N-Gram-Based Machine Learning Approach for Bot or Human Detection from Text Messages

Durga Prasad Kavadi
Department of Information Technology, B V Raju Institute of Technology, Narsapur, Telangana, India Email: durgaprasad85.kavadi@gmail.com

Chandra Sekhar Sanaboina
Department of Computer Science and Engineering, University College of Engineering Kakinada, Jawaharlal Nehru Technological University Kakinada
chandrasekhar.s@jntucek.ac.in

Rizwan Patan
Decentralized Science Lab, College of Computing and Software Engineering, Kennesaw State University, Marietta, GA 30060, USA
rpatan@kennesaw.edu

Amir H. Gandomi*
Faculty of Engineering & Information Systems, University of Technology Sydney, Ultimo, NSW, Australia
gandomi@uts.edu.au

ABSTRACT

Social bots are computer programs created for automating general human activities like the generation of messages. The rise of bots in social network platforms has led to malicious activities such as content pollution like spammers or malware dissemination of misinformation. Most of the researchers focused on detecting bot accounts in social media platforms to avoid the damages done to the opinions of users. In this work, n-gram based approach is proposed for a bot or human detection. The content-based features of character n-grams and word n-grams are used. The character and word n-grams are successfully proved in various authorship analysis tasks to improve accuracy. A huge number of n-grams is identified after applying different pre-processing techniques. The high dimensionality of features is reduced by using a feature selection technique of the Relevant Discrimination Criterion. The text is represented as vectors by using a reduced set of features. Different term weight measures are used in the experiment to compute the weight of n-grams features in the document vector representation. Two classification algorithms, Support Vector Machine, and Random Forest are used to train the model using document vectors. The proposed approach was applied to the dataset provided in PAN 2019 competition bot detection task. The Random Forest classifier obtained the best accuracy of 0.9456 for bot/human detection.

CCS CONCEPTS

• **Human-centered computing**; • **Human computer interaction (HCI)**; • **HCI design and evaluation methods**; • **User models**;

ISMSI 2022, April 09, 10, 2022, Seoul, Republic of Korea

ACM ISBN 978-1-4503-9628-8/22/04…$15.00
https://doi.org/10.1145/3533050.3533063

KEYWORDS

Bot Detection, Character N-Grams, Word N-Grams, Feature Selection Technique, Term Weight Measure

ACM Reference Format:
Durga Prasad Kavadi, Chandra Sekhar Sanaboina, Rizwan Patan, and Amir H. Gandomi*. 2022. N-Gram-Based Machine Learning Approach for Bot or Human Detection from Text Messages. In *2022 6th International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence (ISMSI 2022), April 09, 10, 2022, Seoul, Republic of Korea.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3533050.3533063

1 INTRODUCTION

People are changing their methods for communication day by day by utilizing the social network platforms like Twitter, Facebook, Blogs, Instagram, and Forums. People feel comfortable posting, talking and discussing their comments and reviews about any issue on these platforms. Many people depend on different social forums like Yelp, Reddit, Amazon message boards and Quora to get recommendation, feedback and information about services and products. These platforms are used by several people for good purposes but some set of persons misuse such platforms by creating bots, spam messages and fake profiles [1]. The bots are used to act like a human for influencing the users of social media for political, commercial or ideological purposes. Bots are used for multiple purposes like enhancing the reputation of the products and services by writing positive ratings and positive reviews, damaging the reputation of good products and services by giving negative ratings and reviews [2]. Furthermore, bots are also used for spreading fake news. Therefore, developing an automatic system for bot/human detection is very important in recent times to reduce the damages in social networking platforms.

So many years, the bots are used for different variety of purposes. Initial times, they used purposefully to automate online processes that are not possible to do manually, and now they are used mostly for commercial activities like posting spam information on different social media platforms and directing users of the internet to advertisements. Bots are also used for illegal activities like data collection from users for increasing criminal activities and influencing the results of elections by sharing fake news about politicians. Bot detection becomes an important task in various security-related

activities. Bot detection was helpful in preventing influential operations when monitoring big events like elections [3].

