

RESEARCH

Open Access



Optimization of sentiment analysis using machine learning classifiers

Jaspreet Singh* , Gurvinder Singh and Rajinder Singh

*Correspondence:
profjaspreetbatth@gmail.com
Department of Computer
Science, Guru Nanak Dev
University, Amritsar, India

Abstract

Words and phrases bespeak the perspectives of people about products, services, governments and events on social media. Extricating positive or negative polarities from social media text denominates task of sentiment analysis in the field of natural language processing. The exponential growth of demands for business organizations and governments, impel researchers to accomplish their research in sentiment analysis. This paper leverages four state-of-the-art machine learning classifiers viz. Naïve Bayes, J48, BFTree and OneR for optimization of sentiment analysis. The experiments are performed using three manually compiled datasets; two of them are captured from Amazon and one dataset is assembled from IMDB movie reviews. The efficacies of these four classification techniques are examined and compared. The Naïve Bayes found to be quite fast in learning whereas OneR seems more promising in generating the accuracy of 91.3% in precision, 97% in F-measure and 92.34% in correctly classified instances.

Keywords: Sentiment analysis, Social media text, Movie reviews, Product reviews, Machine learning classifiers

Introduction to sentiment analysis

The popularity of rapidly growing online social networks and electronic media based societies has influenced the young researchers to pursue their work on sentiment analysis. These days organizations quite keen assess their customers or public opinion about their products from social media text [1]. The online service providers are hooked on assessing social media data on blogs, online forums, comments, tweets and product reviews. This assessment is exploited for their decision making or amelioration of their services or quality of products. The applications of sentiment analysis encompass the areas like social event planning, election campaigning, healthcare monitoring, consumer products and awareness services. The immoderate use of internet by business organizations all around the globe has noticed that opinionated web text has molded our business plays and socio-economic systems. The computational power is fueled by burgeon of machine learning techniques. This work focused on four text classifiers utilized for sentiment analysis viz. Naïve Bayes, J48, BFTree and OneR algorithm. The “[Machine learning techniques for sentiment analysis](#)” section of this paper provides the intuition behind the task of sentiment classification by leveraging the modeling of aforementioned four classifiers. The architecture of proposed model using four sentiment classifiers is disposed

in “Proposed methodology for optimization of sentiment prediction using weka” section. The related work with recent contributions of machine learning in the field of sentiment classification is described in “Related work” section. In “Datasets taken” section, the three manually annotated datasets are described along with their preprocessing. The experimental results and discussion of efficacies of classifiers are cataloged in “Results and discussions” section followed by the ending remarks along with a future direction in “Conclusion” section.

Levels of sentiment

Due to scarcity of opinion text available in digital form, very less research interest on computational linguistics in the last decade of twentieth century was witnessed [2–4]. The escalation of social media text on internet attracts young researchers to define the level of granularities of text. The web text is classified into three levels viz. document level, sentence level and word level. In [5], the fourth level granularity is defined by using deep convolution neural network. This fourth level is character level feature extraction approach used for extracting features of each character window from given word (Table 1).

Machine learning techniques for sentiment analysis

The social networking sites dispense their data conveniently and freely on the web. This availability of data entices the interest of young researchers to plunge them in the field of sentiment analysis. People express their emotions and perspectives on the social media discussion forums [6]. The business organizations employ researchers to investigate the unrevealed facts about their products and services. Spontaneous and automatic determination of sentiments from reviews is the main concern of multinational organizations [7–10]. The machine learning techniques have improved accuracy of sentiment analysis and expedite automatic evaluation of data these days. This work attempted to utilize four machine learning techniques for the task of sentiment analysis. The modeling of four techniques is briefly discussed below.

