Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 132 (2018) 937–946

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Intelligence and Data Science (ICCIDS 2018)

An Ensemble Classification System for Twitter Sentiment Analysis

Ankit[a,*], Nabizath Saleena[a]

[a]*Department of Computer Science Engineering, NIT Calicut, Kozhikode, Kerala 673601, India*

Abstract

Twitter Sentiment Analysis is the way of identifying sentiments and opinions in tweets. The main computational steps in this process are determining the polarity or sentiment of the tweet and then categorizing them into the positive tweet or negative tweet. The primary issue with Twitter sentiment analysis is the identification of the most suitable sentiment classifier that can correctly classify the tweets. Generally, base classification technique like Naive Bayes classifier, Random Forest classifier, SVMs and Logistic Regression are being used. In this paper, an *ensemble classifier* has been proposed that combines the base learning classifier to form a single classifier, with an aim of improving the performance and accuracy of sentiment classification technique. The results show that the proposed ensemble classifier performs better than stand-alone classifiers and majority voting ensemble classifier. In addition, the role of data pre-processing and feature representation in sentiment classification technique is also explored as part of this work.

Peer-review under responsibility of the scientific committee of the International Conference on Computational Intelligence and Data Science (ICCIDS 2018).

Keywords: Twitter; sentiment analysis; ensemble learning; majority voting; opinion mining; social network; machine learning

1. Introduction

The Social Network, Twitter is a fast growing online platform where people can create, post, update and read short text messages called *tweets*. Through tweets, users can share their opinions, views and thoughts about a specific topic. In general, Sentiment Analysis (SA) is the way of identifying and categorizing the polarity of a given text at document, sentence and phrase level [5]. This technique is being used in many fields like e-commerce, health care, entertainments and politics, to name a few. As an example, Sentiment Analysis is useful for companies to monitor consumer opinions regarding their product, and for consumers to choose the best products based on public opinions. The main task in Twitter SA is to determine the opinion of the tweet is either positive opinion or negative opinion. The main challenges of twitter sentiment analysis are: (1) tweets are generally written in informal language (2) short messages show limited cues about sentiment and (3) acronyms and abbreviations are widely used on twitter.

Generally, base classification techniques like Naive Bayes classifier, Maximum Entropy classifier (Max. Ent.) and SVMs are used to solve Twitter sentiment classification problem[23]. The ensemble classification techniques have

* Corresponding author. Tel.: +91-7994322603
E-mail address: ankitrhode@gmail.com

Peer-review under responsibility of the scientific committee of the International Conference on Computational Intelligence and Data Science (ICCIDS 2018).
10.1016/j.procs.2018.05.109

been widely used in many areas to solve the classification problem. But in case of tweet sentiment analysis, comparatively less research is done on the use of ensemble classifiers. Majority voting is the simplest ensemble classification method for twitter sentiment analysis. But recent studies show that weighted ensemble classifier can improve the robustness and performance of twitter sentiment analysis. [11]. Therefore, a weighted ensemble classifier has been proposed for tweet sentiment analysis. The main focus is to combine the base learners to form a single classifier. The experiments show that the proposed ensemble classifier performs better than stand-alone classifiers and majority voting ensemble classifier. The role of data pre-processing and feature representation in sentiment classification technique is also explored as part of this work.

Ensemble classification approach has been introduced in this section. In section 2, The related work is summarized. In section 3, The details of the proposed method is presented. This is followed by implementation details in Section 4 and experiments, results and visualization of the results in Section 5. Finally, Section 6 present conclusion and path for future work.

