Determining Fuzzy Membership for Sentiment Classification: A Three-Layer Sentiment Propagation Model

Chuanjun Zhao¹, Suge Wang¹,²*, Deyu Li¹,²

¹ School of Computer and Information Technology, Shanxi University, Taiyuan, 030006, Shanxi, China, ² Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Taiyuan, 030006, Shanxi, China

* wsg@sxu.edu.cn

Abstract

Enormous quantities of review documents exist in forums, blogs, twitter accounts, and shopping websites. Analysis of the sentiment information hidden in these review documents is very useful for consumers and manufacturers. The sentiment orientation and sentiment intensity of a review can be described in more detail by using a sentiment score than by using bipolar sentiment polarity. Existing methods for calculating review sentiment scores frequently use a sentiment lexicon or the locations of features in a sentence, a paragraph, and a document. In order to achieve more accurate sentiment scores of review documents, a three-layer sentiment propagation model (TLSPM) is proposed that uses three kinds of interrelations, those among documents, topics, and words. First, we use nine relationship pairwise matrices between documents, topics, and words. In TLSPM, we suppose that sentiment neighbors tend to have the same sentiment polarity and similar sentiment intensity in the sentiment propagation network. Then, we implement the sentiment propagation processes among the documents, topics, and words in turn. Finally, we can obtain the steady sentiment scores of documents by a continuous iteration process. Intuition might suggest that documents with strong sentiment intensity make larger contributions to classification than those with weak sentiment intensity. Therefore, we use the fuzzy membership of documents obtained by TLSPM as the weight of the text to train a fuzzy support vector machine model (FSVM). As compared with a support vector machine (SVM) and four other fuzzy membership determination methods, the results show that FSVM trained with TLSPM can enhance the effectiveness of sentiment classification. In addition, FSVM trained with TLSPM can reduce the mean square error (MSE) on seven sentiment rating prediction data sets.

Introduction

Following the popularization of forums, blogs, and online shopping websites, amount of user-generated reviews are growing explosively [1]. Techniques for extracting, arranging, and
Competing Interests: The authors have declared that no competing interests exist.

Funding: This work was supported by: the National High-tech R&D Program (863 Program) (2015AA011808), the URL is http://program.most.gov.cn/; the National Natural Science Foundation of China (61573231, 61632011, 61272095, 61432011, U1435212, 61672331), the URL is http://www.nsfc.gov.cn/; the Shanxi Province Science and Technology Basic Platform Construction Project (2015091001-0102), the URL is http://jctj.sxinfo.net/; and Shanxi Province Graduate Student Education Innovation Project (2016BY004), the URL is http://www.sxedu.gov.cn/. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

As compared with traditional classification tasks, sentiment classification is relatively challenging. A deep semantic analysis of the documents is required to judge the sentiment orientation [5, 6]. Supervised machine learning models, such as, support vector machine (SVM), decision tree, and bayesian classification, have been applied to the text sentiment classification task. Among those models, SVM has achieved effective results [7]. However, SVM assigns equal weight to all samples, while different samples affect or contribute to the classification surface very differently [8, 9]. Fuzzy support vector machine (FSVM) introduces fuzzy membership to the SVM. Each sample is assigned a value of fuzzy membership. The samples which are noisy data or make small contributions to the classification have a lower weight, and the samples that make greater contributions to the classification have a higher weight. Using this strategy, FSVM gives different fuzzy memberships to samples contributing a different amount to classification [10, 11]. Comparing with SVM, FSVM can improve the classification accuracy and reduce adverse effects from the noisy data.

Clearly, sentiment scores can describe the sentiment orientation and sentiment intensity of documents in great detail. It is hard for human beings to estimate the accurate sentiment score of a specified document and the results are also unreliable [12]. Therefore, techniques for capturing sentiment scores automatically are very important. In order to obtain the sentiment score of review documents, researchers have adopted a sentiment lexicon to count the positive and negative words and their sentiment intensity. In addition to the sentiment words and sentiment lexicon, researchers have also used the distance to the class centroid for measuring the fuzzy membership [13]. FSVM has been proved to be effective in theory and applications for classification task. In sentiment classification, we should construct the membership function according to the characteristics of data set and data features. In FSVM, the key is to determine the appropriate fuzzy membership of samples. Fuzzy sentiment membership should reflect the contribution degree of a document to sentiment classification. Generally, we think that strong sentiment intensity of positive or negative documents make large contributions to sentiment classification, while weak sentiment intensity samples are unimportant. Therefore the stronger is the sentiment intensity of documents, the bigger is degree of membership to the sentiment labels. To get more accurate sentiment classification results, we use the absolute value of sentiment score as the fuzzy membership to train the FSVM.

To determine the fuzzy sentiment membership of documents, we adopt a three-layer sentiment propagation model (TLSPM). In this context, the so-called three layers refer to documents, topics, and words. First, we construct nine relationship pairwise matrices between documents, topics, and words. The sentiment score of documents, topics, and words are determined by their sentiment neighbors. Then we obtain a steady sentiment score through continuous iterations. In order to achieve better sentiment classification results, we give higher weights to training samples having a strong sentiment intensity of positive or negative polarity, and lower weights to those having weak sentiment intensity. By using these weighted training samples, a text sentiment classifier of an FSVM can be obtained. Fifteen frequently used real-
world sentiment data sets, including eight two-class data sets and seven multi-level data sets, were selected to evaluate the effectiveness of the proposed method. As compared with SVM and four other fuzzy membership determination methods, the experimental results show that FSVM trained with TLSPM can increase the accuracy of sentiment classification. In addition, FSVM trained with TLSPM can also reduce the mean square error (MSE) on seven sentiment rating prediction data sets.

**Related work**

In this section, we briefly review the existing methods for two-class sentiment classification, sentiment rating prediction methods, and application of the topic model to sentiment classification.

**Two-class sentiment classification**

Traditional text sentiment classification in general divides the reviews into positive or negative categories according to their sentiment orientation [14]. Current methods for two-class sentiment classification can be roughly divided into three approach categories: lexicon-based, semi-supervised, and supervised machine learning [15, 16].

- **Lexicon-based approach.** Sentiment lexicons are widely used in the fine-grained sentiment analysis of reviews. The lexicon-based approach calculates the orientation of a document from the sentiment orientation of words or phrases in the document. Turney [17] proposed a simple unsupervised learning algorithm to predict the sentiment orientation using the average sentiment orientation of the phrases in the review. They first identified phrases that contained adjectives or adverbs using the part-of-speech tagger. In their method, the sentiment orientation of a phrase is calculated as the mutual information of the given phrase and the word “excellent” minus the mutual information of the given phrase and the word “poor”. If the average sentiment orientation of all its phrases is positive, the review is considered positive, and if negative, the review is considered negative. Ohana and Tierney [18] applied the SentiWordNet lexicon to the problem of automatic sentiment classification of film reviews. They determined sentiment orientation by counting positive and negative term sentiment scores. On this basis, they used machine learning methods to classify the reviews and found the relevant sentiment features using SentiWordNet. Through a comparative experiment, they found that the feature set approach was better than the sentiment term counting approach. Kanayama and Nasukawa [19] first detected polar clauses that conveyed positive or negative aspects, after which they built a sentiment lexicon that comprised polar atoms through an unsupervised method. The polar atoms were defined as the minimum syntactic elements that express sentiment. They used context coherency to obtain candidate polar atoms. They needed only untagged domain corpora and an initial lexicon to select the appropriate polar atoms from among candidates.

- **Semi-supervised approach.** Sometimes, the labeled training data for sentiment classification are precious and scarce, while abundant unlabeled reviews are easier to get. By designing strategies or techniques, semi-supervised methods combine a certain amount of unlabeled data with the labeled data in the learning process. Wan [20] focused on the problem of cross-lingual sentiment classification, and leveraged an available English corpus for Chinese sentiment classification by using the English corpus as training data. They first used machine translation methods to reduce the gap between Chinese and English. In their method, English features and Chinese features are considered two independent views of the classification problem. They proposed a co-training approach to utilize unlabeled Chinese data. Li et al. [21] adopted two views, personal and impersonal, and employed them in both supervised and semi-supervised sentiment classification systematically. In this method, personal views consist
of those sentences that directly express a speaker’s feelings and preference for a target object, while impersonal views focus on statements about a target object for evaluation. Based on this, an ensemble method and a co-training algorithm are explored to employ the two views in supervised and semi-supervised sentiment classification, respectively. Yu et al. [16] proposed a semi-supervised approach to solve the imbalance between the subjective and objective classes in the twitter sentiment task. The emotion sentiments automatically was extracted from the tweets, and the required training data set was selected in an automatic manner. With more and more social media users sharing their opinions with additional images and videos, You et al. [22] presented a cross-modality consistent regression (CCR) model, which was able to utilize both the state-of-the-art visual and textual sentiment analysis techniques.

Supervised machine learning approach. Supervised sentiment classification methods employ mainly supervised machine learning methods, such as decision tree, naive bayes, SVM, and neural networks [23]. Based on the words that convey sentiment, a new feature selection method based on matrix factorization was proposed by Liang et al. [14] to identify the words with strong inter-sentiment distinguish-ability and intra-sentiment similarity. Ye et al. [24] compared three supervised machine learning methods, naive bayes, SVM, and the character-based N-gram model, for sentiment classification of the reviews of travel blogs for seven popular travel destinations. After experimental verification, they found that the SVM and N-gram approaches performed better than the naive bayes method. The three machine learning approaches reached at least 0.8 when the training set was sufficiently large. Facing with encoding the intrinsic relations between sentences in the semantic meaning of document, Tang et al. [1] presented a neural network approach to learn continuous document representation for sentiment classification. They reported that gated recurrent neural network outperformed traditional recurrent neural network. Aiming at target-dependent Twitter sentiment classification task, Vo et al. [15] explored a rich set of neural pooling functions for automatic feature extraction, drawing theoretical correlations behind these functions.

