Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

Article

Detecting Suspicious Texts using Machine Learning Techniques

Omar Sharif[1], Mohammed Moshiul Hoque *, A. S. M. Kayes[2], Raza Nowrozy[2,3] and Iqbal H. Sarker[1]

1 Chittagong University of Engineering and Technology, Bangladesh; omar.sharif@cuet.ac.bd
2 La Trobe University, Australia; a.kayes@latrobe.edu.au, r.nowrozy@latrobe.edu.au
3 Victoria University, Australia
* Correspondence: moshiul_240@cuet.ac.bd (M.M. Hoque) and a.kayes@latrobe.edu.au (A.S.M. Kayes)

Abstract: Due to the substantial growth of internet users and its spontaneous access via electronic devices, the amount of electronic contents is growing enormously in recent years through instant messaging, social networking posts, blogs, online portals, and other digital platforms. Unfortunately, the misapplication of technologies has boosted with this rapid growth of online content which leads to the rise in suspicious activities. People misuse the web media to disseminate malicious activity, perform the illegal movement, abuse other people, and publicize suspicious contents on the web. The suspicious contents usually available in the form of text, audio or video, whereas text contents have been used in most of the cases to perform suspicious activities. Thus, one of the most challenging issues for NLP researchers is to develop a system that can identify suspicious text efficiently from the specific contents. In this paper, a Machine Learning (ML)-based classification model is proposed (hereafter called STD) to classify Bengali text into non-suspicious and suspicious categories based on its original contents. A set of ML classifiers with various features has been used on our developed corpus, consisting of 7000 Bengali text documents where 5600 documents used for training and 1400 documents used for testing. The performance of the proposed system is compared with the human baseline and existing ML techniques. The SGD classifier 'tf-idf' with the combination of unigram and bigram features are used to achieve the highest accuracy of 84.57%.

Keywords: Natural Language Processing; Suspicious Text Detection; Bengali Language Processing; Machine Learning; Text Classification; Feature Extraction; Suspicious Corpora

1. Introduction

Due to the effortless access of the Internet, www, blogs, social media, discussion forums, and online platforms via digital gadgets are producing a massive volume of digital text contents in recent years. It is observed that all the contents are not genuine or authentic; instead, some contents are faked, fabricated, forged or even suspicious. It is very unpropitious with this rapid growth of digital contents the ill-usage of the Internet has also been multiplied which governs to the boost in suspicious activities [1]. Suspicious contents are increasing day by day because of ill-usage of the Internet by few individuals to promulgate fierceness, share illegal activities, bullying other people, perform smishing, publicize incitement related contents, spread fake news, and so on. According to FBI's Internet Crime Complaint Center (IC3) report, a total of 467,361 complaints received in the year 2019 related to internet-facilitated criminal activity [2]. Moreover, several extremist users use social media or blogs to spread suspicious and violent contents which to be considered one kind of threat to national security [3].

Around 245 million people are speaking in Bengali as their native tongue, which makes it 7^{th} highest spoken language in the world [4]. However, research on Bengali Language Processing (BLP)

© 2020 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

is in initial stage till now, and there are no significant amount of works have been conducted yet like English, Arabic, Chinese or other European languages that makes Bengali a resource constrained language [5]. As far as we concern, none of the research conducted till to date on suspicious text detection in the Bengali language. But such systems are required to ensure the security as well as mitigate national threats in cyber-space.

Suspicious contents are those contents that hurt religious feelings, provoke people against government and law enforcement agencies, motivate people to perform act of terrorism, perform criminal acts by phishing, smishing, pharming, instigate a community without any reason, and execute extortion act [6–9]. As examples, social media has already used as a medium of communication in Boston attack and the revolution in Egypt [10]. The suspicious contents can be available in the form of video, audio, images, graphics and text. However, text plays an essential role in communication and most widely used medium of communication on cyber world. Moreover, the semantic meaning of a conversation can be retrieved by analyzing text contents which is difficult in other forms of content. In this work, we focus on *analyzing text content* and classifying the content into suspicious or non-suspicious.

