International Journal of Innovative Computing, Information and Control
Volume **15**, Number **5**, October **2019**

ICIC International ©2019 ISSN 1349-4198
pp. **1629–1639**

DETECTING FRAUDULENT TWITTER PROFILES: A MODEL FOR FRAUD DETECTION IN ONLINE SOCIAL NETWORKS

Marwan Ali Albahar

School of Management Science
Ibn Rushd College
P.O. Box 447, King Abdul Aziz Rd., Abha, Saudi Arabia
marwanalialbahar@gmail.com

Received January 2019; revised May 2019

Abstract. *This paper presents an approach to detecting fraudulent user profiles on online social networks. The idea is to build a model(s) based on all multiple types of data gathered from online social networks. Here we build models on the Twitter dataset, where we compose data, such as user account information, user connections, and the content that the users are producing. This enables us to create models that are robust, almost data type and content independent. Combining multiple models for the same purpose facilitated the creation of a final model that is self-reliant and adaptable to different data. The results are approximately perfect and are very efficient in terms of detecting fraudulent user profiles. Consequently, this model has the capacity to detect fake or spam user profiles, which can lead to cleaner Internet space.*
Keywords: Fraud profiles detection, Online fraud detection, Social media fraud, Fake profiles, Spam profiles

1. **Introduction.** Online social networks are a modern-day phenomenon. As such, they offer a mass media influence that extends into all aspects of society. Often, they offer free and easy signup and usage. Mechanisms put in place to safeguard the privacy of the users are easily bypassed. These properties make online social networks a target for the compromising behavior of some users, many of which engage in fraudulent and/or malicious behavior while hiding behind false, or fabricated, profiles. Such individuals know how to create public opinion and to distort information, thereby damaging reputations, promoting sales with faked reviews and misinforming network members about public policies or current events. At times, the impact is startling due to fake bot profiles, mass spam campaigns and the dissemination of fake news, as well as the theft of personal data and leaked confidential information. This approach is often used in politics to distort public sentiments or sway voters one way or the other. Also, it can post severe security issues. The mechanisms of many social media websites for verifying the identities of users are limited; they fail to detect a great many fraudulent accounts.

These malicious behaviors can be divided into three classes. Social engineering attacks, which are, in general, represented by spamming and phishing. In spamming, the attacker sends emails, comments, and posts suspicious content, while in phishing, the attacker is posting links to copies of malicious websites. The goal is to get users' private information which, of course, can later be exploited at the owner's expense.

The second class is comprised of social network account attacks, which begins with the hacking of a user's password but shares the same goal. The third class is a malware attack, where the social networks are the victims of malware propagation. So, because

DOI: 10.24507/ijicic.15.05.1629

of that, now, more than ever, detecting and constraining such behavior are crucial for online social networks to be a safe place for networking. Most online social networks have a considerable number of accounts that generate a mass load of data daily. As a result, the process of identifying and stopping malicious users or usage must be automated. Fraudsters often outgrow existing countermeasure mechanisms, and they often find new ways of penetrating the social network. There are a lot of examples of systems that are trying to solve these kinds of problems with accurate results.

In this paper, we propose a robust modular system for fraud detection based on different types of data. We created three different models for disparate types of data extracted from social media. Every model is presented by a combination of several approaches. This makes our approach independent from the type of data and the roughness of one model. This approach is also easily adaptable to other problems.

This paper is organized in the following way. Section 2 presents related works, where we analyze previously proposed system, methods, and approaches for solving the aforementioned problems or threats of a similar nature. In Section 3, we explain our data in detail, and the method by which the models were created. In the same section, we present the results of the proposed model and sub-models. In Section 4, we compare our results to different methods. In Section 5, we conclude the paper.

2. **Related Work.** In this section, we present some of the models or approaches to detecting fraud in online social media. In general, there are three different types of data representation and, thus, we can categorize the proposed solutions as three different approaches.

The first is to see a data as raw data, which means creating a vector from the data and using the appropriate methods for analysis. The second is to represent the data as graph structures and to use graph theory for the analysis of that data. The third is to use natural language processing (NLP) techniques to analyze the free text data.

