Wang *et al. Hum. Cent. Comput. Inf. Sci. (2018) 8:17*
https://doi.org/10.1186/s13673-018-0140-y

Human-centric Computing and Information Sciences

RESEARCH

Open Access

Word clustering based on POS feature for efficient twitter sentiment analysis

Yili Wang[1], KyungTae Kim[2], ByungJun Lee[1] and Hee Yong Youn[2*]

*Correspondence: youn7147@skku.edu
[2] College of Software, Sungkyunkwan University, Suwon 440746, Korea
Full list of author information is available at the end of the article

Abstract

With rapid growth of social networking service on Internet, huge amount of information are continuously generated in real time. As a result, sentiment analysis of online reviews and messages has become a popular research issue [1]. In this paper a novel modified Chi Square-based feature clustering and weighting scheme is proposed for the sentiment analysis of twitter message. Along with the part of speech tagging, the discriminability and dependency of the words in the tagged training dataset are taken into account in the clustering and weighting process. The multinomial Naïve Bayes model is also employed to handle redundant features, and the influence of emotional words is raised for maximizing the accuracy. Computer simulation with Sentiment 140 workload shows that the proposed scheme significantly outperforms four existing representative sentiment analysis schemes in terms of the accuracy regardless of the size of training and test data.

Keywords: Sentiment analysis, Twitter classification, Part of speech training, Word clustering

Introduction

Recently, massive volume of data are generated and shared through internet [2–4]. There exist various forms with the data originated from internet, and especially text is quite popular for expressing and sharing information between individual users. Therefore, text classification has drawn increasing interests [5], which automatically processes and categorizes the text data into predefined categories using an analytical model constructed based on the training data [6]. Twitter is one of the most popular micro-blogging web platforms [7, 8], which has become a lode for text classification as over million active users send and receive about 500 million messages per day in a 140 character message called "tweet" [8, 9]. The "tweet" extracted from the twitter API has been widely adopted as the source data for sentiment analysis [7]. Employing various machine learning techniques, sentiment analysis classifies a twitter message into 'positive' or 'negative', and sometimes 'neutral'.

Typically, there exist two major approaches employed for sentiment analysis. The first one is to analyze the words-bag features of text with supervised machine learning algorithm [10]. In this approach a words-bag vector is established by filtering the words in the text, and the appearance of the words in the vector is regarded as the feature of the text [11]. The other approach is to establish a sentiment classifier based on the syntax

Springer Open

© The Author(s) 2018. This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

tree of sentence, which is constructed to indicate the relationship between the words by parsing the sentences [12–15]. Then the sentiment classifier is built based on the syntax relations, polarity, and features of the words [11]. There exist various challenges in sentiment analysis. The primary issue is the extraction of effective model. Typically, a machine learning algorithm is applied to the classification model extracted from the training dataset having manually tagged class labels [3]. Therefore, proper implementation of the classification model plays a crucial role in deciding the performance of sentiment analysis. Another issue is feature weighting. Assigning appropriate weight to relevant and discriminative features (attribute) is crucial to achieve the sentiment analysis of high accuracy using the classifier.

Among the feature weighting schemes proposed for sentiment analysis, the most widely used one is based on feature frequency (FF) due to the simplicity and effectiveness [10]. Here, the frequency of a word appearing in a document is utilized as the value of the feature of the document, and the highest value of them in the total documents is regarded as the feature value of the whole training data set. FF shows reasonable performance in many cases. However, if the feature values are uniformly distributed, it is difficult to properly analyze the feature information. The scheme based on document frequency (DF) effectively handles the issue of uniform distribution of the features. Here, the number of documents containing the target word is counted from the training dataset, which effectively represents the statistical information of the feature even the case of uniform distribution. The DF scheme has the advantage of simplicity and applicability to the training data of a huge volume at reasonable computational complexity [16]. However, rare words are treated as useless data, which degrades the performance of sentiment analysis [17]. Part of speech-based weighting (PSW) [18] is a recently proposed feature weighting scheme for twitter sentiment analysis, which is a kind of word frequency (WF)-based approach considering the frequency of unique word in each category. The relevance of the word among the training dataset is also considered. As the weights for the words are set empirically, however, its performance may not be robust. The term frequency and inverse document frequency (TF–IDF) [19, 20] is a commonly adopted feature weighting scheme owing to its efficiency and robustness. It assumes that importance of a word is highly dependent on its frequency of occurrences in the document and the ratio of the total number of documents to the number of documents containing the word. It is effective in measuring the importance of the words among the documents of training dataset, which greatly increases the accuracy of sentiment analysis. However, exaggeration of the dimensionality still exists, which treats the size of features as the volume of the words of the entire training dataset. This causes big computation overhead of weighting all the words [21].

Although a variety of feature weighting schemes for sentiment analysis have been developed, few of them investigate the relevancy between the clustered features and the class in assigning the weights. In this paper a novel feature weighting approach is proposed, which is inspired by the expectation that enhancing the strength of the words of strong discriminability may allow higher accuracy of sentiment analysis [22, 23]. In the proposed scheme the words of same type of POS feature of the classes are clustered into predefined sets. The dependency between the clustered set and the corresponding class is measured by the modified Chi Square technique [24]. It serves as a criterion for weighting the emotional words along with the discriminability of the words. The proposed scheme is extensively evaluated by computer simulation and compared with other schemes proposed for twitter sentiment analysis using the workloads of Sentiment 140 [25]. The simulation results reveal that the proposed scheme greatly improves the accuracy of the existing schemes. The main contributions of the paper are summarized below,

- A novel feature reduction method is proposed to reduce the dimensionality (size of features) [26], which omits irrelevant data in classifying the training dataset into a small number of features and achieves a reasonable computational complexity when weighting the words [27, 28].
- A modified Chi Square method is employed since the conventional Chi Square method suffers from the shortcoming of overemphasizing the role of the words of low frequency and measuring the class of a word based on DF. Therefore, WF is proposed to serve as the input to the Chi Square method to avoid such weakness. In addition, the traditional Chi Square method investigates the independency between a single feature and the class in the text classification. In the proposed scheme the dependency of the clustered feature set on the class is explored. The importance of the words is also characterized by the dependency derived from the modified Chi Square method.
- A novel composite feature weighting technique is proposed, which considers the dependency derived using the modified Chi Square technique and discriminability of the clustered feature set. In addition, the influence of the dependency to the weighting is also taken into account. Meanwhile, the importance of the words of strong discriminability is emphasized in the weighting process so that they can take more significant role in the sentiment analysis.

The rest of the paper is organized as follows: "Related work" section discusses the background of sentiment analysis. In "The proposed scheme" section the proposed scheme is presented, and its performance is evaluated in "Performance evaluation" section. Finally, the paper is concluded in "Conclusion" section.

Related work

Naive Bayes classifier (NBC)

NBC is commonly employed for text classification due to its robust performance for various data, especially for high dimensional text data. It is a probability-based classifier employing the Baye's theorem with the assumption of naïve independency between the predictors [29]. Here the properties of each predictor are analyzed to contribute the probability of the category of each predictor to its class. A classifier is constructed based on Bayes theorem of Eq. (1) [30]. With NBC the influence of predictor_x on given class_c is estimated, assuming that the predictors are independent with each other.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \tag{1}$$

Here $P(c|x)$ is the probability of class_c given predictor_x, which is called the posterior probability. $P(x|c)$ is the probability of predictor_x given class_c. $P(c)$ is the probability of class_c to be true, which is called the prior probability of class_c. $P(x)$ is the prior probability of predictor_x. With n predictors, $P(x|c)$ is defined as,

$$P(x|c) = \prod_{k=1}^{n} P(x_k|c) \tag{2}$$

The selective Bayes classifier (SBC) is an enhancement of NBC, which displays good performance when redundant attributes exist. With SBC highly correlated redundant attributes are excluded if the assumption of attribute independency is taken. In [31] greedy search is performed to select all the subsets of the attributes using the forward selection technique, which raises the accuracy of the classifier obtained from the training set.

Twitter sentiment analysis

Sentiment analysis involves language processing, text classification, and computational linguistics to extract emotional information from the source data. It is broadly employed to review the social media used in various fields such as marketing and customer service [32]. Typically, the intention of sentiment analysis is to estimate the mood of the user concerning the target object, and the basic task is to determine the polarity of the given text [32]. The approaches employed for sentiment analysis is roughly categorized into two types; machine learning-based and lexicon-based. With the machine learning-based approach the sentiment classifier is trained using a machine learning algorithm [1]. The lexicon-based approach focuses on the evaluation of the polarity of the text using the lexicons collected from various sources such as MPQA lexicon [33], WordNet [34] and SentiWordNet [35]. The machine learning-based approach is commonly adopted for twitter sentiment analysis, which is a representative binary classifier categorizing the target text into positive or negative. The basic structure of twitter sentiment analysis is shown in Fig. 1.

Recent studies of twitter sentiment analysis focus on usage of various feature sets and methods [36–39]. In [40], the emotional state of tweets is visualized into specific feelings such as sadness, joy, and anger by employing the theory of Naïve Bayesian. SVM and MaxEntropy classifiers are used as competitors to compare the performance. In [41], the authors analyze the emoticons of sport fans using a lexicon-based approach. In [42], the prediction of stock market was analyzed by SVM approach. Tweets derived from the University and financial companies are utilized as source dataset in the performance evaluations. The results proved that SVM achieved best performance compared to KNN and Naïve Bayes classifiers. In [43], a hybrid approach combining several classifiers is investigated. Various cross-validated experiments were conducted, and the results reveal that the hybrid approach greatly improves the accuracy of classification.

Feature selection is one of the key steps in data pre-processing employed to maximize the performance of text classification, and it utilizes a machine learning technique [44]. It eliminates irrelevant or redundant attributes from the original feature space, and

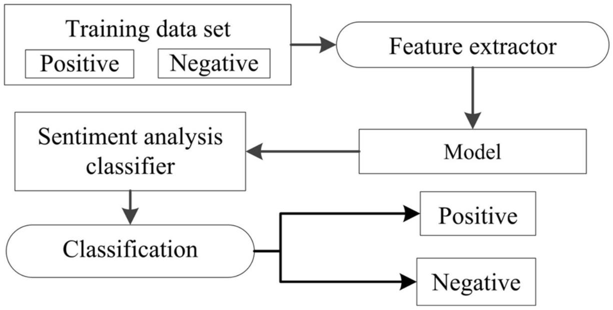


Fig. 1 The structure of twitter sentiment analysis

selects a relevant subset based on the target evaluation criterion to reduce the complexity of the analysis [2, 45]. Twitter is a popular online social networking service (SNS) platform that enables the users to show their thought or opinion in a 140-character message. Sentiment 140 lets the users discover the sentiment on people, product, etc. on Twitter. It also provides the APIs for analyzing the tweets, and supports the integration of sentiment analysis classifier with other personal site or platform [25].

