Automatic Detection of Suspicious Bangla Text Using Logistic Regression

Omar Sharif and Mohammed Moshiul Hoque(✉)

Department of Computer Science and Engineering,
Chittagong University of Engineering and Technology,
Chittagong 4349, Bangladesh
{omar.sharif,moshiul_240}@cuet.ac.bd

Abstract. Suspicious Bangla text detection is a text classification problem of determining Bangla texts into suspicious and non suspicious categories. In this paper, we have proposed a machine learning based system that can classify Bangla texts into suspicious and non-suspicious. For this purpose, a corpus is developed and logistic regression algorithm is used for classification task. In order to measure the effectiveness of the proposed system a comparison of accuracy among other algorithms such as Naive Bayes, SVM, KNN, and decision tree also performed. The experimental result with 1500 training documents and 500 testing documents shows that the logistic regression provides the highest accuracy (92%) than other algorithms.

Keywords: Natural language processing · Suspicious Bangla text · Text classification · Machine learning · Regression

1 Introduction

Classification of text is the task of assigning a text document into a set of predefined classes in an intelligent manner. Text classification has become more important as well as more challenging because of rapid growth in online contents in recent years. The procedure of analyzing information has been changed due to digitization and different AI techniques. We observed an exponential increase in availability of online contents. From news portals to social media, ebooks to science journals, web pages to emails all are full of textual data. For classifying, searching, organizing and concisely representing a large amount of information classification of text is required which performs a significant role in various applications. Detecting suspicious text is typically a text classification problem where we have to classify a text into suspicious and non suspicious categories. Suspicious text detection is a kind of system where suspicious texts are identified by the keywords used in the text body. As most of our communications are text based, if we are able to predict either a text is suspicious or not suspicious it may be very helpful for our law enforcement/security agencies to find the perpetrators and may takes prior measure to stop terrorist events. As far we know,

P. Vasant et al. (Eds.): ICO 2019, AISC 1072, pp. 581–590, 2020.
https://doi.org/10.1007/978-3-030-33585-4_57

there is no such system has been developed yet for detecting suspicious Bangla text. But such system is required to predict criminal activities, reduce virtual social harassment in social media, mitigate national privacy threats and overall ensure our national security by detecting suspicious communications.

The prime concern of this work is to develop a framework for detecting suspicious Bangla texts. In this work, we propose a machine learning techniques using logistic regression for classifying the texts into suspicious and non-suspicious categories based on our developed corpus.

2 Related Work

A number of researches have been done in text classification in English and other European languages. Most of these are email classification, research paper categorization, detecting suspicious profiles etc. However, there is no significant research has been conducted yet in Bangla text classification. Hossain et al. describes categorization of Bengali document based on word embedding which uses statistical learning techniques [14]. It categorizes document into nine predefined categories with mentionable accuracy. A text categorization system of Arabic language is build using Naive bayes with good accuracy [10]. Krendzelak et al. proposed a system which categorize text using hierarchical structures and machine learning accompanied with naive bayesian categorization process [12,18]. It performs with low accuracy due to the methods used in training and feature extraction techniques adopted during training. Alami et al. describes about different techniques for detection of suspicious profiles within social media by analysis of text [9]. A system for detecting suspicious email using enhanced feature selection is proposed but it has low accuracy because of not having enough dataset [19]. A system that categorize Turkish test is developed using support vector machine which achieved better accuracy but due to large feature dimensions time complexity is high [17]. Better result can be obtained by using clustering based approach but a lot of problem subsist with cluster-based solution [8,15]. In this work, we proposed a logistic regression technique to classifying suspicious texts in which is trained by own developed dataset. The proposed system is compare to other techniques such as Naive bayes classifier [16], SVM [21], KNN [13] and decision tree [11].

