Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

Empirical comparison of sentiment analysis techniques for social media

Maria Hameed [1], Faizan Tahir [2,] [*], M. Ali Shahzad [1]

[1]*Department of Computer Science, University of Sargodha, Lahore, Pakistan*
[2]*Department of Computer Science, Virtual University of Pakistan, Faisalabad, Pakistan*

ARTICLE INFO

Article history:
Received 29 November 2017
Received in revised form
15 February 2018
Accepted 27 February 2018

Keywords:
Sentiment analysis
Social media
UCI database
KEEL support vector machine

ABSTRACT

Nowadays the excessive use of internet produces a huge amount of data due to the social networks such as Twitter, Facebook, Orkut and Tumbler. These are microblogging sites and are used to share the people opinions and suggestions on daily basis relevant to the certain topic. These are beneficial for decision making or extracting conclusions. Analysis of these feeds aims to assess the thinking and comments of people about some personality or topic. Sentiment analysis is a type of text classification and is performed by various techniques such as Machine Learning Techniques and shows that the text is negative, positive or neutral. In this work, we provide a comparison of most recent sentiment analysis techniques such as Naïve Bayes, Bagging, Random Forest, Decision Tree, Support Vector Machine and Maximum entropy. The purpose of the study is to provide an empirical analysis of existing classification techniques for social media for analyzing the good performance and better information retrieval. A comprehensive comparative framework is designed to compare these techniques. Various benchmark datasets (UCI, KEEL) available in different repositories are used for comparison purpose. We presented an empirical analysis of six classifiers. The analysis results that Support Vector Machine performs much better as compared to other. Efforts are made to provide a conclusion about different algorithms on the basis of numerical and graphical metrics to conclude that which algorithm is optimal.

1. Introduction

Sentiment analysis is used to collect and examine opinions about the product made in blogs, posts, reviews and tweets. These sentiments indicate his opinion about the particular topic or product. The main focus of sentiment analysis is evaluation of attitudes and opinions on a topic of interest using machine learning techniques.

There are three levels of sentiment analysis. The document level sentiment analysis classifies the complete document as negative or positive (Devika et al., 2016). The sentence level sentiment analysis analyzes and classifies each sentence that either it is positive, negative or neutral (Zhang et al., 2011). The aspect level sentiment analysis evaluates an opinion (Devika et al., 2016).

Microblogging sites are nowadays used for communicating with others on the social networks. These sites are a valuable source of opinion. The sites include Twitter, Tumbler, Orkut, Amazon, Facebook and Google Plus. The aim of the user to search social media is for finding breaking news, reviews about celebrities and hot issues of politics.

Microblogging sites are a useful way for sharing short message, links images and videos.

Each user updates his/her status personally. Blogs that are relevant to company profiles or political parties are updated by teams of community manager.

These sites are normally updated periodically after every hour to provide updates about their area of interest. These sites deal with multiple topics.

Celebrities and artist profiles are concerned with popularity on these sites. The first publication on the Twitter was held on 16th July 2006 and then it became the most popular site in the sentiment analysis community.

Twitter is the eighth most popular website in the world and eighth in the United States based on Alexa (http://www.alexa.com), with an average of nearly eleven million tweets per day. These particular messages have limited size up to 148 characters called tweets and peoples convey their information in a complex way. They use short symbol, lengthy words, hashtags, abbreviations, incorrect grammar and idioms etc.

[*] Corresponding Author.
Email Address: fazitahir@vu.edu.pk (F. Tahir)

https://doi.org/10.21833/ijaas.2018.04.015

According to a survey in 2010 (Kwak et al., 2010) there are 41.7 million user profiles, 4262 trending topics and 106 billion tweets.

There are various techniques used for the classification of sentiment analysis of tweets such as Machine Learning approach, Lexicon based approach and rule-based approach. Nowadays Machine Learning methods are widely used and perform well for the analysis. Machine learning algorithms are used to assess the polarity of the data.

Machine learning is divided into supervised and unsupervised learning. Supervised classification algorithms are probabilistic classifier such Naïve Bayes, linear classifier such as logistic regression and perceptron, decision tree and rule based classifier (Neelamegam and Ramaraj, 2013). Supervised learning technique is based on a labeled dataset to train the model and then this model is applied to test data to assess the output and validate the performance of the classifier.

Supervised algorithms require a set of training data analyzes it and apply on the test data that is learned. There are various algorithms used for the supervised machine learning approaches such as Naïve Bayes (Barbosa and Feng, 2010), Maximum Entropy (Kotsiantis, 2007), Decision Tree (Kumar and Verma, 2012) and Support Vector Machine (Bhavsar and Ganatra, 2012). The performance of these algorithms is checked on the basis of some common factors such as Precision, Recall, Accuracy, and F-measure for each classifier (Belavagi and Muniyal, 2016; Drummond and Holte, 2000). Our work focus on analysis of these machine learning algorithms with ten numerical and three graphical curves and shows which are better and optimal classifier. Previous work done is focused only on accuracy or few parameters because of which algorithms are not evaluated properly. Most of the analyst only uses statistical metrics for evaluation, only a few people used graphical representation for performance metrics.

When the mentioned metrics, analyze the algorithms, more accurate results will be obtained and analysis will be more formal.

The goal is to select the best technique amongst these techniques. It will provide important guidelines in the machine learning community.

This work provides a right direction for the analyst to select the algorithm whose performance is better than others.

1.1. Naïve Bayes

Naïve Bayes classifier is based on Bayes theorem. This algorithm assumes the independent features of dataset.