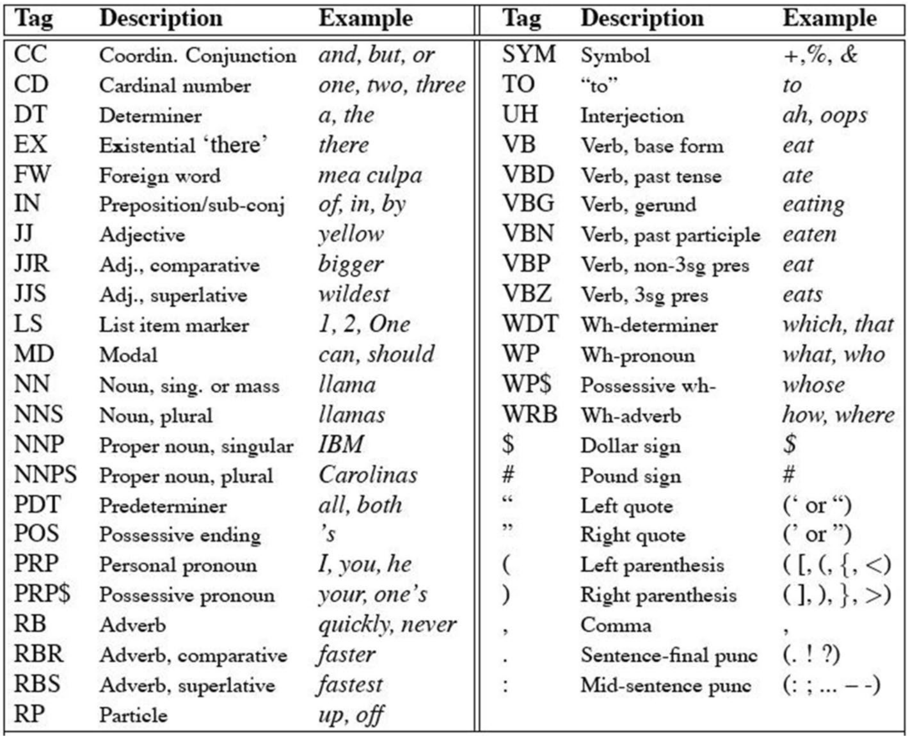
Part of speech

The part of speech (POS) tagging is a method of splitting the sentences into words and attaching a proper tag such as noun, verb, adjective and adverb to each word based on the POS tagging rules [46]. Figure 2 lists the POS tag, and Fig. 3 shows three examples of tagging [47]. POS tagging has been widely used in various tasks including text classification, speech recognition, automatic machine translation, and so on. A variety of POS taggers are available for English such as Brill tagger, Tree tagger, and CLAWS tagger. The POS tagging operation consists of two stages, training stage and tagging stage, which are shown in Figs. 4 and 5, respectively.

In the training stage, the corpus is employed to supply words in different context environments, and the contextual information is used as a clue to construct the rules required to decide the lexical classes of the words. Then the most likely tag for a word is selected by calculating the probability of the appearance of the context of the word and its immediate neighbors in the tagging stage [48].

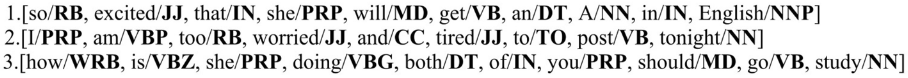
Feature weighting

In sentiment analysis the training data are classified into features (attributes) based on the content, and then weights are assigned to the features to distinguish their importances. Various feature weighting schemes have been proposed, and the commonly used one is term frequency and inversed document frequency (TF–IDF). TF–IDF consists of two parts, term frequency and inverse document frequency. Term frequency, $tf(w,d)$, represents the frequency of word_w appearing in document_d. The inverse document frequency, $idf(w,D)$, is a measurement showing how much information word_w offer for document_d. It is achieved by dividing the total number of documents by the number of documents word_w appears, and then taking the logarithm as,



Tag	Description	Example	Tag	Description	Example
CC	Coordin. Conjunction	*and, but, or*	SYM	Symbol	+,%, &
CD	Cardinal number	*one, two, three*	TO	“to”	*to*
DT	Determiner	*a, the*	UH	Interjection	*ah, oops*
EX	Existential ‘there’	*there*	VB	Verb, base form	*eat*
FW	Foreign word	*mea culpa*	VBD	Verb, past tense	*ate*
IN	Preposition/sub-conj	*of, in, by*	VBG	Verb, gerund	*eating*
JJ	Adjective	*yellow*	VBN	Verb, past participle	*eaten*
JJR	Adj., comparative	*bigger*	VBP	Verb, non-3sg pres	*eat*
JJS	Adj., superlative	*wildest*	VBZ	Verb, 3sg pres	*eats*
LS	List item marker	*1, 2, One*	WDT	Wh-determiner	*which, that*
MD	Modal	*can, should*	WP	Wh-pronoun	*what, who*
NN	Noun, sing. or mass	*llama*	WP$	Possessive wh-	*whose*
NNS	Noun, plural	*llamas*	WRB	Wh-adverb	*how, where*
NNP	Proper noun, singular	*IBM*	$	Dollar sign	$
NNPS	Proper noun, plural	*Carolinas*	#	Pound sign	#
PDT	Predeterminer	*all, both*	“	Left quote	(‘ or “)
POS	Possessive ending	*’s*	”	Right quote	(’ or ”)
PRP	Personal pronoun	*I, you, he*	(	Left parenthesis	([, (, {, <)
PRP$	Possessive pronoun	*your, one’s*	)	Right parenthesis	( ],), }, >)
RB	Adverb	*quickly, never*	,	Comma	,
RBR	Adverb, comparative	*faster*	.	Sentence-final punc	(. ! ?)
RBS	Adverb, superlative	*fastest*	:	Mid-sentence punc	(: ; ... – -)
RP	Particle	*up, off*			

Fig. 2 The POS tag



1.[so/**RB**, excited/**JJ**, that/**IN**, she/**PRP**, will/**MD**, get/**VB**, an/**DT**, A/**NN**, in/**IN**, English/**NNP**]
2.[I/**PRP**, am/**VBP**, too/**RB**, worried/**JJ**, and/**CC**, tired/**JJ**, to/**TO**, post/**VB**, tonight/**NN**]
3.[how/**WRB**, is/**VBZ**, she/**PRP**, doing/**VBG**, both/**DT**, of/**IN**, you/**PRP**, should/**MD**, go/**VB**, study/**NN**]

Fig. 3 Three examples of POS tagging

$$\mathrm{idf}(w, D) = \log \frac{N}{\left|\left\{d \in D{:}w \in d\right\}\right|} \tag{3}$$

The TF–IDF value is then obtained by *TF–IDF*(*w*,*d*,*D*) = *tf*(*w*,*d*)·*idf*(*w*,*D*). A high TF–IDF value of a word denotes a large frequency in few documents, and a small frequency of the documents containing the word in the entire set of the documents. On the contrary, a low value indicates that the word appears evenly in every document. The TF–IDF is useful for selecting the words important for a document and evicting common words [49]. FF is a popular feature weighting scheme because of its simplicity and efficiency, which expresses a document as a vector of features. The method utilizes the frequency of a word appearing in a certain document as the value of the feature of the document [49]. DF is another important feature weighting method used in a variety of applications of text classification and other related tasks, which counts the number of documents that the target word_*w* appears within the entire documents. Only the words of a high DF value are kept which is represented as,

$$\mathrm{DF}(w_i,C_k) = p(w_i|C_k) \tag{4}$$

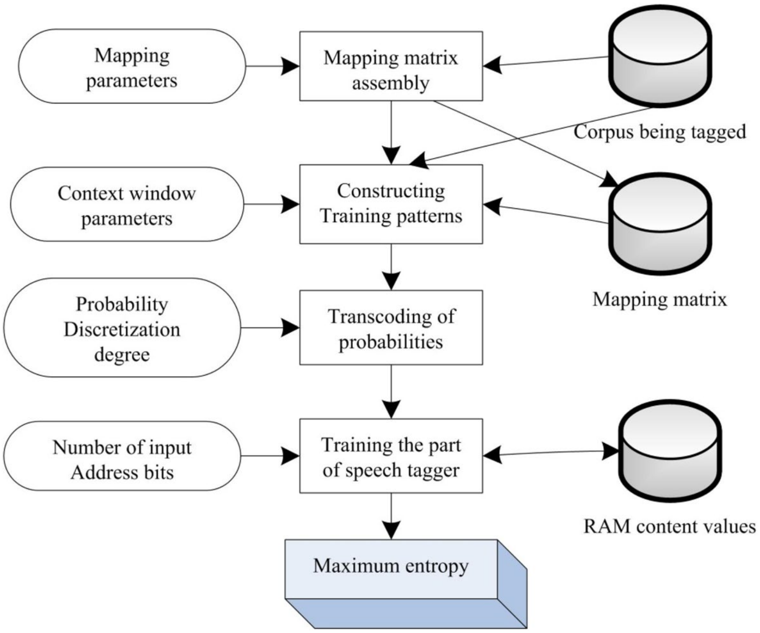

Mapping parameters
Mapping matrix assembly
Corpus being tagged
Context window parameters
Constructing Training patterns
Mapping matrix
Probability Discretization degree
Transcoding of probabilities
Number of input Address bits
Training the part of speech tagger
RAM content values
Maximum entropy

Fig. 4 The Training stage in POS tagging

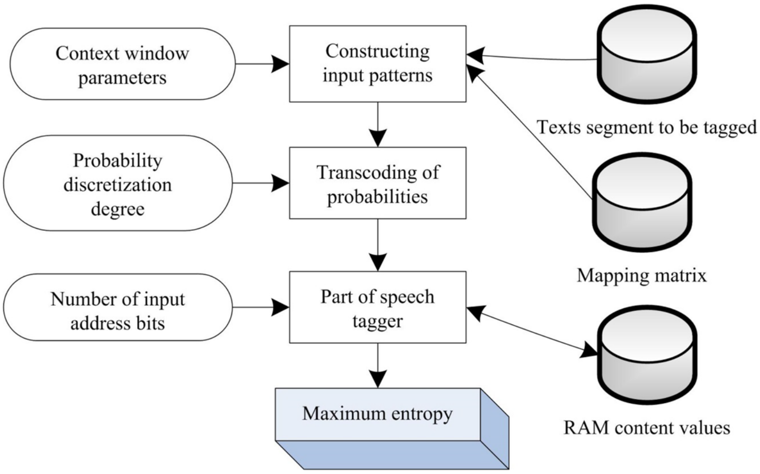

Context window parameters
Constructing input patterns
Texts segment to be tagged
Probability discretization degree
Transcoding of probabilities
Mapping matrix
Number of input address bits
Part of speech tagger
RAM content values
Maximum entropy

Fig. 5 The tagging stage in POS tagging

Part of speech-based weighting (PSW) is a recently proposed feature weighting scheme for improving the accuracy of twitter sentiment analysis. The method utilizes POS tagger, and the words are classified into three predefined subclasses as shown in Table 1.

