Available online at www.sciencedirect.com

ScienceDirect

Karbala International Journal of Modern Science 1 (2015) 26–31
http://www.journals.elsevier.com/karbala-international-journal-of-modern-science/

Statistically weighted reviews to enhance sentiment classification

S. Prakash [a,*], T. Chakravarthy [a], E. Kaveri [b]

[a] *Dept of Computer Science, AVVM Sri Pushpam College, Poondi, Tamil Nadu, India*
[b] *Dept of Computer Science, Bharathidasan University Constituent College for Women, Orathanadu, Tamil Nadu, India*

Received 11 April 2015; revised 28 June 2015; accepted 9 July 2015
Available online 3 September 2015

Abstract

The exponential growth of Internet content, due to social networks, blogs and forums necessitate the research of processing the information in a meaningful way. The research area, Opinion mining is at the cross roads of computation linguistic, machine learning and data mining, which analyze the shared online reviews. Reviews may be about a product, service, events or even a person. Word weighting is a technique that provides weights to words in these reviews to enhance the performance of opinion mining. This study proposes a supervised word weighting method that combined, Word Weighting (WW) and Sentiment Weighting (SW). For WW and SW two function each applied based on word frequency. So totally four statistical functions are applied and checked on categorical labels. Support Vector Machine is used to classify the weighted reviews and it outperforms the existing weighting methods. Two different sizes of corpus are used for the verification.

Keywords: Sentiment analysis; Word weighting; Classification; Support vector machine; Accuracy

1. Introduction

The information shared in social network blogs and forums contain the healthy information about products, event service or popular persons. Opinion mining or sentiment analysis is the new area in which, these information are processed to aid decision for the customers and business people.

The researchers use methods which exist in text processing, machine learning and natural language processing [1–3]. Before classifying the reviews whether the user said about the product positive or negative, the text should be weighted, usually by binary weights [4] and also frequency based [5,6]. Some of the statistical methods are used as feature selection techniques to reduce the dimension [7] and weighting the feature [8]. The proposed method uses two variants of term frequency formula and two statistical methods for finding document frequency, in total there are four combination of weighting methods. Then to evaluate the effectiveness of these methods Support Vector Machine is used to check the weighting influence on classifier. The proposed methods provide best accuracy on bench mark data sets compared with basic tf.idf and BM25 weighting methods.

* Corresponding author.
E-mail addresses: prakashselvakumar@gmail.com (S. Prakash), tcvarthy@gmail.com (T. Chakravarthy), rpkaveri@gmail.com (E. Kaveri).
Peer review under responsibility of University of Kerbala.

http://dx.doi.org/10.1016/j.kijoms.2015.07.001

1.1. Existing methods

Word weighting includes the computation of how much information a word associated to a document is giving relevant to the classes. Though there is no mathematical proof to the tf.idf (term frequency and inverse document frequency), but intuitively many researchers have proved the process [9,10]. Another weighting method, a variant of tf.idf is BM25 used in various studies to provide better results than tf.idf [6,10–12]. Some studies apply weights by selection methods using statistical formulas such as chi square [13], gain ratio, information gain [8]. They had the best result by using CHI in the place of idf on Reuters-21578 dataset, classification by SVM [13]. One of the authors developed a new statistical confidence interval weighting technique which gives more accuracy than tf.idf [14]. To improve the words' discriminating power, tf.rf is used as weighting formula [15]. Class indexing based weighting method computes multiplication of tf.idf with its inverse class space density frequency [16]. Earlier, Pang pointed out binary weights for binary unigram document is the best baseline weighting [4]. Keeping this in mind, BM25, the variant of tf.idf is tried for text classification and proved its efficiency [17]. Both supervised and unsupervised methods are used to learn word features that get semantic and sentiment content, but their results show tf.idf as better method than the proposed one [18]. With this motivation, the proposed weighting techniques take the term frequency variants for word frequency calculation and statistical formulas to get the importance of word to the class.

1.2. Proposed work flow

The role of proposed weighting techniques and its significant role in classification is given in Fig. 1. Online reviews about movies are taken as corpus to prove the performance of the proposed methods. The corpus is preprocessed as given below.

a) Case Folding: Converting the upper case into lower case letters, which is called cleaning the reviews in the documents.
b) Tokenizing: splitting the sentences into separate words of each document.
c) Indexing: Document identification number is created.