Sentiment rating prediction

It is worth noting that most studies of sentiment classification in general divide the reviews into positive and negative categories. This is because two-class sentiment classification is relatively simple, being concerned only with the polarity of the comments and not considering the sentiment intensity of reviews. As compared with two-class sentiment classification, sentiment rating prediction is a challenging task. Not only does it judge the sentiment orientation, but also it classifies the reviews into more detailed categories [25, 26].

Pang and Lee [27] applied a meta-algorithm based on a metric labeling formulation of the problem, which altered a given n-ary classifier’s output in an explicit attempt to ensure that similar items were assigned similar labels. They showed that the meta-algorithm could provide significant improvements over both multi-class and regression versions of SVM when a novel similarity measure appropriate to the problem was employed.

Qu et al. [28] captured the sentiment polarity and intensity of N-grams by introducing a novel kind of bag-of-opinions representation. In their method, each opinion is composed of a root word, a set of modifier words from the same sentence, and one or more negation words. For example, in the opinion “not very helpful”, “helpful” is the root word, “very” is the modifier word, and “not” is the negation word. On this basis, they obtained the sentiment score of each opinion using a constrained ridge regression method over a large number of domain-independent reviews. The ratings of test reviews were determined using the sentiment score of all the opinions in the review and a domain-dependent unigram model. As compared with the
previous sentiment ratings prediction methods, its validation for books, movies, and music data sets showed the effectiveness of the bag-of-opinions model.

Long et al. [29] proposed a novel review selection approach for accurate feature rating estimation. They used a bayesian network classifier to predict the sentiment star for each topic in the reviews. In order to achieve better results, their approach selected only those reviews that were related to the topics by using the Kolmogorov complexity (KC) information measure. The rating estimation of the feature for these selected reviews using machine learning techniques provided more accurate results than that for other reviews. The average of these estimated feature ratings also better represented an accurate overall rating for the feature of the service, which provided feedback that helped other users to choose their satisfactory service.

Snyder and Barzilay [30] formulated the sentiment rating prediction task as a multiple aspect ranking problem, where the goal was to produce a set of numerical scores, one for each aspect. They presented an algorithm that jointly learned ranking models for individual aspects by modeling the dependencies between assigned ranks. This algorithm guided the prediction of individual rankers by analyzing meta-relations between opinions, such as agreement and contrast. They proved that an agreement-based joint model was more expressive than individual ranking models.

Wang et al. [31] proposed a new opinionated text data analysis called latent aspect rating analysis (LARA). To analyze the sentiment star of topical aspects, LARA started with some reviews with sentiment ratings, particular aspects in the reviews, and each reviewer’s ratings for a given aspect. To achieve this deeper and more detailed understanding of a review, they proposed a two-stage approach based on a novel latent rating regression model. First, they adopted a bootstrapping method to select the main aspects and segments of reviews. In the second stage, a new generation of the latent rating regression model (LRR) was trained to predict aspect ratings. An important assumption was that the overall rating was generated based on a weighted combination of the latent ratings over all the aspects. Their evaluation using a hotel data set showed the effectiveness of the latent rating regression model and the aspect ratings generation assumption.

Application of the topic model to sentiment classification

The LDA model can detect topics that are implicit in the texts and achieved great success in the text mining field [32, 33]. In LDA, the generation process is defined as follows. (i) For each document, extract a topic from the topic distribution; (ii) extract a word from the topic to be able to get above the word corresponding to the distribution; and (iii) repeat the process until every word in the document has been traversed. The application of a topic model in sentiment analysis can improve the performance of sentiment analysis by mining the topics that are implicit in the texts and sentiment preferences for the topics.

Mei et al. [34] proposed a novel probabilistic model to capture the mixture of topics and sentiments simultaneously. They proposed a topic-sentiment mixture model (TSM). TSM first classified the words into two categories, where one category was irrelevant to the topics, and the other was related to the topics. Then, the second category was divided into positive, negative, and neutral sub-categories, and the probability distribution of words in each class was estimated by using an EM algorithm. Finally, particular topic life cycles and the relationship between topics and sentiment were extracted.

Li et al. [35] assumed that topics in texts were relevant to the sentiment and proposed the sentiment topic joint model (Sentiment-LDA model). Based on this, they found that sentiment was independent of the local context, and they proposed the Dependency-Sentiment-LDA...
model, in which the sentiment of the words in the text formed a Markov chain, and the sentiment of a word was independent of the previous word.

Lin et al. [36] proposed a novel probabilistic model framework called the joint sentiment-topic model (JST), and the re-parameterization JST model called the Reverse-JST model. Both methods were weakly supervised, and therefore, they could easily be adapted to other domains. Joint sentiment-topic models added the sentiment layer into the text layer and topic layer to form four models. The reverse-JST model was also a four-layer Bayesian model, but the sentiment generation process was independent of the topics, as compared with JST.

Three-layer sentiment propagation model

To determine the fuzzy sentiment membership of documents, three-layer sentiment propagation model (TLSPM) makes full use of the relationships between document, topics, and words by using pre-existing tools, such as cosine distance, LDA, and fisher feature selection methods. An important advantage of TLSPM is to construct the sentiment propagation network and its matrix representation between documents, topics, and words. In the sentiment propagation process, the sentiment forming of a document, a topic, or a word is regarded as a propagation process on its carriers. At the same time, the sentiment scores of a document, a topic, and a word are determined by their sentiment neighbors in the sentiment network. The matrix representation of the sentiment network and three kinds of sentiment propagation process including toward document, toward topic, and toward word in the sentiment network are unified in sentiment propagation algorithm. In sentiment propagation algorithm, the sentiments of documents, topics, and words are imaged as a steady state of the sentiment network after the propagation process. After consecutive iterations of sentiment score sets of documents, topics, and words through sentiment neighbors propagation, we get the fuzzy membership set of documents and fuzzy training document set.

Symbols and notions

\( C_1 = \{1, -1\} \): Two-class sentiment label set, where 1 represents a positive tendency and -1 represents a negative tendency.

\( C_2 = \{1, 2, 3, 4, 5\} \): Sentiment rating prediction label set, where 1 represents a strong negative tendency, 2 represents a negative tendency, 3 represents a neutral sentiment tendency, 4 represents a positive tendency, and 5 represents a strong positive tendency.

\( D = \{d_1, d_2, \cdots, d_{|D|}\} \): Document set. Each \( d_i \) in \( D \) has a sentiment label.

\( T = \{t_1, t_2, \cdots, t_{|T|}\} (1 \leq h \leq D) \): Topic set. A topic \( t_h \) is a series of words.

\( W = \{w_1, w_2, \cdots, w_m\} (1 \leq j \leq m) \): Word set. Each \( w_j \) in \( W \) has a sentiment label.

**Sentiment score:** Sentiment score measures the sentiment tendency and intensity of a document \((d_i)\), topic \((t_h)\), or word \((w_j)\). Their sentiment score are denoted by \(\text{score}(d_i)\), \(\text{score}(t_h)\), and \(\text{score}(w_j)\), respectively.

**Fuzzy sentiment membership of a document:** The absolute value \(|\text{score}(d_i)|\) of the sentiment score \(\text{score}(d_i)\) is defined as the fuzzy sentiment membership of a document \((d_i)\).

**Fuzzy training document set:** The fuzzy training document set is defined as \(D_{\text{fuzzy}} = \{(d_1, y_1, s_1), (d_2, y_2, s_2), \cdots, (d_n, y_n, s_n)\}\), where \(d_i\) is a document, \(y_i\) is the sentiment label of \(d_i\), \(s_i\) is the fuzzy sentiment membership of \(d_i\) and \((d_i, y_i, s_i)\) is a fuzzy training sample. \(S = \{s_1, s_2, \cdots, s_n\}\) is the fuzzy membership set.

**Three-layer sentiment network:** This network is a weighted directed graph, which is used to describe the relationships among documents, topics, and words. A document \((d_i)\) is composed of most relevant topics, meanwhile a topic \((t_h)\) is composed of most relevant words \((w_j)\). In the graph, if a kind of relation is symmetric, the corresponding bidirectional edges are then
drawn as undirected lines. The weight of an edge expresses the relation intensity of both nodes linked by the edge. It should be noted that the sentiment information on the network is propagated together with the direction of the edges. A sketch of the structure of this network is shown in Fig 1.

**Neighbors in the sentiment network**: The neighbors of a node (document, topic, or word) in the sentiment network refer to other nodes that link toward the node. The larger the value of the relation intensity between two nodes in the sentiment network, the higher probability of their becoming sentiment neighbors. For example, in the Book domain, the words “good” and “excellent” are linked by a large value of relation intensity, and therefore, they are sentiment neighbors in the sentiment network.

**Matrix representation of the sentiment network**

Let $G = (D_{train}, T, W, E)$ be a sentiment network, where $E$ is the set of weighted directional edges and each edge links two nodes from $D$, $T$, and $W$. 

