NStackSenti: Evaluation of a Multi-level Approach for Detecting the Sentiment of Users

Md Fahimuzzman Sohan[1(✉)], Sheikh Shah Mohammad Motiur Rahman[1], Md Tahsir Ahmed Munna[2], Shaikh Muhammad Allayear[2], Md. Habibur Rahman[1,3], and Md. Mushfiqur Rahman[1]

[1] Department of Software Engineering, Daffodil International University, Dhaka, Bangladesh
fahisohan2@gmail.com, {motiur.swe, mushfiq.swe}@diu.edu.bd, mdhabibur.r.bd@ieee.org

[2] Department of Multimedia and Creative Technology, Daffodil International University, Dhaka, Bangladesh
{tahsir4ll, drallayear.swe}@diu.edu.bd

[3] Department of Information and Communication Technology, Mawlana Bhashani Science and Technology University, Santosh, Tangail 1902, Bangladesh

Abstract. Sentiment Detection plays a vital role worldwide to measure the acceptance level of any products, movies or facts in the market. Text vectorization (converting text from human readable to machine readable format) and machine learning algorithms are widely used to detect the sentiment of users. This paper presents and evaluates a multi-level architecture based approach using stacked generalization technique named NStackSenti. The presented approach enables the combination of machine learning algorithms to improve the accuracy of detection. Here, Extremely Randomized Tree (ET), Random Forest (RF), Gradient Boost (GB), ADA Boost (ADA), Decision Tree (DT) are used as base classifiers and XGBoost classifier is used as meta estimator. The NStackSenti is applied on two separate datasets to demonstrate the effectiveness in terms of accuracy. NStackSenti provides better accuracy with trigram than unigram and bigram. It provides 83.7% and 86.24% accuracy on 2000 and 50000 data respectively.

Keywords: Machine learning · Sentiment detection · N-gram · Stacked generalization · Ensemble learning

1 Introduction

At present people are expressing their emotions and thoughts on online media with the help of internet. People are reviewing, marking and giving their opinions about the contents that are available online. As a result a huge number of public opinions are gathering on online platform; such as: movie review, product review, Google map review and many more. So, these opinions can play an important role to know the users motive [1] and make curiosity to academic and business world [2]. Sentiment analysis

M. Prateek et al. (Eds.): NGCT 2018, CCIS 922, pp. 38–48, 2019.
https://doi.org/10.1007/978-981-15-1718-1_4

is classifying user's opinion with various class, like good-bad, positive-negative. There are many ways and procedures in sentiment analysis. We can use several levels for sentiment analysis; like Document-level, Sentence-level, Aspect-based, Comparative Sentiment Analysis, Sentiment Lexicon Acquisition [3]. Generally supervised and unsupervised learning techniques are used to classify sentiments. Each technique has several steps.

Firstly, either data have to be collected from a platform, or datasets available on online platform can be used. In document level supervised learning technique, some steps are common and many authors have used in their research work. In a dataset some words, special characters, symbols, tags are not necessary and these can keep negative effect in the accuracy result. So the data pre-processing step removes useless things from a dataset. Some approaches are used in sentiment detection, such as n-gram techniques, TF-IDF, PoS tag, Lexicon etc. [4–6]. In many research works, ensemble methods are used; like Bagging, Boosting and Stacking. The consequence of bagging method is getting more accurate prediction about the result by using various machine learning algorithms together. Boosting is a recovery technique, it recover the mistake of previous learner. Stacking concept [23] is aggregating multiple classifier algorithms. For example, Tsutsumi et al. [7] have used this concept and it gives better accuracy than others. A key process is using machine learning algorithms to classify sentiment [6]. Support Vector Machine (SVM), Naive Bayes (NB), Maximum Entropy (ME) and K Nearest Neighbors (KNN) are the most used algorithms in this area [5, 6]. To enhance performance many studies focused on using parameters like precision, recall, f-measure and accuracy [4].