Naïve Bayes used for sentiment classification

The dichotomy of sentiment is generally decided by the mindset of an author of text whether he is positively or negatively oriented towards his saying [6, 11–13]. Naïve Bayes

Table 1 Levels of sentiment along with their attributes

Level	Delimiter	Depth of Granularity	Multiplicity of sentiments	Interpretation of sentiments
1. Document [15]	“\n” Newline character	Overall opinion at upper level	Single opinion of multiple entities	Overall sentiment of one document
2. Sentence [16]	“.” Period character	Factual polarity of individual sentences	Multiple opinions of multiple entities	Subjectivity classification
3. Entity or aspect level [12]	Space character or named entities	At finest level words are the target entities	Single opinion of single entity	Two-tuple as <Sentiment, target>
4. Character level [5]	Special symbols and space characters are omitted	Micro level of character embedding	Multiple opinions about single word entity	Morphological extraction of words

classifier is a popular supervised classifier, furnishes a way to express positive, negative and neutral feelings in the web text. Naïve Bayes classifier utilizes conditional probability to classify words into their respective categories. The benefit of using Naïve Bayes on text classification is that it needs small dataset for training. The raw data from web undergoes preprocessing, removal of numeric, foreign words, html tags and special symbols yielding the set of words. The tagging of words with labels of positive, negative and neutral tags is manually performed by human experts. This preprocessing produces word-category pairs for training set. Consider a word 'y' from test set (unlabeled word set) and a window of n-words (x_1, x_2, \dots, x_n) from a document. The conditional probability of given data point 'y' to be in the category of n-words from training set is given by:

$$P(y/x_1, x_2, \dots, x_n) = P(y) \times \prod_{i=1}^n \frac{P(x_i/y)}{P(x_1, x_2, \dots, x_n)} \quad (1)$$

Consider an example of a movie review for movie "Exposed". The experimentation with Naïve Bayes yields the following results.

J48 algorithm used for sentiment prediction

The hierarchical mechanism divides feature space into distinct regions followed by the categorization of sample into category labels. J48 is a decision tree based classifier used to generate rules for the prediction of target terms. It has an ability to deal with larger training datasets than other classifiers [14]. The word features for sentences of corpus taken from labeled arff file of training set are represented in the leaf nodes of decision tree. In the test set every time when a near feature qualifies the label condition of internal feature node, its level is lifted up in the same branch of decision tree. The assignment of labels to the word features of test set gradually generates different two branches of decision tree. J48 algorithm uses entropy function for testing the classification of terms from the test set.

$$Entropy(Term) = - \sum_{j=1}^n \frac{|Term_j|}{|Term|} \log_2 \frac{|Term_j|}{|Term|} \quad (2)$$

where (Term) can be unigram, bigram and trigram. In this study we have considered unigrams and bigrams. The example in the Table 2 contains bigrams like "Horrible acting", "Bad writing" and "Very misleading" are labeled with negative sentiment whereas the term "More enjoyable" reflects positive sentiment towards the movie. The decision tree of J48 algorithm for obtaining sentiment form text is represented in the Fig. 1 below.

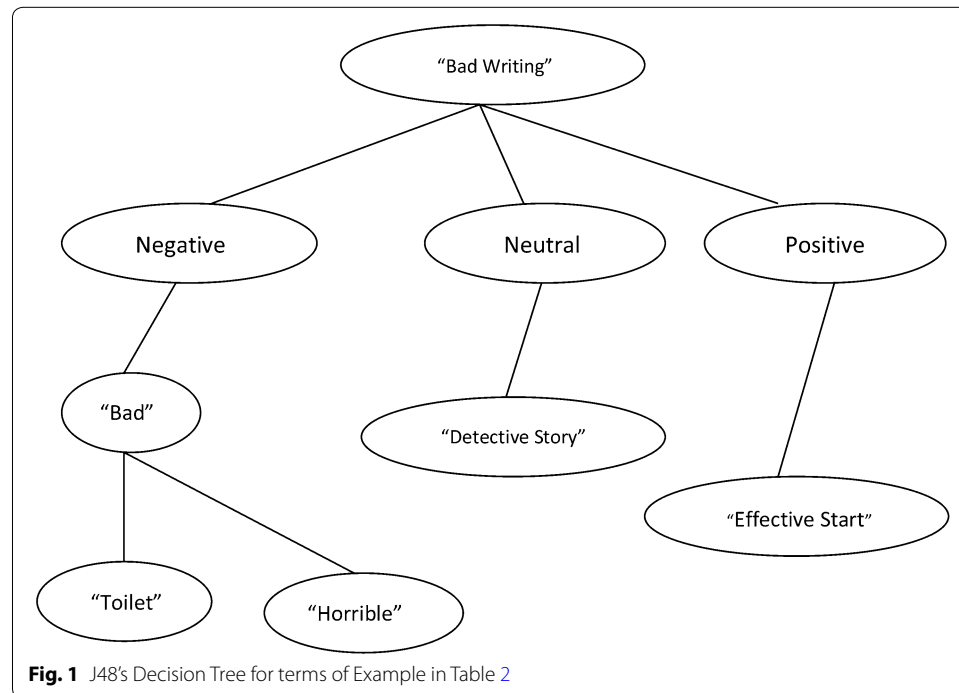
BFTREE algorithm used for sentiment prediction

