Efficient Spam Detection using Single Hidden Layer Feed Forward Neural Network

Dr. S. K. Jayanthi[1], V. Subhashini[2]

[1] *Head and Associate Professor, Dept. of Computer Science, Vellalar College for Women, Erode, Tamilnadu, India*
[2] *Research Scholar, Department of Computer Science, Vellalar College for Women, Erode, Tamilnadu, India*

---***---

Abstract - ***Recent development in social media, which is one of the crucial communication tools owning to wide spreading of internet technologies. On social network sites, spammers frequently cover themselves by creating fake accounts and hijacking normal user account for personal gain. In today's era everybody is in online and use social network sites for interaction to gain knowledge for business purpose, studies, politics and many more. But along with positive approaches, it increases the spammers who continuously expose malicious behavior, which leads to great misunderstanding and inconvenience on users social activities. Apart from email, SMS, Links, spammers in social media behave like normal users and they continue to change their spamming strategies to fool non-spam systems. This kind of spam can contribute to degrade the quality of real time search engines unless mechanisms to fight and stop spammers.***

In this Paper, spam in the Sina Weibo website is detected. Based on it many researchers had detected spam but still cannot achieve that level of accuracy in spam detection. So to gain the greater level of accuracy, this work proposes Single hidden Layer Feed forward Neural network concept to detect spam which are spread by unauthorized users or by spammers. And this is analysed by feature extraction and by applying classifier. Moreover the three classifier algorithms Naive Bayes (NB), Decision tree (C4.5) and Support Vector Machine (SVM), are applied to show the comparative results. Performance evaluation is done based on the metrics Precision, Recall, F-measure, and demonstrated that the proposed method detect spam at the accuracy rate of about 92.49% in terms of F-measure value.

Key Words: *Naive Bayes (NB), C4.5, Support Vector Machine (SVM), Single hidden Layer Feed forward Neural network (SLFN)*

1. INTRODUCTION

Data mining, a recently emerging field, provides an expansive set of techniques to detect useful knowledge from enormous datasets like rules, trends and patterns. Classification is one of the major tasks in data mining. It is a predictive process where prediction about data is made using known results. Classification algorithms can be categorized into 5 groups: Statistical-based, Distance-based, Decision tree-based, Neural network based, and Rule-based. Each of these five categories consists of many algorithms. However, only the most popular algorithms namely Naive Bayes, C4.5, Support Vector Machine and Single hidden Layer Feed Forward Neural Network are used to detect spam.

Internet has become an indispensable tool for communication, because of its fast speed and low cost. Social media systems heavily depend on users for content contribution and sharing. Information will spread across social networks quickly and effectively. However, at the same time social media networks become susceptible to different types of unwanted and malicious spammer or hacker actions. There is a crucial need in the society and industry for security solution in social media. Hence to detect spammers efficiently Single hidden Layer Feed Forward Neural Network (SLFN) algorithm is used in the current work. The microblogPCU dataset is utilized to detect spam which is available in the UCI Machine Learning repository. Feature selection methods, Information gain and Chi-square are used to select a subset of relevant features. The selected features are given as input for the Single hidden Layer feed Forward Neural Network algorithm to classify the spam messages. The SLFN have N hidden nodes, for each input file multiple weight values are added, until the class labels get predicted correctly the weight values are adjusted. So by using SLFN algorithm all files will be classified appropriately. The binary activation function is applied for the weighted sum of all input. The spammers are detected accurately in the proposed work.

This paper is organized as follows: In Section 2, an outlook of the existing research that is significant to the spam classification is carried out. Section 3, describes the process of the spam detection using Single hidden Layer Feed forward Neural network algorithm. Section 4 discusses in detail about the results of both existing and proposed system. Section 4 describes the evaluation metrics namely precision, recall and F-measure which is used to know about the accuracy of the spam detection.

Section 5, finally concludes the work and also discusses about the limitation and future work.

