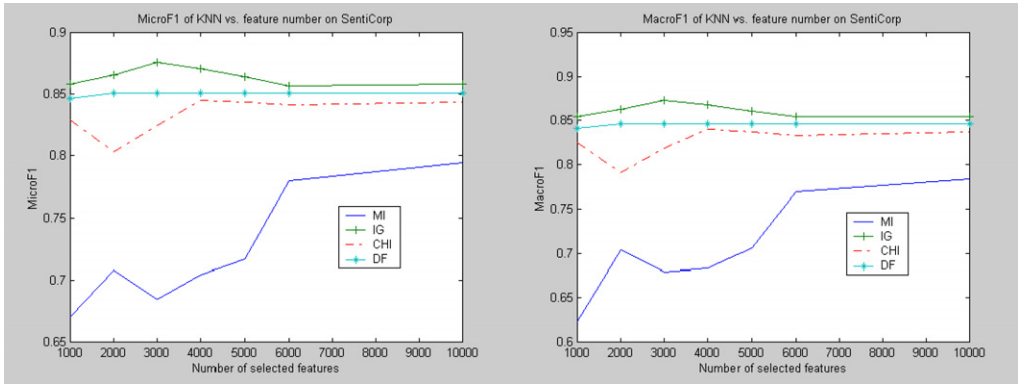

Fig. 2. The performance curves of KNN using IG vs. feature number.

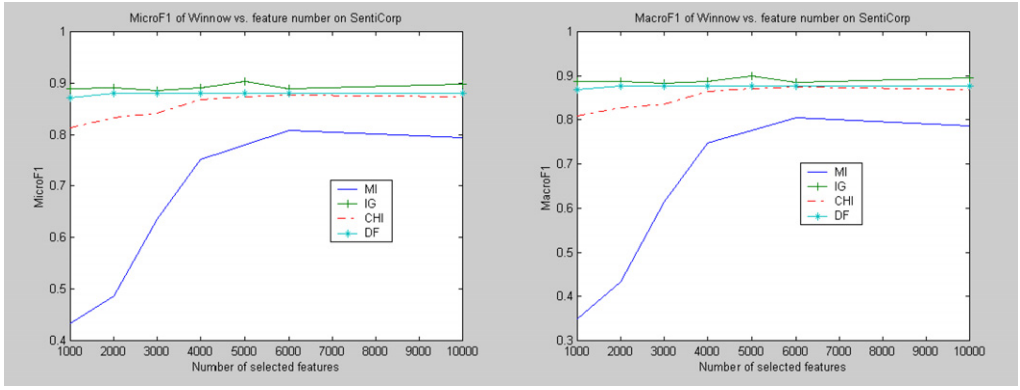

Fig. 3. The performance curves of winnow using IG vs. feature number.

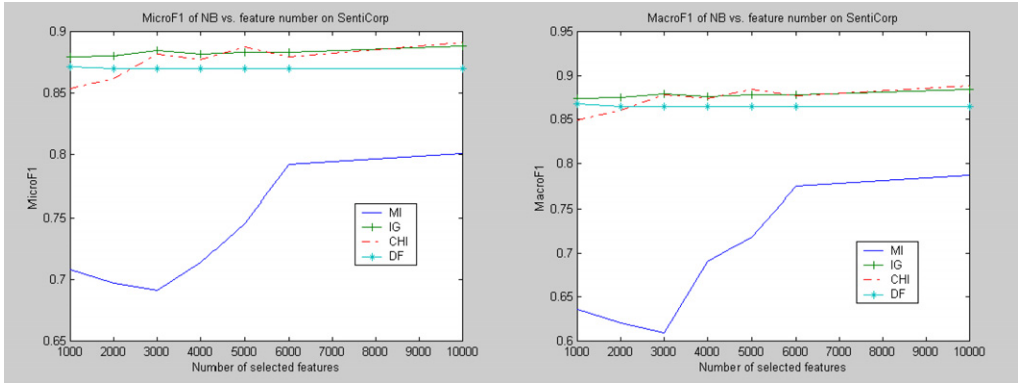

Fig. 4. The performance curves of NB using IG vs. feature number.

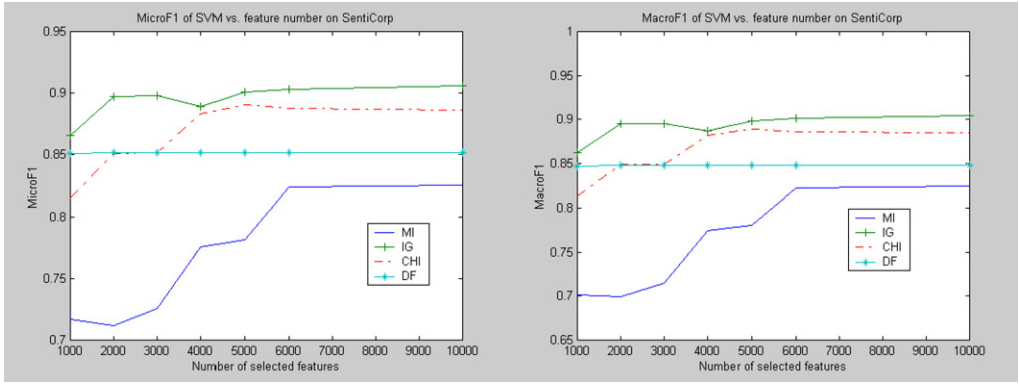

Fig. 5. The performance curves of SVM using IG vs. feature number.

