IJICTR
International Journal of Information & Communication Technology Research

Volume 15- Number 1 – 2023 (35 -44)

Identifying Persian Bots on Twitter; which Feature is More Important: Account Information or Tweet Contents?

Mojtaba Mazoochi*
Information Technology Research Faculty
ICT Research Institute
Tehran, Iran
mazoochi@itrc.ac.ir

Nasrin Asadi
Development and Innovation Center for AI
ICT Research Institute
Tehran, Iran
asadi@itrc.ac.ir

Farzaneh Rahmani
Development and Innovation Center for AI
ICT Research Institute
Tehran, Iran
rahmani@itrc.ac.ir

Leila Rabiei
Development and Innovation Center for AI
ICT Research Institute
Tehran, Iran
l.rabiei@itrc.ac.ir

Received: 5 May 2022 – Revised: 25 August 2022 - Accepted: 27 November 2022

Abstract— **The spread of internet and smartphones in recent years has led to the popularity and easy accessibility of social networks among users. Despite the benefits of these networks, such as ease of interpersonal communication and providing a space for free expression of opinions, they also provide the opportunity for destructive activities such as spreading false information or using fake accounts for fraud intentions. Fake accounts are mainly managed by bots. So, identifying bots and suspending them could very much help to increase the popularity and favorability of social networks. In this paper, we try to identify Persian bots on Twitter. This seems to be a challenging task in view of the problems pertinent to processing colloquial Persian. To this end, a set of features based on user account information and activity of users added to content features of tweets to classify users by several machine learning algorithms like Random Forest, Logistic Regression and SVM. The results of experiments on a dataset of Persian-language users show the proper performance of the proposed methods. It turns out that, achieving a balanced-accuracy of 93.86%, Random Forest is the most accurate classifier among those mentioned above.**

Keywords: social networks; Twitter; bot detection; classification; Persian language

Article type: Research Article

* Corresponding Author

[DOI: 10.61186/itrc.15.1.35]

I. INTRODUCTION

During the last decades the structure of internet has changed completely and online social networks have made it possible to form new communities of users. Perhaps, what makes online social networks favorable is that they allow the users to communicate to each other in the format of groups, never taking care about spatial, temporal, cultural, and economic constraints. Social networks like Facebook, Instagram, Twitter, etc. have gained high popularity among the users. In 2019, Twitter has had about 330 million active monthly users [1].

Social networks are not only a place to connect to friends, but also, due to the free space available, a place to openly express interests and opinions about various subjects. This sense of freeness without having to auto censorship makes these opinions very effective in the real world and analyzing them would be of high importance e.g. to marketing companies, public policies, sociology, etc.

Because of the benefits briefly mentioned above, the ground is also provided for automated and potential malicious activities. In particular, a significant percentage of social users are fake accounts. A Twitter bot is a type of bot software that controls a Twitter account via the Twitter API.[2] The bot account may autonomously perform actions such as tweeting, re-tweeting, liking, following, unfollowing, or direct messaging other accounts. These accounts are usually created and managed by bots through automating some activities of human users [3]. The domain of performance of the bots could be very vast: from creating fake accounts to intentionally influence election results and motivating social riots or strikes, to high jacking one's account information with offensive or personal calumniation purposes or, being optimistic, for marketing uses. Therefore, a systematic attempt has been initiated during recent years that aims at distinguishing between bots and human users active in online social networks. The complexity of bot detection approaches, that generally use machine learning techniques, lies in the fact that, to avoid from being detected by social networks, the bots very often follow each other, mimic the behavior of legal users, and publish regularly a combination of daily tweets and spams [4].

It should be emphasized that most of the research projects in this regard have concentrated on tweets that are published in English and, as to our best knowledge, there is so far no published research article dealing with Persian bot detection. This shortage is the main motivation of the current study.

Our field survey on Persian-language accounts during almost 2 years of analyzing information about Twitter accounts such as login dates, number of followers and followings, number of tweets, shows that accounts with automated or unconventional activities on Twitter can be divided into several categories. Some of them are news bots that publish news as tweets intermittently at short time intervals. The other category would contain accounts that make a large number of retweets, while the third and most important category comprises those accounts, called suspicious ones, with special features such as new Twitter login dates, and close number of followers and followings. Sometimes these accounts conduct a poll to get public flavor concerning a particular issue and perhaps to distort user's opinions based on their predefined policies. Therefore, one of the motivations of the current study is to identify, and label as bot, all aforementioned categories of accounts with unusual activities.

Like many other languages, it is very common to use colloquial Persian in social networks and especially Twitter. As a result, Persian-language bots may publish content in colloquial Persian to mimic the behavior of normal users. Due to its special features and various writing styles, processing Persian language in social networks may face some difficulties. The colloquial style of writing on social networks intensifies the problem of processing the textual content. Sometimes even literature experts encounter difficulties in dealing with the colloquial text of social networks. Abnormal styles such as deleting and changing the order of sentence parts (verb, subject, object, etc.) as in " رفتم خونه ی دوستم" instead of "من به خانه ی دوستم رفتم" (equal to “I went to my friend's house” in English) or abnormal repetition of letters like in "لاااایک" (equal to “like” in English) or using misspelled words such as "حتا" (equal to “even” in English) are often found in social network texts. Proper exposure and processing of these texts requires specific preprocessing tools for each of the social network platforms, because supposedly the way users write on Twitter is different from that on Instagram. On the other hand, intelligent automated pre-processing tools require sufficient amount of data for training, and even though various corpora have been created for the Persian language so far, they are generally scrapped from formal Persian web content [5]. Preparing labeled corpora including part of speech tagging has many challenges which does not fall within the scope of this article. Finally, one of the motivations of this article is to detect bots based on this colloquial text style despite all its processing problems.

