Taylor & Francis
Taylor & Francis Group

International Journal of Computers and Applications

ISSN: 1206-212X (Print) 1925-7074 (Online) Journal homepage: https://www.tandfonline.com/loi/tjca20

An empirical research on sentiment analysis using machine learning approaches

Monika Kabir, Mir Md. Jahangir Kabir, Shuxiang Xu & Bodrunnessa Badhon

To cite this article: Monika Kabir, Mir Md. Jahangir Kabir, Shuxiang Xu & Bodrunnessa Badhon (2019): An empirical research on sentiment analysis using machine learning approaches, International Journal of Computers and Applications

To link to this article: https://doi.org/10.1080/1206212X.2019.1643584

Published online: 24 Jul 2019.

INTERNATIONAL JOURNAL OF COMPUTERS AND APPLICATIONS
https://doi.org/10.1080/1206212X.2019.1643584

An empirical research on sentiment analysis using machine learning approaches

Monika Kabir [a], Mir Md. Jahangir Kabir [b], Shuxiang Xu [c] and Bodrunnessa Badhon [d]

[a]Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Rajshahi, Bangladesh; [b]Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Rajshahi, Bangladesh; [c]School of Engineering and ICT, University of Tasmania, Tasmania, Australia; [d]Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Rajshahi, Bangladesh

ABSTRACT

Nowadays users of social networks are very much interested in expressing their opinions about different sorts of products or services in social media which leads to the growth of user-generated web contents. Their reviews on social media have a significant impact on customers for making effective and optimal decisions for buying products or using services. In sentiment analysis, most of the used approaches are based on machine learning techniques. In this paper, the well-known methods of machine learning are reviewed and compared against each other. Then the comparative studies on the performance of these techniques on online user reviews that come from multiple industry domains are performed. The experiments involve many different data sets from various domains including Amazon, Yelp and IMDb. Well-known methods such as Support Vector Machine, Decision Tree, Bagging, Boosting, Random Forest and Maximum Entropy are implemented in the experiments. Based on the experimental results it is found that users can extract applicable information from review data sets for business intelligence and better product sales production, and that Boosting and Maximum Entropy outperform the other examined machine learning algorithms for detecting sentiments in online user reviews.

ARTICLE HISTORY
Received 29 December 2018
Accepted 9 July 2019

KEYWORDS
Sentiment classification; text mining; machine learning

1. Introduction

With the emerging use of the internet, people are showing their sentiment through different e-commerce websites, blogs or social networks. From the reviews of the e-commerce website, people can know and compare products. To get knowledge from the review data, data mining techniques can be used for mining interesting, valid and significant patterns for the users [1–3]. Different data mining techniques can be used for sentiment analysis.

Analyzing sentiment or extracting opinion is the mathematical study of an individual's attitudes, appraisals, opinions, and emotions toward objects, issues, events, and topics [4]. It is one of the extensively pursued fields of Natural Language Processing (NLP) and text processing. For sentiment analysis, it is very important to define the polarity of textual context which can be positive or negative. Methods of sentiment analysis can be categorized predominantly [5] as machine learning [6], Lexicon-based [7] and hybrid [8].

In this research, the problem of determining the polarity of sentences of customer review data is studied. We have used the sentence-level classification for analysis. Given different sets of customer reviews, the task involves three subtasks. First of all, three real-world review data sets of different domains are used to determine the polarity of each sentence. Secondly, different machine learning approaches are applied to each data set and a comparative analysis is presented, and finally, the discovered information is summarized.

The following example is used to illustrate the sentiment analysis process on sentence-level. Typical review sentences look like the following:

Product review:

Positive: I bought this to use with my Kindle Fire and absolutely loved it!
Negative: This is a simple little phone to use, but the breakage is unacceptable.

Movie review:

Positive: In other words, the content level of this film is enough to easily fill a dozen other films.
Negative: A very, very slow-moving, aimless movie about a distressed, drifting young man.

For product review, in review 1, the potential customer felt positive about the product and in review 2 he gave a negative opinion about a feature. Similarly, for the movie review, in review 1, the potential customer gave a positive opinion about the movie and in review 2 his perspective was a negative opinion about it. So it is very important to get to know the manufacturer as well as the costumers about their viewpoints (whether positive or negative).

For this study, structured reviews are used for training and testing purpose. Different classifiers such as Support Vector Machine (SVM), Maximum Entropy etc. are used for identifying and classifying review sentences from the data sets that are collected from Amazon, Yelp and IMDb. A detailed analysis of those approaches is provided and their effects explained for understanding the significance of reviews. However, a useful summary is produced from the results of different approaches with these data sets.

CONTACT Monika Kabir ✉ monikakabir11@gmail.com 🖃 Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Rajshahi 6204, Bangladesh

As indicated above, the main contributions of this paper are:

(1) Different machine learning approaches- Support Vector Machine, Bagging, Boosting, Random Forest, Decision Tree, and Maximum Entropy are implemented to classify sentiment in sentence-level.
(2) A comparative study is performed considering various evaluation criteria that are- accuracy, recall, precision, *f*-score, ensemble agreement, and runtime of each algorithm.
(3) The results are analyzed with three real-world data sets from three different domains.
(4) Boosting and Maximum Entropy outperform the other examined machine learning algorithms for detecting sentiments in online user reviews.

The remaining part of the paper is structured as follows. After reviewing related work in Section 2, we demonstrate different machine learning algorithms in Section 3. Section 4 presents the detailed study of investigational setup to perform sentiment analysis. We present and analyze our experimental results based on different techniques with a large number of review data in Section 5. Section 6 concludes this report, discussing open issues and possible avenues of further research for Sentiment Analysis (Figures 1–5).

2. Related work

A large number of papers focused on addressing the problem of sentiment classification. They used different techniques for classifying the sentiments of individuals at various levels. Different machine learning [9] and lexicon-based [7] approaches have been applied for sentiment analysis. A hybrid approach was proposed by Zhang et al. [10] for sentiment analysis by combining both lexicon and learning based approaches and implemented on Twitter data. Not only supervised techniques but also unsupervised techniques can achieve a good accuracy which was presented by Turney [11]. Dave, Lawrence, and Pennock [12] designed a model of semantic orientation for positive and negative words with scoring which is used to classify the review data.