A text could be detected as suspicious if it contained suspicious contents. It is impossible to detect suspicious texts from the enormous amount of internet text contents manually [11]. Therefore, the automatic detection of suspicious text contents should be developed. Responsible agencies are demanding some smart tool/ system that can detect suspicious text automatically. It will also be helpful to identify potential threats in cyber-world which are communicated by text contents. Automatic detection of suspicious text system can easily and promptly detect the fishy or threatening texts. Law and enforcement authority can take appropriate measures immediately, which in turn helps to reduce virtual harassment, suspicious and criminal activities mediated through online. But it is a quite challenging task to classify the Bengali text contents into suspicious or non-suspicious class due to its complex morphological structure, enormous numbers of synonym, and rich variations of verb auxiliary with the subject, person, tense, aspect, and gender. Moreover, scarcity of resources and lack of benchmark Bengali text dataset are the major barriers to build a suspicious text detection system and make it more difficult to implement compare to other languages. Therefore, the research question addressing in this paper is - "RQ: How can we effectively classify potential Bengali texts into suspicious and non-suspicious categories?"

To address this research question in this work, we first develop a dataset of suspicious and non-suspicious texts considering a number of well-known Bengali data sources, such as Facebook posts, blogs, websites and newspapers. In order to process the textual data, we take into account unigram, bigram, trigram features using tf-idf and bag of words feature extraction technique. Once the feature extraction has been done, we employ the most popular machine learning classifiers (i.e., logistic regression, naive bayes, random forest, decision tree, stochastic gradient descent) to classify whether a given text is suspicious or not. We have also performed a comparative analysis of these machine learning models utilizing our collected datasets. The key contributions of our work illustrate in the following:

- Develop a corpus containing 7000 text documents labelled as suspicious or non-suspicious.
- Design a classifier model to classify Bengali text documents into suspicious or non-suspicious categories on developed corpus by exploring different feature combination.
- Compare the performance of the proposed classifier with various machine learning techniques as well as the existing method.
- Analyze the performance of the proposed classifier on different distributions of the developed dataset.
- Exhibits a performance comparison between human expert (i.e., baseline) and machine learning algorithms.

We expect that the work presented in this paper will play a pioneering role in the development of Bengali suspicious text detection systems. The rest of the paper organized as follows: Section 2

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

presents related work. In Section 3, a brief description of the development of suspicious Bengali corpus and its several properties have explained. Section 4 explained the proposed Bengali suspicious text document classification system and its significant constituents. Section 5 described the evaluation techniques used to assess the performance of the proposed approach. Several experimental results are presented in Section 6. Finally, in Section 7, we concluded the paper with a summary and discussed the future scopes.

2. Related Work

Suspicious contents detection is a well-studied research issue for the highly resourced languages like Arabic, Chinese, English, and other European languages. However, no meaningful research activities have been conducted yet to classify text with suspicious content in BLP domain. A machine learning-based system developed to detect promotion of terrorism by analyzing the contents of a text. Iskandar et al. [12] have collected data from Facebook, Twitter and numerous micro-blogging sites to train the model. By performing a critical analysis of different algorithms, they showed that Naïve Bayes is best suited for their work as it deals with probabilities [13]. Johnston et al. [14] proposed a neural network-based system which can classify propaganda related to the Sunni[1] extremist users on social media platforms. Their approach obtained 69.9% accuracy on the developed dataset. A method to identify suspicious profiles within social media presented where normalized compression distance was utilized to analyze text [15]. Jiang et al. [16] discusses current trends and provides future direction to determine suspicious behaviour in various mediums of communications. The researchers investigated the novelty of true and false news on 126,000 stories that tweeted 4.5 million times using ML techniques [17]. An automated system explained the technique of detecting hate speech from the twitter data [18]. Logistic regression with regularization outperforms other algorithms by attaining the accuracy of 90%. An intelligent system introduced to detect suspicious messages from Arabic tweets [19]. This system yields maximum accuracy of 86.72% using SVM with a limited number of data and class. Dinakar et al. [20] developed a corpus of YouTube comments for detecting textual cyberbullying using a multiclass and binary classifier. A novel approach presented of detecting Indonesian hate speech by using SVM, lexical, word unigram and tf-idf features [21]. A method described to detects abusive content and cyberbullying from Chinese social media. Their model achieved 95% accuracy by using LSTM and taking characteristical and behavioural features of a user [22]. Hammer H. L. [23] discussed a way of detecting violence and threat from online discussions towards minority groups. This work considered the manually annotated sentences with bigram features of essential words.