2. Related work

Several methods are available in the literature, that use base classifiers for Twitter SA. Medhat et al. [13] presents a survey of Sentiment Analysis algorithms and applications. Davidov et al. [6] and Go et al. [7] determined sentiments by using emoticons and hashtag. Saif et al. [14], Read [20], Agarwal et al. [1] and Zhang et al. [25] integrate lexicon and learning based techniques. They used lexicons and POS as linguistic resources. A. Onan et al. [15] proposed an efficient ensemble classifier using a multi-objective differential evolution algorithm. They compared weighted and unweighted voting schemes. They made no attempt to perform any data preprocessing. N.F.F. da Silva et al. [21] compared feature hashing and bag-of-words for feature representation. They proposed ensemble classifier based on *majority vote* for Twitter sentiment analysis. Lin and Kolcz [12] used feature hashing as feature representation technique and logistic regression as a base learner. Linguistic processing is not fully covered in this paper. The result shows that ensemble classifier performed well. Rodriguez et al. [17] used N-gram, lexicon, POS and Sentiwordnet as feature set. SVMs and Conditional Random Fields are used as base learner. Their ensemble combination of orthogonal methods leads to more accurate classifiers. Hassan et al. [9] developed an ensemble technique which used dataset, feature set and bootstrap aggregation learners. They proposed an algorithm that would select the most appropriate classifier among all the base classifiers. Clark et al. [4] proposed an ensemble classifier which is trained on features like lexical to determine the polarity of each individual phrase within each tweet. The sentiment of a specific phrase may not be same as the sentiment of the whole tweet.

3. Proposed methodology

The proposed system consists of four modules - (1) *Data preprocessing module:* for preprocessing the data (2) *Feature representation module:* for extracting out features from preprocessed tweets (3) *Sentiment classification using base classifiers:* in which different base classifiers are used for sentiment analysis and finally (4) *Sentiment classification using ensemble classifier*. The details regarding each module is presented below.

3.1. Data preprocessing

Data preprocessing module is responsible to decrease the size of the feature set to make it suitable for learning algorithms. This is required because a tweet may contain several features as shown in a sample tweet in Figure 1. Following are the steps in the data preprocessing:

- retweets, which starts with “RT” are eliminated.
- user names preceded by ‘@’ and external links are eliminated.
- hashtag ‘#’ (used to point subjects and phrases that are currently in trending topics) is removed from the tweet.

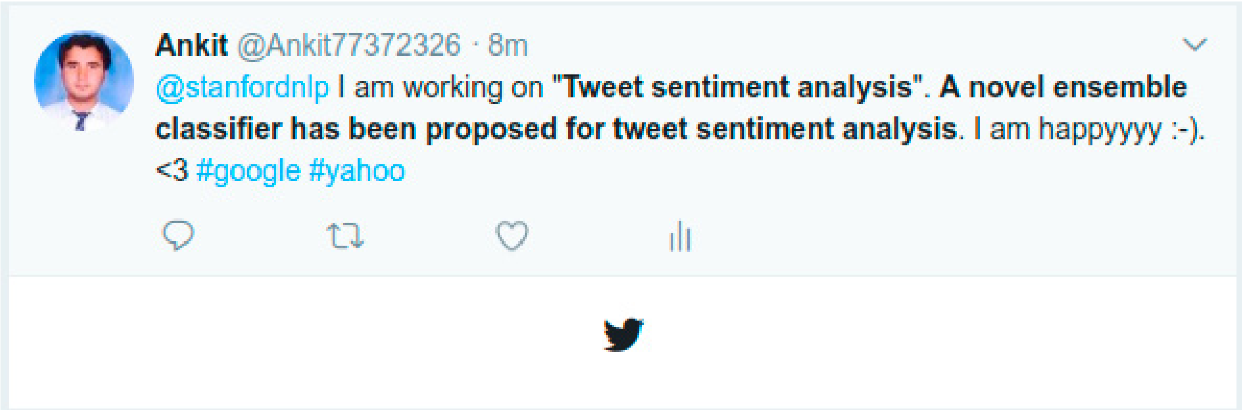

Fig. 1. A sample tweet with various features.

- emoticons[1] are replaced by its equivalent meaning because these can serve as a useful feature to detect sentiments.
- "stemming" is done to reduce each word to its root word.
- slangs are converted to words with equivalent meaning.
- stop-words or useless words are removed from the tweet.