Sentiment analysis is highly dependent on the topic domain and features of data. Hu and Liu [13] presented a feature based summarization to determine the polarity of the review data. They used data mining and natural language processing techniques for identifying, summarizing and classifying the data as positive or negative. A lexicon-based approached was also studied by Taboada et al. [7] in which dictionaries of words annotated with their semantic orientation. An unsupervised hierarchical Bayesian model was proposed by Lin and He [14]. The document-level sentiment was classified by their approach to mine mixture of different topics. This is an extended version of the joint sentiment/topic (JST) and Latent Dirichlet Allocation (LDA) model for detecting the sentiment. For classifying cross-domain sentiment, Pan et al. [15] applied Spectral Feature Alignment approach. A comparative analysis of both machine learning and lexicon-based approach for different web service was performed by Serrano-Guerrero et al. [16]. Since the existing techniques were not suitable, Liu, Hu, and Cheng [17] proposed a novel technique to select the features of the products by analyzing the review data. They compared the opinions of the customers for different products and demonstrated the results.

Machine learning approach, rule-based approach, and lexicon-based approach are used widely for sentiment polarity identification. Different machine learning algorithms such as -Naive Bayes Classifier, Max Entropy Classifier, Boosted Trees Classifier and Random Forest Classifier were implemented on text review data and after classification, an aggregate score was presented [18]. Rule-based and lexicon-based approaches were compared with different machine learning approaches and a summary was represented in [19]. A comparative analysis is performed on SVM and ANN in document-level sentiment analysis by Moraes et al. [20]. Tripathy et al. proposed a hybrid approach for combining SVM and ANN where SVM is used for feature selection and ANN to perform accuracy measurement [21]. This hybrid approach provides better accuracy than single machine learning approaches. Multi-label sentiment classification is performed by Liu et al. and 11 different states are considered and compared with different approaches [22]. Recently domain specific aspect-based sentiment analysis is widely used. Based on topic modeling and unsupervised technique an approach is proposed to sentiment classification in aspect level. This approach is performed and analyzed with different domain and languages [23].

3. Methodologies

Sentiment classification process can be performed with the help of lexicon-based and machine learning-based approaches. Many traditional and ensemble machine learning techniques can classify the polarity of text documents. Here, six machine learning approaches are examined for polarity identification from review data.

As indicated above, the task of this study is performed through three main steps:

(1) Useful and significant patterns from different types of review data sets are mined. These review data sets are of different domain and size. However, for better accuracy, these data sets are preprocessed and different techniques are performed.
(2) Sentence-level classification is used for each opinion sentence of the reviews. The polarity of each sentence is positive or negative. Opinion sentences and their corresponding polarity are used for determining the polarity of each review data with different machine learning approaches, namely, Support Vector Machine, Bagging, Boosting, Random Forest, Decision Tree, and Maximum Entropy. At last, comparative analyses for these techniques with different data sets are evaluated based on performance.
(3) Finally, the results of the previous steps are summarized and presented.

3.1. Maximum entropy classifier

The Maximum Entropy is included in exponential models class and works as a probabilistic classifier. Within the imposed constraints, it calculates probabilities by making the least assumption. Binary feature and their corresponding results from the training process helps to attain the constraints [24].

The Maximum Entropy classification requires a set of features where each feature defines a class. For instance, in the review text documents, words can be considered as features that belong to the documents in that class. A feature *f* is a binary function that maps to '1' if a document belonging to a category contains the feature (word) [25]. For class *c*, data set *d* and weighting vector *w*, this model can be represented as,

$$P_{ME}(c|d,w) = \frac{e^{\sum_i w_i f_i(c,d)}}{\sum_c e^{\sum_i w_i f_i(c,d)}}$$

For text classification, Maximum Entropy can be used by identifying the class variable in the document and calculating their conditional distribution. Nigam, John, and McCallum [26] made a comparative accuracy of Maximum Entropy with Naïve Bayes and showed that ME sometimes performs better based on feature selection.

Maximum Entropy classifier is best suited when any assumption or prior distribution is unsafe and the features of it are not related to one another. Based on this, it is useful for classifying text as words where the main features of a document are dependent on each other. So, this approach can be used to classify text as well as sentiment analysis. But the drawback of Maximum Entropy technique is, it requires more time than the other approaches for training.

3.2. Support vector machine

SVM is one of the dominant classifier algorithms. The main concept of SVM is to construct a linear separator in a hyperplane that can best separate the input data into different classes. The hyperplane is learned by using an optimization procedure in training data. After that, the largest margin is selected in the hyperplane so that it can optimally separate data set into classes [27]. For input vector x, weight vector w and bias b, the linear separator can be defined as

$$y_i(w.x_i + b) \geq 1 \quad \text{for all } 1 \leq i \leq n$$

For sentiment analysis, SVM performs better than many other machine learning approaches [28]. Support vector is selected from training data, and with this support vector, test data is differentiated into different classes. Multi-class SVM is introduced for sentiment analysis that produces multiple variants of SVM [29]. Centroid classification approach is also proposed for polarity categorization. The centroid vector is calculated for the training class and test class, and test data is assigned to its defined class based on similarities.

SVM is well suited for sentiment analysis because of the nature of the text. For the high dimensional input space of review data, a large number of features have to be dealt with. Moreover, most of the document vectors contain a small number of non-zero entries. This high dimensional and sparse data can be handled very well by SVM. As most of the text is linearly separable, it shows very good performance on sentiment analysis [30]. But SVM is computationally expensive and a very slow process.

3.3. Bagging

Bootstrap Aggregation or Bagging is a very dominant ensemble method. In this method, it generates individuals by redistributing training data randomly. For ensemble, these individuals are used to train classifiers and classifying test data [31]. In this technique, for different subsample D and nth bootstrapped training set, each time f_n is calculated and their average result is considered as a final result. This can be represented as

$$f_{bagging}(x) = \frac{1}{D}\sum_{n=1}^{D} f_n(x)$$

Fersini et al. presented several traditional machine learning methods for sentiment analysis and made a comparison of these methods with a Bagging related ensemble. In addition, the complexity of these models was also evaluated [32]. Again, three familiar ensemble techniques- Bagging, Boosting and Random Subspace were implemented by Wang et al. for sentiment analysis and a comparative study was performed based on performance [20]. It is also reported that, for noisy data, Bagging technique shows consistent results, but boosting technique is relatively reactive.