2. LITERATURE REVIEW

Sina Weibo application is similar to Twitter, where users post messages, interact with friends, talk about news and share interesting topics via social network services. Detecting spam in the Sina Weibo has become more challenging task. Spam has been observed in various applications, including e-mail, Web search engines, blogs, videos, etc. Consequently, a number of spam detection and combating strategies have been proposed. Particularly, there have been a considerable number of efforts that rely on machine learning to detect spam. To better understand spam detection, it is beneficial to review and examine the existing systems. Hence, recent approaches and methodologies in the area of spam detection have been discussed. Naive Bayes (NB), Decision tree (C4.5), Support Vector Machine (SVM) algorithms have been considered as the preliminary approaches in this work.

Yang Song, et al (2009) describes about spam detection in Email. The problem is identified in terms of lower classification performance of spam detection. To detect spam efficiently, Naive Bayes algorithm have been proposed here to solve the problem. Based on the conditional probability function, it classifies the maximum number of spam from the dataset. It takes less computation time and produced high precision values for the specified spam dataset. But its classification accuracy decreases when the attributes are not independent.

Hythem Hashim, et al (2015) build a classification model that can be used to improve the student's academic records in Faculty of Mathematical Science and Statistics. Based on the C4.5 classification algorithm, the decision tree is constructed, depending on the most affective attributes. Recursion and repetition upon attribute selecting and set splitting will fulfill the construction of decision tree root node and internal node. After building the decision tree, improper branches are pruned. This algorithm is not suitable for handling large data sets.

Rajesh Wadhvani, et al (2010) describes about the traditional anti-spam techniques like Black and White List. Their goal of Spam Classification is to distinguish between spam and legitimate mail message. But with the popularization of the Internet, it is challenging to develop spam filters that can effectively eliminate the increasing volumes of unwanted mails automatically before they enter a user's mailbox. In this paper, the authors evaluate the performance of Non Linear SVM based classifiers with various kernel functions over Enron Dataset. The main disadvantage of using SVM is that it can't able to handle large dataset. The spam classification accuracy is low.

2.1. Naive Bayes Classifier

Binary classifier is constructed based on Bayes theorem of probability. It is binary because it concludes whether a record belongs to spam or non spam. Let us consider two classes of documents c and c'. The class c is spam and class c' is non spam. Once the dataset is trained, it is used to classify the testing dataset into one of the two classes: spam or non spam. The dataset contains N number of attributes. The dataset x is represented as X= ($x_1, x_2, \ldots,$ x_n), c, c' where c is the spam document, c' is the non spam document. The probability that the document belongs to class c (spam) or class c' (non spam) is computed by using, P (C = c | $X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n$) = probability that the dataset belongs to class c. P (C = c' | $X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n$) = probability that the dataset belongs to class c'. Using Bayes rule we can rewrite the following as,

$$P(C = c \mid X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \frac{p(X_1=x_1, X_2=x_2, \ldots\ldots, X_i=x_i \mid C=c)\, p\,(C=c)}{p(X_1=x_1\,, X_2=x_2, \ldots\ldots, X_i=x_i)}$$

$$P(C = c' \mid X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \frac{p\left(X_1=x_1, X_2=x_2, \ldots\ldots, X_i=x_i \mid C=c'\right) p(C=c')}{p(X_1=x_1\,, X_2=x_2, \ldots\ldots, X_i=x_i)}$$

The probability value is calculated. Based on the probability value the classification is performed.