**Fig 1. Sketch of the three-layer sentiment network.**

doi:10.1371/journal.pone.0165560.g001
We know that any directed graph can be equivalently represented as its adjacent matrix. It is not difficult to see that the graph $G$ can be divided into nine subgraphs with their adjacent matrices as $P$, $Q$, $M$, $N$, $U$, $V$, $G$, $H$, and $Z$, respectively.

Their definitions are given by constructing their adjacent matrices as below.

$P$: The adjacent matrix between documents. The weight of the edge related to documents $d_i$ and $d_j$ is defined as

$$P_{ij} = \frac{d_i \cdot d_j}{|d_i| \times |d_j|}$$

Here, $d_i$ and $d_j$ are also used to denote the vectors of documents $d_i$ and $d_j$, respectively.

$Q$: The adjacent matrix between words. The weight of the edge related to words $w_i$ and $w_j$ is defined as

$$Q_{ij} = BNS(w_i, w_j) = |F^{-1}(p(w_i|w_j)) - F^{-1}(p(w_j|w_i))|$$

$F^{-1}$ represents standard normal distribution of the accumulated anti-probability function, $p(w_i|w_j)$ is the probability of $w_i$ appearing when $w_j$ appears in the same window, and $p(w_j|w_i)$ is the probability of $w_i$ appearing when $w_j$ does not appear. The bi-normal separation (BNS) method was proposed by Forman [37]. We set the window size as 10.

$M$: The adjacent matrix from words to documents. The weight of the edge from word $w_j$ to document $d_i$ is defined as

$$M_{ij} = \frac{tf_{w_j} \times idf_{w_j}}{\sum_{w \in d_i} tf_w \times idf_w}$$

where $tf_{w_j}$ is the word frequency of $w_j$ in $d_i$, $idf_{w_j}$ is the inverse text frequency of $w_j$, $idf_{w_j} = 1 + \log(N/n_w)$, $N$ is the total number of documents, and $n_w$ is the number of documents that contain the word $w_j$. $M_{ij}$ measures the contribution degree of word $w_j$ to document $d_i$.

$N$: The adjacent matrix from documents to words. The weight of the edge from document $d_j$ to word $w_i$ is defined as

$$N_{ij} = \frac{tf_{d_j} \times idf_{d_j}}{\sum_{d \in w_i} tf_d \times idf_d}$$

where $tf_{d_j}$ is the word frequencies of $d_j$, $idf_{d_j}$ is the inverse text frequency of $d_j$, $idf_{d_j} = 1 + \log(N/n_d)$, $N$ is the total number of words, and $n_d$ is the number of words that occur in the document $d_j$. $d \in w_i$ means that the document $d$ contains the word $w_i$. 

$G$ is represented as

$$G = \begin{bmatrix} P & V & M \\ G & U & H \\ N & Z & Q \end{bmatrix}$$

(1)
\(U\): The adjacent matrix between topics. The weight of the edge related to topics \(t_i\) and \(t_j\) is defined as

\[
U_{ij} = \cos(t_i, t_j) = \frac{t_i \cdot t_j}{|t_i| \times |t_j|}
\]  

(6)

Here, \(t_i\) and \(t_j\) are also used to denote the vectors of topics \(t_i\) and \(t_j\), respectively.

\(V\): The adjacent matrix from topics to documents. The weight of the edge from topic \(t_j\) to document \(d_i\) is defined as

\[
V_{ij} = p(t_j|d_i)
\]  

(7)

where \(p(t_j|d_i)\) is the weight of the topic \(t_j\) in document \(d_i\) in the LDA results, and measures the contribution degree of topic \(t_j\) to document \(d_i\).

\(G\): The adjacent matrix from documents to topics. The weight of the edge from document \(d_j\) to topic \(t_i\) is defined as

\[
G_{ij} = p(d_j|t_i)
\]  

(8)

where \(p(d_j|t_i)\) is the weight of the topic \(t_i\) in the document \(d_j\) in the LDA results, and measures the contribution degree of topic \(d_j\) to document \(t_i\).

\(H\): The adjacent matrix from words to topics. The weight of the edge from word \(w_j\) to topic \(t_i\) is defined as

\[
H_{ij} = p(w_j|t_i)
\]  

(9)

where \(p(w_j|t_i)\) is the weight of \(w_j\) in \(t_i\) in the LDA results, and measures the contribution degree of word \(w_j\) to topic \(t_i\).

\(Z\): The adjacent matrix from topics to words. The weight of the edge from topic \(t_j\) to word \(w_i\) is defined as

\[
Z_{ij} = p(t_j|w_i)
\]  

(10)

where \(p(t_j|w_i)\) is the weight of the word \(w_i\) in the topic \(t_j\) in the LDA results, and measures the contribution degree of word \(t_j\) to topic \(w_i\).

**Sentiment propagation process**

Sentiment propagation in this paper refers neither to sentiment propagation among individuals in a social system nor to changes in the sentiment of individuals who already have certain sentiments in a system. In this paper, it means only the acquisition of a more precise sentiment depiction of review documents by using some known exact sentiment information about documents and words, middle layer nodes, i.e., topics hidden in documents, and the relationship among them on the sentiment network through information propagation.

It can be assumed that a review document is generated as follows. The author first selects one or more topics which interest him/her and then selects some of his/her favorite words to describe each topic using his/her sentiments. Therefore, in the aspect of generating review documents, the sentiment of a document determines the sentiments hidden in the document, and the sentiment of a topic determines the sentiments of the words that are related to the topic. Conversely, in the aspect of the composition of a document, the sentiment of a topic is determined by the words that are aggregated to the topic; the sentiments of topics and words determine the sentiment of a document. Therefore, we regard sentiment forming as a propagation process.
process on its carriers, documents, topics, and words in the sentiment network. We image the sentiments of documents, topics, and words as a steady state of the sentiment network after the propagation process. Here, we assume that $score(d_i), score(t_h), score(w_j)$ are influenced by its sentiment neighbors. A flowchart of TLSPM can be seen in Fig 2.

We next design three kinds of sentiment propagation process in the sentiment network. In the following propagation formulas, $\alpha, \beta,$ and $\gamma$ are the document weight, topic weight, and word weight, respectively, and are restrained by $\alpha + \beta + \gamma = 1$.

Toward document ($ToD$):

$$score(D) = \alpha \times \tilde{P} \times score(D) + \beta \times \tilde{V} \times score(T) + \gamma \times \tilde{M} \times score(W)$$ 

(14)

Fig 2. Sketch of three-layer sentiment propagation model.

doi:10.1371/journal.pone.0165560.g002
Toward topic (ToT): 

\[
\text{score}(T) = \alpha \times \tilde{G} \times \text{score}(D) + \beta \times \tilde{U} \times \text{score}(T) + \gamma \times \tilde{H} \times \text{score}(W)
\]  

(15)

Toward word (ToW): 

\[
\text{score}(W) = \alpha \times \tilde{N} \times \text{score}(D) + \beta \times \tilde{Z} \times \text{score}(T) + \gamma \times \tilde{Q} \times \text{score}(W)
\]  

(16)

Remark 1. The number of sentiment neighbors \(k\).

According to the label propagation algorithm (LPA), which was proposed by Liu and Murata [38], \(\text{score}(d_i), \text{score}(t_h), \text{and} \text{score}(w_j)\) in the propagation graph \(G\) are determined by its sentiment neighbors in the sentiment network. We use its neighbors to refer to other nodes that link toward the node.

At the initialization of the adjacent matrices stage, we prune the propagation graph \(G\) and keep \(k\) neighbors of each node. We keep \(k\) major values of each row in \(\tilde{P}, \tilde{Q}, \tilde{U}, \tilde{M}, \tilde{N}, \tilde{V}, \tilde{G}, \tilde{H}, \text{and} \tilde{Z}\), assign 0 to the others, and normalize each row. In the process of sentiment propagation, we choose only the nearest \(k\) neighbors to determine \(\text{score}(d_i), \text{score}(t_h), \text{or} \text{score}(w_j)\). At each step of the sentiment propagation, we update the sentiment score of each node by using adjacent nodes. The greater the similarity of the adjacent nodes, the stronger is the influence weight of adjacent nodes.

Remark 2. Initialization of adjacency matrices.

For guaranteeing the convergence of the sentiment propagation algorithm, we initialize the adjacency matrices \(\tilde{P}, \tilde{Q}, \tilde{U}, \tilde{M}, \tilde{N}, \tilde{V}, \tilde{G}, \tilde{H}, \text{and} \tilde{Z}\) as follows. Two examples of adjacency matrices initialization can be seen in Fig 3.

For \(\tilde{P}\), \(\tilde{Q}\), and \(\tilde{U}\), we first set \(\tilde{P}_{ii} = 0, \tilde{Q}_{ii} = 0, \text{and} \tilde{U}_{ii} = 0\), if any row is 0, we assign \(1/k\) to each element in the corresponding row. Then we keep the \(k\) major value of each row, and assign 0 to the others. Finally, we normalize each row of the matrices \(\tilde{P}, \tilde{Q}, \text{and} \tilde{U}\).

For \(\tilde{M}, \tilde{N}, \tilde{V}, \tilde{G}, \tilde{H}, \text{and} \tilde{Z}\), if any row is 0, we assign \(1/k\) to each element in the corresponding row. Then we keep the \(k\) major values of each row, assign 0 to the others, and normalize each of their rows.

