khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Effectiveness of Naïve Bayes Weighted SVM Method in Movie Review Classification

Fadli Fauzi Zain
Informatics Study Program
Universitas Telkom
Bandung
fadlifzain@gmail.com
Yuliant Sibaroni
Informatics Study Program
Universitas Telkom
Bandung
yuliant@telkomuniversity.ac.id

Abstract-Classification of movie review belongs to the domain of text classification, particularly in the field of sentiment analysis. Popular text classification methods for the process include Support Vector Maching (SVM) and Naïve Bayes. Both methods are known to have good performance in handling text classification individually separately. Combination of the two may be expected to improve the classification performance compared to the performance of each individual method. This paper reports an effort to classify movie review using the combined method of SVM with Naïve Bayes as the weighting factor, which is commonly called NBSVM. Our work shows that higher accuracy is obtained when classification is done using NBSVM rather than using individual methods. Accuracy at the level of 88.8% is attained when using the combined feature of unigram and bigram with only data cleansing in the pre-processing stage.
Keywords: movie review, classification, NBSVM, Naïve Bayes, SVM

1. Introduction

a. Background

Text is a common media to deliver review not exceptionally in the case of movie review. Movie review is believed to give influence to consumers and film lovers in deciding whether or not to watch a screen [1]. Movie fans may first read reviews before deciding to watch or not to watch a movie in order to avoid disappointment of seeing under qualified play. The huge number of movies produced drives film lovers to get more selective in deciding which movie to see.

Movie review is beneficial not only to film consumers but also for film producers. People's sentiment towards films can be used by producers to infer which kind of movies people love and which they don't. Such a knowledge is useful for producers to make films that will find large audience and will fulfill the market demand.

People's sentiment towards a thing or an event may be inferred by sentiment analysis against comments or reviews on the topic. In computer science, sentiment analysis may be conducted using classification techniques. Classification in the context of text mining is a method to label texts into one of known categories or classes. In sentimen analysis, the categories may be positive, negative or neutral. Many works have used classification techniques to conduct sentiment analysis of movie reviews [1] - [3]. It is true that manual works are needed in the training stage to label texts for the classification algorithm to proceed. However, the later process of conducting sentiment analysis can be automatically run, eliminating the need to observe texts one at a time.

SVM (support vector machine) is one of text classification methods that is known to have high performance among many other classifiers [4]-[6]. On the other hand, Naïve Bayes is a classifier that is a simple and easy to implement [7]. The latter method in many cases of text classifications shows performance that is almost equal to that of the SVM method [6]. There is an expectation that combining the two methods will produce a better result than running each individual method in text classification.

The two aforementioned methods are combined by giving each a different role in the course of classification process. Classification of movie review proceeds through several preliminary stages including pre-processing, feature extraction, and feature weighting. Pre-processing stage includes raw text cleansing, stop-word removal, and lemmatization. Feature extraction stage processes text to produce n-gram features. Feature weighting stage is carried

out using Naïve Bayes probability model, while the main process of classification is conducted using the SVM method. The approach is known as NBSVM.

b. Topics and Limitations

This research is focused on the use of the Naïve Bayes method for calculating the weight and SVM method for classification, which is called in previous studies as the NBSVM method [8]. The use of the method may supported by initial data processing including stop-word removal and lemmatization [9]. The method is applied to movie review data that has 2 polarities, namely positive and negative. The data are obtained from English movie review of IMDB, which was collected in 2002 and has been used in many studies [10]. The data contain 2000 movie review records, consisting of 1000 positive and 1000 negative reviews.

This study aims at determining the performance of the NBSMV method in the movie review classification process. The baseline is the performance of the classification of movie review using separate individual SVM and Naïve Bayes methods. The NBSVM combination applies when Naïve Bayes is implemented for the weighting process of n-gram feature. Performance is measured based on the accuracy of the classification process. In addition to observing the performance of the classification method, this study also observes the performance of classification process for different pre-processing algorithms.

2. Related Studies

Research on the classification of movie reviews is closely related to the field of text mining that uses text as input data. Text mining or text data mining tries to find knowledge by analysing textual data. The process refers to the way of taking knowledge or information based on a pattern in the text [11].