According to this algorithm a feature in a class is independent of presence and absence of any other feature in the same class. Naive Bayes are successfully applied in applications related to text classification, system performance management and medical diagnosis. This algorithm requires a small amount of data. It considers the words of the document as a bag of words. The words are in the form of a direct acyclic graph in which nodes are variables and arcs represent the dependency between them.

1.2. Maximum entropy

Maximum Entropy classifiers are normally used for Natural Language Processing and information retrieval (Bhavasar and Ganatra, 2012). It was introduced by (Jaynes, 1957). Maximum entropy is approximately same as Naïve Bayes but it does not assume that each feature of the class is independent of others. The principle behind this algorithm is to maximize the entropy or estimate the weights of the class labels. Another name of maximum Entropy is Logistics in WEKA.

1.3. Support vector machine (SVM)

Support Vector Machine is a binary linear and is widely used for classification and regression problems. It was introduced by Vapnik (1995). It does not require any previous knowledge or past experience for evaluation.

SVMs work on hyper planes and produce the optimal separation of various classes. The purpose of SVM is to find a maximum margin hyper plane and separates one class from another. These maximum margin hyper planes are represented by vectors. SVM are used to train the model for n-grams.

In this algorithm, each data item is considered as a point in n dimensional space and the value of each feature becomes a particular coordinate.

SVM requires a large amount of training dataset. The objects belong to any class and the separating lines define the boundary. A mathematical function kernel is used to map or transform the objects from low dimensional input space to a higher dimensional space.

It clearly presents the margin of separation and performs well in high dimensional spaces. This technique is memory efficient. SVM includes an optimized approach i.e., Sequential Minimal Optimization approach (SMO) (Holmes et al., 1994) in WEKA tool.

SVM based classification is accurate algorithm but has some limitations such as it is computationally expensive and time consuming.

The aim of machine learning analysis is to find the usefulness of these learning algorithms on different collection of datasets.

1.4. Decision tree

Decision Tree was first introduced by Quinlan (1993) and is called J48 in WEKA classifies the data in the form of a tree. It is a supervised technique and follows divide and conquer rule. Every node represents an information set. Every branch indicates the results of the test and leaves of the tree represent the class labels. It generates in a top to down manner with a parent class of the root node. There is a concept of over fitting in trees which means that this algorithm performs well on training data but not on test data. The solution of over fitting is tree pruning.

To avoid over fitting, tree pruning is used. In pre-pruning, don't let the tree be large which is not possible to know in advance. In post pruning the tree is developed, its performance is checked until the required tree is obtained.

The information gain is maximum at the root node and decreases as we go down the tree. Decision tree can have used to classify discrete data.

The most famous approach of the decision tree is known as J48. It first chooses an attribute and then best differentiates the output attribute values.

1.5. Random forest

Breiman (1994) introduced the random forest algorithm and normally used for classification and regression problems. It works on a collection of trees called a forest. Maximize the number of trees in the forest the better will be the results. Random forest does not over fit. Due to this reason, it requires a lot of memory. It takes randomly k input vectors out of total m vectors classifies them with every tree in the forest and provides the class label as an output that have maximum votes. For each training set same numbers of vectors are selected as in the original set by using the bootstrap method. The vectors are chosen randomly with replacement and with each node a new subset is created. These sets are generated from the original training set using the bootstrap method. The size for all nodes and trees are fixed. It works well on large data sets with lots of input variables.

1.6. Bagging

Bagging means Bootstrap aggregation. It was introduced by Breiman (2001) for improving classification accuracy. It is a process of selecting samples from the original sample and using these samples for estimating various statistics or model accuracy. In the process of Bootstrapping random samples are created with a replacement for estimating sample statistics. This algorithm selects n items with replacement from an original sample N. A bootstrap sample may have a few duplicate observations or records as the sampling is done with replacement. Bootstrap samples are created to estimate and validate models for improved accuracy, reduced variance and bias, and improved stability of a model. Once bootstrap samples are created, model classifier is used for training or building a model and then selecting a model based on popularity votes. In classification model, a label with maximum votes will assign to the observations.

In this research, six different machine learning algorithms are applied on four different data sets from UCI repository and evaluated in terms of cross-validation performance and classification accuracy.

These algorithms are successful when the set of features used is properly selected to detect sentiments. There are various data driven algorithms used for the classification of data on different features. The best model or algorithm depends on the characteristics of the dataset and also for the cross validation technique used and also for the quantitative analysis of these models.

2. Literature review

Devika et al. (2016) compared the various techniques used for Sentiment Analysis by analyzing various methodologies. He discussed three types of approaches: Machine learning approach, lexicon based approach and Rule based approach. This paper compares the various techniques used for Sentiment Analysis by analyzing various methodologies. He also discussed pros and cons of these approaches by considering the key factors like performance, efficiency and accuracy.

Belavagi and Muniyal (2016) discussed the intrusion detection system to predict the network data traffic is normal or an intrusion. In this paper classification and predictive models are built by using Machine Learning classification algorithms such as Logistic Regression, Gaussian Naïve Bayes, Support vector Machine and Random Forest.

Experimental results show that Random Forest out performs than other methods in identifying whether the data traffic is normal or an attack.

Kumar and Verma (2012) and Vaghela and Jadav, 2016) gave an idea that if features are selected carefully, then the classification algorithms give better results especially accuracy. He also provides a difference between lexicon and machine learning algorithms that machine learning algorithms are dependent on the domain. they used supervised machine learning algorithms and describes that machine learning algorithms requires prior training, adaptive learning, results generate slow, do not require maintenance and has high accuracy as compared to lexicon based models which are its vice versa. He also used the word net data dictionary for scoring and analyzed. The results concluded that using the word net with classification algorithms provides better accuracy.