The preprocessed documents are weighted by the proposed multiplicative combination of weighting

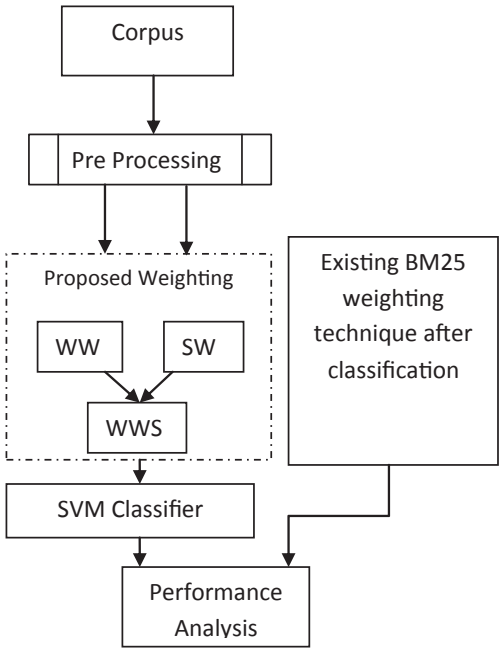

Fig. 1. Proposed work flow.

methods such as two Word Weighting and two Sentiment Weighting methods. The corpus is weighted using four combination of methods separately and given to the Support Vector Machine classifier. SVM learns a model from the labeled data set (Positive/Negative) and classifies the test data set which is given by excluding the labels. The classifier is evaluated based on precision, recall and F1 measure. To verify the proposed weighting techniques, the results are compared against a popular existing weighting technique BM25. For this verification, existing studies which used the same Cornel Movie corpus and did the classification using BM25 are analyzed.

2. Statistical weighting scheme

Word Weighting to Sentiments (WWS) is computed by multiplying Word Weighting (WW) with Sentiment Weighting (SW). This study uses two variants of TF [19] as WW and two statistical formulas as SW.

2.1. Word weighting computation

Let the assumption of positive reviews be R^1 and set of negative reviews be R^2. Let $V = \{v_1, v_2, \ldots v_m\}$ is the unique word set of both review sets. Let document d_j contains word vector $d_j = (w_{1j}, w_{2j}, \ldots, w_{mj})$ and w_{ij} denotes weight of w_i in d_j. w_{ij} is computed as follows

$$\mathrm{WWS} = \mathrm{WW}(\mathrm{w_i}, \mathrm{d_j}) \times \mathrm{SW}(\mathrm{w_i}) \tag{1}$$

where $\mathrm{WW}(\mathrm{w_i},\mathrm{d_j})$ denoted the importance of $\mathrm{w_i}$ in $\mathrm{d_j}$ and $\mathrm{SW}(\mathrm{w_i})$ means the importance $\mathrm{w_i}$ in expressing the sentiments.

2.1.1. Word weighting methods

$$\mathrm{WW}(\mathrm{w_i}, \mathrm{d_j}) = \mathrm{wf_{ij}} \tag{2}$$

$$WW(w_i, d_j) = 0.5 + \frac{0.5 \times w_{ij}}{max_k w_{kj}} \tag{3}$$

where $\mathrm{wf_{ij}}$ is the number of times a word occurs in jth document (Fr). The maximum occurrence of the word in document k of the corpus is mentioned as $max_k w_{kj}$. Eq. (3) does the normalized computation for the count of term, in proportion to the document size (NFr).

2.2. Sentiment weighting methods

SW has two formulas Odds Ratio and Weighted Odds to compute the amount of sentiment possessed by the word related to the class. The explanation for probability notation used in sentiment word weighting is given below.

$P(w_i|R^k)$ − Given condition for a document belongs to class $\mathrm{R^k}$, the probability that word $\mathrm{w_i}$ occurs in the document
$P(w_i|\overline{R^k})$ − Given condition for a document does not belong to class $\mathrm{R^k}$, the probability that word $\mathrm{w_i}$ occurs in the document

The computational notations used to calculate probabilities are given below.

a_i^k − The number of document that both contain the word $\mathrm{w_i}$ and belong to class $\mathrm{R^k}$.
b_i^k − The number of documents that contain word $\mathrm{w_i}$, but do not belong to class $\mathrm{R^k}$.
$\mathrm{t_k}$ − Total number of document in class $\mathrm{R^k}$.