Text mining first appeared in 1674 and was associated with the name Thomas Hyde for the library catalog process at Oxford University [12]. In 1958 a person named Luhn adapted an IBM 701 computer to produce document abstractions [12]. Research on text mining are still continue at this time to get deeper information discovery.

One branch of knowledge in text mining is text classification. Text classification can be applied in various fields such as topic detection, spam e-mail filtering, web classification and sentiment analysis [13]. Movie review classification in this study falls into the field of sentiment analysis because classification labels are only related to the emotions of the commentators, namely positive, negative or neutral.

Focus of research in text classification includes works such as developing methods in labeling texts, developing pre-processing methods (such as stemming, stop-word removal, and data cleansing), feature extraction, feature weighting methods, feature selection methods and also the invention of new classifier methods.

In the field of general text classification, Uysal and Gunal examined the effect of pre-processing on the performance of text classification. The study was conducted with e-mail data and online news and the languages used were Turkish and English. Their results showed that pre-processing affected performance of the classification of texts and the performance were influenced by domain and language used [13]. Other studies conducted by Dasgupta et al. [14] focused on the feature selection. Their research showed that strategies with provable performance guarantees give better results compared to other feature selection methods. Research to improve classification method has also been carried out, for example using particle swarm optimization which is claimed that it improves the process of identifying retinopathy [15].

In the field of sentiment analysis, particularly in the field of movie review classification, a number of studies have also been carried out. Research on classification of movie review is important because movie review turns out to influence consumers' decision to watch or not to watch a film [1]. However, according to Pentheny, this influence does not apply to all types of human personalities.

Multiple classifier strategy was used by Tsutsumi to classify movie reviews [2]. The results showed that the use of three classifiers with a voting mechanism gave better results than the use of a single classifier. In this observation, the classifiers tested were SVM, ME and score calculation.

Sahu and Ahuja focused more on multilabel classification, namely by classifying the polarity of movie reviews on 4 scales, from values 0 to 4 [3]. The structured N-Gram feature was also observed in this study and it proved to give the best accuracy.

Tripaty et al. in the movie review classification proposed a machine learning approach based on n-gram features [16]. The n-gram combination implemented in the study is for n = 1, 2 and 3. While the classifier tested in this study includes Naïve Bayes (NB), Maximum Entropy (ME), Stochastic Gradient Descent (GDE), and Support Vector Machine (SVM). The result showed that SVM classifier with complete n-gram features (n = 1, 2, 3) gave the best results, reaching a level of 88.94%. The study used the IMDB dataset.

3. Method

Our study begins with retrieving movie review data collected by Pang and Lee from the IMDB website [10]. Pre-processing is performed on the data which is then continued with the n-gram feature extraction process. The n-gram feature is then weighted using the Naïve Bayes probabilistic model. The results then become input for the SVM model during the learning process of the model.

a. Pre-processing

Pre-processing is implemented with the aim of reducing noise in the dataset, thereby it results in

increasing classification performance. In our research, pre-processing includes cleansing, stop-word removal and lemmatization.

Cleansing is conducted by removing symbols and numerical characters from texts in the dataset. The goal is to get rid of terms that have no meaning so that noise can be eliminated, which may reduces classification performance. As an example, the cleansing of phrase "when you see the scene on 13:56, urgh" will result in another phrase "when you see the scene on urgh".

Stop-word removal deletes words that are not considered important and do not add to the meaning of the sentence. Words that come out very often or conjunction are considered to have no meaning. For example, stop-word removal of the phrase "when you see the scene on urgh" will result in the phrase "you see the scene", because the words *when*, *the*, *on*, *urgh* belong to stop-words.

Lemmatization is conducted to return words to their basic form. It is assumed that a word has the same meaning even if it is in different forms. Lemmatization removes affixes to a word. A case of lemmatization is to revert past tense to simple tense, for example from "I wrote the letter" to "I write the letter".

In this research, cleansing is the only pre-processing method that is used in all experiments. On the other hand, stop-word removal and lemmatization become testing variables. We applied cleansing by removing symbols other than the alphabet, and we used NLTK library for stop-word removal and lemmatization.

N-gram Feature Extraction

One of the problems with text mining is the failure to take the combinatorial meaning of words that have different meanings when the words are separated. N-gram feature extraction is an effort to overcome this problem [17].