In this investigation, two movie review datasets have been used. In pre-processing step, Stopwords Remove, Leveling reviews (as negative = 0 and positive = 1) is considered and then the reviews are stored in a CSV file. After that, the n-gram technique is used for feature extraction, then the datasets have been divided into 70:30 test-train split ratio. Remaining experiment has been divided into 2 levels: level 1 and level 2. Five base classifiers have been used in level 1 and in this level 10 fold cross validation is taken. Finally, XGboost is used as a meta estimator for final production in level 2.

This paper is organized into six sections. Introduction and Related Work are discussed in Sects. 1 and 2 respectively. Methodologies and demonstration of presented approach are explained briefly in Sects. 3 and 4 respectively. Performance evaluation and comparison result analysis of the model are presented in Sect. 5. The paper is concluded with the outcome of the study, limitation, and future work in Sect. 6.

2 Related Work

Tripathy et al. [4] have showed the better result from other related literature. They have found that unigram and bigram give better result from trigram, four-gram, five-gram classification and the SVM with unigram+bigram+trigram technique shows the best result comparing other techniques. Tsutsumi et al. [7] have considered a method to classify movie review document into positive or negative opinion. In this work,

Stacking concept is the main task; they have showed the accuracy of Single and Multilevel methods are less than stacking concept by using the classifier algorithms.

Su et al. [26] have worked with five base-level algorithms, six meta-level classifiers for sentiment classification. In this experiment, stacking generalization gives more effective result than majority voting. In their presented method, accuracy is measured by classifying first and last two sentences, and they have showed that this method gives better accuracy. Wang et al. [6] have used 10 public datasets to sentiment classification using ensemble methods. Three ensemble methods (Bagging, Boosting and Random Subspace), two feature (Unigram and Bigram), TP, TF-IDF and five types (NB, ME, DT, KNN and SVM) of classification algorithms have used. The combination of this methods and features (Stacking Concept) [24] gives better result comparably in this paper. Tang et al. [8] have introduced two neural network models: Conv-GRNN and LSTM-GRNN for document level sentiment classification. They have showed that neural gates help to gain better performance. Rahman et al. [24] analyzes the performances of ensemble machine learning classifiers. In the combination of Unigram and TF-IDF ADA Boost provides batter accuracy. Bigram and TF-IDF combination shows the better performance in Random Forest, Bagging Classifier and Gradient Boost algorithm.

Dey et al. [9] have compared the accuracy result of Naïve Bayes and k-NN approaches. They have use two review datasets: movie reviews and hotel reviews. Movie reviews dataset gives better accuracy with Naïve Bayes approach and hotel reviews gives almost same accuracy with both approaches. Lu and Tsou [10] have tried to combine a large sentiment lexicon and machine learning techniques for sentiment classification. They have worked three classifiers and Lexicon; combination of this classifiers and Lexicon approach (SVM & MaxEnt, SVM & SVM-Lexicon, MaxEnt & Lexicon, SVM & MaxEnt & Lexicon, SVM & SVM-Lexicon & MaxEnt & Lexicon) achieved good performance. Moraes et al. [1] have worked by six steps and with various techniques. They have used tokenizer, stopwords removal and stemming in pre- processing step. Then they have selected four feature and three classification algorithms. At the end of all, accuracy, recall, precision and F-1 interpretation are used to increase the accuracy result.

3 Methodologies

3.1 Data Collection

There are two movie review datasets have been used in this investigation. The Cornell movie-review document-level polarity dataset contains 1000 positive and 1000 negative reviews [11]. The acl IMDB Dataset has two paths: test review and train review set. Test path contains 12500 positive and 12500 negative reviews. Similarly, 12500 positive and 12500 negative reviews are considered for train path [12].

3.2 Pre-processing

This concept is common for sentiment analysis. For a text dataset, some words are not necessary. They don't play any role in the sentiment [4]. So, before preparing for classification it is needed to remove some words, special character, and numeric number. Next step for this experiment is labeling the positive and negative reviews. For this labeling, positive reviews are assigned to '1' and negative reviews are assigned to '0' as polarity. This labeled data stored in a CSV file.