To avoid damages in social media platforms through smart bots, different platforms like Twitter and Facebook developed algorithms to detect the bots automatically and delete such accounts. In some situations, bot detection becomes a challenging problem because the latest advancements in natural language processing make the problem most complex. In order to detect bot in these platforms, some approaches analyse the set of documents rather than a single document because the bots accounts follow a stylistic and lexical pattern. Some researchers identified that humans discussed different types of topics when compared with bots. The bots have a more compressed feature vector because the variety of words used in their texts is less when compared with texts of humans.

Some researchers used compressed features by concatenating the tweets of the particular user into one document and applying different statistical calculations on compressed documents. The reason for using compressed features is there is an assumption in the case of bots is the bots communicate in a repetitive way when compared with humans. Especially, the spambots are posting the same tweet multiple times without any modifications in the content. Compression features are used to detect such behaviour by determining the difference in compression ratios among bot and human accounts [4].

The machine learning algorithms generally used more varieties of features such as number of followers for a user account, number of users followed by user account, number of retweets in periodic intervals, the history and background information of the user, the content of tweets etc. for bot/human detection in the Twitter dataset. The machine learning algorithms are successfully exploited by the researchers to achieve more accuracy for bot detection [5]. The PAN competition 2019 provided a training dataset for bot/human detection. This dataset contains 4120 authors files, and each author file contains 100 tweets. The provided dataset contains only tweet content information only. The dataset providers are not mentioned other details like followers count, count of followed users, time intervals of tweets and history information of authors. In this work, a machine learning approach is developed by using the content information of tweets.

In general, the bot's writing style is different from the writing styles of humans. The stylistic features are more helpful in differentiating the accounts of bots and humans. The emoticons utilization is also one important feature to differentiate the writing style of humans and bots. Some researchers observed that the bots generate most of the plain text, whereas the human generates both text emoticons in their writings. Some identified that the bots used more emoticons to avoid the retyping of text. The usage of positive and negative words is more in the writings of the bot when compared with human writings because most of the bots create messages to criticise some persons, products, and services.

In this work, the experiment was conducted with content-based features of character and word n-grams for preparing the document vectors. The n range in character n-grams is 2 to 5, the n range in word n-grams is 1 to 3. To reduce the dimensionality of feature vector, a feature selection technique is used which identifies the best informative features. The feature value in the feature vector representation is determined by using different supervised term weight measures. Two Machine Learning (ML) algorithms such as Random Forest (RF) and Support Vector Machine (SVM) are used to create the model for classification. The PAN competition 2019 bot detection dataset is used in this experiment. The efficiency of the proposed approach is evaluated by using an evaluation measure of accuracy.

This paper is planned in 9 sections. The existing works proposed for bot detection is described in section 2. The description of the dataset is presented in section 3. Evaluation measures are explained in section 4. The machine learning algorithms are briefly described in section 5. Section 6 explains the proposed n-gram based approach and also explained the feature selection technique and term weight measures used in the proposed approach. The experimental results are presented and analysed in section 7. The results are discussed in section 8. The paper's conclusions and planned work in the future are explained in section 9.

2 RELATED WORK

The rising of social media platforms changed the habit of people's interactions and communications. These platforms provide an environment for people to share and post their opinions and thoughts on any topic. However, there is no proper controlling mechanism to control unauthorized activities. These platforms allow malicious accounts like social bots to propagate and post false information that shows a negative impact on society. Social bots are computer-based algorithms that show behavior like humans and are able to interact with users by generating content. The bots are created for performing useful tasks like responding to user requests by giving useful information. Although, some bots are used for negative purposes like altering the opinions of users, performing malicious activities, and promoting advertisement of recruitment for terrorists [1].

Several automatic bot detectors are developed for Twitter to detect the bot accounts and to delete such accounts. "Bot or Not?" is the first developed social bot detector for Twitter which is available publicly on the web [6]. This bot detection algorithm used more than 1000 features to detect bots on Twitter. The features are classified into groups such as sentiment, friends, content, temporal, network and user. SentiBot [7] is another bot detection system developed for Twitter. This system concentrated on the factors related to sentiment to identify the bot accounts. Sentibot extracts the relevant features for classifier training by employing different sentiment analysis techniques.