Since Bengali is an under-resourced language, the amount of digitized text (related to suspicious, fake, or instigation) is quite less. In addition to that, no benchmark dataset is available on the suspicious text. Due to these reasons, very few research activities have carried out in this area of BLP, which are mainly related to hate, threat, fake and abusive text detection. Ishmam et al. [24] compare machine learning and deep learning-based model to detect hateful Bengali language. Their method achieved 70.10% accuracy by employing a gated recurrent neural network (GRNN) method on a dataset of 6 classes and 5K documents collected from numerous Facebook pages. The reason behind this poor accuracy is the less number of training documents in each class (approximately 900). Most importantly, they did not define the classes clearly, which is very crucial for the hateful text classification task. Recent work explained a different machine and deep learning technique to detect abusive Bengali comments [25]. The model acquired 82% accuracy by using RNN on 4700 Bengali text documents. Ehsan et al. [26] discussed another approach of detecting abusive Bengali text by combining different n-gram features and ML techniques. Their method obtained the highest accuracy for SVM with trigram features. A method to identify malicious contents from Bengali text is presented by Islam et al. [27]. This method achieved 82.44% accuracy on an unbalanced dataset of 1965 instances

[1] Sunni is a class of Islamic believer group of Muslims: www.britannica.com/topic/Sunni

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

by applying the Naive Bayes algorithm. Hossain et al. [28] develop a dataset of 50k instances to detect fake news in Bangla. They have extensively analyzed linguistic as well as machine learning-based features. A system demonstrated the technique to identify the threats and abusive Bengali words in social media using SVM with linear kernel [29]. The model experimented with 5644 text documents and obtained the maximum accuracy of 78%.

As far as we aware, none of the remarkable research conveyed so far that focuses on detecting suspicious Bengali text. Our previous approach used logistic regression with BoW features extraction technique to detect suspicious Bengali text contents [30]. However, that work considered only 2000 text documents and achieved an accuracy of 92%. In this work, our main concern is to develop the ML-based suspicious Bengali text detection model trained on our new dataset by exploring various n-gram features and feature extraction techniques.

3. A Novel Suspicious Bangla Text Dataset

To till this date, no dataset is available for identifying Suspicious Bengali Texts (SBT). Therefore, we developed a Suspicious Bengali Text Dataset (SBTD), which is a novel annotated corpus to serve our purpose. The following subsection explains the definition of SBT with its inherent characteristics and details statistics of the developed SBTD.

3.1. Suspicious Text and Suspicious Text Detection

Suspicious Text Detection (STD) system classifies a text $t_i \epsilon T$ from a set of texts $T = \{t_1, t_2, ..., t_m\}$ into a class $c_i \epsilon C$ from a set of two classes $C = \{C_s, C_{ns}\}$. The task of STD is to automatically assign t_i to c_i: $< t_i, c_i >$.

Deciding whether a Bengali text is suspicious or not is not so simple even for language experts because of its complicated morphological structure, rich variation in sentence formation and lack of defining related terminology. Therefore, it is very crucial to have a clear definition of SBT for making the task of SBTD smoother. In order to introduce a reasonable definition concerning the Bengali language, several definitions of violence, incitement, suspicious and hatred contents have analyzed. Most of the information collected from the different social networking websites and scientific papers summarized in Table 1.

