Decision Support Systems 57 (2014) 77–93

Contents lists available at ScienceDirect

Decision Support Systems

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

Sentiment classification: The contribution of ensemble learning

Gang Wang [a,b,c,*], Jianshan Sun [c,d], Jian Ma [c], Kaiquan Xu [e], Jibao Gu [d]

[a] School of Management, Hefei University of Technology, Hefei, Anhui 230009, PR China
[b] Key Laboratory of Process Optimization and Intelligent Decision-making, Ministry of Education, Hefei, Anhui, PR China
[c] Department of Information Systems, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong
[d] School of Management, University of Science and Technology of China, Hefei, Anhui, PR China
[e] Department of Electronic Commerce, School of Business, Nanjing University, Nanjing, Jiangsu 210093, PR China

ARTICLE INFO

Article history:
Received 27 August 2012
Received in revised form 1 August 2013
Accepted 5 August 2013
Available online 15 August 2013

Keywords:
Sentiment classification
Ensemble learning
Bagging
Boosting
Random Subspace

ABSTRACT

With the rapid development of information technologies, user-generated contents can be conveniently posted online. While individuals, businesses, and governments are interested in evaluating the sentiments behind this content, there are no consistent conclusions on which sentiment classification technologies are best. Recent studies suggest that ensemble learning methods may have potential applicability in sentiment classification. In this study, we conduct a comparative assessment of the performance of three popular ensemble methods (Bagging, Boosting, and Random Subspace) based on five base learners (Naive Bayes, Maximum Entropy, Decision Tree, K Nearest Neighbor, and Support Vector Machine) for sentiment classification. Moreover, ten public sentiment analysis datasets were investigated to verify the effectiveness of ensemble learning for sentiment analysis. Based on a total of 1200 comparative group experiments, empirical results reveal that ensemble methods substantially improve the performance of individual base learners for sentiment classification. Among the three ensemble methods, Random Subspace has the better comparative results, although it was seldom discussed in the literature. These results illustrate that ensemble learning methods can be used as a viable method for sentiment classification.

1. Introduction

With the rapid development of information technologies, user-generated contents can be easily posted online [1]. The sheer volume and exponential growth of this information provide potential value to governments, businesses, and users themselves. For instance, governments can evaluate online citizen-generated texts to assess public sentiment for making policies. Furthermore, many customer-generated reviews of products and services have become valuable sources for market analysis; these reviews are used to set business strategy of E-commerce websites, such as Amazon.com and Epinion.com [50]. Online users can also benefit from reading others' opinions through recommender systems.

There is an inherent property called sentiment involved in the vast majority of online-generated content. Sentiment is an opinion or feeling you have about something [12]. In this study of sentiment classification, we focus on attempts to identify the sentiment polarity of a given text, which is traditionally classified as either positive or negative. Analyzing and predicting the polarity of the sentiment plays an important role in understanding social phenomena and general society trends [6].

Accordingly, sentiment classification has become a popular research topic [1,4,6]. The sentiment classification problem was initially tackled granularly at the levels of document, sentence, clause, phrase, and word, depending on the specific objectives of applications. Heuristic-based methods and machine learning approaches were frequently employed in previous research. Heuristic-based methods were primarily used in conjunction with linguistic characters and semantic features. For example, Turney [38] used mutual information with predefined sentiment words to score other phrase tags, therefore identifying the sentiment of documents. In parallel, many studies focused on using machine learning algorithms to classify sentiment. For instance, Support Vector Machines (SVM) and Naive Bayes (NB) are commonly used to identity sentiment, due to their predictive power. Pang et al. [29] conducted an empirical study in sentiment classification, concluding that SVM outperformed other classifiers such as NB. In recent years, there has been a growing interest in using ensemble learning techniques, which combine the outputs of several base classification techniques to form an integrated output, to enhance classification accuracy [43,48]. However, compared with other research domains, related work about ensemble methods contributing to sentiment classification are still limited and more extensive experimental work is needed in this area.

To fill this research gap, this paper makes a comparative study of the effectiveness of ensemble learning for sentiment classification and demonstrates that three popular ensemble methods (Bagging [5], Boosting [33] and Random Subspace [19]) can be useful. Research

* Corresponding author at: Department of Information Systems, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong. Tel.: +852 9799 0955; fax: +852 2788 8694.
E-mail address: wgedison@gmail.com (G. Wang).

0167-9236/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.dss.2013.08.002

Table 1
Selected previous studies in ensemble learning for sentiment analysis.

Study	Year	Feature set	Base learner	Ensemble methods	Dataset
Wilson et al. [45]	2006	N-gram, syntactic features	DT	Boosting	MPQA dataset
Tsutsumi et al. [37]	2007	N-gram	SVM, ME, Scoring	Stacking	Movie review dataset
Abbasi et al. [2]	2008	N-gram, lexicon	SVM	SVRCE	Two web forum datasets
Lu & Tsou [26]	2010	N-gram, lexicon	NB, ME, SVM, Scoring	Stacking	NTCIR opinion dataset
Whitehead & Yaeger et al. [43]	2010	N-gram	SVM	Bagging, Boosting and Random Subspace	Five product review datasets
Xia et al. [48]	2011	POS and word-relation based features	NB, ME, SVM	Stacking	Five product review datasets
Su et al. [34]	2012	N-gram	NB, CB, KNN, ME, SVM	Stacking	Three product review datasets
Li et al. [24]	2012	N-gram, lexicon	SVM, KNN, Scoring	Stacking	Chinese review dataset

in many areas has shown the advantages of ensemble methods both theoretically and empirically [30,51]. In ensemble methods, learners composing an ensemble are usually called base learners. In Bagging, the base learners are constructed using random independent bootstrap replicates from a training dataset, and the final result is calculated by a simple majority vote [5,51]. In Boosting, the base learners are constructed on weighted versions of the training set, which are dependent on previous base learners' results and the final result is calculated by a simple vote or a weighted majority vote [33,51]. In Random Subspace, the base learners are constructed in random subspaces of the feature space [19,51].

We employed ten public sentiment analysis datasets to verify the effectiveness of these three ensemble methods when using five base learners (NB, Maximum Entropy (ME), Decision Tree (DT), K Nearest Neighbors (KNN), and SVM). Based on a total of 1200 comparative group experiments, empirical results show that ensemble learning methods achieve better performances than base learners. Among the three ensemble methods, Random Subspace has the better comparative results except with NB as base learner, although it was seldom discussed in the literature. In addition, RS-SVM had the highest average accuracy in 6 datasets and similar results with other methods in the other 4 datasets. These results illustrate that ensemble learning methods can be used as a viable method for identifying sentiment polarities.

The main contribution of this paper is to verify the effectiveness of using ensemble learning for sentiment classification. The remainder of the paper is organized as follows. In Section 2, we survey the related work about sentiment classification. The details of three different types of ensemble methods are introduced in Section 3. Section 4 presents the design and methodology used in the experiments, while the results are analyzed in Section 5. Section 6 discusses conclusions and future research directions.

2. Literature review

Since the late 1990s, sentiment classification has been a hot research topic in the areas of data mining, information retrieval, and natural language processing [4,28]. Many researchers have investigated sentiment classification from different perspectives. Due to the linguistic characteristics involved, sentiment analysis is done at different levels of text units. A word, phrase, clause, sentence, or document may become the text unit in analysis [28]. In order to capture the sentiment of individual words or phrases, a measure of the strength of sentiment polarity is often defined to quantify how strongly a word or phrase is judged to be positive or negative [9,22,35,38]. Furthermore, Thet et al. [36] computed the sentiment of a clause from individual word sentiment scores, considering the grammatical dependency structure of the clause. Other studies used sentence-level attempts to classify the positive or negative sentiments for each sentence [49,50]. The greatest amount of work has been done on document level polarity categorization [1,4,11,29,43,48]. This is also the focus level of our study. The techniques for sentiment classification in prior research can be classified into heuristic-based methods and machine learning methods.

2.1. Heuristic-based methods for sentiment classification

By means of predefined lexicons and calculation rules, heuristic-based methods generally classify text sentiments based on the total number of derived positive or negative sentiment features [28]. For example, Hatzivassiloglou and McKeown [18] considered that adjectives are more predictive of sentiment classification and predicted the sentiment of adjectives by inspecting them in conjunction with “and,” “or,” “but,” “either/or,” and “neither/nor.” However this approach may overestimate the importance of adjectives and underestimate some

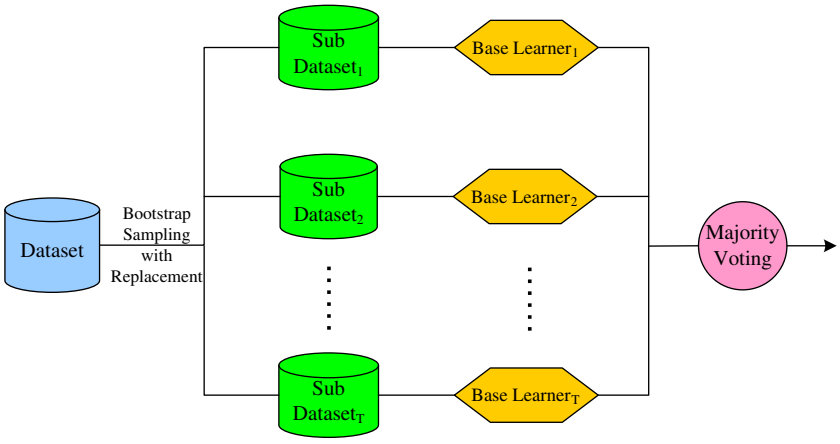

Fig. 1. The Bagging process.

Input: Data set $D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_m,y_m)\}$;

Base learning algorithm L;

Number of learning rounds T.

Process:

For $t=1,2,\cdots,T$:

$D_t = Bootstrap(D)$; % Generate a bootstrap sample from D

$h_t = L(D_t)$ % Train a base learner h_t from the bootstrap sample

end.

Output: $H(x)=\arg\max_{y\in Y}\sum_{t=1}^{T}1(y=h_t(x))$ % the value of $1(\alpha)$ is 1 if α is true % and 0 otherwise

Fig. 2. The Bagging algorithm.

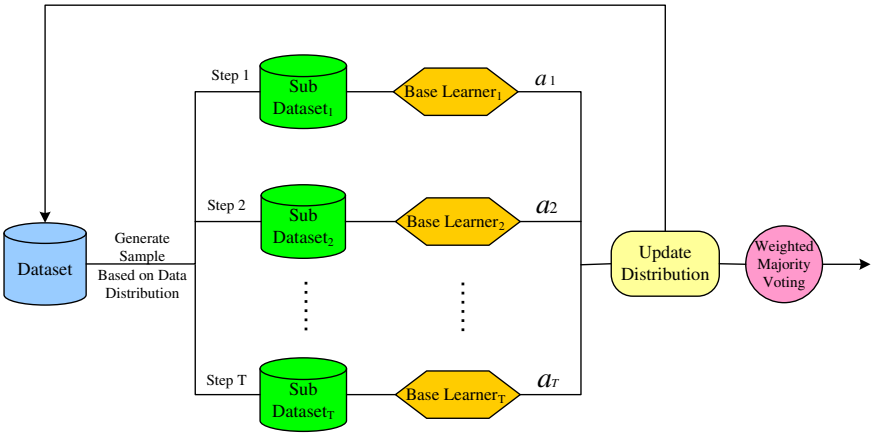

Fig. 3. The AdaBoost process.

Input: Data set $D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_m,y_m)\}$;

Base learning algorithm L;

Number of learning rounds T.

Process:

$D_1(i)=1/m.$ % Initialize the weight distribution

For $t=1,2,\cdots,T$:

$h_t = L(D,D_t)$; % Train a base learner h_t from D using distribution D_t

$\varepsilon_t = \Pr_{i\sim D_t}[h_t(x_i)\neq y_i)]$; % Measure the error of h_t

$\alpha_t = \frac{1}{2}\ln\frac{1-\varepsilon_t}{\varepsilon_t}$; % Determine the weight of h_t

$D_{t+1}(i)=\frac{D_t(i)}{Z_t}\times\begin{cases}\exp(-\alpha_t) & if\ h_t(x_i)=y_i\\ \exp(\alpha_t) & f\ h_t(x_i)\neq y_i\end{cases}$ % Update the distribution, where Z_t is a

$=\frac{D_t(i)\exp(-\alpha_t y_i h_t(x_i))}{Z_t}$ %normalization factor which enables D_{t+1} to be a distribution

end.

Output: $H(x)=\mathrm{sign}(f(x))=\mathrm{sign}\sum_{t=1}^{T}\alpha_t h_t(x)$

Fig. 4. The AdaBoost algorithm.

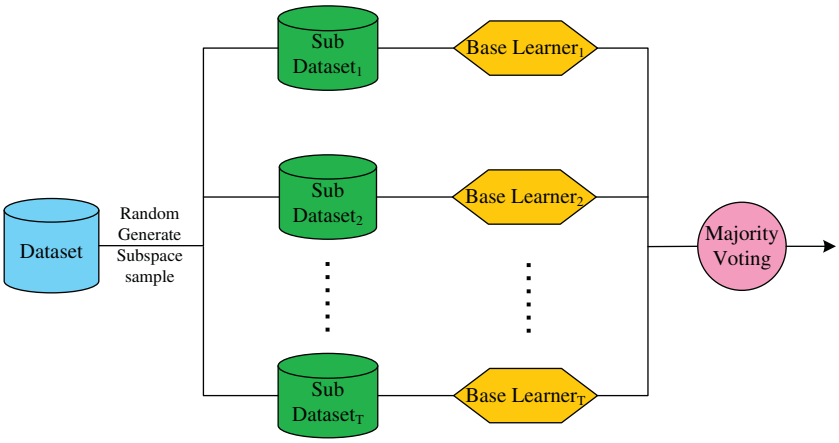

Fig. 5. The Random Subspace process.