In this research, a dataset of Persian-language users and the posts published by them on Twitter has been collected. After annotation, we consider a combination of various features already treated in the literature in conjunction with newly proposed ones to identify bot and human users. Our proposed features, including account information, tweet information and tweet content, will assay the user's behavior from various aspects. We also determine the impact of the aforementioned feature groups on bot detection in Twitter using some well-known classifiers.

This paper is organized as follows: we firstly provided a literature review in Section 2. Section 3 is devoted to presenting the "Proposed Approach" and our experimental results appear in Section 4. The last section will provide the reader with our concluding remarks.

II. RELATED WORK

Botnet refers to a group of online social bots that are organized, managed and scheduled in coordination to each other [4]. Since bots can lead to the spread of incorrect information, identifying them, can effect and

[DOI: 10.61186/itrc.15.1.35]

improve, the performance of social networks. In this paper, we focused on Twitter bots.

A glance at the literature reveals that bots have been classified from different perspectives. Most of the studies in this direction, only deal with how to distinguish between human and bot users [6-7]. However, some researchers have tried to categorize social bots in more details. This is of high importance because some bots are harmless or managed accounts that are not willing to be suspended by social networks [8-9].

Various methods have been proposed to identify bots. These very often use machine learning approaches that may be classified into two categories on its own; namely supervised and unsupervised machine learning methods, the first of which includes traditional classifiers, and deep neural network algorithms.

In supervised methods, bot detection is done through applying a set of features on labeled training data. Among other things, this set usually contains user account information, friends, network and temporal features, and content and sentiment of user tweets [10-13]. Traditional classifier algorithms that have reported the most accurate results include Random Forest, Support Vector Machine, Logistic Regression, etc. [10], [14].

Using discrete wavelet transform (DWT), Igawa et al. [15] proposed an algorithm that would obtain a pattern of writing in the content of tweets of a particular user. They used Random Forest to distinguish between human, legitimate and malicious bot accounts. This work has had several preprocessing steps and therefore the computational complexity has been raised.

In [16], Wei et al. applied LSTM neural network as the classifier and tried to decide whether or not a particular account is a bot, only by considering user account information and one of its tweets. Firstly, they represented the tweet by GloVe pre-trained word vectors [17] and, secondly, added user information and friends to the network as auxiliary features that are intended to increase the classification accuracy.

In unsupervised methods, it is supposed that class labels are not available and, thus, clustering is done based on the similarities between the samples. As a worthwhile work in this direction, we want to mention [18], in which Chavoshi et al. detected suspicious users by analyzing the time series of the tweets, at the first phase. Next, they perform clustering on these users and recognize the so-called singleton users, i.e., those that are left outside the clusters, as false-positives and the others as bots.

Also in [19-20] Cresci et al. tried to use a DNA pattern to model the behavior of the users of the social networks in terms of the sequence of tweets, replies and retweets. Sequences that have been assigned to a group of accounts will then be compared to each other to find anomalous similarities among them. The users with highest similarity in DNA pattern are considered as a botnet.

In addition to improving the performance of social networks and increasing user's satisfaction, bot detection might be useful in various fields of politics, sociology, and economy. As many of political and economic actors are active on Twitter now a days, published comments can influence people's intellectual tendency. Hence identifying bots and suspending them could make the space of opinions more clarified. To point out some more practical experiments, we may mention the papers [21-22]; dealing with fraud bots in Indonesia's 2016 and Russia's 2018 presidential elections, [23], where bot detection in stock markets has been taken into account, and [24-25] as an attempt to identify credulous users, i.e., peoples who have a lot of bot friends and, usually spread false news inside the communities.

III. Proposed Method

In the following, we intend to introduce the collected dataset by giving its particular specifications. We also provide some information on how it has been annotated. The proposed method is then described by introducing selected features and classification algorithms.

A. Basic Idea

The main objective of this paper is to identify Persian-language bots users on Twitter. The proposed approach includes three main phases: In the first phase, we collect and annotate a dataset consisting of Persian-language users and their published tweets. The second phase aims at preprocessing and extracting feature vectors for all users in the dataset. Finally, in the last phase, some classifiers, amongst which we may mention Random Forest, Logistic Regression and Support Vector Machines (SVM), are exploited to classify the users either as bot or human. A comprehensive description of these steps will be presented in the sequel.