Table 1. Definitions of hatred, incitement and violent contents according to different social networking websites, organization, and scientific studies

Source	Definition
Facebook	"Contents that incite or facilitate serious violence pose credible threat to the public or personal safety, instructions to make weapons that could injure or kill people and threats that lead to physical harm towards private individuals or public figures". [6]
Twitter	"One may not promote terrorism or violent extremism, harasses or threaten other people, incite fury toward a particular or a class of people". [31]
YouTube	"Contents that incite others to promote or commit violence against individuals and groups based on religion, nationality, ethnicity, sex/gender, age, race, disability, gender identity/sexual orientation". [32]
Council of Europe (COE)	"Expression which incite, spread, promote or justify violence toward a specific individual or class of persons for a variety of reasons". [33]
Paula et al.	"Language that glorify violence and hate, incite people against groups based on religion, ethnic or national origin, physical appearance, gender identity or other". [7]

Majority of the quoted definitions focus on similar attributes such as incitement of violence, promotion of hate and terrorism, threaten a person or group of people. These definitions cover the larger aspect of suspicious content from video, text, image, cartoon, illustrations, and graphics. Nevertheless,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

Tokenization gives a list of words of the input text such as, t_i = ['Ei', 'khulna', 'titans', 'ke', 'tin', 'wickete', 'hariye', 'dilo', 'comilla', 'victoria']

- **Removal of stop words:** Words which has no contribution in deciding whether a text t_i is (C_s) or (C_{ns}) is considered as unnecessary. Such words are dispelled from the document by matching with a list of stop words. Finally, after removing the stop words, the processed text as, t_i ="Khulna titans ke tin wickete hariye dilo comilla victoria". (*English translation:* Comilla Victoria defeated Khulna Titans by three wickets) will be used for training.

With the help of above operations a set of processed texts is created. These texts are stored chronologically in a dictionary in the form of array indexing $A[t_1]....A[t_{7000}]$ .with a numeric $(0,1)$ label. Here, 0 and 1 represents non-suspicious and suspicious class respectively.

4.2. Feature Extraction

Machine learning models could not possibly learn from the texts that we have prepared. Feature extraction performs numeric mapping on these texts to find some meaning. This work explored the bag of words (BoW) and term frequency-inverse document frequency (tf-idf) feature extraction techniques to extract features from the texts.

BoW technique uses the word frequencies as features. Here each cell gives the count (c) of a feature word (f_{wi}) in a text document (t_i). Unwanted words may get higher weights than the context-related words on this technique. Tf-idf technique tries to mitigate this weighting problem by calculating tf-idf value according to the equation 1.

$$tf - idf(f_{wi}, t_i) = tf(f_{wi}, t_i) \log \frac{m}{|t \epsilon m : f_w \epsilon t|} \tag{1}$$

Here, $tf - idf(f_{wi}, t_i)$ indicates the tf-idf value of word f_{wi} in text document (t_i), $tf(f_{wi}, t_i)$ indicates the frequency of word f_{wi} in text document (t_i), m means total number of text documents, $|t \epsilon m : f_w \epsilon t|$ represents the number of text document t containing word f_w.

Tf-idf value of the feature words $((f_w))$ puts more emphasis on the words related to the context than other words. To find the final weighted representation of the sentences, compute euclidean norm after calculating $tf - idf$ value of the feature words of a sentence. This normalization set high weight on the feature words with smaller variance. Equation 2 compute the norm.

$$X_{norm}(i) = X_i / \sqrt{(X_1)^2 + (X_2)^2 + ... + (X_n)^2} \tag{2}$$

Here, $X_{norm}(i)$ is the normalized value for the feature word f_{wi} and $X_1, X_2, ..., X_n$ are the $tf - idf$ value of the feature word $f_{w1}, f_{w2}, ..., f_{wn}$ respectively. Features picked out by both techniques have been applied on the classifier. Table 4 shows sample feature value for both feature extraction techniques.