Odds Ratio: Odds ratio is generally used in text mining to rank the words based on the relevancy of class by using the frequency of words [19–21].

$$\mathrm{OR}(\mathrm{w_i}, \mathrm{R^k}) = log \frac{P(w_i|R^k)(1 - P(w_i|\overline{R^k})}{(1 - P(w_i|R^k))P\left(w_i|\overline{R^k}\right)} \tag{4}$$

These probability formulas include the following calculations.

$$\mathrm{OR}\left(\mathrm{w_i}, \mathrm{R^k}\right) \approx \log \frac{a_i^k \times \left(t_1 + t_2 - t_i - b_i^k\right)}{(t_k - a_i^k) \times b_i^k} \tag{5}$$

So the sentiment weighting gets the maximum Odds ratio gain among two classes.

$$\mathrm{SW}(\mathrm{w_i}) = \max\left[\mathrm{OR}\left(\mathrm{w_i}, \mathrm{R^1}\right),\ \mathrm{OR}\left(\mathrm{w_i}, \mathrm{R^2}\right)\right] \tag{6}$$

Weighted Odds: On checking the weight result of Odds ratio, this study proposes a new Weighted Odds formula which performs at par with Odds ration and for some corpus outperforms Odds ratio.

$$\mathrm{WO}\left(\mathrm{w_i}, \mathrm{R^k}\right) = \mathrm{P}\left(\mathrm{w_i}\middle|\mathrm{R^k}\right)^{\alpha} \log\left(\frac{\mathrm{P}(\mathrm{w_i}|\mathrm{R^k})}{\mathrm{P}\left(\mathrm{w_i}\middle|\overline{\mathrm{R^k}}\right)}\right)^{1-\alpha} \tag{7}$$

The probability of WO $(\mathrm{w_i},\mathrm{R^k})$ is estimated as

$$\mathrm{WO}\left(\mathrm{w_i}, \mathrm{R^k}\right) \approx \left(\frac{\mathrm{a_i^k}}{\mathrm{t_i}}\right)^{\alpha} \log\left(\frac{\mathrm{a_i^k}(\mathrm{t_1} + \mathrm{t_2} - \mathrm{t_i})}{\mathrm{b_i^k t_i}}\right)^{1-\alpha} \tag{8}$$

The sentiment weighting gets the maximum Weighted Odds of two classes.

$$\mathrm{SW}(\mathrm{w_i}) = \max\left[\mathrm{WO}\left(\mathrm{w_i}, \mathrm{R^1}\right),\ \mathrm{WO}\left(\mathrm{w_i}, \mathrm{R^2}\right)\right] \tag{9}$$

Training data should have equal class prior probability. The unbalanced training data can be equalized by changing the importance of frequency and odds of frequency among classes. General intuitive conclusions about a good feature are words with high document frequency and words with high category ratio. But any one of the intuitions does not give best weight to the word. Each domain data set distribution is different from other sets. So weighting the corpus with odds ratio can improve accuracy of classifier to some extent. So, combination of measurements is needed to compute best weight. The frequency and odds should be tuned for the corpus which differs based on feature set size. For 500 features, α value can be set to 0.5 and when the size is increasing, α value must be decreased to 0.2 up to 30,000 features, above 30,000, it should be set to 0.01.

3. Experimental discussions

The implementation of proposed word weighting technique is using Support Vector Machine to check its robustness. Two bench mark data sets are taken to verify the method.

3.1. Data set

Earlier, Pang et al. [20] used Cornel Movie review set that contains 1000 positive and 1000 negative sets and they processed these documents from IMDB movie reviews (http://www.cs.cornell.edu/people/pabo/movie-review-data/).

To verify the influence of size of corpus for weighting method, Stanford Movie data set [22], with 25,000 positive and 25,000 negative documents are taken (http://ai.stanford.edu/~amaas/data/sentiment/).

The corpuses consist of equal number of positive and negative documents to avoid bias and to get better model.

3.2. Implementation

Stemming and stop words removal are done for the data set before weighting. LIBSVM (http://www.csie.ntu.edu.tw/~cjlin/libsvm/) is used to implement Support Vector Machine. For text classification SVM out performs in most of the experiments, so linear kernel SVM is used for this experiment.

3.2.1. Support vector machine

Support Vector Machine is the optimum classifier of Machine Learning, which learns from the labeled corpus, creates a model and using the model, it classifies the unlabeled corpus. Statistical inference theory and Vapnik–Chervonenkis dimension concept, Vapnik developed SVM for binary classification. The base is, non separable low dimensional data is to be converted into linearly separable data by plotting into higher dimensional spaces and meanwhile maximizing the margin between the binary classes [23,24].