B. Dataset

Our dataset in this study contains account information of 755 Persian-language users on Twitter. For a period of more than two months, starting from Dec. 11, 2019 to Feb. 20, 2020, all posts, i.e., tweets, retweets and replies, published by the aforementioned users have been collected using offered APIs. We have then removed 66 users because of no activity in the selected period. Overall, the final dataset comprises 629758 posts published by 689 users. Next, two computer experts were required to annotate the dataset, dividing them into two prescribed classes: bot and human. These annotators are 30 and 35 years old and have been active on Twitter for the past 4 years, so they are quite familiar with the environment of this social network. These two people actually browse Twitter with a predefined help document which they check the profile details for each user based on the help document. For example, if the profile is related to a news agency that has a high number of tweets sent in a short period of time, that user will be labeled as a news bot. Or the time of join date, the number of followers and followings of the user is also considered and based on the help document, if the time of join date is new and the number of followers and followings is high and with close values, the user is considered as a bot. Non-news profiles that have published a large number of tweets or retweets in a short period of time are also considered as

[DOI: 10.61186/itrc.15.1.35]

bots and so on other rules and solutions for human diagnosis of bots. Each annotator examines the user's profile for up to 5 minutes and, if necessary, uses statistical analysis such as the average number of tweets or retweets per day to make decisions. TABLE I. shows the total number of users and posts lying in each class.

As illustrated by **Error! Reference source not found.**, the number of human users is twice that of bot users, while the number of posts published by bots is more than twice compared to those published by humans. The table also shows that the original tweets and replies achieved respectively the highest number of published posts in both user groups.

TABLE I. STATISTICS OF DATASET

	Bot	Human	Total
Number of users	229	460	689
Number of tweets	178183	123585	301768
Number of retweets	21781	27397	48078
Number of replies	203060	76852	279912
Number of posts(all)	430024	199733	629758

C. Text Preprocessing

In this section, we will introduce text preprocessing tools that have been used to extract features from the context of tweets.

1) Persian NLP-preprocessing

Due to the importance of Persian preprocessing in NLP applications, some attempts have been done in recent years to develop integrated Persian preprocessing packages. Amongst these, we may name "Hazm"[26] and "ParsiPardaz" [27] that are almost complete and open source. Hazm includes some major preprocessing tasks such as normalization, tokenization and POS tagging. Besides these tasks, ParsiPardaz further provides morphological analysis and spell checking.

Although Hazm outperforms ParsiPardaz toolkits from the run time point of view, its output results are not as accurate as expected. Moreover, despite having key preprocessing steps, the mentioned toolkits have some drawbacks while applied over colloquial Persian. For example, in these toolkits, no conversion strategy has been considered for expressions like "salam" that shows "سلام" (equal to “hello” in English) in Persian. This writing style, known as Pingilish or Fingilish, is very common among Persian-language users of social networks. Among other issues, substantial inability in normalizing three-part expressions (such as "گفت و گو" equal to “conversation” in English) is another disadvantage of Hazm.

Twitter allows users to send their followers 280 characters per tweet. As a result of this limitation, one of the main specifications of the tweets is the use of abbreviated words and expressions. The use of colloquial terms, and using hashtags to indicate the main purpose of the tweet as well as streaming news are other features of Twitter texts.

In conjunction to correcting spaces and half spaces, the normalization process conducted further corrects the spaces in tree-part expressions, standardizes the expression of time, removes emojis, and deletes links and punctuation marks. Moreover, correcting Pingilish expressions, unifying the display of Arabic letters and separating non-Persian letters are considered. Furthermore, using two dictionaries of colloquial terms, recognizing and unifying colloquial phrases and specific Twitter terms has been done.

2) Persian FastText

Being provided by Facebook [28], FastText is a library for effective word representation with the ability to train words and sentences with and without supervision. It is a sub-word embedding method that uses the morphological information of words and is founded almost over the same ideas as those applied for word2vec. In this model, each target word is represented by a subset of words. For example, for the word 'ایران', if we consider the number of characters to be 3, then the vectors ['ایران', 'ران', 'ایر', 'یرا'] will be constructed. Notice that FastText is implemented at different levels of characters and word characters. TABLE II. exhibits the information about Persian FastText embedding that is trained based on social networks text contents.

D. Features

As pointed out earlier, our dataset includes user account information and the tweets they published within a certain time interval. Based on diversity of the fields provided by the dataset, we introduce three set of features; namely, *account information*, *tweet statistical information*, and *tweet context*.

Let us elaborate a bit more on these feature groups. *Account information* consists of features that have been extracted from user profiles. These include the number of followers and followings, the total number of tweets published by a particular user, the age of user account, etc. *Tweet statistical information* involves the features that have been obtained from user's activity within the period Des. 12, 2019 to Feb 20, 2020. These include the number of user tweets within this timespan, the number of replies posted for user's tweets, the average of tweet length, lexical richness of the tweets, etc. Finally, *tweet context* comprises features, like TF-IDF, that are extracted from the text of the tweets published by users with the aforementioned period of time. TABLE III. lists all the items associated to each feature group.

To extract features from the context of tweets, the following sequence of preprocessing steps are applied:

- Merge tweets: all tweets published by a particular user within the specified period of time are connected to each other to form a document.

[DOI: 10.61186/itrc.15.1.35]

TABLE II. INFORMATION OF WORD EMBEDDING USING FASTTEXT

Number of Words	Run time for each epoch	Dimension	RAM	CPU
115867	13min 9 seconds	100	13GB	1 Core, 2.3GHZ

- Normalization: the document arising from the previous step are normalized using Persian NLP-preprocess toolkit. During normalization process, all Urls found in the documents have been replaced by <URL> tag.
- Tokenization process: Using spaces, words have been separated, and the sentences are determined using punctuation symbols.