As representative of the first approach, Perdana et al. attempted to find spambots on Twitter [1]. Based on the success of Bayesian spam email filters, they applied the Bayesian classifier to calculating the probability of spam activity for each user's behavior. They also examined different classification methods like k-nearest neighbors, neural networks, decision trees and support vector machines (SVM). However, they reported that Bayesian networks have the best performance. There are also many other approaches where scientists have observed social networks as graphs [2,3,5,12,13]. Many scientists have analyzed the fraudulent activity and interaction among users by the representation of their relationships as graphs (friendship graph, interaction graph, latent graph, following graph, etc.). These approaches work by finding local communities which are represented by a clustering of nodes around a valid node; the nodes representing the communities are more tightly connected than the rest of the graph. Also, several authors have inspected traffic activity from a network perspective (monitoring traffic, locality of interest, navigation characteristic) [4,14,15]. Another approach to discovering fraudulent activity is to look for spam features in the content posted by users. Other authors looked for textual posts that contained URLs or web addresses [5]. They created graph structures where each post is a node and the edge forms if those nodes contained the same URL, or if they contained very similar text, which is the equivalent to textual fingerprint [5]. Another type of approach is the use of principal component analysis (PCA) to model normal user behavior [6]. Some authors analyzed Facebook activities in comparison to the model of normal user behavior; users whose behavior did not fit the model were flagged as anomalous. One group in [8] applied a deep learning technique, known as a convolutional neural network (CNN), on two different datasets: SMS and Twitter. Another group of researchers in [9]

made a comparison between multiple well-known techniques, including a Facebook posts dataset. Similar to the works in [8,9], the authors in [10,17,18] presented a comparison between multiple articles on different datasets on spam detection in social networks. The authors of the study in [18] introduced an attribute selection methodology to improve the classification by finding a smaller subset of the attributes. In [7], the authors presented a comprehensive analysis of several risks pertaining to security and privacy, which are the most common security threats to online social network (OSN) users. In [16], the authors discussed spam detection in e-mail communication. They proposed a method, which utilized the term space partition (TSP) approach, where several vector subspaces were created, and this method was extended by other techniques, like sliding window. Noticeably, these approaches facilitated the use of local and global classifiers based on different feature vectors. In this study, we build a system, which is on a higher level. Our system encapsulates all the ideas for analyzing social network behavior and uses the best of them. In addition, the proposed system is more robust and adaptable to different datasets.

3. **Data, Methods and Results.**

3.1. **Data.** We used data collected by the authors of "Seven Months with the Devils: A Long-Term Study of Content Polluters on Twitter" [19]. The dataset contains 41,499 different users and more than 5.5 million tweets. From that 22,223 users and more than 2.3 million tweets are marked as spammers, and 19,276 and 3.2 million tweets are marked as legitimate. The data is collected in 7 months. The data is separated into six different files that contain tweets content, user information and user followings for legitimate and fraudulent users. The structure of the dataset is shown in Table 1.

3.2. **Methods and results.** The data is structured in three different parts of a dataset, so we are using three different approaches. The first one is based on data, which contains the followings: the number of followers, number of tweets, the length of the screen name and the length of the description in user profiles. The second one receives the timeline of several followings. This is not the real timeline because we do not have information about when the number is collected; we only possess data on how the number of followings is changing. Finally, the third one is for data, which contain the actual tweets.

For the first part of the dataset, we did data pre-processing. The dates were represented in comma separated values (CSV) format and structured as a vector so that one row fits one user data vector. We removed unnecessary columns such as ID and added a class attribute with fixed values (0 for fraudulent profiles and 1 for genuine profiles). After that we made data standardization and data shuffling. Because we have 8,300 data instances, we decided to split the data set between training and test such that we used 80% for training and 20% for test. We also tried the cross-validation, but the results were pretty much the same. For model building, we used several classifications, such as nearest neighbors, SVM, decision trees, random forest, neural networks, AdaBoost (short for adaptive boosting), Naive Bayes, and qualitative data analysis (QDA), with several different parameter values. Experimentally, we chose the best three (decision tree, random forest, and AdaBoost) and created a voting model. The model is shown in Figure 1.

This approach protects us from relying on one classifier. If one classifier fails, the other two could succeed. The results of this model are shown in Table 2.

The second part of the dataset contains a series of the number of followings. One of the assumptions is that a rapid increase in the number of followings can be considered an indication of fraud profile. That is because the fraudulent profiles are building their followings group to get followers back and that following process is done in several chunks

TABLE 1. Dataset structure

Filename	Description
Fraud user profiles	
Content_poluters.txt	This file contains information about user ID, the date when the account is created and collected, number of followings and followers, number of tweets and the length of the screen name and description in the user profile. The file structure is csv.
Content_polluters_followings.txt	This file contains some kind of time series about the number of following for a certain user. There is user ID as a first record in the row, followed by number of following separated by coma.
Content_polluters_tweets.txt	This file is csv formatted and contains four attributes. The user ID, the tweet ID, the content of the tweet and the day when the tweet is created.
Legitimate user profiles	
Legitimate_users.txt	This file has the same structure like the content_poluters.txt. The main difference is that this file contains information about legitimate users.
Legitimate_users_followings.txt	This file contains the number of followings on the legitimate users. The have the same structure like content_polluters_followings.txt.
Legitimate_users_tweets.txt	This file has the same format like content_polluters_tweets.txt but with information about legitimate users.

