Expert Systems with Applications 41 (2014) 3506–3513

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A study of supervised term weighting scheme for sentiment analysis

Zhi-Hong Deng*, Kun-Hu Luo, Hong-Liang Yu

Key Laboratory of Machine Perception (Ministry of Education), Department of Machine Intelligence, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China

ARTICLE INFO

Keywords:
Sentiment analysis
Term weighting
Supervised learning
Experimentation
Performance

ABSTRACT

Term weighting is a strategy that assigns weights to terms to improve the performance of sentiment analysis and other text mining tasks. In this paper, we propose a supervised term weighting scheme based on two basic factors: Importance of a term in a document (*ITD*) and importance of a term for expressing sentiment (*ITS*), to improve the performance of analysis. For *ITD*, we explore three definitions based on term frequency. Then, seven statistical functions are employed to learn the *ITS* of each term from training documents with category labels. Compared with the previous unsupervised term weighting schemes originated from information retrieval, our scheme can make full use of the available labeling information to assign appropriate weights to terms. We have experimentally evaluated the proposed method against the state-of-the-art method. The experimental results show that our method outperforms the method and produce the best accuracy on two of three data sets.

1. Introduction

With the rapid growth of Web 2.0, users generate huge numbers of online information in the forms of reviews, blogs, tweets, etc. This wealth of information includes users' opinions about events, products, or people. It provides new opportunities for institutions and companies to understand their consumers, improve product quality and enhance their competitiveness. Sentiment analysis (or opinion mining), which deals with the computational treatment of opinion, sentiment, and subjectivity in text (Pang & Lee, 2008), has emerged as a direct response to such new situation. Since the pioneering work (Das & Chen, 2001; Pang, Lee, & Vaithyanathan, 2002; Pang & Lee, 2004; Tong, 2001; Turney, 2002) was published, there have been hundreds of follow-up research publications. Sentiment analysis has become a popular research topic and enjoyed a huge burst of research activity .

The representation of documents is a critical component of many text mining tasks including sentiment analysis. Usually, Vector Space Model (VSM) is adopted to represent documents. By VSM, each document is represented by a vector of term (or feature) space. The weight (or value) of each term in vectors is the key component of the VSM of document representation. Early work by Pang, Lee, and Vaithyanathan. (2002) show that binary weight scheme provides good accuracy. Recent research has focused on more complex term weighting methods including methods from information retrieval (Martineau & Finin, 2009; Paltoglou & Thelwall, 2010), and based on Learning term weight by optimization (Maas et al., 2011).

Although the complex term weighting methods from information retrieval have proved to be effective and achieve the best accuracy so far, the authors Paltoglou and Thelwall, (2010) indicated that these methods are not sufficiently intuitive because they only provide information about the general distribution of terms without providing any additional evidence of class preference. As we know, most of the work in sentiment analysis has focused on supervised learning task (Sebastiani, 2002) where the category labels of training documents are available in advance. Intuitively, if this known information are used to weight terms, we may get better term weights than those methods from information retrieval, which generates term weights without using any available labeling information. Maas et al. first study the problem of learning term weights by using available labeling information (Maas et al., 2011). They transform the problem of learning term weights into the problem of maximizing an objective function, which can be handled by traditional techniques for solving optimization problem. However, their method has two issues. The first, the learning model adopted by them is so complex that it is easily overfitting. That is, their method is not able to achieve good accuracy on documents out of the training ones. In fact, experiments show that their method does not perform better than binary weighting method. The second, it is highly time-consuming to solve optimization problem for large-scale and high-dimension datasets. However, the document sets, which sentiment analysis faces, are typical large-scale and high-dimension datasets.

Based on the above discussion, a question is raised. That is, can we get good term weights, which perform better than the

[*] Corresponding author. Tel.: +86 10 62755592; fax: +86 10 62754911.
E-mail address: zhdeng@cis.pku.edu.cn (Z.-H. Deng).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.10.056

state-of-the-art unsupervised methods from information retrieval, by using simple supervised learning models? This is the motivation of our study.

After some careful examination, we believe that such a goal can be achieved. In this paper, we propose a supervised term weighting scheme for sentiment analysis. In the proposed scheme, the weighting of each term (feature) is measured by two factors: *ITD* and *ITS*. The *ITD* reflects the importance of a term in a document, while the *ITS* reflects the importance of a term for expressing sentiment. For *ITD*, we introduce three definitions based on term frequency. Furthermore, we employ six statistical functions to learn *ITS* from labelled training documents. Finally, we conduct extensive experiments to evaluate the performance of our proposed scheme. Experimental results show it outperforms the previous state-of-the-art unsupervised method. To the best of our knowledge, this paper is the first work that presents the study of supervised term weighting scheme based on statistical functions for sentiment analysis.

The remainder of the paper is organized as follows. Section 2 presents related work. Section 3 introduces the supervised term weighting scheme and relevant concepts in details. Experimental results and performance study are presented and analyzed in Section 4. Section 5 summarizes our study and points out some future research issues.

2. Related work

Term weighting is the problem of automatically assigning weights (values) to terms. These weights measure the importance of terms and denote how much these terms contribute to many tasks associated with documents, such as information retrieval, text classification, text clustering, and sentiment analysis. Term weighting is a basic problem in information retrieval and text mining. It has proved to be effective on improving the performance of many tasks.