The importance of a word is measured based on its POS tag. Refer to Table 1. The word of *Adverb*, *Adjective* and *Verb* with the corresponding POS tags are regarded as

Table 1 The subclasses with POS tag

Subclass	Property	POS tag
Emotion	Adverb, adjective, verb	*JJ,JJR,…,RB,RBR,…,VB,VBD,…*
Normal	Norm	*NN,NNS,NNP,NNPS…*
Remain	Remaining	*Remaining*

related to emotion, and thus retained in the Emotion subclass. In addition, a weight value, $wt_{i,j}$ ($j = 1, 2, 3$), is assigned to reflect the importance of the words.

$$wt_{i,j} = \begin{cases} x \cdot f_i, & f_i \geq E\left[F_j\right] \\ f_i, & f_i < E\left[F_j\right] \end{cases} \tag{5}$$

Here f_i is the frequency of word_*i* appearing in the training dataset, and x is a constant factor used to adjust the degree of influence of the words of different property in deciding the sentiment. It is 2, 1.5, and 1 for the emotion, normal and remain subclass, respectively. $E[F_j]$ is obtained as follows, where F_j($j = 1, 2, 3$) represents the subclass of Emotion, Normal, and Remain, respectively.

$$E\left[F_j\right] = \sum_{i=1}^{n} f_i p_i, \ \ p_i = \frac{f_i}{\sum_{i=1}^{n} f_i} \quad \left(j = 1, 2, 3\right) \tag{6}$$

The proposed scheme

Basic operation

The overall operation flow of the proposed scheme is as follows. For the sentences of twitter, POS tagging is firstly performed. Some sentences are selected as training data set based on the criteria, and then categorized into two classes, positive and negative according to their polarity. The words in the classes are clustered using their POS tags. A weight value is then assigned to every word based on the dependency and word discriminability of the clustered feature set to which the word belongs. When the training stage is over, a table of statistical data is obtained. The sentiment of the sentences of twitter in the test document is judged based on the statistics table. The overall operation flow of the proposed scheme is depicted in Fig. 5. Generally, the objective of the proposed scheme is to reinforce the strength of emotional words through the weighting and make them more influential in sentiment analysis, where the dependency among the cluster feature sets and classes serves as a criterion for the weighting. The detail implementation is presented in the next subsection (Fig. 6).

Preprocessing

Sentiment analysis mainly depends on the availability of initial corpus, $\Phi = \{d_1,\ldots, d_{|\Phi|}\}$, $d_i = \{s_1,\ldots,s_{|S|}\}$ and predefined class, $C = \{c_1,\ldots,c_{|C|}\}$. Here d_i represents a document consisting of $|S|$ sentences out of $|\Phi|$ documents of original corpus and $|C|$ classes. Firstly, the components of Φ are classified into the predefined set of categories, C. The task can be formalized as a function Ψ: $\Phi \times C \rightarrow \{N,P\}$, which

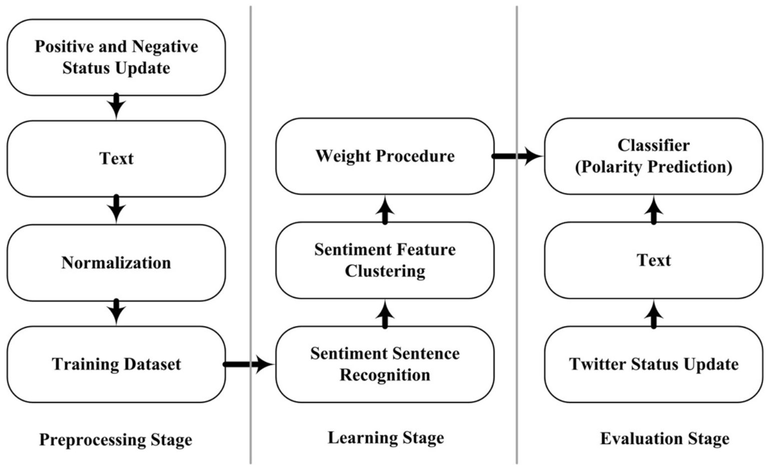

Positive and Negative Status Update
Text
Normalization
Training Dataset
Preprocessing Stage
Weight Procedure
Sentiment Feature Clustering
Sentiment Sentence Recognition
Learning Stage
Classifier (Polarity Prediction)
Text
Twitter Status Update
Evaluation Stage

Fig. 6 The operation flow of the proposed scheme

characterizes the process of classification. Specifically, if $\Psi(d_i,c_j)=N$, document_d_i is a negative dataset of c_j. Otherwise, if $\Psi(d_i,c_j)=P$, d_i is positive [50]. Then the initial corpus Φ is classified into two class sets, D_{neg} and D_{pos} ($D_{neg} \cup D_{pos} = \Phi$), $D_{neg} = \{d_1,\ldots,d_{sn}\}$, $D_{pos} = \{d_1,\ldots,d_{sp}\}$, $sn + sp = |\Phi|$, where sn and sp are the size of documents consisting of negative and positive sentences, respectively. Then every sentence in D_{neg} and D_{pos} is parsed through POS tagger, and every word of the sentence is assigned a corresponding POS tag serving as its feature. Refer to the example of Fig. 3.

Dimensionality reduction by feature selection

The sentences of twitter in the training data are classified into positive and negative sentences, while they can also be classified into subjective or objective. Note that emotional words in a sentence is important in judging the sentiment of the sentence. Therefore, removing the sentences having few emotional words can improve the accuracy of sentiment analysis. In this paper, *Adverb*, *Adjective* and *Verb* are regarded as emotional features important in deciding the sentiment. The sentence containing less than two types of emotional feature is regarded as unrelevant to sentiment analysis, and thus removed from the training data set [51]. In the example of Fig. 3, even though the meaning of the third sentence seems negative, it is not used for training because only one type of emotional feature of *Verb* appears in the sentence.

Feature clustering

The unigram feature extractor is utilized to retrieve the features from the tweets due to its simplicity and efficiency, which treats each unique word in the training dataset as a unit representing separate features [52]. Therefore, the set of document-built classes, D_α, is expressed as feature-based array consisting of unique words excluding the stop words. It is represented as [53], $D_{i,\alpha} = \{w_1, w_2,\ldots, w_{|w|} |\ \alpha = neg^{\vee} pos\}$, where w_i is a unique word occurring in the class set, $D_{i,\alpha}$. It is expressed as, $w_i = \{(f_i, t_i, D_{i,\alpha}, NW_i) |\ \alpha = negvpos\}$. Here f_i is the number of occurrences of unique word, w_i, in the entire documents of $D_{i,\alpha}$ with its POS feature tag, t_i. NW_i is the weighted frequency of w_i reflecting

Table 2 The feature set in the proposed scheme

Feature Set	Property	POS tag
Emotional	Adverb, adjective, verb	*JJ,JJR,… ,RB,RBR,… ,VB,VBD,…*
Normal	Remaining	*Remaining*

Table 3 The words and their frequencies

Term	$D_{i,neg}$	$D_{i,pos}$
w_1	$f_{1,neg}$	$f_{1,pos}$
w_2	$f_{2,neg}$	$f_{2,pos}$

the importance of the word as discussed in the following subsection. Different from the existing schemes counting the term frequency in every document and choosing the largest value to represent the feature of the document, the frequency of the words of the documents is computed with the class dataset to avoid the problem of exaggerating the role of low-frequency terms [22, 49]. A novel feature clustering method is proposed to aggregate the words of same POS features of $D_{i,neg}$ and $D_{i,pos}$ into the clustered feature set, C_E and C_N, as follows [54–56]. Here C_E is the clustered set of emotional feature maintaining the words of the POS tag of *Adverb* (*JJ*, *JJR*, *JJS* in Fig. 1), *Adjective* and *Verb* in $D_{i,\alpha}$. C_N serves as normal feature set keeping the words of remaining tags [57]. The detail classification is shown in Table 2. This process is formulated as, $D_{i,\alpha} = \{C_E, C_N \mid \alpha = neg^{\vee}pos\}$, $C_E = \{w_1, \ldots, w_p\}$, $C_N = \{w_1, \ldots, w_q\}$, $p + q = |w|$.

Measuring importance

The emotional words are classified into the clustered feature set, C_E. Here it is crucial to reflect the importance of the words to decide whether the tagged emotional words are actually important to the class or not [58]. Typically, word discrimination (*WD*) is applied to measure how much discriminative information a word owns with respect to the class [59]. The importance of a word for the class is quantified as, $WD_{i,\mu} = f_{i,\mu} - f_{i,v}$, $\mu \neq v$. Where $WD_{i,\mu}$ represents the *WD* of word_*i* of class_μ against class_v, which is measured by the difference in the frequency of word_*i* appearing in class_μ, $f_{i,\mu}$, and that in class_v, $f_{i,v}$. Intuitively, the word of high *WD* is regarded as important to the class as it contains a strong flavor on the class differentiating from other classes. This in turn greatly facilitates the judgement of the sentiment of the sentences. If $f_{i,\mu} > 0$, word_*i* has positive correlation with class_μ. Otherwise, it is deemed unrelated to class_μ. For instance, assume that two words, w_1 and w_2 coexist in $D_{i,neg}$ and $D_{i,pos}$ with the frequency listed in the Table 3.