3.2. Feature Representation

This module is responsible to extract features from preprocessed tweets. In this paper, Bag-of-Words technique [8] is used to convert training tweets into numeric representation. Bag-of-Words(BOW) learns a vocabulary of known words from all of the tweets [21]. After learning vocabulary, BOW describes the presence of known words within a tweet. For example, consider the following three tweets:
Tweet1: "yesterday is past" **Tweet2:** "today is present" **Tweet3:** "tomorrow is future".
The vocabulary is **{yesterday, is, past, today, present, tomorrow, future}**
Now, the above tweets are represented as:
tweet1_vector: [1 1 1 0 0 0 0] **tweet2_vector:** [0 1 0 1 1 0 0] **tweet3_vector:** [0 1 0 0 0 1 1]
The parameter values of the BOW are tuned as: $analyzer$ = "$word$", $ngram\_range = (1, 2)$, $max\_features = 4000$

3.3. Sentiment Classification using base classifiers

Base classifiers have been widely used to solve the task of sentiment analysis.

3.3.1. Naive Bayes(NB)

This is a probabilistic classification technique. This classifier performs well when applied to large datasets [8]. NB classifier computes posterior probability by using the formula

$$\text{posterior probability} = \frac{\text{likelihood} \times \text{prior probability}}{\text{evidence}}$$

Equivalently,

$$P(Class_i|z) = \frac{p(z|Class_i) \times P(Class_i)}{p(z)}$$

[1] https://pc.net/emoticons/

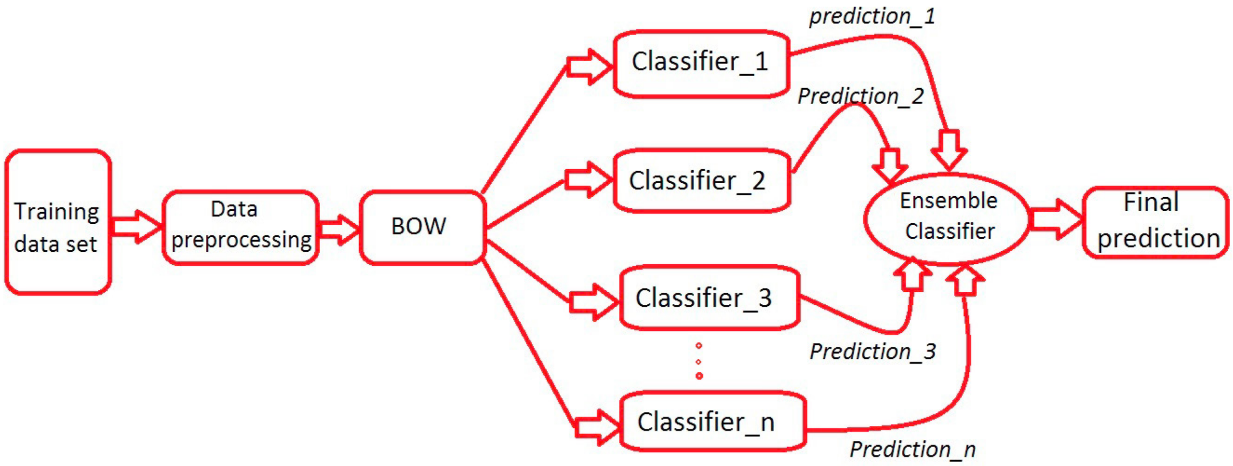

Fig. 2. An overview of tweet sentiment classification approach using ensemble classifier.

Where z represents the feature vector and $Class_i$ represents the i^{th} class.
NB classifier makes an assumption that features are conditionally independent. Smoothing techniques are used to eliminate undesirable effects.

3.3.2. Random Forest(RF)

RF is an ensemble method. Every classifier in the Random Forest is a decision tree classifier. RF classifier builds a set of decision trees from the training dataset [10]. After collecting votes from the different decision trees, it decides the final label or class of the test object. The parameter values of the RF classifiers are tuned as: $n\_estimators = 150$, $max\_depth = 30$.

3.3.3. Support Vector Machine(SVM)

This model requires training data to train the model. It is also called a probabilistic classifier [16]. SVM uses a nonlinear mapping whose aim is to find large margin between different classes. Although training time of SVM can be slow but it is highly accurate. SVMs attempt to find a decision boundary which maximizes the separation gap between the classes. Unlike Naive Bayes classifier, SVM makes no class conditional independence assumption. SVM yields good result for the task of the Twitter SA problem. The parameter values of the SVM classifiers are tuned as: $C = 0.1$, $kernel = linear$.