2.2. C4.5 Algorithm

Decision tree modeling is one of the classifying and predicting data mining techniques, belonging to inductive learning and supervised knowledge mining. It is a tree-diagram-based method, depending on two manners; the node on the top of its tree structure is a root node, and node in the bottom is leaf node. Each leaf node will have target class attribute. The training data set can give an improper branch on decision tree building; this is usually denoted by the term of over-fitting. Therefore, after building the decision tree, it has to be pruned to remove improper branches, so as to enhance decision tree model accuracy in predicting new data. C4.5 follows a "divide-and-conquer" strategy to build a decision tree through recursive partitioning of a training dataset. By using the C4.5 classifier, the Tree is formed by computing the Class Frequency. If there is only one Class means the Leaf node is returned. For each attribute, Information Gain is computed. The attribute with the highest information gain is selected as the root node. The decision node has S children, $T_1, \ldots,$ Ts are the sets of the splitting produced by the test on the selected attribute. If Ti is empty, the child node is directly set to be a leaf node. If Ti is not empty, the divide and conquer approach is recursively applied for all the attribute.

2.3. Support Vector Machine (SVM)

Support Vector Machines (SVMs) are the supervised learning methods used for classification and regression. The Support Vector Machine constructs a set of hyper planes in high-dimensional space, which can be used for classification, regression or other tasks. The goal of a SVM is to find the optimal separating hyper plane which maximizes the margin of the training data. SVM is a classification algorithm. The training data set is trained by using libSVM tool available on Weka, data mining software upon Java tool. Given the dataset X=((x1,y1), ... , (xn, yn) , C) ,x and y are samples, c – labeled class (spam or non- spam). The testing dataset T = ((t1, t2, . . , ti) , C). For each $t_i \in X$ (t_i is a vector containing features), t_i is classified using the function f(t). f (t) = t_i.w + c (w- weight vector, c- bias). If the testing dataset t_i matched with the training dataset X_i, the class label is predicted, else the xi is pruned. The process is continued until all the records get classified.

In case of Naive Bayes classifier, dependencies exist among variables. So the classification is not done accurately. Hence there is a loss in accuracy. In case of C4.5 algorithm, small variation in data can lead to different decision trees (especially when the variables are close to each other in value). It does not work well on a small training set. In case of Support Vector Machine algorithm, the weight values are added to the attributes only once, if the dataset doesn't match, it gets pruned. So the classification is not done accurately. Its takes long time to train the training dataset. Hence to improve the spam classification accuracy, the SLFN algorithm is used in the proposed work.

3. SYSTEM METHODOLOGY

The advancing internet technologies rapidly increase the spamming activities. Current work is based on detecting spammers in Sina Weibo Social network site. The Single hidden Layer Feed forward Neural network algorithm is used to detect spammers. The proposed work consists of five phases of spam detection in the Sina Weibo.

- Dataset Collection
- Preprocessing
- Feature Selection
- Single hidden Layer Feed Forward Neural Network algorithm
- Performance Evaluation

3.1 Dataset Collection

The UCI Machine Learning Repository is a collection of datasets, domain theories and data generators that are used by the machine learning community for the empirical analysis of machine learning algorithms. In order to evaluate and compare the existing and proposed work, the needed dataset is obtained from the UCI Machine Learning Repository named as microblogPCU Data Set. This dataset is downloaded from the website http://archive.ics.uci.edu/ml/datasets/microblogPCU .

Description of the dataset

The listed attributes are taken from the microblogPCU dataset, Number of attributes: 20. The attributes are User _id, User _name, Gender, Class, Message, Post _num, Follower _num, Followee _num, Scratch _time, is_ spammer, Post _id, Post _time, Poster _id, Repost _num, Comment _num, Follower, Follower _id, Followee, Followee _id, Content. The description of each feature is given below,

- User _id: account ID in Sina Weibo,
- User _name: account nickname,
- Gender: account registration gender,
- Class: account level given by Sina Weibo,
- Message: account registration location or other personal information,
- Post _num: the number of posts of this account up to now,
- Follower _num: the number of followers of this account,
- Followee _num: the number of followee of this account,
- Scratch _time: account created time,
- is_spammer: manually annotated label, 0 means spammer and 1 means non-spammer,
- Post _id: user post ID given by Sina Weibo,
- Post _time: the time when a post is posted,
- Poster _id: the user ID who posted this post,
- Repost _num: the number of retweet by others,
- Comment _num: the number of comment by others,
- Follower: the nickname of follower;,
- Follower _id: the user ID of follower,
- Followee: the nickname of followee,
- Followee _id: the user ID of followee,
- Content: the post text.