Remark 3. Initialization and normalization of sentiment score.

For the two-class sentiment classification condition, the initial sentiment score of positive reviews is 1 and that of negative reviews is -1. For the sentiment rating prediction condition, the initial sentiment scores of 5-star, 4-star, 3-star, 2-star, and 1-star reviews are 1, 0.5, 0, -0.5, and -1, respectively.

Normalizing \(\text{score}(D)\) to make 

\[
\sum_{d_i \in D_{pos}} \text{score}(d_i) = 1, \quad \sum_{d_i \in D_{neg}} \text{score}(d_i) = -1
\]  

(17)

Normalizing \(\text{score}(W)\) to make 

\[
\sum_{w_j \in W_{pos}} \text{score}(w_j) = 1, \quad \sum_{w_j \in W_{neg}} \text{score}(w_j) = -1
\]  

(18)

Initializing \(\text{score}(T)\) to make \(\text{score}(t_h) = 0, 1 \leq h \leq l\).
Normalizing score(D) in accordance with Eq (19):

\[
\text{score}(d_i) = \begin{cases} 
\sum_{d_j \in D_{\text{pos}}} \frac{\text{score}(d_j)}{\text{score}(d_i)} & \text{score}(d_i) \geq 0 \\
\sum_{d_j \in D_{\text{neg}}} (-\text{score}(d_j)) & \text{score}(d_i) < 0 
\end{cases}
\] (19)

Normalizing score(T) in accordance with Eq (20):

\[
\text{score}(t_h) = \begin{cases} 
\sum_{t_h \in T_{\text{pos}}} \frac{\text{score}(t_h)}{\text{score}(t_h)} & \text{score}(t_h) \geq 0 \\
\sum_{t_h \in T_{\text{neg}}} (-\text{score}(t_h)) & \text{score}(t_h) < 0 
\end{cases}
\] (20)
Normalizing score($W$) in accordance with Eq (21):

$$score(w_j) = \begin{cases} \frac{\sum_{w_j \in W_{pos}} score(w_j)}{\text{score}(w_j) \geq 0} \\ \frac{\sum_{w_j \in W_{neg}} (-score(w_j))}{\text{score}(w_j) < 0} \end{cases}$$

To obtain the steady sentiment score of documents, we repeat the ToD, ToT, and ToW processes until the variation amount of all $score(d_i)$ is less than 0.00001.

We renormalize $score(D)$ using Eq (22) in order to obtain the sentiment score of documents.

$$score(d_i) = \begin{cases} \frac{score(d_i)}{\max (score(d_i))} & \text{score}(d_i) \geq 0 \\ \frac{-score(d_i)}{\min (score(d_i))} & \text{score}(d_i) < 0 \end{cases}$$

The absolute value $|score(d_i)|$ of the sentiment score $score(d_i)$ is defined as the fuzzy sentiment membership of a document $(d_i)$. An example of initialization and normalization of sentiment score can be seen in Fig 4.

**Sentiment propagation algorithm and discussions**

**Sentiment propagation algorithm.** We use sentiment propagation algorithm to get the sentiment scores of documents, topics, and words in TLSPM. Specifically, for implementing sentiment propagation in the sentiment network, we construct the sentiment network and its matrix representation between documents, topics, and words. The overall sentiment propagation process in the sentiment network can be divided into toward document, toward...
document, and toward word. We design the sentiment propagation process by using the sentiment propagation Formulas (14–16) and the normalization of the sentiment score Formulas (17–22). Generally, we think that strong sentiment intensity of positive or negative documents make large contributions to sentiment classification, while weak sentiment intensity samples are unimportant. The sentiment intensity can reflect the fuzzy membership to the sentiment labels. Therefore we determine the absolute value of sentiment score as the fuzzy sentiment intensity can reflect the fuzzy membership.

The complete algorithm is described in Algorithm 1.

**Algorithm 1: Sentiment propagation algorithm**

**Input:** Training text set \( D_{\text{train}} = \{d_1, d_2, \ldots, d_n\}, (1 \leq i \leq n) \), topic set \( T = \{t_1, t_2, \ldots, t_l\}, (1 \leq h \leq l) \), word set \( W = \{w_1, w_2, \ldots, w_m\}, (1 \leq j \leq m) \), initial sentiment score vector \( \text{score}(W) \), \( \text{score}(T) \), and \( \text{score}(D) \), weighting parameters \( \alpha, \beta, \) and \( \gamma \), \( \alpha + \beta + \gamma = 1 \).

**output:** Fuzzy training document set \( D_f^{\text{train}} \).

1. Construct the sentiment network \( G = (D_{\text{train}}, T, W, E) \);
2. Initialize sentiment score vector \( \text{score}(D) \) and \( \text{score}(W) \) using Eqs (17) and (18), initialize \( \text{score}(T) \) to make \( \text{score}(t_h) = 0 \);
3. repeat
4. for \( 1 \leq i \leq n \) do
5. Calculate \( \text{score}(d_i) \) using Eq (14);
6. end
7. Normalize \( \text{score}(D) \) using Eq (19);
8. for \( 1 \leq h \leq l \) do
9. Calculate \( \text{score}(t_h) \) using Eq (15);
10. end
11. Normalize \( \text{score}(T) \) using Eq (20);
12. for \( 1 \leq j \leq m \) do
13. Calculate \( \text{score}(w_j) \) using Eq (16);
14. end
15. Normalize \( \text{score}(W) \) using Eq (21);
16. until converges
17. Renormalize \( \text{score}(D) \) using Eq (22);
18. Calculate fuzzy membership set \( S \) as \( s_i = |\text{score}(d_i)| \);
19. Return fuzzy training document set \( D_f^{\text{train}} \).

**Complexity analysis and convergence.** In each iteration of TLSPM, we need \( O(n(n^2 + \tilde{F} + m^2)) \) to update \( \text{score}(D) \) in ToD process, \( O(l(n^2 + \tilde{F} + m^2)) \) to update \( \text{score}(T) \) in ToT process, and \( O(m(n^2 + \tilde{F} + m^2)) \) to update \( \text{score}(W) \) in ToW process. The complexity in each iteration is \( O((n + l + m)(n^2 + \tilde{F} + m^2)) \), where \( n \) is the number of training documents, \( l \) is the number of extracted topics, and \( m \) is the number of words.

Now, we illustrate the convergency of the algorithm. To keep the convergence of the sentiment propagation algorithm, if any row of \( P, Q, M, N, U, V, G, H, \) and \( \tilde{Z} \) is 0, we assign \( 1/k \) to each element in the corresponding row. In the sentiment network \( G \) for sentiment propagation, for any given \( d_i, t_h \) or \( w_j \), a path constituted by other documents, topics, or words connected to it must exist. Therefore, the sum of each row in \( P, Q, M, N, U, V, G, H, \) and \( \tilde{Z} \) is not 0. This indicates that the sentiment network \( G \) is strongly connected and the corresponding matrix \( G \) is irreducible. According to [39] and [40], \( \text{score}(D) \) must be able to converge to a stable value.

**Parameter selection.** In this paper, we use the validation set to determine the parameters set \( \theta = \{k, nt\text{opics}, \alpha, \beta, \gamma\} \), where \( k \) is the selected number of sentiment neighbors, \( nt\text{opics} \) is the number of topics, \( \alpha \) is the texts weight, \( \beta \) is the topics weight, and \( \gamma \) is the words weight.
The loss function is defined as

$$L(y, \hat{y}) = \frac{1}{2} |y - f(d, \theta)|$$

(23)

Where $y$ is the true label of $d$, $\hat{y} = f(d, \theta)$ is the prediction label by FSVM.

The parameter optimization goal is to estimate the parameters:

$$\hat{\theta} = \arg \min_{\theta} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(d_i, \theta))$$

(24)

Where $\hat{\theta}$ is the optimal parameters set, $n$ is the number of training documents.

To get the optimal parameters, we test the influence of parameters on the validation data set in the proposed approach. To test the parameters which influence the accuracy of the algorithm, we fix the other parameters remaining unchanged during the testing, test various parameters influencing the accuracy of the algorithm individually. After obtaining the optimal parameters with the best accuracy on the validation data set, we get the results on the testing data with the selecting the optimal parameters. In the experimental results and analysis section, we test the performance with varying number of neighbors, selected topics, and documents, topics, words weight, iteration times performance with varying number of neighbors on the validation data set.

**Sentiment classification**

In this section, after describing the manner in which the sentiment scores and fuzzy membership of all training documents are obtained by the sentiment network and the sentiment propagation algorithm (SPA), we introduce their usage in sentiment classification by the FSVM.

In order to obtain a more accurate sentiment orientation of reviews from the testing set, we use $s_i = |\text{score}(d_i)|$ and obtain the fuzzy train set $(d_1, y_1, s_1), (d_2, y_2, s_2), \ldots, (d_n, y_n, s_n)$. Then, we use $\{(d_1, y_1, s_1), (d_2, y_2, s_2), \ldots, (d_n, y_n, s_n)\}$ to train an FSVM $f$. Large value of sentiment intensity of $d_i$ indicates strong sentiment expression and high fuzzy sentiment membership degree. Therefore $d_i$ makes big contribution to sentiment classification.