Kharde and Sonawane (2016) provided a survey and comparative analysis of supervised and unsupervised approaches with some evaluation metrics. They use the classification algorithms like Naïve Bayes, Maximum Entropy and Support Vector Machine and also discuss the challenges and applications of sentiment analysis.

Kalarikkal and Remya (2015) focused on the data set for sentiment analysis and used the machine learning techniques such as Maximum Entropy, Naïve Bayes and SVM method. They use good quality training set for better performance and results.

Das et al. (2014) developed an application that collected data from twitter, analyzed it with and generate reports containing tables and pie chart graphs.

Shrivatava et al. (2014) introduced an efficient method to classify the features of tweets and uses support vector machine to classify the tweets and attain an accuracy of 70.5 %.

Chavan et al. (2014) examined and compared the effectiveness of applying machine learning techniques to sentiment classification problem. He performed text categorization by using classifier algorithms such as Decision Trees, Support Vector Machine and Naïve Bayes. The author uses relevant results and examples and prove that SVM provides better accuracy than other two and can find and adjust automatically to parameter settings.

Medhat et al. (2014) presented the various applications and algorithms. Feature Selection in Sentiment Classification. The survey also presents which algorithms are used in research papers and in which years they used. How to select the features and also sentiment classification techniques?

Neelamegam and Ramaraj (2013) provided a review of various classification techniques in data mining. He discussed several major kinds of classification techniques that is decision tree, Bayesian networks, k-nearest neighbor classifier, Neural Network, Support vector machine are discussed in this paper. He concludes that good data and appropriate technique produced better results to mine the data.

Moraes et al. (2013) classified the textual reviews that are expressing positive and negative sentiments. An empirical comparison of SVM and Artificial Neural Networks is performed regarding the document level sentiment analysis. ANN produces better results than SVM. The computational cost of these algorithms is also discussed.

Padmapriya (2012) used well known classification algorithms such as Naïve Bayes and Decision tree to analyze the higher education admissibility. The performance of these algorithms is assessed and compared to find the optimal algorithm.

Kang et al. (2012) introduced a new improved Naïve Bayes Algorithm. This algorithm will give better results when used with unigram and bigram features. This improved algorithm was also proved by experiments and comparison with the SVM and Naïve Bayes and provides better results than simple Naïve Bayes and SVM.

Lane et al. (2012) discussed the challenges facing by a machine learning approaches. One of them is class imbalance in positive and negative samples. The results are based on experiments by using some features and conclude the optimal classifier, feature set and training approach depends on the data set.

Genc et al. (2011) used Decision tree classifier to categorize tweets and identify about breaking news.

Wahbeh et al. (2011) presented a comparison about the best tool between Orange, Knime, Tanagra and sidewalk. WEKA tool has the highest applicability than Orange, Tanagra and KNIME. The paper concludes that the performance of the tools used for classification are affected by the kind of dataset used and the way in which the classification algorithms were implemented in the toolkits. Two modes split test mode and cross validation test mode are used for evaluation and the results show that the performance of the tools depends on the kind of the dataset.

Saleh et al. (2011) used different domains dataset and applied SVM on it to accomplish the sentiment analysis by using several weighing schemes.

Pak and Paroubek (2010) used a method for automatically collecting twitter data and perform a sentiment classifier. Naive Byes classifier is used in this paper and has N-gram and Parts of Speech (POS) features.

Barbosa and Feng (2010) used SVM and analyzed tweets for determining the polarity of words.

Batra and Rao (2010) explored a dataset of tweets and find the probability of a unigram that either it is positive, negative or neutral.

Parikh and Movassate (2009) implemented two Models Naïve Bayes Bigram and Maximum Entropy to analyze and classify tweets and concluded that the Naïve Bayes model gives improved results than the Maximum Entropy Model.

Abbasi et al. (2008) proposed the sentiment analysis for multiple languages. Various features are used to classify hate and extremist posts on tweets relevant to politics or some other issues. He combines the Maximum Entropy and Genetic Algorithm to form Entropy Weighted Genetic Algorithm (EWGA) is used for this purpose and obtained an accuracy of 95.55%. The technique uses in his paper is SVM and gives a better accuracy.

The machine learning approach Naïve Bayes was used by Ye et al. (2009), Smeureanu and Bucur (2012), Xia et al. (2011), and Melville et al. (2009) works for text mining.

It is the most famous technique for classification of text. Naïve Bayes uses a small training set. This approach is a simple and effective approach of the Natural Language Processing and works on probability.

Support Vector Machine was used by Zhang (2011) to determine the closest points and calculating the hyper-plane to separate labels. SVM used a large amount of training set (Kumar and Verma, 2012). For Text classification, Multiclass SVM can also be used (Moraes et al., 2013).

Kotalwar et al. (2014) predicted that employee performance is predicted on the basis of data mining techniques that help in decision making. The performance of employees is calculated by using the employee database.

Huang et al. (2003) compared different machine learning algorithms such as SVM, Naïve Bayes and decision trees on the basis of accuracy and AUC (area under curve) and proves that area under the curve is a better measure than accuracy.

Pang et al. (2002) performed a comparison by using Naive Bayes, Maximum Entropy and Support Vector Machine by using different features like unigram and bigram, the combination of unigram and bigram, parts of speech and information about position using adjectives and achieve the accuracy of 82.9%.

SVM gives better results when feature space is increased. They consider various problems faced by the sentiment classification tasks. Pang et al. (2002) found that term presence is more important to sentiment analysis than term frequency.

3. Proposed solution

The work presented here contains datasets on various topics like politics (Donald Trump's Tweets) and some datasets from UCI repository contains the data about social media. All the datasets contain a number of instances stored within datasets, the number of attributes and the types of the attributes (integer, categorical, Real). One of the dataset is multiclass and the other three have binary classes. I preprocess these datasets manually. I convert upper case to lower case letter, remove punctuation, remove special character and remove uniform resource locator.