The problem is converted into quadratic programming using Lagrange multipliers α for a set $\{x_k, y_k\}_{k=1}^{N}$ as follows [24]:

$$\max_{\alpha}\tilde{L}(\alpha) = -\frac{1}{2}\sum_{k,l=1}^{n} y_k y_1 K(x_k x_1) \propto_k \propto_1 + \sum_{k=1}^{N} \propto_k, \tag{10}$$

$$\textit{Such that}, \quad \begin{cases} \sum_{k=1}^{N} \propto_k y_k = 0 \\ 0 \le \propto_k \le c, \ \ k = 1, \ldots, N. \end{cases} \tag{11}$$

The label is calculated as given below using kernel function K(x,y).

$$\mathrm{y(x)} = \mathrm{sign}\left[\sum_{k=1}^{N} \propto_k y_k K(x,y) + b\right] \tag{12}$$

The classifier applies tenfold cross validation in such a way that one of ten parts of the corpus is taken out and training is done on remaining nine parts and testing is done on the separated part. This procedure is applied on all ten part of data set and the average accuracy is taken for performance measurements. Tenfold cross validation is done on the corpus for the combination of WW(2) and SW(2) formulas. Two WW formulas (2) and (3) is multiplied with two SW formulas (5) and (8), in total 4 combinations are available for weighting. The result is compared with the existing weight method BM25 [21].

4. Result and discussions

The performance results of SVM on Cornel Movie reviews are depicted in Fig. 2. The results are given in cases, to ease the discussions.

Case 1: Term frequency calculation and Odds Ratio (Fr*OR) provides 91.7% of accuracy which is lesser than Nf with Weighted Odds (Fr*WO: 93.9%). This gives the inference that the balanced WO works well compared to OR.

Case 2: For further improvement, the Normalized Term frequency is included for weight computation individually with OR and WO. Since the points in the hyper planes are normalized, the NFr multiplicative weights achieve more accuracy than Fr. The Odds Ratio with NFr gives 95.1%, which is higher than Fr*OR by 3.6%.

Case 3: The classification accuracy of NFr*WO is 94.7% which is higher than Fr*WO (93.9%) by just 0.8%.

Case 4: From case 3 and 4, the inference is that the combination NFr*OR has more influence than NFr*WO, in which WO has significance influence in weight computation.

Generally, the classification accuracy differs from the others because of the distribution difference and

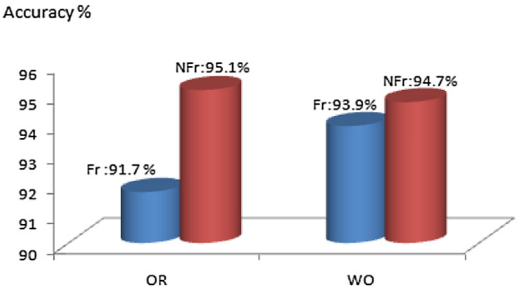

Fig. 2. SVM performances on Cornel Movie Corpus.

size of the corpus. To verify the performance of proposed methods, Stanford movie reviews are taken for weight computation and then fed to the classifier and the outcomes are compared (Fig. 3).

Case 5: Similar to case 1 of Cornel movie analysis, SVM performance on Stanford reviews is higher for Fr*WO (92.7%) than Fr*OR (89.2%).

Case 6: When Fr*OR for Cornel movie set result (91.7%) is compared with Stanford movie set (89.2%), it is found that the accuracy is reduced (2.5%). The inference is the frequency computation is affected by the increased size of corpus.

Case 7: At the same time, Fr*WO accuracy difference on both data set is 1.2%, which is lesser than case 6 statement. This implies WO has balanced the frequency computation.

Case 8: The NFr*OR computation difference between Cornel movie set and Stanford movie set is very less (0.5%), which implies OR is robust and steady even on big data set.

Case 9: The NFr*WO computation difference between Cornel movie set and Stanford movie set is 1.5%, which implies WO is sensitive to size of corpus i.e., accuracy is directly proportional to the corpus size.

Case 10: Fr*WO provides lea accuracy of 92.7% and NFr*WO (96.2%) gives the highest accuracy among all computation.

For both data sets, Fr combination provides less accuracy than NFr. So for further analysis, NFr combination is taken for comparison with existing BM25 weighting method (Fig. 4).