For sentiment mining, ensemble learning techniques can improve the accuracy. But it is very important to select the best ensemble model that can provide maximum accuracy for any data. Aside from all these advantages, the limitation of this method is, it requires more time for training and classification than others [33].

3.4. Boosting

Boosting is a good ensemble learning method for text classification. For classification, it uses weak learners to transform into a strong one by putting weights. This weaker components become trained to decrease variance and achieve better performances [34]. Mathematically,

$$f(x) = f_0(x) + \sum_{n=1}^{N} \theta_n \varnothing_n(x)$$

where f_0 is initial predicted value, for nth iteration θ_n represent weight, and $\varnothing_n(x)$ is base estimator.

For sentiment analysis, ensemble learning performs very well. An improved Boosting algorithm is presented by adding decision-making conditions for sentiment analysis on an article of newspaper [35]. Silva et al. used sentiment analysis for Twitter data by taking lexicon, true hashing and specific syntactic as input features. They also combined SVM and multinomial Naïve Bayes (MNB) algorithm with the AdaBoost algorithm and showed that MNB performs better as a component for AdaBoost [36].

Boosting method can show a great performance in sentiment analysis. It is very easy and can combine with any method for finding base classifier. For an underfit model it works well by reducing variances. But Boosting method is very sensitive to noise [37].

3.5. Random forest

Random Forest is very simple supervised ensemble learning method for classification. For prediction and decision making, a forest is created by constructing multiple trees. From the random subsets of features, the best one is selected for splitting a node in the forest [38]. From p input variables, m numbers of variables are selected as $m < p$. Finally, the average prediction value of its trees is used for the final prediction. Mathematically,

$$F(x) = \frac{1}{J}\sum_{j=1}^{J} f_j(x)$$

Amrani et al. proposed a hybrid approach by making the combination of Random Forest and SVM techniques to classify data set. After constructing a forest, random tree selects the best feature to classify test data and make predictions. This classifier is independent on input parameter and default parameters are used for classifying a data set. For polarity determination, a binary classifier is used where 1 represents 'positive' polarity and 0 represents 'negative' polarity [39].

The Major advantages of this algorithm are that it can generate the significance of word spontaneously to classify text and for non-significant words, it shows robust performance [28]. But the main drawback of this method is that it shows a high degree of overfitting problem with the increase of complexity [18].

3.6. Decision tree classifiers

Decision Tree is one of the most used machine learning approaches which are based on predictive modeling. To construct the tree structure and for classification, the value of each variable is calculated. To measure the level of uncertainty of an element, *Entropy* can be used. For probability p_i of any class p, *Entropy* can be mathematically represented as

$$E(T) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

The class level of the tree is represented by leaves and a combination of features is represented by a branch. For classification, the best attribute is selected and data is split recursively until the nodes reach the minimum value [40]. Based on *Entropy E,* the classes are divided into multiple branches measuring *Information Gain*,

$$I(T, a) = E(T) - E(T|a)$$

Decision Tree classifier can be used for text classification with a small variety of different packages. A successor of Decision Tree algorithm C5 was used for document classification [41]. Lewis and Ringuette made a comparative study of Decision Tree and Naïve Bayes algorithm for text categorization [42]. Here, to perform split in a single attribute, the existence of phrases or words in a tree at a specific node was inspected.

Decision Tree algorithm is rarely used for sentiment classification. It works fine for smaller dimension space to preserve the semantics. For high dimensional data such as customer reviews, it may give inconsistent results [43].

4. Experimental setup

The procedure of sentiment analysis can be divided mainly into four steps [44]. These steps are discussed in this section.

4.1. Steps of sentiment analysis

(1) *Data collection:* Data collection is the very first step to sentiment analysis. Data can be collected from different websites, social networks, blogs, review sites and so on. This unstructured and unorganized text data of individual domains are used for the next steps.
(2) *Text preprocessing:* Text preprocessing helps to make data useable for further analysis [45]. Removal of digits, punctuation, stop words, case conversion, and tokenization make the unprocessed data organized and efficient.
(3) *Sentiment classification:* In this step, the polarity of each sentence is classified into positive or negative.
(4) *Presentation of output:* At the last step, the unstructured text data is transformed into significant and useable information. The resultant output is represented graphically using charts and purposeful information retrieved from it.

4.2. Data sets collection

In order to evaluate and compare the different approaches of sentiment analysis, customer review data from the UCI machine learning repository[1] is used.

This review data contains three different real-world data sets which are collected from Amazon, Yelp and IMDb. Each data set contains Sentiment Labeled Sentences with positive or negative sentiment. The characteristic of the data set is text type containing 3000 number of instance (Table 1). It contains positive or negative sentiment labeled sentences with corresponding sentence scores. The score is either 1 (for positive) or 0 (for negative).

Table 1. Data sets used for the experiments.

Data Source	Category	No of Sentences (positive/negative)	No of words
Amazon	Cell phone and accessory	1000 (500/500)	11,248
Yelp	Restaurant	1000 (500/500)	11,894
IMDb	Movie	1000 (500/500)	15,366

The first data set is of cell phone and accessory category that is collected from Amazon.com. It contains reviews about cell phones that are sold on Amazon. In this review data set, the 1000 labeled sentence consists of total 11,248 words. The next data set is about restaurant reviews that are extracted from Yelp which consists of 11,894 words. The third and last sentiment data set regards movie review that is referred from IMDb. This is the largest data set with 15,366 words for 1000 sentences.

For each of the data sets, 1000 labeled sentences are split off for testing and training. Among them, 50% represent positive polarity and the rest 50% represent negative polarity. These data sets are used to evaluate sentence-level classifiers and model training.

4.3. Text preprocessing

Text preprocessing is an important part of opinion mining. The unstructured and unorganized text data needs further processing to extract predominant knowledge from it [46]. The key steps of preprocessing are syntax and semantic preprocessing.