3 Proposed Suspicious Text Detector

The key objective of our work is to design a system that can classify Bengali texts into suspicious and non-suspicious classes. Figure 1 shows an abstract view of the proposed suspicious text detection classifier. This system is consists of four major phases: training, feature extraction, classification and testing respectively.

3.1 Training Set Preparation

Training set $T = \{t_1, t_2, t_3, ..., t_n\}$ consists of n training text documents. Each text is labeled as either suspicious or non suspicious. Suspicious class is denoted

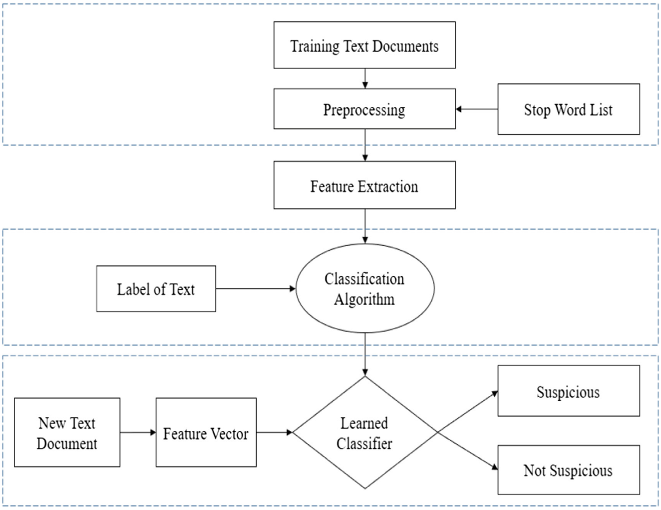
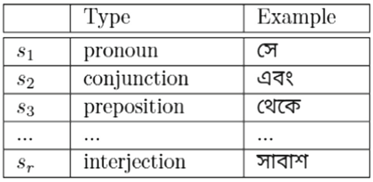


Fig. 1. Proposed framework for suspicious text

by C_s and non suspicious class is denoted by C_{ns}. A random text t_i with k words is represented by a word vector $W[] = \{w_1, w_2, w_3, ..., w_k\}$ in the system. All texts are prepossessed by the pre-processor in order to remove inconsistencies from dataset. A list $S[] = \{s_1, s_2, s_3, ..., s_r\}$ with r stop words has been developed which is a column vector where each row contains a stop word. In the proposed system a word w_i which has no contribution in deciding whether a text t_i is suspicious (C_s) or not suspicious (C_{ns}) is referred as stop word s_i. Pronoun, conjunction, preposition, interjection, prefix and suffix are considered as stop words. Stop words $s_1, s_2, s_3, ..., s_r$ are removed form the text t_i by matching with the stop word list $S[]$. Punctuation's in a text are also removed in pre-processing step (Fig. 2).



	Type	Example
s_1	pronoun	সে
s_2	conjunction	এবং
s_3	preposition	থেকে
...	...	...
s_r	interjection	সাবাশ

Fig. 2. Stop word list

3.2 Feature Extraction

A word list is created by the tokenizer by tokenizing the main body of a text. Word frequencies are used as features in this system. For the representation of features we use bag of words model. Table 1 shows a small fragment of feature space used in the system. Feature space ($F[][]$) is a two dimensional ($i \times j$) matrix with i rows and j columns. In this table, rows represents the texts $t_1, t_2, t_3, ..., t_i$ available in the corpus and columns represents total number of unique words $w_1, w_2, w_3, ..., w_j$ in the corpus. The value of i is 1500 and value of j is 3250 respectively as we have 1500 text document and 3250 unique words in our training set T. Each cell of the array represents the frequency (f_{ij}) of a specific word w_j occurs in a specific text t_i. Each row of the feature matrix represents features $F[][] = \{F[1], F[2], F[3], ..., F[n]\}$ for the texts of the dataset.