In information retrieval, term weighting has been a fertile field of research and development since the 1970s. Salton discussed three factors that should be considered for term weighting (Salton & Buckley, 1988). The three factors are term frequency, inverse document frequency, and normalization. Based on the above assumption, a lot of methods of term weighting have been proposed for information retrieval. The simplest one is the binary representation, which only concerns whether a term occurs in a document. *tf∗idf* proposed by Jones (1972) is the most widely used one. A famous variant of *tf∗idf* is BM25 (Robertson, Zaragoza, & Taylor, 2004). The BM25 method is a probabilistic model for information retrieval originated from RSJ (Robertson & Jones, 1976) and is one of the most popular and effective weighting methods used in information retrieval. It has proved to be the best one in TREC (Robertson, Walker, Jones, Hancock-Beaulieu & Gatford, 1994; Robertson, Walker, Jones, Hancock-Beaulieu & Gatford, 1996). Term weighting is still a very active research area in information retrieval. Recent research progress can be found in Armstrong, Moffat, Webber, and Zobel (2009), Manning, Raghavan, and Schutze (2008), Robertson et al. (2004), Manning et al. (2008), or Baeza-Yates & Ribeiro-Neto (2011) for a more in-depth analysis and study.

In text classification, the standard *tf∗idf* method is widely used (Sebastiani, 2002). However, text classification is a kind of supervised learning task where the category labels of training documents are available in advance. Therefore, some researchers focus on weighting terms by learning from labeled training documents. One main approach is to weight terms by employing methods of feature selection, such as information gain, mutual information, χ^2 statistic. As we know, the goal of feature selection is to reduce the high dimensionality of term (feature) space by selecting the most relevant terms for text classification. The higher a score of a term has, the more it contributes to the classification task. Debole and Sebastiani (2003) replaced *idf* with information gain, χ^2 statistic and gain ratio. In Debole and Sebastiani (2003), these supervised term weighting methods have not shown a consistent superiority over the standard *tf∗idf*. Similarly, Deng et al. (2004) studied the performance of text classification by replacing *idf* with some feature selection functions. They reported that *tf∗CHI* is more effective than *tf∗idf* in their experiments with SVM as the classifier and benchmark Reuter-21578 as the document set. Soucy and Mineau (2005) introduced *ConfWeight*, a new term weighting method based on statistical confidence intervals. Their experimental results showed that *ConfWeight* outperformed *tf∗idf* on three document sets. Lan, Tan, Su, and Lu (2009) proposed a new simple supervised term weighting method, *tf∗rf*, to improve the terms' discriminating power for text categorization task. The experimental results show that *tf∗rf* has a consistently better performance than *tf∗idf* and other supervised term weighting methods. In recent, Ren and Sohrab proposed a Class-indexing-based term weighting scheme for automatic text classification (Ren & Sohrab, 2013). In this scheme, a term weight is computed by multiplying its *tf∗idf* by its ICS_{δ}, which is the inverse class space density frequency of the item.

Recently, sentiment analysis has become a popular research topic because of its wide applications. The purpose of sentiment analysis is to classify the attitude expressed in the text (such as positive or negative) rather than some facts (such as entertainment or sport). The work in sentiment analysis has mainly focused on supervised learning techniques (Pang & Lee, 2008). One of the main issues for supervised approaches is the representation of documents while the core problem of the document representation is to weight terms. In early work (Pang et al., 2002), Pang indicated that a binary unigram based representation of documents with binary weight provides the best baseline classification accuracy in sentiment analysis in comparison to other more intricate representations. In recent years, research has focused on more efficient term weighting methods to improve the performance of sentiment analysis. Paltoglou and Thelwall (2010) examine the issue that whether term weighting functions adopted from information retrieval based on the standard *tf∗idf* formula can help improve accuracy. Their experimental results show that the term weighting methods based on BM25, a variant of the original *tf∗idf*, can provide significant increases in the performance of sentiment analysis. Moreover, Maas presented a model (Maas et al., 2011) that uses a mix of unsupervised and supervised techniques to learn word vectors capturing semantic term-document information as well as rich sentiment content. In fact, the authors transform the problem of learning term weights into an optimization problem. Their experimental results show that the proposed method does not have a better performance than simple binary weighting method and Delta *tf∗idf* (Martineau & Finin, 2009).

3. Supervised weighting scheme

In this section, we describe the supervised weighting scheme for sentiment analysis. In the supervised weighting scheme, the weighting of each term (word) is measured by two factors: *Importance of a term in a document* (*ITD*) *and importance of a term for expressing sentiment (ITS)*. *ITD* reflects the importance of a term in a document, while *ITS* reflects the importance of a term for expressing sentiment. In our scheme, we employ function based on *term frequency* to compute *ITD*. For computing *ITS*, we introduce seven statistical functions to learn the sentimental importance of a term by its statistical distribution in positive documents and negative documents. These statistical functions are: document frequency (*DF*), information gain (*IG*), mutual information (*MI*), odds ratio (*OR*), χ^2 statistic (*CHI*), Weighted Log Likelihood Ratio (*WLLR*) and Weighed Frequency and Odds (*WFO*).

3.1. Preliminary

Let's denote the set of positive documents by D^1, and the set of negative documents by D^2. Assume $F = \{f_1, f_2, \ldots, f_m\}$ is the vocabulary, which is set of unique words in $D^1 \cup D^2$, In this paper, a document d_j is represented by a bag-of-words term vector: $d_j = (w_{1j}, w_{2j}, \ldots, w_{mj})$, where w_{ii} stands for the weight of f_i in d_j. In our supervised weighting scheme, w_{ii} is defined as

$$w_{ij} = ITD(f_i, d_j) \times ITS(f_i) \tag{1}$$

where $ITD(f_i, d_j)$ stands for the importance of f_i in d_j and $ITS(f_i)$ means the importance of f_i for expressing sentiment. These formal definitions of $ITD(\cdot)$ and $ITS(\cdot)$ are based on some probabilities. For the convenience of better description, we introduce some notations of these probabilities as shown in Table 1.