Another classification approach outperforms J48, C4.5 and CART by expanding only best node in the depth first order. BFTree algorithm excavates the training file for locating best supporting matches of positive and negative terms in the test file. BFTree algorithm keeps heuristic information gain to identify best node by probing all collected word features. The only difference between J48 and BFTree classifier is the computation order in which decision tree is built. The decision tree disparate feature terms of plain

Table 2 Initial four reviews of training set and two reviews test set

Set	Sentence	Review	Class
Train	1	Horrible acting and bad writing	Neg
	2	I never did work out what the dog death scene was all about!	Neg
	3	The trailer is exciting but very misleading	Neg
	4	The basic structure of storyline is good	Pos
Test	1	The effective start and a very detective story	Pos
	2	An evening locked up in the toilet is more enjoyable	Pos

Second review of test set is negative but Naïve Bayes is lacking in context based sentiment classification



text taken from movie reviews and classify them at document level by tagging appropriate labels. BFTree extracts best node from labeled and trained binary tree nodes to reduce the error computed from information gain.

$$Info_{gain}(S, A) = Entropy(S) - \sum_{i \in V(A)} \frac{|S_i|}{S} \times Entropy(S_i) \quad (3)$$

where S is word feature term of test set and A is the attribute of sampled term from training set. $V(A)$ denotes set of all possible values of A . The binary tree stops growing when an attribute A captures single value or when value of information gain vanishes.

OneR algorithm used for sentiment prediction

OneR algorithm is a classification approach which restricts decision tree to level one thereby generating one rule. One rule makes prediction on word feature terms with minimal error rate due to repetitive assessment of word occurrences. The classification of

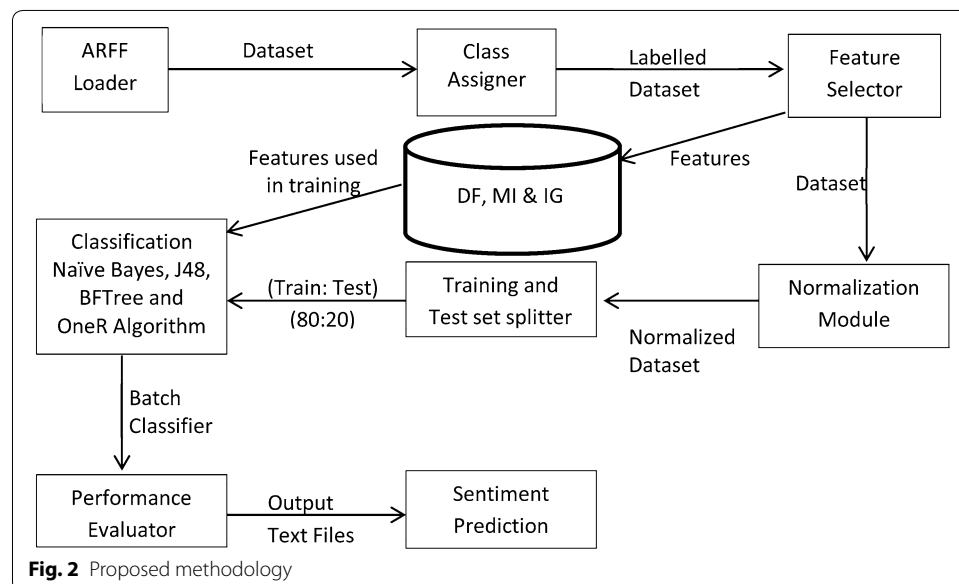
most frequent terms of a particular sentence is made on the basis of class of featured terms from training set. The demonstration of OneR algorithm for sentiment prediction with smallest error of classification is given below:

- Step 1 Select a featured term from training set.
- Step 2 Train a model using step 3 and step 4.
- Step 3 For each prediction term.
For each value of that predictor.
Count frequency of each value of target term.
Find most frequent class.
Make a rule and assign that class to predictor.
- Step 4 Calculate total error of rules of each predictor.
- Step 5 Choose predictor with smallest error.