Input: Data set $D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_m,y_m)\}$;

Base classifier algorithm L;

Number of random subspace rate k;

Number of learning rounds T.

Process:

For $t=1,2,\cdots,T$:

$D_t = RS(D,k)$; % Random generate a subspace sample from D

$h_t = L(D_t)$; % Train a base classifier h_t from the subspace sample

end.

Output: $H(x)=\arg\max_{y\in Y}\sum_{t=1}^{T}1(y=h_t(x))$; % the value of $1(\alpha)$ is 1 if α is true

% and 0 otherwise

Fig. 6. The Random Subspace algorithm.

Table 2
Description of sentiment analysis datasets.

Dataset	Description	# of features (Unigram)	# of features (Bigram)	# of instances	Source
Camera	Digital camera reviews from Amazon.com. These reviews were taken from cameras that had a large number of ratings. This dataset and the laptop review set both fall under the broader domain of consumer electronics.	1352	1704	498 (250:248)	[42]
Camp	Summer camp reviews from CampRatingz.com. A significant number of these reviews were written by the young people who attended the summer camps.	2045	1735	804 (402:402)	[42]
Doctor	Reviews of physicians from RateMDs.com. This dataset and the lawyer review set could both be considered part of the larger "ratings of people" domain.	1578	1679	1478 (739:739)	[42]
Drug	Reviews of pharmaceutical drugs from DrugRatingz.com.	1438	1662	802 (401:401)	[42]
Laptop	Laptop reviews from Amazon.com. Various laptops are reviewed from different manufacturers.	2010	3136	176 (88:88)	[42]
Lawyer	Reviews of lawyers from LawyerRatingz.com.	2474	7734	220 (110:110)	[42]
Movie	Movie reviews of various movies from (Pang and Lee [27,29]).	1165	1232	2000 (1000:1000)	[27,29]
Music	Musical CD reviews from Amazon.com. The albums being reviewed were recently released popular music from a variety of musical genres.	1398	1705	582 (291:291)	[42]
Radio	Reviews of radio shows from RadioRatingz.com. This dataset and the TV dataset had the shortest reviews on average.	1923	3054	1004 (502:502)	[42]
TV	Television show reviews from TVRatingz.com. These reviews were typically very short and not very detailed.	2834	9195	470 (235:235)	[42]

Table 3
Confusion matrix for sentiment classification.

		Actual condition	
		Positive sentiment	Negative sentiment
Test result	Positive sentiment	True positive (TP)	False positive (FP)
	Negative sentiment	False negative (FN)	True negative (TN)

predictive words of other parts-of-speech. Along this line, Turney [38] determined the semantic orientation of a phrase using its point-wise mutual information with predefined sentiment words, such as "excellent" and "poor." Therefore, sentiment classification can be achieved by aggregating the overall sentiment information of phrases. The adjective–verb–adverb (AVA) combinations were thoroughly analyzed in [35] for sentiment polarity predication. There are also other fruitful studies following this line [20,49].

The heuristic-based methods are similar to using knowledge engineering methods to classify text sentiment. One of the problems of this approach is it relies heavily on pre-defined lexicons and rules that are difficult to update and use in multiple domains [28]. In this research, we focus on machine learning methods for sentiment classification.

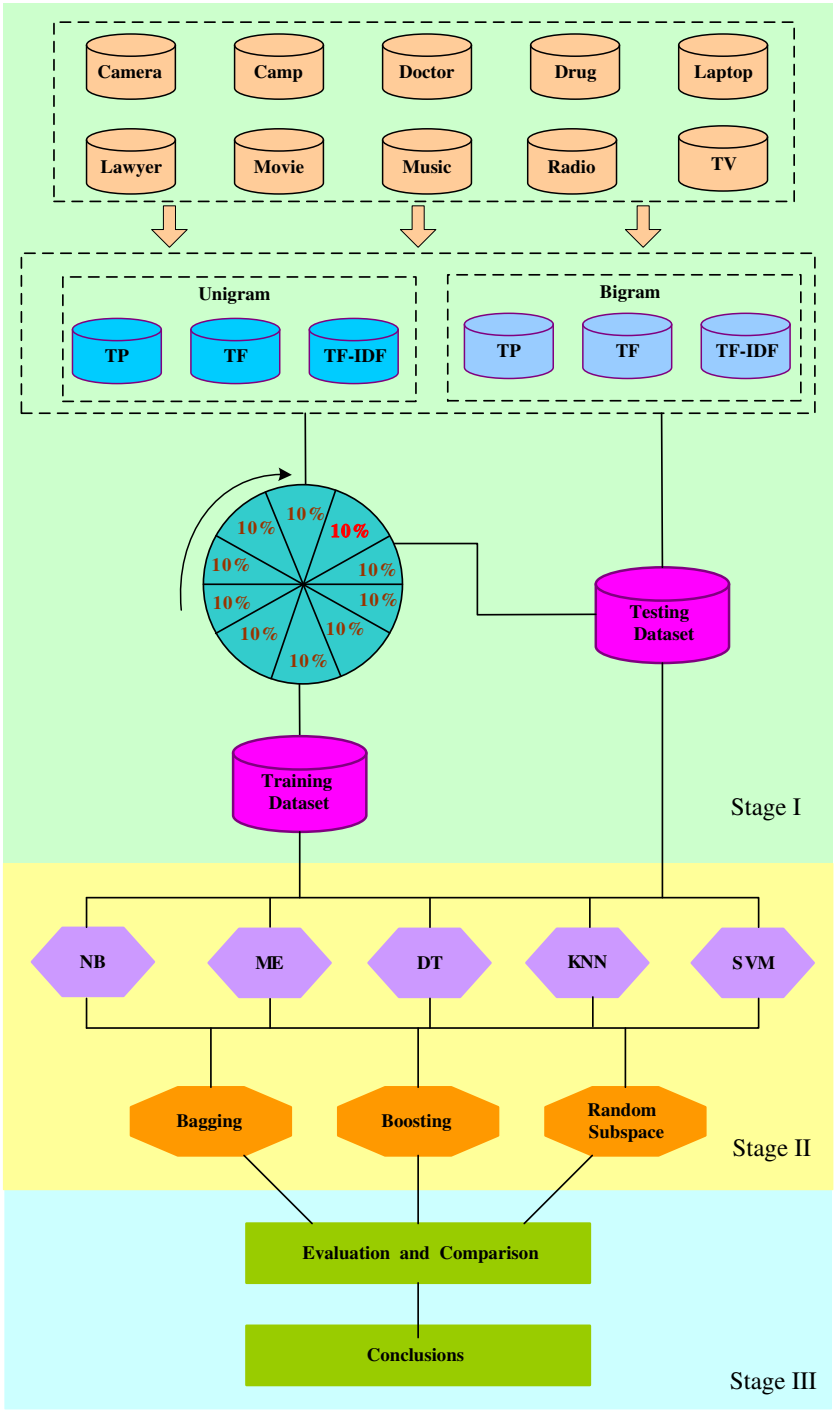

Fig. 7. Experimental procedure.

2.2. Machine learning methods for sentiment classification

Machine learning approaches for sentiment classification have been extensively studied, due to their predominant classification performance [2,28]. By constructing predictive models from labeled training datasets, these methods can model more features and adapt to changing inputs more robustly, than heuristic-based methods [28]. Machine learning methods for sentiment classification typically consist of two steps: (1) extraction of features from training data and their conversion to feature vectors and (2) training of the classifier on the feature vectors and application of the classifier to unseen instances [23]. Accordingly, both feature construction and learning method are crucial for accurate sentiment classification.

2.2.1. Feature construction

Converting a piece of text into a feature vector is an important part of machine learning methods for sentiment classification. From the machine learning perspective, it is useful for the features to include only relevant information and also to be independent of each other [15].

The feature representation method dominating the sentiment classification literature is known as the bag-of-words (BOW) framework [28]. In this framework, the text is considered as a bag of words and represented by a vector containing all the words appearing in the corpus. Besides BOW features, many other types of features are proposed, such as part-of-speech (POS), syntax, negation, and topic-oriented features [28]. However, these features rely heavily on linguistic resources. Popular among them are lexicons, such as SentiWordNet, General Inquire, and POS tagger [32]. In addition, the construction of these features is time consuming and tedious. Just like heuristic-based methods for sentiment classification, they need specific domain knowledge and are difficult to update [28]. Since the aim of this study is to verify the effectiveness of using ensemble learning for sentiment classification, we use N-gram features to present the text in this research.

Another important problem is feature selection, a problem that has been tackled by many researchers in different ways [15,28]. However, recent studies have drawn no consistent conclusions that one technique is superior. In addition, some studies have found that feature selection was not always effective in enhancing sentiment classification accuracy [23]. Since discussions of effective feature selection are beyond the scope of this research, in this study we follow [29] and use Unigram and Bigram features.

In the area of text classification, term frequency–inverse document frequency (TF–IDF) weighting has been successful. In sentiment classification, Pang et al. pointed out that a topic is more likely to be emphasized by occurrences of certain keywords, and overall sentiment may not usually be highlighted through repeated use of the same terms [29]. To verify this point, we use three types of weights, i.e., term present (TP), term frequency (TF), and TF–IDF.

2.2.2. Learning methods

Many machine learning methods have been investigated for sentiment classification in the literature. Employing learning-based methods on sentiment analysis, Pang et al. [29] compared three different leaning algorithms (NB, ME, and SVM), concluding that SVM generally achieved the best results. Subsequently, many other studies attempted to improve the performance of machine learning-based sentiment classification [28]. However, within the sentiment classification community, techniques and methods are often used in narrow and separate domains

Table 4
Experiment results (Unigram-TP).

Camera	NB	ME	DT	KNN	SVM
BL	**78.19**±4.98	78.12 ± 5.64	65.52 ± 6.72	60.02 ± 5.92	74.69 ± 5.79
Bagging	77.89 ± 5.28	76.86 ± 5.57	71.49 ± 6.70	59.25 ± 5.92	76.42 ± 6.01
Boosting	76.24 ± 6.10	76.72 ± 5.64	69.96 ± 6.13	60.02 ± 5.92	74.69 ± 5.79
RS	77.93 ± 5.82	77.71 ± 6.13	70.98 ± 6.22	62.13 ± 6.39	76.52 ± 5.47

Camp	NB	ME	DT	KNN	SVM
BL	81.19 ± 4.58	80.06 ± 4.81	75.11 ± 5.06	67.90 ± 4.43	83.05 ± 4.26
Bagging	81.31 ± 4.27	80.21 ± 4.03	79.08 ± 5.01	67.50 ± 4.47	83.46 ± 3.88
Boosting	82.17 ± 4.16	80.06 ± 4.81	79.54 ± 4.20	67.90 ± 4.43	82.79 ± 4.25
RS	80.82 ± 5.25	81.98 ± 3.50	80.20 ± 4.01	72.71 ± 4.72	**85.48** ± 3.54

Doctor	NB	ME	DT	KNN	SVM
BL	75.02 ± 3.51	74.01 ± 3.97	74.74 ± 3.27	65.98 ± 3.26	83.13 ± 2.81
Bagging	74.93 ± 3.49	73.88 ± 3.38	80.71 ± 3.06	64.80 ± 3.35	84.74 ± 2.72
Boosting	81.12 ± 3.90	68.20 ± 4.77	79.67 ± 2.99	65.70 ± 3.28	83.67 ± 2.97
RS	74.78 ± 3.69	73.46 ± 4.00	80.59 ± 2.98	67.59 ± 3.66	**85.97** ± 2.94

Drug	NB	ME	DT	KNN	SVM
BL	68.87 ± 5.23	60.54 ± 6.54	56.10 ± 5.29	52.98 ± 5.22	67.29 ± 4.94
Bagging	68.70 ± 5.61	63.79 ± 6.17	62.19 ± 4.73	53.94 ± 4.98	68.24 ± 4.63
Boosting	68.56 ± 5.09	62.34 ± 6.61	60.89 ± 5.17	52.98 ± 5.22	66.88 ± 4.74
RS	68.69 ± 5.42	61.89 ± 5.37	62.58 ± 4.88	56.18 ± 5.29	**70.26** ± 5.41

Laptop	NB	ME	DT	KNN	SVM
BL	79.90 ± 9.38	71.38 ± 8.69	64.42 ± 11.50	51.76 ± 11.69	79.29 ± 9.00
Bagging	79.52 ± 9.16	69.33 ± 8.12	68.45 ± 11.11	51.29 ± 11.34	**79.96** ± 8.90
Boosting	77.87 ± 9.66	70.59 ± 8.69	69.83 ± 10.38	51.76 ± 11.69	79.29 ± 9.00
RS	78.78 ± 9.38	71.09 ± 9.18	71.05 ± 11.37	57.58 ± 10.96	78.48 ± 9.51

Lawyer	NB	ME	DT	KNN	SVM
BL	80.93 ± 7.14	76.73 ± 8.91	66.25 ± 10.18	64.27 ± 10.52	83.55 ± 7.47
Bagging	81.14 ± 7.63	75.09 ± 9.20	72.00 ± 9.40	63.45 ± 10.61	83.36 ± 7.89
Boosting	81.19 ± 8.73	75.91 ± 8.91	72.82 ± 9.14	64.27 ± 10.52	83.55 ± 7.47
RS	80.96 ± 7.32	79.18 ± 9.67	72.64 ± 9.78	68.41 ± 10.42	**83.82** ± 7.65

Movie	NB	ME	DT	KNN	SVM
BL	81.36 ± 2.92	61.57 ± 3.56	65.85 ± 3.53	55.95 ± 3.25	79.21 ± 2.30
Bagging	81.12 ± 2.97	71.81 ± 3.25	74.39 ± 2.95	56.39 ± 3.27	81.26 ± 2.33
Boosting	82.49 ± 2.87	61.15 ± 3.56	73.35 ± 3.15	55.95 ± 3.25	79.21 ± 2.30
RS	80.95 ± 2.89	79.92 ± 3.05	74.61 ± 3.10	60.79 ± 3.64	**82.54** ± 2.50