However, we cannot directly obtain these probabilities. Therefore, we need some statistical information from the training data to estimate them. Some notations concerning related statistical information are given in Table 2.

Then, the estimation of probabilities in Table 1 is listed as below.

$$P(D^k) \approx \frac{N_k}{N_1 + N_2}; \quad P(f_i) \approx \frac{x_i^1 + x_i^2}{N_1 + N_2};$$

$$P(\overline{D^k}) = 1 - P(D^k) \approx 1 - \frac{N_k}{N_1 + N_2};$$

$$P(\bar{f_i}) = 1 - P(f_i) \approx 1 - \frac{x_i^1 + x_i^2}{N_1 + N_2};$$

$$P(f_i, D^k) \approx \frac{x_i^k}{N_1 + N_2}; \quad P(f_i|D^k) \approx \frac{x_i^k}{N_k};$$

$$P(D^k|f_i) \approx \frac{x_i^k}{x_i^1 + x_i^2}; \quad P(f_i|\overline{D^k}) \approx \frac{y_i^k}{N_1 + N_2 - N_k};$$

$$P(D^k|\bar{f_i}) \approx \frac{u_i^k}{N_1 + N_2 - x_i^1 - x_i^2}.$$

3.2. The definition of ITD(f_i, d_j)

In this paper, we choose *term frequency* to define $ITD(f_i, d_j)$. *Term frequency* is the classical method for evaluating the capability that a term describes the content of a document. It has proved useful in information retrieval, text classification, text clustering, and other text mining tasks. Three kinds of *term frequency* is used to define $ITD(f_i, d_j)$ as follows:

$$ITD(f_i, d_j) = \begin{cases} 1 & f_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$ITD(f_i, d_j) = tf_{ij} \tag{3}$$

$$ITD(f_i, d_j) = 0.5 + \frac{0.5 \times f_{ij}}{\max\limits_k f_{kj}} \tag{4}$$

Formula (2) uses the binary weight to define $ITD(f_i, d_j)$. That is, if term f_i occurs in document d_j, $ITD(f_i, d_j)$ is equal to 1; otherwise, $ITD(f_i, d_j)$ is equal to 0. Obviously, Formula (2) does not take into consideration the count that term f_i occurs in document d_j. Generally speaking, the count of a term in a document often indicates the importance of the term in presenting the context of the document. Formula (3) uses the raw term frequency of term f_i in document d_j to define $ITD(f_i, d_j)$. As we know, terms in long documents can occur more than in short documents, leading to unfairness. Previous experiments show that normalized term frequency is more effective than the rawversion. Formula (4) uses the normalization of the raw term frequency to define $ITD(f_i, d_j)$.

3.3. The definition of ITS(f_i)

As indicated in Paltoglou and Thelwall (2010), using *idf* to assign weights to terms only provides information about the general distribution of a term amongst documents of all classes, without providing any additional evidence of class preference. Therefore, we employ some weighting methods based on learning to define $ITS(f_i)$ instead of *idf*. These methods are based on statistical function to learn the importance of a term for expressing sentiment. Note that, these methods have been widely used for feature selection in text classification (Li, Xia, Zong, & Huang, 2009; Yang & Pedersen, 1997).

3.3.1. Symbols for related statistical information

Document Frequency (*DF*) is the number of documents in which a term occurs. We first computed the document frequency for term f_i in the training positive document set (D^1) and negative documents set (D^2) respectively. Then, we choose the bigger one to define $ITS(f_i)$. Formula (5) shows the definition.

$$ITS(f_i) = \max\{x_i^1, x_i^2\} \tag{5}$$

The way that Formula (5) directly use document frequency to define $ITS(\cdot)$ may result in some problems. For example, there are 100 positive documents and 100 negative documents. f_1 and f_2 are two terms. Term f_1 occurs in 65 positive documents and 60 negative documents. This means $ITS(f_1)$ is equal to 65. Term f_2

Table 1
Symbols for related probabilities.

Symbol	Meaning
$P(D^k)$	The probability that a document belongs to document set D^k.
$P(\overline{D^k})$	The probability that a document does not belong to document set D^k.
$P(f_i)$	The probability that term f_i occurs in a document.
$P(\bar{f}_i)$	The probability that term f_i does not occur in a document.
$P(f_i, D^k)$	The joint probability that a document both contains term f_i and belongs to document set D^k.
$P(f_i\|D^k)$	Given the condition that a document belongs to document set D^k, the probability that term f_i occurs in the document.
$P(f_i\|\overline{D^k})$	Given the condition that a document does not belong to document set D^k, the probability that term f_i occurs in the document.
$P(D^k\|f_i)$	Given the condition that a document contains term f_i, the probability that the document belongs to document set D^k.
$P(D^k\|\bar{f}_i)$	Given the condition that a document does not contain term f_i, the probability that the document belongs to document set D^k.

occurs in 30 positive documents and 0 negative documents. This means $ITS(f_2)$ is equal to 30. Therefore, we have $ITS(f_1) > ITS(f_2)$. As a result, the ability of f_1 for discriminating positive and negative documents is stronger than that of f_2. However, it should be the other way around by intuition because f_2 only occurs in positive documents. Our experiments also show that $ITS(\cdot)$ defined by Formula (5) is ineffective compared with other definitions.