Proposed methodology for optimization of sentiment prediction using weka

The preprocessing of raw text from web is done in python 3.5 using NLTK and bs4 libraries. Each review in the first dataset is parsed with NLTK's parser and title of the review is considered as a feature. We have obtained 15 features from first dataset and 42 features from each of second and third dataset. The CSV files generated from Python are converted to ARFF files for WEKA 3.8. Only two sentiment labels namely Pos for positive and Neg for negative are used for assigning sentences. The working methodology of proposed work for optimization of sentiment prediction is given below in Fig. 2.

After loading files with ARFF loader, the class assigner picks up appropriate class labels from dataset and performs feature selection on the basis of frequently used headings and most frequent titles. The feature selector module is implemented using three feature selection methods namely Document Frequency (DF), Mutual Information (MI) and Information Gain (IG). The mathematical modeling of these feature selection methods requires some probability distributions and statistical notations described below:



$P(w)$: Probability that a document 'd' contains term 'w'.

$P(c')$: Probability that document 'd' does not belongs to category 'c'.

$P(w, c)$: Joint probability that document 'd' contains word term 'w' of category 'c'.

$P(c/w)$: Conditional probability that a document 'd' belongs to category 'c' under the condition that 'd' contains word term 'w'.

Similarly other notations like $P(w')$, $P(w/c)$, $P(w/c')$, $P(c/w')$ and $P(c'/w)$ are taken and {c} is the set of categories.

N_1 : Number of documents that exhibit category 'c' and contain term 'w'.

N_2 : Number of documents that do not belong to category 'c' but contains term 'w'.

N_3 : Number of documents that belong to category 'c' and do not contain term 'w'.

N_4 : Number of documents that neither belong to category 'c' nor contain term 'w'.

N : Total number of document reviews.

DF method qualifies only those documents in which a higher frequency terms are considered.

$$DF = \sum_{i=1}^m N_{1i} \quad (4)$$

The MI method measures features of text by computing similarity of word terms 'w' and category 'c'.

$$Sim_{Info}(w, c) = \log \frac{P(w/c)}{P(w)} \quad (5)$$

$$MI = \log \frac{N_1 \times N}{(N_1 + N_3)(N_1 + N_2)} \quad (6)$$

The IG-construct measures similarity information for category by exploiting probabilities of absence or presence of terms in a document review.

$$IG(w) = - \sum P(c) \cdot \log P(c) + P(w) \left[\sum P(c/w) \cdot \log P(c/w) \right] \\ + P(w') \left[\sum P(c/w') \cdot \log P(c/w') \right] \quad (7)$$

The normalization module converts all letters into lowercase, removal of punctuation marks and special symbols, conversion of numbers into words, expansion of abbreviation and limiting the average length of twenty words in a sentence. Each sentence is delimited by a newline character. The Python's NLTK and bs4 libraries are used for this purpose. Data splitter take the ratio of (80:20) of (Train: Test) subsets. We have used manual splitting of dataset at the time of retrieval of data from web. The four classifiers are trained with training subsets followed by performance evaluation. The evaluation metrics taken in the experiment are precision, recall, accuracy and F-measure.