The prioritization scheme of the FSVM can be formalized as

$$\max W(z) = \sum_{i=1}^{n} z_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} z_i x_i y_j (d_i \cdot d_j)$$

(25)

subject to $\sum_{i=1}^{n} z_i y_i = 0$

(26)

$$0 \leq z_i \leq s_i C, i = 1, 2, \ldots, n$$

(27)

According to Eqs (25)–(27), we obtain the optimal solution $z^* = (z^*_1, z^*_2, \ldots, z^*_n)^T$.

In $z^* = (z^*_1, z^*_2, \ldots, z^*_n)^T$, if $z^*_i > 0$, the corresponding $d_i$ is the support vector. If $0 < z^*_i < s_i C$, this type of support vector lies in the edge of the hyper plane. If $z^*_i = s_i C$, this type of support vector is a misclassified sample.

An important difference between the SVM and FSVM models is that points with the same value of $z^*_i$ may indicate different types of support vector because of the value of $s_i$: A small value makes the sample $d_i$ less important in the training and a big value makes the sample $d_i$ more important to the classification [41].
In this study, we used the Gaussian kernel function \( K(d_i, d_j) = \exp \left( \frac{-|d_i - d_j|^2}{2\sigma^2} \right) \) as the kernel for constructing the FSVM classifier. The corresponding optimal solution is \( \alpha^* = (\alpha^*_1, \alpha^*_2, \ldots, \alpha^*_n)^T \). Therefore, the fuzzy optimal classification is

\[
f^*(x) = \text{sgn} \left\{ \sum_{j=1}^{n} \alpha^*_j y_j K(d_i, d_j) + b^* \right\}
\]

(28)

\[
b^* = y_i - \sum_{j=1}^{n} \alpha^*_j y_j K(d_i, d_j), j \in \{j | 0 < \alpha^*_j < \delta C\}
\]

(29)

For the two-class sentiment classification task, we use the positive tendency reviews as the positive category samples and negative tendency reviews as the negative category samples to train the FSVM. For the sentiment rating prediction task, the only difference is that we use one of the classes as the positive category and the remaining classes as the negative category to train the FSVM (one versus the rest). For example, in the K-class classification task, we use one of the classes as the positive and the remaining K-1 classes as the negative category to train the FSVM. Finally, we obtain K classifiers \( \{f_1, f_2, \ldots, f_K\} \). Therefore, each test sample \( d_i \) has K results:

\[
\{(f_1(d_i), \varepsilon(f_1(d_i))), \ldots, (f_K(d_i), \varepsilon(f_K(d_i)))\}
\]

(30)

where \( f_k(d_i) \) is the result of the \( k \)th classifier and \( \varepsilon(f_k(d_i)) \) is the confidence of the \( k \)th classifier. Finally, we select the corresponding label of \( \max \{\varepsilon(f_1(d_i)), \varepsilon(f_2(d_i)), \ldots, \varepsilon(f_K(d_i))\} \) as the final label of \( d_i \).

### Experimental design

#### Data sets

We constructed experiments using eight two-class sentiment classification data sets and seven sentiment rating prediction data sets. Books (2), DVD (2), Electronics (2), and Kitchen (2) are review sets from Blitzer et al. [42]. Each data set contains 2000 reviews, of which 1000 are positive and 1000 are negative. Notebook (2), Hotel (2) and E-commerce (2) are Chinese review sets from Tan et al. [43]. Each data set has 4000 reviews, of which 2000 are positive and 2000 are negative. Movie (2) is a review set from Pang et al. [27]. This data set contains 50000 reviews, of which 25000 are positive and 25000 are negative. The details of the eight two-class sentiment classification data sets are shown in Table 1. All eight two-class sentiment classification data sets are balanced. Books (4), DVD (4), Electronics (4), Kitchen (4) are review sets

| Category | Books(2) | DVD(2) | Electronics(2) | Kitchen(2) |
|----------|---------|--------|----------------|------------|
| Positive | 1000    | 1000   | 1000           | 1000       |
| Negative | 1000    | 1000   | 1000           | 1000       |
| Total    | 2000    | 2000   | 2000           | 2000       |

| Category     | Notebook(2) | Hotel(2) | E-commerce(2) | Movie(2) |
|--------------|-------------|---------|---------------|---------|
| Positive     | 2000        | 2000    | 2000          | 25000   |
| Negative     | 2000        | 2000    | 2000          | 25000   |
| Total        | 4000        | 4000    | 4000          | 50000   |

doi:10.1371/journal.pone.0165560.t001

PLOS ONE | DOI:10.1371/journal.pone.0165560 November 15, 2016

16 / 32
from Blitzer et al. [42]. Hotel (5) and MP3 (5) are review sets from Wang et al. [44]. Movie (5) is a review set from Pang et al. [27]. Table 2 shows the sentiment rating distributions of seven sentiment rating prediction data sets. In Table 2, it can be seen that all the sentiment rating prediction data sets, except MP3 (5), are primarily balanced. The sentiment ratings distribution of the MP3 (5) data set is unbalanced: the 5-star ranking has the most reviews, and 2-star and 3-star rankings have the least reviews.

| Data set | Reviews | 1-star | 2-star | 3-star | 4-star | 5-star |
|----------|---------|--------|--------|--------|--------|--------|
| Books (4) | 5501 | 1386 | 1400 | 0 | 1381 | 1344 |
| DVD (4) | 5118 | 1321 | 1244 | 0 | 1273 | 1280 |
| Electronics (4) | 5901 | 1468 | 1466 | 0 | 1483 | 1484 |
| Kitchen (4) | 5149 | 1300 | 1313 | 0 | 1262 | 1274 |
| Movie (5) | 15414 | 2899 | 3012 | 3100 | 3047 | 3356 |
| Hotel (5) | 21567 | 3985 | 4180 | 4566 | 4215 | 4621 |
| MP3 (5) | 31000 | 4404 | 2077 | 2531 | 7508 | 14479 |

doi:10.1371/journal.pone.0165560.t002

Text representation and processing

For 12 English data sets, we use the fisher feature selection method [45] to choose the top-800 effective features after removing the stop words. The top 15 features from the Books (2) data set are “poor, fan, lack, repeat, evidence, negative, disappointment, completely, democracy, classic, level, rich, strange, great, and intelligence”. If a word appears in the text, the weight is 1; otherwise the weight is 0. Each review is represented as a bag of words, and then, expressed as a vector space model.

For the three Chinese data sets (Notebook (2), Hotel (2), and E-commerce (2)), we first use an MMSEG segmentation algorithm (http://technology.chtsai.org/mmseg) for segmentation. MMSEG is a word identification system for Mandarin Chinese text based on two variants of the maximum matching algorithm. Then, we remove stop words taken from the Chinese stopping words vocabulary. Top-1000 features are selected by fisher feature selection method [45] for each Chinese data set. For example, the top 15 features from the Notebook (2) data set are as follows: 外观 (appearance), 不错 (not bad), 配置 (configuration), 漂亮 (beautiful), 很好 (good), 性能 (performance), 麻烦 (trouble), 系统 (system), 满意 (satisfy), 便宜 (cheap), 舒服 (comfortable), 价位 (price), 轻便 (light), 做工 (workmanship), 喜欢 (love). Then we express each text as a vector space model, and the feature weight is determined by Boolean value.

We use the JGibbLDA model (http://jgibblda.sourceforge.net) to extract topics in the document. -Alpha is set as 50/ntopics, -twords is set as 50, -savestep is set as 200, -niter is set as 1000, -beta is set as 0.1.

Evaluation metrics

In this study, the evaluation metrics [46] for two-class sentiment classification are shown in Table 3.

\[ RN(Recall) = \frac{d}{b+d} \times 100\% \]  

(31)
Table 3. Confusion matrix of two-class sentiment classification results.

|                | actual positive | actual negative |
|----------------|-----------------|-----------------|
| positive prediction | a               | b               |
| negative prediction | c               | d               |

do:10.1371/journal.pone.0165560.t003

\[
PN(\text{Precision}) = \frac{d}{c+d} \times 100\% 
\]

\[
FN(\text{F1 - measure}) = \frac{2 \times RN \times PN}{RN + PN} \times 100\% 
\]

\[
RP(\text{Recall}) = \frac{a}{a+c} \times 100\% 
\]

\[
PP(\text{Precision}) = \frac{a}{a+b} \times 100\% 
\]

\[
FP(\text{F1 - measure}) = \frac{2 \times RP \times PP}{RP + PP} \times 100\% 
\]

\[
Acc(\text{Accuracy}) = \frac{a+d}{a+b+c+d} \times 100\% 
\]

For sentiment rating prediction, we used accuracy and mean square error (MSE) as the evaluation metrics. The calculation method is

\[
Acc = \frac{n(\text{right answer})}{N - n} 
\]

\[
MSE = \frac{\sum_{i=1}^{N} (\text{answer}_i - \text{result}_i)^2}{N - n} 
\]

where \(n(\text{right answer})\) is the number of samples, the output ratings of which are in accordance with the original ratings, \(N - n\) is the total number of test samples, \(i\) is the number of test samples, \(\text{answer}_i\) is the sentiment rating of the original rating of \(d_i\), and \(\text{result}_i\) is the output rating.

Following most experiment results that were used in studies in the literature, we extract 60% as the training set, 20% as the validation set, and the remaining 20% as the testing set by using stratified sampling [22, 47]. We train the FSVM with TLSPM on the training set, tune parameters on the validation set, and evaluate the effectiveness on the testing set.