The labels are assigned to the text, 0 for negative and 1 for positive in binary and 0, 1, 2 for multi class. These files are saved as .CSV (Comma Separated Value) file then I will convert these files into. ARFF (Attribute Relation File) format.

I split the data into training and test data sets and save the files individually. The files are then used for further processing and various supervised machine learning classifiers such as Naïve Bayes, Maximum Entropy, Decision Tree, Support Vector Machine, Random Forest, and Bagging are analyzed on the basis of some performance metrics. According to these metrics we properly assess which algorithm performs better than others. The tool used for evaluation is Weka which is the most famous tool nowadays.

The implementation will be done in Java in Net Beans and evaluate the machine learning algorithms on these datasets by using K-fold cross validation mode. The performance measures are based on two class confusion matrix. The evaluation of these algorithms is based on ten performance metrics:

- Precision
- Recall
- Specificity
- Sensitivity
- F-measure
- J-Coefficient
- G-means
- Kappa
- Error rate
- Accuracy

Some Graphical metrics such as

- Receiver Operating Characteristic Curve
- Cost\Benefit Analysis Curve
- Cost Curve

Then it will be concluded that which algorithm is more efficient for the above datasets with respect to these parameters.

This research is related to the study of existing algorithms and their comparison in terms of performance, accuracy and efficiency of the algorithms so that the strength of the algorithms can be evaluated. As the complexity of algorithms differs from each other so this work presents the visualization of performance metrics. We provide empirical experiment results for evaluation in order to prove the optimality of the algorithm. We are acknowledged about algorithm's performance in various datasets.

Literature studied shows that previous work done is focused only on accuracy of a few parameters because of which algorithms are not evaluated properly. Most of the analyst only uses statistical metrics for evaluation, only a few people used graphical representation for performance metrics.

When the mentioned metrics, analyze the algorithms, more accurate results will be obtained and analysis will be more formal. The goal is to select the best technique amongst these techniques. It will provide important guidelines in the machine learning community. This work provides a right direction for the analyst to select the algorithm whose performance is better than others.

The methodology which I adopted is based on the optimality of algorithm and is as follows:

I have taken a dataset of amazon reviews that indicates the polarity of positive and negative reviews about the cell phones and its accessories.

It has two attributes: first is amazon reviews and the second attribute is its polarity. The dataset indicates 0 for negative and 1 for positive reviews. This dataset is found in sentiment labeled sentences dataset in UCI repository. After collecting the data next process is to preprocess which means removing all the special characters such as and, comma, capital letters, etc. Preprocessing is performed manually and split this dataset into training and test files. Save these files in. ARFF format. Write a Java Code in Net Beans that provides filters to the dataset, Evaluation and search attributes and perform cross validation by using seed 1 to apply supervised algorithms on the dataset. Compare the results for conclusion.

In Preprocess, all the data which are in textual form is converted to numerical form. The dataset applies the string to word vector unsupervised filter that converts string attributes in a set of attributes that shows word occurrence information from the text contained in a string. I use its default settings for stemmer and stop words.

After filtering the dataset, the Gain Ratio Attribute measures the value of an attribute by considering the Gain ratio according to positive and negative class or label. Rankers search attribute is applied with Gain Ratio.

After applying the evaluation and search attributes next is to classify the test dataset on the basis of the training data set. These algorithms predict the labels of the Test dataset.

The detail of these metrics is based on binary confusion matrix shown in Table 1:

- Binary confusion matrix is a 2x2 confusion matrix that contains instances
- Number of True Positives (T_P)
- Number of False Positives (F_P)
- Number of True Negatives (T_N)
- Number of False Negatives (F_N)

Table 1: Confusion matrix

	Machine Says Yes	Machine Says No
Human Says Yes	T_P	F_N
Human Says No	F_P	T_N

Most classifiers performance metrics are based on these values. A technique used for evaluation is k fold Cross Validation. K folds cross validation where k is the number of folds or groups. Under this procedure, data are randomly divided into k equal size folds where first fold is treated as validation set and remaining n-1 as training set. The procedure is repeated k times for each one of the folds. K fold cross validation is simply the average of the RMSE (Root Mean Squared Error) divided by k. folds. RMSE is the difference between predicted values and observed values. In practice k is 5 or 10 folds.

The goal of cross validation is to overcome the problem of over fitting, making the predictions more general and improving the holdout method by reducing the variance among data. Cross-validation includes splitting of the data set into k parts, one is to test the data and remaining k-1 are used for training, process repeats k times till all the parts are taken as a test set. To analyze and visualize the curves we choose four data sets from the UCI repository (Frank, 2010). Cars, Amazon and IMDB Dataset and one dataset are of Trump Tweets. Cars are a multi-class dataset and remaining are binary class data sets. Six different machine learning algorithms are applied to these data sets in the WEKA workbench (Kalarikkal and Remya, 2015) i.e., Decision Tree, Naïve Bayes, Random Forests, Support Vector Machine, Bagging and Maximum Entropy. We will start by applying ten-fold cross validation on the dataset to obtain the quantitative results shown in Table 2 and Figs. 1-2.

Then by evaluating the datasets on WEKA, the results are collected for each data set indicating the visual representation of these datasets. Three types of curves are shown in results. Threshold Curve, Cost Curve and Cost/Benefit Analysis. The results concluded that the performance of SVM is better on these datasets as compare to other Algorithms.