3.3.2. Definition based on information gain

Information gain is employed as a criterion for evaluating the goodness of features in machine learning (Quinlan, 1986). It measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term in a document (Yang & Pedersen, 1997). In this paper, the information gain of term f_i is defined as

$$IG(f_i) = -\sum_{k=1}^{2} P(D^k)\log P(D^k) + P(f_i)\sum_{k=1}^{2} P(D^k|f_i)\log P(D^k|f_i) + P(\bar{f_i})\sum_{k=1}^{2} P(D^k|\bar{f_i})\log P(D^k|\bar{f_i}) \tag{6}$$

It can be estimated as

$$IG(f_i) \approx -\sum_{k=1}^{2}\frac{N_k}{N_1+N_2}\log\left(\frac{N_k}{N_1+N_2}\right) + \left(\frac{x_i^1+x_i^2}{N_1+N_2}\right)\sum_{k=1}^{2}\left(\frac{x_i^k}{x_i^1+x_i^2}\right)\log\left(\frac{x_i^k}{x_i^1+x_i^2}\right) + \left(1-\frac{x_i^1+x_i^2}{N_1+N_2}\right)\sum_{k=1}^{2}\left(\frac{u_i^k}{N_1+N_2-x_i^1-x_i^2}\right)\log\left(\frac{u_i^k}{N_1+N_2-x_i^1-x_i^2}\right) \tag{7}$$

For term f_i, $ITS(f_i)$ can be defined as $IG(f_i)$.

3.3.3. Definition based on mutual information

Mutual information is a classical criterion widely used in statistical language modelling of word associations (Church & Hanks, 1989). Given term f_i and document set D^k, the mutual information between them is defined as

$$MI(f_i, D^k) = \log\frac{P(f_i, D^k)}{P(f_i)\times P(D^k)} \tag{8}$$

It can be estimated as

$$MI(f_i, D^k) \approx \log\frac{\frac{x_i^k}{N_1+N_2}}{\frac{x_i^1+x_i^2}{N_1+N_2}\times\frac{N_k}{N_1+N_2}} = \log\frac{x_i^k\times(N_1+N_2)}{(x_i^1+x_i^2)\times N_k} \tag{9}$$

Based on mutual information, we define $ITS(f_i)$ as

$$ITS(f_i) = \max\{MI(f_i, D^1), MI(f_i, D^2)\} \tag{10}$$

3.3.4. Definition based on odds ratio

Odds ratio is commonly used in information retrieval where the problem is to rank out documents according to their relevance for the positive class value using occurrence of different words as features (Mladenic & Grobelnik, 1998; van Rijsbergen, Harper, & Porter, 1981). Given term f_i and document set D^k, the odds ratio between them is defined as

$$OR(f_i, D^k) = \log\frac{P(f_i|D^k)(1-P(f_i|\overline{D^k}))}{(1-P(f_i|D^k))P(f_i|\overline{D^k})} \tag{11}$$

It can be estimated as

$$OR(f_i, D^k) \approx \log\frac{x_i^k\times(N_1+N_2-N_k-y_i^k)}{(N_k-x_i^k)\times y_i^k} \tag{12}$$

Based on odds ratio, we define $ITS(f_i)$ as

$$ITS(f_i) = \max\{OR(f_i, D^1), OR(f_i, D^2)\} \tag{13}$$

3.3.5. Definition based on χ^2 statistic

χ^2 statistic (*CHI*) measures the lack of independence between two random variables and can be compared to the distribution with one degree of freedom to judge extremeness (Yang and Pedersen, 1997). The *CHI* between term f_i and document set D^k is defined as

$$CHI(f_i, D^k) = \frac{(N_1+N_2)\times(x_i^k v_i^k - y_i^k u_i^k)^2}{(x_i^k+u_i^k)\times(y_i^k+v_i^k)\times(x_i^k+y_i^k)\times(u_i^k+v_i^k)} \tag{14}$$

Based on χ^2 Statistic, we define $ITS(f_i)$ as

$$ITS(f_i) = \max\{CHI(f_i, D^1), CHI(f_i, D^2)\} \tag{15}$$

3.3.6. Definition based on Weighted Log Likelihood Ratio

Weighted Log Likelihood Ratio (*WLLR*) is a metric that has proved effective at selecting good features for text classification (Ng, Dasgupta, & Niaz Arifin, 2006; Nigam, McCallum, Thrun, & Mitchell, 2000). The *WLLR* between term f_i and document set D^k is defined as

$$WLLR(f_i, D^k) = P(f_i|D^k)\log\frac{P(f_i|D^k)}{P(f_i|\overline{D^k})} \tag{16}$$

$WLLR(f_i, D^k)$ can be estimated as

$$WLLR(f_i, D^k) \approx \frac{x_i^k}{N_k}\log\frac{x_i^k(N_1+N_2-N_k)}{y_i^k N_k} \tag{17}$$

Based on Weighted Log Likelihood Ratio, we define $ITS(f_i)$ as

$$ITS(f_i) = \max\{WLLR(f_i, D^1), WLLR(f_i, D^2)\} \tag{18}$$

3.3.7. Definition based on Weighed Frequency and Odds

Weighed Frequency and Odds (*WFO*) is a new feature selection method proposed in 2009 (Li et al., 2009). The experimental results in Li et al. (2009) indicate that *WFO* is robust across different tasks and numbers of selected features on data sets from both topic-based and sentiment classification tasks. Given term f_i and document set D^k, the *WFO* between them is defined as

$$WFO(f_i, D^k) = P(f_i|D^k)^{\lambda}\log\left(\frac{P(f_i|D^k)}{P(f_i|\overline{D^k})}\right)^{1-\lambda} \tag{19}$$

Note that, the definition of *WFO* is a little different from the original definition proposed in Li et al. (2009). We find Formula (19) is more effective than the original one.