Chu, et al. [8], developed a system for the classification of cyborg, bot and human accounts in the Twitter dataset. Cyborg is a type of bot that are assisted by humans or human-assisted bots. They collected 500000 accounts data from Twitter. The authors considered different types of features such as behaviour of tweeting, the content of a tweet, and properties of accounts to study the differences among cyborg, bot, and human accounts. The experiment was performed with a Random Forest classifier and applied on 2000 accounts test dataset. The proposed model obtained accuracies of 91%, 96%, and 98% for classifying cyborgs, bots, and humans, respectively.

Shaina Ashraf et al. [9], build the system for Bots detection for differentiating messages whether they are generated by the bot or human account by using 27 language-independent Stylometry

features. The Stylometry features are classified into two classes such as character-based features of 18 and emotion-based features of 9. The character-based features include count of spaces, count of URL's, count of capital letters, length of a text in characters, count of curly brackets, count of round brackets, count of underscores, question marks, exclamation mark, dollar symbol, ampersand symbol, tags, hashes, slashes, operators, punctuations, lines, words. The emotion-based features include a count of emojis, emoticons, smiling face emojis, affection face emojis, count of face emojis contain tongue, count of face emojis contain hand, count of the face neutral sceptical emojis, count of face concerning emojis and count of money face emojis. The experiment was conducted with different ML algorithms such as RF, Logistic Regression, Multinomial NB, Bernoulli NB, Support Vector Classifier (SVC), and Linear SVC for bot detection. They achieved accuracy scores of 0.97 and 0.92 for bot detection on the training dataset and test dataset respectively.

Lee, et al., [10] introduced a strategy to filter content polluters like bots content by using social honeypots. They gathered polluters' content of 23,869 accounts by creating a set of honeypots. The experiment was conducted with a tree-based classifier, SVM, Logistic Regression (LR), and naive Bayes (NB) to distinguish content polluters from legitimate users. The Random Forest classifier generated the highest accuracy of 98.42% when compared with other classifiers. Andrea et al., used [11] an ensemble architecture to solve the task of Bot Detection. The proposed model achieved an accuracy of over the 90% in the bot detection task. They experimented with a set of features such as count of semicolons, count of emojis, count hashtags, count of web links, tweets length, retweets length, cosine similarity score and text distortion. Principle Component Analysis (PCA) is used to reduce the dimensionality of the feature from 1009 to 56 features for bot detection. The ensemble AdaBoost classifier is used for bot classification.

Flóra et al. [12], proposed a Machine Learning approach to detect whether the Twitter user is a human or bot. LR classifier is used to recognize whether the Twitter author is a human or bot. the classification system contains two regression classifiers, one for predicting bot/human at tweet level and the other one is for prediction bot/human at author level. Different types of features are extracted at tweet level and aggregate features on the author level. They experimented with three types of features such as dictionary-based features, stylistic features and meta-features. Dictionary-based features include count of emojis, count of stopwords, count of misspelled words, sentiment score. Stylistic features include lexical diversity, POS tagged features like proportion counts of verbs, nouns and adjectives, text characteristics like proportion counts of apostrophes, uppercase letters, numbers, points, commas, punctuation marks, words, letters, number of times character flooding is used and sentence length. Meta features include counts of calls, retweets and links. The proposed approach is able to achieve an accuracy of 79% and 91.36% for bot/human detection on development and test dataset respectively. They observed that the proposed approach performance is good for differentiating the tweets of human or bots on the test dataset.

Alarifi, et al. [13], collected labeled datasets manually from accounts of Twitter. This dataset contains data of humans, bots, and hybrid posts. Hybrid posts are tweet messages posted by both bots and humans. The same dataset was used by the classifiers for training and testing purposes. The Random Forest and Naïve Bayes classifiers show the best performance for bot/human detection as well as bot/human/hybrid detection. Daniel Yacob Espinosa et al., developed [14] an approach by using character bi-grams. The experiment was conducted with several data cleaning techniques, different machine learning algorithms, and various feature representations for bot/human detection. They applied different pre-processing techniques such as removal of URL's, digits, emoticons, punctuation marks and mentions. They experimented with unigrams, bigrams and trigrams. Different machine learning algorithms such as decision tree variants of J48, RF, Naïve Bayes and SVM. The SVM with 10-fold cross-validation obtained good accuracy of 92.86 for bot/human classification when experimented with character bigrams.