Music	NB	ME	DT	KNN	SVM
BL	65.86 ± 5.50	67.49 ± 6.58	59.23 ± 5.45	49.83 ± 5.75	68.73 ± 5.81
Bagging	66.13 ± 5.79	70.03 ± 5.81	64.48 ± 6.42	50.14 ± 5.48	70.03 ± 5.45
Boosting	69.83 ± 4.95	68.72 ± 6.58	63.92 ± 5.46	49.83 ± 5.75	68.73 ± 5.81
RS	65.89 ± 5.90	65.98 ± 5.84	63.35 ± 6.31	54.55 ± 4.87	**71.41** ± 5.05

Radio	NB	ME	DT	KNN	SVM
BL	67.79 ± 5.57	65.08 ± 4.43	62.00 ± 5.11	59.46 ± 4.33	72.36 ± 4.25
Bagging	67.99 ± 5.56	63.96 ± 3.39	66.17 ± 5.69	59.30 ± 4.21	72.39 ± 4.32
Boosting	70.84 ± 5.39	64.27 ± 4.26	65.09 ± 5.06	59.13 ± 4.23	71.74 ± 4.50
RS	67.31 ± 5.60	65.46 ± 3.73	65.41 ± 5.14	59.25 ± 4.04	**74.14** ± 4.40

TV	NB	ME	DT	KNN	SVM
BL	71.90 ± 6.17	72.04 ± 6.61	62.87 ± 6.72	60.64 ± 5.61	**77.94** ± 5.55
Bagging	71.87 ± 6.16	69.79 ± 6.66	68.66 ± 7.21	59.64 ± 5.91	77.26 ± 5.62
Boosting	73.65 ± 5.83	72.04 ± 6.61	66.89 ± 7.07	60.64 ± 5.61	77.94 ± 5.55
RS	70.97 ± 6.55	72.04 ± 6.67	66.89 ± 6.16	62.47 ± 5.91	76.34 ± 6.10

and datasets, and which classification algorithms consistently perform better than others is often unclear. There is no consensus as to which methodology an algorithm developer should adopt for a given problem in a given domain. With respect to this uncertainty, it is not uncommon to construct multiple classifiers and then create an integrated classifier based on overall performance [2,45,48]. Table 1 presents selected previous studies dealing with sentiment analysis using ensemble methods.

Prior studies have shown that such ensemble methods have performed better than single machine learning techniques for sentiment classification [2,34,43]. For example, Wilson et al. [45] first used Boosting for sentiment classification and achieved 23% to 96% improvement in accuracy. Abbasi et al. [2] proposed a new correlation ensemble method, named Support Vector Regression Correlation Ensemble (SVRCE), for affect analysis. Whitehead and Yaeger [43] compared Bagging, Boosting, and Random Subspace for sentiment classification, but in this case only one type of base learner (SVM) was considered. Xia et al. [48] made a comparative study of the effectiveness of Stacking for sentiment classification.

Although some foundational studies have investigated potential ensemble approaches in the area of sentiment classification, research has been limited and more in-depth empirical comparative work is needed. In addition, it is not known whether the ensemble learning methods will accurately predict sentiment across different domains. To fill this research gap, this study will conduct comprehensive empirical experiments over ten multi-domain datasets to systematically compare performance of ensemble learning methods in sentiment classification. Although Stacking method is often used for sentiment classification, the performance of Stacking is difficult to analyze theoretically [30,51]. Similarly, little guidance is available on how to select base learners [30,51]. In this research, therefore, three ensemble techniques (Bagging, Boosting, and Random Subspace), using five well-known base learner classification methods, are tested on ten public sentiment analysis datasets. Undoubtedly, this research will provide more insights into sentiment classification.

3. Ensemble learning for sentiment classification

Ensemble learning is a machine learning paradigm where multiple learners are trained to solve the same problem [30,51]. In contrast to ordinary machine learning approaches that try to learn one hypothesis from the training data, ensemble methods try to construct a set of hypotheses and combine them for use [25].

One of the earliest studies on ensemble learning is Dasarathy and Sheela's research [10], which discussed partitioning the feature space using two or more classifiers. In 1990, Hansen and Salamon showed that the generalization performance of an Artificial Neural Network (ANN) can be improved using an ensemble of similarly configured ANNs [17]. Schapire demonstrated that a strong classifier in Probably Approximately Correct (PAC) sense can be generated by combining weak classifiers through Boosting [33], the predecessor of the suite of AdaBoost algorithms. Since these seminal works, studies in ensemble learning have expanded rapidly, appearing often in the literature under many creative names and ideas [30].

The generalization ability of an ensemble method is usually much stronger than that of a single learner, which makes ensemble methods very attractive. Dietterich [14] gave three reasons based on viewing the nature of machine learning as searching a hypothesis space for the most accurate hypothesis. Firstly, the training data might not provide

Table 5
Experiment results (Unigram-TF).

Camera	NB	ME	DT	KNN	SVM	Camp	NB	ME	DT	KNN	SVM
BL	78.19 ± 5.40	75.77 ± 7.25	64.29 ± 6.50	59.63 ± 6.04	74.89 ± 6.11	BL	81.13 ± 4.75	82.56 ± 5.42	74.48 ± 4.54	68.02 ± 4.67	83.29 ± 4.65
Bagging	**78.72**±5.75	73.93 ± 6.52	71.44 ± 5.43	58.55 ± 6.29	75.94 ± 6.11	Bagging	81.90 ± 4.39	82.42 ± 4.21	80.08 ± 3.75	67.52 ± 5.06	83.69 ± 4.71
Boosting	75.90 ± 6.49	75.77 ± 7.25	70.58 ± 6.91	59.63 ± 6.04	74.89 ± 6.11	Boosting	81.85 ± 4.78	82.56 ± 5.42	79.58 ± 4.83	68.02 ± 4.67	83.16 ± 4.32
RS	78.03 ± 6.18	71.35 ± 6.43	71.08 ± 5.76	61.45 ± 5.68	75.61 ± 5.53	RS	80.90 ± 4.81	76.97 ± 3.97	80.13 ± 3.77	72.21 ± 5.24	**84.86** ± 4.33
Doctor						**Drug**					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	74.73 ± 3.01	67.79 ± 3.84	73.66 ± 4.02	66.09 ± 3.37	82.05 ± 2.94	BL	68.78 ± 4.59	64.48 ± 5.06	57.41 ± 5.49	52.75 ± 4.58	66.72 ± 4.95
Bagging	74.88 ± 3.02	70.51 ± 3.86	80.37 ± 3.19	64.65 ± 3.32	84.21 ± 3.07	Bagging	**69.08** ± 4.80	62.75 ± 4.84	61.90 ± 5.05	52.61 ± 3.98	67.68 ± 4.71
Boosting	81.58 ± 2.90	71.08 ± 4.30	78.56 ± 4.07	65.94 ± 3.32	82.66 ± 3.53	Boosting	68.07 ± 4.73	64.48 ± 5.06	61.28 ± 6.10	52.75 ± 4.58	66.81 ± 4.78
RS	74.70 ± 2.91	73.04 ± 3.35	80.04 ± 3.22	67.80 ± 4.33	**85.84** ± 2.50	RS	68.36 ± 4.45	58.24 ± 4.62	62.38 ± 5.05	54.85 ± 4.68	68.70 ± 5.09
Laptop						**Lawyer**					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	78.65 ± 9.57	81.35 ± 8.37	62.15 ± 11.79	51.31 ± 10.72	77.50 ± 9.31	BL	79.91 ± 7.48	87.91 ± 6.30	64.55 ± 9.71	64.27 ± 7.91	**84.09** ± 7.66
Bagging	77.93 ± 10.30	81.33 ± 8.29	66.82 ± 11.10	52.42 ± 9.40	77.05 ± 9.87	Bagging	80.73 ± 6.75	83.50 ± 6.82	70.73 ± 8.96	63.82 ± 8.47	83.27 ± 7.33
Boosting	78.74 ± 10.62	81.35 ± 8.37	70.25 ± 12.31	51.31 ± 10.72	77.50 ± 9.31	Boosting	81.09 ± 7.61	87.91 ± 6.30	72.79 ± 8.06	64.27 ± 7.91	**84.09** ± 7.66
RS	78.31 ± 11.14	**84.14** ± 8.53	66.95 ± 13.98	56.99 ± 11.85	76.61 ± 9.52	RS	80.91 ± 7.02	86.00 ± 6.19	71.00 ± 9.13	69.00 ± 8.10	83.45 ± 7.30
Movie						**Music**					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	81.15 ± 3.33	63.58 ± 3.18	66.58 ± 2.53	56.19 ± 2.66	79.11 ± 2.40	BL	66.02 ± 7.07	64.37 ± 6.65	58.32 ± 6.78	50.18 ± 6.27	68.87 ± 6.56
Bagging	80.98 ± 3.17	70.50 ± 2.89	74.09 ± 3.12	56.55 ± 2.70	80.79 ± 2.76	Bagging	66.12 ± 6.77	63.85 ± 5.43	64.65 ± 6.37	51.25 ± 5.60	69.69 ± 5.49
Boosting	**82.90** ± 2.83	63.58 ± 3.18	73.03 ± 3.18	56.19 ± 2.66	79.11 ± 2.40	Boosting	70.79 ± 6.86	64.37 ± 6.65	64.65 ± 5.68	50.18 ± 6.27	68.87 ± 6.56
RS	80.69 ± 3.28	81.29 ± 2.94	74.79 ± 2.93	60.38 ± 3.16	81.94 ± 2.88	RS	66.40 ± 6.78	61.19 ± 5.71	63.68 ± 6.24	55.44 ± 5.71	**71.21** ± 5.68
Radio						**TV**					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	67.34 ± 4.53	63.18 ± 6.27	61.07 ± 4.23	59.10 ± 5.35	72.60 ± 4.93	BL	71.15 ± 6.05	74.43 ± 5.68	63.11 ± 6.48	61.15 ± 6.83	**77.15** ± 5.95
Bagging	67.51 ± 4.45	69.23 ± 4.36	67.08 ± 4.70	58.39 ± 5.35	72.18 ± 5.16	Bagging	71.40 ± 6.37	71.13 ± 6.01	68.68 ± 5.46	59.87 ± 6.42	76.85 ± 6.54
Boosting	70.26 ± 4.08	69.39 ± 5.39	64.93 ± 4.77	58.18 ± 5.41	71.68 ± 4.42	Boosting	72.55 ± 6.29	74.43 ± 5.68	65.87 ± 6.99	61.15 ± 6.83	77.15 ± 5.95
RS	66.88 ± 4.42	62.57 ± 4.85	64.34 ± 4.48	59.00 ± 3.52	**73.60** ± 3.83	RS	71.19 ± 7.50	75.23 ± 6.00	67.45 ± 6.20	62.55 ± 6.96	76.51 ± 5.74

Table 6
Experiment results (Unigram-TF–IDF).

Camera	NB	ME	DT	KNN	SVM
BL	**78.22**±5.19	77.01 ± 6.00	66.24 ± 7.18	59.14 ± 6.33	73.04 ± 6.59
Bagging	78.14 ± 5.33	75.36 ± 5.79	73.40 ± 6.60	59.52 ± 6.09	75.41 ± 6.47
Boosting	71.99 ± 6.33	77.01 ± 6.00	72.26 ± 6.76	59.14 ± 6.33	73.04 ± 6.59
RS	77.69 ± 5.52	78.71 ± 6.34	71.50 ± 6.44	59.98 ± 6.75	75.54 ± 6.33

Camp	NB	ME	DT	KNN	SVM
BL	81.64 ± 4.22	80.30 ± 3.80	76.55 ± 6.18	65.66 ± 4.36	82.29 ± 3.95
Bagging	81.77 ± 4.07	80.19 ± 3.95	78.38 ± 5.80	64.29 ± 4.27	**82.54** ± 3.66
Boosting	81.69 ± 4.04	80.30 ± 3.67	75.92 ± 6.18	65.66 ± 4.43	79.18 ± 3.95
RS	80.35 ± 4.21	82.26 ± 4.80	77.53 ± 4.82	74.86 ± 3.98	81.77 ± 3.69

Doctor	NB	ME	DT	KNN	SVM
BL	78.84 ± 3.06	73.73 ± 4.39	72.11 ± 3.44	58.77 ± 4.61	82.17 ± 3.21
Bagging	79.13 ± 3.12	75.42 ± 3.50	76.10 ± 3.36	58.73 ± 4.39	82.59 ± 3.03
Boosting	80.49 ± 3.26	68.24 ± 5.01	73.15 ± 3.50	58.93 ± 4.16	80.69 ± 3.84
RS	78.95 ± 2.91	72.88 ± 3.68	76.40 ± 3.36	71.02 ± 3.01	**83.38** ± 3.09

Drug	NB	ME	DT	KNN	SVM
BL	67.71 ± 4.51	62.44 ± 5.35	59.06 ± 4.97	49.13 ± 5.34	67.31 ± 4.75
Bagging	68.23 ± 4.52	66.00 ± 4.69	63.55 ± 4.28	49.23 ± 4.66	67.64 ± 4.82
Boosting	67.24 ± 5.33	61.94 ± 5.64	61.08 ± 5.06	49.13 ± 5.34	67.13 ± 5.00
RS	67.56 ± 4.76	61.62 ± 4.33	63.53 ± 6.30	55.04 ± 4.62	**69.83** ± 4.51

Laptop	NB	ME	DT	KNN	SVM
BL	**80.29** ± 9.02	70.46 ± 9.18	69.32 ± 9.01	50.51 ± 2.46	76.20 ± 11.22
Bagging	78.42 ± 9.81	66.12 ± 9.59	71.30 ± 11.35	50.22 ± 2.31	74.43 ± 11.77
Boosting	78.21 ± 11.00	70.46 ± 9.18	72.53 ± 10.84	50.51 ± 2.46	76.20 ± 11.22
RS	78.13 ± 9.05	71.70 ± 10.16	74.71 ± 11.08	50.62 ± 2.79	75.27 ± 10.14