$WFO(f_i, D^k)$ can be estimated as

$$WFO(f_i, D^k) \approx \left(\frac{x_i^k}{N_k}\right)^{\lambda}\log\left(\frac{x_i^k(N_1+N_2-N_k)}{y_i^k N_k}\right)^{1-\lambda} \tag{20}$$

Based on Weighed Frequency and Odds, we define $ITS(f_i)$ as

Table 2
Symbols for related statistical information.

Symbol	Meaning
tf_{ij}	The number of times that term f_i occurs in document d_j.
x_i^k	The number of documents that both contain term f_i and belong to document set D^k.
y_i^k	The number of documents that contain term f_i but do not belong to document set D^k.
u_i^k	The number of documents that do not contain term f_i but belong to document set D^k.
v_i^k	The number of documents that neither contain term f_i nor belong to document set D^k.
N_k	The number of documents that belong to document set D^k.

$$ITS(f_i) = \max\{WFO(f_i, D^1), WFO(f_i, D^2)\} \tag{21}$$

4. Experimental evaluation

The experiments are carried out on three benchmark data sets, all of which have been widely used for sentiment analysis. The classifier used in this paper is based on support vector machine. We use accuracy, a measure extensively used in sentiment analysis, as the performance criterion in this paper.

4.1. Data set

The Cornell movie review data set,[1] first used by Pang et al. (2002), has been used widely as the benchmark dataset by more than one hundred papers as of April 2012, according to the website. The polarity dataset contains 1000 positive and 1000 negative processed reviews of movies, extracted from the Internet Movie Database.

The second data set [2] is a multi-domain sentiment dataset of product reviews, taken from many product domains of Amazon.com. Some domains have hundreds of thousands of review but others much less. In our experiment we used the reviews from four different domains, each of approximately 4000 reviews. The whole set contains in total about 16,000 reviews.

We also use the Stanford large movie review data set[3] provided by Maas et al. (2011). It comprises far more reviews than the previous ones. The data set contains 50,000 highly polar movie reviews, divided equally between positive and negative. The original split is to further divide positive and negative reviews into two parts, one for training and the other for testing.

4.2. Implementation

There is no stemming applied or stop words removed for both movie reviews data sets, but the product review set comes processed to have each term (word or phrase) counted. The implementation of support vector machine, *libSVM*,[4] is utilized. A linear kernel is chosen and all parameters are set to their default value, except for gamma (set to the reverse of the term set's size in this paper). SVM is the most prominent method and is widely used in the field of sentiment analysis because its performance greatly exceeds other methods. Therefore, we only use SVM as the classifier for sentiment analysis.

The results are based on the standard ten-fold cross validation for the Cornell movie review data set and the Amazon product review data set. Due to the large number of terms (features), the Stanford movie review data set is reported on its original split.

Three versions of *ITD*, which are Formulas (2)–(4) respectively, and eight versions of *ITS*, including two kinds of *WFO* with different parameters, are implemented in the experiment, along with BM25 (Paltoglou & Thelwall, 2010). Note that, there are some kinds of BM25. In this paper, we choose the best one, $o\Delta(k)n$, according to the report provided by Paltoglou and Thelwall (2010). All of above have been tested on the Cornell movie review set and the Amazon product review set. But due to the time and space constrains, only approaches with high accuracy on the Cornell movie review data set and the Amazon product review data set are chosen to validate on the large Stanford movie review data set.

It should be noticed that those results reported by Paltoglou and Thelwall (2010) cannot be directly comparable to ours because those results are based on leave-one out cross validation while our results are based on the standard ten-fold cross validation. As we know, the accuracy based on leave-one out cross validation usually much higher than the one based on the standard ten-fold cross validation. Therefore, the best setting of BM25, $o\Delta(k)n$, produces a lower accuracy in this paper than in Paltoglou and Thelwall (2010).

4.3. Results

The complete results on the Cornell movie review data set and the Amazon product review data set are shown in Figs. 1 and 2 respectively, and results of preferable approaches for the Stanford movie review set are displayed in Fig. 3. Note that, $ITD(k)$ stands for the definition of *ITD* by the Formula (k) in Figs. 1–3. For example, $ITD(2)$ means that we employ the Formula (2) as the definition of *ITD*. In addition, each pillar in Figs. 1–3 means the result that use the corresponding *ITD* and *ITS* to generate term weights.

4.3.1. Performance comparison on Cornell movie review data set

On the Cornell movie review data set, 6 of our approaches result well, with the accuracy over 87%. Surprisingly, we have discovered that *IG* produces results drastically low in accuracy. The reason is waiting for further study. As a simple and traditional baseline, we have also implemented *DF* to obtain the basic results (79.0%, 78.2% and 79.2% respectively for three sorts of *ITD*) for reference. With labelling information sophisticatedly utilized, our approaches outperform the simple baseline in apparent advantages. Among these approaches $ITD(4)*OR$ performs the best, achieving an accuracy of 88.5%.

