Journal of Engineering and Applied Sciences 12 (20): 5226-5232, 2017
ISSN: 1816-949X

Enhanced Malay Sentiment Analysis with an Ensemble Classification Machine Learning Approach

[1]Tareq Al-Moslmi, [1]Nazlia Omar, [2]Mohammed Albared and [1]Adel Alshabi
[1]Center for Artificial Intelligence Technology, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Malaysia
[2]Faculty of Computer and Information Technology, Sana'a University, Sana'a, Yemen

Abstract: Sentiment analysis is one of the challenging and important tasks that involves natural language processing, web mining and machine learning. This study aims to propose an enhanced ensemble of machine learning classification methods for Malay sentiment analysis. Three classification approaches (Naive Bayes, Support vector machine and K-Nearest Neighbour) and five ensemble classification algorithms (Bagging, Stacking, Voting, AdaBoost and MetaCost) were experimented to achieve the best possible ensemble model for Malay sentiment classification. A wide range of ensemble experiments are conducted on a Malay Opinion Corpus (MOC). This study demonstrates that ensemble approaches improve the performance of Malay sentiment-based classification, however, the results depend on the classifier used and the ensemble algorithm as well as the number of classifiers in the ensemble approach. The experiments also show that the ensemble classification approaches achieve the best result with an F-measure of 85.81%.

Key words: Malay sentiment analysis, opinion mining, machine learning, classification, approaches achieve, sentiment-based

INTRODUCTION

The web has turned into the most vital spot for expressing opinion, sentiments and reviews about policies, services and products. An extensive number of individuals openly exchange their sentiment and opinions through online social networking and review sites. The significant growth of the user-generated content of "What other individual's think" represents the extremely important information source for many interested groups. Identifying and analyzing useful review efficiently and rightly to fulfill both present and potential client needs have turned into a critical challenge for market-driven product design. As of late, data mining and natural language processing have been attracting many interests, especially to develop text analysis and mining techniques with the ability of correctly extracting people's sentiment from large volumes of review in unstructured text (Al-Moslmi *et al*., 2017a).

Sentiment analysis and classification is a key issue in a special type of text classification that focuses on classifying reviews of overall sentiment polarity into positive or negative categories. There is a diversity of methods and approaches for sentiment classification and opinion mining. The majority of techniques fall into two main methodologies: supervised (Deng *et al*., 2014) and unsupervised learning approaches (Hu *et al*., 2013). In the supervised machine learning approach, sentiment corpora are used to train classifiers. Most of the studies on sentiment classification consider only English reviews, perhaps due to the lack of resources in other languages. Work on other languages is still growing (Al-Moslmi, 2014; Al-Moslmi *et al*., 2017b; Albared *et al*., 2016). The lack of language resources, i.e., annotated training corpora is a general problem even for well-studied languages. The Malay language also suffers from the same problem. The Malay language is widely used in Malaysia, Brunei, Singapore and Indonesia with approximately 300 million users. However, people typically use their own language to express their experiences, opinions and points of view. Consequently, the need for constructing resources and tools for subjectivity and sentiment analysis in languages other than English is growing. The research presented in this study is mainly motivated by the need to develop sentiment classification systems in the Malay language.

In this study, we aim to make an intensive study of the effectiveness of ensemble techniques for Malay sentiment classification tasks. First, we utilize Naive Bayes (NB), Support Vector Machine (SVM) and

Corresponding Author: Tareq Al-Moslmi, Center for Artificial Intelligence Technology, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Malaysia

K-Nearest Neighbor (KNN) as the base-classifiers to predict classification scores. In the ensemble stage, we apply five algorithms of meta-classification of ensemble method (Bagging, Stacking, Voting, AdaBoost and MetaCost). A wide range of comparative experiments are conducted on Malay Opinion Corpus (MOC) datasets.

Literature review: Several approaches have been proposed for sentiment analysis. These approaches can be classified into three main categories; lexicon based approaches (Xianghua *et al*., 2013; Kang *et al*., 2012; Moreo *et al*., 2012; Allison, 2008; Ba-Alwi *et al*., 2017) machine learning approaches and hybrid approaches (Khan *et al*., 2014; Ghiassi *et al*., 2013; Omar *et al*., 2014). However, there is a lack of research on Malay sentiment analysis and only a few research have been published (Samsudin *et al*., 2011; Isa *et al*., 2013; Alsaffar and Omar, 2014; Sharma and Dey, 2012).