Lawyer	NB	ME	DT	KNN	SVM
BL	78.68 ± 7.53	77.23 ± 8.49	64.91 ± 10.07	51.00 ± 9.85	77.09 ± 7.86
Bagging	79.64 ± 6.73	74.27 ± 8.12	67.77 ± 8.72	51.91 ± 9.95	77.45 ± 8.17
Boosting	73.82 ± 9.28	77.23 ± 8.49	63.45 ± 9.86	51.00 ± 9.85	63.50 ± 11.44
RS	78.18 ± 8.02	**80.23** ± 7.57	67.32 ± 9.94	62.77 ± 12.12	78.27 ± 7.80

Movie	NB	ME	DT	KNN	SVM
BL	81.09 ± 2.62	61.54 ± 3.76	65.80 ± 3.35	50.66 ± 1.48	78.77 ± 2.69
Bagging	80.80 ± 2.65	75.13 ± 3.21	75.78 ± 2.61	51.41 ± 2.10	81.35 ± 2.60
Boosting	81.54 ± 2.59	61.54 ± 3.76	73.90 ± 3.02	50.66 ± 1.48	78.77 ± 2.69
RS	80.97 ± 2.73	80.08 ± 2.61	74.98 ± 2.75	51.99 ± 2.01	**81.60** ± 2.22

Music	NB	ME	DT	KNN	SVM
BL	68.83 ± 6.66	67.77 ± 6.75	57.83 ± 6.62	50.65 ± 5.47	69.31 ± 5.58
Bagging	69.15 ± 6.72	71.74 ± 5.65	63.92 ± 5.29	51.38 ± 5.64	69.69 ± 5.49
Boosting	69.32 ± 6.00	67.77 ± 6.75	60.62 ± 6.66	50.65 ± 5.47	66.66 ± 5.67
RS	68.77 ± 6.99	64.84 ± 5.96	62.96 ± 5.97	53.50 ± 6.07	**72.13** ± 5.52

Radio	NB	ME	DT	KNN	SVM
BL	63.94 ± 4.65	64.22 ± 3.89	58.30 ± 4.95	56.87 ± 4.15	**68.03** ± 4.33
Bagging	64.52 ± 4.55	62.84 ± 3.39	61.75 ± 4.49	56.45 ± 4.35	67.13 ± 3.80
Boosting	65.93 ± 5.20	63.96 ± 3.66	59.95 ± 5.15	56.87 ± 4.15	65.50 ± 4.30
RS	64.02 ± 4.90	66.87 ± 4.09	59.75 ± 5.61	62.53 ± 4.61	66.87 ± 3.90

TV	NB	ME	DT	KNN	SVM
BL	70.77 ± 6.94	72.09 ± 6.10	61.91 ± 6.69	54.87 ± 4.61	**73.47** ± 6.55
Bagging	70.70 ± 6.16	67.02 ± 6.32	65.30 ± 7.51	53.64 ± 4.00	72.68 ± 6.12
Boosting	71.02 ± 6.20	72.09 ± 6.10	63.96 ± 7.27	54.87 ± 4.61	**73.47** ± 6.55
RS	70.23 ± 6.34	72.60 ± 6.64	61.51 ± 6.63	61.85 ± 5.35	71.72 ± 6.16

sufficient information for choosing a single best learner. For example, there may be many base learners performing equally well on the training set. Thus, combining these learners may be a better choice. Secondly, the search processes of the learning algorithms might be imperfect. For example, even if there is a unique best hypothesis, it might be difficult to attain this goal, since running the algorithms results in sub-optimal hypotheses. Thus, ensembles can compensate for such imperfect search processes. Thirdly, the hypothesis space being searched might not contain the true target function, while ensembles can give some good approximation. For example, the classification boundaries of DTs are linear segments parallel to coordinate axes. If the target classification boundary is a smooth diagonal line, using a single DT cannot lead to a good result. But a good approximation can be achieved by combining a set of DTs. Although these intuitive explanations are reasonable, they lack rigorous theoretical analyses.

In practice, to achieve a good ensemble, two necessary conditions should be satisfied: accuracy and diversity [46]. The base learner should be more accurate than random guessing, and each base learner should have its own knowledge about the problem, with a different pattern of errors than other base learners. In general, ensemble learning methods can be divided into two categories: instance partitioning methods and feature partitioning methods [30,51]. Bagging and Boosting are instance partitioning methods; Random Subspace is a feature partitioning method.

3.1. Bagging

Bagging (short for bootstrap aggregating) is one of the earliest ensemble learning algorithms [5]. It is also one of the most intuitive and simplest to implement, with a surprisingly good performance. Diversity in Bagging is obtained by using bootstrapped replicas of the training data: different training data subsets are randomly drawn—with replacement—from the entire training dataset [40,51]. Each training data subset is used to train a different base learner of the same type.

The base learners' combination strategy for Bagging is majority vote. This simple strategy can reduce variance when combined with the base learner generation strategies. The Bagging algorithm process and pseudo-code are shown in Figs. 1 and 2.

Bagging is particularly appealing when the available data are of limited size. To ensure that there are sufficient training samples in each subset, relatively large portions of the samples (75% to 100%) are drawn into each subset. This causes individual training subsets to overlap significantly, with many of the same instances appearing in most subsets and some instances appearing multiple times in a given subset. To ensure diversity under this scenario, a relatively unstable base learner is used so that sufficiently different decision boundaries can be obtained for small perturbations in different training datasets.

3.2. Boosting

Boosting [33] encompasses a family of methods. Unlike Bagging, Boosting creates different base learners by sequentially reweighting the instances in the training dataset. Each instance misclassified by the previous base learner will get a larger weight in the next round of training.

The basic idea of Boosting is to repeatedly apply a base learner to modified versions of the training dataset, thereby producing a sequence of base learners for a predefined number of iterations. To begin with, all

Table 7
Experiment results (Bigram-TP).

Camera						Camp					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	77.07 ± 5.82	**80.75**±6.73	60.12 ± 7.51	45.95 ± 7.46	74.03 ± 5.81	BL	77.24 ± 5.24	74.41 ± 6.81	69.67 ± 4.26	51.59 ± 4.63	77.76 ± 3.89
Bagging	76.83 ± 6.73	77.50 ± 5.92	67.40 ± 6.93	47.97 ± 7.15	75.03 ± 6.34	Bagging	77.41 ± 5.52	78.06 ± 4.31	72.11 ± 4.63	51.87 ± 4.81	78.80 ± 3.92
Boosting	74.84 ± 6.34	77.42 ± 7.13	64.48 ± 7.55	45.09 ± 7.12	72.46 ± 6.43	Boosting	77.81 ± 4.50	74.33 ± 5.67	70.72 ± 4.15	51.59 ± 4.63	73.98 ± 4.14
RS	76.47 ± 7.12	80.43 ± 6.30	64.81 ± 7.25	55.09 ± 8.02	76.28 ± 7.25	RS	77.01 ± 5.42	74.55 ± 4.82	73.13 ± 4.49	67.62 ± 4.80	**81.42** ± 4.63
Doctor						Drug					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	72.27 ± 4.06	66.26 ± 4.58	66.66 ± 3.45	62.73 ± 3.98	77.61 ± 3.11	BL	68.28 ± 4.24	66.93 ± 4.71	56.61 ± 4.60	51.04 ± 4.76	65.56 ± 4.81
Bagging	72.80 ± 3.82	70.95 ± 3.54	70.78 ± 3.31	62.52 ± 3.46	79.35 ± 2.83	Bagging	68.78 ± 5.36	68.42 ± 5.72	61.87 ± 5.02	50.94 ± 4.93	66.63 ± 5.10
Boosting	77.12 ± 3.66	66.65 ± 4.01	69.16 ± 3.38	62.28 ± 3.90	78.21 ± 3.31	Boosting	66.46 ± 5.14	66.46 ± 4.82	59.06 ± 5.27	51.04 ± 4.76	66.03 ± 4.69
RS	72.40 ± 4.17	67.72 ± 4.12	72.18 ± 3.32	70.47 ± 3.74	**81.41** ± 2.97	RS	67.80 ± 4.51	62.50 ± 5.38	61.57 ± 4.25	56.16 ± 5.31	**68.88** ± 4.96
Laptop						Lawyer					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	77.26 ± 9.63	92.09 ± 5.65	58.85 ± 11.42	49.93 ± 4.73	73.44 ± 10.52	BL	76.50 ± 7.56	**82.59** ± 8.10	59.68 ± 8.87	51.91 ± 3.94	78.45 ± 8.09
Bagging	78.14 ± 10.14	84.02 ± 6.75	63.17 ± 11.40	50.14 ± 3.65	71.56 ± 9.92	Bagging	77.59 ± 7.62	77.91 ± 7.62	64.05 ± 8.55	51.27 ± 4.39	78.27 ± 8.54
Boosting	75.57 ± 9.88	92.09 ± 5.65	61.55 ± 11.40	49.93 ± 4.73	73.44 ± 10.52	Boosting	76.82 ± 9.66	**82.59** ± 8.10	61.45 ± 7.95	51.91 ± 3.94	78.45 ± 8.09
RS	75.62 ± 9.98	**92.20** ± 5.83	62.32 ± 9.98	50.71 ± 5.81	75.85 ± 10.36	RS	75.68 ± 8.26	79.68 ± 8.11	63.05 ± 7.82	52.00 ± 4.10	77.27 ± 9.36
Movie						Music					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	**76.55** ± 2.81	57.98 ± 3.45	60.73 ± 2.87	51.57 ± 2.47	71.76 ± 2.91	BL	64.29 ± 6.32	61.11 ± 8.01	53.92 ± 6.22	50.35 ± 5.96	69.28 ± 5.61
Bagging	75.96 ± 2.68	61.17 ± 3.07	68.09 ± 2.91	51.94 ± 2.89	74.76 ± 2.84	Bagging	65.29 ± 6.90	67.58 ± 6.70	58.56 ± 6.44	50.66 ± 5.39	68.96 ± 5.50
Boosting	75.93 ± 2.96	57.98 ± 3.45	66.49 ± 3.31	51.57 ± 2.47	71.76 ± 2.91	Boosting	67.25 ± 6.10	61.11 ± 8.01	57.46 ± 5.84	50.35 ± 5.96	62.68 ± 7.68
RS	75.85 ± 2.99	74.33 ± 2.90	67.36 ± 2.95	54.15 ± 3.49	75.40 ± 2.34	RS	65.05 ± 6.56	66.91 ± 6.05	57.15 ± 6.71	55.61 ± 7.34	**72.02** ± 4.26
Radio						TV					
	NB	ME	DT	KNN	SVM		NB	ME	DT	KNN	SVM
BL	70.41 ± 3.88	**82.71** ± 3.36	56.25 ± 4.16	58.20 ± 4.34	70.94 ± 4.62	BL	66.98 ± 6.35	**71.28** ± 6.19	56.17 ± 6.14	50.77 ± 2.14	70.55 ± 6.85
Bagging	69.80 ± 4.29	76.52 ± 3.58	59.88 ± 4.53	58.29 ± 4.48	71.09 ± 4.28	Bagging	66.43 ± 6.11	65.13 ± 5.31	55.79 ± 5.92	50.55 ± 2.16	69.96 ± 6.47
Boosting	70.97 ± 4.07	79.54 ± 3.80	56.93 ± 3.97	57.42 ± 4.50	68.50 ± 4.52	Boosting	66.98 ± 6.20	**71.28** ± 6.19	56.89 ± 5.92	50.77 ± 2.14	70.55 ± 6.85
RS	70.19 ± 4.26	75.31 ± 4.53	59.37 ± 4.65	62.10 ± 3.88	73.78 ± 4.20	RS	67.19 ± 6.54	64.38 ± 5.47	56.00 ± 6.23	51.57 ± 4.25	70.00 ± 7.76

the instances are initialized with uniform weights. After this initialization, each boosting iteration fits a base learner to the weighted training data. Error is computed and the weight of the correctly classified instances is lowered while the incorrectly classified instances will get higher weights. The final model obtained by the Boosting algorithm is a linear combination of several base learners weighted by their own performance.

Even though there are several versions of Boosting algorithms, the most widely used is the one proposed by Freund and Schapire [30,33], which is known as AdaBoost. Therefore, we use the AdaBoost algorithm in this study. The algorithm's process and pseudo-code are shown in Figs. 3 and 4.

3.3. Random Subspace

The Random Subspace method is an ensemble construction technique proposed by Ho [19]. In Random Subspace, the training dataset is modified as in Bagging. However, this modification is performed in the feature space rather than in the instance space. The process and pseudo-code for the Random Subspace algorithm are shown in Figs. 5 and 6.

The Random Subspace method may benefit from using random subspaces for both constructing and aggregating the base classifiers. When the dataset has many redundant or irrelevant features, one may obtain better base classifiers in random subspaces than in the original feature space [19]. The combined decision of such base classifiers may be superior to a single classifier constructed on the original training dataset in the complete feature sets.

4. Experimental design

4.1. Experimental datasets

To verify the effectiveness of ensemble learning for sentiment analysis, we investigated ten public sentiment analysis datasets from a wide variety of domains. The Movie dataset was collected from the commonly-used Cornell movie-review dataset [27]. It consisted of four collections of movie-review documents labeled with sentiment polarities (positive or negative) or ratings on a scale from 1 to 5, and movie-review sentences labeled with subjectivity statuses (subjective or objective) or polarities. In our experiments, the documents labeled with polarities were chosen as the dataset, which contained 1000 positive and 1000 negative reviews. The other nine sentiment analysis datasets were provided by Whitehead and Yaeger [42]. These datasets included reviews and corresponding ratings, in which the rating of "1" was a positive sentiment and the rating of "−1" was a negative sentiment. Except for the Camera dataset that contained 250 positive instances and 248 negative instances, the other eight datasets had the same number of positive and negative instances. The summary descriptions of the datasets are shown in Table 2.