Related work

Existing approaches of sentiment prediction and optimization widely includes SVM and Naïve Bayes classifiers. Hierarchical machine learning approaches yields moderate performance in classification tasks whereas SVM and Multinomial Naïve Bayes are proved better in terms of accuracy and optimization. Sentiment analysis using neural network architectures has appeared in very few works. The sentiment prediction methods using recursive neural networks and deep convolution neural networks are bit complex in capturing compositionality of words. Extracting character level features and embeddings of complex words is found hard in many neural network architectures whereas extracting sentence level or word level features such as morphological tags and stems are more effectively achieved in convolutional neural networks. A very few researchers have used J48, BFTree and OneR for the task of sentiment prediction. These three classifiers are utilized for other classification tasks like emotion recognition from text and twitter's text categorizations. The summary of benchmarks related to machine learning techniques in terms of accuracy of classification is listed in the Table 2. SVM and Naive Bayes are proving better in terms of benchmarks than other machine learning techniques (Table 3).

Datasets taken

Three Datasets are manually annotated from <http://www.amazon.in>. First dataset consists of product reviews of Woodland's wallet are taken from 12th October 2016 to 25th October 2016 for training set containing 88 reviews and from 25th October 2016 to 30th October 2016 for testing set containing 12 randomly chosen product reviews with their sentiments prediction using four machine learning algorithms. Second dataset consists of 7465 Digital Camera reviews of Sony are taken from 01st October 2016 to 25th October 2016 for training set and 1000 reviews are from 25th October 2016 to 30th October 2016 for test dataset. Third dataset consists of movie reviews taken from <http://www.imdb.com>. It contains 2421 reviews for training set and 500 reviews for test set.

Results and discussions

The experiment is carried out by using freeware WEKA software tool for classification of sentiments in the text. Standard implementations of Naïve Bayes, J48, BFTree and OneR algorithms are exploited from WEKA version 3.8. The classification accuracy of first dataset shows 100% classification accuracy with Naïve Bayes in some of the epochs because of small size of dataset. The average of 29 epochs for all four classifiers on second and third datasets is presented in Table 4 below. Naïve Bayes shows faster learning among four classifiers whereas J48 found to be slower. OneR classifier is leading from other three classifiers in percentage of correctly classified instances. The accuracy of J48 algorithm is promising in true positive and false positive rates.

Results of classification accuracies for the test subsets with 42 and 15 attributes are recorded. The average accuracies of 29 runs on three datasets is presented in Table 5

Table 3 Benchmarks of classifier's accuracies

Author/(s) (year of publication)	Classifiers and features used	Description	Accuracy of classification (%age)
Kiritchenko and Mohammad (2016)	SVM with RBF kernel, POS, sentiment score, emoticons, embedding vectors [17]	Supervised sentiment analysis system using real-valued sentiment score to analyze social networking data	82.60 for bigrams 80.90 for trigrams
Dashtipour et al. (2016)	SVM, MNB, maximum entropy [18]	Multilingual sentiment analysis for improving the performance of sentiment classification	86.35
Tan and Zhang (2008)	Naive Bayes, SVM and k-NN [15]	A text classification system for sentiment detection of Chinese documents	82
Mohammad et al. (2015)	Automatic SVM elongated words, emoticons, negation feature, position feature. [16]	Automatic emotion detection system for 2012 US presidential election tweets	56.84
Sobhani et al. (2016)	Linear kernel SVM, n-grams and word embeddings [19]	Stance and sentiment detection system	70.3
Poria et al. (2014)	Maximum entropy, naive Bayes and SVM, ELM [20]	Concept level sentiment analysis for movie review dataset	67.35 for ELM and 65.67 for SVM
Socher (2016)	SVM, Naive Bayes [21]	Deep learning for sentiment analysis	85.4
Turney and Mohammad (2014)	Lexicon based entailment, SVM [22]	Proposed three algorithms namely balAPinc, ConVecs and SimDiffs. Tested on three different datasets i.e. KDSZ dataset created by Kotlerman et al. (2010), BBDS dataset introduced by Baroni et al. (2012) and JMTH dataset created by Jurgens et al. (2012) in SemEval-2012 Task2	68.70 for balAPinc 70.20 for ConVecs 74.50 for SimDiffs
Mohammad et al. (SemEval-2016) task 6	SVM, unigrams, n-grams, hashtags, combined features [23]	Automatic stance detection system from tweets. Where team MITRE has achieved highest level in accuracy	$F_{avg} = 67.82$
Cernian et al. (2015)	POS, SentiWordNet, Sentiment Score and Synset [7]	Proposed framework for sentiment analysis is tested on 300 product reviews from Amazon	61