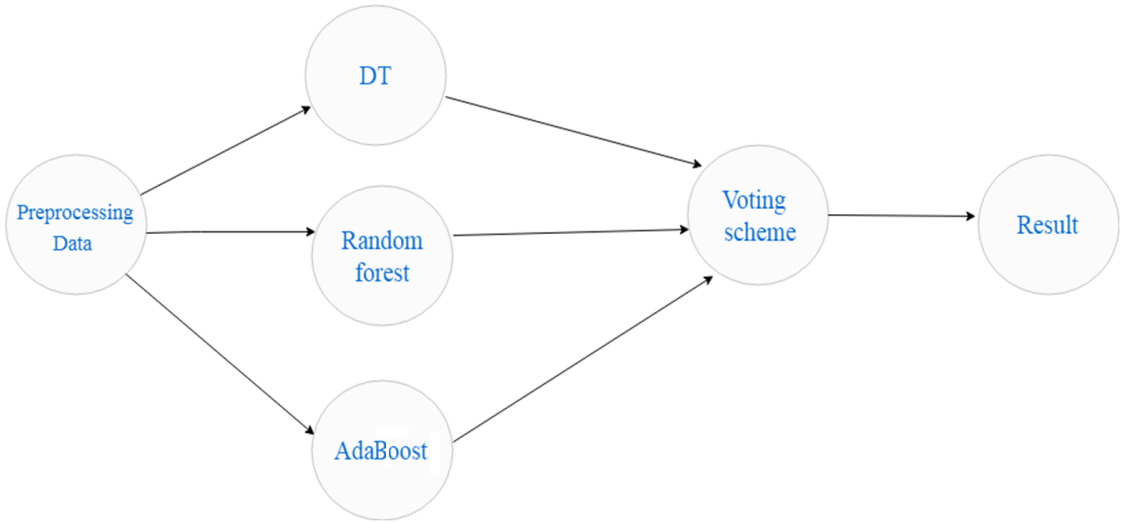


FIGURE 1. Model 1

TABLE 2. Model 1 result

Class	Precision	Recall	F1-score	Support
0	0.96	0.96	0.96	4462
1	0.95	0.94	0.94	3838
Avg/total	0.95	0.95	0.95	8300

with a huge number of accounts. On the other hand, the legitimate profiles usually build their followings list day by day with the smooth number increasing. The main problem in our data is a lack of information about the time when the data is collected, and there are different lengths of the series for different users. Because the data cannot be treated as a regular data series, we decided to make data conversion, and create vectors based on that data. So for every row, not depending on the length of the series, we created a vector with information about that series. We included the descriptive statistic information and the information about the rapid changes. Then, based on the transformed data, we built classifiers.

In the second model, we used an approach like that of Model 1. At first, we tested the preprocessed dataset with several classifiers, and then we put the best three of them in the voting model. The idea is the same as in Model 1: we tried to avoid tying up with one model and being strictly dependent on it. For this kind of dataset, the best performance, experimentally, is the nearest neighbors, random forest and AdaBoost. The model is shown in Figure 2.