4.2. Performance evaluation

The established standard measure in sentiment analysis, average accuracy, was adopted to evaluate the performance of the proposed method. The definition of average accuracy can be explained with a confusion matrix as shown in Table 3.

Table 8
Experiment results (Bigram-TF).

Camera	NB	ME	DT	KNN	SVM
BL	78.15 ± 5.50	**80.83**±5.50	62.22 ± 6.29	46.67 ± 5.64	74.52 ± 5.77
Bagging	78.03 ± 5.35	77.77 ± 5.63	66.99 ± 7.39	47.91 ± 6.26	75.44 ± 5.48
Boosting	74.72 ± 6.08	75.90 ± 5.91	65.33 ± 6.84	45.72 ± 5.80	72.56 ± 6.32
RS	76.99 ± 4.71	79.56 ± 4.13	66.13 ± 6.09	55.48 ± 5.80	76.13 ± 5.52

Camp	NB	ME	DT	KNN	SVM
BL	77.23 ± 4.72	74.05 ± 6.97	69.69 ± 4.32	51.33 ± 5.04	78.58 ± 4.01
Bagging	77.35 ± 4.80	77.97 ± 4.14	71.91 ± 4.20	51.74 ± 4.84	79.33 ± 3.69
Boosting	77.35 ± 5.03	74.43 ± 5.22	71.38 ± 5.37	51.33 ± 5.04	74.66 ± 3.39
RS	77.84 ± 5.63	73.39 ± 4.13	71.91 ± 5.31	67.25 ± 4.86	**81.24** ± 4.55

Doctor	NB	ME	DT	KNN	SVM
BL	72.41 ± 4.42	66.10 ± 3.86	67.09 ± 3.95	62.31 ± 3.84	77.27 ± 3.56
Bagging	72.59 ± 4.55	70.49 ± 3.74	71.48 ± 3.91	61.94 ± 3.78	78.31 ± 3.28
Boosting	77.58 ± 4.36	66.36 ± 4.22	68.64 ± 3.92	62.04 ± 3.35	77.83 ± 3.45
RS	72.40 ± 4.39	66.33 ± 2.98	72.20 ± 3.96	70.11 ± 4.24	**80.75** ± 3.19

Drug	NB	ME	DT	KNN	SVM
BL	68.37 ± 5.48	66.77 ± 5.01	57.47 ± 5.43	51.16 ± 4.50	65.10 ± 4.80
Bagging	68.62 ± 4.97	67.20 ± 5.05	60.84 ± 5.04	50.81 ± 4.31	66.28 ± 5.11
Boosting	66.83 ± 4.69	66.10 ± 5.01	59.26 ± 5.27	51.25 ± 4.57	66.00 ± 4.91
RS	67.79 ± 5.45	63.29 ± 6.06	61.48 ± 5.39	56.21 ± 5.31	**69.12** ± 4.43

Laptop	NB	ME	DT	KNN	SVM
BL	77.04 ± 8.62	92.45 ± 5.97	57.78 ± 11.36	49.54 ± 3.42	74.29 ± 11.44
Bagging	77.70 ± 7.68	86.71 ± 6.84	61.65 ± 8.81	50.05 ± 2.97	74.47 ± 10.58
Boosting	74.08 ± 10.27	92.45 ± 5.97	63.84 ± 8.96	49.54 ± 3.42	74.29 ± 11.44
RS	76.41 ± 9.84	**92.62** ± 5.75	61.24 ± 8.93	50.34 ± 3.61	76.02 ± 12.25

Lawyer	NB	ME	DT	KNN	SVM
BL	76.91 ± 7.96	**81.00** ± 7.23	59.00 ± 10.11	52.00 ± 3.44	80.27 ± 7.40
Bagging	76.82 ± 7.71	77.36 ± 7.06	63.45 ± 10.29	52.09 ± 4.40	78.73 ± 7.56
Boosting	75.64 ± 7.47	**81.00** ± 7.23	61.91 ± 9.96	52.00 ± 3.44	80.27 ± 7.40
RS	75.91 ± 7.66	80.91 ± 9.18	62.45 ± 9.62	53.91 ± 6.26	76.00 ± 7.48

Movie	NB	ME	DT	KNN	SVM
BL	**76.70** ± 2.24	58.73 ± 3.60	60.82 ± 3.18	51.50 ± 2.09	72.62 ± 3.23
Bagging	76.43 ± 2.43	60.63 ± 2.79	67.92 ± 3.71	51.87 ± 2.39	74.81 ± 2.93
Boosting	75.75 ± 3.04	58.73 ± 3.60	66.56 ± 3.08	51.50 ± 2.09	72.62 ± 3.23
RS	76.08 ± 2.39	74.57 ± 2.87	67.91 ± 3.25	54.59 ± 2.93	75.61 ± 3.26

Music	NB	ME	DT	KNN	SVM
BL	64.90 ± 5.20	60.30 ± 7.86	53.47 ± 5.98	50.10 ± 6.13	69.68 ± 5.39
Bagging	65.70 ± 5.24	64.45 ± 6.08	58.49 ± 5.52	50.10 ± 5.90	69.16 ± 6.26
Boosting	66.90 ± 6.15	60.30 ± 7.86	57.28 ± 5.57	50.10 ± 6.13	62.30 ± 5.15
RS	65.42 ± 4.80	68.39 ± 5.37	57.34 ± 6.34	56.53 ± 5.33	**72.10** ± 6.12

Radio	NB	ME	DT	KNN	SVM
BL	70.42 ± 4.18	**82.45** ± 3.71	56.29 ± 4.33	58.07 ± 4.00	71.68 ± 4.26
Bagging	70.04 ± 4.59	76.24 ± 4.23	58.87 ± 3.95	58.09 ± 3.66	71.01 ± 4.16
Boosting	70.80 ± 4.25	79.46 ± 4.11	56.59 ± 3.90	57.07 ± 4.23	69.05 ± 4.56
RS	70.42 ± 4.26	75.96 ± 4.93	59.86 ± 4.25	62.37 ± 4.72	74.48 ± 4.23

TV	NB	ME	DT	KNN	SVM
BL	66.72 ± 6.72	**70.34** ± 6.93	55.02 ± 5.19	51.11 ± 2.85	69.91 ± 7.17
Bagging	66.55 ± 6.58	64.11 ± 6.85	57.83 ± 6.84	50.60 ± 2.27	68.94 ± 7.17
Boosting	65.11 ± 6.67	**70.34** ± 6.93	57.06 ± 6.42	51.11 ± 2.85	69.91 ± 7.17
RS	66.43 ± 7.51	66.13 ± 6.50	55.57 ± 5.64	51.02 ± 3.67	68.47 ± 6.45

Formally, average accuracy is defined as follows:

$$\text{Average accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

4.3. Experimental procedure

To minimize the influence of variability in the training set, 10-fold cross validation was performed ten times on the ten sentiment analysis datasets. In detail, each sentiment analysis dataset was partitioned into ten subsets with similar sizes and distributions. Then, the union of nine subsets was used as the training set while the remaining subset is used as the test set. The process was repeated ten times, such that every subset had been used as the test set once. The average test result was regarded as the result of the 10-fold cross validation. The process was repeated for 10 times with random partitions of the ten subsets, and the average results of these different partitions were recorded.

Ensemble methods are composed of several base learners. Based on the literature review [29,34,48], we chose five widely used base learners for our experiment: NB, ME, DT, KNN, and SVM.

NB is a simple probabilistic classification method based on applying Bayes' theorem with strong independence assumptions [7]. It is very easy to construct, not requiring any complicated iterative parameter estimation. Also, it is readily applied to huge datasets. The main disadvantage is that the conditional independence assumption is violated by real-world data.

ME is one of the best methods for natural language processing [29]. Unlike NB, ME makes no assumptions about the relations between features, and therefore it may perform better when conditional independence assumptions are not met.

DT has been widely used in building classification models because it closely resembles human reasoning and is easy to understand [31]. DT is a sequential model, which logically combines a sequence of simple tests. Each test compares a numeric attribute against a threshold value or a nominal attribute against a set of possible values. In this study, we chose the widely used C4.5 for our experiments.

KNN is one of the simplest and rather trivial classification methods [8]. An object of KNN is classified by a majority vote of its neighbors. If $K = 1$, then the object is simply assigned to the class label of its nearest neighbor. One of the major drawbacks of KNN is that the classifier needs available data. This may lead to considerable overhead if the training dataset is large. In this study, we choose $K = 1$.

SVM is a state-of-the-art data mining technique that has proven its performance in many applications [39]. It has a sound theoretical foundation and requires a dozen instances for training. The strength of this technique lies with its ability to model non-linearity, resulting in complex mathematical models. SVM can capture the inherent characteristics of the data better than ANN can.

Three ensemble methods, i.e., Bagging, Boosting, and Random Subspace, were implemented respectively with the five base learners. As discussed in the literature review and following [29], Unigram and Bigram weighted by term present, term frequency, and TF–IDF were selected to express the text information. A total of 1200 comparative group experiments (6 feature sets × 20 classifiers × 10 datasets) were

Table 9
Experiment results (Bigram-TF–IDF).

Camera	NB	ME	DT	KNN	SVM
BL	77.44 ± 6.20	**80.86**±5.40	61.33 ± 5.42	46.48 ± 6.49	74.51 ± 5.75
Bagging	76.99 ± 6.36	76.82 ± 4.88	66.99 ± 6.93	47.97 ± 6.32	75.92 ± 5.86
Boosting	73.67 ± 6.78	77.24 ± 5.86	65.02 ± 7.64	45.95 ± 6.25	73.42 ± 5.20
RS	76.35 ± 6.13	80.85 ± 4.39	66.64 ± 6.16	55.50 ± 7.38	77.12 ± 5.16

Camp	NB	ME	DT	KNN	SVM
BL	77.26 ± 4.62	73.86 ± 5.03	68.71 ± 4.22	51.10 ± 5.51	78.65 ± 4.40
Bagging	77.45 ± 4.60	77.65 ± 4.68	72.04 ± 4.55	51.46 ± 5.34	79.12 ± 4.28
Boosting	77.56 ± 4.76	73.94 ± 4.66	70.87 ± 4.66	51.10 ± 5.51	74.33 ± 4.83
RS	76.59 ± 4.04	74.66 ± 4.32	72.21 ± 3.66	67.24 ± 4.36	**81.59** ± 4.13

Doctor	NB	ME	DT	KNN	SVM
BL	72.27 ± 3.54	65.93 ± 6.19	67.52 ± 3.38	62.69 ± 4.09	77.22 ± 3.07
Bagging	72.79 ± 3.80	70.97 ± 3.96	70.99 ± 3.51	62.27 ± 3.87	78.89 ± 3.40
Boosting	76.80 ± 3.52	65.32 ± 5.45	69.21 ± 3.62	62.36 ± 4.02	78.10 ± 2.97
RS	72.30 ± 3.47	66.35 ± 3.45	71.72 ± 3.00	70.31 ± 3.69	**81.17** ± 3.31

Drug	NB	ME	DT	KNN	SVM
BL	67.92 ± 4.77	67.66 ± 5.69	56.67 ± 5.96	50.73 ± 5.28	65.09 ± 4.69
Bagging	68.17 ± 4.71	67.87 ± 5.20	60.56 ± 4.51	50.99 ± 5.44	66.47 ± 4.78
Boosting	66.53 ± 4.51	66.86 ± 6.02	58.63 ± 5.43	51.05 ± 5.22	65.79 ± 4.73
RS	67.45 ± 4.73	63.63 ± 5.48	60.84 ± 4.60	56.51 ± 4.35	**68.88** ± 4.17

Laptop	NB	ME	DT	KNN	SVM
BL	76.46 ± 10.12	92.05 ± 6.05	58.95 ± 11.31	49.32 ± 4.54	72.42 ± 9.29
Bagging	75.90 ± 9.99	85.80 ± 6.90	60.31 ± 10.17	50.44 ± 3.47	72.44 ± 10.03
Boosting	75.33 ± 9.39	92.05 ± 6.05	61.24 ± 10.59	49.32 ± 4.54	72.42 ± 9.29
RS	76.33 ± 11.21	**90.92** ± 6.26	60.27 ± 11.05	50.55 ± 4.13	75.59 ± 9.81

Lawyer	NB	ME	DT	KNN	SVM
BL	76.18 ± 8.40	**81.64** ± 8.62	60.18 ± 8.50	51.55 ± 2.98	79.00 ± 7.42
Bagging	77.36 ± 8.30	76.36 ± 8.03	62.64 ± 7.77	51.82 ± 3.17	77.45 ± 7.13
Boosting	75.55 ± 8.85	**81.64** ± 8.62	62.82 ± 9.78	51.55 ± 2.98	79.00 ± 7.42
RS	75.18 ± 8.28	79.18 ± 8.43	62.45 ± 7.96	53.18 ± 5.27	76.27 ± 7.15

Movie	NB	ME	DT	KNN	SVM
BL	76.59 ± 2.79	58.36 ± 3.22	60.41 ± 3.79	51.70 ± 2.32	72.08 ± 2.88
Bagging	**76.51** ± 2.82	55.48 ± 3.30	68.31 ± 3.01	52.26 ± 2.59	74.05 ± 3.34
Boosting	75.93 ± 3.01	58.36 ± 3.22	66.69 ± 2.70	51.70 ± 2.32	72.08 ± 2.88
RS	76.06 ± 2.83	74.77 ± 2.81	67.60 ± 2.88	54.43 ± 2.66	75.71 ± 2.74

Music	NB	ME	DT	KNN	SVM
BL	64.32 ± 6.73	60.22 ± 7.93	53.81 ± 6.97	50.38 ± 4.43	68.48 ± 6.42
Bagging	64.93 ± 6.83	63.80 ± 6.16	58.86 ± 5.44	50.24 ± 4.56	70.10 ± 6.03
Boosting	66.29 ± 6.16	60.22 ± 7.93	55.63 ± 6.81	50.38 ± 4.43	63.02 ± 5.69
RS	64.15 ± 7.35	67.69 ± 5.82	57.17 ± 6.15	56.49 ± 5.93	**72.02** ± 6.46

Radio	NB	ME	DT	KNN	SVM
BL	70.35 ± 4.50	**82.76** ± 3.57	55.72 ± 3.79	57.68 ± 4.62	71.02 ± 4.50
Bagging	69.87 ± 4.10	77.13 ± 3.94	59.05 ± 4.67	57.88 ± 4.60	71.05 ± 4.94
Boosting	71.33 ± 4.44	80.02 ± 3.87	57.51 ± 4.36	57.08 ± 4.04	68.72 ± 3.39
RS	69.97 ± 4.41	75.52 ± 4.94	59.14 ± 4.55	61.99 ± 4.20	74.70 ± 4.27

TV	NB	ME	DT	KNN	SVM
BL	67.66 ± 7.08	**71.32** ± 5.77	56.26 ± 7.09	50.72 ± 2.61	70.34 ± 5.87
Bagging	67.40 ± 6.88	64.83 ± 5.97	58.85 ± 7.04	50.43 ± 2.54	70.55 ± 6.74
Boosting	67.74 ± 5.85	**71.32** ± 5.77	58.60 ± 6.59	50.72 ± 2.61	70.34 ± 5.87
RS	67.40 ± 6.29	65.74 ± 6.42	57.40 ± 4.92	51.36 ± 3.78	69.91 ± 7.40

conducted to verify the effectiveness of ensemble learning for sentiment classification. The experimental procedure is shown in Fig. 7.