By Khan *et al*. (2014), a comparative study has been conducted to evaluate the effect of feature selection methods on the performance of machine learning classification methods for Malay sentiment analysis.

This research introduces the use of ensemble classification algorithms for Malay sentiment classification tasks.

MATERIALS AND METHODS

The methodology used in Malay sentiment analysis models is shown in (Fig. 1). First, pre-processing tasks are used to eliminate the incomplete noisy and inconsistent data. Data must be pre-processed to perform any further data mining functionality. Then, classification task have been conducted using three classifiers; Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbours (KNN). After that five of the ensemble's algorithms (Bagging, Stacking, Voting, AdaBoost and MetaCost) have been used as a meta-classifier to combine the output of the three machine learning methods.

Data collection: The corpus contains 2000 movie reviews collected from different web pages and blogs in Malay; 1000 of them are considered positive reviews and the other 1000 are considered negative. Table 1 shows the example of positive and negative review in MOC corpus with their English translation. MOC corpus can be downloaded from Github website (https://github.com/almoslmi/MOC).

Pre-processing: Data pre-processing comprises two steps; tokenization and normalization. All of the reviews involve a pre-processing stage.

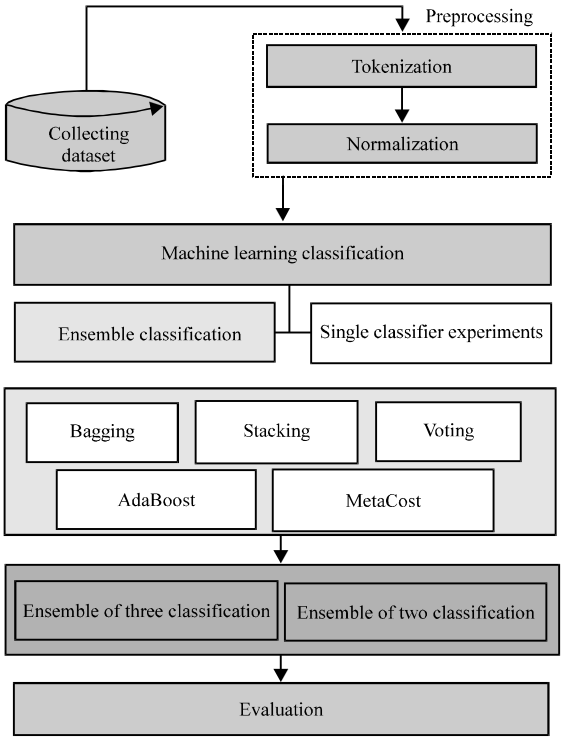

Fig. 1: Architecture of ensemble classification model for Malay sentiment analysis

Classification method: In this study, three classifier methods are used in Malay sentiment classification; the NB, SVM and KNN. These methods are used due to their simplicity, effectiveness and accuracy.

Single classifier method

Support vector machine classifier: SVM is considered to be one of the most effective classification methods according to its performance on text classification as proven by many researchers (Hu *et al*., 2013).

Based on the structural risk minimization principle from computational learning theory, SVMs seek a decision surface to separate the training data points into two classes and to make decisions based on the support vectors that are selected as the only effective elements in the training set. Multiple variants of SVMs have been developed. In this study, our discussion is limited to linear SVMs due to their popularity and high performance in text categorization. The optimization procedure of SVMs (dual form) is to minimize the following:

$$\vec{\alpha} = \arg\min\left\{-\sum_{i=1}^{n}\alpha_i + \sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \left\langle \vec{x_i}, \vec{x_j} \right\rangle\right\} \quad (1)$$

$$\text{subject to}: \sum_{i=1}^{n}\alpha_i y_i = 0;\ 0 \le \alpha_i \le C$$

Table 1: An example of the positive and negative review in MOC corpus

Review id	Malay review	Polarity
Rawneg 988	Just got back from watching this movie with my 3 year old son. He shouted loudly at the cinema saying "This movie is not nice, Umi. I want Ipin movie!" To me, it is an average movie. There are certain parts that made me chuckled but most of the time I was sleepy. The 2/5 is the best I could give	Negative
Rawpos 790	I like the movie. Some said it was boring, probably they are lazy to think and watch heavy drama? I give 8/10 in rating. In fact, I woul want to see this movie again if I have the opportunity	Positive