The n-gram feature extraction helps solve the problem by combining n words into one lexical or term, so that the meaning of a term or phrase like "good morning" can be interpreted better by machine as opposed to the separated words "good" and "morning". This greatly helps the task of Natural Language Processing in interpreting term or lexical units.

N-gram is conducted after pre-processing, so that the combination of terms occurs when the data is clean from noise. In this research, we use several n-gram, namely unigram, bigram, and trigram. Unigram breaks up sentences into one gram per term. For example, unigram feature extraction of the phrase "what do you want to say?" becomes "what", "do", "you", "want", "to", and "say?". The use of bigram feature extraction to the same sentence will produce features "what do", "do you", "you want", "want to", and "to say?". While the trigram feature extraction results "what do you", "do you want", "you want to", and "want to say?".

b. Naïve Bayes weighting

Naïve Bayes is one of the algorithms for classification process. It uses probability and statistical methods. Naïve Bayes algorithm predicts the likelihood of an event to occur by learning information that has been obtained previously. The probability theory involved is called the Bayes Theorem [18].

Naïve Bayes has various advantages. The algorithm is easy to implement because it has low complexity, it does not need too many training data, and it does not require model optimization. Attributes on training data that have independent assumptions are outside the scope of this study. If these conditions are not met, the performance of the Naïve Bayes method will diminish [18].

Naïve Bayes method is a classifier that uses a probabilistic model for the classification process. Probalilistic formula for an attribute X is

$$P(C \mid X) = \frac{P(X \mid C).P(C)}{P(X)} \quad (1)$$

where, P(C|X) is the probability of attribute X to be classified as class C, P(C) is the probability of class C to appear in all training data, while P(X) is the probability that attribute X appears in all training data and P (X|C) is the probability that attribute X to occur in class C.

Naïve Bayes weighting is conducted after the process of feature extraction with n-gram has finished. The process of weighting calculates the probability of occurrence of each term in each class, which produces a matrix containing the weighting value.

As an illustration, suppose we have the following two sentences in training data.

1) *"This movie is great, I like it"* – positive
2) *"I don't like the movie, the villain is too dumb"* – negative

Unigram feature extraction for sentence 1 and 2 (see Table 1) displays the number of unigram occurrences and their polarity. Naïve Bayes weighting calculates the probability of the occurrence of a unigram in a positive or negative class using equation (1) and the results is described in Table 2.

The result of weighting with Naïve Bayes probability makes the matrix data in Table 2 more detailed than the matrix data in Table 1 that does not use Naïve Bayes weighting. More detailed weight values are expected to support the SVM algorithm when building classifier models so that they can classify data better [8].

Table 1. Example of unigram matrix extracted from two sentences as discussed in the text

	dont	dumb	great	I	is	It	like	movie	the	This	too	villain
pos	0	0	1	1	1	1	1	1	0	1	0	0
neg	1	1	0	1	1	0	1	1	2	0	1	1

Table 2 Example of the Naïve Bayes weighting matrix

	dont	dumb	great	I	Is	It	like	movie	the	This	too	villain
pos	0	0	2.315	1.157	1.157	2.315	1.157	1.157	0	2.315	0	0
neg	0.578	0.578	0	1.157	1.157	0	1.157	1.157	0.385	0	0.578	0.578

c. SVM Classification

Support Vector Machine (SVM) can be said to be a semi-eager learner classification algorithm because it requires training. SVM also stores a small portion of the training data for reuse during the prediction process. Some of the data that is still stored is support vector so this method is called Support Vector Machine [19].

The basic idea of SVM is to separate support vectors between classes by creating restrictions on support vectors. The boundary is called a hyperplane. The delimiter is chosen based on the maximum margin (distance between delimiters). These hyperplane borders have different line shapes called kernels. The best known kernel is the linear kernel because it is easy to implement. Equation 2 below is an example of a linear kenel equation.

$$\overline{wx} + b = 0 \tag{2}$$

In the equation, $\overline{w}$ is a weight vector (*weight*), $\overline{x}$ is a vector of the attribute of the dataset, while b is a bias value.

Hyperplane with the selected kernel is then used as a model to predict the class of data. Prediction is obtained by mapping the vector data that is sought and reading the value of the support vector is located in which part of the whole class [19].

SVM has the advantage of being one of the most powerful and accurate methods among common methods and rarely overfitting when the model is right. However, computing from SVM is known to be heavy, because the more training data, the heavier the process of SVM is also [19].