Table 1. A small fragment of feature space

c r	w_1	w_2	w_3	w_4	...	w_j
t_1	2	0	0	4	...	1
t_2	1	2	1	1	...	5
t_3	5	1	2	2	...	0
t_4	2	3	0	0	...	2
...	...	...	...	...	...	...
t_i	0	0	3	1	...	0

3.3 Classification

By using extracted vector of features $F[1], F[2], F[3], ..., F[n]$ and applying logistic regression is trained to classify texts as suspicious C_s and non suspicious C_{ns} [20].

SVM, decision tree and k-nearest neighbour are also use to evaluate the proposed system. The result of training phase is used in testing phase. This result is saved as a model (M) which classifies a new text document (t) into suspicious (C_s) and non suspicious (C_{ns}) category.

3.4 Testing Phase

Classification accuracy of the text classifier is calculated in testing phase. In the proposed suspicious text detector model testing phase is quite similar as training phase but in this part the learned classifier model is used for predicting. A test set $TS = \{ts_1, ts_2, ts_3, ..., ts_l\}$ is built to test the system which has l text document. Sample texts are taken to test the system. After processing, using feature extraction methods features ($F[][] = \{F[1], F[2], F[3], ..., F[l]\}$) are evicted from the testing texts $ts_1, ts_2, ts_3, ..., ts_l$. The trained classifier model use these features to classify a text ts_i as suspicious (C_s) and non suspicious (C_{ns}).

4 Dataset Preparation

Bengali is known as the low resource language thus, it is a very challenging task to build a corpus which contains a large amount of suspicious and non suspicious texts. Non suspicious have been collected data from a pre-build corpus [6]. Suspicious data are collected from different online and offline resources. All these data are stored in (.txt) format. U.S. department of homeland security and Berwyn police department define some properties of the suspicious activity [7]. We may adopt these properties for defining a text as suspicious if it has one of the following features.

- Texts contain words which hurt our religious feelings.
- Texts which provoke people against government.
- Texts which provoke people against law enforcement agencies.
- Texts which motivate people in terrorist events.
- Texts which excite a community without any reason.
- Texts which instigate our political parties.

Most of our suspicious data about religion are collected from online blogs [3,5]. Suspicious data about politics are collected from websites of different newspaper [1,2]. Data is also collected from different public pages of Facebook [4]. Table 2 represents the data statistics used in our model.

Table 2. Data statistics

	Training set	Testing set
Number of text documents	1500	500
Number of sentences	6744	2247
Number of words	26973	8991
Total unique words	3250	1045

In order to classify the texts, collected documents have been fed to the classifier model. As dataset is collected manually it may have some inconsistencies.

5 Evaluation Measures

In order to evaluate the proposed system several statistical measures is performed such as precision, recall, F_1 score, confusion matrix, and ROC curve respectively.

- Confusion Matrix: a table that is used for evaluating a classification model performance. As ours is a binary classification model, the confusion matrix of our system has two rows and two columns. This matrix reports total true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN) numbers.

- Precision: refers as positive predictive value. It calculates the ratio of exactly classified suspicious text to the total number of texts classified as suspicious. Precision can be obtained by Eq. 1.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

- Recall: calculates the ratio of correctly classified suspicious texts to the total number of suspicious texts. It is also referred as true positive rate (Eq. 2).

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

  Here, TP denotes the number of documents that is suspicious and also classified as suspicious, TN denotes the number of documents that is non suspicious and also classified as non suspicious, FN represents the number of documents that is suspicious but classified as non suspicious and FP means the number of documents that is non suspicious but classified as suspicious respectively.
- F_1 score: obtained from averaging the value of recall and precision. To choose a learning algorithm between several algorithms we have to find F_1 Score of algorithms. F1-score can be calculated by Eq. 3.

$$F_1 = \frac{2 * precision * recall}{precision + recall} \tag{3}$$

6 Experimental Results

Proposed suspicious text detection system is tested with logistic regression, Naive bayes, SVM, KNN and decision tree classification algorithms. Table 3 represents measures of these algorithms on our dataset. For all of the algorithms similar number of training and test documents have been used. Table 3 shows that logistic regression and SVM with different kernels are performing up to the mark on our dataset. Naive bayes and decision tree also doing really well. But accuracy of k-nearest neighbour is really poor compared to other algorithms.