3.3.4. Logistic Regression(LR)

This is a regression model that is used for classification purpose. LR is generally used to relate a single categorical dependent variable to one or more independent variables[15]. LR attempts to find a hyper-plane which maximizes the separation gap between the classes. The parameter values of the LR classifiers are tuned as: $C = .01$, $max\_iter = 100$.

3.4. Proposed Ensemble Classifier

Ensemble classifier aggregates multiple base classifiers in order to obtain a robust classifier [18]. Generally ensemble classifiers have been used to enhance the performance and accuracy of base learning techniques. Figure 2 shows an overview of tweet sentiment analysis approach using the ensemble classifier. Base learners like NB, RF, SVM, and

LR are used in ensemble classifier.

Algorithm 1: Proposed ensemble algorithm to calculate the Sentiment_score of a tweet

Function Calculate Sentiment_score (Test_tweet);
Input : Test_tweet
Output: Sentiment_score
foreach $Tweet_i$ *in Test_tweet* **do**
  $Positive\_count_i = 0$
  $Negative\_count_i = 0$
  foreach *classifier* c_i *in classifier ensemble* **do**
    if c_i *predict Positive* **then**
      $Positive\_count_i$ += 1;
    end
    else
      $Negative\_count_i$ += 1;
    end
  end

$$Probability(Positive_i) = \frac{Positive\_count_i}{Positive\_count_i + Negative\_count_i}$$

$$Probability(Negative_i) = \frac{Negative\_count_i}{Positive\_count_i + Negative\_count_i}$$

end
foreach *classifier* c_i *in classifier ensemble* **do**

$$Weight_{c_i} = \frac{acc_{c_i}}{\sum_{j=1}^{n} acc_{c_j}}$$

  // Where acc_{c_i} is the accuracy of i^{th} classifier, j denotes the no. of learning classifiers in the ensemble classifier and acc_{c_j} represents to the accuracy of j^{th} learning classifier.
end
foreach $Tweet_i$ *in Test_tweet* **do**
  $Positive\_score_i = 0$
  $Negative\_score_i = 0$
  foreach *classifier* c_i *in classifier ensemble* **do**
    if c_i *predict Positive* **then**
      $Positive\_score_i$ += $Weight$c$_i$ * Probability(Positive$_i$);
    end
    else
      $Negative\_score_i$ += $Weight$c$_i$ * Probability(Negative$_i$);
    end
  end
  return $Positive\_score_i$, $Negative\_score_i$
end

Algorithm 1 calculates sentiment score of the tweet. The system was trained using the training data. The $Test\_tweet$ is a set of tweets that was used to test the system. Each base classifier in ensemble classifier determines the sentiment (Positive/Negative) of each tweet in $Test\_tweet$. In addition, the classification report of each base classifier was calculated on the testing data ($Test\_tweet$). The next step is to calculate the probability of each tweet being positive and negative. After assigning this probability, we assign the weight to each classifier in the ensemble technique based on the accuracy of each classifier. Finally, the algorithm calculates the positive and negative score of the tweet based on the prediction of each classifier.

Algorithm 2 predicts the sentiment of the tweet. The inputs to this algorithm are the positive score and negative score of the tweet. If the positive score of the tweet is more than its negative score, then the sentiment of that tweet is taken as positive. And, If the negative score of the tweet is more than positive score then the sentiment of that tweet is taken as negative. Finally, If the positive score and the negative score of a tweet are equal then the system calculates the cosine similarity of that tweet with all other tweets in the testing data and identifies the most similar tweet. Then it calculates the positive and negative score of the identified tweet. Now if positive score is more than negative score then tweet is positive otherwise it is taken as negative.

Algorithm 2: Proposed ensemble algorithm to predict the sentiment of a tweet

function SentimentPredictor ($Tweet_i$, $Positive\_score_i$, $Negative\_score_i$);
Input : $Tweet_i$, $Positive\_score_i$, $Negative\_score_i$
Output: $Sentiment$
if $Positive\_score_i > Negative\_score_i$ **then**
    Sentiment = "Positive";
else
    if $Negative\_score_i > Positive\_score_i$ **then**
        Sentiment = "Negative";
    else
        Calculate cosine similarity of $Tweet_i$ with all other tweets in test_data using distance calculation formula.
        Find the most similar tweet of $Tweet_i$ in Test_tweet, say $Tweet_j$.
        calculate $Positive\_score_j$ and $Negative\_score_j$ of $Tweet_j$ using Algorithm 1.
        if $Positive\_score_j >= Negative\_score_j$ **then**
            Sentiment = "Positive";
        else
            Sentiment = "Negative";
        end
    end
end
return $Sentiment$