Here $f_{1,neg}$ is the frequency of w_1 appearing in $D_{i,neg}$ and so on. *WD* of w_1 and w_2 of $D_{i,neg}$ is calculated as, $WD_{1,neg} = f_{1,neg} - f_{1,pos}$, $WD_{2,neg} = f_{2,neg} - f_{2,pos}$. If $(f_{1,neg} - f_{1,pos}) > 0$, w_1 is regarded as an essential word of $D_{i,neg}$ instead of $D_{i,pos}$. Meanwhile, if $WD_{1,neg} > WD_{2,neg}$, w_1 is regarded as the word containing more information on $D_{i,neg}$ than w_2 for the analysis of the sentiment. Recall that two feature sets, C_E and C_N, were built considering the POS feature of every word in $D_{i,\alpha}$, where C_E retains the emotional words and C_N the others.

Intuitively, the emotional words are expected to carry more information regarding the polarity of the sentiment compared to other words [10]. In addition, emotional words are not likely to occur in more than one class owing to its strong discriminability. For example, "*Like*" is an emotional word representing positive sentiment, and thus $WD_{'Like',pos}$ will be much larger than $WD_{'Like',neg}$. Since the emotional words are classified into C_E based on the POS tag, the mean value of WD of the words in C_E will be greater than that of C_N. If p and q are the number of unique words of C_E and C_N in $D_{i,\alpha}$, respectively,

$$\sum_{i=1}^{p} \frac{WD_{i,\alpha}}{p} > \sum_{j=1}^{q} \frac{WD_{j,\alpha}}{q}, \alpha = neg \vee pos \tag{7}$$

Measuring dependency

The conventional Chi Square method is commonly used for feature selection, which ranks the words and selects the word of the highest x^2 value. Meanwhile, it suffers from the weakness of overemphasizing low frequency words as only DF is considered without WF. Different from DF, WF is the frequency a word appears in the entire dataset, and it is expressed as,

$$WF_{w_i} = \sum_{i=1}^{d} f_{w_i} \tag{8}$$

WF measures the importance of a word in the whole data space. With the traditional Chi Square method the uniformly distributed words are deemed to best represent the class. Specifically, the method is based on the intuition that the optimal words for a specific class are the ones distributed most evenly among the documents of the training dataset [50]. Therefore, only the evenly distributed words of high DF and low WF value are selected from the dataset as the features representing the class, while the words of low DF and high WF value are discarded as they may cause inefficiency. However, some words of low DF and high WF might also be important and thus need to be reserved. Especially, the emotional words are regarded as rare with the conventional Chi Square method because they usually have uneven distribution. They also have high WF and low DF value [22]. By incorporating WF as an input to the Chi Square method, the dependency of C_E on $D_{i,\alpha}(\alpha = neg \vee pos)$ is measured as follows. Firstly, the hypothesis statements are set up as follows.

*Null Hypothesis*_1: C_E and $D_{i,\alpha}$ are independent.

*Alternative Hypothesis*_1: C_E and $C_{i,\alpha}$ are dependent.

Then contingency table for $r{\cdot}c$ is constructed as in Table 4, where r and c are the number of rows and columns, respectively. The Chi Square value, χ^2, is calculated using

Table 4 Different cases of summing the frequencies

Feature selection	$\in D_{i,a}$	$\notin D_{i,a}$	Sum
Containing C_E	A	B	$A+B$
Not Containing C_E	C	D	$C+D$
Sum	$A+C$	$B+D$	N

$\chi^2=\sum(O-E)^2/E$. Here O is the observed frequency and E is the expected frequency under the hypothesis. Table 4 lists different cases of feature selections with respect to C_E.

In Table 4, A is the sum of the WF of the words in C_E which is expressed as $A = \sum_{i=1}^{p} WF_i$, where p is the number of words in C_E. Similarly, the value of C is calculated as $C = \sum_{i=1}^{q} WF_i$ with q as the number of words in C_N. B is the case for the opposite class which is represented as,

$$B = \sum_{i=1}^{z} WF_{w_i}, \left\{ w_i \in \underset{C_E \in D_{i,\alpha}}{C_E} \middle| \Psi\left(d_j, c_i\right) = \overline{\alpha},\, D_{i,\alpha} \cup D_{i,\overline{\alpha}} = \Phi,\, z \le q \right\} \tag{9}$$

where z is the number of words of C_E in class $D_{i,\alpha}$ appearing in $D_{i,\bar{\alpha}}$. As the number of words in $D_{i,\alpha}$ is q, $z \le q$. D is calculated as,

$$D = \sum_{i=1}^{p'+q'} WF_{w_i} - B \tag{10}$$

where p' and q' are the number of words in C_E and C_N in $D_{i,\bar{\alpha}}$, respectively. For example, if the dependency of $D_{i,neg}$ with its corresponding clustered feature set, C_E, is measured, A is the total WF of the words in C_E of $D_{i,neg}$ and B is the sum of WF that the words of C_E appear in $D_{i,pos}$. C and D are obtained by subtracting the total WF of $D_{i,neg}$ and $D_{i,pos}$ from A and B, respectively. The expected frequency of the words of C_E belonging to $D_{i,j}$ is obtained as, $E_{11}=(A+C)\cdot(A+B)/N$. Here E_{11} is the expectation of A in the first row and first column of Table 4, and the deviation between the expectation is calculated as, $D_{11}=(A-E)^2/E_{11}$. Similarly, the other values are obtained as follows.

$$E_{12} = (B+D) * \frac{(A+B)}{N},\ D_{12} = \frac{(B-E_{12})^2}{E_{12}}$$

$$E_{21} = (A+C) * \frac{(C+D)}{N},\ D_{21} = \frac{(C-E_{21})^2}{E_{21}}$$

$$E_{22} = (B+D) * \frac{(C+D)}{N},\ D_{22} = \frac{(D-E_{22})^2}{E_{22}}$$

And then, C_E and $D_{i,\alpha}$ are derived as

$$\begin{aligned}\chi^2\left(C_E, D_{i,\alpha}\right) = D_{11}+D_{12}+D_{21}+D_{22} = \left(\frac{A^2}{E_{11}}+\frac{B^2}{E_{12}}+\frac{C^2}{E_{21}}+\frac{D^2}{E_{22}}\right) \\ -2(A+B+C+D)+(E_{11}+E_{12}+E_{21}+E_{22})\end{aligned} \tag{11}$$

$$= \frac{N*(AD-BC)^2}{(A+B)(A+C)(B+D)(C+D)} \tag{12}$$

In addition, the value of $\chi^2(C_N, D_{i,\alpha})$ can also be computed by constructing the table similar to Table 4, and the hypotheses is shown below.

Null Hypothesis_2: C_N and $D_{i,\alpha}$ are independent.

Alternative Hypothesis_2: C_N and $D_{i,\alpha}$ are dependent.

Weighting of the word

Word weighting is performed based on the importance of the word in the training dataset. The proposed weighting scheme considers the dependency of the clustered feature set with the class, which is measured by the value of χ^2. The greater the value, the stronger the dependency. For measuring the dependency, the critical value (CV) of Chi Square is given. In this paper 95% is taken as a metric for the measurement which indicates the null hypothesis is wrong with the probability of 0.95 or more. CV is computed as 3.84 with one degree of freedom [$\mathrm{DF} = (r-1)\cdot(c-1)$] based on the cumulative distribution function of Chi Square expressed as [60],

$$F(cv;k) = \frac{\gamma\left(k/2,cv/2\right)}{\Gamma\left(k/2\right)} \tag{13}$$

Here k is the degree, γ is incomplete gamma function, and Γ is gamma function represented as,

$$\Gamma(\alpha) = \int_0^{\infty} t^{\alpha-1}e^{-t}dt \tag{14}$$

The probability density function (PDF) of χ^2 distribution, $f(\chi^2)$, is drawn in Fig. 7, which shows a two-sided test of χ^2 distribution with the CV of 3.84. If the given χ^2 value is greater than the CV, the null hypothesis would be rejected. The tested clustered feature set is regarded as dependent on the class. Otherwise, they are treated as independent with each other. The CV of χ^2 is set high to ensure the reliability of the dependency. The region for the probability of one degree of freedom with the CV is marked with slashed lines.

Then $\chi^2(C_E,D_{i,\alpha})$ is compared with the CV. The *Null Hypothesis*_1 is rejected if $\chi^2(C_E,D_{i,\alpha}) > \mathrm{CV}$, and *Alternative Hypothesis*_1 is chosen which indicates that C_E is highly dependent on $D_{i,\alpha}$ with the probability of 95%. Otherwise, they are regarded as independent from each other. Similarly, $\chi^2(C_N,D_{i,\alpha})$ is compared with the CV, and the *Null Hypothesis*_2 or *Alternative Hypothesis*_2 is chosen based on the result of the

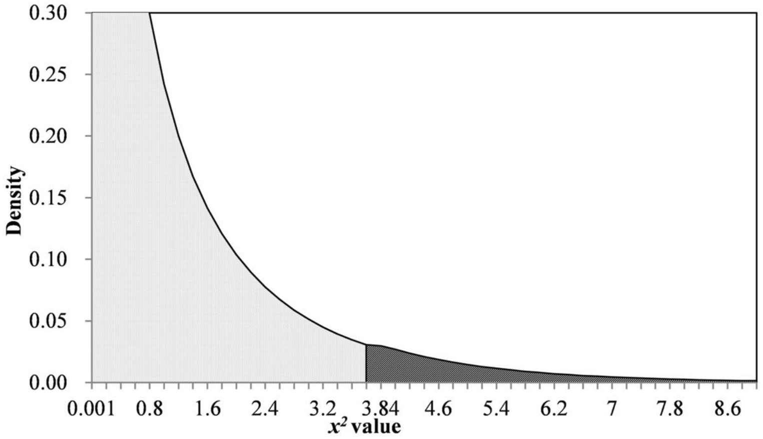


Fig. 7 The PDF of χ^2 distribution

comparison. As C_E holds emotional words of relative strong discriminability which are more likely to represent the class, the proposed weighting scheme strengthens the discriminability to highlight the role of emotional words in category prediction. Recall that the larger the χ^2 value is, the more discriminative information of the class the feature holds [22]. If $\chi^2(C_E, D_{i,\alpha}) < \mathrm{CV}$, C_E and $D_{i,\alpha}$ are regarded as independent from each other, which indicates that C_E does not contain enough information on class $D_{i,\alpha}$, and thus no weighting is applied. Only when $\chi^2(C_E, D_{i,\alpha}) > \mathrm{CV}$, the words in C_E are supposed to be highly dependent on $D_{i,\alpha}$, and the proposed weighting scheme is performed. Observe from Table 4 that $\chi^2(C_E, D_{i,\alpha})$ increases with the growth of A because B, C and D are constant values. Therefore, the value A is increased to make the words in C_E more discriminative, and firstly the distortion of the importance of a word is defined as,

$$\Theta(w_i, D_{i,\alpha}) = |\alpha(w_i, D_{i,\alpha}) - \beta(w_i, D_{i,\alpha})| = |WF_{w_i} - \exp(w_i)| \tag{15}$$