4.3.1. Stop words elimination

Stop words are extremely common words but contain little significance for any analysis. The elimination of stop words makes the text less heavy and reduces dimensionality. Determiners, coordinating conjunctions, prepositions etc. are some types of stop words that are not meaningful for the documents. Some customized keywords can also be used based on the content of the testing data. For text mining applications, stop words are eliminated from documents as those words are not appraised as keywords.

4.3.2. Stemming

Stemming is the basic text processing method that removes prefixes, infixes or suffixes from a word. The aim of stemming is to reduce the conjugation forms of each word into a root. For example, English words like 'complicate' can be infected with a morphological suffix to produce 'complicatedly,' 'complication.' They share the same stem 'complicate.' Stem helps to map conjugational forms of word which become beneficial for information retrieval.

4.3.3. Syntactic preprocessing

Syntactic preprocessing is a very important step to prepare a text for further analysis. In this step, the following tasks are performed: set all characters to lowercase, remove or convert numbers to textual representations, remove punctuation, strip white space, and replace abbreviation, contraction, and symbol.

4.3.4. Weighting

Tf-Idf weighting is very important preprocessing step for large text with limited term diversity. Term frequency-inverse document frequency (Tf-Idf) is used here to measure the importance of a word in text data based on the number of times it appears in the document. The least weighted unimportant words are removed from the text by this process.

4.4. Classification process

Our experiment is accomplished using customer reviews for three different data sets. The reviews were collected from Amazon, Yelp and IMDb. Each of the reviews contains the text of sentence-level opinion and a polarity. After preprocessing, different machine learning techniques such as SVM, Bagging, Boosting, Random Forest, Decision Tree, and Maximum Entropy have been implemented (Table 2). These techniques are applied by using a machine learning package named 'RTextTools' in R. For sentiment polarity

Table 2. Parameters considered for the comparison.

Algorithm	Parameter
Maximum entropy	l1_regularizer = 0.0, l2_regularizer = 0.0, use_sgd = FALSE, set_heldout = 0, verbose = FALSE
SVM	method = 'C-classification,' cross = 0, cost = 100, kernel = 'radial'
Random forest	ntree = 200
Boosting	maxitboost = 100

analysis in sentence-level, each data set was divided into training and testing classes. The preprocessed review data set is split up into two classes with 4 cross-validations, for training and testing respectively.

4.5. Evaluation criteria

Accuracy, Performance, ensemble-agreement, and runtime is used to evaluate the result for different machine learning algorithms on different data sets. To measure the accuracy of different classifier 4-fold cross-validation is performed. Recall, precision, and f-score are used to evaluate performance. Ensemble coverage and recall are measured to identify the ensemble agreement of each label. At last execution time is calculated for each classifier to assess their performances.

4.5.1. K-fold cross-validation

Concerning the evaluation criteria, the accuracy measured by k-fold cross-validation. In this process, the sample data is divided into k subsample of equal size. Of the k subsample, $k - 1$ is used for training and the rest single one is used for testing. This process is repeated k times so that each subsample can be used as test data at least once. Then the folded results are averaged to calculate the final prediction.

4.5.2. Recall, precision, and f-score

For assessment of performance, different metrics - precision, recall and f-score along with the accuracy are used for comparing the result obtained by different machine learning approaches [47]. Those indices are computed based on four binary classification test-

True positive (tp): The case when a value is correctly classified as positive.

True negative (tn): The case when a value correctly classified as negative.

False positive (fp): The case when a value is wrongly classified as positive.

False negative (fn): The case when a value is wrongly classified as negative.

Precision (p): Precision is the ratio of true positive and all positive documents. High Precision indicates an example labeled as positive is indeed positive. Precision can be represented by the following relation,

$$p = \frac{tp}{tp + fp}$$

Recall (r): Recall is the ratio of true positive and actual positive documents. High Recall indicates the class is correctly recognized. Recall can be defined as

$$r = \frac{tp}{tp + fn}$$

F-score (f): Finally f-score measures accuracy by calculating the weighted average of precision and recall, and it is computed as

$$f = \frac{2 * p * r}{p + r}$$

4.5.3. Ensemble agreement

The purpose of using ensemble agreement is to increase accuracy at each label. For different class labels, multiple algorithms prediction similarity can be measured by ensemble agreement. Both coverage and recall accuracy is used here to evaluate the performance. Coverage can be represented as follows,

$$coverage = \frac{k}{n}$$

Where n represents the total number of event and k is the percentage that meets ensemble threshold. Recall value increase with the decrease of coverage.

4.5.4. Runtime

Runtime is very important for measuring performance. Sys.time() function is used here to measure the running time of different algorithms on each data sets.

5. Experiment results and discussions

In this section, we demonstrate our experiment results and assess performance based on various machine learning classifiers. We run the classifiers on each data set and compare the performance based on cross-validation, recall, precision, f-score, ensemble agreement, and runtime.

5.1. Accuracy analysis

The results for the experiment are obtained by applying cross-validation with 4 folds, computing the accuracy for each fold and taking the mean result. Here we have used 90% of corpora for training for each algorithm.

Tables 3–5 present the accuracies of different algorithms with review data sets collected from Amazon, Yelp and IMDb. We have evaluated our results with different data sets by comparing the performance of different machine learning models (Figure 1).

5.2. Recall, precision, and f-score analysis

Table 6 gives the recall, precision, and f-score of the different algorithms with different review data sets. For each algorithm, the performance is evaluated in every step. In the following table, the record

Table 3. Accuracy measurement of Amazon review data with 4-fold cross-validation.

Algorithm	Maximum entropy	SVM	Bagging	Boosting	Random forest	Decision tree
1st fold	**0.9715447**	0.7769517	0.7647059	0.9090909	0.7773585	0.7434783
2nd fold	**0.9465649**	0.7991968	0.7500000	0.9101562	0.7924528	0.7126437
3rd fold	**0.9395161**	0.7929515	0.7509881	0.8955823	0.7903226	0.7755906
4th fold	0.9098361	0.7960784	0.742616	**0.9166667**	0.8198198	0.7568627
Mean accuracy	**0.9418654**	0.7912946	0.7520775	0.9078740	0.7949884	0.7471438

Bold values represent the best performances of the experiment.