3.3 Feature Extraction

In this study, n-gram method have been used, n-gram is mostly-used settings in sentiment classification [2]. It was reported in [13] that n-gram model helps to get more accurate sentiment from a sentence. The n-gram refers to unigram, bigram, trigram, four-gram, five-gram and so on. Here we worked with unigram, bigram and trigram process. By using unigram we get comparatively better result, but sometimes it fails. It works with each individual word and categorizes them. If we compare two words or three words at a time, then it provides more accurate results and it helps to increase the accuracy [4]. For a sentence 'one of the greatest movie', how n-gram process works, the format of the process is given in Table 1.

Table 1. N-gram technique

Features	Example
Unigram	It will consider single word one by one: 'one', 'of', 'the', 'greatest', 'movie'
Bigram	Where two words are considered at a time: 'one of', 'of the', 'the greatest', 'greatest movie'
Trigram	Where three word are considered at a time: 'one of the', 'of the greatest', 'the greatest movie'

Another one of the feature extraction is training and test set splitting. The training set is used to fit and tune and test set is used for final prediction. In this research, original data is spitted into test and train sets with a 70:30 split ratio [8, 14].

3.4 Used Algorithms and Techniques

Statistical machine learning works well on sentimental features and the machine learning approach to determine the sentiment [15]. This experiment is based on ensemble method and it has several base learners. In many literature review [6, 7, 15] various learner algorithms have been used. Five base learner algorithms are being used in this work: AdaBoost, Decision Tree, Gradient Boost, Extra Tree and Random Forest.

N-fold Cross Validation: N-fold cross validation is dividing the dataset into 'N' number of subsets. Most of the previous authors have used 10-fold or 5-fold cross validation in their sentimental analysis [1, 2, 5, 16, 17]. In this study, 10-fold cross

validation is used; one of the reports [5] have mentioned in their paper that a good number of researchers have used 10-fold cross validation technique.

In this study, XGBoost classifier is used as meta estimator in level 2 and get final prediction of sentiment analysis.

4 Presented Approach

Figure 1 represents the presented approach (NStackSenti) where Pre-Processing and Feature Extraction both are counted as Pre-Level. In level 1 P_{T1}, P_{T2}, …… P_{TN} are the temporary predictions for the base classifiers according with N-fold cross validation and output probability predictions or crisps predictions are expressed by VP_{O1}, VP_{O2}, VP_{O3}……., VP_{ON} which are from base classifiers.

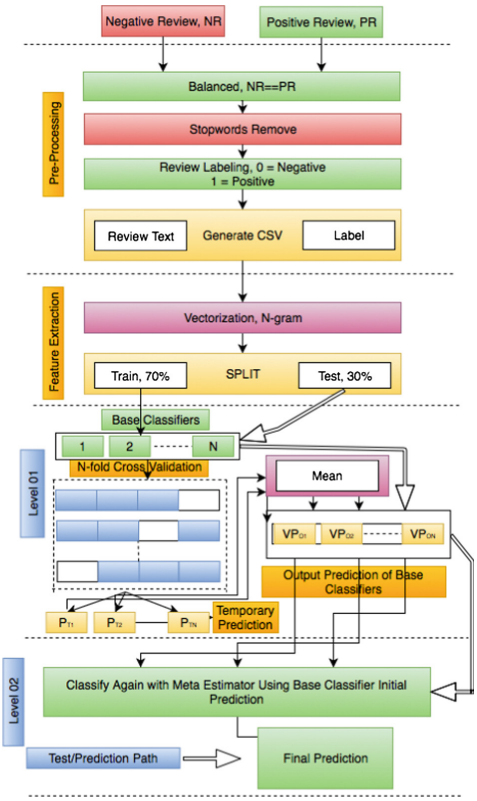

Fig. 1. Experiment procedure

The temporary predictions and crisp predictions are utilized in Level 2 for computation during the construction of our model. The concepts [25] applied in NStackSenti are:

(i) With base classifiers in Level 1, train and test set used to predict and after that the predictions obtained are used on Level 2 as features
(ii) On Level 1 or Level 2, any model can be used
(iii) In presented model N-fold cross-validation is used to avoid over fitting for training data and can predict out-of-fold (OOF) part of train part in every fold
(iv) 3–10 folds are commonly used
(v) Predict (test set):