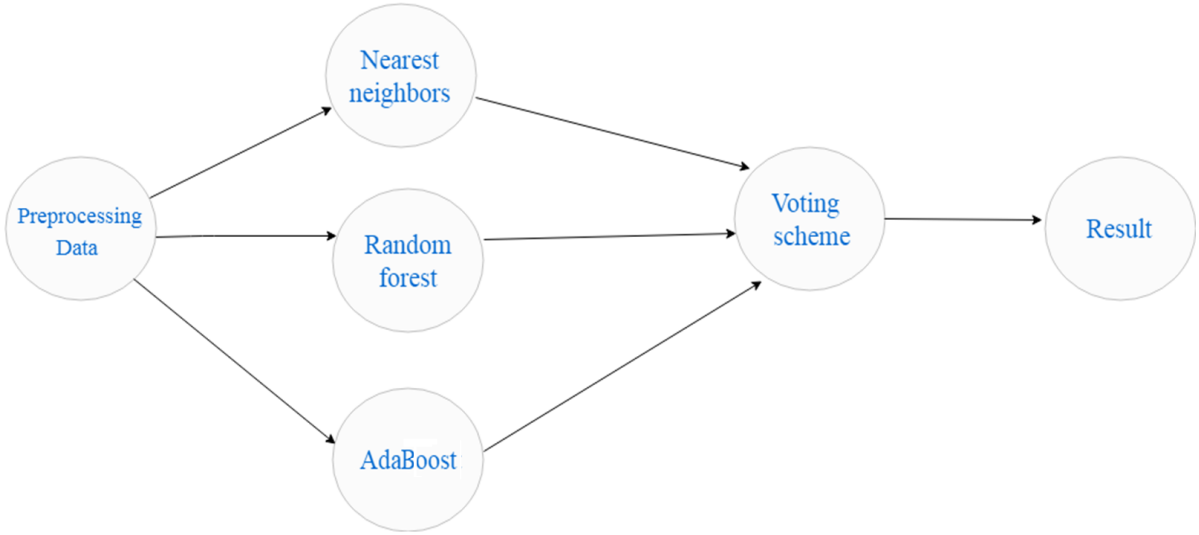


FIGURE 2. Model 2

For the evaluation of Model 2, percentage splitting is 80% used for training and 20% for testing. Also, we tried the cross-validation and the results were the same. The results from Model 2 are shown in Table 3.

TABLE 3. Model 2 results

Class	Precision	Recall	F1-score	Support
0	0.97	0.94	0.96	4488
1	0.96	0.97	0.97	3812
Avg/total	0.97	0.96	0.97	8300

Last but not the least, the third part of the dataset contains the user ID, twitter message (actual tweet) and timestamp. This may be the most important part of the data set because it contains the content of the tweet and its relations to other users via mentioning. For the processing of this part, we use some NLP techniques to create a vector model from the text. Also, we use some specific information for the fraud profile tweets, such as link usages and user tagging. Furthermore, we create a graph to analyze the connections between the users, based on accounts motioning in the post.

TABLE 4. Examples of twitter messages content

Twitter messages content examples
@oneismusic there gonna b on cpt time so closer to 9. Lol I'll come by!
Gostaria de ter milhares de seguidores? Cada vez que usar http://bit.ly/trmass vai ganhar 50 na hora. E participar de sorteios #txatatxa
二時だと結構起きてるのねｗｗｗｗｗｗ
@NaomyNakamura aahhhh ta com saudade... vem pra casa vem!! saudade tbm!! :-$
meu deus minha cidade no verão é um #forno e no inverno um #freezer ;s
Bln.de: Umfrage: Schulessen lässt zu wünschen übrig: Berlin - Zu wenig Gemüse, Obst und Fisch, zu viel S.. http://bit.ly/3L4SC9
ひっさびさに血尿です
@shimax831 わかってはいるんですが…あのきゅいーんが…ううっ(´; ω; `)\u3000 しま子さんがんばって！＞＜
Don't let the pharmacy companies beat you. Buy Tamiflu online for 17 USD http://healthsalexxl.com//pill/Tamiflu?ref_id=4051

TABLE 5. Number of tweets with a certain number of users mentions

No. of mentions	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Fraud users	1925056	346527	36891	7817	3809	3533	2108	2317	2224
Legitimate users	1761444	1292548	137056	29793	9435	4351	2711	2160	2057

Table 4 shows some of the contents of the tweets. The main problem here is that the tweets are in multiple languages, so it is hard to use the more advanced NLP techniques for analyzing the twitter message content. There are 3,246,377 different tweets.

Based on the assumption that the fraud twitter user is using a lot of mentions, we first analyzed the number of mentions in the fraud users tweets and the legitimate users. However, the distribution was almost the same, and no conclusion could be made based on that part of the data. There are 2,333,691 tweets from fraud users and 3,246,377 tweets from legitimate users. Table 5 represents the number of tweets with between no (zero) and eight mentions. As we can see from Table 5, the percentage of the fraud users that do not have any mentions is 82%, while 15% had one mention, and 2% had two mentions. The other values are negligible. We have a very similar situation with legitimate users, where 54% had no mentions, 40% had one mention, and just 5% had two mentions. The other remaining values are below 1%.