5. Experimental results and analysis

The experiments were performed on a PC with a 3.10 GHz AMD FX(tm)-8120 Eight-Core CPU and 8 GB RAM, using Windows 7 operating system. We used the data mining toolkit WEKA (Waikato Environment for Knowledge Analysis) version 3.7.0. This open-source toolkit includes a collection of machine learning algorithms for solving data mining problems [47].

In this study, we compared the performances of 20 methods, including NB, ME, DT, KNN and SVM, and their corresponding ensemble methods of Bagging, Boosting, and Random Subspace. Among these methods, the NB algorithm, ME algorithm, KNN algorithm, and SVM algorithm were implement by the Naive Bayes module, logistic module (WEKA's own version of multinomial logistic regression), IBk module, and SMO module of WEKA, respectively. The DT algorithm was implemented by the J48 module (WEKA's own version of C4.5). The Bagging module, AdaBoost M1 module, and Random SubSpace module of WEKA were used to implement respective algorithms. As the original datasets were text forms, WEKA's StringToWordVector filter was used to convert original texts into an N-gram representation. Except when stated otherwise, all the default parameters in WEKA were used.

5.1. Basic experimental results

Tables 4 to 9 summarize the experimental results of base learners and ensemble methods in sentiment classification, where the values following ± are standard deviations. The highest average accuracies of different datasets are boldfaced.

The highest average accuracy of the Camera dataset is 80.86% using ME. The highest average accuracy of the Camp dataset is 85.48% using RS-SVM. The highest average accuracy of the Doctor dataset is 85.97% using RS-SVM. The highest average accuracy of the Drug dataset is 70.26% using RS-SVM. The highest average accuracy of the Laptop dataset is 92.62% using RS-ME. The highest average accuracy of the Lawyer dataset is 84.09% using SVM. The highest average accuracy of the Movie dataset is 82.54% using RS-SVM. The highest average accuracy of the Music dataset is 72.13% using RS-SVM. The highest average accuracy of the Radio dataset is 82.76% using ME. The highest average accuracy of the TV dataset is 77.94% using SVM.

Among the ten datasets, SVM and ensemble methods using SVM as the base learner have eight of the highest average accuracies. These findings indicate SVM has more powerful competitiveness in sentiment classification. This is consistent with previous research [2,29,43]. In addition, RS-SVM has the six highest average accuracies and similar average accuracies with other four datasets. It is interesting that the ensemble methods with the highest average accuracies are all based on Random Subspace. A potential explanation is that since the sentiment classification problem has tens of thousands of features, a feature partitioning method is better able to address this problem.

5.2. Analysis and discussion from the ensemble methods perspective

To ensure that the assessment does not happen by chance, we tested the significance of these results. Following [13,16], we firstly conducted an Iman–Davenport test [21], to ascertain whether there are significant differences among all methods. Then, pairwise differences were measured using a Wilcoxon test [13]. The formulation of the test

Table 10
Outcomes of Wilcoxon matched-pairs signed-ranks test (Unigram-TP).

	Bagging NB		Boosting NB		RS NB	
	s	P_w	s	P_w	s	P_w
NB	2/3/5	1.812	6/3/1	7.434**	0/3/7	5.338**
Bagging NB			6/2/2	8.031**	0/3/7	2.384*
Boosting NB					2/2/6	8.996**
	Bagging ME		Boosting ME		RS ME	
	s	P_w	s	P_w	s	P_w
ME	3/2/5	2.828**	0/8/2	8.514**	5/2/3	6.996**
Bagging ME			5/1/4	6.141**	7/0/3	7.709**
Boosting ME					6/2/2	9.654**
	Bagging DT		Boosting DT		RS DT	
	s	P_w	s	P_w	s	P_w
DT	10/0/0	19.700**	10/0/0	19.688**	10/0/0	19.884**
Bagging DT			3/0/7	3.332**	5/0/5	0.138
Boosting DT					7/2/1	3.080**
	Bagging KNN		Boosting KNN		RS KNN	
	s	P_w	s	P_w	s	P_w
KNN	3/0/7	2.689**	0/8/2	2.725**	9/0/1	14.280**
Bagging KNN			6/0/4	2.092*	9/1/0	15.433**
Boosting KNN					9/1/0	14.547**
	Bagging SVM		Boosting SVM		RS SVM	
	s	P_w	s	P_w	s	P_w
SVM	7/2/1	7.305**	1/6/3	0.880	7/1/2	9.942**
Bagging SVM			1/1/8	7.612**	7/1/2	5.749**
Boosting SVM					7/1/2	10.030**

Note: Iman–Davenport test: 0.000. *P-values significant at alpha = 0.05. **P-values significant at alpha = 0.01.

Table 11
Outcomes of Wilcoxon matched-pairs signed-ranks test (Unigram-TF).

	Bagging NB		Boosting NB		RS NB	
	s	P_w	s	P_w	s	P_w
NB	6/2/2	2.679**	7/1/2	9.306**	2/4/4	0.912
Bagging NB			6/2/2	7.959**	1/3/6	2.785**
Boosting NB					2/2/6	9.635**
	Bagging ME		Boosting ME		RS ME	
	s	P_w	s	P_w	s	P_w
ME	3/2/5	1.134	2/8/0	11.035**	4/0/6	0.027
Bagging ME			6/3/1	4.086**	5/0/5	0.094
Boosting ME					4/0/6	3.467**
	Bagging DT		Boosting DT		RS DT	
	s	P_w	s	P_w	s	P_w
DT	10/0/0	20.779**	10/0/0	20.928**	10/0/0	19.162**
Bagging DT			2/1/7	3.331**	2/3/5	1.890
Boosting DT					6/0/4	1.580
	Bagging KNN		Boosting KNN		RS KNN	
	s	P_w	s	P_w	s	P_w
KNN	3/2/5	3.224**	0/8/2	3.520**	9/1/0	12.992**
Bagging KNN			4/2/4	2.314*	10/0/0	14.153**
Boosting KNN					10/0/0	13.514**
	Bagging SVM		Boosting SVM		RS SVM	
	s	P_w	s	P_w	s	P_w
SVM	6/0/4	5.073**	1/8/1	0.625	7/0/3	8.455**
Bagging SVM			3/0/7	5.386**	6/3/1	5.213**
Boosting SVM					7/0/3	8.478**

Note: Iman–Davenport test: 0.000. *P-values significant at alpha = 0.05. **P-values significant at alpha = 0.01.

[44] is as follows. Let d_i be the difference between the error values of the methods in ith data set. These differences are ranked according to their absolute values; in case of ties, an average rank is assigned. Let R^+ be the sum of ranks for the data sets on which the second algorithm outperformed the first, and let R^- be the sum of ranks where the first algorithm outperformed the second. Ranks are split evenly among the sums

$$R^+ = \sum_{d_i>0} \text{rank}(d_i) + \frac{1}{2}\sum_{d_i=0} \text{rank}(d_i) \quad (2)$$

and

$$R^- = \sum_{d_i<0} \text{rank}(d_i) + \frac{1}{2}\sum_{d_i=0} \text{rank}(d_i) \quad (3)$$

Let T be the smaller of the two sums and N be the number of data sets. For a small N, there are tables with the exact critical values for T. For a larger N, the statistics

$$z = \frac{T - \frac{1}{4}N(N+1)}{\sqrt{\frac{1}{24}N(N+1)(2N+1)}} \quad (4)$$

is distributed approximately according to $N(0,1)$. We combined these two tests to assess the performance difference of the different algorithms. When the comparison was between two algorithms only the Wilcoxon test was used.

We also employed the statistics used in [41] to compare two learning algorithms across all data sets, namely, the win/draw/loss record. The win/draw/loss record presents three values, the number of data sets for which algorithm A obtained better, equal, or worse performance than algorithm B with respect to classification accuracy. We also reported the statistically significant win/draw/loss record, where a win or loss was only counted if the difference in values was determined to be significant at the 0.05 level by a paired t-test.

Tables 10 to 15 show the comparison among the methods. Columns labeled s present the win/draw/loss record, where the first value is the number of data sets for which $row < col$, the second is the number for which $row = col$, and the last is the number for which $row > col$. Columns labeled P_w present the results of Wilcoxon tests. For all methods, the Iman–Davenport test had a P-value of 0.000, showing significant differences among them.

As seen in the tables, for all groups except ME and related ensemble methods using Bigram-TF–IDF as a feature, at least one ensemble method has better comparative results than the base learner. Thus, we can conclude that ensemble methods are appropriate for sentiment classification.

Furthermore, some interesting phenomena were observed in the experiments. Among the three ensemble methods, Boosting had poor accuracy except when it used DT as the base learner. A potential explanation is because the BOW framework directly converts text information into space vectors, the space vectors contain many redundant and relevant features and some noise. Empirical and theoretical results have shown that Boosting is easily influenced by noisy data [3,13,51]. The second interesting thing is that ensemble methods using DT as the base learner all have better comparative results. This result is consistent with prior research [5,13,30,51] and

Table 12
Outcomes of Wilcoxon matched-pairs signed-ranks test (Unigram-TF–IDF).

	Bagging NB		Boosting NB		RS NB	
	s	P_w	s	P_w	s	P_w
NB	5/3/2	0.707	4/2/4	3.434**	1/4/5	5.328**
Bagging NB			4/3/3	3.816**	1/1/8	4.935**
Boosting NB					3/1/6	0.728
	Bagging ME		Boosting ME		RS ME	
	s	P_w	s	P_w	s	P_w
ME	4/1/5	1.647	0/7/3	9.038**	7/0/3	8.686**
Bagging ME			5/1/4	4.353**	7/0/3	8.571**
Boosting ME					8/1/1	11.238**
	Bagging DT		Boosting DT		RS DT	
	s	P_w	s	P_w	s	P_w
DT	10/0/0	18.146**	8/0/2	11.090**	9/0/1	15.448**
Bagging DT			1/0/9	9.835**	2/2/6	4.124**
Boosting DT					7/1/2	5.395**
	Bagging KNN		Boosting KNN		RS KNN	
	s	P_w	s	P_w	s	P_w
KNN	4/2/4	1.120	1/9/0	2.027*	9/1/0	19.895**
Bagging KNN			5/1/4	1.308	10/0/0	19.634**
Boosting KNN					9/1/0	19.903**
	Bagging SVM		Boosting SVM		RS SVM	
	s	P_w	s	P_w	s	P_w
SVM	6/1/3	3.007**	0/5/5	13.400**	6/0/4	6.080**
Bagging SVM			2/0/8	12.403**	6/1/3	3.781**
Boosting SVM				13.400**	8/0/2	13.386**

Note: Iman–Davenport test: 0.000. *P-values significant at alpha = 0.05. **P-values significant at alpha = 0.01.

Table 13
Outcomes of Wilcoxon matched-pairs signed-ranks test (Bigram-TP).

	Bagging NB		Boosting NB		RS NB	
	s	P_w	s	P_w	s	P_w
NB	5/1/4	1.826	4/2/4	1.806	1/3/6	3.037**
Bagging NB			5/1/4	0.351	2/2/6	3.246**
Boosting NB					2/3/5	3.125**
	Bagging ME		Boosting ME		RS ME	
	s	P_w	s	P_w	s	P_w
ME	5/0/5	4.430**	1/6/3	7.994**	3/2/5	0.643
Bagging ME			4/1/5	1.025	4/0/6	3.518**
Boosting ME					4/2/4	1.785
	Bagging DT		Boosting DT		RS DT	
	s	P_w	s	P_w	s	P_w
DT	9/0/1	16.971**	10/0/0	11.558**	9/1/0	15.654**
Bagging DT			1/0/9	8.325**	2/1/7	1.833
Boosting DT					8/0/2	6.787**
	Bagging KNN		Boosting KNN		RS KNN	
	s	P_w	s	P_w	s	P_w
KNN	4/3/3	2.799**	0/7/3	4.348**	9/1/0	20.014**
Bagging KNN			2/2/6	4.571**	10/0/0	19.539**
Boosting KNN					9/1/0	20.234**
	Bagging SVM		Boosting SVM		RS SVM	
	s	P_w	s	P_w	s	P_w
SVM	5/2/3	5.110**	2/4/4	9.468**	8/0/2	14.399**
Bagging SVM			2/1/7	10.636**	8/1/1	11.257**
Boosting SVM					8/0/2	16.747**

Note: Iman–Davenport test: 0.000. *P-values significant at alpha = 0.05. **P-values significant at alpha = 0.01.