Table 3 continued

Author/(s) (year of publication)	Classifiers and features used	Description	Accuracy of classification (%age)
Kiritchenko et al. (SemEval-2016) task 7	Supervised learning, random forest, PMI, Gaussian regression, NRC emoticons, SentiWordNet [6]	Automatic sentiment score determination model for general English, English Twitter corpus and Arabic Twitter corpus	Kendalls rank coeff. (K) K = 0.704 for Gen, English 0.523 for English Twitter, and 0.536 for Arabic Twitter
Pang et al. (2002)	Naive Bayes, SVM, maximum entropy classifiers with unigrams, bigrams, POS, adjectives and word frequency features [3]	Performed feature based analysis on movie reviews using three machine learning classifiers for sentiment classification	78.7 for NB, 77.7 for ME and 82.9 for SVM
Nogueira dos Santos and Gatti (2014)	Convolution neural network using word-level and character-level embedding vectors [5]	Proposed convolution neural network for classification of short text messages from Twitter using character level word embeddings	85.7 for binary classification, 48.3 for fine grained classification and 86.4 for STS corpus
Poria et al. (2015)	Ensemble classifier using POS, Sentic, negation, modification and common sense knowledge features [4]	The proposed algorithm captures contextual polarity and flow of concepts from text for dynamic sentiment analysis	88.12 for movie review dataset, 88.27 for Blitzer derived dataset and 82.75 for Amazon corpus
Poria et al. (2016)	SVM and Naive Bayes, CNN used for extracting video, audio and textual features (word embeddings and POS) [24]	Convolutional multiple kernel learning for enhancing the performance of sentiment analysis and emotion recognition	96.12 for proposed model without feature selection and 96.55 with feature selection
Wang et al. (2016)	Back propagation and stochastic gradient descent used to learn model parameters along with features such as n-gram and word vector for Valence-Arousal prediction [25]	Proposed regional convolutional neural network and long short term memory model for fine grained sentiment analysis	Pearson correlation coefficient $r = 0.778$ between CNN-LSTM and LSTM, for English text and $r = 0.781$ for Chinese text

below. All four classifiers improved in accuracies with the increase of features from 15 to 42. This shows the direct proportionality of multiple features with learning capability for machine learning algorithms.

Conclusion

This paper exploits four machine learning classifiers for sentiment analysis using three manually annotated datasets. The mean of 29 epochs of experimentation recorded in Table 4 shows that OneR is more precise in terms of percentage of correctly classified instances. On the other hand, Naïve Bayes exhibits faster learning rate and J48 reveals adequacy in the true positive and false positive rates. Table 5 reveals the truth that J48 and OneR are better for smaller dataset of woodland's wallet reviews. The preprocessing of proposed methodology is limited to extract foreign words, emoticons and elongated words with their appropriate sentiments. The future work in the task of sentiment analysis has scope to improve preprocessing with word embeddings using deep neural networks and can also extend this study through convolution neural networks.

Table 4 Performance evaluation of four classifiers

Classifiers	Time taken (s)	Correctly classified instances	Incorrectly classified instances	Accuracy TP rate	Accuracy FP rate	Precision	F-measure
Naïve Bayes	7.79	85.24	14.61	0.456	0.134	0.831	0.812
J-48	49.73	89.73	11.33	0.967	0.003	0.877	0.917
BFTree	21.12	90.07	9.03	0.892	0.025	0.883	0.721
OneR	24.45	92.34	8.66	0.9	0.061	0.913	0.97

Table 5 Test accuracies of classification algorithms for three datasets

Classification algorithm	Data set	Test accuracy with 42 features of text (%age)	Test accuracy with 15 features of text (%age)
Naïve Bayes	D1	85.127	78.814
	D2	73.517	69.298
	D3	76.782	68.864
J-48	D1	87.622	85.452
	D2	76.865	63.454
	D3	74.231	69.895
BFTree	D1	84.982	80.232
	D2	69.452	65.156
	D3	61.512	60.564
OneR	D1	87.652	84.452
	D2	76.563	72.788
	D3	65.876	63.521

D1: Woodland's wallet reviews, D2: Sony digital camera reviews, D3: IMDB movie reviews

Authors' contributions

JS made substantial contributions to conception and design, or acquisition of data, or analysis and interpretation of data. GS helped in revision and has given final approval of the version to be published. RS agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved. All authors read and approved the final manuscript.