Here $\Theta(w_i, D_{i,\alpha})$ measures the importance of a word between the observation and prediction. $\alpha(w_i, D_{i,\alpha})$ is the observed importance of the word_w_i for class_$D_{i,\alpha}$ measured by the relative frequency, WF_{wi}, of word_w_i. $\beta(w_i, D_{i,\alpha})$ represents the predicted importance, and it is computed by the deviation between the expectation of Chi Square method. The distortion of the importance for the clustered set, $C_\alpha(\alpha = E \vee N)$, is obtained as,

$$\Theta(C_\alpha, D_{i,\alpha}) = \sum_{i=1}^{m} \Theta(w_i, D_{i,\alpha}) = \sum_{i=1}^{m} |WF_{w_i} - \exp(w_i)| \tag{16}$$

$\Theta(C_\alpha, c_k)$ measures the distortion of the importance between cluster_C_α and class_$D_{i,\alpha}$. The increment rate of A, r_A, is computed as,

$$r_A = \frac{\Theta(C_\alpha, D_{i,\alpha})}{\sum\limits_{i=1}^{m} WF_i} \tag{17}$$

Since $\Theta(C_\alpha, c_k)$ is equal to $|A - E_{11}|$, Eq. (17) can be rewritten as,

$$r_A = \frac{\sqrt{D_{11} \cdot E_{11}}}{A} \tag{18}$$

Here D_{11} and E_{11} are the deviation and expected frequency, and $\sqrt{D_{11} * E_{11}}$ measures the difference between the observed frequency and expected frequency in $D_{i,\alpha}(\alpha = neg \vee pos)$. By dividing it by the value of A, the increment rate of r_A is calculated. Meanwhile, $\chi^2(C_N, D_{i,\alpha}) > \chi^2_\alpha$ might be possible since some non-emotional words also have small discriminability. Moreover, as the volume of data of C_N is much larger than C_E, the

Table 5 Different cases of summing of frequencies

Feature selection	$\in D_{i,\alpha}$	$\notin D_{i,\alpha}$	Sum
Containing C_N	E	F	$E+F$
Not containing C_N	G	H	$G+H$
Sum	$E+G$	$F+H$	M

value of $\chi^2(C_N,D_{i,\alpha})$ might be greater than the CV of χ^2. Therefore, r_N is computed based on Table 5 and Eq. (18). The increment rate, r_D, is then,

$$r_D = \begin{cases} r_E - r_N, & r_E > r_N \\ 0, & r_E \le r_N \end{cases} \tag{19}$$

Here r_D increases the frequency of A when C_E holds enough words of strong discriminability and contains more class information than C_N. Otherwise, it is set to be zero. In addition, as the value of A is the sum of the WF of the words of all the documents in C_E $(=\{w_1,\ldots w_p\}, w_i=\{(f_i,t_i,D_{i,\alpha},NW_i)|j=neg\vee pos\}$, the new weight of every word in C_E, NW_i, is calculated as $NW_\mathrm{i}=(1+r_D)\cdot WF_\mathrm{i}$.

Testing

Bayes theorem is widely used in supervised learning for text classification. In this paper Multinomial Naïve Bayes (MNB) model is employed as the classifier for the given text, which is based on naïve assumption of conditional independence for the features [61]. Specifically, in the text classification of sentiment analysis, the goal is to find the best matching class for the tested sentences. It is the most likely or maximum a posteriori (MAP) class, c_{map}, which is calculated as,

$$c_{map} = arg \max_{c\in C} (P(c|S)) \tag{20}$$

where c is a class in the total classes in training dataset, C, and $P(c|S)$ is posterior probability of class_c measuring the probability of sentence_S being in class_c as computed by Eq. (21). In the proposed scheme two classes are defined, D_{neg} and D_{pos}. Therefore, $C=\{D_{neg}, D_{pos}\}$, and the objective is to find the best matching class among C for every tested sentence.

$$P(c|S) = \frac{P(S|c)P(c)}{P(S)} \tag{21}$$

Since the probability of the sentence, $P(S)$, is a constant, it can be discarded. Equation (21) can then be expressed as,

$$P(c|S) \propto P(c)P(S|c) = P(c) \prod_{1\le i\le n} P(w_i|c) \tag{22}$$

$S_i=\{w_1,\ldots,w_{|k|}\}$ represents one sentence composed of $|k|$ words, and w_j $(j=1,\ldots,n)$ is a word in the sentence. For example, for the sentence of "*peace is important*", $S=\{$*peace*, *is*, *important*$\}$, with $|k|=3$. $P(w_i|c)$ is the conditional probability of w_j occurring in a sentence of class_c, which measures how much w_j contributes for class_c to be the matching class. $P(c)$ is the prior probability of a sentence in class_c. In Eq. (22), multiplying many conditional probabilities may lead to the problem of floating point underflow. Therefore, adding the logarithms of the probabilities instead of the multiplication is carried out. As the logarithmic function is monotonic, the class of the highest probability can still be selected as the target class. Equation (22) is thus converted to,

$$c_{map} = \arg\max_{c \in C} \left[\log P(c) + \sum_{1 \leq i \leq n} \log P(w_i|c) \right] \tag{23}$$

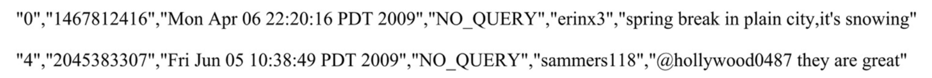
$P(c)$ is obtained as, $P(c)=N_c/N$. Where N_c is the number of sentences in class_*c* and N is the total number of sentences in the training dataset. $P(w_i|c)$ is the posterior probability of T_i, which is denoted as, $P(w_i|c)=count(w_i,c)/count(c)$. Here $count(w_i,c)$ is the number of appearances of w_i in class_*c* of the training dataset and $count(c)$ is the total number of words in class_*c*. Meanwhile, the problem of zero probability can occur if a word in a sentence does not appear in the training dataset. Then, no matter how strong evidence could be gained from other words for the class, the estimation becomes zero. Laplace smoothing is employed to avoid this issue as [62], $P(w_i|c)=(count(w_i,c)+1)/(count(c)+|V|)$. Where $|V|$ is the number of distinct words in the training dataset. Recall that, in the proposed scheme, the weighted frequency (*NW*) of every word in the training dataset has been adjusted considering its relative popularity. Using it, $P(w_i|c)$ is obtained as, $P(w_i|c)=(NW_i+1)/(count(c)+|V|)$.

Recall that NW_i was obtained based on the dependency and discriminability of the word in the target class, where the influence of emotional words was strengthened by assigning more weight than the others for more accurate prediction. Note that redundant feature words are considered with the MNB model. For instance, assume that sentence_S_i is composed of four words as $S_i=\{w_1, w_2, w_3, w_1\}$. Then the numerator of Eq. (22) becomes $P(C_i)\cdot P(w_1|C_i)^{2\cdot}P(w_2|C_i)\cdot P(w_3|C_i)$. Where w_1 has twice as much influence as the other words. The redundant word is therefore given more weight, which leads to biased and low accuracy prediction. In the proposed scheme, thus, only distinct words in the tested sentence are counted.

Performance evaluation

In this section the proposed scheme is evaluated by computer simulation using Matlab. For this, the workload obtained from Sentiment 140 [25] is used to analyze the accuracy of the proposed scheme for twitter sentiment analysis. It is also compared against the previously existing FF, PSW, DF, and TF–IDF scheme.

The simulator consists of three parts; preprocessor, POS tagging API, and Bayes-based classifier. The preprocessor classifies the data of the training data set, and converts them to the customized format accessible by the API of POS tagging [46]. A Matlab function is implemented for accessing the Stanford POS tagger [63], which provides the API for the data in the workload. The Bayesian classifier is used to classify the tested document and predict the sentiment of the target sentences. A Multinomial Naïve Bayes (MNB) model is employed as the classifier in the simulation. The workload used in the simulation is extracted from Sentiment 140, which contains 1,600,000 lines of tweet data. 800,000 of them are negative class and the others are positive class. In the simulation the documents



```
"0","1467812416","Mon Apr 06 22:20:16 PDT 2009","NO_QUERY","erinx3","spring break in plain city,it's snowing"
"4","2045383307","Fri Jun 05 10:38:49 PDT 2009","NO_QUERY","sammers118","@hollywood0487 they are great"
```