Naive bayes classifier: The NB algorithm is a widely used algorithm for review classification. Given a feature vector table, the algorithm computes the posterior probability that the review belongs to different classes and assigns it to the class that has the highest posterior probability. There are two commonly used models (i.e., the multinomial model and multi-variate Bernoulli Model) for applying the NB approach to text categorization. NB assumes a stochastic model of document generation and uses Baye's rule. To classify the most probable class c* for a new document d, NB computes:

$$c^* = \text{argmax}_c P(c/d) \quad (2)$$

K-nearest neighbor classifier: The KNN is a well-known example-based classifier. The KNN has been called lazy learners because it defers its decision on how to generalize beyond the training data until each new query instance is encountered. To categorize a review, the KNN classifier ranks the review's neighbors among the training reviews. Then, the KNN uses the class labels of the K most similar neighbors.

Given a test review d, the system finds the K nearest neighbors among the training reviews. The similarity score of each nearest neighbor review to the test review is used as the weight of the classes of the neighbor review. The weighted sum in KNN classification can be written as follows:

$$\text{score}(d, t_i) = \sum_{d_j=\text{KNN}(d)} \text{sim}(d, d_j)\delta(d_j, c_i) \quad (3)$$

where KNN (d) indicates the set of K nearest neighbors of review d. If d_j belongs to c_i, then δ (d_j, c_i) equals one; otherwise, it is zero. For test review d, it should belong to the class that has the highest resulting weighted sum.

Ensemble classification: In this part, five ensemble algorithms (Bagging, Stacking, Voting, AdaBoost and MetaCost) were used as an ensemble classifier to combine the output of the three machine learning methods.

Bagging: Bagging is one of the ensemble classification algorithms that uses only one base-level machine classifier at one go (Breiman, 1996). In this algorithm, each classifier will be trained on a random redistribution of the training set. Therefore, each training set for every single classifier is randomly generated by drawing, with replacement, N samples from the raw training set. N here, indicates the raw training set size. Some of the examples in the original set may be repeated in the training set's result while the rest may be not repeated. The last bagged estimator, h_{bag} (.) is the expected prediction value over every trained hypothesis. h_k (.) is the value if the hypothesis trained for training instance k.

Stacking: For combination using a meta-classifier, the output for all the class labels of the component classifier are viewed as new features for meta-learning. Among the various kinds of classification models, Naive bayes is used to combine the output of the three classifiers. The stacking combination consists of two phases. In the first phase a set of base-level classifiers is generated. In the second phase a meta-level classifier is learned that combines the outputs of the base-level classifiers. When using a meta-classifier for combination, the outputs of all the labels of the class of the participating classifiers are used as features for meta-learning. In this case to combine the output of the three classifiers Naive bayes, KNN and SVM, the Naive Bayes (NB) can be used as a meta-classifier. The formula of the NB as meta-classifier, given the output of three classifiers $o_{1\text{-}3}$:

$$P(c_i \mid o_1, o_2, o_3) = \frac{P(c_i) \times P(o_1, o_2, o_3 \mid c_i)}{P(o_1, o_2, o_3)} \quad (4)$$

Where:

$P(c_i) \times P(o_1, o_2, o_3)$ = The posterior probability of the class

c_i = The new output of the three classifiers

$o_1, o_2, o_3, P(c_i)$ = The probability of class

Voting: The voting algorithm enumerates the outcomes of each single classifier (Omar *et al*., 2013):

$$O_j = \sum_{K=0}^{D} I\left(\text{argmax}\left(O_{kj}\right) = j\right) \quad (5)$$

Where:
I (…) = The indicator function
O_{kj} = The outcome of J classifier

AdaBoost: Boosting algorithm is used to enhance the classification performance of any specific base-level classifier (Meir and Ratsch, 2003). It is repeatedly applied to individual learning algorithm and integrate the hypothesis trained every time (using voting) like that the last classification performance is enhanced. It does this by giving a specific weight to every instance in the training set and after that amending the weight after every iteration based on whether the instance was correctly or incorrectly labeled by the current hypothesis. Therefore, the last hypothesis trained can be given as:

$$f(x) = \sum_{t=1}^{T} a_t h_t(x) \quad (6)$$

Where:
a_t = The coefficient with which the hypothesis h_t is combined
a_t and h_t = The trained during the boosting procedure