Table 2: Accuracy and error rate of Datasets

Techniques	Amazon		Donald Trump		IMDB		Cars	
	Acc %	Er %	Acc %	Er %	Acc %	Er %	Acc %	Er %
SVM	54.74	45.45	59.25	43.33	51.38	45.00	79.05	25
NB	55.99	43.93	63.75	38.88	63.19	35.00	67.94	35
ME	59.21	40.90	65.25	37.77	71.04	35.00	57.83	42.4
DT	55.25	43.93	58.24	43.33	60.06	39.76	54.13	50
RF	58.47	40.90	56.25	45.55	58.33	36.66	67.66	37.5
Bg	53.27	45.45	54.25	46.66	52.08	41.66	57.69	50

The performance evaluation is on the basis of the area under the curve. The area where Probability of False alarm is low and Probability of Detection is high has good impact on the performance of the classifier. The color that is nearest to blue in the threshold curve indicates the lower threshold value.

An ROC curve is best suited for the evaluation of binary class problems. An ROC curve indicates the comparison between sensitivity and specificity. Area under the curve is used for measuring the quality of the classifier. A best classifier is the one whose AUC value is equal to 1. Normally area under the curve for various classification algorithms is between 0.5 and 1.ROC curve shows sensitivity on X-axis and specificity on Y-axis. The probability is 1 when the false positive rate is 0 and true positive rate is 1. Figs. 3-4 indicate the ROC curves. Ideally the curve will proceed towards the top left, which means that the model correctly predicted the result. The area which has low sensitivity and high specificity has some impact on the classifier performance. The Bar graph indicates these datasets along with their accuracy and error rate. In Cost/Benefit analysis, there are several panels. First is the threshold curve frame contains the threshold curve also called Lift curve corresponds to the part of sample size.

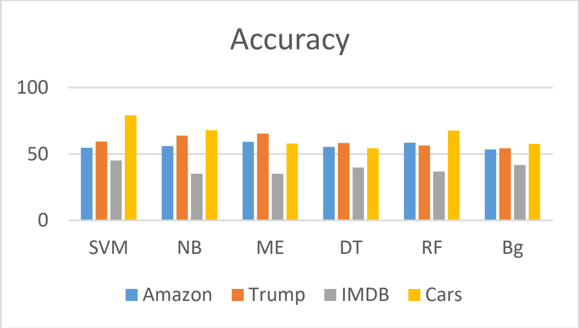


Fig. 1: Accuracy of various algorithms on different datasets

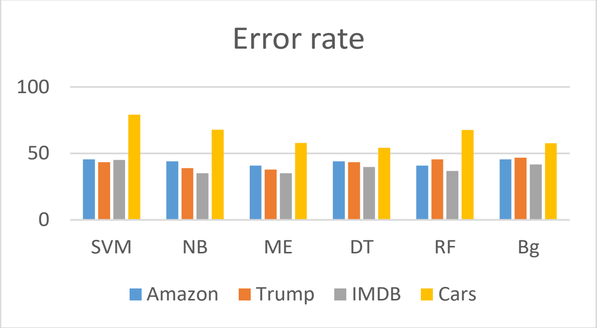
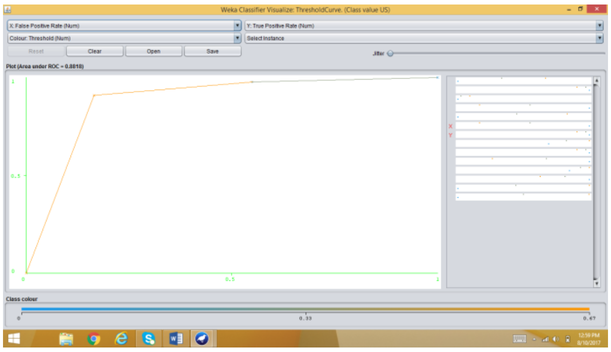


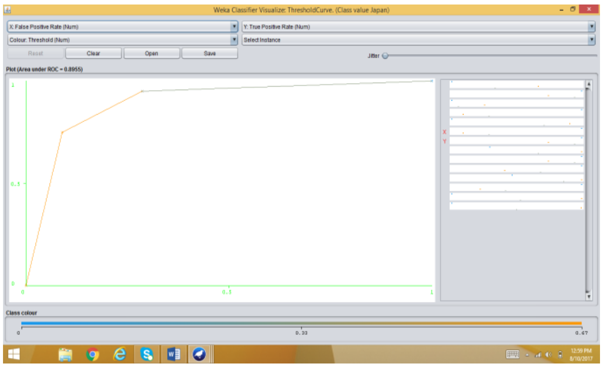
Fig. 2: Error rate of various algorithms on different data sets


Weka Classifier Visualize: ThresholdCurve. (Class value US)

X: False Positive Rate (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance

Plot (Area under ROC = 0.8818)

Fig. 3: Threshold curve (Class 0)


Weka Classifier Visualize: ThresholdCurve. (Class value Japan)

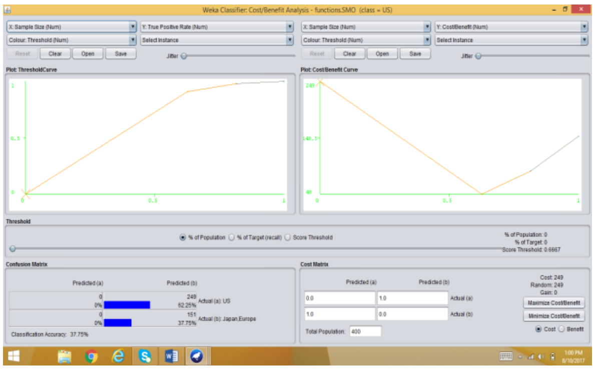
X: False Positive Rate (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance

Plot (Area under ROC = 0.8955)

Fig. 4: Threshold curve (Class 1)

Threshold curve is similar to ROC curve. Y-axis represents the true positive rate and X-axis indicates the sample size. The right bottom corner shows the cost matrix frame. Its entries depict the cost a person paid on the basis of classification. The cost one should pay for decisions taken on the base of the classification mode. Its default value is 1. Figs. 5-7 show the cost/benefit analysis curve for US, Japan and Europe classes.