Table 4. Accuracy measurement of Yelp review data with 4-fold cross-validation.

Algorithm	Maximum entropy	SVM	Bagging	Boosting	Random forest	Decision tree
1st fold	**0.9715447**	0.7642276	0.7142857	0.9043825	0.7279693	0.7083333
2nd fold	**0.9666667**	0.7842324	0.7725118	0.8880309	0.7791667	0.7251908
3rd fold	**0.9754098**	0.7795918	0.7166667	0.894958	0.7550201	0.6586345
4th fold	**0.9722222**	0.7388060	0.7137931	0.8650794	0.7640000	0.7469880
Mean accuracy	**0.9721644**	0.7667145	0.7293143	0.8881127	0.7565390	0.7097867

Bold values represent the best performances of the experiment.

Table 5. Accuracy measurement of IMDb review data with 4-fold cross-validation.

Algorithm	Maximum entropy	SVM	Bagging	Boosting	Random forest	Decision tree
1st fold	0.9141631	0.7711864	0.7346154	**0.9173554**	0.6931818	0.607438
2nd fold	**0.9010989**	0.8108108	0.6640316	0.8834586	0.7272727	0.5938697
3rd fold	**0.9139344**	0.7481203	0.6693548	0.9111111	0.7834646	0.6190476
4th fold	**0.912**	0.7615063	0.7656904	0.8988764	0.7947598	0.5959184
Mean accuracy	**0.9102991**	0.7729060	0.7084231	0.9027004	0.7496697	0.6040684

Bold values represent the best performances of the experiment.

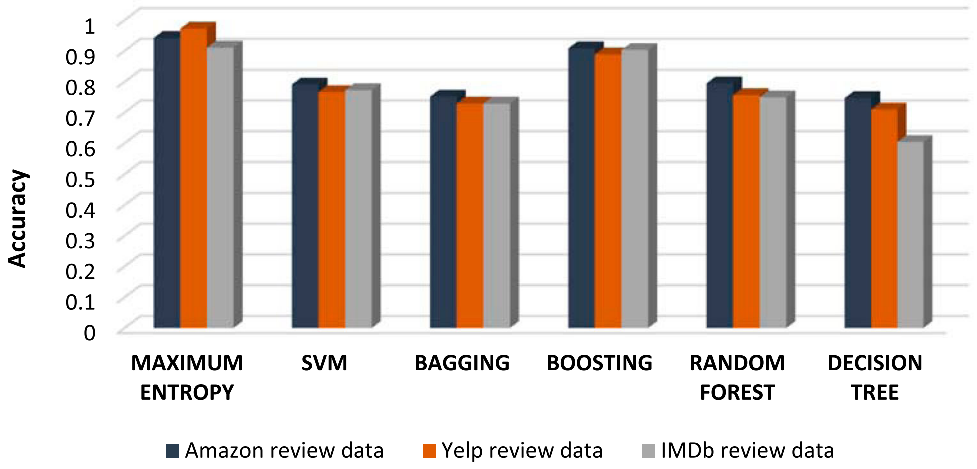

Figure 1. Accuracy analysis of algorithms on different review data sets based on cross-validation.

Table 6. Recall, precision, and *f*-score for different algorithms with review data sets.

	Amazon review data			Yelp review data			IMDb review data		
Algorithm	Recall	Precision	*F*-score	Recall	Precision	*F*-score	Recall	Precision	*F*-score
Maximum Entropy	0.785	0.800	0.785	**0.760**	**0.760**	**0.755**	0.735	0.730	0.730
SVM	0.850	0.855	0.850	0.705	0.700	0.700	**0.760**	**0.755**	**0.755**
Bagging	**0.870**	**0.870**	**0.865**	0.655	0.685	0.645	0.705	0.710	0.690
Boosting	0.840	0.840	0.840	0.680	0.690	0.670	0.690	0.730	0.690
Random Forest	0.860	0.860	0.860	0.670	0.675	0.660	0.730	0.735	0.715
Decision Tree	0.785	0.800	0.785	0.655	0.675	0.645	0.595	0.705	0.505

Bold values represent the best performances of the experiment.

of the algorithm is listed in column 1. Columns 2, 3 and 4 show the recall, precision, and *f*-score for product review collected from Amazon. Similarly, Columns 5, 6 and 7 show the results for restaurant review data from Yelp, and the last three columns present the generated results from IMDb movie review data. A graphical representation of this analysis is also shown in Figures 2–4.

5.3. Ensemble agreement analysis

A summary is represented in Table 7 with ensemble coverage and precision values for an ensemble greater than the threshold specified. Here seven-ensemble agreement approach is used to measure the agreement of different algorithms on review data sets.

5.4. Runtime analysis

Runtime for different machine learning algorithms- Maximum Entropy, SVM, Bagging, Boosting, Random Forest, and Decision Tree is measured with review data sets from Amazon, Yelp and IMDb (Table 8). Figure 5 shows the statistical comparison of runtime obtained by the analyzed algorithms.

For sentiment classification, several machine learning algorithms are applied to different data sets and various well-known performance evaluation metrics such as accuracy, recall, precision, *f*-score, and runtime are considered. From the experimental result of cross-validation, it is found that Maximum Entropy outperforms the other algorithms on each category of data sets. Ensemble machine learning methods such as Bagging, Boosting and Random Forest are

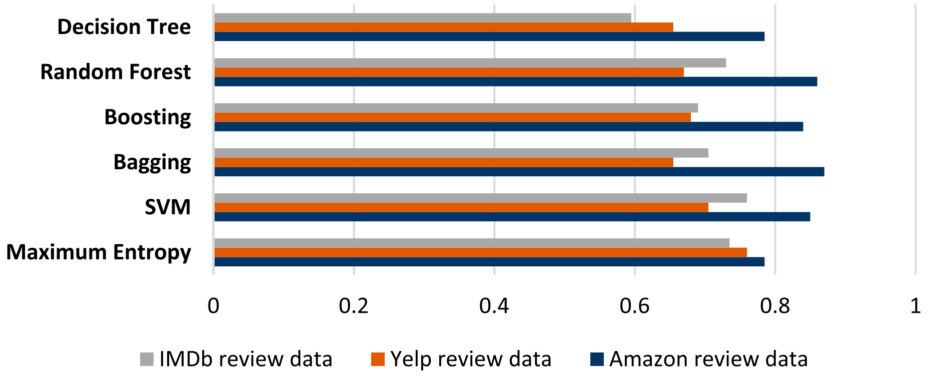

Figure 2. Performance analysis of algorithms on different review data sets based on recall.