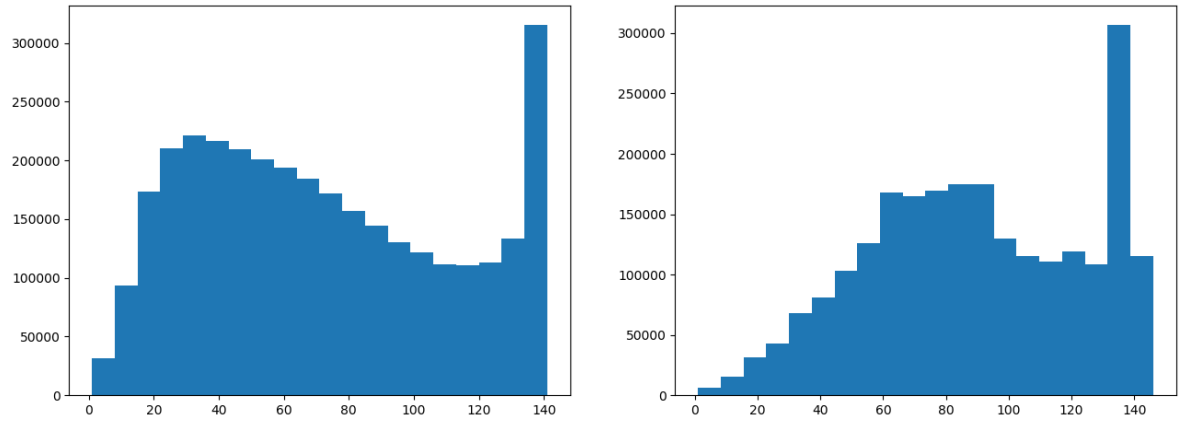
We also created a graph based on mentions but, because of the small number of mentions and the huge number of different users mentioned in the tweets, there were no specific structures like strong and weak connected components, and groups. The condition was a little bit different when we tried to analyze the number of links in the fraud tweets against the legitimate tweets. The numbers are presented in Table 6. The percentage of the legitimate users' tweets without any link is 79%, while the fraud users' profiles without links is 31%; 67% of fraud users' profiles have one link. This information can be considered for the classification.

Figure 3 shows the histograms for the number of characters in the tweets by legitimate and fraudulent users. Figure 4 shows the histograms for the number of words per tweet. We can see that the fraud tweets usually consist of five to twenty-five words.

Because 98 different languages were detected in the tweets, we made some counts of tweets' languages. Figure 5 presents the percentages of the languages in the dataset.

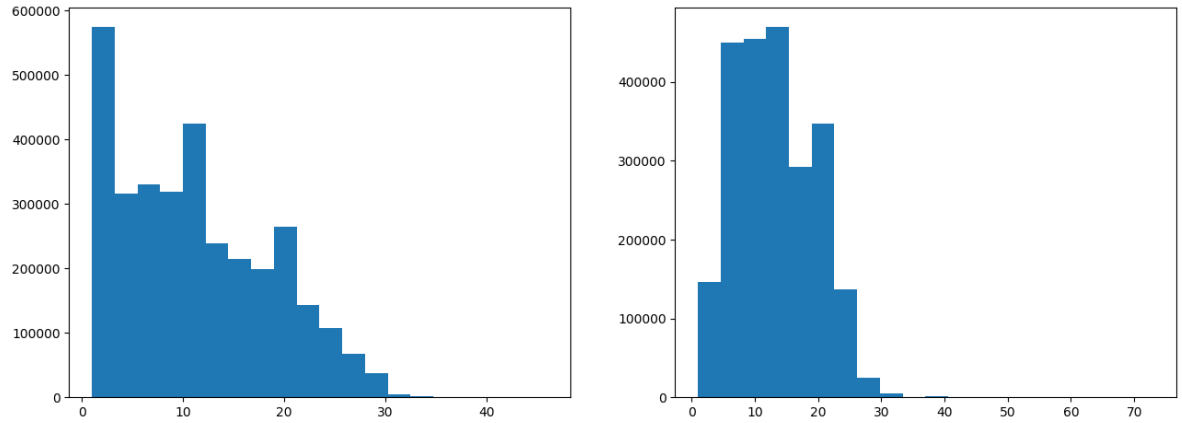
TABLE 6. Number of tweets with different number of links in tweets by legitimate and fraudulent users

No. of links	**0**	**1**	**2**	**3**	**4**
Fraud users	727398	1569515	34347	1678	284
Legitimate users	2577232	651738	16170	885	171



(a) (b)

FIGURE 3. Histograms for the number of characters in tweets of legitimate users (a) and fraudulent users (b)



(a) (b)

FIGURE 4. Histograms for the number of words in tweets of legitimate users (a) and fraudulent users (b)

Almost 70% of the tweets are written in English. After English, Japanese and Portuguese are represented with 6% and 5%, respectively. Then Spanish and German with 3% and 2%, and with less than 2%, but more than 0.8%, are the following languages: Indonesian, Dutch, French, Italian and Malay. All other languages have less than 0.5%, and all together they make up 7.5% of the tweets. Creating an NLP model that spans all the various languages made little sense. As a result, we only extracted the tweets in English. It is noteworthy, however, that the same approach can be used to build models for all