- **Level 1:** After training, test set will be predicted and when it has done with all folds then need to apply or calculate the majority voting technique or mean (mode) of all temporary predictions $\{P_{T1}, P_{T2,} \ldots\ldots\ldots P_{TN}\}$ from each fold
- **Level 2:** During n-fold cross validation, final prediction has not done for test set. When all folds cross validation complete then on Level 2 need to fit another additional classifier called meta-estimator on full train set and performs final predictions of test set. The approach takes much time rather than others because it performs additional fitting

(vi) At Fig. 1, stacking concept has implemented after pre-level and from level 1 by applying 10-fold cross validation
(vii) Level 1 is a cycle which can be repeated to get more features for next Level.

5 Performance Evaluation and Result Analysis

This section analyzes the accuracy level and the overall performance of the document level sentiment classification using stacked generalization with n-grams and also experimental results from 2 different datasets. Bigram, Unigram and Trigram vectorization models (N-Grams) and a well-established super learner - stacked generalization have been implemented and evaluated in this study. Five well-known machine learning classifiers were used for the classification process such as Extra Tree (ET), Random Forest (RF), Gradient Boost (GB), Adaboost (ADA), Decision Tree (DT) as base classifier and eXtreme Gradient Boost (XGBoost) as meta estimator.

Figures 2, 3 and 4 depict the first level accuracy results for unigram, bigram and trigram with base classifiers applied on polarity dataset (2000) respectively. After 10-fold cross validation, the accuracy result of unigram (Fig. 2), bigram (Fig. 3) and trigram (Fig. 4) with ET, RF, GB, ADA, DT is (0.69, 0.68, 0.8, 0.75, 0.62), (0.71, 0.66, 0.8, 0.76, 0.64) and (0.69, 0.65, 0.8, 0.76, 0.63) respectively. From unigram, bigram and trigram it is observable that Gradient Boost with this three approaches give highest accuracy 0.8, which means 80% (Fig. 5).

The accuracy result of unigram, bigram and trigram with base classifiers applied on IMDb movie review dataset (50000) are represented on Figs. 6, 7 and 8 respectively. The accuracy result after 10-fold cross validation of unigram (Fig. 6), bigram (Fig. 7) and trigram (Fig. 8) with ET, RF, GB, ADA, DT is (0.77, 0.77, 0.81, 0.8, 0.73), (0.79, 0.77, 0.81, 0.8, 0.74) and (0.79, 0.77, 0.82, 0.8, 0.74) respectively. Also here Gradient Boost with unigram, bigram and trigram give the highest accuracy 0.82 which means 82% (Fig. 9).

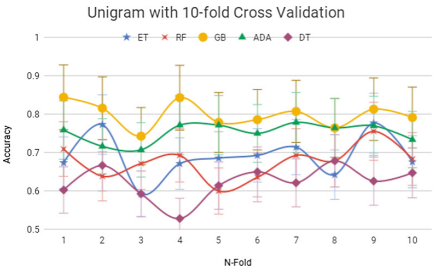

Fig. 2. Accuracy comparison with 10-fold cross validation (unigram with 2000 data)

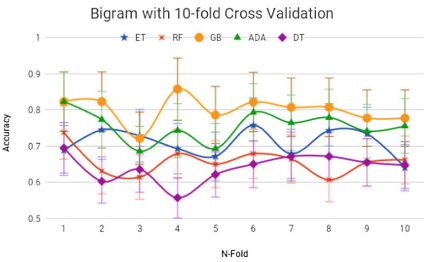

Fig. 3. Accuracy comparison with 10-fold cross validation (bigram with 2000 data)

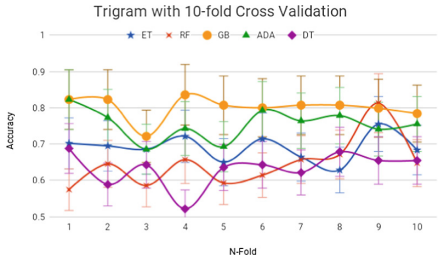

Fig. 4. Accuracy comparison with 10-fold cross validation (trigram with 2000 data)