Johan Fernquist experimented [4] with TFIDF features of character and word n-grams, compression features and tweet specific features. The Term Frequency-Inverse Document Frequency (TFIDF) is used to compute the weight of word n-grams (where n range is from 1 to 3). They considered the maximum of 2000 features based on the TFIDF scores of word n-grams. TFIDF scores of character n-grams (n range is 1 to 4) are computed and top 2000 features are identified based on the scores. The experiment was carried out with a total of 4158 components, including 19 features of Compression features, 139 features of tweet features, 2000 word n-grams features based on TFIDF scores, 2000 character n-gram features based on TFIDF scores. The catBoost classification algorithm is used for training with 5000 iterations in their experiment. The proposed system achieved 94.96% accuracy for bot/human classification.

3 DATASET DESCRIPTION

The PAN competition organizers organize competitions on various tasks every year. In 2019, they selected one of the tasks as a bot and gender detection [15]. In this task, the aim is first to classify whether the text was written by a bot or human, once the text was written by a human, then identify whether male or female wrote the text. In this task, they provided a Twitter dataset of 4120 author tweets. Each author file contains 100 tweets. The number of human accounts is 2060, and the number of bot accounts is 2060. Evaluation Measures

4 MACHINE LEARNING ALGORITHMS

In this work, two ML algorithms such as SVM and RF are used for predicting the accuracy of bot/human detection.

4.1 N-Grams Based Approach for Bot/Human Detection

In this work, N-gram based approach is proposed for bot detection. The proposed approach is displayed in figure 1. In this approach, firstly, apply the pre-processing techniques such as removal of hashtags, @mentions, re-tweets, stop-word removal and punctuation marks to remove unnecessary information from the dataset. Once the dataset is cleaned, extract the word N-Grams (N range is 1 to 3) and character N-Grams (N range is 2 to 5). It was observed that the number of word and character N-Grams are more in the dataset. To reduce the number of N-Grams, a feature selection technique is used. The feature selection technique identifies the most important

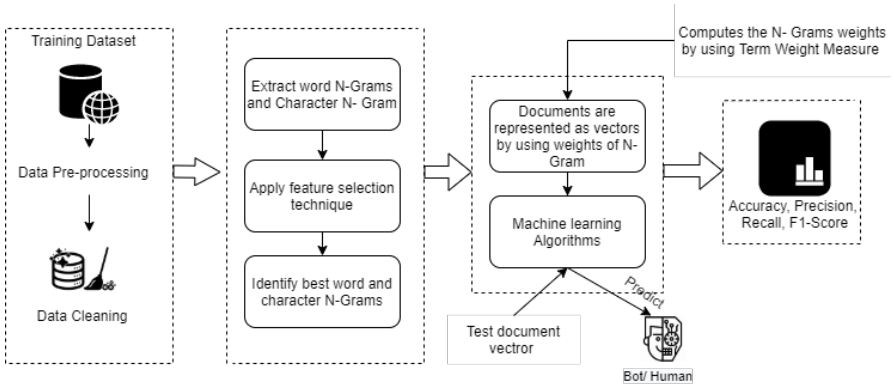

Figure 1: The N-Gram based approach for Bot Detection

features from the whole feature set. After identification of relevant features, represent the samples of the training dataset with these features as vectors. The feature value in vector representation is computed by using a term weight measure. Once the vectors are ready, these vectors are used to train the machine learning algorithms to build the predictive model. This model is used to predict the bot or human of a test vector. In this approach, the feature selection technique and term weight measure play a crucial role to improve the accuracy of bot detection.