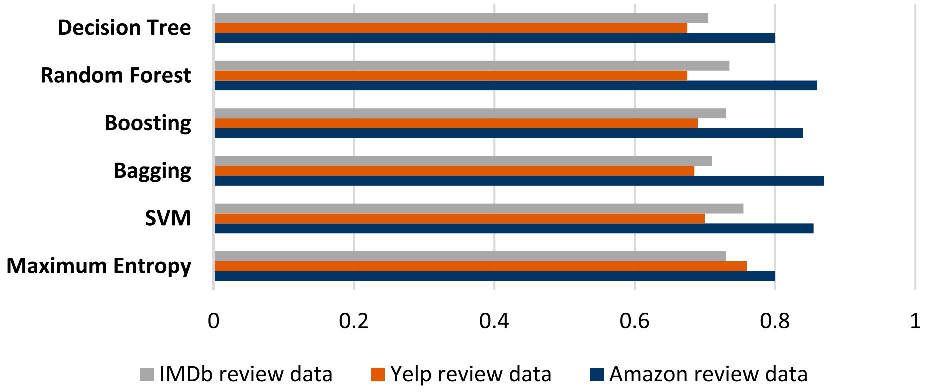

Figure 3. Performance analysis of algorithms on different review data sets based on precision.

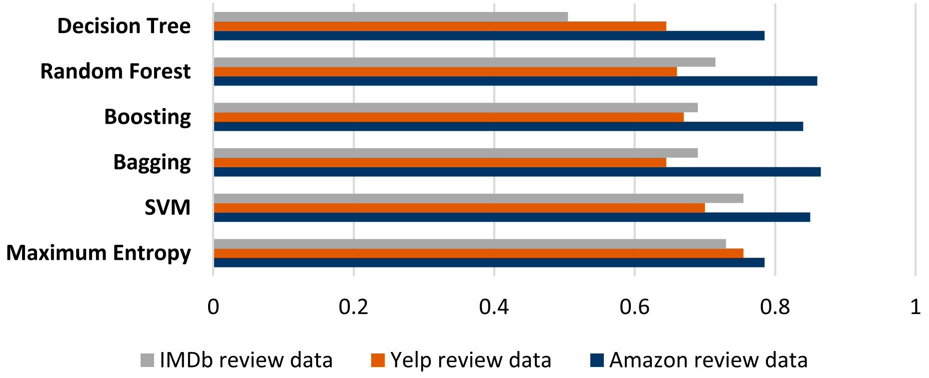

Figure 4. Performance analysis of algorithms on different review data sets based on *f*-score.

Table 7. Ensemble agreement on coverage and recall accuracy.

	Amazon review data		Yelp review data		IMDb review data	
n-ensemble	Coverage	Recall	Coverage	Recall	Coverage	Recall
$n \geq 1$	**1.00**	**0.86**	**1.00**	0.69	**1.00**	0.75
$n \geq 2$	**1.00**	**0.86**	**1.00**	0.69	**1.00**	0.75
$n \geq 3$	**1.00**	**0.86**	**1.00**	0.69	**1.00**	0.75
$n \geq 4$	**1.00**	**0.86**	**1.00**	0.69	**1.00**	0.75
$n \geq 5$	0.92	**0.89**	0.88	0.70	**0.94**	0.76
$n \geq 6$	**0.83**	**0.93**	0.78	0.74	0.74	0.80
$n \geq 7$	0.60	**0.93**	**0.62**	0.81	0.34	0.87

Bold values represent the best performances of the experiment.

Table 8. Expended runtime (seconds) by all the algorithms on different review data sets.

	Runtime (in second)		
Algorithm	Amazon review data	Yelp review data	IMDb review data
Maximum entropy	0.59373	0.7425411	0.8280959
SVM	1.046953	1.321182	1.393546
Bagging	25.52137	26.52654	48.35391
Boosting	6.282129	8.700261	12.68047
Random forest	**44.29559**	**117.49116**	**151.09842**
Decision tree	1.328087	1.763772	2.299856

Bold values represent the best performances of the experiment.

also implemented for different review data sets. These methods perform better compared with Decision Tree. Among the implemented machine learning classifier Boosting performs quite well after Maximum Entropy. From the experimental analysis, it can be concluded that Maximum Entropy and Boosting algorithm show nearly 90% accuracy for all data sets whereas Decision Tree performs

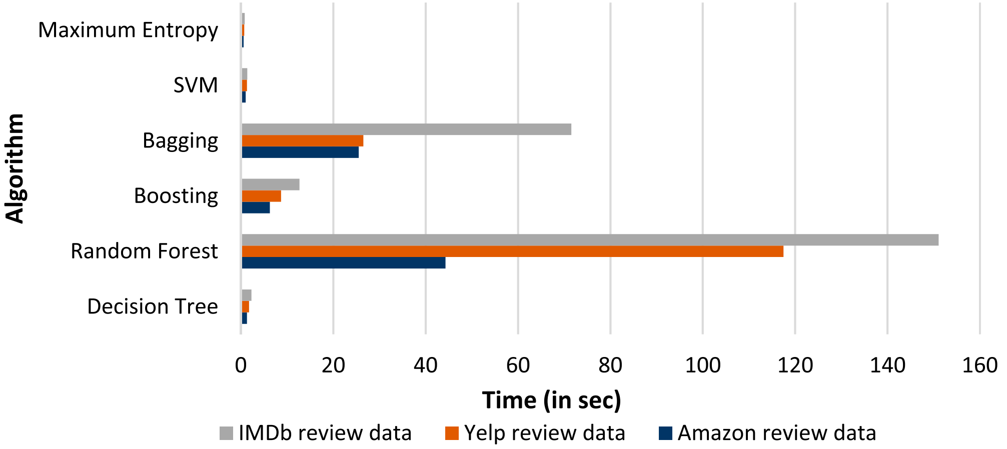

Figure 5. Runtime analysis of algorithms on different review data sets.

the least. Observing the result of recall, precision and f-score it is evident that Maximum Entropy, SVM and Bagging show significant outcomes. As a matter of runtime, Maximum Entropy needs the minimum amount of time and Random Forest is the largest one. Considering all these facts, it can be summarized that, among these implemented algorithms Maximum Entropy attained a better quality result because of its tolerance to irrelevant attribute and runtime. However, in this experiment, all the other machine learning classifiers produced nearly 80% accuracy for all the data sets. From this vast experimental analysis, it can be noted that machine learning approaches can be one of the good alternatives for sentiment classification.