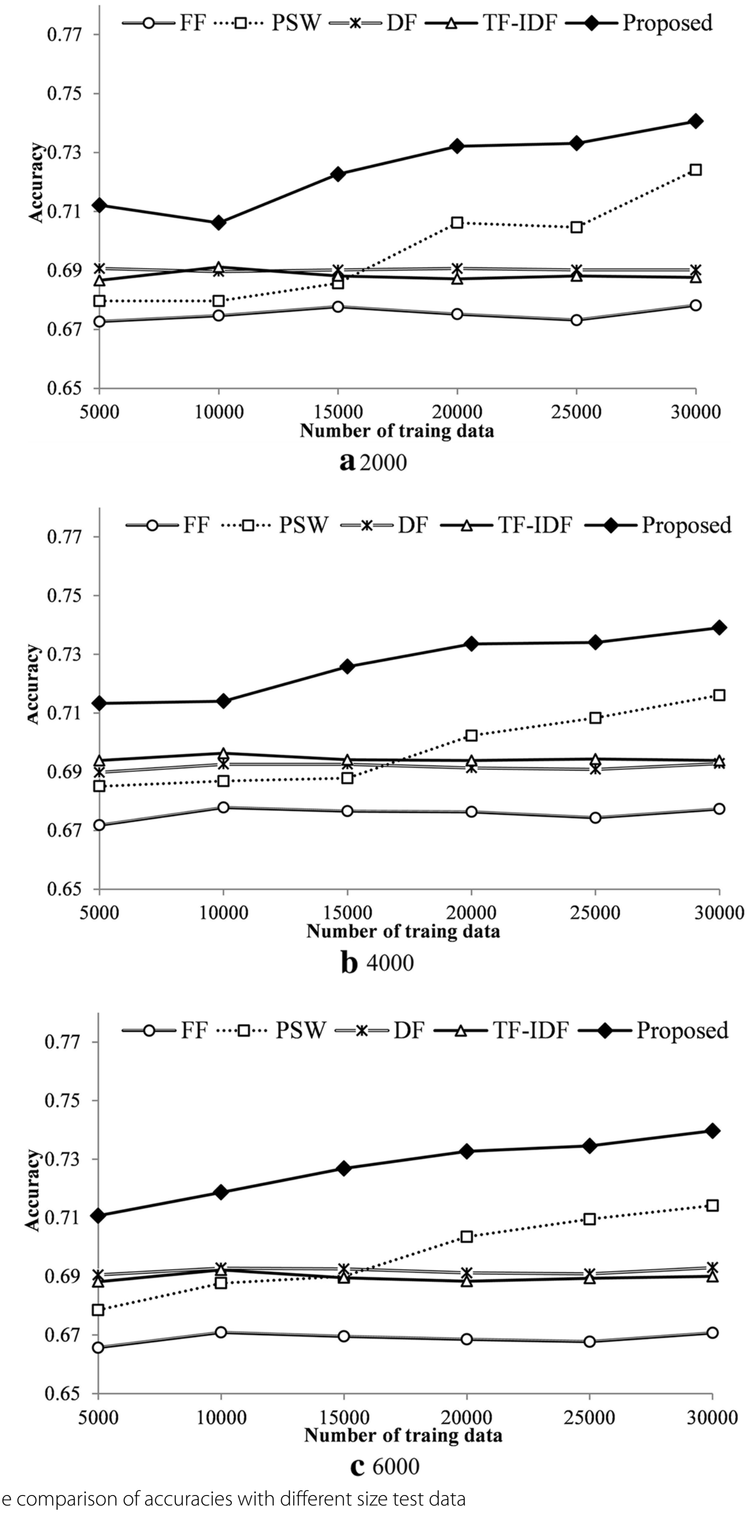
Fig. 8 Two examples of tweets in the data set

used for the training process are also classified into two categories; positive and negative. Figure 8 shows two examples of data in the data set. There are six components in one tweet; polarity (0 = negative, 4 = positive), id, data, query condition, user name, and text of the tweet. The text is used in the training stage of the simulation.

Extensive simulations are run to obtain reliable performance data. Six training data sets of (5000, 10,000, 15,000, 20,000, 25,000, 30,000) randomly selected tweets data are built, and each of which consists of an equal number of positive and negative randomly selected tweets extracted from the Sentiment 140. In addition, a series of tested data sets are formed to verify the performance. Firstly, three tested data sets consisting of 2000, 4000 and 6000 data of equal size of negative and positive document are built to compare the accuracy of the schemes with six different sizes of training data ranging from 5000 to 30,000. The results are shown in Fig. 9. Observe from the figure that the proposed scheme consistently outperforms the other schemes regardless of the size of training and test data set. Intuitively, the accuracy of sentiment analysis increases as the volume of training data grows. It is because the larger the training data, the more evidences could be provided for sentiment judgement. Also notice that the accuracy of the proposed scheme gradually increases with the growth of the size of training data set. However, there is no such improvement with the other schemes excluding the PSW scheme. This is because the parameters of the feature classification model were decided empirically. Moreover, the accuracy generally drops as the test data increases because a limited size training data cannot consistently provide robust evidence for sentiment analysis. Observe from Fig. 9c that the proposed scheme is substantially more accurate than the others even in the worst condition of 'minimum training data (5000) and maximum test data (6000)'. This is because *WF* is utilized as a parameter in the Chi Square method of the proposed scheme, which overcomes the drawback of the traditional Chi Square method in analyzing low frequency terms. Moreover, as a large value of Chi Square implies more class information of the feature (attribute), the weight applied to the words properly takes the interclass dependency into consideration. This enhances the feature of important words of high discriminability, which in turn produces higher accuracy than other schemes.

In order to check the robustness of the proposed scheme, the accuracy is also measured with three randomly selected test data sets containing 3000 positive and 3000 negative data, respectively. The outcomes are shown in Fig. 10. Observe from the figure that the proposed scheme substantially displays higher accuracy regardless of the volume of training and test data. It is also worth to note that the accuracy decreases as the number of test data increases from 1000 to 6000. This is because the accuracy of sentiment analysis is significantly affected by the pattern of the test sentences. TF–IDF also shows reasonable performance since it employs feature weighting. The PSW scheme offers good accuracy when the size of training data is large. The DF scheme is consistently superior than the FF scheme. Pan et al. [10] identified that considering the presence or absence of features can allow higher accuracy than considering only the feature frequency. This is the reason why the DF scheme outperforms the FF scheme.

In the previous simulations the test data consists of equal number of positive and negative tweets. In order to evaluate the sensitivity of the proposed scheme with