Distance calculation "Cosine similarity" measures the similarity of a pair of tweets [2]. Cosine similarity can be computed by using this formula:

$$cos(tweet_1, tweet_2) = \frac{tweet_1 \cdot tweet_2}{||tweet_1|| \cdot ||tweet_2||}$$

Where $tweet_1$ and $tweet_2$ represent vectors and output value 1 represents high similarity.

4. Implementation details

The implementations are done in *Python*. *Scikit-learn* is used for feature representation, classification, similarity measures and evaluation purpose. *Natural Language Toolkit*(Nltk) is used for stemming and stop word removal in data preprocessing. *Pandas* is used for handling dataset. *NumPy* is used to handle multi-dimensional arrays.

5. Evaluation

5.1. Datasets

The proposed system was tested on the following datasets collected from Twitter pertaining to different topics:

5.1.1. Stanford - Sentiment140 corpus

Sentiment140 dataset [7] is generally used to train and test the system. This consists of 1,600,000 training tweets with eight lakh tweets labelled positive label and eight lakh labelled negative.

5.1.2. Health Care Reform (HCR)

This dataset [22] was assembled by searching tweets with the #hcr. The tweets with positive sentiment and negative sentiment are considered for the experiment. This dataset consists of 888 tweets (365 positive and 523 negative).

5.1.3. First GOP debate twitter sentiment dataset

This dataset from Crowdflower consists tweets on the first GOP debate for the 2016 presidential nomination. This GOP debate dataset consists of 13871 tweets with the positive, negative or neutral sentiment. Neutral sentiment tweets were not taken for research. So final dataset contains 10729 tweets with 2236 positive and 8493 negative labels.

5.1.4. Twitter sentiment analysis dataset

This dataset has 99989 training tweets. Each tweet is either positive or negative. This dataset consists of 43532 negative and 56457 positive tweets. This dataset is available at kaggle.

5.2. Results

Table 1. Cross comparison of the results obtained from base classifiers, majority voting ensemble and proposed ensemble classifier. Pre, Rec and F1 refer to the Precision, Recall and F-measure.

Techniques	Accuracy(%)	Positive class			Negative class			Average
		Pre(%)	Rec(%)	F1(%)	Pre(%)	Rec(%)	F1(%)	F1(%)
Stanford- Twitter Sentiment Corpus								
Naive Bayes	75.19	75.63	74.47	75.05	74.76	75.91	75.33	75.19
Random Forest	71.76	67.78	83.18	74.70	78.12	60.30	68.06	71.38
Support Vector Machine	75.61	73.98	79.18	76.49	77.50	72.03	74.67	75.58
Logistic Regression	74.15	72.37	78.30	75.21	76.25	69.98	72.98	74.09
Majority Voting	74.80	71.43	82.83	76.71	79.47	66.73	72.55	74.63
Proposed Ensemble	75.81	74.80	78.00	76.36	76.91	73.61	75.22	**75.79**
Health Care Reform Dataset								
Naive Bayes	71.80	59.37	61.29	60.32	78.82	77.46	78.13	69.22
Random Forest	71.43	67.35	35.48	46.48	72.35	90.75	80.51	63.49
Support Vector Machine	70.30	56.86	62.36	59.49	78.66	74.57	76.56	68.02
Logistic Regression	69.92	65.85	29.03	40.30	70.67	91.91	79.90	60.10
Majority Voting	71.05	66.00	35.48	46.15	72.22	90.17	80.20	63.17
Proposed Ensemble	73.68	63.85	56.99	60.23	78.14	82.66	80.34	**70.28**
First GOP Debate Dataset								
Naive Bayes	82.20	57.48	69.24	62.82	90.95	85.79	88.30	75.56
Random Forest	82.57	85.20	23.89	37.32	82.40	98.85	89.88	63.60
Support Vector Machine	83.44	62.17	60.66	61.40	89.16	89.76	89.46	75.43
Logistic Regression	81.51	83.77	18.45	30.25	81.40	99.01	89.35	59.80
Majority Voting	82.60	85.64	23.89	37.36	82.41	98.89	89.90	63.63
Proposed Ensemble	85.83	73.59	54.22	62.44	88.16	94.60	91.27	**76.85**
Twitter Sentiment Analysis Dataset								
Naive Bayes	73.65	77.96	77.18	77.57	67.57	68.56	68.06	72.81
Random Forest	70.61	68.73	92.13	78.73	77.73	39.58	52.45	65.59
Support Vector Machine	74.36	76.06	82.56	79.18	71.33	62.55	66.65	72.91
Logistic Regression	73.44	73.89	85.08	79.09	72.49	56.68	63.61	71.35
Majority Voting	73.83	72.99	88.36	79.94	75.92	52.88	62.34	71.14
Proposed Ensemble	74.67	76.62	82.18	79.30	71.31	63.85	67.37	**73.33**