Table 4. Small portion of (BoW and tf-idf) feature space

r \ c	Technique	f_{w1}	f_{w2}	f_{w3}	...	f_{wn}
t_1	BoW	1	0	4	...	2
	tf-idf	0.35	0.43	0.31	...	0.23
t_2	BoW	5	2	1	...	10
	tf-idf	0.07	0.30	0.20	...	0.57
...	...	...	...	...	...	...
	...		...	...	...	...
t_m	BoW	2	0	7	...	9
	tf-idf	0.17	0.63	0.09	...	0.65

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

Features exhibited by an array of size $(m * n)$ having m rows and n columns. The total number of text documents $t_1, t_2, ..., t_m$ are represented in rows while all the feature words $f_{w1}, f_{w2}, .., f_{wn}$ are represented in columns. In order to reduce complexity and computational cost, 3000 most frequent words considered as feature words among thousands of unique words.
The model extracted linguistic n-gram features of the texts. N-gram approach uses to take account the sequence order in a sentence in order to make more sense from the sentences. Here 'n' indicates the number of consecutive words that can treat as one gram. N-gram, as well as a combination of n-gram features, will be applied in the proposed model. Table 5 shows the illustration of various n-gram features.

Table 5. Representation of different N-gram features for a sample Bangla text (Banglish form)

N-grams	"Khulna titans ke tin wickete hariye dilo comilla victoria"
unigrams	'khulna', 'titans', 'ke', 'tin', 'wickete', 'hariye', 'dilo', 'comilla', 'victoria'
bigrams	'khulna titans', 'titans ke', 'ke tin', 'tin wickete', 'wickete hariye', 'hariye dilo', 'dilo comilla', ' comilla victoria'
trigrams	'khulna titans ke', 'titans ke tin', 'ke tin wickete', 'tin wickete hariye', 'wickete hariye dilo', 'hariy dilo comilla', 'dilo comilla victoria'

Combination of two feature extraction techniques and n-gram features will apply to find the best-suited model for the accomplishment of suspicious Bengali text detection.

4.3. Classification

Features that we got from the previous step uses to train the machine learning model by employing different popular classification algorithms [37]. These algorithms are stochastic gradient descent (SGD), logistic regression (LR), decision tree (DT), random forest (RF) and multi-nomial naïve Bayes (MNB). Logistic regression [38] is well suited for our task as it is a binary classification problem. Equation 3-4 defines the logistic function which determines the output of logistic regression.

$$h_\theta(x) = \frac{1}{1 + \exp(-\theta^T x)} \tag{3}$$

Cost function is,

$$C(\theta) = \frac{1}{m} \sum_{i-1}^{m} c(h_\theta(x^i), y^i) \tag{4}$$

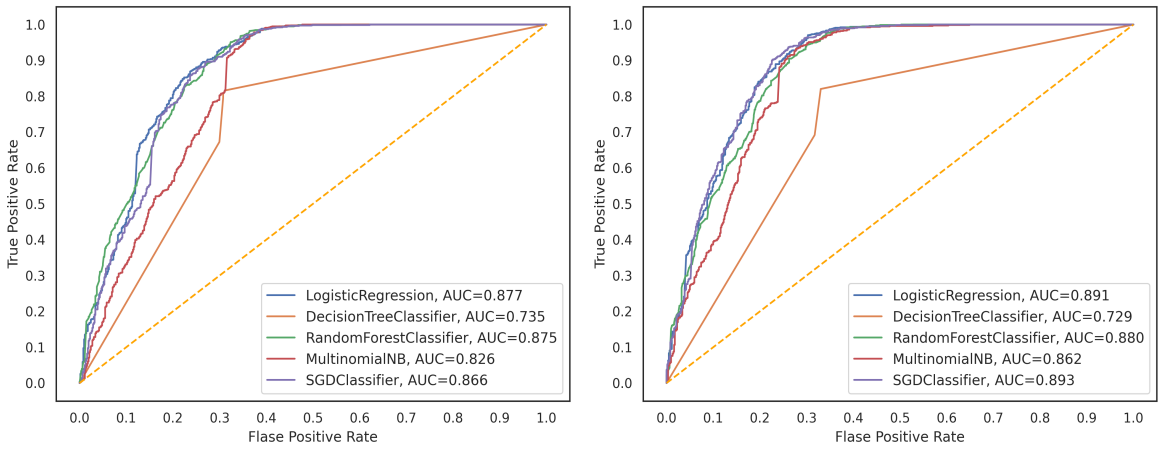
$$c(h_\theta(x), y) = \begin{cases} -\log(1 - h_\theta(x)) & \texttt{if } y = 0 \\ -\log(h_\theta(x)) & \texttt{if } y = 1 \end{cases}$$

Here,
m = Number of training examples.
$hp_\theta(x^i)$ = Hypothesis function of i^{th} training example
y^i = Input label of i^{th} training example.