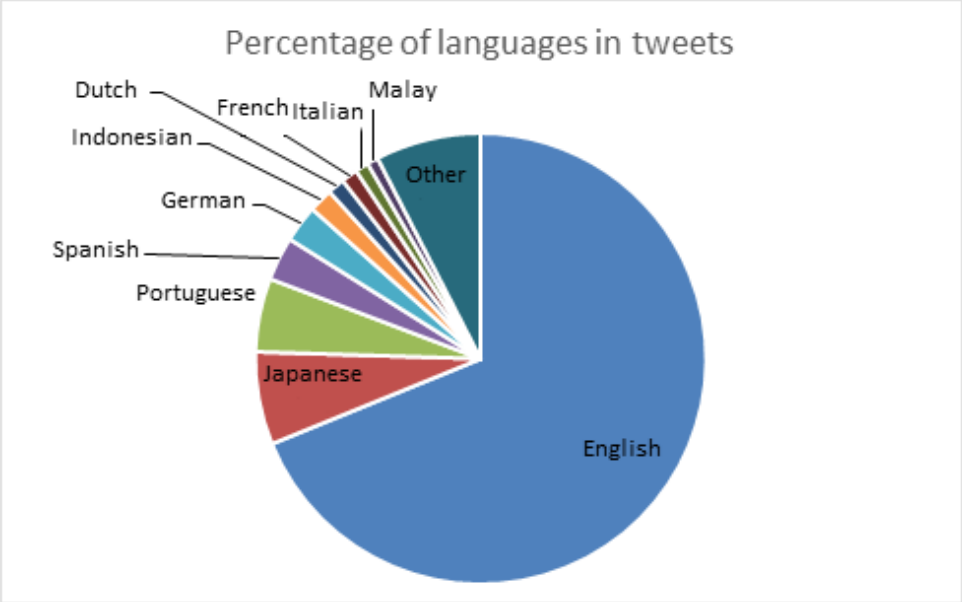


FIGURE 5. Percentage of languages of tweets

TABLE 7. Model 3 results

Class	Precision	Recall	F1-score
0	0.95	0.93	0.94
1	0.94	0.94	0.94
Avg/total	0.95	0.94	0.94

other languages. So then, after language detection, the tweet transfers to the appropriate model.

Third, the NLP model is created in several steps. In the first step, the text is extracted and some preprocessing techniques, like tokenization, capitalization, stop words removal and stemming are applied. After that, a term frequency-inverse document frequency (TF-IDF) matrix is created for all tweets. The Naïve Bayes model based on these vectors is created. Similarly, the dataset is split into train and test data with ratio 80%-20%. The results from this part are presented in Table 7.

The entire architecture of the proposed system is presented in Figure 6. For different types of data, we built different models and then aggregated the output of the models in one voting scheme. In our voting scheme, all voters have the same right to vote. We have three electors, and this is a problem of binary classification, so one of the classes will always have most of the voters. The object is classified in that class that has more votes from the models. The output can be presented as a result of the voting, not just 0 or 1. Thus, we know the object gets two votes, or all three of them, which means that it is classified by all models in that class. We can also use the weighted approach, which means that we will assign weight to the model results. Unlike the previous approach, where all the voters have the same weight, the vote of one model can be considered more significant than the vote of another. In this way, we can play with the weight of the votes on the models until we get the optimum result.

Figure 7 presents the receiver operating curve (ROC), the area under the curve (AUC) curve (left), and the reliability curve (right) for the proposed model. The area under the curve is 0.987. Also, the final model results are presented in Table 8.

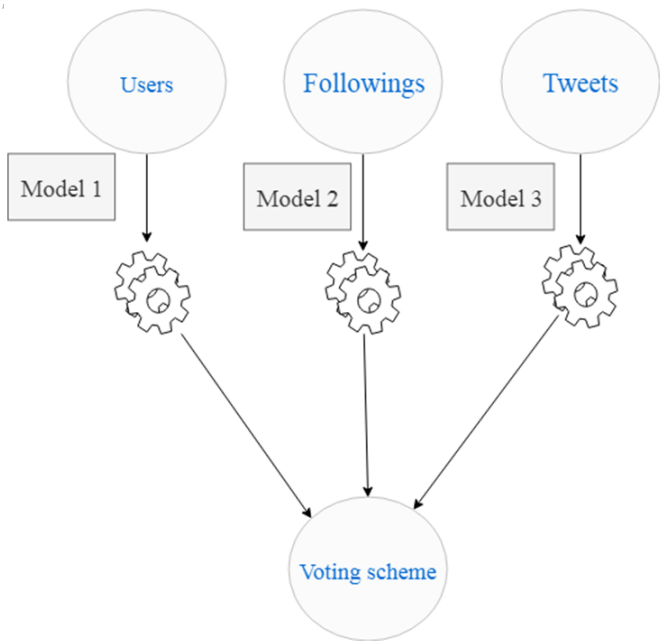
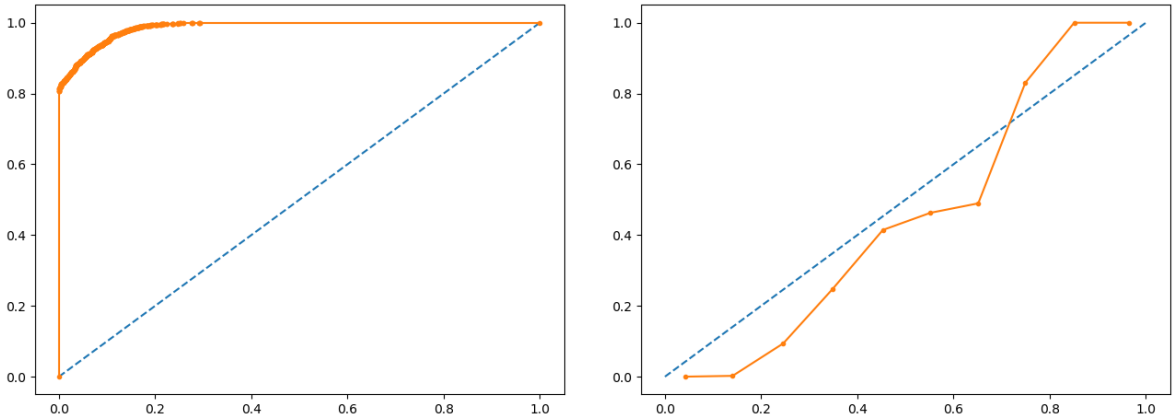