4.2 Feature Selection Technique

Feature engineering is a method of developing a feature set to analyze the characteristics of data. It is very difficult to train machine learning algorithms if the number of features in the feature set is high. Dimensionality reduction is a popular technique to transform high-dimensional feature set into lower-dimensional representations [20]. Dimensionality reduction methods are divided into two types such as feature extraction and feature selection. Feature extraction techniques generate a lower-dimensional feature set by combining multiple features in the high-dimensional feature space. Feature selection techniques select a feature subset by computing the ranks of features using criterion measures. Feature selection methods are classified into three methods such as filter methods, wrapper methods, and embedded methods.

Filter methods assign ranks to the features based on a statistical measure that considers different factors of features. These methods are not dependent on any classification algorithm. Wrapper methods select various feature subsets and evaluate these subsets by using classification algorithms. It identifies the best feature subset based on the results of the classification algorithms. Wrapper methods are accurate when compared with filter methods but computationally slower than filter methods [21]. Embedded methods embed the classification algorithms in the identification of feature subset. In this work, a filter-based feature selection technique such as RDC is used to find the informative features.

4.3 Relative Discriminative Criterion (RDC)

Most of the feature selection techniques only consider the distribution of documents in positive and classes and ignore the term frequencies in positive and negative classes of documents to determine the rank of a feature. RDC feature selection technique is proposed to resolve this issue [22]. RDC measure determines the difference among the document frequencies of positive and negative classes which contain the term. RDC technique is an efficient technique in filter-based univariate methods. This technique assumes that the terms that occurred frequently in a particular class have high discriminative power and assign higher ranks to the terms when compared with other terms. Equation (1) is used to determine the RDC of a term.

$$RDC\left(wi, tcj\left(wi\right)\right) = \left(\frac{\left|dfpos\left(wi\right) - dfneg\left(wi\right)\right|}{\min\left(dfpos\left(wi\right), dfneg\left(wi\right)\right) \times tcj\left(wi\right)}\right) \tag{1}$$

Where dfpos(wi) and dfneg(wi) are the document counts of a feature wi in positive class and a negative class of documents respectively. tcj(wi) is the number of times the feature wi occurred in jth class. The feature wi may repeat several times in a specific class of documents.

RDC is an efficient technique that assigns scores to the terms in a class of documents by considering both document counts and term frequencies in positive and negative classes of documents. However, this measure ignores the feature's inter-correlation in the process of evaluation.

4.4 Term Weight Measure (TWM)

The term weight measures determine the importance of a term in a document. In general, the term weight measures consider different types of information like the frequency of a term in a document, the frequency of a term in a positive class and negative class of documents, the number of classes discussed the term, the number of documents in positive class contain term and count of documents in negative class contain the term etc. the term weight measures are classifier into two categories such as Supervised TWM (STWM) and unsupervised TWM (UTWM) based on the utilization of class information of documents. The STWM consider class membership information of documents while calculating the weight of terms. The UTWM doesn't consider the class information of documents when computing the term weight. In this work, the experiment carried out with both supervised and unsupervised term weight measures.

- Term Frequency (TF)
- Term Frequency-Inverse Document Frequency (TFIDF)

The IDF measure allocates more weight to the terms that occurred at least one time in a fewer number of documents [23]. TFIDF is used in several research domains. Equation (7) is used to compute the TFIDF of a term in document irrespective of length of a document.

$$TFIDF\,(Ti, Dk) = \frac{TFik \times \log\left(\frac{N}{DFi}\right)}{\sqrt{\sum_{i=1}^{n}\left(TFik \times \log\left(\frac{N}{DFi}\right)\right)^2}} \quad (2)$$

Where TFik is the frequency of term Ti in document Dk, N is total documents count in dataset and DFi is documents count in dataset that contain term Ti.