The decision tree has two types of nodes external and internal. External nodes represent decision class while internal nodes hold the features essential for making classification [39]. The random forest comprises of several decision trees which operate individually. Each decision tree produces a class prediction, and the class with the maximum number of votes will be the model prediction. Naïve Bayes [40] follows Bayesian theorem where variables $V_1, V_2, ..., V_n$ of class C are conditionally independent of each other given C. Stochastic gradient descent takes account of only one sample point randomly while adjusting weights. SGD classifier [41] performs better than others, especially on a large dataset, although it takes more steps to converge.

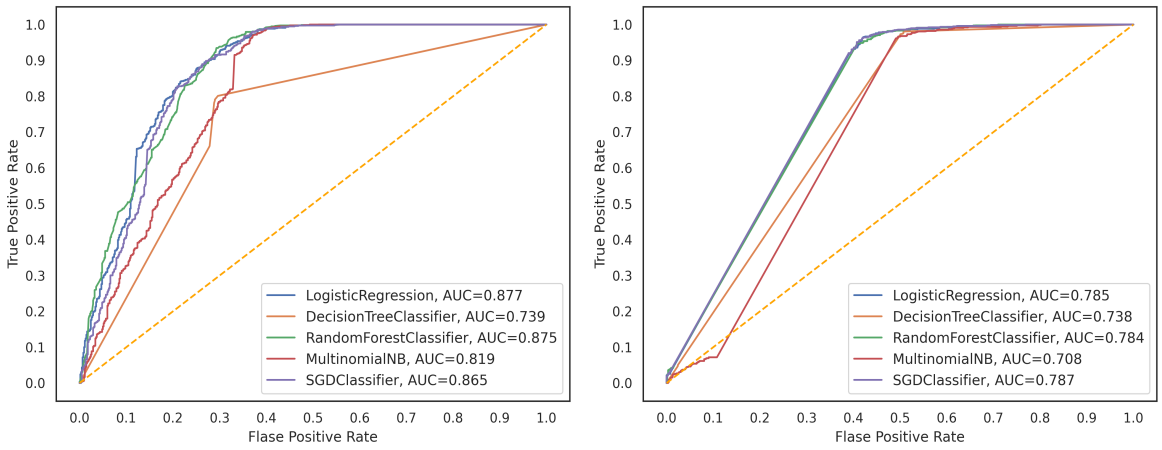
Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

technique for F1, F2 and F3 feature respectively. For BoW with F1 feature, logistic regression and random forest both provide the similar AUC values of 87.8%. SGD achieved 87.1% AUC, which increased by 2.2% by using tf-idf FE technique. AUC value of other algorithms also increased by employing the tf-idf feature extraction technique.



(a) ROC curve for (BoW+F2) features **(b)** ROC curve for (tf-idf+F2) features

Figure 6. ROC curve analysis for F2 feature. F2 represents combination of unigram and bigram feature



(a) ROC curve for (BoW+F3) features **(b)** ROC curve for (tf-idf+F3) features

Figure 7. ROC curve analysis for F3 feature. F3 represents combination of unigram, bigram and trigram feature

With tf-idf and F2 feature, SGD beats others by getting maximum AUC of 89.3%. The value of all the classifiers raised except the decision tree. Its value decreased by 0.06%. For F3 feature with BoW LR, RF, SGD performed quite well by acquiring 87.7%, 87.5% and 86.5% AUC value respectively. AUC value decreased abruptly by (9-10)% for tf-idf and F3 feature, which make it worst feature combination. Critical analysis of results brings to the notation that, SGD classifier with the combination of unigram and bigram feature for tf-idf feature extraction technique achieved the highest value for most of the evaluation parameters compare to others. The performance of the proposed classifier (SGD) analyzed further by varying the number of training documents to get more insight.