Baselines

To verify the validity of TLSPM, we design the following comparison test. The fuzzy membership that generated by Lexicon, Centroid, S-type, Compact, and TLSPM are the input of FSVM. The classification results of FSVM can verify the effectiveness of the listed five fuzzy membership determining methods.
- **SVM:** Use LIBSVM (http://www.csie.ntu.edu.tw/~cjlin/libsvm) with a linear kernel and default parameters [48].

- **Lexicon:** SentiWordNet (http://sentiwordnet.isti.cnr.it) 3.0 is a lexical resource publicly available for research purposes and is an improved version of SentiWordNet 1.0 [49]. We used the positivity, negativity, and neutrality scores that are annotated by SentiWordNet 3.0. We used the positive sentiment scores minus the negative sentiment scores of all the terms in the review as the final score of a review. Finally, the fuzzy membership is determined by the absolute value of sentiment score.

- **Centroid:** Fuzzy membership determining mechanism based on distance to the class centroid [8]. The definition of the centroid of the training set $S$ is $\hat{d} = \frac{1}{n} \sum_{i=1}^{n} d_i$ and the distance between sample $d_i$ and class centroid $\hat{d}$ is defined as

$$ dis(d_i, \hat{d}) = \frac{d_i \cdot \hat{d}}{|d_i| \times |\hat{d}|} $$

(40)

The fuzzy membership $s_i$ of $d_i$ is

$$ s_i = \mu(d_i) = 1 - \frac{dis(d_i, \hat{d})}{r} $$

(41)

where $r$ is the radius of the class and $r = \max \{dis(d_i, \hat{d})\}$.

- **S-type:** Lin and Wang [41] first calculated the distance between $d_i$ and class centroid $\hat{d}$,

$$ dis(d_i, \hat{d}) = \sqrt{\sum_{k=1}^{m} (d_{ik} - \hat{d}_k)^2}, $$

where $m$ was the dimension of $d_i$, and set the parameters $b_1 = 0.1$, $b_2 = 0.5$, and $b_3 = 0.9$. They rehabilitated Zadeh’s proposed standard for S-function transformation and obtained the fuzzy membership function

$$ s_i = \mu(dis(d_i, \hat{d}); b_1, b_2, b_3) =
\begin{cases}
1 & dis(d_i, \hat{d}) < b_1 \\
1 - 2[(dis(d_i, \hat{d}) - b_1)/(b_3 - b_1)]^2 & b_1 \leq dis(d_i, \hat{d}) < b_2 \\
2[(dis(d_i, \hat{d}) - b_1)/(b_3 - b_1)]^2 & b_2 \leq dis(d_i, \hat{d}) < b_3 \\
0 & dis(d_i, \hat{d}) \geq b_3
\end{cases} $$

(42)

where $dis(d_i, \hat{d})$ is the distance between $d_i$ and class centroid $\hat{d}$, $b_1$, $b_2$ and $b_3$ are predefined parameters, and $b_2 = (b_1 + b_3)/2$. If $dis(d_i, \hat{d}) = b_2$, $s_i = 0.5$.

- **Compact:** Batuwita et al. [13] defined the fuzzy membership as

$$ s_i = f(\mu(d_i), \mu_k(d_i, \hat{d})) $$

(43)

where $\mu(d_i)$ is the membership of $d_i$, and belongs to $\hat{d}$ and $\mu_k(d_i, \hat{d})$ is a fuzzy connectivity membership to the centroid and is determined by

$$ \mu_k(d_i, \hat{d}) = \max_{\mu(d_i, \hat{d}) \in \mathbb{R}} \left[ \min \{\mu_k(e_1, e_2), \ldots, \mu_k(e_{m-1}, e_m)\} \right] $$

(44)
where $\rho(d_i, \hat{d})$ is a path from $d_i$ to $\hat{d}$, on which each point is represented by $e_1, e_2, \cdots, e_m$.

where $e_1 = d_i$ and $e_m = \hat{d}$. $P(d_i, \hat{d})$ is the set of all the paths from $d_i$ to $\hat{d}$.

We define $s_i$ using Eq (45):

$$s_i = f(\mu(d_i), \mu_k(d_i, \hat{d})) = \mu(d_i) \times \mu_k(d_i, \hat{d})$$  \hspace{1cm} (45)

$\mu(d_i)$ is calculated by Eq (41).

• **TLSPM**: Our proposed three-layer sentiment propagation model.

### Experimental results and analysis

#### Comparing results and analysis

In order to validate the effectiveness of the presented TLSPM, we designed experiments using 15 real-world sentiment data sets. At the same time, we compared TLSPM with SVM and four other fuzzy membership determination methods.

The comparative experimental results on the testing data can be seen in Tables 4–12. Fig 5 shows the accuracy comparison of the different methods on the testing data. The parameters $k$, $ntopics$, $\alpha$, $\beta$, and $\gamma$ having the best accuracy on the validation set are shown in Table 13.

In Tables 4–12 and Fig 5, we can see that:

1. The accuracies of the six methods (SVM, Lexicon, Centroid, S-type, Compact, and TLSPM) on the seven sentiment rating prediction data sets are lower than on the eight two-class data sets.

2. As compared with using the SVM method directly, the accuracy of the five fuzzy membership determination methods improves greatly; for example, the Lexicon, Centroid, S-type, Compact, and TLSPM methods improved 0.044, 0.032, 0.029, 0.045, and 0.091, respectively on the Books (2) data set.

### Table 4. Experimental results for Books (2) data set.

| Methods | RN    | PN    | FN    | RP    | PP    | FP    | Acc   |
|---------|-------|-------|-------|-------|-------|-------|-------|
| SVM     | 0.782 | 0.776 | 0.779 | 0.774 | 0.780 | 0.777 | 0.778 |
| Lexicon | 0.824 | 0.820 | 0.822 | 0.819 | 0.823 | 0.821 | 0.822 |
| Centroid| 0.819 | 0.805 | 0.812 | 0.801 | 0.816 | 0.808 | 0.810 |
| S-type  | 0.815 | 0.801 | 0.808 | 0.798 | 0.812 | 0.805 | 0.807 |
| Compact | 0.821 | 0.824 | 0.823 | 0.825 | 0.822 | 0.823 | 0.823 |
| TLSPM   | 0.867 | 0.870 | 0.868 | 0.870 | 0.867 | 0.869 | 0.869 |

doi:10.1371/journal.pone.0165560.t004

### Table 5. Experimental results for DVD (2) data set.

| Methods | RN    | PN    | FN    | RP    | PP    | FP    | Acc   |
|---------|-------|-------|-------|-------|-------|-------|-------|
| SVM     | 0.785 | 0.806 | 0.795 | 0.811 | 0.790 | 0.801 | 0.798 |
| Lexicon | 0.815 | 0.806 | 0.811 | 0.804 | 0.813 | 0.808 | 0.810 |
| Centroid| 0.802 | 0.818 | 0.810 | 0.821 | 0.806 | 0.813 | 0.812 |
| S-type  | 0.817 | 0.799 | 0.808 | 0.794 | 0.800 | 0.797 | 0.806 |
| Compact | 0.824 | 0.820 | 0.822 | 0.819 | 0.823 | 0.821 | 0.822 |
| TLSPM   | 0.888 | 0.854 | 0.861 | 0.851 | 0.867 | 0.858 | 0.860 |

doi:10.1371/journal.pone.0165560.t005
### Table 6. Experimental results for Electronic (2) data set.

| Methods  | RN   | PN   | FN   | RP   | PP   | FP   | Acc  |
|----------|------|------|------|------|------|------|------|
| SVM      | 0.787| 0.772| 0.780| 0.768| 0.783| 0.775| 0.778|
| Lexicon  | 0.806| 0.801| 0.804| 0.800| 0.805| 0.802| 0.803|
| Centroid | 0.795| 0.779| 0.787| 0.774| 0.791| 0.782| 0.785|
| S-type   | 0.776| 0.795| 0.785| 0.800| 0.782| 0.791| 0.788|
| Compact  | 0.789| 0.808| 0.798| 0.812| 0.794| 0.803| 0.801|
| TLSPM    | 0.837| 0.844| 0.840| 0.845| 0.838| 0.842| 0.841|

doi:10.1371/journal.pone.0165560.t006

### Table 7. Experimental results for Kitchen (2) data set.

| Methods  | RN   | PN   | FN   | RP   | PP   | FP   | Acc  |
|----------|------|------|------|------|------|------|------|
| SVM      | 0.800| 0.807| 0.803| 0.808| 0.802| 0.805| 0.804|
| Lexicon  | 0.811| 0.808| 0.809| 0.807| 0.810| 0.809| 0.809|
| Centroid | 0.806| 0.810| 0.808| 0.811| 0.807| 0.809| 0.809|
| S-type   | 0.815| 0.822| 0.818| 0.823| 0.817| 0.820| 0.819|
| Compact  | 0.804| 0.813| 0.808| 0.815| 0.808| 0.811| 0.810|
| TLSPM    | 0.836| 0.843| 0.839| 0.844| 0.837| 0.841| 0.840|

doi:10.1371/journal.pone.0165560.t007

### Table 8. Experimental results for Notebook (2) data set.

| Methods  | RN   | PN   | FN   | RP   | PP   | FP   | Acc  |
|----------|------|------|------|------|------|------|------|
| SVM      | 0.807| 0.793| 0.800| 0.789| 0.804| 0.796| 0.798|
| Lexicon  | 0.822| 0.820| 0.821| 0.820| 0.822| 0.821| 0.821|
| Centroid | 0.813| 0.807| 0.810| 0.805| 0.812| 0.808| 0.809|
| S-type   | 0.806| 0.808| 0.807| 0.809| 0.807| 0.808| 0.808|
| Compact  | 0.815| 0.814| 0.815| 0.814| 0.815| 0.814| 0.815|
| TLSPM    | 0.868| 0.876| 0.872| 0.877| 0.869| 0.873| 0.873|