When it comes to the influence of *ITD*, the results indicate that most of the high-performance approaches, such as $ITD(4)*MI$ (87.8%), $ITD(4)*OR$ (88.5%), $ITD(4)*WFO(0.1)$ (88.0%) and $ITD(4)*WFO(0.01)$ (87.7%), introduce the normalization of raw term frequency. Yet simply utilizing the binary weight we obtain results almost as great, e.g. $ITD(2)*DF$ (79.0%) vs. $ITD(4)*DF$ (79.2%), $ITD(2)*MI$(87.5%) vs. $ITD(4)*MI$(87.8%) and $ITD(2)*WFO(\lambda = 0.1)$ (87.7%) vs. $ITD(4)*WFO(\lambda = 0.1)$ (88.0%), except for some rare case such as $ITD(2)*OR$ (85.4%) vs. $ITD(4)*OR$ (88.5%). This reconfirms the conclusion drawn in (Paltoglou and Thelwall, 2010). Applying raw term frequency will produce visibly different results, yet the difference could be either increase or decrease in accuracy, e.g. $ITD(3)*CHI$ (85.0%) vs. $ITD(4)*CHI$ (83.7%), $ITD(3)*MI$ (85.4%) vs. $ITD(4)*MI$ (87.8%) and $ITD(3)*IG$ (60.3%) vs. $ITD(4)*IG$ (68.0%). There are also some *ITS* settings on this data set that will cause

[1] http://www.cs.cornell.edu/people/pabo/movie-review-data/
[2] http://www.cs.jhu.edu/~mdredze/datasets/sentiment/
[3] http://ai.stanford.edu/~amaas/data/sentiment/
[4] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

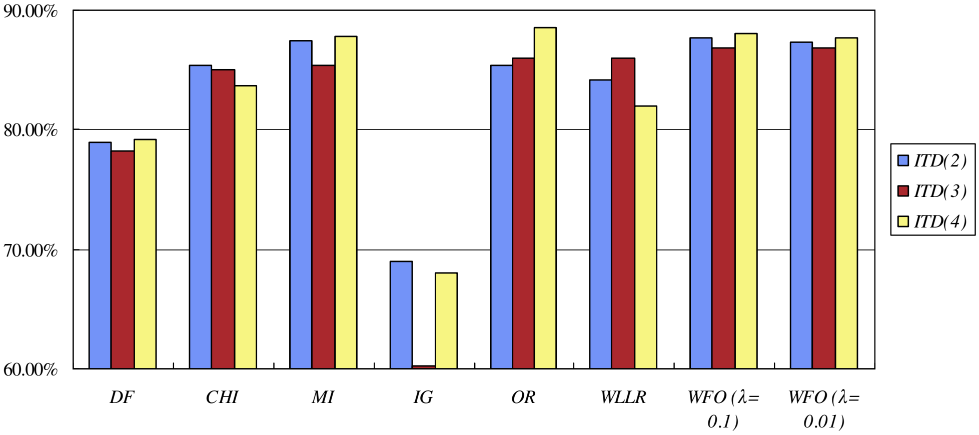

Fig. 1. Reported accuracy on the Cornell movie review data set.

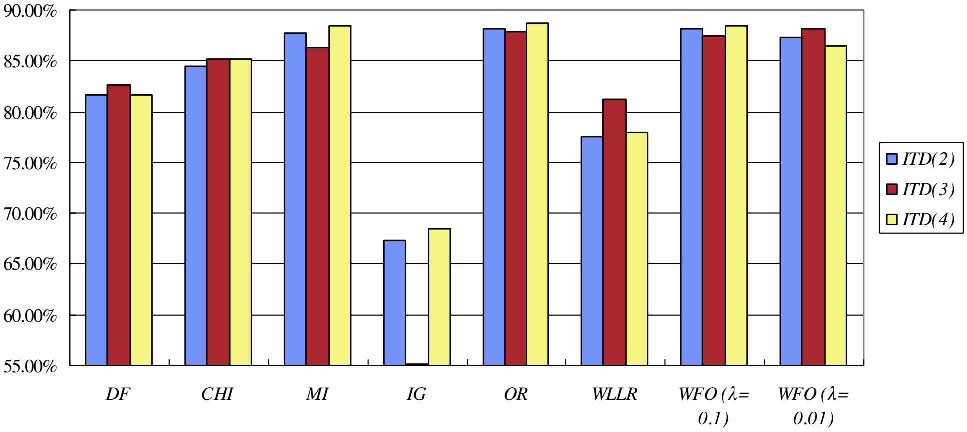

Fig. 2. Reported accuracy on the Amazon product review data set.

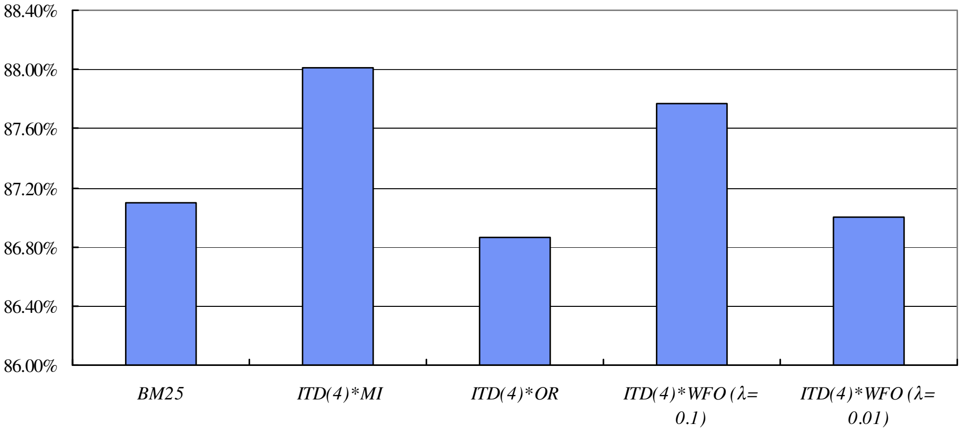

Fig. 3. Reported accuracy on the Stanford movie review data set.

apparent difference of accuracy for the three *ITD* methods, such as *OR* or *WLLR*.