Drummond and Holte (2006) proposed the cost curve that describes the performance of classifier depends on the cost of misclassification. Cost curves are used for estimating the expected cost.


Weka Classifier: Cost/Benefit Analysis - functions.SMO (class = US)

X: Sample Size (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance
X: Sample Size (Num)
Y: Cost/Benefit (Num)
Colour: Threshold (Num)
Select Instance

Plot: ThresholdCurve
Plot: Cost/Benefit Curve

Threshold
● % of Population ○ % of Target (recall) ○ Score Threshold
% of Population: 0
% of Target: 0
Score Threshold: 0.6667

Confusion Matrix

	Predicted (a)	Predicted (b)	
	0 0%	249 62.25%	Actual (a): US
	0 0%	151 37.75%	Actual (b): Japan,Europe

Classification Accuracy: 37.75%

Cost Matrix

Predicted (a)	Predicted (b)	
0.0	1.0	Actual (a)
1.0	0.0	Actual (b)

Total Population: 400

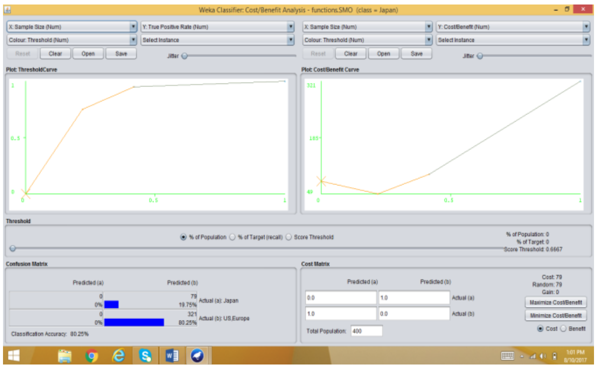
Cost: 249
Random: 249
Gain: 0

● Cost ○ Benefit

Fig. 5: Cost/benefit analysis (class 0)

Cost curves are produced by (0, specificity) and (1, 1-sensitivity). Cost curve minimizes the lower envelope area. Classifier's performance is better when smaller the area in lower envelope and hence better cost benefit ratio. Cost curves analyze the performance on the basis of operating points (Drummond and Holte, 2000). Operating points depend on probability of class and misclassification costs. The normalized expected cost is similar to the error rate and represents the classification performance on the x-axis. Lower the value of normalized expected cost better will be the classifier.

The lowest error rate is of Maximum Entropy and highest accuracy is on Support Vector Machine. SVM gives lower envelope area and so better cost curve. Figs 8-10 show the cost curves. The lowest error rate and highest accuracy is of Support Vector Machine.


Weka Classifier: Cost/Benefit Analysis - functions.SMO (class = Japan)

X: Sample Size (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance
X: Sample Size (Num)
Y: Cost/Benefit (Num)
Colour: Threshold (Num)
Select Instance

Plot: ThresholdCurve
Plot: Cost/Benefit Curve

Threshold
● % of Population ○ % of Target (recall) ○ Score Threshold
% of Population: 0
% of Target: 0
Score Threshold: 0.6667

Confusion Matrix

	Predicted (a)	Predicted (b)	
	0 0%	79 19.75%	Actual (a): Japan
	0 0%	321 80.25%	Actual (b): US,Europe

Classification Accuracy: 80.25%

Cost Matrix

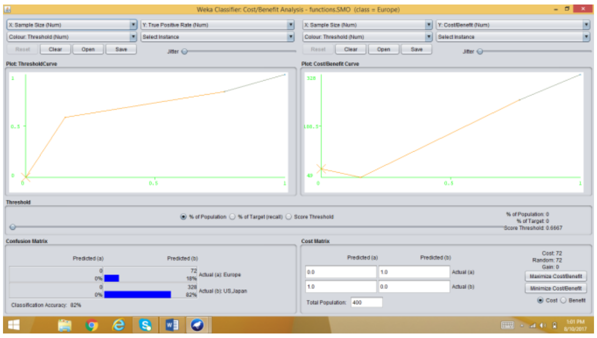
Predicted (a)	Predicted (b)	
0.0	1.0	Actual (a)
1.0	0.0	Actual (b)

Total Population: 400

Cost: 79
Random: 79
Gain: 0

● Cost ○ Benefit

Fig. 6: Cost/benefit analysis (class 1)


Weka Classifier: Cost/Benefit Analysis - functions.SMO (class = Europe)

X: Sample Size (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance
X: Sample Size (Num)
Y: Cost/Benefit (Num)
Colour: Threshold (Num)
Select Instance

Plot: ThresholdCurve
Plot: Cost/Benefit Curve

Threshold
● % of Population ○ % of Target (recall) ○ Score Threshold
% of Population: 0
% of Target: 0
Score Threshold: 0.6667

Confusion Matrix

	Predicted (a)	Predicted (b)	
	0 0%	72 18%	Actual (a): Europe
	0 0%	328 82%	Actual (b): US,Japan

Classification Accuracy: 82%

Cost Matrix

Predicted (a)	Predicted (b)	
0.0	1.0	Actual (a)
1.0	0.0	Actual (b)

Total Population: 400

Cost: 72
Random: 72
Gain: 0

● Cost ○ Benefit

Fig. 7: Cost/benefit analysis (class 2)

4. Conclusion

The summary states a model has a good generalization performance if it maximizes the accuracy and minimize the error rate. The accuracy of SVM is better than the Naïve Bayes, Decision Tree, Random Forest, Maximum Entropy and Bagging. The size and nature of the dataset affects the accuracy of the algorithm.