The data in microblogPCU Data Set is collected from the Sina Weibo website. Dataset may contain incomplete, noisy and inconsistent data, so preprocessing is done in the next step.

3.2 Preprocessing

Classifier algorithm cannot be directly interpreted the text, the preprocessing is needed. The preprocessing step is used to reduce the size of the specified dataset and to improve the classification results. Real world data are generally incomplete (lacking attributes values), noisy (containing errors or outliers) and inconsistent. Data must be preprocessed in order to perform any data mining functionality. The preprocessing has been performed on the specified dataset. The preprocessing step includes, stop word removal and stemming. Stop word removal eliminates the words which provide less or no information to the text analysis. Words like articles, prepositions, conjunctions, common verbs (e.g. 'know', 'see', 'do', 'be'), auxiliary verbs, adjectives (e.g. 'big', 'late', 'high'), and pronouns are removed, leaving only content words likely to have some meaning. The words are passed through a stemmer which reduces multiple instances of a single word to the root word. E.g. flying and flied are reduced to fly. Stemming is the process where the words suffixes are removed.

After preprocessing, the incomplete, noisy and inconsistent data are removed. Then in the next step, from the preprocessed dataset, the relevant features are selected by using the feature selection process.

3.3 Feature Selection

From the preprocessed dataset the feature selection process is done. Feature Selection is basically selection of a subset of features from a larger pool of available features. The goal is to improve the prediction performance of the predictors. This is a crucial step in the design of any classification system, as a poor choice of features drives the classifier to perform badly. The two feature selection methods such as Information gain and Chi Square (available on Weka) are applied to find the ranking of importance of the attributes. Information gain is measure of the reduction in entropy of the class variable after the value for the feature is observed. Entropy is simply the average amount of information from the event. For each dataset, the subset of features with non-zero information gain value is selected. The Chi Square test also referred to as χ^2. Chi square test is a statistical method assessing the goodness of fit between a set of observed values and those expected theoretically.

Based on these two (Information gain and Chi square) feature selection measure, the relevant features are selected. After feature selection process, in the next step the selected features are given as input to the classifier algorithm (SLFN) to detect spam.

3.4 Single hidden layer Feed forward Neural network algorithm

The Single hidden Layer Feed Forward Neural Network (SLFN) which randomly chooses the input weights and analytically determines the output weights of SLFNs. This algorithm tends to provide the best generalization performance at extremely fast learning speed. The experimental results based on real-world benchmarking function approximation and classification problems including large complex applications show that the SLFN algorithm can produce best generalization performance. A key feature of neural networks is an iterative learning process in which records (rows) are presented to the network one at a time, and the weights associated with the input values are adjusted each time. After all cases are presented, the process is often repeated. During this learning phase, the network is trained by adjusting the weights to predict the correct class label of input samples. Advantages of neural networks include their high tolerance to noisy data. Neural network is a set of connected input/output units, where each connection has a weight associated with it. Neural network learning is also called connectionist learning due to the connections between units. It is a case of supervised, inductive or classification learning. The labeled dataset is divided into two sets, one is training set on the basis of which the training process is done and second is the testing set on the basis of which the performance of the system is observed that how exactly the results are. To classify the dataset the SLFN algorithm is utilized. The algorithm is first trained on the labeled data to develop classification models then that are applied to unlabelled data to predict the spam messages. Fig. 1. illustrates the basic concept of proposed spammer detection model.