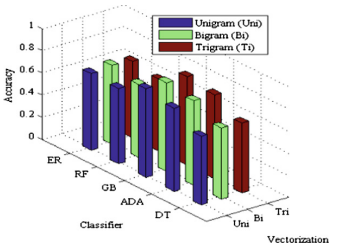

Fig. 5. Accuracy comparison after 10-fold cross validation (2000 data)

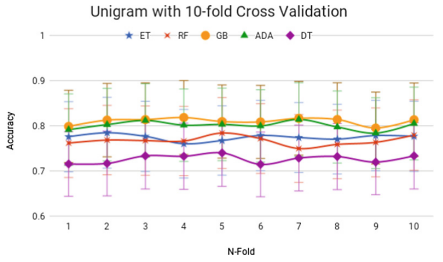

Fig. 6. Accuracy comparison with 10-fold cross validation (unigram with 50000 data)

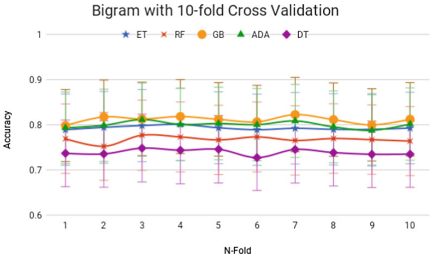

Fig. 7. Accuracy comparison with 10-fold cross validation (bigram with 50000 data)

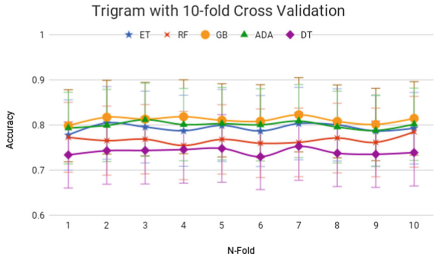

Fig. 8. Accuracy comparison with 10-fold cross validation (trigram with 50000 data)

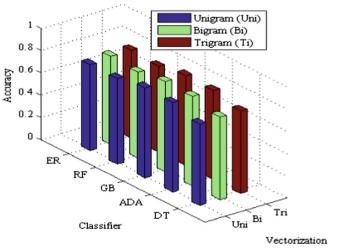

Fig. 9. Accuracy comparison after 10-fold cross validation (50000 data)

Table 2 represents the comparison of the accuracy found after successfully applied stacked generalization in two datasets. The table shows the accuracy for both datasets, as their level. It has been observed that:

(i) With base classifiers in Level 1, train and test set used to predict and after that the predictions obtained are used on Level 2 as features
(ii) For the higher amount of data, higher accuracy has been found
(iii) Accuracy within Unigram, Bigram and Trigram has increased respectively: Unigram < Bigram < Trigram
(iv) Level 02 gives better accuracy than level 01 competitively for both datasets.

Table 2. Comparison of the accuracy with Unigram, Bigram and Trigram

Dataset	Accuracy		
	Unigram	Bigram	Trigram
2000 (Level 01)	75%	76%	80%
2000 (Level 02)	83%	82.5%	83.7%
50000 (Level 01)	81%	81%	82%
50000 (Level 02)	84.2%	84.3%	86.24%

Figure 10 shows the comparison of execution time for 2000 and 50000 dataset individually by graph. Times represents in seconds. It has been observed from Fig. 10:

(i) Trigram vectorization needs maximum time to analyze sentiment with stacked generalization
(ii) For the higher amount of data, higher time has been needed
(iii) The ratio of the need of time for Unigram, Bigram and Trigram has increased respectively: Unigram < Bigram < Trigram