Case 11: The existing weighting method BM25 is applied on corpus and then fed to the SVM classifier. For all the three weighting techniques Stanford set provides higher accuracy and of course as explained in case 8, NFr*OR with less difference.

Case 12: BM25 provides less accuracy than the other two proposed methods. Though NFr*WO is not

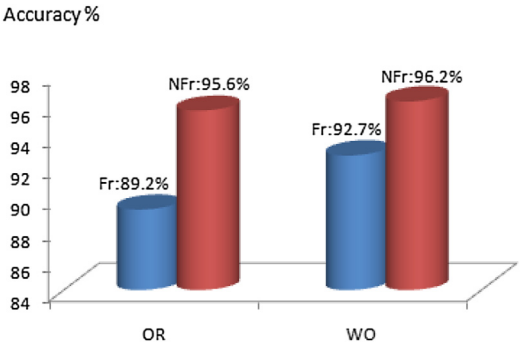

Fig. 3. SVM performances on Stanford Movie Corpus.

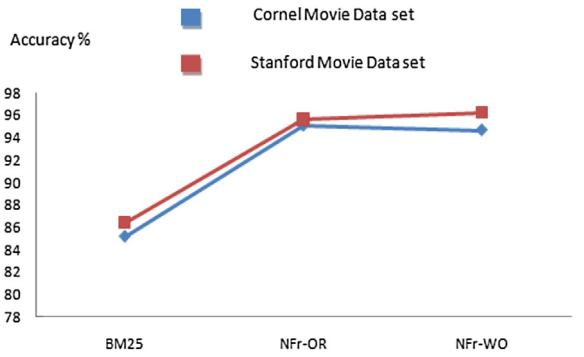

Fig. 4. Comparative performances of existing and proposed weights.

steady as NFr*OR, it provides the maximum accuracy of 96.2%, which is a drastic increase compared to other studies [3,22,26].

The following Table 1 shows the other text evaluation metrics (in percentage) such as Precision, Recall and F1 measure of the classifier for the Cornel movie corpus. Selected document that relevant is measured by Precision and relevant document that are selected is measured by Recall. The weighted average of Precision and Recall is measured by F1 measure.

Case 13: For the existing weighting method (BM25), the difference between precision and recall is more and not balanced. F1 measure is less compared to the proposed computations.

Case 14: Frequency based computation without normalization provides better results when computed with OR and WO. Plain frequency based OR and WO provides more Recall value compared to Precision.

Case 15: On the other hand Normalized frequency based computation provides higher Precision.

Case 16: OR computation using both Word Weighting methods provides the Precision and Recall with more difference (1.63, 2.08), meanwhile WO computation balances the retrieval with less differences (0.19, 0.58).

The above cases explain the importance of Word Weighting (WW) and Sentiment Weighting (SW) computations.

Table 1
Performance evaluation of proposed method through classifier.

Evaluation metric/ Weight method	BM25	Fr-OR	Fr-WO	NFr-OR	NFr-WO
Accuracy	85.16	91.5	93.9	93.3	96.2
Precision	81.26	90.57	93.81	94.38	96.48
Recall	91.5	92.2	94	92.3	95.9
F1 Measure	86.08	91.38	93.91	93.33	96.19

5. Conclusion

In this study, a new feature weighting method that combines word weighting (WW) and sentiment weighting (SW) used. The proposed four statistical functions learn the sentiment from the training set with class labels. The proposed weighting method differs from existing method in extracting even the very minute information (in terms of weights) that is to be conveyed to the class. The results are checked on three popularly known bench mark opinion review sets. Results reveal that the proposed method outperforms compared to the existing methods. The success of the proposed technique lies in utilizing the correlation between the words and sentiments. Since the blogs and forums contents are not following proper dictionary based words, dictionary based corpus will not be a good choice. Hence, corpus based method is followed in this study. The results convey that, the performance vary based on size, domain-wise distribution difference and frequency of bag of words. The negligible limitation of this proposed computation is the time complexity, when the corpus is large such as Stanford movie data set. It takes additional one tenth of time when compared to the normal weighting technique.

More statistical formulas are available in data mining. In future a combination of statistical functions can be used than effective single formula. Though the corpus is for the area sentiment analysis, the proposed word weighting method can be used for any text classifications.

References

[1] B. Pang, L. Lee, Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, vol. 2, Now Publishers Inc, 2008, pp. 1–135.