While preprocessing steps have been accomplished successfully, our features are extracted from the tweets by using the following approaches:

- TF-IDF: each document is represented using the so-called Term Frequency-Inverse Document Frequency model. In this regard, we experimentally prefer to consider all word n-grams (i.e., 1-grams, 2-grams and 3-grams) in the tweets and, accordingly, 1000 most frequent n-grams are chosen as our TF-IDF features.
- Word embedding: A pre-trained word embedding is created by Persian FastText and, further, the word vectors of the document are extracted. In order to convert the 2-dimentional matrices associated to the word vectors into 1-dimentional ones, we let u_t, $1 \le t \le 689$, denote the t-th user and wv_{ij}, $1 \le j \le d, 1 \le i \le n_t$, be the j-th entry representing the word vector of the word w_i in the document assigned to the user u_t.

  Note that here, n_t stands for the number of the words appearing in the document of user u_t, and $d = 100$ is dimension of pre-trained word embedding. In this notation, set

$$k_{t,j} = \frac{\sum_{i=1}^{n_t} wv_{ij}}{n_t}. \quad (1)$$

  So, $k_t = (k_{t1}, \dots, k_{td})$ will be the feature vector of user u_t.
- Human-bot lexicon: Specified words that are usually used by human and bot users are extracted separately by appealing to the annotated dataset and a ranking algorithm:

  The score $S_{c(w)}$ of a word w in a given class C is computed via

$$S_{c(w)} = \frac{freq_{total}(w)}{freq_C(w)}, \quad (2)$$

  where $freq_{total}(w)$ and $freq_C(w)$ denote the number of occurrences of w in the whole dataset, and in the tweets lying in C respectively [3].

In order to remove rare words, suppose conventionally that $diff_w = freq_C(w) - freq_{\neg C}(w)$ exceeds a prescribed threshold which we set to be 90 throughout the paper. Finally, 250 words of the highest score in each class are considered to form the human-bot lexicon feature.

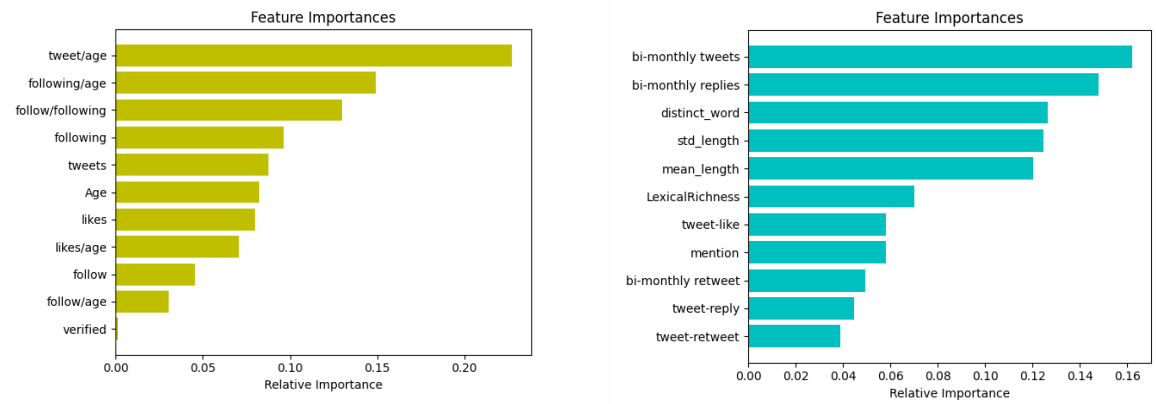
It should be noticed that since 5-fold cross-validation has been used to evaluate the proposed method, "TF-IDF" and "Human-bot lexicon" are calculated, for every repetitions, only based on prominent words in the training data (not the entire dataset). Therefore, the classifiers have no information from the test set while the classification is in progress.

TABLE III. DESCRIPTION OF FEATURES

Feature category	Name	Description
Account information (AI)	Age	Age of account (days)
	Tweets	Total number of tweets posted by user
	Followers	Number of followers
	followings	Number of followings
	Likes	Total number of likes
	Verified	User is/isn't verified by Twitter
	likes/age	likes divided by age
	followers/age	followers divided by age
	followings/age	followings divided by age
	tweets/age	tweets divided by age
	followers/followings	followers divided by followings
Tweet Statistical Information (TSI)	bi-monthly tweets	Total number of tweets posted by user in the period
	bi-monthly retweets	Total number of retweets posted by user in the period
	bi-monthly replies	Total number of replies posted by user in the period
	tweet-reply	Total number of replies for bi-monthly tweets
	tweet-like	Total number of likes for bi-monthly tweets
	tweet-retweet	Total number of retweets for bi-monthly tweets
	mean-length	Average length of tweets (characters)
	std-length	Standard deviation of length of tweets
	distinct_word	Lexical richness of tweets (unique words in user bi-monthly tweets divided by all words used in user bi-monthly tweets)
	mention	Number of users that are mentioned by a user

[DOI: 10.61186/itrc.15.1.35]

Tweet context (TC)	URL	Existence of URL in tweets
	TF-IDF	TF-IDF vector for tweet context
	human-bot lexicon	Special words used by bot and human
	word embedding	Average of word embedding vector for tweet words



(a)Account Information(AI)

(b)Tweet Statistical Information(TSI)

Figure 1. Feature Importance using MDI and Random Forest

E. Classification

In order to classify Twitter users into bot or human, we use machine learning algorithms to train the classifier on the annotated dataset. Among possible candidates, we prefer to use *logistic regression*, *random forest*, and *linear support vector machine* as classifiers since these have proved powerful and have been widely applied in relevant literature. *Logistic regressio*n is a statistical model that uses a logistic function to model a binary dependent variable. *Linear SVM* is a linear model for classification and regression problems. The idea of SVM is to create a hyperplane which separates the data into classes. Finally, *Random forest* consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes is considered as the final prediction.