doi:10.1371/journal.pone.0165560.t008

### Table 9. Experimental results for Hotel (2) data set.

| Methods  | RN   | PN   | FN   | RP   | PP   | FP   | Acc  |
|----------|------|------|------|------|------|------|------|
| SVM      | 0.805| 0.801| 0.803| 0.800| 0.804| 0.802| 0.803|
| Lexicon  | 0.808| 0.814| 0.811| 0.816| 0.809| 0.813| 0.812|
| Centroid | 0.808| 0.810| 0.809| 0.811| 0.809| 0.810| 0.810|
| S-type   | 0.818| 0.819| 0.818| 0.819| 0.818| 0.819| 0.819|
| Compact  | 0.827| 0.836| 0.832| 0.838| 0.829| 0.833| 0.833|
| TLSPM    | 0.847| 0.849| 0.848| 0.849| 0.847| 0.848| 0.848|

doi:10.1371/journal.pone.0165560.t009
3. The accuracy of the Compact method is higher than that of the Centroid and S-type methods on 14 sentiment data sets, but not on the Kitchen (2) data set. For example, the Compact method improved 0.016 and 0.013 over the Centroid and S-type methods on the Electronic (2) data set.

4. TLSPM behaves better than Lexicon, Centroid, S-type, and Compact methods on 15 data sets, for example, TLSPM improved 0.036, 0.038, 0.029, and 0.015 on the Hotel (2) data set.
Fig 5. Accuracy comparison of different methods on the testing set. (01) Books (2); (02) DVD (2); (03) Electric (2); (04) Kitchen (2); (05) Notebook (2); (06) Hotel (2); (07) E-commerce (2); (08) Movie (2); (09) Books (4); (10) DVD (4); (11) Electric (4); (12) Kitchen (4); (13) Movie (5); (14) Hotel (5); (15) MP3 (5).

doi:10.1371/journal.pone.0165560.g005
These results can be concluded as below.

1. As compared with two-class sentiment classification, sentiment rating prediction is a more challenging task, because it must not only judge the sentiment orientations of reviews, but also measure their intensity.

2. SVM has been successfully applied to sentiment classification, but it is sensitive to irrelevant and noisy training samples. The FSVM sentiment classification results that using sentiment score as fuzzy membership are very stable and robust.

3. Although TLSPM is more complex than the other methods, it can achieve more accurate sentiment score of documents. Clearly, the fuzzy membership of documents should be determined by using the semantic relations between the documents, topics, and words other than three universal spatial location fuzzy membership determining methods.

### Performance with varying number of neighbors

The second focus of the research was to study the performance with varying number of neighbors on the validation set. ntopics is set as 50, $\alpha$ is set as 0.4, $\beta$ is set as 0.3, and $\gamma$ is set as 0.3, we test the accuracy variation when the value of $k$ increases from 5 to 50. The experimental results are given in Fig 6. It can be clearly seen that when the value of $k$ changes from 5 to 50, the accuracy first increases and then stabilizes on (03) Electric (2), (08) Movie (2) and (11) Electric (4) data sets. The accuracy for the remaining data sets first increases and then decreases. Meanwhile, the accuracy of the seven sentiment rating prediction data sets is lower than that of the eight two-class sentiment data sets. The sentiment score of document, topic, or word is likely to be influenced of noise if $k$ is too small. Instead, the sentiment score will be influenced of irrelevant neighbors if the selected number of neighbors is too big.

### Iteration times performance with varying number of neighbors

To test the iteration times performance with varying neighbors when the value of $k$ changes from 5 to 50, $\alpha$ is set as 0.4, $\beta$ is set as 0.3, $\gamma$ is set as 0.3, and ntopics is set as 50. The results can
Fig 6. Performance with varying values of $k$. (01) Books (2); (02) DVD (2); (03) Electric (2); (04) Kitchen (2); (05) Notebook (2); (06) Hotel (2); (07) E-commerce (2); (08) Movie (2); (09) Books (4); (10) DVD (4); (11) Electric (4); (12) Kitchen (4); (13) Movie (5); (14) Hotel (5); (15) MP3 (5).

doi:10.1371/journal.pone.0165560.g006
be seen in Fig 7. The curves of (02) DVD (2), (04) Kitchen (2), (07) E-commerce (2), (13) Movie (5), and (15) MP3 (5) are very similar: the iteration times first increase and then decrease. The only difference is that the k values that obtain the maximum value are different. The curves of the remaining data sets are very similar, and are not very sensitive to the number of selected neighbors k. This indicates that TLSPM can converge quickly and a large value of k frequently leads to fast convergence.

Performance with varying selected topics

In order to test the accuracy variation when the value of ntopics changes from 10 to 100, k is set as 35, α is set as 0.4, β is set as 0.3, and γ is set as 0.3. The results for the 15 sentiment data sets can be seen in Fig 8. As shown in Fig 8, the curves of the accuracy of the 15 data sets are very similar. The only difference is that the accuracy of the seven sentiment rating prediction sets is lower than that of the eight two-class sentiment data sets. The accuracy first increases and then decreases with the increase number of selected topics.

Performance with varying documents, topics, words weight

To further demonstrate the performance with varying documents, topics, words weight, k is set as 35, ntopics is set as 50, restricted conditions are γ = 1 − α − β and α + β < 1. As shown in Fig 9, we can see that different data sets have different accuracy ternary contour distributions, and the different weights value affect the final sentiment classification results. Specifically, when α, β, and γ have the same value in principle, we get the maximum accuracy. This indicates that words, topics, and documents are all the seem important in determining fuzzy membership of document.

Conclusions and future work

In this paper, a new framework of determining the fuzzy sentiment membership of documents is adopted for sentiment classification task. Our main findings and contributions include the following items. The sentiment score can describe the sentiment orientation and sentiment intensity in great detail. Therefore, sentiment score determination methods are very useful for fine-grained sentiment analysis. Our experiments verified this finding and a better sentiment classification result is achieved by using the sentiment score as fuzzy sentiment membership. SVM has been successfully applied to sentiment classification, but it is sensitive to irrelevant and noisy training samples. Our experiments show that the FSVM model can resolve this problem by assigning different fuzzy memberships to different samples, and different fuzzy membership determination methods lead to different classification results. As is known, sentiment expression is very domain-specific. The same word may have different sentiment orientations and intensity in different domains. Therefore, it is not appropriate to determine the sentiment score of reviews by using an universal sentiment lexicon. The proposed three-layer sentiment propagation model determines the sentiment score of reviews by using the semantic relationships of documents, topics, and words. Therefore, it performs better than three universal spatial location fuzzy membership determining methods (Centroid, S-type, and Compact).

For large-scale sentiment classification task, TLSPM need to solve the following problems: hyper-high dimensional space, massive operation time, and so on. To reduce storage space and running time, we plan to use the partition and combination of the adjacent matrices between documents, topics, and words from MapReduce framework. In the later research, we will promote TLSPM to decrease the algorithm complexity in the matrix and validate our method in big data sets.
Fig 7. Maximum iterations for different values of $k$. (01) Books (2); (02) DVD (2); (03) Electric (2); (04) Kitchen (2); (05) Notebook (2); (06) Hotel (2); (07) E-commerce (2); (08) Movie (2); (09) Books (4); (10) DVD (4); (11) Electric (4); (12) Kitchen (4); (13) Movie (5); (14) Hotel (5); (15) MP3 (5).

doi:10.1371/journal.pone.0165560.g007
Fig 8. Performance with varying values of ntopics. (01) Books (2); (02) DVD (2); (03) Electric (2); (04) Kitchen (2); (05) Notebook (2); (06) Hotel (2); (07) E-commerce (2); (08) Movie (2); (09) Books (4); (10) DVD (4); (11) Electric (4); (12) Kitchen (4); (13) Movie (5); (14) Hotel (5); (15) MP3 (5).

doi:10.1371/journal.pone.0165560.g008
Fig 9. Accuracy ternary contour distribution for different values of the documents, topics, words weight. (01) Books (2); (02) DVD (2); (03) Electric (2); (04) Kitchen (2); (05) Notebook (2); (06) Hotel (2); (07) E-commerce (2); (08) Movie (2); (09) Books (4); (10) DVD (4); (11) Electric (4); (12) Kitchen (4); (13) Movie (5); (14) Hotel (5); (15) MP3 (5).

doi:10.1371/journal.pone.0165560.g009
Acknowledgments

This work was supported by: the National High-tech R&D Program (863 Program) (2015AA011808), the URL is http://program.most.gov.cn/; the National Natural Science Foundation of China (61573231, 61632011, 61272095, 61432011, U1435212, 61672331), the URL is http://www.nsfc.gov.cn/; the Shanxi Province Science and Technology Basic Platform Construction Project (2015091001-0102), the URL is http://jctj.sxinfo.net/. Shanxi Province Graduate Student Education Innovation Project (2016BY004), the URL is http://www.sxedu.gov.cn/. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Author Contributions

Conceptualization: SW CZ DL.
Data curation: CZ SW.
Formal analysis: CZ SW DL.
Funding acquisition: SW DL.
Investigation: CZ SW.
Methodology: SW CZ DL.
Project administration: SW DL CZ.
Resources: CZ SW.
Software: CZ SW DL.
Supervision: SW DL CZ.
Validation: CZ SW.
Visualization: CZ SW DL.
Writing – original draft: CZ SW DL.
Writing – review & editing: CZ SW DL.