Table 3. Performance comparison

Classification algorithm	Accuracy	Error	Precision	Recall	f_1 score
Naive bayes [16]	0.85	0.15	0.89	0.85	0.87
SVM (Linear kernel) [22]	0.91	0.09	0.91	0.91	0.91
SVM (RBF kernel) [21]	0.90	0.10	0.90	0.91	0.90
Logistic Regression (proposed)	0.92	0.08	0.92	0.93	0.93
K-Nearest Neighbor [13]	0.73	0.27	0.82	0.73	0.77
Decision Tree [11]	0.88	0.12	0.88	0.92	0.89

Classification report gives us precision, recall and f_1 score of each class which is really helpful to examine and find out shortcomings of the algorithm. Table 4 shows classification report of logistic regression for our system. In Table 4, the value of precision is shown for suspicious class (C_s) equal 0.92. It means the number of texts logistic regression classified as suspicious among them 92% are actually suspicious. It can correctly classify all non suspicious text (C_{ns}). From the recall value we get true positive rate for (C_s) is 1.00 and for (C_{ns}) is 0.93 respectively. Logistic regression gives similar f_1 score for both class that is 0.93. Average precision, recall and f_1 score is calculated by taking the mean value of C_s and C_{ns}.

Table 4. Classification report (proposed method)

Class (C)	Precision	Recall	f_1 score
Suspicious (C_s)	0.92	1.00	0.93
Non suspicious (C_{ns})	1.00	0.93	0.93
Avg./total	0.92	0.93	0.93

Precision-recall curves as well as receiver operating characteristics curves have been used for the evaluation of proposed model. There exists a trade off between positive predicted value and true positive rate which is summarized by precision-recall curve while ROC curve shows trade-off between the true and false positive rate using different probability thresholds for a predictive model. Figures 3, 4, 5 and 6 shows precision-recall and ROC curves for different algorithms used to test the proposed system.

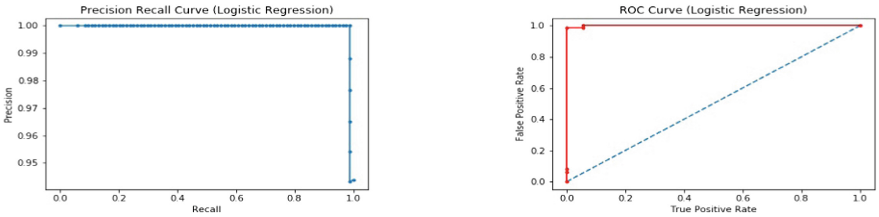


Fig. 3. Result of logistic regression

Figure 3 shows that we get high precision and recall for different threshold values with logistic regression and auc value of roc curve is also high which indicates it is a good classifier.

Figures 4 and 5 shows the result of SVM using different kernel tricks. Both algorithms give higher precision, recall, auc under roc curve for different values of threshold.

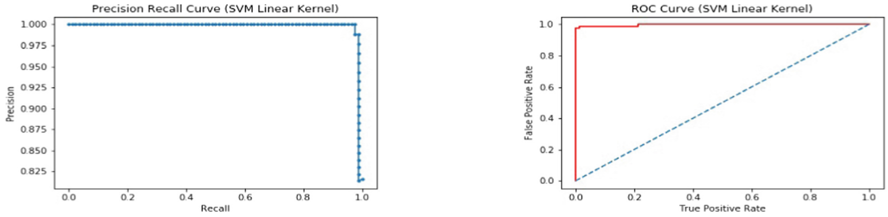


Fig. 4. Result of SVM (Linear Kernel)