FIGURE 6. Representation of the entire system architecture



(a) (b)

FIGURE 7. ROC curve (a), the reliability curve (b) of the proposed model

TABLE 8. Final model results

Class	Precision	Recall	F1-score
0	0.98	0.98	0.98
1	0.99	0.98	0.99
Avg/total	0.99	0.98	0.99

4. **Comparison to Other Methods.** It is very hard to make a comparison between the results of different models, especially when they are built and evaluated on different datasets. Moreover, almost all the systems that are solving this kind of problem are working on nonpublic datasets that are collected from the social networks by the authors. So we cannot make a future analysis of the data that they are using to compare with our data and to extract better conclusions. Still, we try to make some comparison between

our system and some of the latest state-of-the-art systems used for the same purpose (see Table 9). All of them have different approaches, such as the k-nearest neighbors, CNN, and graph analysis, and all of them are using different datasets. What these approaches have in common, however, is that they are using social networks datasets, and they are trying to solve the fraud detection problem. In other words, we all share the same goal. If we are making some comparison based on the accuracy of the system, we can easily conclude that all of them are achieving more than 95%. We are achieving a very similar accuracy in our system but, as we said, we could not conclude that one system is superior over the others based on only one value. Furthermore, such high accuracy may mean model overfitting. The main advantage and novelty of our system compared to the other recent systems, is that we do not rely on one model. The possibility of replacing and adding more models to our system makes it almost independent of the dataset. Additionally, the fact that the system, itself, is built on different information and structure of datasets, makes it very easy to adjust.

TABLE 9. Comparison of the proposed model with previous methods

Ref	**Method**	**Dataset (source)**	**Accuracy (%)**
[8]	CNN	SMS (UCI)	98.65
		Twitter (scrapping)	94.40
[9]	LMT	Facebook (Max Planck)	98.71
[11]	K-NN	Twitter (kaggle)	94.7
[17]	Random Forest	Twitter (scrapping)	97.3
		Facebook (scrapping)	94.7
[18]	Naïve Bayes	Twitter (scrapping)	84.4
This paper	**Robust voting models**	**Twitter (scrapping)**	**98.93**

5. **Conclusion.** The model that we propose contains three different models which handle three different datasets combined in one. Every one of the datasets is composed of three different classifiers. The architecture of the proposed model is better than others because it offers a high flexibility and easily adapts to a new problem domain, as well as other datasets. Therefore, it can easily outperform different architectures with just a little adoption on the problem domain and the dataset. According to the results from the methods, which are based on the previously explained dataset, we can conclude that this approach is working well. There are several advantages to this system. The main advantage is that the system is robust and can be adjusted to different types of data, just as the dataset that we use. This means that you may not have all the data from the user, like the user's name, bio, and number of followers, the changes in the number of followings or the actual users' tweets to make fraud classification. The more data, the more accurate the results. However, you can base a decision on just a part of the data. The second advantage is the voting scheme. All the models are composed of multiple sub-models, and that enables us not to rely on only one model. So, the fraud detection is made by various sub-models. The addition and replacement of the sub-models are not only possible but can be done in a straightforward way. The third advantage is that all the models are modular and easily updateable. New classified data can be adapted easily. Should some of the sub-models fail to yield good results, they can be replaced by more accurate models. This system is suitable for purposes other than the Twitter dataset. It can be used for any social network that has user connections and textual data. Therefore, in this article, we built a very precise system for fraud detection which can be used in different environments.