IV. Experimental Results

The main objective in this section is to examine the performance of our feature groups and their capability to distinguish between human and bot users. To measure the performance of the classification models, we exploit 5-fold cross validation.

As our dataset is imbalanced, in order to be able to compare the performance of the aforementioned classifiers, precision, recall, F1-score and balanced accuracy are applied as metrics. Based on the confusion matrix

TABLE IV. CONFUSION MATRIX FOR TWO CLASS CLASSIFICATION PROBLEM

	Actual (+)	**Actual (-)**
Predict (+)	TP (True Positive)	FP (False Positive)
Predict (-)	FN (False Negative)	TN (True Negative)

These metrics are calculated as follows:

$$precision = \frac{TP}{TP + FP} \quad (3)$$

$$recall\ (TPR) = \frac{TP}{TP + FN} \quad (4)$$

$$TNR = \frac{TN}{TN + FP} \quad (5)$$

$$F1 - score = \frac{2 * precision * recall}{precision + recall} \quad (6)$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

$$balanced - accuracy = \frac{TPR + TNR}{2} \quad (8)$$

Though F1-score keeps the balance between precision and recall and is a great scoring metric for imbalanced data, it does not care about how many true negatives have been classified. Taking into account that it is rather important to detect both positive (i.e., bots) and negative (i.e., humans) classes in our dataset, it follows that balanced-accuracy is a better metric in comparison with F1-score.

Balanced-accuracy is defined as the average of the accuracies of classes. So, if the classifier performs equally well on each class, this metric tends to approach the accuracy. In contrast, if the classifier is biased towards the majority class, the balanced-accuracy will drop to chance [29].

A. Feature Importance Analysis

Mean Decreasing Impurity (sometimes also called Gini Importance) is one of the most common methods to measure the importance of features in Random Forest. Mean Decreasing Impurity (MDI, for breviety) uses a splitting function called Gini Index which measures the level of inequality of the samples assigned to a node based on a split at its parent [30].

In this regard, we calculate feature importance corresponding to each proposed feature group. The bar charts in Figure 1 show, separately, the values of MDI for the features included in Account Information and Tweet Statistical Information (TSI).

[DOI: 10.61186/itrc.15.1.35]

As indicated in Figure 1. "tweet/age", "following/age" and "follow/following" are respectively the most important features amongst those included in AI, while "bi-monthly tweets", "bi-monthly replies" and "distinct_word", besides "std_length" and "mean_length", are the most important ones for TSI. To better display the salient features, the distribution plots of the above features are drawn separately for humans and bots.

In Figure 2. we observe that the mean and the standard deviation of length of tweets published by bots is less than those published by humans. Perhaps this happens since bots usually use a specific template and concise sentences as tweet texts. Also, the variety of distinct words usually used by bots is not so diverse as they very often use duplicate words in their tweets. Figure 3. shows that bots usually keep the number of their follows and followings close to each other and despite the short age of their accounts, they publish a large number of tweets and collect a large number of followings.

Furthermore, we explore the "tweet context" features to determine those of highest accuracy. In this regard, these features are evaluated separately and also combined to other ones in the same group. TABLE V. shows the results obtained by applying Random Forest, linear SVM and Logistic Regression algorithms and 5-fold cross-validation.

The results in TABLE V. reveal that "Word embedding" combined to "TF-IDF" achieve the best

F1-score over all the classifiers. Random Forest reports the best results (86.5%, using 5-fold cross-validation). Also, SVM with 85.62% F1-score has taken the second place. Moreover, "Human-bot lexicon" features besides "Word embedding" leads to desirable results (86.39%, using Random Forest), while the rest of the classifiers do not obtain accurate results. Inspired to these observations, we are convinced to concatenate "TF-IDF" and "Word embedding" in order to represent the text of tweets in the sequel.

B. Model Evaluation

In the next experiment, we evaluate the impact of the proposed feature groups that is *account information (AI)*, *tweet statistical information (TSI)* and *tweet context (TC)*, both in single and in combined form, on the accuracy of classification models. Among all possible combinations of these groups, six feature sets are chosen; namely, (AI, TSI, TC, statistical features (AI+TSI), tweet features (TSI+TC) and all features (AI+TSI+TC)). TABLE VI. **Error! Reference source not found.** shows the results obtained by applying Logistic Regression, linear SVM and Random Forest as classifiers.