Table 4
The performance of SVM when trained on one domain and transferred to another domain

Training set	Testing set					
	ChnSentiCorpEdu		ChnSentiCorpMov		ChnSentiCorpHou	
	MicroF1	MacroF1	MicroF1	MacroF1	MicroF1	MacroF1
ChnSentiCorpEdu	–	–	0.6165	0.4753	0.8992	0.8982
ChnSentiCorpMov	0.7475	0.7058	–	–	0.7339	0.7328
ChnSentiCorpHou	0.7574	0.7573	0.4962	0.4653	–	–

Table 4 presents the performance of domain transfer of SVM. The number of features is fixed as 10,000. The row "Training Set" indicates the domains for training the SVM classifier; the column "Testing Set" indicates the domains for testing the SVM classifier.

Compared with above results involved with SVM, the performance of Table 4 is not desirable at all. Especially when using "ChnSentiCorpHou" as training set and using "ChnSentiCorpMov" as testing set, SVM yields very poor performance, 0.4962 for MicroF1 and 0.4653 for MacroF1. The only reasonable result is achieved when "ChnSentiCorpEdu" is employed as training set and "ChnSentiCorpHou" is regarded as testing set.

These discouraging results when the classifier is transferred across different domains inspire us a rule: machine learning techniques based sentiment classifier is severely dependent on its domains or topics.

5. Conclusion remarks

In this work, we conducted an empirical study of sentiment categorization on Chinese documents. Our main contributions are:

In order to conduct this experiment, we by ourselves collect Chinese corpus with a size of 1021 documents. It consists of news or reviews from three domains: education, movie, and house.

Secondly, we found that IG performs the best for sentimental terms selection and SVM exhibits the best performance for sentiment classification.

Thirdly, the experimental results indicate that 6000 or larger size of features are sufficient for sentiment analysis.

Finally, we found that sentiment classifiers are severely dependent on domains or topics.

With the consideration of last conclusion, our future effort is to investigate how to train a sentiment classifier that is independent on domain or topics. Another alternative to this issue is to construct a semantic lexicon.

References

Chaovalit, Pimwadee, & Zhou, Lina (2005). Movie review mining: A comparison between supervised and unsupervised classification approaches. In *IEEE proceedings of the 38th Hawaii international conference on system sciences, Big Island, Hawaii* (pp. 1–9).

Galavotti, L., Sebastiani, F., & Simi, M. (2000). Feature selection and negative evidence in automated text categorization. In *Proceedings of KDD*.

Gamon, Michael (2004). Sentiment classification on customer feedback data: Noisy data,large feature vectors, and the role of linguistic analysis. In *Proceedings the 20th international conference on computational linguistics*.

Han, E., & Karypis, G. (2000). Centroid-based document classification analysis and experimental result. *PKDD*.

Hatzivassiloglou, Vasileios, & McKeown, Kathleen (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the 35th ACL/8th EACL* (pp. 174–181).

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. *ECML*, 137–142.

Kennedy, Alistair, & Inkpen, Diana (2005). Sentiment classification of movie and product reviews using contextual valence shifters. In *Workshop on the analysis of informal and formal information exchange during negotiations (FINEXIN 2005)*.

McCallum, Andrew, & Nigam, Kamal (1998). A comparison of event models for Naive Bayes text classification. *AAAI/ICML-98 workshop on learning for text categorization [C]* (pp. 41–48). Menlo Park, CA: AAAI Press.

Mullen, Tony, & Collier, Nigel (2004). Sentiment analysis using support vector machines with diverse information sources. In Dekang Lin & Dekai Wu (Eds.). *Proceedings of EMNLP-2004, Barcelona, Spain, July 2004* (pp. 412–418). Association for Computational Linguistics.

Pang, Bo, Lee, Lillian, & Vaithyanathan, Shivakumar (2002). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of EMNLP, 2002*.

Stone, Philip J., Dunphy, Dexter C., Smith, Marshall S., & Ogilvie, Daniel M.and associates. (1966). *The general inquirer: A computer approach to content analysis*. The MIT Press.

Turney, Peter D. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the association for computational linguistics 40th anniversary meeting, New Brunswick, NJ*.

Turney, D. Peter, & Littman, Michael L. (2002). Unsupervised learning of semantic orientation from a hundred-billion-word corpus. Technical Report EGB-1094, National Research Council Canada.

Turney, Peter D., & Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS), 21*(4), 315–346.

van Mun, P. P. T. M. Text classification in information retrieval using winnow. http://citeseer.ist.psu.edu/cs.

van Rijsbergen, C. (1979). *Information retrieval*. London: Butterworth.

Vapnik, Vladmimir N. (1995). *The nature of statistical learning theory*. New York: Springer.

Whitelaw, Casey, Garg, Navendu, & Argamon, Shlomo (2005). Using appraisal groups for sentiment analysis. *CIKM*, 625–631.

Yang, Y. (2001). A study on thresholding strategies for text categorization. *SIGIR*, 137–145.

Yang, Y., & Lin, X. (1999). A re-examination of text categorization methods. *SIGIR*, 42–49.

Yang, Y., & Pedersen, Jan O. (1997). A comparative study on feature selection in text categorization. *ICML*, 412–420.

Ye, Qiang, Lin, Bin, & Li, Yijun (2005). Sentiment classification for Chinese reviews: A comparison between SVM and semantic

approaches. In *The 4th international conference on machine learning and cybernetics ICMLC2005, June 2005*.

Ye, Qiang, Shi, Wen, & Li, Yijun (2006). Sentiment classification for movie reviews in Chinese by improved semantic oriented approach. In *Proceedings of HICSS-39 Hawaii international conference on system sciences, January 2006*.

Zhang, T. (2001). Regularized winnow methods. *Advances in Neural Information Processing Systems, 13*, 703–709.

Zhang, Huaping (2003). Chinese lexical analysis using hierarchical hidden Markov model. In *Second SIGHAN workshop affiliated with 41th ACL, Sapporo Japan, July 2003* (pp. 63–70).