MetaCost: MetaCost (Domingos, 1999) is depending on the Bayes optimal prediction that reduces the expected cost R(j|x) (Ting, 2002):

$$R(j|x) = \sum_{i}^{1} P(i|x)\,cost(i, j) \quad (7)$$

Where:
P(i|x) = The class i probability given instance x
cost (i, j) = The cost of misclassifying a class i instance as class j

Experimental setup: We conduct several experiments to evaluate our model using RapidMiner 5.3. First, we evaluate the performance of the classification algorithms. We measure the performance of these classification algorithms on a collected corpus (MOC). All of the algorithms are evaluated using K-fold cross-validation. The objective of this step is to tune the parameters and select the best methods for Malay sentiment analysis. To measure the performance of these classification methods, experimental results are sorted into the following; True Positive (TP) is the set of reviews that is correctly assigned to the given category, False Positive (FP) is the set of reviews that is incorrectly assigned to the category, False Negative (FN) is the set of reviews that is incorrectly not assigned to the category and True Negative (TN) is the set of the set of reviews that is correctly not assigned to the category. However, we use the F1 and macro-F1 measures. The following describes these metrics:

$$Precision = \frac{TP}{(TP+FP)} \quad (8)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (9)$$

$$F_1 = \frac{2\times Recall\times Precision}{(Recall+Precision)} \quad (10)$$

$$F_1^{macro} = \frac{1}{m}\sum_{i=1}^{m} F_1(i) \quad (11)$$

RESULTS AND DISCUSSION

Results of individual classifiers: To examine the classifier's overall performance on Malay sentiment analysis without any reduction, NB, SVM and KNN classifiers are initially applied on the entire document-term feature space. The experimental results using the NB, SVM and KNN classifiers are summarized in Table 2. The experiments were condcuted without using any feature reduction or ensemble methods. The highest performance is obtained with the NB classifier and the lowest performance is obtained by the SVM classifier.

In addition to the comparative studies which have been done before by Alsaffar and Omar (2014) and Al-Moslmi *et al.* (2015) we extended the experiments in this paper to include nine feature selection methods. The macro-averaging F-measure results for the NB classifier with the nine Feature Selection Methods (FSM) at different feature subset sizes are presented in Table 3.

The 7 FSMs (IG, PCA, SVM, Relief, Chi, Gini and uncertainty) perform lower than the classifier without FMSs. The IG FSM typically yields the best performance in terms of the macro-averaging F-measure (the average row in Table 2. According to Table 3, the highest performance (80.88%) of the NB classifier is obtained when using 100 of the weighted features from the IG-based methods. The macro-averaging F-measure results for the SVM classifier with the seven FSM selection methods at different feature subset sizes, as it presented in Table 3. All the seven FSMs perform better than the original classifier. GI and IG tend to yield the highest performance in the terms of macro-averaging the F-measure (the average row in Table 3). According to Table 4, the highest performance (85.33) of the IG classifier is obtained when using 300 of the weighted features by the IG method.

For the KNN classifier with the seven FSM selection methods at different feature subset sizes, the

Table 2: Performance (the average value of macro-F1 and the F-measure for each class) of the NB, SVM and KNN classifiers

Classification	Macro F-measure (%)	F1 measure for the positive class (%)	F1 measure for the negative class (%)
NB	82.32	81.28	83.36
KNN	76.01	75.34	76.67
SVM	57.27	72.45	42.08

Table 3: Macro-averaging precision values for the NB classifier with the seven FSMs with different sizes of feature sets

	NB								
Feature size	CHI	IG	GI	RE	SVM	PCA	CRE	Correlation	IGR
100	79.13	80.88	80.83	72.62	76.42	60.86	78.03	80.08	74.07
200	80.28	80.58	80.48	75.43	74.62	65.32	79.78	79.33	76.68
300	79.38	80.08	80.13	77.87	73.72	69.21	79.53	78.28	75.28
400	77.83	78.78	78.23	76.93	71.12	71.52	76.98	76.43	74.72
500	76.48	77.23	77.28	75.83	70.12	73.67	76.33	74.83	74.83
600	74.83	74.83	74.83	74.83	74.83	74.83	74.83	74.88	74.88
Average	77.99	78.73	78.63	75.59	73.47	69.24	77.58	80.08	74.07

Table 4: Macro-averaging precision values for the SVM classier with the seven FSMs with different sizes of feature sets