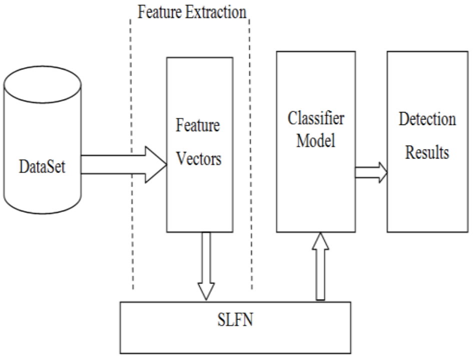

Fig. 1. Overview of Spammer detection model

In this solution, training data is converted into a series of feature vectors that consist of set values for attributes. These vectors construct the input of SLFN algorithm. After training, a classification model is applied to distinguish whether the specific user belong to normal user or spammer. Because spammers and non-spammers have different social behavior, through analyzing feature, it is capable to distinguish abnormal behavior from legitimate ones. The following process is done in the SLFN algorithm to detect spam.

SLFN ALGORITHM

Input: Consider the dataset as an input which holds the attribute X= {(xi, t_i), $x_i \in R^n$, ti $\in R^m$, i= 1,…. N }

Output: Class 0 (spam) or 1 (non spam)

1. Initialize the input vector X_i ($X_i = x_1, x_2, \ldots, x_n$)
2. Randomly allocate the weight values w_j for each input vector X_i, where j = 1,2,..m.
3. Compute the weighted sum of input,
$$U = \sum_{i=1}^{n}\sum_{j=1}^{m} x_i w_{ij}$$
4. For each node of hidden layer, the Binary activation function is applied over weighted sum of its all inputs.
5. Binary activation function is applied using the equation
$$f(u) = \begin{cases} 0 \; for \; u < 0 \\ 1 \; for \; u \geq 0 \end{cases}$$
6. The output Z is computed by
$$Z = f\left(\sum_{i=1}^{n}\sum_{j=1}^{m} x_i w_{ij}\right)$$
if Z= 0 it is spam else if Z= 1 it is non spam
7. The Training data set is trained with the adjusting weight.
8. If the match Exist the class label is predicted
9. Else Repeat Step 2 to 6 until the match exist
10. Return result.

In the Single hidden Layer Feed Forward Neural Network each element of neural network is a node called unit. Units are connected by links. Each link has a numeric weight. In the Fig. 2. the architecture of the SLFN algorithm is given. The input layer consist of n number of nodes Xi= ($x_1, x_2, \ldots, x_n$). The hidden layers consist of n number of hidden nodes with random weight values and in the Output layer the class label is predicted.

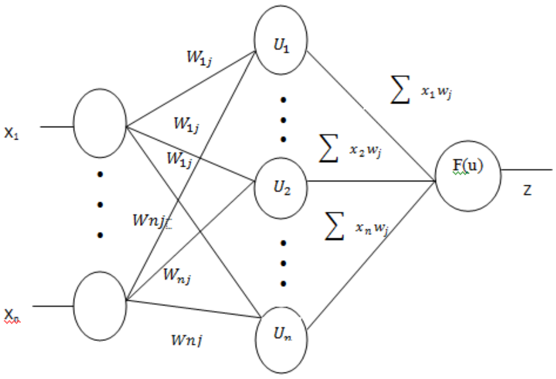

Fig. 2. Architecture of SLFN

In the collection of dataset, x_i is given as the input to the input layer. Each input is given to the hidden layer U with respective weight values. The weight of the input node x_i is given as W_{ij}, where $W_{ij} = 0 \leq W_{ij} \leq 1$. The input weight matrix W can be expressed as:

$$W = \begin{array}{cccc} W_{11} & W_{12} \ldots. & W_{1j} \ldots & W_{1m} \\ W_{21} & W_{22} \ldots. & W_{2j} \ldots & W_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ W_{n1} & W_{n2} \ldots. & W_{nj} \ldots & W_{nm} \end{array}$$

This matrix is of n × m dimension corresponding to n neurons in the input layer and w as the adjusting weight value for each input. The first input record will be checked with the weight value w_{11} , if the match exist between the records, the class label is predicted else the weight value is adjusted from W_{12} to W_{1m} until the match exist. The input with the corresponding weight values are given as the input to the hidden layer. For each input the weighted sum of the input is calculated by using,

$$U = \sum_{i=1}^{n}\sum_{j=1}^{m} x_i w_{ij}$$

Where x_i is the input and W_{ij} is the corresponding weight values. After calculating the weight values in the hidden layer U, the Binary activation function f(u) is applied in the Output layer.