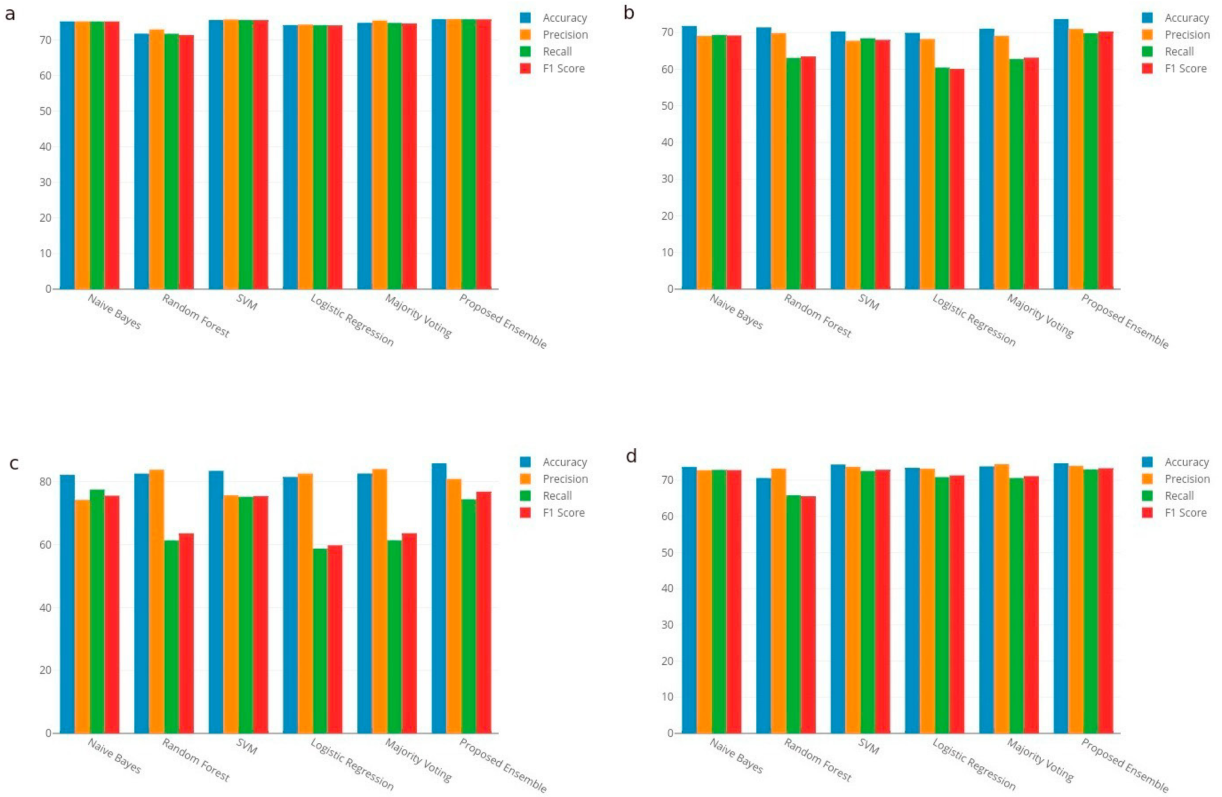

Fig. 3. Performance evaluation of different classifiers with (a) Stanford dataset; (b) HCR dataset; (c) GOP debate dataset; (d) Twitter sentiment dataset.