Compared with the change of *ITD*, incorporating different versions of *ITS* would lead to more significant difference on results. At most of the time, the accuracy mainly depends on the choice of *ITS* rather than that of *ITD*. For example, using *CHI* with three different *ITS*s (85.4%, 85.0% and 83.7% respectively) all performs better than *DF* (81.7%, 82.6% and 81.6% respectively), and using *OR* (85.4%, 86.0% and 88.5% respectively) beats the former two. Leaving out the extreme case produced by *IG*, the largest difference of accuracy when using the same *ITS* is 4% (*ITD*(3)$*$*WLLR* vs. *ITD*(4)$*$*WLLR*), while that when using the same *ITD* is 9.3% (*ITD*(4)$*$*DF* vs. *ITD*(4)$*$*OR*). The results indicate that the choice of *ITS* has larger influence on accuracy than that of *ITD*.

The accuracy of BM25 is slightly higher on this data set, reaching 88.7%, 0.2% higher than our best approach (*ITD*(4)$*$*OR*). But with only 200 reviews as the test set, such little disparity does not make much difference and no convincing conclusion could be drawn.

4.3.2. Performance comparison on Amazon product review data set

On the Amazon product review data set, our approaches provide larger advantages over the BM25, and the conclusions are in accordance to those on the previous data set. Nine of those approaches have an accuracy of higher than 87%, and the best one (*ITD*(4)∗*OR*) reaches up to 88.7%, definitely yielding better performance than BM25 (87.7%). Other than *ITD*(4)∗*OR*, five approaches also have an accuracy over 88.0%. This indicates that our approaches outperform the BM25 on sentiment analysis of larger data sets.

In terms of Amazon product review data set, the influence of choosing different *ITS*s is still consistent with the observation on the Cornell movie review data set. When we choose *MI*, *OR* or *WFO*($\lambda = 0.1$) as the *ITS*, *ITD*(4)∗*MI* (88.5%), *ITD*(4)∗*OR* (88.7%), *ITD*(4)∗*WFO*($\lambda = 0.1$) (88.4%) are still better than the others. But on this set it's better for *WFO*($\lambda = 0.01$) to use binary or raw term frequency(87.3% for binary, 88.1% for raw term frequency, and 86.5% for normalization). Overall, the normalization of raw term frequency still performs better.

The results for the change of *ITS* on this data set also agree with those on the Cornell movie data set. Some methods perform well with any kind of *ITD*, such as *OR* (88.1%, 87.9% and 88.7% respectively) and *WFO*($\lambda = 0.1$) (88.2%, 87.4% and 88.4% respectively), re-confirming the conclusion that *ITS* cause more significant difference. It is also worth mentioning that the two kinds of *WFO* have shown good stability: these two methods produce little change in accuracy throughout the above experiments.

4.3.3. Performance comparison on Stanford movie review data set

Fig. 3 shows the results of top four approaches and the BM25 Stanford movie review data set. In consideration of the performance on the first two data sets, we decide to apply four favourable approaches on the Stanford movie review data set. Stanford movie review data set is a rather huge data set with 25,000 reviews for training set and another 25,000 for test set, dozens of times larger than the other two data sets, so we tend to consider the results on this data set of greater significance. On this set, *ITD*(4)∗*MI* yields the best result, reaching a high accuracy of 88.008%. *ITD*(4)∗*WFO* ($\lambda = 0.1$) performs also at a high accuracy of 87.771%. *ITD*(4)∗*OR*, though of the best performance on the other two sets, does not get as well a result (86.858%). The large increase of size of the training set might have caused the difference. Yet our best results on this data set are still better than the BM25 (87.096%), saying that our method has a consistent advantage over the BM25 on sufficiently large data set.

4.3.4. Summarization and discussion

To sum up, the results on Cornell movie review data set may not seem to be adequately accurate. With only 200 reviews as the test set, any accident on a single review can cause the error of 1%. While on the Amazon product review data set and the Stanford movie review data set, with 1600 and 25,000 reviews as the test set respectively, such accident errors could be reduced effectively.

Table 3 shows the comparison of our best results and the results of the BM25 on all three data sets. Clearly, our approach definitely outperforms the BM25 on two of three data sets while the difference is indistinctive on the small Cornell movie review data set.

Note that, the accuracy of the method based on optimization learning (Maas et al., 2011) is 87.44% on the Stanford movie review data set, which is higher than the BM25 (87.10%) but lower than our best result (88.00%). As indicated by Maas et al. (2011), combining supervised methods with unsupervised methods can produce higher accuracy. How to improve the accuracy by combining our method with unsupervised methods is our future work.

Table 3
Comparison of best results.

	BM25(%)	Our best
Cornell movie review data set	**88.70**	88.50% (*ITD*(4)∗*OR*)
Amazon product review data set	87.70	**88.70%** (*ITD*(4)∗*OR*)
Stanford movie review data set	87.10	**88.00%** (*ITD*(4)∗*MI*)

As for the time used to compute *ITS*(f_i), it is not much because the computational complexity is linear. Assume there are *N* features. For each feature, we should compute its statistic information with D^1 (the set of positive documents) and D^2 (the set of negative documents). Therefore, the overall computational complexity is O(N), which is linear.

5. Conclusion and perspectives

In this paper, we propose a term weighting scheme based on supervised learning by statistical functions. In this scheme, term weighting consists of two parts: *ITD* and *ITS*. The *ITD* is based on term frequency while seven statistical functions are employed to learn the *ITS* of each term from training documents with category labels. The proposed term weighting scheme were evaluated on three benchmark document sets which are widely used in sentiment analysis. The experimental results show that our approach outperforms the state-of-the-art unsupervised approach and produce the best accuracy. Our main contribution is that we reveal that term weighting schemes based on supervised learning are more effective than those schemes originated from information retrieval without considering the correlation between terms and sentiment polarity. Our work introduces a new direction for sentiment analysis and thus will enhance its development.