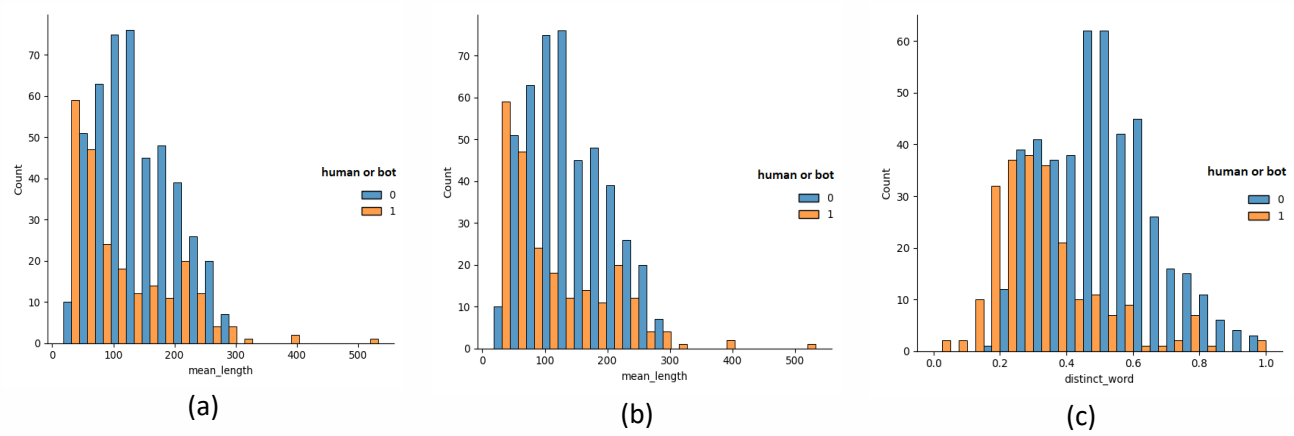


Figure 2. a) distribution of mean_length feature for bots (1) and humans (0) b) distribution of distinct_words feature for bots and humans c) distribution of std_length feature for bots and humans

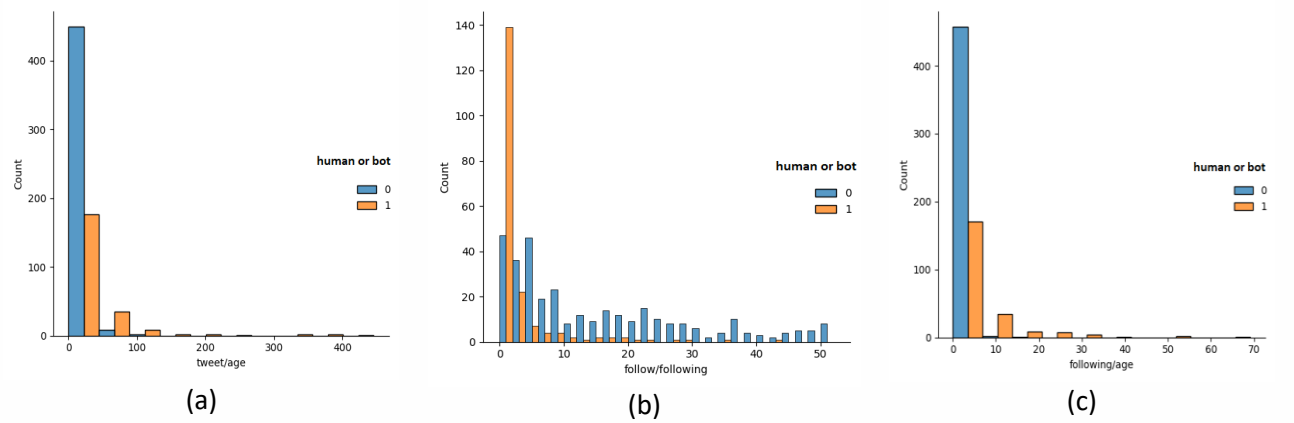


Figure 3. a) distribution of tweet/age over bots and humans b) distribution of follow/following over bots and humans c) distribution of follow/age over bots and humans

[DOI: 10.61186/itrc.15.1.35]

TABLE V. COMPARE TWEET CONTEXT FEATURES USING RANDOM FOREST, SVM AND LOGISTIC REGRESSION

Tweet Context feature	F1-score (%)		
	SVM	Random Forest	Logistic Regression
TF-IDF	80.24	83.77	82.36
Human-bot lexicon	75.66	83.46	80.6
Word embedding	73.09	83.94	70.14
TF-IDF & Human-bot lexicon	74.72	83.01	78.96
TF-IDF & Word embedding	**85.62**	**86.5**	**83.76**
Human-bot lexicon & Word embedding	75.92	**86.39**	81.04
TF-IDF & Human-bot lexicon & Word embedding	76.98	85.1	79.76

As shown in TABLE VI. , Random Forest is the best classifier to detect bot users. Moreover, with a slightly less balanced-accuracy, SVM lies in the second position. However, Logistic Regression does not achieve an accuracy higher than 90% in all experiments. The value of precision, recall and therefore that of F1-score, are close to each other showing that the models are not biased towards the majority class (human users).

From the point of view of time, Logistic Regression is the best. More specifically, on average, total training and testing time for each repetition is 0.20s, 0.23s and 230s when Logistic Regression, Random Forest and linear SVM have been respectively applied.

From the perspective of applied features, in case of Random Forest, the lowest accuracy is achieved by using TC feature group (81.56%), while using SVM and Logistic Regression, TSI feature group does not get an accuracy higher than 80.32% and 74.95%, respectively. On the other hand, the accuracy of 93.45% obtained by applying AI as a single feature group reveals that a significant role is played by an account information in the route of decision making. Indeed, a careful analysis of the results also clarifies that the high number of tweets, followers and followings in a short lifetime lead the classifier to label the user as a bot.

We also note that adding TSI to AI feature group will increase the F1-score a bit more than 2%. Summing up, it should be said that combining all proposed feature groups leads to the almost the same accuracy. Through, as pointed out above, account information has fundamental role to detect bot users.