Acknowledgements

This research was supported by Department of Computer Science, Guru Nanak Dev University, Amritsar. I thank Dr. Gurvinder Singh and Dr. Rajinder Singh for their participation in experimental work and their assistance to improve the manuscript.

Competing interests

This research work has non-financial Academic and intellectual competing interests.

Availability of data and materials

I submit that I can make the experimental data and materials available after the completion of my thesis.

Consent for publication

We hereby grant and assign all rights to Human-centric Computing and Information Sciences for publication.

Ethics approval and consent to participate

The article submitted is an original work and has neither been published in any other peer-reviewed journal nor under consideration for publication by any other journal.

Funding information

No funding was received from any funder.

Publisher's Note

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

Received: 24 December 2016 Accepted: 4 September 2017

Published online: 11 December 2017

References

1. Parvathy G, Bindhu JS (2016) A probabilistic generative model for mining cybercriminal network from online social media: a review. *Int J Comput Appl* 134(14):1–4. doi:[10.5120/ijca2016908121](https://doi.org/10.5120/ijca2016908121)
2. Cambria E, White B (2014) Jumping NLP curves: a review of natural language processing research. *IEEE Comput Intell Mag* 9(2):48–57. doi:[10.1109/mci.2014.2307227](https://doi.org/10.1109/mci.2014.2307227)
3. Pang B, Lee L, Vaithyanathan S (2002) Thumbs up? In: Proceedings of the ACL-02 conference on empirical methods in natural language processing—EMNLP '02. doi:[10.3115/1118693.1118704](https://doi.org/10.3115/1118693.1118704)
4. Poria S, Cambria E, Gelbukh A, Bisio F, Hussain A (2015) Sentiment data flow analysis by means of dynamic linguistic patterns. *IEEE Comput Intell Mag* 10(4):26–36. doi:[10.1109/mci.2015.2471215](https://doi.org/10.1109/mci.2015.2471215)
5. Nogueira dos Santos C, Gatti M (2014) Deep convolution neural networks for sentiment analysis of short texts. In: Proceedings of COLING 2014, the 25th international conference on computational linguistics. p 69–78
6. Kiritchenko S, Mohammad S, Salameh M (2016) SemEval-2016 task 7: determining sentiment intensity of English and Arabic phrases. In: Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016). doi:[10.18653/v1/s16-1004](https://doi.org/10.18653/v1/s16-1004)
7. Cernian A, Sgarciu V, Martin B (2015) Sentiment analysis from product reviews using SentiWordNet as lexical resource. In: 2015 7th international conference on electronics, computers and artificial intelligence (ECAI). doi:[10.1109/ecai.2015.7301224](https://doi.org/10.1109/ecai.2015.7301224)
8. Hammer HL, Solberg PE, Øvrelid L (2014) Sentiment classification of online political discussions: a comparison of a word-based and dependency-based method. In: Proceedings of the 5th workshop on computational approaches to subjectivity, sentiment and social media analysis. doi:[10.3115/v1/w14-2616](https://doi.org/10.3115/v1/w14-2616)
9. Zadeh L (2006) Toward human-level machine intelligence. In: 2006 18th IEEE international conference on tools with artificial intelligence (ICTAI'06). doi:[10.1109/ictai.2006.114](https://doi.org/10.1109/ictai.2006.114)
10. Joachims T (2002) Text classification. Learning to classify text using support vector machines. p 7–33. doi:[10.1007/978-1-4615-0907-3_2](https://doi.org/10.1007/978-1-4615-0907-3_2)
11. Wanton TM, Porrata AP, Guijarro AM, Balahur A (2010) Opinion polarity detection—using word sense disambiguation to determine the polarity of opinions. In: Proceedings of the 2nd international conference on agents and artificial intelligence. doi:[10.5220/0002703504830486](https://doi.org/10.5220/0002703504830486)
12. Xia Y, Cambria E, Hussain A, Zhao H (2014) Word polarity disambiguation using bayesian model and opinion-level features. *Cogn Comput* 7(3):369–380. doi:[10.1007/s12559-014-9298-4](https://doi.org/10.1007/s12559-014-9298-4)
13. Dey L, Chakraborty S, Biswas A, Bose B, Tiwari S (2016) Sentiment analysis of review datasets using Naïve Bayes' and K-NN classifier. *Int J Inform Eng Electron Bus* 8(4):54–62. doi:[10.5815/ijieeb.2016.04.07](https://doi.org/10.5815/ijieeb.2016.04.07)
14. Nie CY, Wang J, He F, Sato R (2015) Application of J48 decision tree classifier in emotion recognition based on chaos characteristics. In: Proceedings of the 2015 international conference on automation, mechanical control and computational engineering. doi:[10.2991/amcce-15.2015.330](https://doi.org/10.2991/amcce-15.2015.330)
15. Tan S, Zhang J (2008) An empirical study of sentiment analysis for Chinese documents. *Expert Syst Appl* 34(4):2622–2629. doi:[10.1016/j.eswa.2007.05.028](https://doi.org/10.1016/j.eswa.2007.05.028)