	SVM								
Feature size	CHI	IG	GI	RE	SVM	PCA	CRE	Correlation	IGR
100	82.08	84.24	83.63	79.38	61.76	64.47	84.08	82.28	76.53
200	84.03	83.28	83.38	82.03	57.86	67.87	82.63	81.18	80.88
300	83.98	85.33	85.23	83.13	57.01	69.87	82.63	81.78	81.43
400	83.18	84.63	84.88	83.29	57.56	73.67	82.93	80.38	80.83
500	82.03	82.89	82.94	82.59	65.06	80.78	82.13	80.08	80.08
600	82.13	82.13	82.13	82.13	82.13	82.13	82.13	79.78	79.78
Average	82.91	83.75	83.70	82.09	63.56	73.13	82.76	82.28	76.53

Table 5: Macro-averaging precision values for the KNN classier with the seven FSMs with different sizes of feature sets

	KNN								
Feature size	CHI	IG	GI	RE	SVM	PCA	CER	Correlation	IGR
100	71.17	74.57	74.68	69.67	68.17	62.26	72.37	75.73	70.52
200	67.07	69.77	70.32	69.82	64.86	60.91	68.32	68.82	67.52
300	62.86	68.22	67.52	66.42	63.71	62.96	66.11	65.12	61.71
400	62.66	64.07	63.56	65.07	62.92	60.61	63.76	62.86	63.66
500	62.41	62.51	62.76	64.26	62.56	60.16	62.61	62.46	62.46
600	62.36	62.36	62.36	62.36	62.36	62.36	62.36	62.46	62.46
Average	64.76	66.92	66.87	66.27	64.10	61.54	65.92	75.73	70.52

macro-averaging F-measure results are showed in Table 5. All the FSMs perform lower than the original classifier. GINI performs best in terms of macro- averaging the F-measure (the average row in Table 4). According to Table 5, the highest performance (74.68%) of the KNN classifier is obtained when using 100 of the weighted features from the GI method.

Comparing the classifiers performances (Table 2 and 3), the SVM algorithm outperforms the NB and KNN algorithms. Furthermore, the highest accuracies are obtained when the feature selection operations are made by the IG-based method. In general, using FSMs positively contributed to the performance of all classifiers (Table 2-5) in an affirmative manner. As noted from results reported in this experiment and previous experiments, there is a great effect of the feature selection methods on the performance of the individual classifiers in general.

Results of ensemble of classification algorithms: After that the three machine-learning classifiers (KNN, SVM and NB) are used for all ensemble algorithms to determine the importance of these algorithms. All experiments have been conducted on five different ensemble algorithms (Bagging, Stacking, Voting, AdaBoost and Metacost).

Table 6: Performance (the average value of macro-F1 and the F-measure for each class) of the ensemble of two classifiers

Ensemble algorithm	NB+KNN	SVM+NB	SVM+KNN
Bagging	69.12	69.12	64.44
Stacking	72.26	72.62	72.62
Voting	77.30	78.30	81.64
AdaBoost	73.79	73.79	72.12
MetaCost	74.29	73.29	68.95

In the first experiment an ensemble of two classifiers is applied to the test set using 10-fold cross-validation. As shown in Table 6, the stacking algorithm outcomes are almost similar for all experiments in this part. Moreover, AdaBoost algorithm's results slightly change and MetaCost algorithm has the most changes from one experiment to another as it can be noticed in Table 6.

However, the voting ensemble algorithm outperforms all other ensemble algorithms in all these experiments while the worst performance is obtained by the bagging

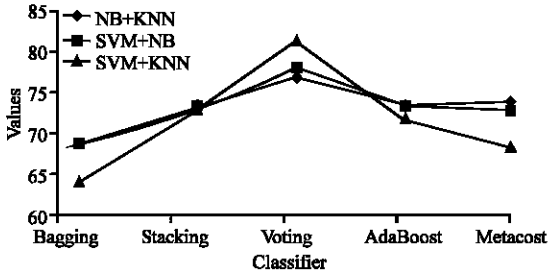

Fig. 2: Results of ensemble of two classifiers

Table 7: Performance (the average value of macro-F1 and the F-measure for each class) of the ensemble of three classifiers

Ensemble algorithm	SVM+NB+KNN
Bagging	72.12
Stacking	73.29
Voting	82.14
AdaBoost	72.12
MetaCost	85.81

algorithm. In addition, the best result achieved in this part is obtained by voting algorithm when SVM and KNN were used.