6. Conclusions

In this paper, we present a vast experimental analysis of sentiment classification using different well-known machine learning approaches. Different classification techniques such as Maximum Entropy, SVM, Bagging, Boosting, Random Forest and Decision Tree are applied to conduct the experiments. For this experiments, three well-known review data sets collected from Amazon, Yelp and IMDb of different domains such as products, restaurants, and movies are used. To conduct the experiments, these review data sets are used to evaluate these models and classify sentiment polarity. For every different data sets accuracy, performance factors (recall, precision, f-score, and ensemble agreement) and runtime are considered to evaluate the result. Finally, the results are analyzed, compared and presented through a statistical approach.

For further research, the following directions could be followed to extend the work. Firstly, mining not only positive or negative polarity but also neutral opinion from review data. Secondly, making a classifier which will be topic and domain independent. Thirdly, combining the existing approaches to make a new model that can perform better with a large amount of data. At last, collaborating with various organizations to observe the influence on products with the analyzed results of review data and making advance decisions for the businesses.

Note

1. Available online at https://archive.ics.uci.edu/ml/datasets/Sentiment+Labelled+Sentences.

Disclosure statement

No potential conflict of interest was reported by the authors.

Notes on contributors

Monika Kabir is currently working as a lecturer of Computer Science and Engineering Department, Varendra University, Bangladesh. She received BSc of Computer Science and Engineering from Rajshahi University of Engineering and Technology (RUET), Bangladesh in 2016. She is also pursuing her MSc degree in the same university. Her research interests include Machine Learning, Natural Language Processing, Fuzzy Logic, and Artificial Intelligence.

Dr Mir Md Jahangir Kabir is currently a Professor of Computer Science and Engineering Department, Rajshahi University of Engineering and Technology, Bangladesh. He received BSc, MSc, and PhD degrees, from Rajshahi University of Engineering and Technology, Bangladesh, University of Stuttgart, Germany, and University of Tasmania, Australia in 2004, 2009, and 2016 respectively. After working as a Lecturer (from 2004) and Assistant Professor (from 2010), he was an Associate Professor (from 2017) in the Dept. of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Bangladesh. He joined as a Professor (from 2018) in the same department of that University. He received an Overseas Postgraduate Research Award from the Australian government in 2013 to research in PhD. His research interests include the theory and applications of Data Mining, Genetic Algorithm, Machine Learning, and Artificial Intelligence.

Dr Shuxiang Xu is currently a lecturer and PhD student supervisor within the Discipline of ICT, School of Technology, Environments and Design, University of Tasmania, Australia. He received a Bachelor of Applied Mathematics from University of Electronic Science and Technology of China (1986), China, a Master of Applied Mathematics from Sichuan Normal University (1989), China, and a PhD in Computing from University of Western Sydney (2000), Australia. He received an Overseas Postgraduate Research Award from the Australian government in 1996 to research his Computing PhD. His research interests are Artificial Intelligence, Machine Learning, and Data Mining. Much of his work is focused on developing new Machine Learning algorithms and using them to solve problems in various application fields.

Bodrunnessa Badhon is currently pursuing her Masters in Computer Science and Engineering from Rajshahi University of Engineering and Technology (RUET), Bangladesh. She completed her BSc in Computer Science and Engineering from the same university in 2016. She has interest in Data Mining, Machine Learning, Deep Learning, Genetic Algorithm, Fuzzy Logic and Artificial Intelligence.

ORCID

Monika Kabir http://orcid.org/0000-0001-5183-7036
Mir Md. Jahangir Kabir http://orcid.org/0000-0003-4963-8905
Shuxiang Xu http://orcid.org/0000-0003-0597-7040
Bodrunnessa Badhon http://orcid.org/0000-0002-5485-5552

References

[1] Kabir MMJ, Xu S, Kang BH, et al. A new multiple seeds based genetic algorithm for discovering a set of interesting Boolean association rules. Expert Syst Appl. 2017;74:55–69.

[2] Kabir MMJ, Xu S, Kang BH, et al. Discovery of interesting association rules using genetic algorithm with adaptive mutation discovery of interesting