A more detailed examination of the results show that news bots and bots with large number of retweets can be identified only in terms of TSI and AI features. However, it turns out that identifying the third category of bots (suspicious accounts) requires all the predefined features, particularly "tweet content".

In case all predefined features are used, the confusion matrix for Random Forest has been displayed in TABLE VII. , a glance of which makes us believe that the classifier is able to detect simultaneously both bot and human users in a favorable way.

TABLE VI. CLASSIFICATION RESULTS BY USING DIFFERENT FEATURE SET (AI: ACCOUNT INFORMATION, TSI: TWEET STATISTICAL INFORMATION, TC: TWEET CONTEXT)

Feature set	Classifier	Precision	Recall	F-score	Balanced Accuracy
AI	Random Forest	93.41	93.46	92.32	**93.45**
	SVM	86.2	86.36	86.23	83.97
	Logistic Regression	88.53	88.54	88.35	85.66
TSI	Random Forest	87.04	87.08	86.98	84.59
	SVM	83.01	83.02	82.92	80.32
	Logistic Regression	82.01	81.56	80.35	74.95
TC	Random Forest	86.13	85.91	85.5	81.56
	SVM	86.66	86.78	86.62	83.9
	Logistic Regression	84.47	84.32	83.78	79.73
AI+TSI	Random Forest	**94.82**	**94.77**	**94.75**	**93.86**
	SVM	92.75	92.59	**92.61**	**91.71**
	Logistic Regression	90.41	90.42	**90.37**	88.76
TSI+TC	Random Forest	88.65	88.53	88.28	85.08
	SVM	83.31	83.31	83.19	80.53
	Logistic Regression	82.21	81.85	80.73	75.32
AI+TSI+TC	Random Forest	**93.91**	**93.89**	**93.82**	**91.97**
	SVM	92.69	92.6	**92.6**	**91.68**
	Logistic Regression	90.98	91	**90.94**	**89.21**

[DOI: 10.61186/itrc.15.1.35]

TABLE VII. CONFUSION MATRIX OF RANDOM FOREST CLASSIFIER FOR AI AND TSI FEATURE GROUPS

Classified as Actual class	bot	human	Total
bot	208	21	229
human	18	442	450
Total	226	463	689

V. CONCLUSION

This paper aims at identification of Persian-language bot users on Twitter. To this goal, after collecting and annotating a dataset consisting of Persian-language users and their posts, in a certain period of time, three feature groups are extracted. These include account information, tweet statistical information and tweet context features. To extract features from the tweet context, we use NLP text preprocessing toolkit to normalize and tokenize the text of Persian tweets. We then use the three models of tweet representation; namely, TF-IDF, word embedding and human-bot lexicon. Finally, some well-known classifiers namely, Random Forest, linear SVM and logistic regression, the users are classified as bot or human.

Applying MDP feature importance approach, it turns out that "follow/following", "tweets/age", "distinct_word", "bi-monthly tweets", "bi-monthly replies", "mean_length" and "std_length" are the most important features amongst all. It also turns out that Random Forest classifier works well over all feature groups. Moreover, the results indicate that the features related to account information play, on their own right, a crucial role in identifying bot users (particularly in case of news and automated bots), while tweets language is of less importance and impact. Finally, a combination of account information, tweet statistical information, and tweet context features may lead to the best possible result specially in case of suspicious accounts.

REFERENCES

[1] Q1-2019 earnings report, Available at:

[2] https://s22.q4cdn.com/826641620/files/doc_financials/2019/q1/Q1-2019-Slide-Presentation.pdf.

[3] Chu, Zi; Gianvecchio, Steven; Wang, Haining; Jajodia, Sushil (2012). "Detecting Automation of Twitter Accounts: Are You a Human, Bot, or Cyborg?" (PDF). IEEE Transactions on Dependable and Secure Computing. 9 (6): 811–824. doi:10.1109/TDSC.2012.75. ISSN 1545-5971. S2CID 351844.

[4] P. Gamallo and S. Almatarneh (2019) "Naive-Bayesian Classification for Bot Detection in Twitter Notebook for PAN at CLEF 2019", *CLEF 2019*, Lugano, Switzerland.

[5] M. Latah (2020) "Detection of Malicious Social Bots: A Survey and a Refined Taxonomy", *Expert Systems with Applications*, vol. 151, pp. 113383.

[6] M. Shamsfard (2019) " Challenges and Opportunities in Processing Low Resource Languages: A study on Persian", International Conference Language Technologies for All (LT4All), Dec 2019, Paris, France.

[7] I. Inuwa-Dutse, M. Liptrott, I. Korkontzelos (2018) "Detection of spam-posting accounts on Twitter", Neurocomputing, vol. 315, pp 496-511.

[8] Z. Gilani, R. Farahbakhsh, G. Tyson, L. Wang, and J. Crowcroft (2017) "Of bots and humans (on twitter)", In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, pp. 349-354. ACM.

[9] S. B. Jr, G. F. C. Campos, G. M. Tavares, R. A. Igawa and M. L. P. Jr (2018) "Detection of Human, Legitimate Bot, and Malicious Bot in Online Social Networks Based onWavelets", *ACM Trans. Multimedia Comput. Commun. Appl*, vol. 26, no. 1, pp. 1-17.