- Term Frequency-Relevance Frequency (TF-RF) Measure

Term Frequency-Relevance Frequency (TF-RF) measure computes the weight of a term based on its Relevance Frequency (RF), which is the ratio among the term frequency in the positive class of documents (A) and the term frequency in the negative class of documents (B) [24]. TF-RF measure assigned more weight to the terms which are specific to a class because of A » B. The basic plan of TF-RF measure is the terms that occur in a more positive class of documents when compared with the negative class of documents were more useful to select positive text from a negative text. Equation (8) is used to compute the TFRF value of a term.

$$TFRF\,(Ti, Dk) = TFik * \log\left(2 + \frac{A}{MAX(1, B)}\right) \quad (3)$$

- TF-PROB Term Weight Measure

TF-RF measure includes only the term's inter-class distribution and is represented by A and B. But TF-Prob measure includes the intra-class distribution and inter-class distribution of a given term, represented by A and C. The reason for introducing the intra-class distribution in TF-Prob measure is that the terms which were appeared in most documents in a positive class i.e., A » C obtained good weight to represent the positive class.

- TF-IDF-ICSDF

The TFIDF is popularly used in various text classification problems like authorship analysis, sentiment analysis, information retrieval, etc., to assign weight to the terms. The TFIDF allocates a higher weight to terms which are occurred in less number of documents. Several researchers proposed term weight measures based on the TFIDF measure by replacing IDF with other weight metrics factor-like TFRF (Term Frequency and Relevance Frequency), TFCHI2 (Term Frequency and Chi-square), TFIG (Term Frequency and Information gain), etc. Some researchers enhanced the TFIDF with additional weight factor-like TF-IDF-ICF, TF-IDF-ICSDF, etc. to increase the performance of the term weight process. In this work, the TF-IDF-ICSDF measure is used to determine the weight of a term.

5 EXPERIMENTAL RESULTS

In this work, n-gram based approach is proposed for bot/human detection. In this approach, RDC feature selection technique is used to reduce the number of character and word n-grams and to identify the best informative n-grams as features. We identified the top 8000 n-grams as the best informative features. It was observed that the accuracy is dropped and accuracy is not changed when experimented with 8000+ features. The identified features are used to represent the bot/human profiles as vectors. The feature value in vector representation is determined with different TWM's such as TF, IF-IDF, TF-RF, TF-PROB and TF-IDF-ICSDF. The dataset's description is presented in table 1.

Table 1: The description of the dataset

Statistics	Counts
Train Dataset	2880
Dev Dataset	1240
Total Profiles	4120
No. of Humans	2060
No. of Bots	2060
Maximum Number of Characters in a tweet	933
Minimum Number of Characters in a tweet	1
Average No. of tweets per bot user	100
Average No. of tweets per human user	100

The profile vectors are used to train the classification techniques such as SVM and RF. The experiment start with 1000 features and increased by 1000 in every iteration. The accuracies of bot/human detection when experimented with SVM are represented in table 2.

In Table 2, the TF-IDF-ICSDF measure achieved an accuracy of 0.9278 for bot/human prediction when experimented with the most frequent 8000 character and word n-grams. It was observed in TF-IDF-ICF results, the accuracies are increased when the number of features is increased to represent the document vectors. The TF-PROB, TF-RF, TF-IDF and TF measures achieved accuracies of 0.8841, 0.8623, 0.8043 and 0.7320 when experimented with most frequent character and word n-grams of 7000, 8000, 7000, and 6000 respectively.

In Table 2, the TF-IDF-ICSDF measure achieved an accuracy of 0.9456 for bot/human prediction when experimented with the most frequent 6000 character and word n-grams. It was observed in TF-IDF-ICF results, the accuracies are increased when the number of features is increased to 6000 to represent the document vectors. After 6000 features, it was observed that the accuracy is dropped. The TF-PROB, TF-RF, TF-IDF, and TF measures achieved accuracies of 0.9147, 0.9027, 0.8233, and 0.7521 when experimented with most frequent character and word n-grams of 7000, 6000, 7000, and 8000 respectively.

6 DISCUSSION OF RESULTS

Table 3 shows the accuracies of bot/human prediction when experimented with different classifiers and various term weight measures.