Table 14
Outcomes of Wilcoxon matched-pairs signed-ranks test (Bigram-TF).

	Bagging NB		Boosting NB		RS NB	
	s	P_w	*s*	P_w	*s*	P_w
NB	4/4/2	0.492	3/1/6	1.617	2/3/5	2.760**
Bagging NB			3/1/6	1.970*	2/1/7	2.417*
Boosting NB					6/1/3	0.172
	Bagging ME		Boosting ME		RS ME	
	s	P_w	*s*	P_w	*s*	P_w
ME	5/0/5	5.528**	1/6/3	8.257**	3/2/5	1.316
Bagging ME			4/0/6	1.299	6/1/3	6.442**
Boosting ME					3/3/4	4.891**
	Bagging DT		Boosting DT		RS DT	
	s	P_w	*s*	P_w	*s*	P_w
DT	10/0/0	16.999**	10/0/0	13.505**	10/0/0	15.997**
Bagging DT			1/0/9	5.690**	3/3/4	0.916
Boosting DT					6/2/2	4.701**
	Bagging KNN		Boosting KNN		RS KNN	
	s	P_w	*s*	P_w	*s*	P_w
KNN	4/3/3	1.593	1/6/3	3.872**	9/1/0	21.072**
Bagging KNN			2/3/5	3.410**	10/0/0	20.579**
Boosting KNN					9/1/0	21.489**
	Bagging SVM		Boosting SVM		RS SVM	
	s	P_w	*s*	P_w	*s*	P_w
SVM	5/1/4	2.644**	2/4/4	10.145**	8/0/2	10.074**
Bagging SVM			2/1/7	9.825**	8/0/2	8.979**
Boosting SVM					8/0/2	14.093**

Note: Iman–Davenport test: 0.000. *P-values significant at alpha = 0.05. **P-values significant at alpha = 0.01.

Table 15
Outcomes of Wilcoxon matched-pairs signed-ranks test (Bigram-TF–IDF).

	Bagging NB		Boosting NB		RS NB	
	s	P_w	*s*	P_w	*s*	P_w
NB	5/2/3	0.902	4/1/5	0.739	0/4/6	5.265**
Bagging NB			4/1/5	0.387	0/3/7	4.662**
Boosting NB					3/1/6	3.606**
	Bagging ME		Boosting ME		RS ME	
	s	P_w	*s*	P_w	*s*	P_w
ME	3/1/6	9.318**	0/6/4	8.021**	3/2/5	0.122
Bagging ME			6/0/4	5.448**	6/0/4	6.209**
Boosting ME					5/0/5	3.318**
	Bagging DT		Boosting DT		RS DT	
	s	P_w	*s*	P_w	*s*	P_w
DT	10/0/0	16.425**	10/0/0	12.603**	10/0/0	15.465**
Bagging DT			1/2/7	6.083**	2/5/3	2.050*
Boosting DT					7/1/2	4.906**
	Bagging KNN		Boosting KNN		RS KNN	
	s	P_w	*s*	P_w	*s*	P_w
KNN	6/2/2	3.473**	1/6/3	1.980*	10/0/0	21.312**
Bagging KNN			1/4/5	4.786**	9/1/0	20.104**
Boosting KNN					10/0/0	21.469**
	Bagging SVM		Boosting SVM		RS SVM	
	s	P_w	*s*	P_w	*s*	P_w
SVM	6/3/1	7.611**	2/4/4	9.391**	8/1/1	14.287**
Bagging SVM			1/2/7	12.154**	8/0/2	10.191**
Boosting SVM					8/1/1	16.576**

Note: Iman–Davenport test: 0.000. *P-values significant at alpha = 0.05. **P-values significant at alpha = 0.01.

can explain why previous researchers were more likely to choose DT as the base learner to test and verify their ensemble methods. The third interesting thing is that Random Subspace has better comparative results when using DT, KNN, and SVM as the base learner, but it has the worst comparative result when using NB as the base learner. A possible explanation is that NB is sensitive to the feature set because the default random subspace rate is set to 0.5 in the experiments.

5.3. Analysis and discussion from the base learner perspective

The average accuracy of different methods across the ten datasets from the base learner perspective is shown in Figs. 8 and 9.

Firstly, as shown in Figs. 8 and 9, RS-SVM has the best average accuracy, i.e., 78.50%, 77.83%, and 75.64% when using Unigram feature sets and gets the best average accuracy, i.e., 75.23%, 74.99%, and 75.30% when using Bigram feature sets. These results further testify that among 20 classifiers, RS-SVM has a distinct comparative advantage for sentiment classification.

Secondly, when using Unigram feature sets, SVM and NB have better results in the base learner group, Bagging group, Boosting group, and Random Subspace group. When using Bigram feature sets, SVM, NB, and ME obtain better results. These results are consistent with previous research [1,29,43]. Moreover, they further verify why SVM, NB, and ME are the most commonly used machine learning methods for sentiment classification [1,28].

Thirdly, KNN and ensemble classifiers using KNN as the base learner all have the worst results in the different groups. Following KNN, DT and ensemble classifiers using DT as the base learner have the second worst results. This is also consistent with prior studies [1,15,28]. KNN and DT can be used as classifiers when there are relatively few features to consider; however they become difficult to manage for large numbers of features [15].

5.4. Analysis and discussion from the feature set perspective

To compare different weight calculation methods, the performance of Unigram-TP was set to the baseline. The average accuracy improvement of the other five methods was calculated as:

$$\text{Accuracy improvement} = \frac{\text{Average accuracy}_{\text{Unigram-TP}} - \text{Average accuracy}_{\text{Compared}}}{\text{Average accuracy}_{\text{Compared}}} \quad (5)$$

The results are shown in Figs. 10 to 14.

As shown in Figs. 10 to 14, the Unigram-TP is the best choice for NB, DT, KNN, SVM, and ensemble methods using them as the base learner. It is interesting that for ME and ensemble methods using ME as the base learner, the Unigram-TP is the worst choice. For them, the Bigram-TP is the better choice. These results are consistent with previous research [29,43]. Firstly, for the term present vs. frequency problem [28], the sentiment classification may not be highlighted through repeated use of the same terms. Our experimental results further support these analyses. Secondly, for the problem of whether higher-order N-gram are useful features, our experimental results show that Bigram yield better results only for the ME and ensemble methods using ME as the base learner. In addition, for the Camera, Laptop, and Radio datasets, the best sentiment classification results

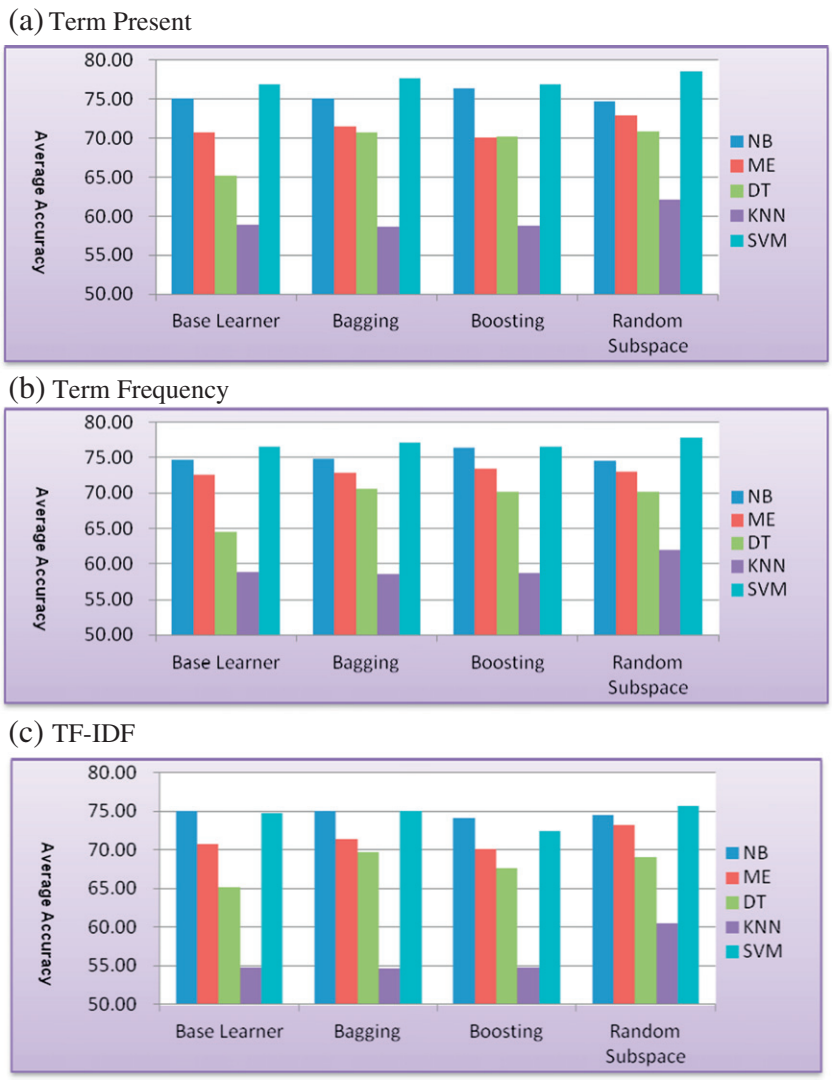

Fig. 8. Average accuracy of different methods (Unigram).

are from the Bigram feature set. As explanation in [28], this problem appears to be a matter of debate, and the choice depends on different classifiers and datasets.

6. Conclusions and future directions

The rise of social media has fueled interest in sentiment classification. Promptly and correctly classifying sentiment from the text has become an important task for individuals and companies. In this study, we empirically evaluated ensemble methods (Bagging, Boosting, and Random Subspace) for use in sentiment analysis. Ten public sentiment analysis datasets were investigated to verify the effectiveness of ensemble learning for sentiment analysis. Empirical results showed that ensemble methods can get better results than base learners. Among twenty methods, Random Subspace SVM had the best accuracy. All these results illustrate that ensemble learning methods can be used as a viable method for sentiment classification.

There are several future research directions for this study. Firstly, as the sentiment datasets are often imbalanced, large datasets should be collected to further validate the conclusions of the study. Secondly, feature set is one important factor for classification, but we only used bag-of-word feature sets in this research. In the next step, feature construction based on linguistics should be considered. Thirdly, as ensemble learning methods need a lot of computing time, parallel computing techniques should be explored to tackle this problem. Fourthly, a major limitation of ensemble learning methods is the lack of interpretability of the results: the knowledge learned by ensembles is difficult for humans to understand. Therefore improving the interpretability of ensembles is another important research direction.

Acknowledgments

The authors would like to thank the Editor-in-Chief and reviewers for their recommendation and comments. This work is partially supported by the National Natural Science Foundation of China (Nos. 71071045, 71131002, 71101042), Specialized Research Fund for the Doctoral Program of Higher Education (20110111120014), the China Postdoctoral Science Foundation (2011M501041, 2013T60611), Special Fund of AnHui Province Key Research Institute of Humanities and Social Sciences at Universities (SK2013B400), and Special Fund of Political Theory Research Center of HeFei University of Technology (2012HGXJ0392).

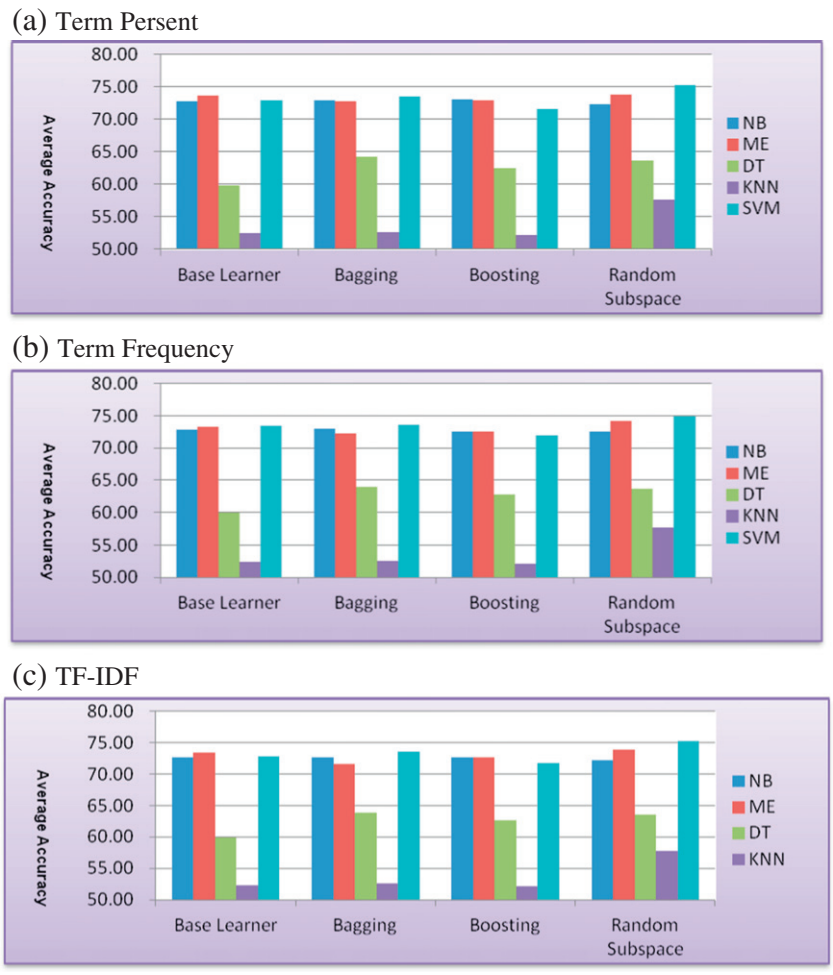

Fig. 9. Average accuracy of different methods (Bigram).