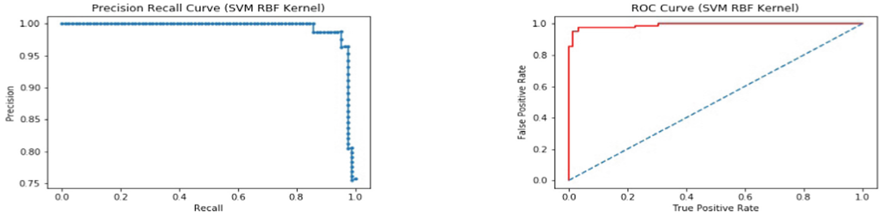


Fig. 5. Result of SVM (RBF Kernel)

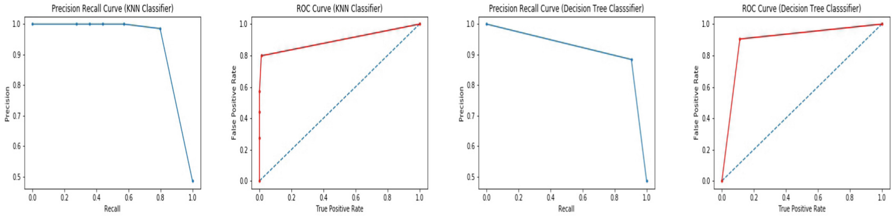
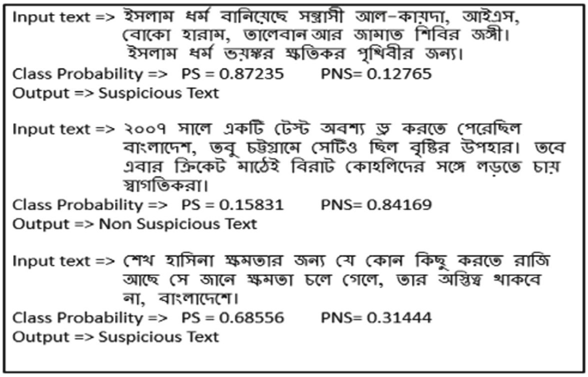


Fig. 6. Results of KNN and decision tree



```
Input text => ইসলাম ধর্ম বানিয়েছে সন্ত্রাসী আল-কায়দা, আইএস,
              বোকো হারাম, তালেবান আর জামাত শিবির জঙ্গী।
              ইসলাম ধর্ম ভয়ঙ্কর ক্ষতিকর পৃথিবীর জন্য।
Class Probability =>  PS = 0.87235      PNS= 0.12765
Output => Suspicious Text

Input text => ২০০৭ সালে একটি টেস্ট অবশ্য ড্র করতে পেরেছিল
              বাংলাদেশ, তবু চট্টগ্রামে সেটিও ছিল বৃষ্টির উপহার। তবে
              এবার ক্রিকেট মাঠেই বিরাট কোহলিদের সঙ্গে লড়তে চায়
              স্বাগতিকরা।
Class Probability =>  PS = 0.15831      PNS= 0.84169
Output => Non Suspicious Text

Input text => শেখ হাসিনা ক্ষমতার জন্য যে কোন কিছু করতে রাজি
              আছে সে জানে ক্ষমতা চলে গেলে, তার অস্তিত্ব থাকবে
              না, বাংলাদেশে।
Class Probability =>  PS = 0.68556      PNS= 0.31444
Output => Suspicious Text
```

Fig. 7. Output in system environment

For both k-nearest neighbor and decision tree after a certain value of recall these is a rapid decrease in precision. Figure 7 illustrates a sample input and its corresponding output of the proposed system as an example.