In this paper, we use seven statistical functions to compute the correlation between terms and sentiment polarity. There are much more functions (Geng & Hamilton, 2006) available in statistics and data mining which can be used. In addition, we plan to investigate the combination of statistical functions to produce more appropriate term weights. As we know, different statistical functions focus on different aspects. This suggests that combining them would be more effective than single one. Finally, we will also explore the study that aims to improve the accuracy by combining our method with unsupervised methods.

Acknowledgments

This work is partially supported by Project 61170091, supported by National Natural Science Foundation of China. We would also like to thank the anonymous reviewers for their helpful comments.

References

Armstrong, T. G., Moffat, A., Webber, W., & Zobel, J. (2009). Improvements that don't add up: ad-hoc retrieval results since 1998. In *Proceedings of the 18th ACM conference on information and knowledge management (CIKM 2009)* (pp. 601–610) Hong Kong, China.

Baeza-Yates, R., & Ribeiro-Neto, B. (2011). *Modern information retrieval*. Wokingham, UK: Addison-Wesley.

Church, K. W., & Hanks, P. (1989). Word association norms, mutual information and lexicography. In *Proceeding of ACL 27* (pp. 76–83) Vancouver, Canada.

Das, S., & Chen, M. (2001). Yahoo! for Amazon: Extracting market sentiment from stock message boards. In *Proceedings of the Asia Pacific finance association annual conference (APFA 2001)*.

Debole, F., & Sebastiani, F. (2003). Supervised term weighting for automated text categorization. In *Proceedings of the 2003 ACM symposium on applied computing (SAC 2003)* (pp. 784–788) New York, NY, USA.

Deng, Z., Tang, S., Yang, D., Zhang, M., Li, L., & Xie, K. (2004). A comparative study on feature weight in text categorization. In *Proceedings of the sixth Asia-Pacific web conference (APWeb 2004)* (pp. 588–597) Hangzhou, China.

Geng, L., & Hamilton, H. J. (2006). Interestingness measures for data mining: A survey. *ACM Computing Surveys, 38*(3).

Manning, C. D., Raghavan, P., & Schutze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.

Jones, K. S. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation, 28*(1), 11–21.

Lan, M., Tan, C. L., Su, J., & Lu, Y. (2009). Supervised and traditional term weighting methods for automatic text categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 31*(4), 721–735.

Li, S., Xia, R., Zong, C., Huang, C. R. (2009). A Framework of Feature Selection Methods for Text Categorization. In *Proceeding of ACL/AFNLP 2009* (pp. 692–700).

Maas, A. L., Daly, R. E., Pham, P. T., Huang, Dan., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics (ACL 2011)* (pp. 142–150) Portland, Oregon, USA.

Mladenic, D., Grobelnik, M. (1998). Feature selection for classification based on text hierarchy. In *Proceeding of conference on automated learning and discovery (CONALD 1998)*.

Martineau, J., & Finin, T. (2009). Delta tfidf: An improved feature space for sentiment analysis. In *Proceedings of the third AAAI international conference on weblogs and social media* (pp. 258–261) San Jose, California, USA.

Ng, V., Dasgupta, S., & Niaz Arifin, S. M. (2006). Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In *Proceedings of the COLING/ACL main conference poster sessions*.

Nigam, K., McCallum, A., Thrun, S., & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine Learning, 39*(2/3), 103–134.

Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the ACL 2004* (pp. 271–278).

Pang, B., & Lee, L. (2008). *Opinion mining and sentiment analysis*. Now Publishers Inc.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 conference on empirical methods in natural language processing (EMNLP 2002)*.

Paltoglou, G., Thelwall, M. (2010). A study of information retrieval weighting schemes for sentiment analysis. In *Proceeding of ACL 2010* (pp. 1386–1395).

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*(1), 81–106.

Ren, F., & Sohrab, M. G. (2013). Class-indexing-based term weighting for automatic text classification. *Information Sciences, 236*, 109–125.

Robertson, S. E., & Jones, S. (1976). Relevance weighting of search terms. *Journal of American Society for Information Science, 27*, 129–146.

Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M., & Gatford M. (1994). Okapi at TREC-3. In *TREC 1994* (pp. 109–127).

Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M., & Gatford, M. (1996). Okapi at TREC-5. In *The fifth text retrieval conference (TREC-5)*.

Robertson, S. E., Zaragoza, H., & Taylor, M. J. (2004). Simple bm25 extension to multiple weighted fields. In *Proceedings of the thirteenth ACM international conference on information and knowledge management (CIKM 2004)* (pp. 42–49). New York, NY, USA: ACM.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management, 24*(5), 513–523.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys, 34*(1), 1–47.

Soucy, P. & Mineau, G. W. (2005). Beyond tfidf weighting for text categorization in the vector space model. In *Proceedings the nineteenth international joint conference on artificial intelligence (IJCAI 2005)* (pp. 1130-1135) Edinburgh, Scotland, UK.

Tong, R. M. (2001). An operational system for detecting and tracking opinions in on-line discussion. In *Proceedings of the workshop on operational text classification (OTC)*.

Turney, P. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the association for computational linguistics (ACL 2002)* (pp. 417–424).

van Rijsbergen, C. J., Harper, D. J., & Porter, M. F. (1981). The selection of good search terms. *Information Processing & Management, 17*, 77–91.

Yang Y. and Pedersen J. (1997). A comparative study on feature selection in text categorization. In *Proceedings of international conference of machine learning (ICML 1997)* (pp. 412–420).