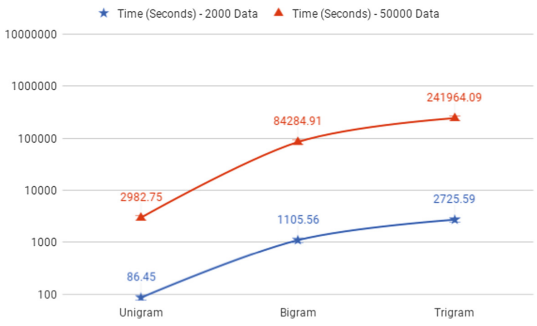

Fig. 10. Comparison of execution time for Unigram, Bigram and Trigram with 2000 and 50000 data accordingly

Table 3. Comparison among existing related works and presented approach

Authors	Approach	Algorithm	Dataset	Accuracy
Lu [10]	N-gram, lexicon and stacking generalization	Naive Bayes (NB), Maximum Entropy (ME), Support Vector Machine (SVM) and Scoring	NTCIR opinion dataset	SVM & SVM-Lexicon accuracy: 74.2%
Pang [13]	Classify the dataset using different machine learning algorithms and n-gram model	Naive Bayes (NB), maximum entropy (ME), support vector machine (SVM)	Internet movie database (IMDb)	Unigram: SVM (82.9%), Bigram: ME (77.4%), Unigram + Big: SVM 82.7%
Carvalho [22]	Used POS technique, tokenization, word classifier as feature selection	Genetic algorithm	STD and Health Care Reform (HCR)	77.2% and 75.5% respectively as dataset
Poria [18]	Multimodal sentiment analysis framework and relevant features	Naïve Bayes, SVM, ELM, and Neural Networks	YouTube dataset	Highest accuracy 78%
Sobhani [19]	Stance and sentiment detection system, n-grams and word embedding	Linear-kernel SVM	Twitter sentiment dataset in SemEval-2016	SVM: 70.3%
Poria [20]	POS-based bigram, Single-word concepts bag-of-words, s state-of-the-art	Maximum entropy, naive Bayes and SVM, ELM	Movie review dataset	67.35% for ELM and 65.67% for SVM
Keshavarz [21]	Lexicon-based approach	Genetic Algorithm	Sanders– Twitter Sentiment Corpus, Obama- McCain debate (OMD), Strict OMD, Healthcare reform (HCR), SemEval and Stanford Twitter dataset	85.71%, 80.90%, 85.85%, 82.61%, 83.81% and 84.44% respectively as dataset
Presented and Evaluated Approach	**N-Gram and Stacked Generalization**	**Extra Tree, Random Forest, AdaBoost, Gradient Boost and Decision Tree**	**Cornell Dataset (Polarity), IMDB movie review dataset (Stanford)**	**Stacking (XGBoost) Highest Accuracy: 86.24%**

Table 3 represents the comparison of the accuracy of presented approach with existing related works. To detect sentiment, this approach provides highest accuracy compare to others.

6 Conclusion

In this paper, we have presented a multilevel approach to get highest accuracy of the combination of machine learning algorithms. As a pre-level of the model, preprocessing and feature extraction have been completed. Then two level stacking concepts [24] have been applied to minimize the error rate. For text vectorization, n-grams methods such as unigram, bigram and trigram have applied. After that, the presented model achieves highest accuracy after level 02 with trigram vectorization method which is 86.24%. We have analyzed the required time for detecting the sentiment and also widely compared with various dimension to established NStackSenti.

Sentiment analysis has scope to implement tf-IDF and feature extraction. In future, we want to investigate the sentiment detection with tf-IDF and different feature selection methods.