In table 3, the TF-IDF-ICSDF attained the best accuracy for bot/human prediction when compared with other term weight measures. The RF classifier attained the best accuracy of 0.9456 for bot/human when the TF-IDF-ICSDF measure is used. In table 3, the RF classifier achieved the best accuracy for bot/human detection when compared with the accuracies of the SVM classifier. The SVM classifier attained an accuracy of 0.9278 for bot/human

Table 2: The accuracies of Bot/Human Detection when SVM and RF classifiers are used

Term Weight Measures / N-Grams	SVM					RF				
	TF	TF-IDF	TF-RF	TF-PROB	TF-IDF-ICSDF	TF	TF-IDF	TF-RF	TF-PROB	TF-IDF-ICSDF
1000	0.7008	0.7428	0.8121	0.8477	0.8926	0.7174	0.7911	0.8748	0.8789	0.9156
2000	0.7081	0.7530	0.8176	0.8520	0.8992	0.7228	0.7947	0.8831	0.8856	0.9236
3000	0.7113	0.7633	0.8233	0.8612	0.9023	0.7256	0.8022	0.8863	0.8897	0.9274
4000	0.7216	0.7674	0.8346	0.8653	0.9079	0.7319	0.8076	0.8914	0.8964	0.9321
5000	0.7218	0.7832	0.8394	0.8726	0.9116	0.7382	0.8133	0.8972	0.8991	0.9377
6000	0.7320	0.7984	0.8430	0.8807	0.9147	0.7437	0.8167	0.9027	0.9082	0.9456
7000	0.7286	0.8043	0.8511	0.8841	0.9211	0.7473	0.8233	0.9013	0.9147	0.9433
8000	0.7277	0.8012	0.8623	0.8810	0.9278	0.7521	0.8220	0.8987	0.9123	0.9421

Table 3: The comparison of bot/human prediction accuracies

Algorithm	SVM	RF
TF	0.7286	0.7521
TF-IDF	0.8043	0.8233
TF-RF	0.8623	0.9027
TF-PROB	0.9147	0.9147
TF-IDF-ICSDF	0.9278	0.9456

detection. In most cases, it was observed from the results that the accuracies are increased when the number of features is increased in the experimentation.

7 CONCLUSIONS AND FUTURE WORK

Social bots are computer programs that are created for the purpose of automating the tasks like giving replies automatically, asking questions, adding likes to posts etc. Later, the bots are created for executing malicious activities like posting negative opinions or reviews about a product, person, and service. The bots detection becomes a challenge for the research community to avoid damages for the opinions of users. Different researchers used different stylistic features to identify writing style differences among bot or human writings. In this work, the character and word n-grams are used for experimentation. The number of n-grams features is reduced by using the Relevant Discrimination Criterion feature selection technique. The feature value in the feature vector is determined by using the TF-IDF-ICSDF term weight measure. SVM and RF algorithms are used for classification. The RF attained the best accuracy of 96.56 for bot/human detection than the SVM classifier. In future work, we will plan to identify the best informative stylistic features that are helpful for discriminating the writing styles of bots or humans. We also have a plan to implement deep learning techniques for bot/human detection.

REFERENCES

[1] Ferrara, E., Varol, O., Davis, C., Menczer, F., Flammini, A.: The rise of social bots. Communications of the ACM 59(7), 96–104 (2016)

[2] Zakaria el Hjouji, D. Scott Hunter, N.G.d.M.T.Z.: The impact of bots on opinions in social networks. In: arXiv preprint arXiv:1810.12398 (2018)

[3] Fernquist, J., Kaati, L.: Online monitoring of large events. In: 2019 IEEE International Conference on Intelligence and Security Informatics (ISI). IEEE (2018)

[4] Johan Fernquist, "A Four Feature Types Approach for DetectingBot and Gender of Twitter Users", Notebook for PAN at CLEF 2019

[5] Yang, Z., Wilson, C., Wang, X., Gao, T., Zhao, B.Y., Dai, Y.: Uncovering social network sybils in the wild. ACM Trans. Knowl. Discov. Data 8(1), 2:1–2:29 (Feb 2014).