REFERENCES

[1] R. S. Perdana, T. H. Muliawati and R. Alexandro, Bot spammer detection in twitter using tweet similarity and time interval entropy, *Jurnal Ilmu Komputer dan Informasi*, vol.8, no.1, pp.19-25, 2015.

[2] S. Y. Bhat and M. Abulaish, OCTracker: A density-based framework for tracking the evolution of overlapping communities in OSNs, *IEEE/ACM Int'l Conf. Advances in Social Networks Analysis and Mining*, pp.501-555, 2012.

[3] Q. Cao, X. Yang, J. Yu and C. Palow, Uncovering large groups of active malicious accounts in online social networks, *Proc. of the 2014 ACM SIGSAC Conf. Computer and Commun. Security*, pp.477-488, 2014.

[4] L. Jin, Y. Chen, T. Wang, P. Hui and A. V. Vasilakos, Understanding user behavior in online social networks: A survey, *IEEE Commun. Magazine*, vol.51, no.9, pp.144-150, 2013.

[5] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen and B. Y. Zhao, Detecting and characterizing social spam campaigns, *Proc. of the 10th ACM SIGCOMM Conf. Internet Measurement*, pp.35-47, 2010.

[6] B. Viswanath, M. A. Bashir, M. Crovella, S. Guha, K. P. Gummadi, B. Krishnamurthy and A. Mislove, Towards detecting anomalous user behavior in online social networks, *Proc. of the 23rd USENIX Security Symp.*, pp.223-238, 2014.

[7] M. Fire, R. Goldschmidt and Y. Elovici, Online social networks: Threats and solutions, *IEEE Communications Surveys & Tutorials*, vol.16, no.4, pp.2019-2036, 2014.

[8] G. Jain, M. Sharma and B. Agarwal, Spam detection on social media using semantic convolutional neural network, *Int'l J. Knowledge Discovery in Bioinformatics (IJKDB)*, vol.8, no.1, pp.12-26, 2018.

[9] M. N. Sarpiri, T. J. Gandomani, M. Teymourzadeh and A. Motamedi, A hybrid method for spammer detection in social networks by analyzing graph and user behavior, *J. Computers*, vol.13, no.7, pp.823-829, 2018.

[10] B. A. Kamoru, A. B. Jaafar and M. A. A. Murad, Understanding security threats in spam detection on social networks, *Circulation in Computer Science*, vol.2, no.5, pp.18-22, 2017.

[11] K. Srinivasan and V. Sureka, Profiling online social networks for spam detection, *Int'l J. Scientific Research in Computer Science, Eng. and Information Technology*, vol.2, no.5, 2017.

[12] W. Min, Z. Tang, M. Zhu, Y. Dai, Y. Wei and R. Zhang, Behavior language processing with graph based feature generation, *Proc. of WSDM Workshop on Misinformation and Misbehavior Mining on the Web*, 2018.

[13] A. Beutel, L. Akoglu and C. Faloutsos, Fraud detection through graph-based user behavior modeling, *Proc. of the 22nd ACM SIGSAC Conf. Computer and Commun. Security*, pp.1696-1697, 2015.

[14] X. Wang, Z. Sheng, S. Yang and V. C. M. Leung, Tag-assisted social-aware opportunistic device-to-device sharing for traffic offloading in mobile social networks, *IEEE Wireless Commun.*, vol.23, no.4, pp.60-67, 2016.

[15] Z. Ning, F. Xia, N. Ullah, X. Kong and X. Hu, Vehicular social networks: Enabling smart mobility, *IEEE Commun. Magazine*, vol.55, no.5, pp.16-55, 2017.

[16] Y. Tan, Q. Wang and G. Mi, Ensemble decision for spam detection using term space partition approach, *IEEE Trans. Cybernetics*, pp.1-13, 2018.

[17] H. Xu, W. Sun and A. Javaid, Efficient spam detection across online social networks, *IEEE Int'l Conf. on Big Data Analysis (ICBDA)*, pp.1-6, 2016.

[18] S. Dutta, S. Ghatak, R. Dey, A. K. Das and S. Ghosh, Attribute selection for improving spam classification in online social networks: A rough set theory-based approach, *Social Network Analysis and Mining*, vol.8, no.1, 2018.

[19] K. Lee, B. D. Eoff and J. Caverlee, Seven months with the devils: A long-term study of content polluters on Twitter, *Proc. of the 5th Int'l AAAI Conf. on Weblogs and Social Media*, pp.185-192, 2011.