In the sentiment analysis process, the SVM method is applied to the data in the weighted matrix with Naïve Bayes (Table 2). This study uses a linear SVM kernel that utilizes the LinearSVC module from Scikit-Learn with default parameters (C = 1, weight = 0). In this study, SVM parameter optimization has not been done.

d. Classification Performance

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Figure 2. Confusion matrix

The results of the classification by the SVM method are summarized in a format called a confusion matrix. Confusion matrices are commonly used to describe the performance of a classification method whose actual class is known [20]. The form of the confusion matrix for the classification of two classes can be seen in Figure 2.

The confusion matrix entry in Figure 2 is only a number illustration only as an example of calculating the accuracy value. The data in the confusion matrix shows the number of class predictions that correspond to the actual class. Accuracy which is a measure of classification performance is calculated using equation 3.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

TP or True Positive is the number of positive predictive results whose actual class is also positive, while TN or True Negative is the number of negative predictive results whose actual class is also negative. For the example data in Figure 2, the result of the calculation of the accuracy value is 150/165 = 90.9%.

e. Validation

Evaluation of the film review classification process is done using the k-fold cross validation method with k = 10. This evaluation method is common in several studies of text classification [21] - [23]. This method guarantees that the results obtained are more objective, and not obtained by chance because of the good data composition. This method is done by dividing the dataset into 10 parts, where 9 parts are used in the learning process and 1 part is used as testing. The choice of k = 10 is based on the results of previous studies to minimize overfit [8].

4. Results and Discussion

This study attempts to observe the performance of the SVM algorithm for the classification of movie reviews. Research parameters include the use of Naïve Bayes weighting, n-gram feature extraction, and pre-processing treatment. N-gram feature extraction testing is done with several n-gram ranges, namely {(1, 1), (2, 2), (3, 3), (1, 2), (1, 3), (2, 3) }. The range in question is a combination of n-gram features with a range from the initial value to the final value, for example range (1, 3) means the combination of unigram, bigram and trigram.

The results of the classification performance calculation for the Naïve Bayes method, Support Vector Machine and a combination of both (B

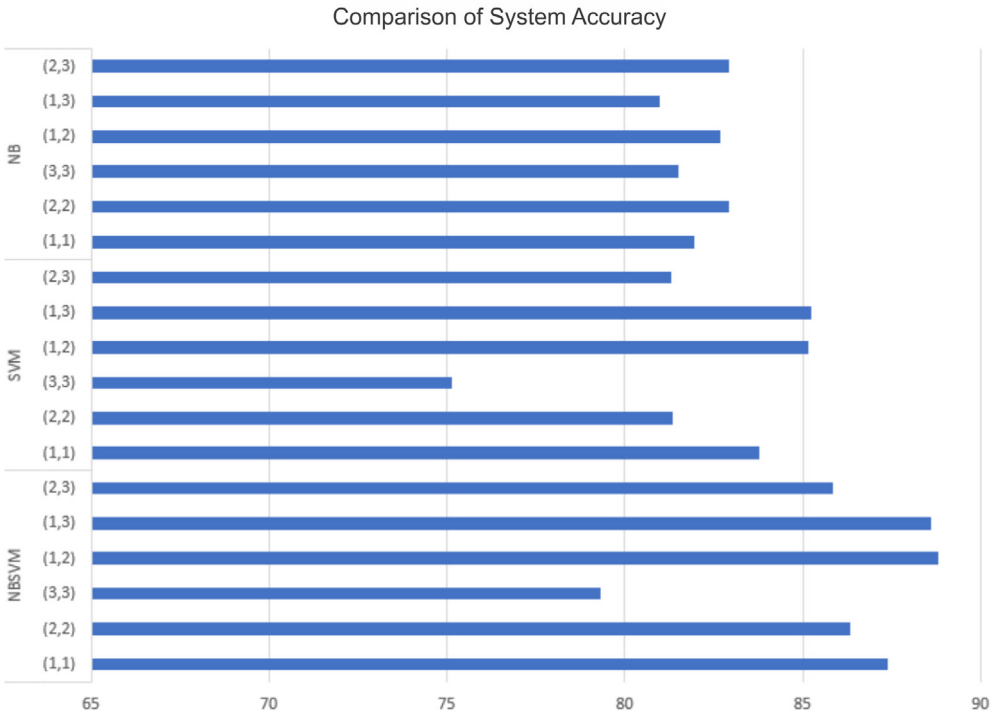

Figure 3. Accuracy of the classification process using the NB, SVM and NBSVM methods for various n-gram ranges