[6] Davis, C.A., Varol, O., Ferrara, E., Flammini, A., Menczer, F.: Botornot: A system to evaluate social bots. In: Proceedings of the 25th International Conference Companion on World Wide Web. pp. 273–274. WWW '16 Companion, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland (2016), https://doi.org/10.1145/2872518.2889302

[7] Dickerson, J.P., Kagan, V., Subrahmanian, V.S.: Using sentiment to detect bots on twitter: Are humans more opinionated than bots? In: Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. pp. 620–627. ASONAM '14, IEEE Press, Piscataway, NJ, USA (2014), http://dl.acm.org/citation.cfm?id=3191835.3191957

[8] Chu, Z., Gianvecchio, S., Wang, H., Jajodia, S.: Detecting automation of twitter accounts: Are you a human, bot, or cyborg? IEEE Trans. Dependable Secur. Comput. 9(6), 811–824 (Nov 2012), http://dx.doi.org/10.1109/TDSC.2012.75

[9] Shaina Ashraf, Omer Javed, Muhammad Adeel, Haider Ali Rao Muhammad Adeel Nawab, "Bots and Gender Prediction Using Language Independent Stylometry-Based Approach", Notebook for PAN at CLEF 2019

[10] Lee, K., Eoff, B.D., Caverlee, J.: Seven months with the devils: a long-term study of content polluters on twitter. In: In AAAI Int'l Conference on Weblogs and Social Media (ICWSM (2011)

[11] Andrea Bacciu, Massimo La Morgia, Alessandro Mei, Eugenio Nerio Nemmi, Valerio Neri, and Julinda Stefa, "Bot and Gender Detection of Twitter Accounts Using Distortion and LSA", Notebook for PAN at CLEF 2019

[12] Flóra Bolonyai, Jakab Buda, Eszter Katona, "Bot Or Not: A Two-Level Approach In Author Profiling", Notebook for PAN at CLEF 2019

[13] Alarifi, A., Alsaleh, M., Al-Salman, A.: Twitter turing test. Inf. Sci. 372(C), 332–346 (Dec 2016), https://doi.org/10.1016/j.ins.2016.08.036

[14] Daniel Yacob Espinosa, Helena Gómez-Adorno, and Grigori Sidorov, "Bots and Gender Profiling using Character Bigrams", Notebook for PAN at CLEF 2019

[15] Rangel, F., Rosso, P.: Overview of the 7th Author Profiling Task at PAN 2019: Bots and Gender Profiling. In: Cappellato, L., Ferro, N., Losada, D., Müller, H. (eds.) CLEF 2019 Labs and Workshops, Notebook Papers. CEUR-WS.org (Sep 2019)

[16] A.-Z. Ala'M, A. A. Heidari, M. Habib, H. Faris, I. Aljarah, M. A. Hassonah, Salp chainbased optimization of support vector machines and feature weighting for medical diagnostic information systems, in: Evolutionary Machine Learning Techniques, Springer, 2020, pp. 11–34

[17] C. Cortes, V. Vapnik, Support-vector networks, Machine learning 20 (3) (1995) 273–297.

[18] B. Scholkopf, A. J. Smola, Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT press, 2001.

[19] Breiman L., "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[20] Van Der Maaten, L., Postma, E. and Van den Herik, J., 2009. Dimensionality reduction: a comparative. J Mach Learn Res, 10(66-71), p.13.

[21] Das, S., 2001, June. Filters, wrappers and a boosting-based hybrid for feature selection. In Icml (Vol. 1, pp. 74-81).

[22] Rehman, A., Javed, K., Babri, H. A., & Saeed, M. (2015). Relative discrimination criterion–A novel feature ranking method for text data. Expert Systems with Applications, 42, 3670–3681

[23] G. Salton, A. Wong, C. Yang, A vector space model for automatic indexing, Communications of the ACM 18 (11) (1975) 613- 620.

[24] M. Lan, C. Tan, J. Su, Y. Lu, Supervised and traditional term weighting methods for automatic text categorization, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (4) (2009) 721- 735.

[25] Liu, Y., Loh, H. T., & Sun, A. (2009). Imbalanced text classification: A term weighting approach. Expert Systems with Applications, 36 (1), 690–701. http://doi.org/10.1016/j.eswa.2007.10.042