Evaluation metrices are illustrated as [8]

$$Recall = \frac{True\_Pos\_Sentiment}{True\_Pos\_Sentiment + False\_Neg\_Sentiment}$$

$$Precision = \frac{True\_Pos\_Sentiment}{True\_Pos\_Sentiment + False\_Pos\_Sentiment}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

$$Accuracy = \frac{True\_Pos\_Sentiment + True\_Neg\_Sentiment}{True\_Pos\_Sentiment + False\_Neg\_Sentiment + False\_Pos\_Sentiment + True\_Neg\_Sentiment}$$

The performance of the proposed ensemble classifier is compared with the individual traditional classifier and majority voting ensemble classifier. The results are shown in Table 1. Stanford - Sentiment140 corpus consist of 1.6 million tweets. Bakliwal et al. [3], Go et al. [7], Sperioosu et al. [22] and Prusa et al. [19] are also used this dataset to evaluate their system. Due to the computational limitation of system, it is very difficult to test the proposed system with 1.6 million tweets. Therefore, only 1,00,000 tweets are used for experiments as sampling which is only 6.25% of total tweets to test the proposed system. Over 1,00,000 tweets, 70,000(70%) tweets are used for training the system and 30,000(30%) are used for testing the system. The results show that the proposed ensemble classifier performed better than stand-alone classifier and majority voting ensemble classifier on different types of datasets.

6. Conclusion

In the area of twitter SA, the major approach is to compare the different base classifiers and select the best among them to implement tweet SA. The ensemble classification techniques have been widely used in many areas to solve the classification problem. But in case of tweet sentiment analysis, comparatively little work has been done on the use of ensemble classifiers. In this paper, an ensemble classification system has been proposed which is shown to improve the performance of tweet sentiment classification. The performance of the proposed ensemble classification system has been compared with various traditional sentiment analysis methods and most popular majority voting ensemble classification system. The ensemble classification system is formed by different base learners like Naive Bayes classifier, Random Forest classifier, SVMs, and Logistic Regression. The results show that proposed ensemble classifier performs better than stand-alone classifiers and the popular majority voting ensemble classifier. This approach is applicable for companies to monitor consumer opinions regarding their product, and for consumers to choose the best products based on public opinions. As future work, the main focus will be the study of the neutral tweets because some of the tweets have neither positive sentiment nor negative sentiment. Even though proposed work is in the domain of twitter data, it implies that this paper can be extended to the analysis of data on other social network platforms as well.

References

[1] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R., 2011. Sentiment analysis of twitter data, in: Proceedings of the workshop on languages in social media, Association for Computational Linguistics. pp. 30–38.

[2] Azzouza, N., Akli-Astouati, K., Oussalah, A., Bachir, S.A., 2017. A real-time twitter sentiment analysis using an unsupervised method, in: Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics, ACM, New York, NY, USA. pp. 15:1–15:10. URL: `http://doi.acm.org/10.1145/3102254.3102282`, doi:10.1145/3102254.3102282.

[3] Bakliwal, A., Arora, P., Madhappan, S., Kapre, N., Singh, M., Varma, V., 2012. Mining sentiments from tweets., in: WASSA@ ACL, pp. 11–18.

[4] Clark, S., Wicentwoski, R., 2013. Swatcs: Combining simple classifiers with estimated accuracy., in: SemEval@ NAACL-HLT, pp. 425–429.

[5] Coletta, L.F.S., da Silva, N.F.F., Hruschka, E.R., Hruschka, E.R., 2014. Combining classification and clustering for tweet sentiment analysis, in: Intelligent Systems (BRACIS), 2014 Brazilian Conference on, IEEE. pp. 210–215.

[6] Davidov, D., Tsur, O., Rappoport, A., 2010. Enhanced sentiment learning using twitter hashtags and smileys, in: Proceedings of the 23rd international conference on computational linguistics: posters, Association for Computational Linguistics. pp. 241–249.