Fig. 8 shows the accuracy versus the number of training examples graph. The analysis reveals that the classification accuracy is increasing with the increased dataset and the tf-idf predominates the BoW with F2 feature.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

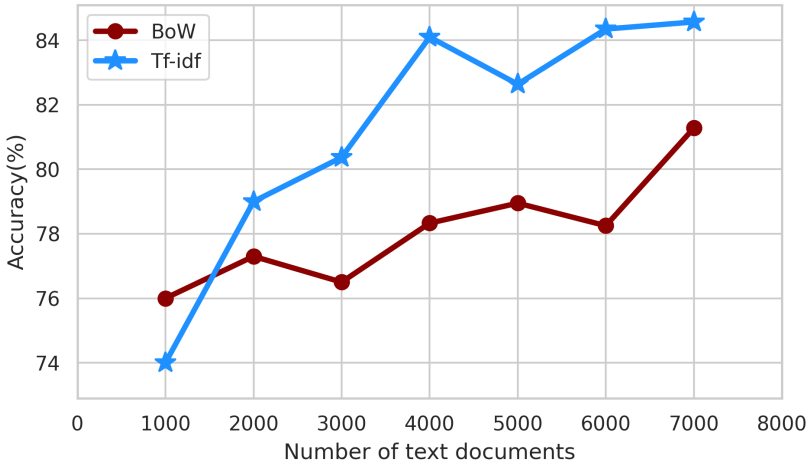


Figure 8. Effects of training set size on accuracy

6.3. Human Baseline vs. ML Techniques

The performance of the classifiers compared with two human experts for further investigation. Expert 1 is a university professor with BLP research domain. Expert 2 is a PhD fellow pursuing his PhD on BLP. Experts have manually classified testing text documents into one of the predefined categories. Among 1400 test text samples 621 texts are from non-suspicious (C_{ns}) class and 779 texts are from suspicious (C_s) class. Accuracy of each class can be computed by the ratio between the number of correctly predicted texts and the total number of texts of that class by using the confusion matrix. Suppose, a system can correctly predict 730 texts among 779 suspicious texts, then its accuracy in suspicious class will be 93.7% (730/779). As the tf-idf outperformed the BoW in the previous evaluation, so we compared the performance of the classifiers only for tf-idf feature extraction technique with human experts. Table 7 exhibits the summary of comparison.

Table 7. Accuracy comparison between experts and classifiers with tf-idf FE technique.

Approach	C_{ns} **Accuracy(%)**	C_s **Accuracy(%)**
Expert 1	98.71	98.58
Expert 2	99.19	98.33
LR+F1	76.16	90.24
LR+F2	75.36	90.62
LR+F3	56.04	95.25
DT+F1	67.63	81.01
DT+F2	70.20	79.46
DT+F3	52.49	94.99
RF+F1	69.88	94.74
RF+F2	69.24	94.35
RF+F3	55.71	95.63
MNB+F1	76.65	90.75
MNB+F2	74.87	91.01
MNB+F3	44.61	96.02
SGD+F1	72.94	92.81
SGD+F2	80.52	85.50
SGD+F3	53.62	95.89

The experts outperformed the ML classifiers in both classes. Experts can more accurately classify non suspicious texts than suspicious texts. We found approximately 0.5% accuracy deviation between experts. SGD classifier with F2 feature obtained good accuracy for both classes while others were done well on suspicious class and performed very poorly on non-suspicious class. A significant difference has