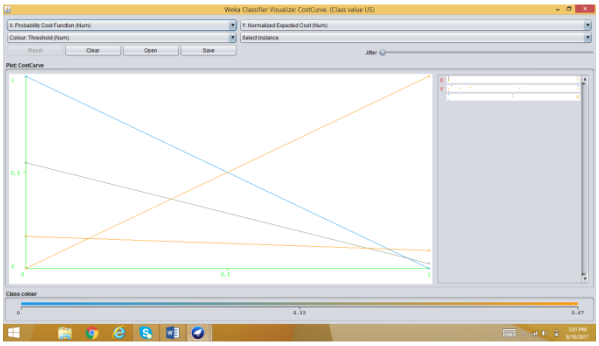
Maximum Entropy and Naïve Bayes have a better learning speed according to attributes and instances than decision tree, random forest, support vector machine and bagging. Approximately same speed of classification But IMDB dataset takes the maximum time because movie reviews take more time to assess. The algorithms decision tree and maximum entropy take more time for evaluation than others.

Naïve Bayes and Maximum Entropy work well when the size of dataset is small and SVM works well in the multiclass environment so it is working well for Car data set.

Decision Tree also works well on large datasets. The classification performance of the classifiers based on different training set is different.

If the data sets have high quality, the performance of the classifier will automatically become good. The

Best model or algorithm depends on the characteristics of the dataset, on cross validation technique used and also on the quantitative analysis of these models.



Weka Classifier Visualize: CostCurve. (Class value US)

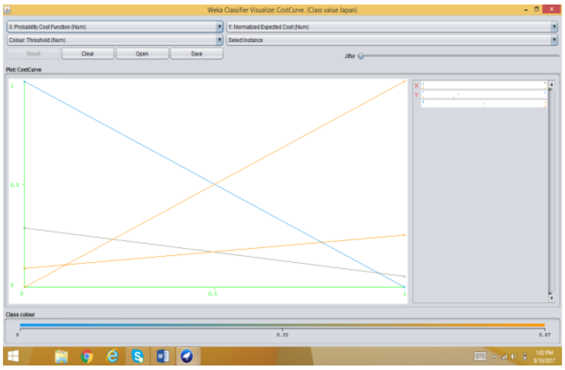
X: Probability Cost Function (Num)

Y: Normalized Expected Cost (Num)

Colour: Threshold (Num)

Select Instance

Fig. 8: Cost curve (class 0)



Weka Classifier Visualize: CostCurve. (Class value Japan)

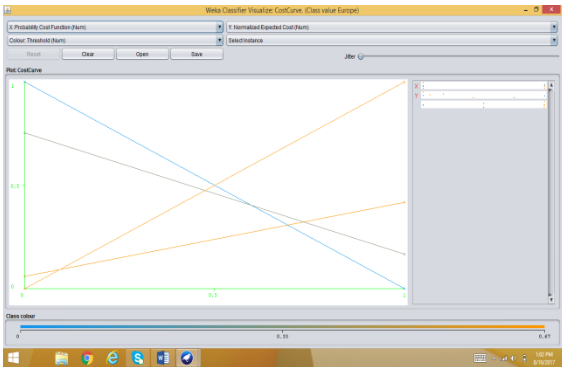
X: Probability Cost Function (Num)

Y: Normalized Expected Cost (Num)

Colour: Threshold (Num)

Select Instance

Fig. 9: Cost curve (Class 1)



Weka Classifier Visualize: CostCurve. (Class value Europe)

X: Probability Cost Function (Num)

Y: Normalized Expected Cost (Num)

Colour: Threshold (Num)

Select Instance

Fig. 10: Cost curve (Class 2)

References

Abbasi A, Chen H, and Salem A (2008). Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. ACM Transactions on Information Systems (TOIS), 26(3): 1-35.

Barbosa L and Feng J (2010). Robust sentiment detection on twitter from biased and noisy data. In the 23rd International Conference on Computational Linguistics: Posters, Association for Computational Linguistics, Beijing, China: 36-44.

Batra S and Rao D (2010). Entity based sentiment analysis on twitter. Science, 9(4): 1-12.

Belavagi MC and Muniyal B (2016). Performance evaluation of supervised machine learning algorithms for intrusion detection. Procedia Computer Science, 89: 117-123.

Bhavsar H and Ganatra A (2012). A comparative study of training algorithms for supervised machine learning. International Journal of Soft Computing and Engineering (IJSCE), 2(4): 2231-2307.

Breiman L (1994). Heuristics of instability in model selection. Technique Report, Statistics Department, University of California at Berkeley, Berkeley, USA.

Breiman L (2001). Random forests. Machine Learning, 45(1): 5-32.

Chavan GS, Manjare S, Hegde P, and Sankhe A (2014). A survey of various machine learning techniques for text classification. International Journal of Engineering Trends and Technology (IJETT), 15(6): 288-292.

Das TK, Acharjya DP, and Patra MR (2014). Opinion mining about a product by analyzing public tweets in Twitter. In the International Conference on Computer Communication and Informatics, IEEE, Coimbatore, India: 1-4. https://doi.org/10.1109/ICCCI.2014.6921727

Devika MD, Sunitha C, and Ganesh A (2016). Sentiment analysis: A comparative study on different approaches. Procedia Computer Science, 87: 44-49.