References

1. Tang D, Qin B, Liu T. Document modeling with gated recurrent neural network for sentiment classification. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing; 2015. p. 1422–1432.
2. Liu Y, Yu X, Chen Z, Liu B. Sentiment analysis of sentences with modalities. In: Proceedings of the 2013 International Workshop on Mining Unstructured Big Data Using Natural Language Processing. ACM; 2013. p. 39–44.
3. Ranco G, Aleksovski D, Caldarelli G, Grčar M, Mozetic I. The effects of Twitter sentiment on stock price returns. PLoS one. 2015; 10(9):e0138441. doi: 10.1371/journal.pone.0138441 PMID: 26390434
4. Agarwal B, Mittal N, Bansal P, Garg S. Sentiment analysis using common-sense and context information. Computational intelligence and neuroscience. 2015; 2015:30. doi: 10.1155/2015/715730
5. Zhao C, Wang S, Li D. Fuzzy sentiment membership determining for sentiment classification. In: Proceedings of the 2014 IEEE International Conference on Data Mining Workshop. IEEE Computer Society; 2014. p. 1191–1198.
6. Liu D, Li T, Liang D. Incorporating logistic regression to decision-theoretic rough sets for classifications. International Journal of Approximate Reasoning. 2014; 55(1):197–210. doi: 10.1016/j.ijar.2013.02.013
7. Singh N, Mishra RK. Unintentional activation of translation equivalents in bilinguals leads to attention capture in a cross-modal visual task. PloS one. 2015; 10(3):e0120131. doi: 10.1371/journal.pone.0120131 PMID: 25775184
8. Lin CF, Wang SD. Fuzzy support vector machines. IEEE Transactions on Neural Networks. 2002; 13(2):464–471. doi: 10.1109/72.991432 PMID: 18244447
9. Yu H, Liu Z, Wang G. An automatic method to determine the number of clusters using decision-theoretic rough set. International Journal of Approximate Reasoning. 2014; 55(1):101–115. doi: 10.1016/j.ijar.2013.03.018
10. Liu D, Li T, Li H. A multiple-category classification approach with decision-theoretic rough sets. Fundamenta Informaticae. 2012; 115(2-3):173–188. doi: 10.1016/jifu.2011.09.002
11. Zou Q, Zeng J, Cao L, Ji R. A novel features ranking metric with application to scalable visual and bioinformatics data classification. Neurocomputing. 2016; 173:346–354. doi: 10.1016/j.neucom.2014.12.123
12. Neagu DC, Guo G, Trundle PR, Cronin M. A comparative study of machine learning algorithms applied to predicitive toxicology data mining. Alternatives to laboratory animals: ATLA. 2007; 35(1):25–32. PMID: 17411348
13. Batuwita R, Palade V. FSVM-CIL: Fuzzy support vector machines for class imbalance learning. IEEE Transactions on Fuzzy Systems. 2010; 18(3):558–571. doi: 10.1109/TFUZZ.2010.2042721
14. Liang J, Zhou X, Guo L, Bai S. Feature selection for sentiment classification using matrix factorization. In: Proceedings of the 24th International Conference on World Wide Web. ACM; 2015. p. 63–64.
15. Vo DT, Zhang Y. Target-dependent twitter sentiment classification with rich automatic features. In: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015); 2015. p. 1347–1353.
16. Yu Z, Wong RK, Chi CH, Chen F. A semi-supervised learning approach for microblog sentiment classification. In: 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity). IEEE; 2015. p. 339–344.
17. Turney PD. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics; 2002. p. 417–424.
18. Ohana B, Tierney B. Sentiment classification of reviews using SentiWordNet. In: Proceedings of the 9th Annual Information Technology Telecommunications Conference; 2009. p. 13–21.
19. Kanayama H, Nasukawa T. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics; 2006. p. 355–363.
20. Wan X. Co-training for cross-lingual sentiment classification. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP. Association for Computational Linguistics; 2009. p. 235–243.
21. Li S, Huang CR, Zhou G, Lee SYM. Employing personal/impersonal views in supervised and semi-supervised sentiment classification. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics; 2010. p. 414–423.
22. You Q, Luo J, Jin H, Yang J. Cross-modality consistent regression for joint visual-textual sentiment analysis of social multimedia. In: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining. ACM; 2016. p. 13–22.
23. Novak PK, Smalovijc J, Sluban B, Mozetic I. Sentiment of emojis. PloS one. 2015; 10(12):e0144296. doi: 10.1371/journal.pone.0144296
24. Ye Q, Zhang Z, Law R. Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. Expert Systems with Applications. 2009; 36(3):6527–6535. doi: 10.1016/j.eswa.2008.07.035
25. Yu H, Zhang C, Wang G. A tree-based incremental overlapping clustering method using the three-way decision theory. Knowledge-Based Systems. 2016; 91:189–203. doi: 10.1016/j.knosys.2015.05.028
26. Zhou B. Multi-class decision-theoretic rough sets. International Journal of Approximate Reasoning. 2014; 55(1):211–224. doi: 10.1016/j.ijar.2013.04.006
27. Pang B, Lee L. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics; 2005. p. 115–124.
28. Qu L, Iftin G, Weikum G. The bag-of-opinions method for review rating prediction from sparse text patterns. In: Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics; 2010. p. 913–921.
29. Long C, Zhang J, Zhut X. A review selection approach for accurate feature rating estimation. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters. Association for Computational Linguistics; 2010. p. 766–774.

30. Snyder B, Barzilay R. Multiple aspect ranking using the good grief algorithm. In: Proceedings of the 2007 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics; 2007. p. 300–307.

31. Wang H, Lu Y, Zhai C. Latent aspect rating analysis on review text data: a rating regression approach. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM; 2010. p. 783–792.

32. Cao L, Fei-Fei L. Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In: Proceedings of the 11th International Conference on Computer Vision. IEEE; 2007. p. 1–8.

33. Ramage D, Hall D, Nallapati R, Manning CD. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics; 2009. p. 248–256.

34. Mei Q, Ling X, Wondra M, Su H, Zhai C. Topic sentiment mixture: Modeling facets and opinions in weblogs. In: Proceedings of the 16th International Conference on World Wide Web. ACM; 2007. p. 171–180.

35. Li F, Huang M, Zhu X. Sentiment analysis with global topics and local dependency. In: Proceedings the 2010 International Conference of American association for Artificial Intelligence; 2010. p. 1371–1376.

36. Liu X, Murata T. Advanced modularity-specialized label propagation algorithm for detecting communities in networks. Physica A: Statistical Mechanics and its Applications. 2010; 389(7):1493–1500. doi: 10.1016/j.physa.2009.12.019

37. Forman G. An extensive empirical study of feature selection metrics for text classification. The Journal of Machine Learning Research. 2003; 3:1289–1305.

38. Liu X, Murata T. Advanced modularity-specialized label propagation algorithm for detecting communities in networks. Physica A: Statistical Mechanics and its Applications. 2010; 389(7):1493–1500. doi: 10.1016/j.physa.2009.12.019

39. Brin S, Page L. The anatomy of a large-scale hyper textual web search engine. Computer Networks and ISDN Systems. 1998; 30(1):107–117. doi: 10.1016/S0169-7552(98)00110-X

40. Austin D. How Google finds your needle in the web’s haystack. American Mathematical Society Feature Column. 2006; 10:1–13.

41. Lin CF, Wang SD. Training algorithms for fuzzy support vector machines with noisy data. Pattern Recognition Letters. 2004; 25(14):1647–1656. doi: 10.1016/j.patrec.2004.06.009

42. Blitzer J, Dredze M, Pereira F. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In: ACL. vol. 7; 2007. p. 440–447.

43. Tan S, Cheng X. Improving SCL model for sentiment-transfer learning. In: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers. Association for Computational Linguistics; 2009. p. 181–184.

44. Wang H, Lu Y, Zhai C. Latent aspect rating analysis without aspect keyword supervision. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM; 2011. p. 618–626.

45. Wang S, Li D, Song X, Wei Y, Li H. A feature selection method based on improved fisher’s discriminant ratio for text sentiment classification. Expert Systems with Applications. 2011; 38(7):8696–8702. doi: 10.1016/j.eswa.2011.01.077

46. Wang S, Li D, Zhao L, Zhang J. Sample cutting method for imbalanced text sentiment classification based on BRC. Knowledge-Based Systems. 2013; 37:451–461. doi: 10.1016/j.knosys.2012.09.003

47. Shields MD, Teferra K, Hapij A, Daddazio RP. Refined stratified sampling for efficient Monte Carlo based uncertainty quantification. Reliability Engineering & System Safety. 2015; 142:310–325. doi: 10.1016/j.ress.2015.05.023

48. Chang CC, Lin CJ. LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST). 2011; 2(3):1–39. doi: 10.1145/1961189.1961199

49. Baccianella S, Esuli A, Sebastiani F. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: Proceedings of the 7th International Conference on Language Resources and Evaluation. vol. 10; 2010. p. 2200–2204.