Table 3. Effect of Pre-processing treatment on NBSVM performance

Classifier	Range N-Gram	*Cleansing*	*Cleansing + Stop-word*	*Cleansing + Lemma*	*Cleansing + stop-word + Lemma*
NBSVM	(1,1)	87.4 %	86.6 %	87.1 %	86.4 %
	(1,2)	88.8 %	87.8 %	88.25 %	87.7 %
	(1,3)	88.6 %	87.4 %	88.05 %	86.5 %

NBSVM) is shown in Figure 3 which is presented in the form of a bar chart. There are 18 bars in the diagram, where the top 6 bars are the performance of the Naïve Bayes method, the middle 6 bars are the performance of the SVM method and the rest are the performance of the NBSVM method. Each bar represents a performance value for a different n-gram range as written on the label on the left. This diagram is obtained to treat pre-processing only in the form of data cleansing.

Figure 3 clearly shows that the NBSVM method shows better performance than the other two methods, for almost all n-gram ranges except range (3, 3). For range (3, 3), the Naïve Bayes method shows the best performance. The highest performance is obtained if the NBSVM method with n-gram range (1, 2) gives an accuracy value of 88.8%. This means that the use of the NBSVM method with unigram and bigram feature extraction together provides the highest classification performance.

Further observations were made on the NBSVM method to see the effect of pre-processing and n-gram range. The pre-processing process is varied to see the effect of each subprocess on classification performance. Table 3 shows the classification accuracy values for pre-processing which involve data cleansing only, cleansing with stop word removal, cleansing with lemmatization, and cleansing with both stop word and lemmatization.

Table 3 shows that the movie review classification provides the best performance when data cleansing is only done at the pre-processing stage. The stop word removal and lemmatization process does not improve the accuracy of the classification process. This phenomenon can occur if the deleted stop-word is actually an important word in the context of a movie review. The stop-word list used in this study is derived from the general NLTK module. It is suspected that some stops may need to be maintained and further observation is needed to verify this suspicion.

Table 4. Effect of range on NBSVM performance

Classifier	N-Gram	Accuracy
NBSVM	(1, 1)	87.4
	(1, 2)	88.8
	(1, 3)	88.6

The lemmatization process turns down the classification performance. The reason is probably the inadequate performance of the lemmatization method. The lemmatization process might produce words without affixes that have a different meaning than when the prefixes still exist. However, the change in accuracy due to

the stop word removal and lemmatization process is not too significant so it is recommended to use the simplest process, namely pre-processing with data cleansing only.

Subsequent observations were made on the effect of the n-gram range made for the NBSVM method. The results are shown in Table 4. The best accuracy is obtained when feature extraction is done by combining unigram and bigram, that is in the range (1, 2). The addition of the tri-gram extraction feature in our observations did not improve the performance of the movie review classification process. While the use of the unigram feature alone gives a relatively smaller performance. For this reason, it is recommended to use range (1, 2) in the film review classification process with the NBSVM method.

5. Conclusion

Based on the description in the Results and Discussion section it can be concluded that the best performance for movie review classification is obtained when the NBSVM method is used, which gives accuracy at a value of 88.8%. The method combines SVM method for text classification and the Naïve Bayes for weighting the n-gram extraction. The use of SVM and Naïve Bayes methods separately gives significantly lower accuracy.

The use of mere data cleansing at the pre-processing stage turns out to provide the best classification results. Classification performance does not improve when we included stop-word removal that cleaned data from unnecessary terms, nor when lemmatization which picked the basic form of words in the text. Classification performance is influenced by the addition of bigrams in the feature extraction process, but not affected by further addition of trigrams. Therefore, the authors recommend the use of unigram and bigram together during the feature extraction process.

References

[1] J. R. Pentheny, "The Influence of Movie Reviews on Consumers," University of New Hampshire, 2015.

[2] K. Tsutsumi, K. Shimada, and T. Endo, "Movie Review Classification Based on a Multiple Classifier *," *Proc. 21st pacific Asia Conf. Lang. Inf. Comput.*, no. 2007, pp. 481–488, 2007.