Drummond C and Holte RC (2000). Explicitly representing expected cost: An alternative to ROC representation. In the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, Boston, Massachusetts, USA: 198-207. https://doi.org/10.1145/347090.347126

Drummond C and Holte RC (2006). Cost curves: An improved method for visualizing classifier performance. Machine Learning, 65(1): 95-130.

Frank A (2010). UCI machine learning repository. University of California, School of Information and Computer Science, Irvine, USA.

Genc Y, Sakamoto Y, and Nickerson J (2011). Discovering context: classifying tweets through a semantic transform based on wikipedia. In: Schmorrow DD and Fidopiastis CM (Eds.), Foundations of augmented cognition: Directing the future of adaptive systems: 484-492. Springer Science and Business Media, Berlin, Germany.

Holmes G, Donkin A, and Witten IH (1994). Weka: A machine learning workbench. In the 2nd Australian and New Zealand Conference on Intelligent Information Systems, IEEE, Brisbane, Qld., Australia: 357-361. https://doi.org/10.1109/ANZIIS.1994.396988

Huang J, Lu J, and Ling CX (2003). Comparing naive Bayes, decision trees, and SVM with AUC and accuracy. In the 3rd IEEE International Conference on Data Mining, IEEE, Melbourne, USA: 553-556. https://doi.org/10.1109/ICDM.2003.1250975

Jaynes ET (1957). Information theory and statistical mechanics. Physical review, 106(4): 620-630.

Kalarikkal S and Remya PC (2015). Sentiment analysis and dataset collection: A comparitive study. In the IEEE International Advance Computing Conference, IEEE, Banglore, India: 519-524. https://doi.org/10.1109/IADCC.2015.7154762

Kang H, Yoo SJ, and Han D (2012). Senti-lexicon and improved Naïve Bayes algorithms for sentiment analysis of restaurant reviews. Expert Systems with Applications, 39(5): 6000-6010.

Kharde V and Sonawane P (2016). Sentiment analysis of twitter data: A survey of techniques. International Journal of Computer Applications, 139(11): 5-15.

Kotalwar R, Gandhi S, and Chavan R (2014). Data mining: Evaluating performance of employee's using classification algorithm based on decision tree. Engineering Science and Technology: An International Journal, 4(2): 29-35.

Kotsiantis SB (2007). Supervised machine learning: A review of classification techniques. Informatica, 31(3): 249-268

Kumar R and Verma R (2012). Classification algorithms for data mining: A survey. International Journal of Innovations in Engineering and Technology (IJIET), 1(2): 7-14.

Kwak H, Lee C, Park H, and Moon S (2010). What is Twitter, a social network or a news media?. In the 19th International Conference on World Wide Web, ACM, Raleigh, North Carolina, USA: 591-600. https://doi.org/10.1145/1772690.1772751

Lane PC, Clarke D, and Hender P (2012). On developing robust models for favourability analysis: Model choice, feature sets and imbalanced data. Decision Support Systems, 53(4): 712-718.

Medhat W, Hassan A, and Korashy H (2014). Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal, 5(4): 1093-1113.

Melville P, Gryc W, and Lawrence RD (2009). Sentiment analysis of blogs by combining lexical knowledge with text classification. In the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, Paris, France: 1275-1284. https://doi.org/10.1145/1557019.1557156

Moraes R, Valiati JF, and Neto WPG (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications, 40(2): 621-633.

Neelamegam S and Ramaraj E (2013). Classification algorithm in data mining: An overview. International Journal of P2P Network Trends and Technology (IJPTT), 4(8): 369-374.

Padmapriya A (2012). Prediction of higher education admissibility using classification algorithms. International Journal of Advanced Research in Computer Science and Software Engineering, 2(11): 330-336.

Pak A and Paroubek P (2010). Twitter as a corpus for sentiment analysis and opinion mining. In LREc, 10(2010): 1320-1326.

Pang B, Lee L, and Vaithyanathan S (2002). Thumbs up?: sentiment classification using machine learning techniques. In the ACL-02 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Stroudsburg, USA: 10: 79-86. https://doi.org/10.3115/1118693.1118704

Parikh R and Movassate M (2009). Sentiment analysis of user-generated twitter updates using various classification techniques. CS224N Final Report: 1-18.

Quinlan JR (1993). C4. 5: Programming for machine learning. Morgan Kaufmann Publishers, Burlington, USA.

Saleh MR, Martín-Valdivia MT, Montejo-Ráez A, and Ureña-López LA (2011). Experiments with SVM to classify opinions in different domains. Expert Systems with Applications, 38(12): 14799-14804.

Shrivatava A, Mayor S, and Pant B (2014). Opinion mining of real time twitter tweets. International Journal of Computer Applications, 100(19):1-4.

Smeureanu I and Bucur C (2012). Applying supervised opinion mining techniques on online user reviews. Informatica Economica, 16(2): 81-91.

Vaghela VB and Jadav BM (2016). Analysis of various sentiment classification techniques. Analysis, 140(3): 22-27.

Vapnik VN (1995). The nature of statistical learning theory. Springer Verlag, Germany.

Wahbeh AH, Al-Radaideh QA, Al-Kabi MN, and Al-Shawakfa EM (2011). A comparison study between data mining tools over some classification methods. International Journal of Advanced Computer Science and Applications, 8(2): 18-26.

Xia R, Zong C, and Li S (2011). Ensemble of feature sets and classification algorithms for sentiment classification. Information Sciences, 181(6): 1138-1152.

Ye Q, Zhang Z, and Law R (2009). Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. Expert Systems with Applications, 36(3): 6527-6535.

Zhang Z, Ye Q, Zhang Z, and Li Y (2011). Sentiment classification of Internet restaurant reviews written in Cantonese. Expert Systems with Applications, 38(6): 7674-7682.