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

24. Ishmam, A.M.; Sharmin, S. Hateful Speech Detection in Public Facebook Pages for the Bengali Language. 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA). IEEE, 2019, pp. 555–560.
25. Emon, E.A.; Rahman, S.; Banarjee, J.; Das, A.K.; Mittra, T. A Deep Learning Approach to Detect Abusive Bengali Text. 2019 7th International Conference on Smart Computing & Communications (ICSCC). IEEE, 2019, pp. 1–5.
26. Eshan, S.C.; Hasan, M.S. An application of machine learning to detect abusive bengali text. 2017 20th International Conference of Computer and Information Technology (ICCIT). IEEE, 2017, pp. 1–6.
27. Islam, T.; Latif, S.; Ahmed, N. Using Social Networks to Detect Malicious Bangla Text Content. 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT). IEEE, 2019, pp. 1–4.
28. Hossain, M.Z.; Rahman, M.A.; Islam, M.S.; Kar, S. BanFakeNews: A Dataset for Detecting Fake News in Bangla. *arXiv preprint arXiv:2004.08789* **2020**.
29. Chakraborty, P.; Seddiqui, M.H. Threat and Abusive Language Detection on Social Media in Bengali Language. 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT). IEEE, 2019, pp. 1–6.
30. Sharif, O.; Hoque, M.M. Automatic Detection of Suspicious Bangla Text Using Logistic Regression. International Conference on Intelligent Computing & Optimization. Springer, 2019, pp. 581–590.
31. Twitter. Hateful Conduct. Available online: https://help.twitter.com/en/rules-and-policies/twitter-rules/ (accessed on 25 April 2019).
32. Youtube. Harmful or dangerous content policy. Available online: https://support.google.com/youtube/answer/2801939/ (accessed on 27 April 2019).
33. COE. Hate Speech and Violence. Available online: https://www.coe.int/en/web/european-commission-against-racism-and-intolerance/hate-speech-and-violence/ (accessed on 18 April 2019).
34. U.S Department of Homeland Security. Available online: https://www.dhs.gov/see-something-say-something/what-suspicious-activity (accessed on 13 April 2019).
35. Dash, N.S.; Ramamoorthy.; Naicker, L. Utility & Application of Language Corpora **2019**. pp. 17–34.
36. Cohen, J. A coefficient of agreement for nominal scales. *Educational and psychological measurement* **1960**, *20*, 37–46.
37. Sarker, I.H.; Kayes, A.; Watters, P. Effectiveness analysis of machine learning classification models for predicting personalized context-aware smartphone usage. *Journal of Big Data* **2019**, *6*, 57.
38. Pranckevičius, T.; Marcinkevičius, V. Application of logistic regression with part-of-the-speech tagging for multi-class text classification. 2016 IEEE 4th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE). IEEE, 2016, pp. 1–5.
39. Pranckevičius, T.; Marcinkevičius, V. Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification. *Baltic Journal of Modern Computing* **2017**, *5*, 221.
40. Le, C.C.; Prasad, P.; Alsadoon, A.; Pham, L.; Elchouemi, A. Text classification: Naïve bayes classifier with sentiment Lexicon. *IAENG International Journal of Computer Science* **2019**, *46*, 141–148.
41. Kabir, F.; Siddique, S.; Kotwal, M.R.A.; Huda, M.N. Bangla text document categorization using stochastic gradient descent (sgd) classifier. 2015 International Conference on Cognitive Computing and Information Processing (CCIP). IEEE, 2015, pp. 1–4.
42. Google. Google Colaboratory. Available online: https://colab.research.google.com/notebooks/welcome (accessed on 7 January 2020).
43. Tagami, T.; Ouchi, H.; Asano, H.; Hanawa, K.; Uchiyama, K.; Suzuki, K.; Inui, K.; Komiya, A.; Fujimura, A.; Yanai, H.; others. Suspicious News Detection Using Micro Blog Text. *arXiv preprint arXiv:1810.11663* **2018**.
44. Ahmed, H.M.; Bethoon, N.B. Cybercrime: Suspicious Viber Messages Detection Model. *International Journal of Scientific & Engineering Research* **2017**, *8*, 1496–1502.
45. Nizamani, S.; Memon, N.; Wiil, U.K.; Karampelas, P. Modeling suspicious email detection using enhanced feature selection. *arXiv preprint arXiv:1312.1971* **2013**.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 August 2020 doi:10.20944/preprints202008.0033.v1

46. Diab, S. Optimizing stochastic gradient descent in text classification based on fine-tuning hyper-parameters approach. A Case Study on Automatic Classification of Global Terrorist Attacks. *arXiv preprint arXiv:1902.06542* **2019**.