Comparing the classifiers performances in Fig. 2 an ensemble of SVM and KNN outperforms the ensemble of SVM with NB as well as NB with KNN algorithms. In general, we can notice that the performance of ensemble between NB with other classifiers is stable and almost similar while the ensemble of SVM and KNN is unstable and the results are diverse.

In the 2nd phase of ensemble experiments, we aim to understand the effect of the aforementioned ensemble algorithms on classifiers combination method which combines the three classifiers (NB, SVM and KNN classifier) for Malay sentiment analysis. The classifier combination method was applied on the test set by using 10-fold cross-validation. Table 7 shows the F-measure of the Malay sentiment analysis by applying the ensemble algorithms with a combined classifiers.

As shown Table 7, the use of MetaCost algorithm has an obvious effect on the quality of Malay sentiment analysis. It is also noticeable that the voting algorithm achieved the second best result in this experiments. Furthermore, Bagging and AdaBoost algorithms obtained similar and the lowest performance.

In conclusion, these results indicate that the NB classifier is the best individual classifier for Malay sentiment analysis when it is applied without any feature selection method while SVM classifier is best individual machine learning technique when it is used with IG feature selection method. Furthermore, as noted from results reported and the results of the individual classifiers, the results obtained using ensemble algorithms out performed the results obtained using individual classifiers for Malay sentiment analysis. These results indicate that the classifier combination using ensemble algorithms is the most suitable technique for Malay sentiment analysis. In addition, we noticed that using feature selection method together with the ensemble algorithms reduce the model's performance. Moreover, the use of ensemble algorithms has an obvious effect on the quality of Malay sentiment analysis. It is also clear that MetaCost ensemble algorithms have a higher effect on the performance of KNN, NB and SVM classification model than other methods.

CONCLUSION

This study empirically evaluates three individual classifiers (Naive Bayes, Support Vector Machine and K-Nearest Neighbour) and 5 algorithms of ensemble classification (Bagging, Stacking, Voting, AdaBoost and Metacost) for Malay sentiment analysis. A wide range of experiments are conducted on a Malay Opinion Corpus (MOC). This study demonstrates that using ensemble classification improve the performance of the classification approaches for Malay sentiment classification. Experimental results demonstrate that using ensemble classification is an effective way to combine different classification algorithms for better classification performance. The experimental results also show that the ensemble classification of SVM++NB+KNN with the use of Metacost achieved the best result with an F-measure of 85.81%.

ACKNOWLEDGEMENT

This research received fund by the Ministry of Higher Education in Malaysia (grant no. FRGS 1/2016/ICT02/UKM/02/11).

REFERENCES

Al-Moslmi, T., M. Albared, A. Al-Shabi, N. Omar and S. Abdullah, 2017a. Arabic senti-lexicon: Constructing publicly available language resources for Arabic Sentiment analysis. J. Inf. Sci., Vol. 1.

Al-Moslmi, T., N. Omar, M. Albared and A. Al-Shabi, 2017b. Feature transfer through new statistical association measure for cross-domain sentiment analysis. J. Eng. Appl. Sci., 12: 164-170.

Al-Moslmi, T., S. Gaber, A. Al-Shabi, M. Albared and N. Omar, 2015. Feature selection methods effects on machine learning approaches in Malay sentiment analysis. Proceedings of the 1st ICRIL-International Conference on Innovation in Science and Technology (IICIST), April 20, 2015, University of Technology Malaysia, Kuala Lumpur, Malaysia, pp: 444-447.

Al-Moslmi, T.A.A., 2014. Machine learning and lexicon-based approach for Arabic sentiment analysis. Master Thesis, National University of Malaysia, Bangi, Malaysia.

Albared, M., T. Al-Moslmi, N. Omar, A. Al-Shabi and F.M. Ba-Alwi, 2016. Probabilistic arabic part of speech tagger with unknown words handling. J. Theor. Appl. Inf. Technol., 90: 236-246.

Allison, B., 2008. Sentiment Detection using Lexically-Based Classifiers. In: Text, Speech and Dialogue, Sojka, P., H. Ales, K. Ivan and P. Karel (Eds.). Springer, Berlin, Germany, pp: 21-28.

Alsaffar, A. and N. Omar, 2014. Study on feature selection and machine learning algorithms for Malay sentiment classification. Proceedings of the International Conference on Information Technology and Multimedia (ICIMU), November 18-20, 2014, IEEE, New York, USA., ISBN:978-1-4799-5423-0, pp: 270-275.