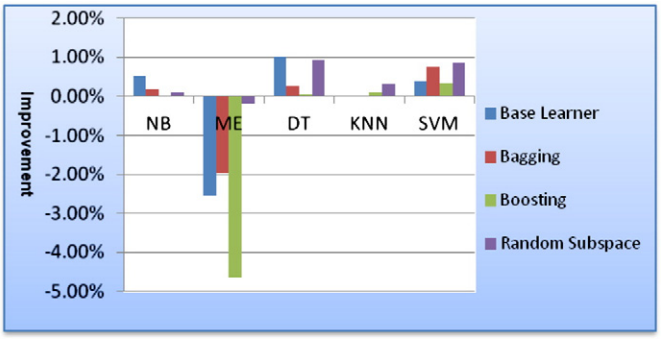

Fig. 10. Average accuracy improvement (Unigram-TP vs. Unigram-TF).

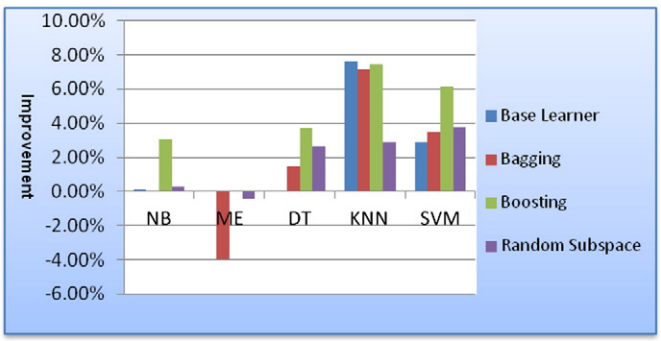

Fig. 11. Average accuracy improvement (Unigram-TP vs. Unigram-TF–IDF).

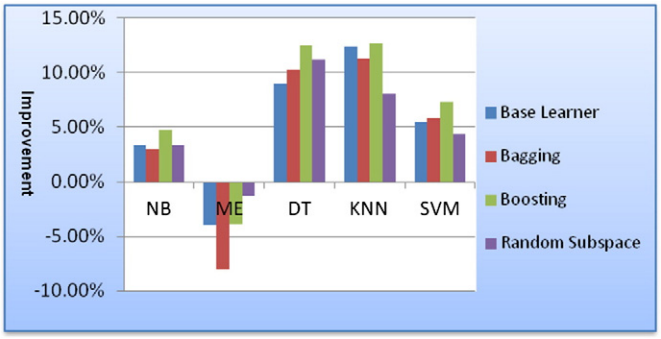

Fig. 12. Average accuracy improvement (Unigram-TP vs. Bigram-TP).

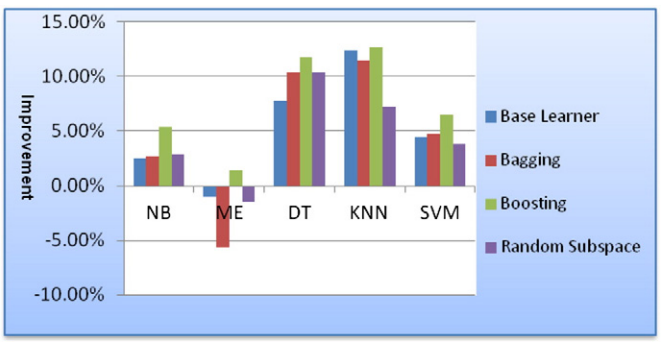

Fig. 13. Average accuracy improvement (Unigram-TP vs. Bigram-TF).

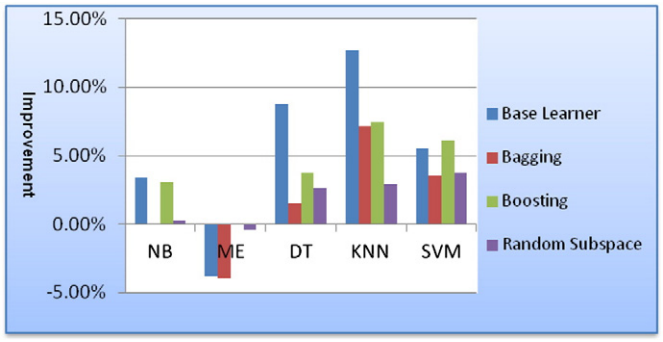

Fig. 14. Average accuracy improvement (Unigram-TP vs. Bigram-TF–IDF).

References

[1] A. Abbasi, H. Chen, A. Salem, Sentiment analysis in multiple languages: feature selection for opinion classification in web forums, ACM Transactions on Information Systems (TOIS) 26 (3) (2008) 12.
[2] A. Abbasi, H. Chen, S. Thoms, T. Fu, Affect analysis of web forums and blogs using correlation ensembles, IEEE Transactions on Knowledge and Data Engineering 20 (9) (2008) 1168–1180.
[3] E. Bauer, R. Kohavi, An empirical comparison of voting classification algorithms: bagging, boosting, and variants, Machine Learning 36 (1–2) (1999) 105–139.
[4] E. Boiy, M.-F. Moens, A machine learning approach to sentiment analysis in multilingual web texts, Information Retrieval 12 (5) (2009) 526–558.
[5] L. Breiman, Bagging predictors, Machine Learning 24 (2) (1996) 123–140.
[6] H. Chen, C. Yang, Special issue on social media analytics: understanding the pulse of the society, Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on 41 (5) (2011) 826–827.
[7] J. Chen, H. Huang, S. Tian, Y. Qu, Feature selection for text classification with Naïve Bayes, Expert Systems with Applications 36 (3) (2009) 5432–5435.
[8] T. Cover, P. Hart, Nearest neighbor pattern classification, Information Theory, IEEE Transactions on 13 (1) (1967) 21–27.
[9] Y. Dang, Y. Zhang, H. Chen, A lexicon-enhanced method for sentiment classification: an experiment on online product reviews, Intelligent Systems, IEEE 25 (4) (2010) 46–53.
[10] B.V. Dasarathy, B.V. Sheela, A composite classifier system design: concepts and methodology, Proceedings of the IEEE 67 (5) (1979) 708–713.
[11] K. Dave, S. Lawrence, D.M. Pennock, Mining the peanut gallery: opinion extraction and semantic classification of product reviews, Proceedings of the 12th International Conference on World Wide Web, (ACM, 2003), 2003, pp. 519–528.
[12] L. Delacroix, Longman Advanced American Dictionary, Pearson Education, Edinburgh, UK, 2007.
[13] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.
[14] T. Dietterich, Machine learning research: four current directions, AI Magazine 18 (4) (1997) 97–136.
[15] G. Forman, An extensive empirical study of feature selection metrics for text classification, Journal of Machine Learning Research 3 (2003) 1289–1305.
[16] N. García-Pedrajas, Constructing ensembles of classifiers by means of weighted instance selection, Neural Networks, IEEE Transactions on 20 (2) (2009) 258–277.
[17] L.K. Hansen, P. Salamon, Neural network ensembles, Pattern Analysis and Machine Intelligence, IEEE Transactions on 12 (10) (1990) 993–1001.
[18] V. Hatzivassiloglou, K.R. McKeown, Predicting the semantic orientation of adjectives, Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics, (Association for Computational Linguistics) 1997, pp. 174–181.
[19] T.K. Ho, The random subspace method for constructing decision forests, Pattern Analysis and Machine Intelligence, IEEE Transactions on 20 (8) (1998) 832–844.
[20] Y. Hu, W. Li, Document sentiment classification by exploring description model of topical terms, Computer Speech & Language 25 (2) (2011) 386–403.
[21] R.L. Iman, J.M. Davenport, Approximations of the critical region of the fbietkan statistic, Communications in Statistics—Theory and Methods 9 (6) (1980) 571–595.
[22] S.-M. Kim, E. Hovy, Determining the sentiment of opinions, Proceedings of the 20th international conference on Computational Linguistics, (Association for Computational Linguistics), 2004, p. 1367.
[23] P.C.R. Lane, D. Clarke, P. Hender, On developing robust models for favourability analysis: model choice, feature sets and imbalanced data, Decision Support Systems 53 (4) (2012) 712–718.
[24] W. Li, W. WANG, Y. CHEN, Heterogeneous ensemble learning for Chinese sentiment classification, Journal of Information & Computational Science 9 (15) (2012) 4551–4558.
[25] L. Liu, M.T. Zsu, Encyclopedia of Database Systems, Springer Publishing Company, Incorporated, 2009.
[26] B. Lu, B.K. Tsou, Combining a large sentiment lexicon and machine learning for subjectivity classification, Machine Learning and Cybernetics (ICMLC), 2010 International Conference on, (IEEE), 2010, pp. 3311–3316.
[27] B. Pang, L. Lee, A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts, Proceedings of the 42nd annual meeting on Association for Computational Linguistics, (Association for Computational Linguistics), 2004, p. 271.
[28] B. Pang, L. Lee, Opinion mining and sentiment analysis, Foundations and Trends in Information Retrieval 2 (1–2) (2008) 1–135.
[29] B. Pang, L. Lee, S. Vaithyanathan, Thumbs up?: sentiment classification using machine learning techniques, Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, (Association for Computational Linguistics, 2002, pp. 79–86.
[30] R. Polikar, Ensemble based systems in decision making, Circuits and Systems Magazine, IEEE 6 (3) (2006) 21–45.
[31] Morgan kaufmann, J.R. Quinlan, C4. 5: Programs for Machine Learning, , 1993.
[32] K. Sarvabhotla, P. Pingali, V. Varma, Sentiment classification: a lexical similarity based approach for extracting subjectivity in documents, Information Retrieval 14 (3) (2011) 337–353.
[33] R.E. Schapire, The strength of weak learnability, Machine Learning 5 (2) (1990) 197–227.
[34] Y. Su, Y. Zhang, D. Ji, Y. Wang, H. Wu, Ensemble learning for sentiment classification, Chinese Lexical Semantics, Springer, 2013, pp. 84–93.
[35] V.S. Subrahmanian, D. Reforgiato, AVA: adjective–verb–adverb combinations for sentiment analysis, Intelligent Systems, IEEE 23 (4) (2008) 43–50.
[36] T.T. Thet, J.-C. Na, C.S. Khoo, Aspect-based sentiment analysis of movie reviews on discussion boards, Journal of Information Science 36 (6) (2010) 823–848.
[37] K. Tsutsumi, K. Shimada, T. Endo, Movie review classification based on a multiple classifier, the 21th Pacific Asia Conference on Language, Information and Computation (PACLIC), 2007.
[38] P.D. Turney, Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews, Proceedings of the 40th annual meeting on association for computational linguistics, (Association for Computational Linguistics, 2002, pp. 417–424.
[39] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer, 2000.
[40] G. Wang, J. Ma, S. Yang, Igf-bagging: information gain based feature selection for bagging, International Journal of Innovative Computing, Information and Control 7 (11) (2011) 6247–6259.
[41] G.I. Webb, Multiboosting: a technique for combining boosting and wagging, Machine Learning 40 (2) (2000) 159–196.
[42] M. Whitehead, L. Yaeger, Building a general purpose cross-domain sentiment mining model, World Congress on Computer Science and Information Engineering, 2009 WRI, IEEE, 2009, pp. 472–476.
[43] M. Whitehead, L. Yaeger, Sentiment mining using ensemble classification models, Innovations and Advances in Computer Sciences and Engineering, Springer, 2010, pp. 509–514.
[44] F. Wilcoxon, Individual comparisons by ranking methods, Biometrics Bulletin 1 (6) (1945) 80–83.
[45] T. Wilson, J. Wiebe, R. Hwa, Recognizing strong and weak opinion clauses, Computational Intelligence 22 (2) (2006) 73–99.
[46] T. Windeatt, G. Ardeshir, Decision tree simplification for classifier ensembles, International Journal of Pattern Recognition and Artificial Intelligence 18 (5) (2004) 749–776.
[47] Morgan Kaufmann, I.H. Witten, E. Frank, M.A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, , 2011.
[48] R. Xia, C. Zong, S. Li, Ensemble of feature sets and classification algorithms for sentiment classification, Information Sciences 181 (6) (2011) 1138–1152.
[49] J. Yi, T. Nasukawa, R. Bunescu, W. Niblack, Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques, Data Mining, 2003. ICDM 2003. Third IEEE International Conference on, (IEEE, 2003), 2003, pp. 427–434.
[50] C. Zhang, D. Zeng, J. Li, F.Y. Wang, W. Zuo, Sentiment analysis of Chinese documents: from sentence to document level, Journal of the American Society for Information Science and Technology 60 (12) (2009) 2474–2487.
[51] Z.-H. Zhou, Ensemble Methods: Foundations and Algorithms, Chapman & Hall, 2012.

Gang Wang is an Associate Professor in the School of Management, HeFei University of Technology. He received his Ph. D. in the School of Management, Fudan University. His current research focuses on Data Mining And Business Intelligence, ensemble learning, and social network analysis.

Jianshan Sun is a Ph. D. candidate of joint training of City University of Hong Kong and University of Science and Technology of China. His current research interests include ensemble learning, and business intelligence.

Jian Ma is a Professor in the Department of Information Systems, City University of Hong Kong. He received his Doctor of Engineering degree in Computer Science from Asia Institute of Technology. Dr. Ma's research areas include decision and decision support systems, business intelligence, research information systems, research and innovation social networks. His past research has been published in IEEE Transactions on Engineering Management, IEEE Transactions on Education, IEEE Transactions on Systems, Man and Cybernetics, Decision Support Systems, Information and Management, and European Journal of Operational Research.

Kaiquan Xu is a Assistant Professor in the School of Business, Nanjing University. He has received his Ph.D. in Business Information Systems from the City University of Hong Kong. His research interests include business intelligence and analytics, and knowledge management.

Jibao Gu is a Professor in the School of Management, University of Science and Technology of China. His research interests include marketing and international financial.