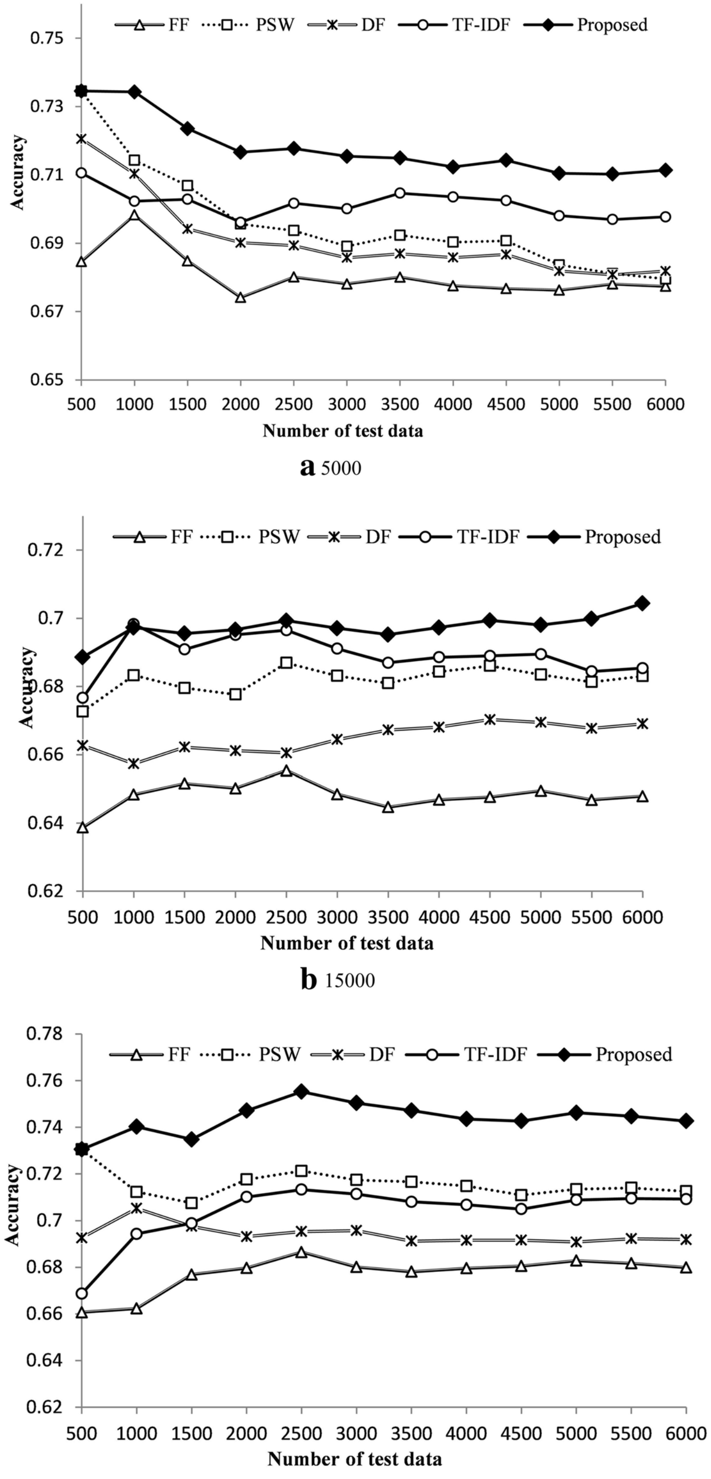


a 2000

b 4000

c 6000

Fig. 9 The comparison of accuracies with different size test data

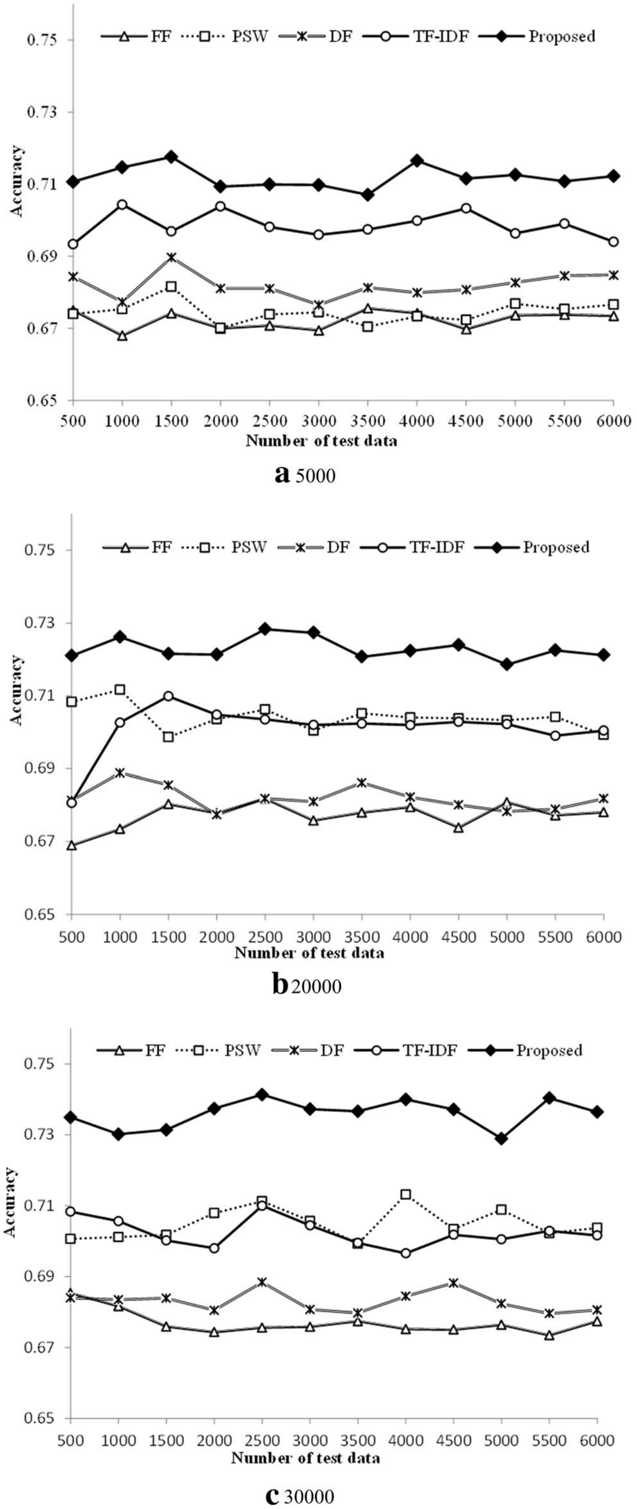


a 5000

b 15000

c 25000

Fig. 10 The comparison of accuracies with the test data of balanced polarity

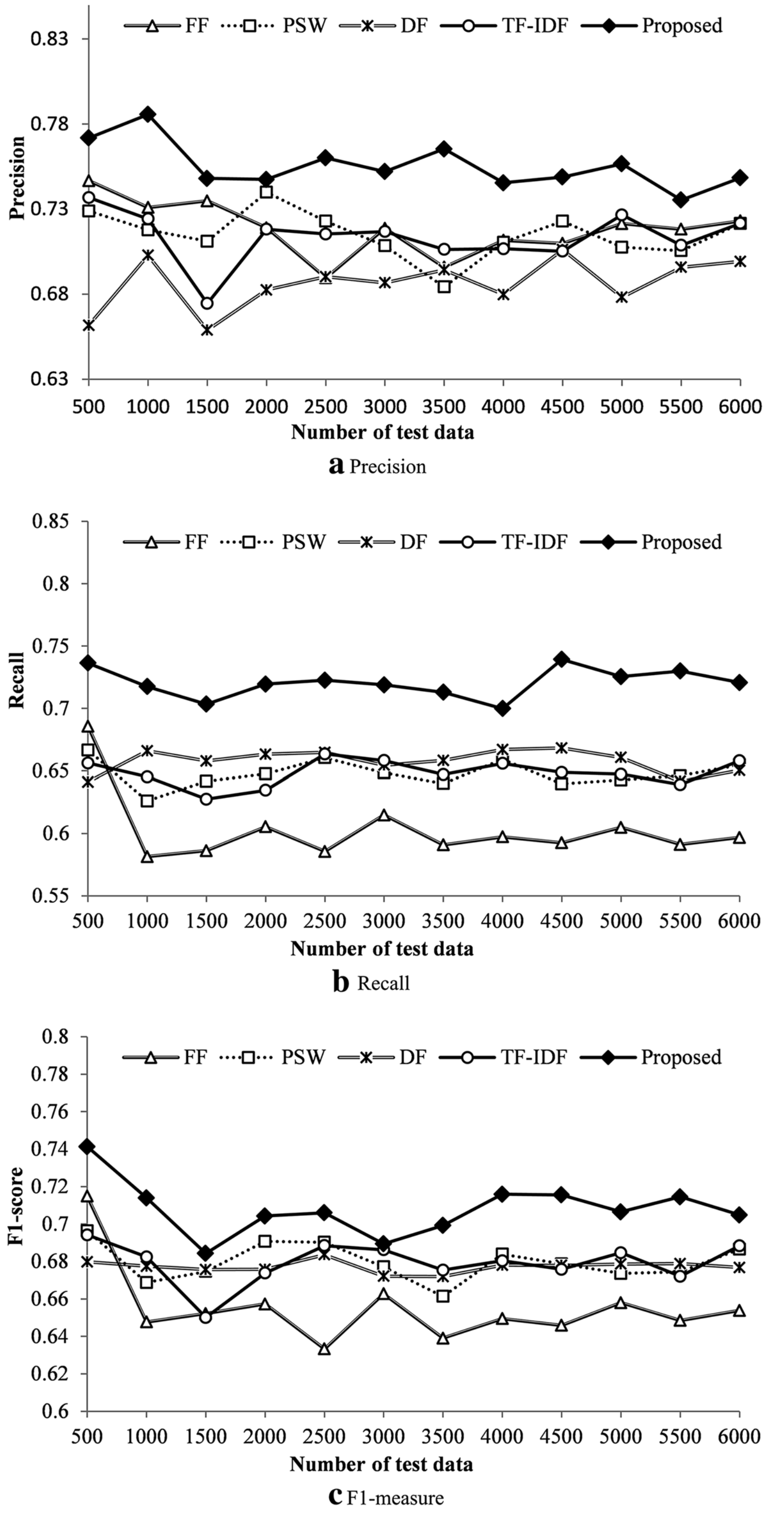


a 5000

b 20000

c 30000

Fig. 11 The comparison of accuracies with the test data of unbalanced polarity



a Precision

b Recall

c F1-measure

Fig. 12 The comparison of three benchmarks with the randomly selected test data

respect to the polarity, simulations are made with the test document randomly selected from the test dataset without considering the polarity. Three different sizes of training data set of 5000, 20,000 and 30,000 are taken to handle the test data. The average accuracy is shown in Fig. 10 revealing that the proposed scheme consistently outperforms the others (Fig. 11).

Figure 12 shows the performance of the five schemes when the test dataset is randomly selected from Sentiment 140. Note that the proposed method significantly outperforms other schemes for the three well-known benchmarks. The proposed scheme produces best performance in terms of precision, recall, and F1-measures when the size of test dataset varies from 500 to 6000. It reveals that the proposed scheme is very sensitive to the sentiment of test documents and is capable of classifying test data into correct category.

Conclusion

Twitter sentiment analysis has become a promising technique for industry and academia. In this paper a novel feature weighting approach for sentiment analysis of twitter data has been proposed using a Bayes-based text classifier. An effective feature selection strategy recognizing sentiment sentence is presented to select informative data for classification. Moreover, each term is grouped into target cluster considering the POS property of the term. A novel feature weighting scheme considering discriminability and dependency derived from modified Chi Square statistics is introduced, which computes a proper weight value for each term reflecting the importance degree of the term. Extensive experiments were conducted on Sentiment 140, and four representative feature weighting schemes were also tested to demonstrate the performance. The experimental results show that the proposed scheme consistently outperforms others in terms of accuracy, precision, recall, and F1-measure. In the future a fine-grained clustering strategy is planned to be developed to accurately define the margin of the clusters. Moreover, unsupervised learning techniques will be incorporated into the proposed scheme to further improve the performance of sentiment analysis. In addition, the proposed scheme will be tested using various classifiers such as SVM, decision tree, and neural network.

Authors' contributions

All authors read and approved the final manuscript.

Author details

[1] College of Information and Communication Engineering, Sungkyunkwan University, Suwon 440746, Korea. [2] College of Software, Sungkyunkwan University, Suwon 440746, Korea.

Acknowledgements

This work was partly supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No. 2016-0-00133, Research on Edge computing via collective intelligence of hyperconnection IoT nodes), Korea, under the National Program for Excellence in SW supervised by the IITP (Institute for Information & communications Technology Promotion) (2015-0-00914), Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2016R1A6A3A11931385, Research of key technologies based on software defined wireless sensor network for realtime public safety service, 2017R1A2B2009095, Research on SDN-based WSN Supporting Real-time Stream Data Processing and Multiconnectivity), the second Brain Korea 21 PLUS project, and Samsung Electronics.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

Not applicable.

Consent for publication
Not applicable.

Ethics approval and consent to participate
Not applicable.

Funding
Not applicable.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 6 April 2018 Accepted: 1 June 2018
Published online: 18 June 2018

References

1. Lizhen L et al (2014) A novel feature-based method for sentiment analysis of Chinese product reviews. China Commun 11:154–164. https://doi.org/10.1109/CC.2014.6825268
2. Bidi N, Elberrichi Z (2016) Feature selection for text classification using genetic algorithms. Paper presented at the 2016 8th international conference on modelling, identification and control, 806–810 Nov 2016. https://doi.org/10.1109/icmic.2016.7804223
3. Qiang G (2010) An effective algorithm for improving the performance of Naive Bayes for text classification. Paper presented at the second international conference on computer research and development, 699–701 May 2010. https://doi.org/10.1109/iccrd.2010.160
4. Sharma N et al (2016) Text classification using combined sparse representation classifiers and support vector machines. Paper presented at the 4th international symposium on computational and business intelligence, 181–185 November 2016. https://doi.org/10.1109/iscbi.2016.7743280
5. Joachims T (1998) Text categorization with support vector machines: learning with many relevant features. Paper presented at the European conference on machine learning, 137–142 April 1998. https://doi.org/10.1007/BFb0026683
6. Qiaowei J (2016) Deep feature weighting in Naive Bayes for Chinese text classification. Paper presented at the 4th international conference on cloud computing and intelligence systems, 160–164 December 2016. https://doi.org/10.1109/ccis.2016.7790245
7. Krouska A, Troussas C, Virvou M (2016) The effect of preprocessing techniques on twitter sentiment analysis. Paper presented at the 7th international conference on information, intelligence, systems and applications, 1–5 December 2016. https://doi.org/10.1109/iisa.2016.7785373
8. Suresh H (2016) An unsupervised fuzzy clustering method for twitter sentiment analysis. Paper presented at the international conference on computation system and information technology for sustainable solutions, 80–85 December 2016. https://doi.org/10.1109/csitss.2016.7779444
9. Yang A et al (2015) Enhanced twitter sentiment analysis by using feature selection and combination. Paper presented at the international symposium on security and privacy in social networks and big data, 52–57 Nov 2015. https://doi.org/10.1109/socialsec2015.9
10. Pang B, Lillian L, Vaithyanathan S (2002) Thumbs up?: sentiment classification using machine learning techniques. Paper presented at the ACL-02 conference on empirical methods in natural language processing, 10:79–86 July 2002. https://doi.org/10.3115/1118693.1118704
11. Zou H et al (2015) Sentiment classification using machine learning techniques with syntax features. Paper presented at the international conference on computational science and computational intelligence, 175–179 March 2015. https://doi.org/10.1109/csci.2015.44
12. Socher R et al (2013) Recursive deep models for semantic compositionality over a sentiment treebank. Paper presented at the conference on empirical methods in natural language processing, 1631–1642, 2013
13. Singh J, Singh G, Singh R (2017) Optimization of sentiment analysis using machine learning classifiers. Hum Comput Inf Sci 7:32. https://doi.org/10.1186/s13673-017-0116-3
14. Yu N et al (2016) A comprehensive review of emerging computational methods for gene identification. J Inf Proc Syst 12:1. https://doi.org/10.3745/JIPS.04.0023
15. Jiadong Z, San-Segundo R, Pardo JM (2017) Feature extraction for robust physical activity recognition. Hum Comput Inf Sci 7:16. https://doi.org/10.1186/s13673-017-0097-2
16. Yang Y, Pedersen JO (1997) A comparative study on feature selection in text categorization. In: ICML. 97:412–420. ISBN:1-55860-486-3
17. Xu Y, Chen L (2010) Term-frequency based feature selection methods for text categorization. Paper presented at the 4th international conference on genetic and evolutionary computing, 280–283 Dec 2010. https://doi.org/10.1109/icgec.2010.76
18. Yili W et al (2017) A novel feature-based text classification improving the accuracy of twitter sentiment analysis. Paper presented at the 12th international conference on future information technology, 440–445 May 2017. https://doi.org/10.1007/978-981-10-7605-3_72
19. Salton G, Buckley C (1988) Term-weighting approaches in automatic text retrieval. Inf Pro Man 24:513–523. https://doi.org/10.1016/0306-4573(88)90021-0
20. Zhihua X et al (2016) A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data. IEEE Trans Parallel Dis Syst 27:340–352. https://doi.org/10.1109/TPDS.2015.2401003