7 Conclusion

This paper proposes a system that can classify a Bengali text in terms of suspicious and non-suspicious categories. For this purpose, we developed a corpus and used logistic regression algorithm for classification task. The proposed system is evaluated on the test datasets and compare other machine learning techniques such as Naive Bayes, SVM, KNN, and decision tree. Among these logistic regression performed better in terms of accuracy (92%). As far our knowledge there are no work have been done on Bengali to detect suspicious text which makes this work a little attempt to compensate the scarcity. The overall exactness of the system can be improved by increasing the number of training text documents. Removing more stop words may also effect the system outcome in a positive way.

References

1. The Daily Jugantor. www.jugantor.com/
2. The Daily Kaler Kantho. http://www.kalerkantho.com/
3. Dhormockery Blog. https://www.dhormockery.com/
4. Facebook Page Basher Kella. https://www.facebook.com/basherkellanews/
5. Istishoner Blog. www.istishon.com/
6. Open Source Bengali Corpus. https://scdnlab.com/corpus/
7. U.S Department of Homeland Security. https://www.dhs.gov/see-something-say-something/what-suspicious-activity
8. Ahmad, A., Amin, M.R.: Bengali word embedding and it's application in solving document classification problem. In: International Conference Computer and Information Technology, pp. 425–430. IEEE (2016)
9. Alami, S., Beqali, O.: Detecting suspicious profiles using text analysis within social media. J. Theor. Appl. Inf. Technol. **73**(3) (2015)
10. Alsaleem, S., et al.: Automated arabic text categorization using SVM and NB. Int. Arab J. e-Technol. **2**(2), 124–128 (2011)
11. Chavan, G.S., Manjare, S., Hegde, P., Sankhe, A.: A survey of various machine learning techniques for text classification. Int. J. Eng. Trends Tech. **15**(6) (2014)
12. Chy, A.N., Seddiqui, M.H., Das, S.: Bangla news classification using naive Bayes classifier. In: International Conference on Computer and Information Technology, pp. 366–371. IEEE (2014)
13. Harisinghaney, A., Dixit, A., Gupta, S., Arora, A.: Text and image based spam email classification using KNN, Naïve Bayes and reverse DBSCAN algorithm. In: International Conference on Optimization, Reliability, and Information Technology, pp. 153–155. IEEE (2014)
14. Hossain, M.R., Hoque, M.M.: Automatic Bengali document categorization based on word embedding and statistical learning approaches. In: International Conference on Computer, Communication, Chemical, Material and Electronic Engineering, pp. 1–6. IEEE (2018)

15. Ismail, S., Rahman, M.S.: Bangla word clustering based on n-gram language model. In: International Conference on Electrical Engineering and Information and Communication Technology, pp. 1–5. IEEE (2014)
16. Jong, Y.Y., Dongmin, Y.: Classification scheme of unstructured text document using TF-IDF and naive Bayes classifier. In: Computer and Computing Science
17. Kaya, M., Fidan, G., Toroslu, I.H.: Sentiment analysis of Turkish political news. In: IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology, pp. 174–180. IEEE Computer Society (2012)
18. Krendzelak, M., Jakab, F.: Text categorization with machine learning and hierarchical structures. In: International Conference on Emerging eLearning Technologies and Applications, pp. 1–5. IEEE (2015)
19. Nizamani, S., Memon, N., Wiil, U.K., Karampelas, P.: Modeling suspicious email detection using enhanced feature selection. arXiv preprint arXiv:1312.1971 (2013)
20. Sharma, M., Zhuang, D., Bilgic, M.: Active learning with rationales for text classification. In: Conference of the North American Chapter of the ACL: Human Language Technologies, pp. 441–451 (2015)
21. Villmann, T., Bohnsack, A., Kaden, M.: Can learning vector quantization be an alternative to SVM and deep learning? - recent trends and advanced variants of learning vector quantization for classification learning. J. Artif. Intell. Soft Comput. Res. **7**(1), 65–81 (2017)
22. Wei, L., Wei, B., Wang, B.: Text classification using support vector machine with mixture of kernel. J. Softw. Eng. Appl. **5**, 55 (2012)