[2] S. Das, M. Chen, Yahoo! for Amazon: extracting market sentiment from stock message boards, in: Proceedings Asia Pacific Finance Association Annual Conference – APFA, 2001.

[3] V. Ng, S. Dasgupta, S.M. Niaz Arifin, Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews, in: Proceedings COLING/ACL Main Conference Poster Sessions, 2006.

[4] B. Pang, L. Lee, S. Vaithyanathan, Thumbs up? sentiment classification using machine learning techniques, in: Proceedings Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), 2002.

[5] K.S. Jones, A statistical interpretation of term specificity and its application in retrieval, J. Doc. 28 (1) (1972) 11–21.

[6] S.E. Robertson, H. Zaragoza, M.J. Taylor, Simple bm25 extension to multiple weighted fields, in: Proceedings Thirteenth ACM International Conference on Information and Knowledge Management, CIKM, 2004, pp. 42–49.

[7] S. Li, R. Xia, C. Zong, C.R. Huang, A framework of feature selection methods for text categorization, in: Proceeding ACL/AFNLP, 2009, pp. 692–700.

[8] F. Debole, F. Sebastiani, Supervised term weighting for automated text categorization, in: Proceedings of the 2003 ACM Symposium on Applied Computing, SAC, New York, NY, USA, 2003, pp. 784–788.

[9] F. Sebastiani, Machine learning in automated text categorization, ACM Comput. Surv. 34 (1) (2002) 1–47.

[10] S.E. Robertson, S. Jones, Relevance weighting of search terms, J. Am. Soc. Inf. Sci. 27 (1976) 129–146.

[11] S.E. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, Gatford Mike, Okapi at TREC-3, in: Proceedings TREC, 1994, pp. 109–126.

[12] S.E. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, M. Gatford, Okapi at TREC-5, in: Proceedings Fifth Text Retrieval Conference, TREC-5, 1996.

[13] Z. Deng, S. Tang, D. Yang, M. Zhang, L. Li, K. Xie, A comparative study on feature weight in text categorization, in: Proceedings Sixth Asia-Pacific Web Conference, APWeb, Hangzhou, China, 2004, pp. 588–597.

[14] P. Soucy, G.W. Mineau, Beyond TFIDF weighting for text categorization in the vector space model, in: Proceedings Nineteenth International Joint Conference on Artificial Intelligence, IJCAI, Edinburgh, Scotland, UK, 2005, pp. 1130–1135.

[15] M. Lan, C.L. Tan, J. Su, Y. Lu, Supervised and traditional term weighting methods for automatic text categorization, IEEE Trans. Pattern Anal. Mach. Intell. 31 (4) (2009) 721–735.

[16] F. Ren, M.G. Sohrab, Class-indexing-based term weighting for automatic text classification, Inf. Sci. 236 (2013) 109–125.

[17] G. Paltoglou, M. Thelwall, A study of information retrieval weighting schemes for sentiment analysis, in: Proceeding of ACL 2010, 2010, pp. 1386–1395.

[18] J. Martineau, T. Finin, Delta TFIDF: an improved feature space for sentiment analysis, in: Proceedings 3rd AAAI International Conference on Weblogs and Social Media, San Jose, California, USA, 2009, pp. 258–261.

[19] C.D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, vol. 1, Cambridge university press, Cambridge, 2008, p. 496.

[20] D. Mladenic, Marko Grobelnik, Feature selection for classification based on text hierarchy, in: Proceeding Automated Learning and Discovery, CONALD, 1998.

[21] C.J. van Rijsbergen, D.J. Harper, M.F. Porter, The selection of good search terms, Inf. Process. Manag. 17 (1981) 77–91.

[22] A.L. Maas, R.E. Daly, P.T. Pham, Dan Huang, A.Y. Ng, C. Potts, Learning word vectors for sentiment analysis, in: Proceedings 49th Annual Meeting of the Association for Computational Linguistics, ACL, Portland, Oregon, USA, 2011, pp. 142–150.

[23] A. Musa, Comparative study on classification performance between support vector machine and logistic regression, Int. J. Mach. Learn. Cybern. (2013) 13–24.

[24] R. Khemchandani, A. Karpatne, S. Chandra, Twin support vector regression for the simultaneous learning of a function and its derivatives, Int. J. Mach. Learn. Cybern. (2012) 1–13.

[26] B. Pang, L. Lee, A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts, Proc. ACL 2004 (2004) 271–278.