21. Wen Z, Yoshida T, Xinjing T (2011) a comparative study of TF* IDF, LSI and multi-words for text classification. Expert Syst Appl 38:2758–2765. https://doi.org/10.1016/j.eswa.2010.08.066
22. Chuanxin J et al (2015) Chi square statistics feature selection based on term frequency and distribution for text categorization. IETE J Res 61:351–362. https://doi.org/10.1080/03772063.2015.1021385
23. Zhangjie F et al (2016) Enabling personalized search over encrypted outsourced data with efficiency improvement. IEEE Trans Parallel Distrib Syst 27:2546–2559. https://doi.org/10.1109/TPDS.2015.2506573
24. Tinghuai M et al (2016) LED: a fast overlapping communities detection algorithm based on structural clustering. Neurocomput 207:488–500. https://doi.org/10.1016/j.neucom.2016.05.020
25. Sentiment analysis workload Sentiment 140. http://help.sentiment140.com/home. Accessed 15 May 2016
26. Zebin W et al (2016) Parallel and distributed dimensionality reduction of hyperspectral data on cloud computing architectures. IEEE J Sel Top App Earth Obs Remote Sens 9:2270–2278. https://doi.org/10.1109/JSTARS.2016.2542193
27. Paul S, Das S (2015) simultaneous feature selection and weighting–an evolutionary multi-objective optimization approach. Pattern Recognit Lett 65:51–59. https://doi.org/10.1016/j.patrec.2015.07.007
28. Zhaoqing P et al (2016) Fast motion estimation based on content property for low-complexity H.265/HEVC encoder. IEEE Trans Broad 62:675–684. https://doi.org/10.1109/TBC.2016.2580920
29. Wikipedia Naïve Bayes classifier. https://en.wikipedia.org/wiki/Naive_Bayes_classifier. Accessed 3 June 2016
30. Suresh Y (2016) Software quality assessment for open source software using logistic and Naive Bayes classifier. Paper presented at the International conference on computation system and information technology for sustainable solutions, 267–272 Oct 2016. https://doi.org/10.1109/csitss.2016.7779369
31. Singh M, Provan, GM (1996) A comparison of induction algorithms for selective and non-selective Bayesian classifiers. Paper presented at the international conference on machine learning, 497–505 May 1996. https://doi.org/10.1016/b978-1-55860-377- 6.50068-2
32. Wikipedia Sentiment analysis. https://en.wikipedia.org/wiki/Sentiment_analysis. Accessed 20 Jan 2016
33. Wilson T, Wiebe J, Hoffmann P (2005) Recognizing contextual polarity in phrase-level sentiment analysis. Paper presented at the conference on human language technology and empirical methods in natural language processing, 347–354 Oct 2015. https://doi.org/10.3115/1220575.1220619
34. Miller G et al (1990) Introduction to wordnet: an on-line lexical database. Int J Lexicogr 3:235–244. https://doi.org/10.1093/ijl/3.4.235
35. Baccianella S, Esuli A, Sebastiani F (2010) Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: LREC, 10:2200–2204 May 2010
36. Troussas C et al (2013) Sentiment analysis of Facebook statuses using Naive Bayes classifier for language learning. Paper presented at the 4th international conference on information, intelligence, systems and applications, 1–6 July 2013. https://doi.org/10.1109/iisa.2013.6623713
37. Krouska A, Troussas C, Virvou M (2016) The effect of preprocessing techniques on twitter sentiment analysis. Paper presented at the 7th international conference on information, intelligence, systems and applications, 1–5 July 2016. https://doi.org/10.1109/iisa.2016.7785373
38. Troussas C, Krouska A, Virvou M (2016) Evaluation of ensemble-based sentiment classifiers for twitter data. Paper presented at the 7th international conference on information, intelligence, systems and applications, 1–6 July 2016. https://doi.org/10.1109/iisa.2016.7785380
39. Krouska A, Troussas C, Virvou M (2017) Comparative evaluation of algorithms for sentiment analysis over social networking services. J Univers Comput Sci 23(8):755–768. https://doi.org/10.3217/jucs-023-08-0755
40. Ravichandran M, Kulanthaivel G (2014) Twitter sentiment mining (TSM) framework based learners emotional state classification and visualization for e-learning system. J Theor Appl Inf Technol 69(1):84–90
41. Yu Y, Xiao W (2015) World Cup 2014 in the twitter World: a big data analysis of sentiments in US sports fans. Comput Hum Behav 48:392–400. https://doi.org/10.1016/j.chb.2015.01.075
42. Smailović J et al (2014) Stream-based active learning for sentiment analysis in the financial domain. Inf Sci 285:181–203. https://doi.org/10.1016/j.ins.2014.04.034
43. Silva Da et al (2014) Tweet sentiment analysis with classifier ensembles. Decis Support Syst 66:170–179. https://doi.org/10.1016/j.dss.2014.07.003
44. Yuhui Z et al (2017) Student's t-hidden Markov model for unsupervised learning using localized feature selection. IEEE Trans Circuits Syst Video Technol. https://doi.org/10.1109/TCSVT.2017.2724940
45. Bahassine S, Madani A, Kissi M (2016) An improved Chi-sqaure feature selection for Arabic text classification using decision tree. Paper presented at the international conference on intelligent systems: theories and applications, 1–5 Oct 2016. https://doi.org/10.1109/sita.2016.7772289
46. Stanford Log-linear Part of Speech Tagger. http://nlp.stanford.edu/software/tagger.shtml. Accessed 13 Mar 2017
47. Slide share text analysis for security. https://www.slideshare.net/taoxiease/text-analytics-for-security. Accessed 16 Apr 2017
48. Mekuria Z, Assabie Y (2014) A hybrid approach to the development of part-of-speech tagger for Kafi-noonoo text. Paper presented at the international conference on intelligent text processing and computational linguistics, 214–224 April 2014. https://doi.org/10.1007/978-3-642-54906-9_17
49. O'Keefe T, Koprinska I (2009) feature selection and weighting methods in sentiment analysis. Paper presented at the 14th Australasian document computing symposium, 67–74 Dec 2009
50. Sebastiani F (2002) machine learning in automated text categorization. Paper presented at the ACM computing surveys, 34:1–47 March 2002. https://doi.org/10.1145/505282.505283
51. Zhaoqing P et al (2016) Fast reference frame selection based on content similarity for low complexity HEVC encoder. J Vis Commun Image Rep 40:516–524. https://doi.org/10.1016/j.jvcir.2016.07.018
52. Go A, Bhayani R, Huang L (2009) twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 12, Dec 2009
53. Jinwei W et al (2017) Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics. Multimedia Tools Appl 76:23721–23737. https://doi.org/10.1007/s11042-016-4153-0

54. Jin W et al (2015) Bio-inspired ant colony optimization based clustering algorithm with mobile sinks for applications in consumer home automation networks. IEEE Trans Consumer Electron 61:438–444. https://doi.org/10.1109/TCE.2015.7389797
55. Jin W et al (2005) A load-balancing and energy-aware clustering algorithm in wireless ad-hoc networks. Paper presented at the international conference on embedded and ubiquitous computing, 1108–1117 Dec 2005. https://doi.org/10.1007/11596042_113
56. Jin W et al (2017) Energy-efficient cluster-based dynamic routes adjustment approach for wireless sensor networks with mobile sinks. J Supercomput 73:3277–3290. https://doi.org/10.1007/s11227-016-1947-9
57. Zhangjie F et al (2015) Privacy-preserving smart similarity search based on Simhash over encrypted data in cloud computing. J Int Technol 16:453–460. https://doi.org/10.6138/JIT.2015.16.3.20140918
58. Huan R et al (2018) A novel subgraph K+-isomorphism method in social network based on graph similarity detection. Soft Comput 22:2583–2601. https://doi.org/10.1007/s00500-017-2513-y
59. Gu B, Sun X, Sheng VS (2017) Structural minimax probability machine. IEEE Trans Neural Netw Learn Syst 28:1646–1656. https://doi.org/10.1109/TNNLS.2016.2544779
60. Wikipedia.Chi squared distribution. https://en.wikipedia.org/wiki/Chisquared_distribution. Accessed 2 July 2017
61. Gu B, Sheng VS (2017) A robust regularization path algorithm for v-support vector classification. IEEE Trans Neural Netw Learn Syst 28:1241–1248. https://doi.org/10.1109/TNNLS.2016.2527796
62. The Stanford Natural language processing group. Naive Bayes text classification. http://nlp.stanford.edu/IR-book/html/htmledition/naive-bayes-text-classification-1.html. Accessed 25 June 2016
63. GitHub. Matlab-standford-postagger. https://github.com/musically-ut/matlab-stanford-postagger. Accessed 15 Mar 2017

Submit your manuscript to a SpringerOpen journal and benefit from:

► Convenient online submission

► Rigorous peer review

► Open access: articles freely available online

► High visibility within the field

► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com