association rules using genetic algorithm with adaptive mutation. Proceedings of the 22nd International Conference on Neural Information Processing (ICONIP), 2015, pp. 96–105.
[3] Kabir MMJ, Xu S, Kang BH. A new evolutionary algorithm for extracting a reduced set of interesting association rules. Proceedings of the 22nd International Conference on Neural Information Processing (ICONIP), 2015, pp. 133–142.
[4] Aggarwal CC, Zhai CX. Mining text data. 1st ed. New York: Springer-Verlag; 2012.
[5] Medhat W, Hassan A, Korashy H. Sentiment analysis algorithms and applications: a survey. Ain Shams Eng J. 2014;5(4):1093–1113.
[6] Sebastiani F. Machine learning in automated text categorization. ACM Comput Surv. 2002;34(1):1–47.
[7] Taboada M, Brooke J, Tofiloski M, et al. Lexicon-based methods for sentiment analysis. Comput Linguist. 2011;37(2):267–307.
[8] Prabowo R, Thelwall M. Sentiment analysis: a combined approach. J Informetr. 2009;3(2):143–157.
[9] Pang B, Lee L, Vaithyanathan S. Thumbs up?: sentiment classification using machine learning techniques. Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2002, July, Vol. 10, pp. 79–86.
[10] Zhang L, Ghosh R, Dekhil M, et al. Combining lexicon-based and learning-based methods for twitter sentiment analysis. Int J Electron Commun Soft Comput Sci Eng. 2015;89:1–8.
[11] Turney PD. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL), 2002, pp. 417–424.
[12] Dave K, Lawrence S, Pennock DM. Mining the peanut gallery. Proceedings of the 12th International Conference on World Wide Web (WWW), 2003, p. 519.
[13] Hu M, Liu B. Mining and summarizing customer reviews. Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2004, pp. 168–177.
[14] Lin C, He Y. Joint sentiment/topic model for sentiment analysis. Proceeding of the 18th ACM Conference on Information and Knowledge Management (CIKM), 2009, p. 375.
[15] Pan SJ, Ni X, Sun J-T, et al. Cross-domain sentiment classification via spectral feature alignment. Proceedings of the 19th International Conference on World Wide Web (WWW), 2010, p. 751.
[16] Serrano-Guerrero J, Olivas JA, Romero FP, et al. Sentiment analysis: a review and comparative analysis of web services. Inf Sci (Ny). 2015;311:18–38.
[17] Liu B, Street SM. Opinion Observer: analyzing and comparing opinions on the Web. Proceedings of the 14th International Conference on world Wide Web (WWW), 2005, pp. 342–351.
[18] Gupte A, Joshi S, Gadgul P, et al. Comparative study of classification algorithms used in sentiment analysis. Int J Comput Sci Inf Technol. 2014;5(5):6261–6264.
[19] Devika MD, Sunitha C, Ganesh A. Sentiment analysis: a comparative study on different approaches. Procedia Comput Sci. 2016;87:44–49.
[20] Moraes R, Valiati JF, Neto WPG. Document-level sentiment classification: an empirical comparison between SVM and ANN. Expert Syst Appl. 2013;40(2):621–633.
[21] Tripathy A, Anand A, Rath SK. Document-level sentiment classification using hybrid machine learning approach. Knowl Inf Syst. 2017;53(3): 805–831.
[22] Liu SM, Chen J. A multi-label classification based approach for sentiment classification. Expert Syst Appl. 2015;42(3):1083–1093.
[23] García-pablos A, Cuadros M, Rigau G. W2VLDA: Almost unsupervised system for aspect based sentiment analysis. Expert Syst Appl. 2018;91:127–137.
[24] Berger AL, Della Pietra VJ, Della Pietra SA. A maximum entropy approach to natural language processing. Comput Linguist. 1996;22(1):39–71.
[25] Ratnaparkhi A. Maximum entropy models for natural language processing. Encyclopedia of Machine Learning and Data Mining, 2010, pp. 647–651.
[26] Nigam K, John L, McCallum A. Using maximum entropy for text classification. IJCAI-99 Work Mach Learn Inf Filter. 1999;1(1):61–67.
[27] Vinodhini G, Chandrasekaran RM. Sentiment analysis and opinion mining: a survey. Int J Adv Res Comput Sci Softw Eng. 2012;2(6):1–11.
[28] Xia R, Zong C, Li S. Ensemble of feature sets and classification algorithms for sentiment classification. Inf Sci (Ny). 2011;181(6):1138–1152.
[29] Xu K, Liao SS, Li J, et al. Mining comparative opinions from customer reviews for competitive intelligence. Decis Support Syst. 2011;50(4): 743–754.
[30] Joachims T. Text categorization with Support vector Machines: learning with many relevant features. European Conference on Machine Learning (ECML), 1998, pp. 137–142.
[31] Breiman L. Bagging predictors. Mach Learn. 1996;24(2):123–140.
[32] Fersini E, Messina E, Pozzi FA. Sentiment analysis: Bayesian ensemble learning. Decis Support Syst. 2014;68:26–38.
[33] Whitehead M, Yaeger L. Sentiment mining using ensemble classification models. Innovations and Advances in Computer Sciences and Engineering, 2010, pp. 509–514.
[34] Araque O, Corcuera-Platas I, Sánchez-Rada JF, et al. Enhancing deep learning sentiment analysis with ensemble techniques in social applications. Expert Syst Appl. 2017;77:236–246.
[35] Kaur P, Gurm RK. Design and implementation of boosting classification algorithm for sentiment analysis on newspaper articles. Int J Comput Sci Inf Technol. 2016;7(4):1767–1770.
[36] Silva FF, Hruschka ER, Rafael E, et al. Biocom Usp: tweet sentiment analysis with adaptive boosting ensemble department of computer science. Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval), 2014, pp. 123–128.
[37] Schapire RE. The boosting approach to machine learning: an overview. Nonlinear Estimation and Classification, 2003, pp. 149–171.
[38] Breiman L. Random forests. Mach Learn 2001;45(1):5–32.
[39] Al Amrani Y, Lazaar M, El Kadiri KE. Random forest and support vector machine based hybrid approach to sentiment analysis. Procedia Comput Sci. 2018;127:511–520.
[40] Quinlan JR. Induction of decision trees. Mach Learn. 1986;1(1):81–106.
[41] Li YH, Jain AK. Classification of text documents. Comput J. 1998;41(8): 537–546.
[42] Lewis DD, Ringuette M. A comparison of two learning algorithms for text categorization. Proceeding of the 3rd Annual Symposium on Document Analysis and Information Retrieval (SDAIR), 1994, pp. 81–93.
[43] Wang S, Manning C. Baselines and bigrams: simple, good sentiment and topic classification. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, 2012, pp. 90–94.
[44] D'Andrea A, Ferri F, Grifoni P, et al. Approaches, tools and applications for sentiment analysis implementation. Int J Comput Appl. 2015;125(3): 26–33.
[45] Haddi E, Liu X, Shi Y. The role of text pre-processing in sentiment analysis. Procedia Comput Sci. 2013;17:26–32.
[46] Patil MG, Gale MV, Kekan MV, et al. Sentiment analysis using support vector machine. Int J Innov Res Comput Commun Eng. 1970;2(1):2607–2612.
[47] Zaki MJ, Meira Jr. MJ. Fundamental concepts and algorithms. 1st ed. New York: Cambridge University Press; 2014.