[3] T. P. Sahu and S. Ahuja, "Sentiment analysis of movie reviews: A study on feature selection & classification algorithms," in *2016 International Conference on Microelectronics, Computing and Communications (MicroCom)*, pp. 1–6.

[4] S. K. Saritha, "Methods for Identifying Comparative Sentences," *Comput. Appl.*, vol. 108, no. 19, pp. 23–26, 2014.

[5] P. Das and S. Sharma, "An Entropy Based Effective Algorithm for Data Discretization," vol. 4, no. 3, 2017.

[6] H. Houngbo and R. E. Mercer, "An automated method to build a corpus of rhetorically-classified sentences in biomedical texts," in *Proceedings ofthe First Workshop on Argumentation Mining*, 2014, pp. 19–23.

[7] A. M. F. Al Sbou, A. Hussein, B. Talal, and R. A. Rashid, "A Survey of Arabic Text Classification Models," vol. 8, no. 6, pp. 4352–4355, 2018.

[8] S. Wang and C. Manning, "Baselines and Bigrams: Simple, Good Sentiment and Topic Classification," *Proc. 50th Annu. Meet. Assoc. Comput. Linguist.*, vol. 94305, no. July, pp. 90–94, 2012.

[9] S. Vijayarani, J. Ilamathi, and M. Nithya, "Preprocessing Techniques for Text Mining - An Overview," *Int. J. Comput. Sci. Commun. Networks*, vol. 5, no. 1, pp. 7–16, 2015.

[10] P. Bo and L. Lee, "Movie Review Data," 2004. .

[11] A.-H. Tan, "Text Mining: The state of the art and the challenges," *Proc. PAKDD 1999 Work. Knowl. Disocovery from Adv. Databases*, vol. 8, pp. 65–70, 1999.

[12] G. Miner, J. Elder, T. Hill, R. Nisbet, D. Delen, and A. Fast, *Practical Text Mining and Statistical Analysis for Non - Structured Text Data Applications*. Waltham: Elsevier, 2012.

[13] A. K. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Inf. Process. Manag.*, vol. 50, no. 1, pp. 104–112, 2014.

[14] A. Dasgupta, P. Drineas, B. Harb, V. Josifovski, and M. W. Mahoney, "Feature Selection Methods for Text Classification," *KDD*, pp. 230–239, 2007.

[15] T. Arifin and A. Herliana, "Optimasi Metode Klasifikasi Dengan Menggunakan Particle Swarm Optimization Untuk Identifikasi Penyakit Diabetes Retinopathy," *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 4, no. 2, pp. 77–81, 2018.

[16] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach _ Elsevier Enhanced Reader.pdf," *Expert Syst. with Appl.*, pp. 117–126, 2016.

[17] Y. Heights, "Class-Based n-gram Models of Natural Language Iwl)'" Pr (Wk Iw - -l). Wk," *Comput. Linguist.*, no. 1950, 1992.

[18] I. Rish, "An empirical study of the Naïve Bayes classifier," *Empir. methods Artif. Intell. Work. IJCAI*, vol. 22230, no. JANUARY 2001, pp. 41–46, 2001.

[19] H. Bhavsar and A. Ganatra, "A Comparative Study of Training Algorithms for Supervised Machine Learning," *Int. J. Soft Comput. Eng.*, vol. 2, no. 4, pp. 74–81, 2012.

[20] K. Markham, "Simple guide to confusion matrix terminology," *Data School*, 2014. .

[21] Kuspriyanto, O. S. Santoso, D. H. Widyantoro, H. S. Sastramihardja, K. Muludi, and S. Maimunah, "Performance Evaluation of SVM-Based Information Extraction using τ Margin Values," *Int. J. Electr. Eng. Informatics* -, vol. 2, no. 4, pp. 256–265, 2010.

[22] S. Teufel, A. Siddharthan, and D. Tidhar, "Automatic classification of citation function," *Proc. EMNLP-06, Sydney, Aust.*, 2006.

[23] S. Teufel and A. Athar, "Detection of Implicit Citations for Sentiment Detection," *Proc. ACL-12 Work. Discov. Struct. Sch. Discourse, Jeju Island, South Korea, 2012*, no. July, pp. 18–26, 2012.