Ba-Alwi, F.M., M. Albared and T. Al-Moslmi, 2017. Choosing the optimal segmentation level for POS tagging of the Quranic Arabic. Br. J. Appl. Sci. Technol., 19: 1-10.

Breiman, L., 1996. Bagging predictors. Mach. Learn., 24: 123-140.

Deng, Z.H., K.H. Luo and H.L. Yu, 2014. A study of supervised term weighting scheme for sentiment analysis. Expert Syst. Appl., 41: 3506-3513.

Domingos, P., 1999. Metacost: A general method for making classifiers cost-sensitive. Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 15-18, 1999, ACM, San Diego, California, ISBN:1-58113-143-7, pp: 155-164.

Ghiassi, M., J. Skinner and D. Zimbra, 2013. Twitter brand sentiment analysis: A hybrid system using N-gram analysis and dynamic artificial neural network. Expert Syst. Appl., 40: 6266-6282.

Hu, X., J. Tang, H. Gao and H. Liu, 2013. Unsupervised sentiment analysis with emotional signals. Proceedings of the 22nd International Conference on World Wide Web, May 13-17, 2013, ACM, New York, USA., ISBN:978-1-4503-2035-1, pp: 607-618.

Isa, N., M. Puteh and R.M.H.R. Kamarudin, 2013. Sentiment classification of Malay newspaper using immune network (SCIN). Proceedings of the World Congress on Engineering Vol. 3, July 3-5, 2013, WEC, London, UK., ISBN:978-988-19252-9-9, pp: 1-6.

Kang, H., S.J. Yoo and D. Han, 2012. Senti-lexicon and improved Naive Bayes algorithms for sentiment analysis of restaurant reviews. Exp. Syst. Appl., 39: 6000-6010.

Khan, F.H., S. Bashir and U. Qamar, 2014. TOM: Twitter opinion mining framework using hybrid classification scheme. Decis. Support Syst., 57: 245-257.

Meir, R. and G. Ratsch, 2003. An Introduction to Boosting and Leveraging. In: Advanced Lectures on Machine Learning, Mendelson, S. and J.S. Alexander (Eds.). Springer, Berlin, Germany, ISBN:978-3-540-00529-2, pp: 118-183.

Moreo, A., M. Romero, J.L. Castro and J.M. Zurita, 2012. Lexicon-based comments-oriented news sentiment analyzer system. Expert Syst. Appl., 39: 9166-9180.

Omar, N., M. Albared, A.A.Q. Shabi and A.T. Moslmi, 2013. Ensemble of classification algorithms for subjectivity and sentiment analysis of arabic customers' reviews. Int. J. Advancements Comput. Technol., 5: 77-85.

Omar, N., M. Albared, A.T. Moslmi and A.A. Shabi, 2014. A Comparative Study of Feature Selection and Machine Learning Algorithms for Arabic Sentiment Classification. In: Information Retrieval Technology, Azizah, J., N.M. Ali, S.A.M. Noah, A.F. Smeaton and P. Bruza et al. (Eds.). Springer, Berlin, Germany, ISBN:978-3-319-12843-6, pp: 429-443.

Samsudin, N., M. Puteh and A.R. Hamdan, 2011. Bess or xbest: Mining the Malaysian online reviews. Proceedings of the 2011 3rd International Conference on Data Mining and Optimization (DMO), June 28-29, 2011, IEEE, Putrajaya, Malaysia, ISBN:978-1-61284-211-0, pp: 38-43.

Sharma, A. and S. Dey, 2012. A comparative study of feature selection and machine learning techniques for sentiment analysis. Proceedings of the 2012 ACM Symposium on Research in Applied Computation, October 23-26, 2012, ACM, New York, USA., ISBN:978-1-4503-1492-3, pp: 1-7.

Ting, K.M., 2002. Cost-Sensitive Classification using Decision Trees, Boosting and MetaCost. In: Heuristics and Optimization for Knowledge Discovery, Sarker, R.A., ?H.A. Abbass and S.N.? Charles (Eds.). Idea Group Inc, Calgary, Alberta, ISBN:9781930708266, pp: 123-290.

Xianghua, F., L. Guo, G. Yanyan and W. Zhiqiang, 2013. Multi-aspect sentiment analysis for Chinese online social reviews based on topic modeling and HowNet lexicon. Knowl. Based Syst., 37: 186-195.