[7] Go, A., Bhayani, R., Huang, L., 2009. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford 1, 12.

[8] Han, J., 2005. Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[9] Hassan, A., Abbasi, A., Zeng, D., 2013. Twitter sentiment analysis: A bootstrap ensemble framework, in: Social Computing (SocialCom), 2013 International Conference on, IEEE. pp. 357–364.

[10] Jianqiang, Z., 2016. Combing semantic and prior polarity features for boosting twitter sentiment analysis using ensemble learning, in: Data Science in Cyberspace (DSC), IEEE International Conference on, IEEE. pp. 709–714.

[11] Kim, H., Kim, H., Moon, H., Ahn, H., 2011. A weight-adjusted voting algorithm for ensembles of classifiers. Journal of the Korean Statistical Society 40, 437 – 449. URL: `http://www.sciencedirect.com/science/article/pii/S1226319211000238`, doi:https://doi.org/10.1016/j.jkss.2011.03.002.

[12] Lin, J., Kolcz, A., 2012. Large-scale machine learning at twitter, in: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, ACM. pp. 793–804.

[13] Medhat, W., Hassan, A., Korashy, H., 2014. Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal 5, 1093 – 1113. URL: `http://www.sciencedirect.com/science/article/pii/S2090447914000550`, doi:https://doi.org/10.1016/j.asej.2014.04.011.

[14] Mohammad, S., Kiritchenko, S., Zhu, X.D., 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets, in: SemEval@NAACL-HLT.

[15] Onan, A., Korukolu, S., Bulut, H., 2016. A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification. Expert Systems with Applications 62, 1 – 16. URL: `http://www.sciencedirect.com/science/article/pii/S095741741630286X`, doi:https://doi.org/10.1016/j.eswa.2016.06.005.

[16] Pang, B., Lee, L., Vaithyanathan, S., 2002. Thumbs up?: sentiment classification using machine learning techniques, in: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, Association for Computational Linguistics. pp. 79–86.

[17] Penagos, C.R., Batalla, J.A., Codina-Filbà, J., Narbona, D.G., Grivolla, J., Lambert, P., Saurí, R., 2013. Fbm: Combining lexicon-based ml and heuristics for social media polarities, in: SemEval@NAACL-HLT.

[18] Prusa, J., Khoshgoftaar, T.M., Dittman, D.J., 2015a. Using ensemble learners to improve classifier performance on tweet sentiment data, in: Information Reuse and Integration (IRI), 2015 IEEE International Conference on, IEEE. pp. 252–257.

[19] Prusa, J., Khoshgoftaar, T.M., Napolitano, A., 2015b. Utilizing ensemble, data sampling and feature selection techniques for improving classification performance on tweet sentiment data, in: Machine Learning and Applications (ICMLA), 2015 IEEE 14th International Conference on, IEEE. pp. 535–542.

[20] Read, J., 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification, in: Proceedings of the ACL student research workshop, Association for Computational Linguistics. pp. 43–48.
[21] da Silva, N.F., Hruschka, E.R., Hruschka, E.R., 2014. Tweet sentiment analysis with classifier ensembles. Decision Support Systems 66, 170 – 179. URL: `http://www.sciencedirect.com/science/article/pii/S0167923614001997`, doi:https://doi.org/10.1016/j.dss.2014.07.003.
[22] Speriosu, M., Sudan, N., Upadhyay, S., Baldridge, J., 2011. Twitter polarity classification with label propagation over lexical links and the follower graph, in: Proceedings of the First workshop on Unsupervised Learning in NLP, Association for Computational Linguistics. pp. 53–63.
[23] Wan, Y., Gao, Q., 2015. An ensemble sentiment classification system of twitter data for airline services analysis, in: Data Mining Workshop (ICDMW), 2015 IEEE International Conference on, IEEE. pp. 1318–1325.
[24] Wang, H., Can, D., Kazemzadeh, A., Bar, F., Narayanan, S., 2012. A system for real-time twitter sentiment analysis of 2012 us presidential election cycle, in: Proceedings of the ACL 2012 System Demonstrations, Association for Computational Linguistics. pp. 115–120.
[25] Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., Liu, B., 2011. Combining lexicon-based and learning-based methods for twitter sentiment analysis.