16. Mohammad SM, Zhu X, Kiritchenko S, Martin J (2015) Sentiment, emotion, purpose, and style in electoral tweets. *Inf Process Manage* 51(4):480–499. doi:[10.1016/j.ipm.2014.09.003](https://doi.org/10.1016/j.ipm.2014.09.003)
17. Kiritchenko S, Mohammad SM (2016) sentiment composition of words with opposing polarities. In: Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies. doi:[10.18653/v1/n16-1128](https://doi.org/10.18653/v1/n16-1128)
18. Dashtipour K, Poria S, Hussain A, Cambria E, Hawalah AY, Gelbukh A, Zhou Q (2016) Multilingual sentiment analysis: state of the art and independent comparison of techniques. *Cogn Comput* 8(4):757–771. doi:[10.1007/s12559-016-9415-7](https://doi.org/10.1007/s12559-016-9415-7)
19. Sobhani P, Mohammad S, Kiritchenko S (2016) Detecting stance in tweets and analyzing its interaction with sentiment. In: Proceedings of the 5th joint conference on lexical and computational semantics. doi:[10.18653/v1/s16-2021](https://doi.org/10.18653/v1/s16-2021)
20. Poria S, Cambria E, Winterstein G, Huang G (2014) Sentic patterns: dependency-based rules for concept-level sentiment analysis. *Knowl Based Syst* 69:45–63. doi:[10.1016/j.knosys.2014.05.005](https://doi.org/10.1016/j.knosys.2014.05.005)
21. Socher R (2016) deep learning for sentiment analysis—invited talk. In: Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis. doi:[10.18653/v1/w16-0408](https://doi.org/10.18653/v1/w16-0408)
22. Turney PD, Mohammad SM (2014) Experiments with three approaches to recognizing lexical entailment. *Nat Lang Eng* 21(03):437–476. doi:[10.1017/s1351324913000387](https://doi.org/10.1017/s1351324913000387)
23. Mohammad S, Kiritchenko S, Sobhani P, Zhu X, Cherry C (2016) SemEval-2016 task 6: detecting stance in tweets. In: Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016). doi:[10.18653/v1/s16-1003](https://doi.org/10.18653/v1/s16-1003)
24. Poria S, Chaturvedi I, Cambria E, Hussain A (2016) Convolutional MKL based multimodal emotion recognition and sentiment analysis. In: 2016 IEEE 16th international conference on data mining (ICDM). doi:[10.1109/icdm.2016.0055](https://doi.org/10.1109/icdm.2016.0055)
25. Wang J, Yu L, Lai KR, Zhang X (2016) Dimensional sentiment analysis using a regional CNN-LSTM model. In: Proceedings of the 54th annual meeting of the association for computational linguistics, vol 2: short papers. doi:[10.18653/v1/p16-2037](https://doi.org/10.18653/v1/p16-2037)

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