References

1. Moraes, R., Valiati, J.F., Neto, W.P.G.: Document-level sentiment classification: an empirical comparison between SVM and ANN. Expert Syst. Appl. **40**(2), 621–633 (2013)
2. Xia, R., Xu, F., Yu, J., Qi, Y., Cambria, E.: Polarity shift detection, elimination and ensemble: a three-stage model for document-level sentiment analysis. Inf. Process. Manag. **52**(1), 36–45 (2016)
3. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Found. Trends® Inf. Retrieval **2** (1–2), 1–135 (2008)
4. Tripathy, A., Agrawal, A., Rath, S.K.: Classification of sentiment reviews using n-gram machine learning approach. Expert Syst. Appl. **57**, 117–126 (2016)
5. Tripathy, A., Rath, S.K.: Classification of sentiment of reviews using supervised machine learning techniques. Int. J. Rough Sets Data Anal. (IJRSDA) **4**(1), 56–74 (2017)
6. Wang, G., Sun, J., Ma, J., Xu, K., Gu, J.: Sentiment classification: the contribution of ensemble learning. Decis. Support Syst. **57**, 77–93 (2014)
7. Tsutsumi, K., Shimada, K., Endo, T.: Movie review classification based on a multiple classifier. In: Proceedings of the 21st Pacific Asia Conference on Language, Information and Computation, pp. 481–488 (2007)
8. 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, pp. 1422–1432 (2015)
9. Dey, L., Chakraborty, S., Biswas, A., Bose, B., Tiwari, S.: Sentiment analysis of review datasets using Naive Bayes and K-NN classifier. Int. J. Inf. Eng. Electron. Bus. **8**(4), 54–62 (2016)
10. Lu, B., Tsou, B.K.: Combining a large sentiment lexicon and machine learning for subjectivity classification. In: 2010 International Conference on Machine Learning and Cybernetics (ICMLC), vol. 6, pp. 3311–3316 (2010)

11. Pang, B., Lee, L.: A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, 271. Association for Computational Linguistics (2004)
12. Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1, pp. 142–150. Association for Computational Linguistics (2011)
13. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing, vol. 10, pp. 79–86. Association for Computational Linguistics (2002)
14. Singh, J., Singh, G., Singh, R.: Optimization of sentiment analysis using machine learning classifiers. Hum.-Centric Comput. Inf. Sci. **7**(1), 32 (2017)
15. Zheng, L., Wang, H., Gao, S.: Sentimental feature selection for sentiment analysis of Chinese online reviews. Int. J. Mach. Learn. Cybern. **9**(1), 75–84 (2018)
16. Xia, R., Zong, C., Li, S.: Ensemble of feature sets and classification algorithms for sentiment classification. Inf. Sci. **181**(6), 1138–1152 (2011)
17. Tripathy, A., Anand, A., Rath, S.K.: Document-level sentiment classification using hybrid machine learning approach. Knowl. Inf. Syst. **53**(3), 805–831 (2017)
18. Poria, S., Cambria, E., Howard, N., Huang, G.B., Hussain, A.: Fusing audio, visual and textual clues for sentiment analysis from multimodal content. Neurocomputing **174**, 50–59 (2016)
19. Sobhani, P., Mohammad, S., Kiritchenko, S.: Detecting stance in tweets and analyzing its interaction with sentiment. In: Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, pp. 159–169 (2016)
20. Poria, S., Cambria, E., Winterstein, G., Huang, G.B.: Sentic patterns: dependency-based rules for concept-level sentiment analysis. Knowl.-Based Syst. **69**, 45–63 (2014)
21. Keshavarz, H., Abadeh, M.S.: ALGA: adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs. Knowl.-Based Syst. **122**, 1–16 (2017)
22. Carvalho, J., Prado, A., Plastino, A.: A statistical and evolutionary approach to sentiment analysis. In: Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 02, pp. 110–117 (2014)
23. Wolpert, D.H.: Stacked generalization. Neural Netw. **5**(2), 241–259 (1992)
24. Rahman, S.S.M.M., Rahman, M.H., Sarker, K., Rahman, M.S., Ahsan, N., Sarker, M.M.: Supervised ensemble machine learning aided performance evaluation of sentiment classification. In: Journal of Physics: Conference Series, vol. 1060, no. 1, p. 012036. IOP Publishing, July 2018
25. Python package for stacking (machine learning technique). https://github.com/vecxoz/vecstack. Accessed 12 July 2018
26. Su, Y., Zhang, Y., Ji, D., Wang, Y., Wu, H.: Ensemble learning for sentiment classification. In: Ji, D., Xiao, G. (eds.) CLSW 2012. LNCS (LNAI), vol. 7717, pp. 84–93. Springer, Heidelberg (2013).https://doi.org/10.1007/978-3-642-36337-5_10