$$Z = f\left(\sum_{i=1}^{n}\sum_{j=1}^{m} x_i w_{ij}\right)$$

If Z > 0, the class label is predicted as spam else it is non spam. The process is repeated for all the inputs. All the messages get classified since the weight values are randomly adjusted based on the classification. Therefore, the proposed algorithm tends to have good classification performance for Single hidden Layer Feed Forward Neural Network.

4. RESULTS AND DISCUSSION

Based on the exiting work given in Section 2 and on the proposed work in Section 3, the results are discussed in this section through experimental analysis.

4.1 Experimental Analysis

The Process of existing and proposed work contains the following steps:

Step 1: The microblogPCU dataset is downloaded from the UCI Machine Learning Repository and stored in the SQL server 2005 database.

Step 2: The records are trained with the respective Naive bayes, C4.5 and Support Vector Machine and Single hidden Layer Feed forward Neural network algorithms.

Step 3: The classification is done for the both existing and proposed algorithms, Naive Bayes, C4.5, Support Vector Machine and Single hidden Layer Feed forward Neural network by using the training and testing dataset.

Step 4: Finally Precision, Recall and F-measure values are calculated to compute the accuracy.

4.2 Performance metrics

To measure the accuracy in classification, the Precision, Recall and F-measures values are calculated.

- TP (True Positive) represents the number of spammers correctly classified,
- FN (False Negative) refers to the number of spammers misclassified as non-spammers,
- FP (False Positive) expresses the number of non-spammers misclassified as spammers
- TN (True Negative) is the number of non-spammers correctly classified.

Precision

Precision (P) is the ratio of number of instances correctly classified to the total number of instances and is expressed by formula

$$P=\frac{TP}{(TP+FP)}$$

Recall

Recall (R) is the ratio of the number of instances correctly classified to the total number of predicted instances and is expressed by formula

$$R=\frac{TP}{(TP+FN)}$$

F-measure

F-measure is the harmonic mean between precision and recall, and is defined as

$$F=\frac{2(P\,.\,R)}{P+R}$$

Confusion matrix is evaluated to make decision that can be made by classifier. We consider a confusion matrix illustrated in Table I,

TABLE I CONFUSION MATRIX

		Predicted	
		Spammer	Non-spammer
True	Spammers	TP	FN
	Non-spammer	FP	TN

Based on the True records and Predicted records the TP, TN, FP, FN values are given in the Table II.

TABLE II PREDICTED VALUES FOR EXISTING AND PROPOSED WORK

INFO GAIN				
Classifier	TP	FP	TN	FN
SLFN	235	29	204	32
SVM	210	66	154	70
C4.5	197	93	113	97
NB	155	108	142	95
CHI SQUARE				
SLFN	250	19	209	22
SVM	212	47	198	43
C 4.5	217	84	118	81
NB	161	92	160	87

4.3 COMPARISON RESULT

The Table III describes the comparison of spam classification in existing and proposed system. The table contains Precision, Recall and F-measure values for the existing and proposed system.

TABLE III COMPARISON RESULT OF EXISTING AND PROPOSED WORK

INFO GAIN			
Classifier	precision	recall	f-measure
SLFN	89.0%	88.0%	88.49%
SVM	76.0 %	75.0%	75.49%
C4.5	68.0%	67.0%	67.49%
NB	59.0 %	62.0%	60.46%
CHI SQUARE			
SLFN	93.0%	92.0%	92.49%
SVM	82.0%	83.0%	82.49%
C4.5	72.0%	73.0%	72.49%
NB	64.0%	65.0%	64.49%

The proposed SLFN algorithm has higher F-measure values when compared to the existing NB, C4.5 and SVM algorithm. Through the Comparison result, it is shown that the proposed SLFN solution is capable to achieve best accuracy. The Fig. 3, Shows the spam classification accuracy result for the NB, C4.5, SVM and SLFN classifier.

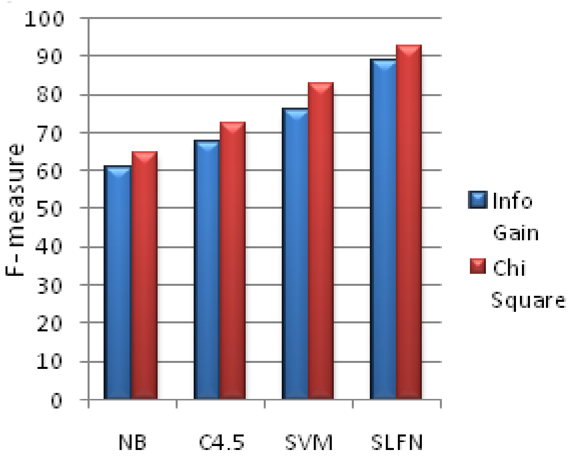

Fig. 3. Accuracy Comparison

5. CONCLUSION AND FUTURE WORK

Sina Weibo is the top microblogging network in China, its popularity makes it being a great platform to do marketing. However the problem is the spam messages due to malicious users. A key feature of Single hidden Layer Feed forward Neural network is an iterative learning process in which records are presented to the network one at a time, and the weights associated with the input values are adjusted each time. Hence the classification of the spam message is done appropriately. It is capable to handle the complex dataset and also produced high specification results. From the performance evaluation of the F-measure values it is concluded that the SLFN algorithm could detect spammers efficiently at the accuracy rate of about 92.49 %. Through the multitude of analysis and evaluation, it is concluded that the proposed solution is feasible and is capable to reach much better classification result than the other existing approaches. The work can be extended in the following direction. Selection of features can be automated. In the era of big data with huge data volume the artificial intelligence technology can be used. Another issue includes Online Spammer detection.

REFERENCES

[1] Xianghan Zheng , Zhipeng Zeng , Zheyi Chen, Yuan long Yu ,Chumming Rong, [2015], "Detecting spammers on social networks", Neurocomputing 159 (www.elsevier.com/locate/neucom), PP 27-34.

[2] Guang-Bin Huang, Lei Chen, and Chee-Kheong Siew, [2006], "Universal approximation using Incremental Constructive Feed forward Networks with random hidden nodes", IEEE Transactions on Neural Networks, Vol. 17, No. 4.

[3] Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, [2006], "Extreme learning machine: Theory and applications", Neuro computing 70 (www.elsevier.com/ locate/ neucom), PP 489–501.

[4] Hythem Hashim, A.Ahmed, Ali Satty and A. Samani, [2015] "Data mining methodologies to study student's academic performance using the C4.5 algorithm", International Journal on Computational Sciences & Applications (IJCSA) Vol.5, No.2.

[5] Yong Xu, Yi Zhou, Kai Chen, [2013] "Observation on Spammers in Sina Weibo", Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering.

[6] Priyanka Chhabra, Rajesh Wadhvani, Sanyam Shukla, [2010] "Spam Filtering using Support Vector Machine", Special Issue of IJCCT Vol.1 Issue 2, 3, 4.

[7] Fabricio Benevenuto, Gabriel Mango, Tiago Rodrigues and Virgilio Almeida, [2010] " Detecting Spammers on Twitter" , Seventh annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference.

[8] Yin Zhu. Xiao Wang, Erheng Zhong, Nanthan N. Liu, He Li, Qiang Yang, [2012], "Discovering spammers in social Networks", Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence.

[9] Yang Song, Aleksander Kołcz and C. Lee Giles, [2009] "Better Naive Bayes classification for high-precision spam detection", Wiley InterScience (www.interscience.wiley.com), PP 1003-1024.