[10] Z. Chu, S. Gianvecchio, H. Wang and S. Jajodia (2012) "Detecting Automation of Twitter Accounts: Are You a Human, Bot, or Cyborg?", *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 6, pp. 811-824.

[11] C. A. Davis, O. Varol, E. Ferrara, A. Flammini and F. Menczer (2016) " Botornot: A system to evaluate social bots", *In Proceedings of the 25th international conference companion on world wide web*, pp. 273-274.

[12] O. Loyola-Gonzalez, R. Monroy, J. Rodriguez, A. Lopez-Cuevas, J. I. Mata-Sanchez (2019) " Contrast pattern-based classification for bot detection on twitter", IEEE Access, vol. 7, pp. 45800-45817.

[13] D. M. Beskow, K. M. Carley (2018) "Bot conversations are different: leveraging network metrics for bot detection in twitter", *In 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp 825-832.

[14] O. Loyola-Gonzalez, R. Monroy, J. Rodriguez, A. Lopez-Cuevas, J. I. Mata-Sanchez (2019) " Contrast pattern-based classification for bot detection on twitter", IEEE Access, vol. 7, pp. 45800-45817.

[15] I. Inuwa-Dutse, M. Liptrott, I. Korkontzelos (2018) "Detection of spam-posting accounts on Twitter", Neurocomputing, vol. 315, pp 496-511.

[16] R. A. Igawa, S. Barbon Jr, K. C. S. Paulo, G. S. Kido, . R. C. Guido, M. L. P. Júnior and I. N. d. Silva (2016) "Account classification in online social networks with LBCA and wavelets", *Information Sciences*, vol. 332, pp. 72-83.

[17] Wei and U. T. Nguyen (2019) " Twitter Bot Detection Using Bidirectional Long Short-term Memory Neural Networks and Word Embeddings", *In First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, pp. 101-109.

[18] J. Pennington, R. Socher and C. Manning (2014) "Glove: Global vectors for word representation" in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), Doha, Qatar.

[19] N. Chavoshi, H. Hamooni and A. Mueen (2016) " DeBot: Twitter Bot Detection via Warped Correlation", *In IEEE 16th International Conference on Data Mining (ICDM)*, pp. 817-822.

[20] S. Cresci, R. D. Pietro, M. Petrocchi, A. Spognardi and M. Tesconi (2018) "Social Fingerprinting: Detection of Spambot Groups Through DNA-Inspired Behavioral Modeling," in *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 561-576.

[21] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi and M. Tesconi (2016), "DNA-Inspired Online Behavioral Modeling and Its Application to Spambot Detection" in *IEEE Intelligent Systems*, vol. 31, no. 5, pp. 58-64.

[22] P. G. Pratama and N. A. Rakhmawati (2019) "Social Bot Detection on 2019 Indonesia President Candidate's Supporter's Tweets" Procedia Computer Science, vol. 161, pp. 813-820.

[23] D. Stukal, S. Sanovich, J. A. Tucker and R. Bonneau (2019) "For whom the bot tolls: A neural networks approach to measuring political orientation of Twitter bots in Russia", SAGE Open, vol. 9, no. 2.

[24] S. Cresci, F. Lillo, D. Regoli, S. Tardelli and M. Tesconi (2019) " Cashtag piggybacking: uncovering spam and bot activity in stock microblogs on twitter", ACM Transactions on the Web (TWEB), vol. 13, no. 2, pp. 1-27.

[25] A. Balestrucci, R. De Nicola, O. Inverso, and C. Trubiani (2019) " Identification of credulous users on Twitter", In Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, pp. 2096-2103.

[DOI: 10.61186/itrc.15.1.35]

[26] A. Balestrucci, R. De Nicola, M. Petrocchi and C. Trubiani (2019) "Do you really follow them? Automatic detection of credulous Twitter users", In International Conference on Intelligent Data Engineering and Automated Learning, pp. 402-410, Springer, Cham.

[27] Hazm. (2014). Python library for digesting Persian text, “Https://github.com/sobhe/hazm.”

[28] Z. Sarabi, H. Mahyar, M. Farhoodi (2013, October) "ParsiPardaz: Persian Language Processing Toolkit", In Computer and Knowledge Engineering (ICCKE), 2013 3th International eConference on (pp. 73-79). IEEE.

[29] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov (2017) "Enriching word vectors with sub-word information", *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135-146.

[30] K. H. Brodersen, C. S. Ong , K. E. Stephan and J. M. Buhmann (2010) " The balanced accuracy and its posterior distribution", In 2010 International Conference on Pattern Recognition, (pp. 3121-3124). IEEE.

[31] Y. Qi (2012) "Random forest for bioinformatics." Ensemble machine learning. Springer, pp. 307-323.

Mojtaba Mazoochi received his B.Sc. degree in Electrical Engineering from Tehran University, Iran in 1992. He received his M.Sc. degree from Khajeh Nasir Toosi University of Technology, Iran in 1995 and his Ph.D. degree from Islamic Azad University, Tehran, Iran in 2015 in Electrical Engineering (Telecommunication). He is an Assistant Professor and head of Digital Transformation Skills Training Center in ICT Research Institute (ITRC), Tehran, Iran. His research interests include Data Analytics, Quality of Service (QoS), and Network Management.

Nasrin Asadi received her Ph.D. degree in Software Engineering from ICT Research Institute (ITRC) with emphasis on text summarization. She received her B.Sc. in the field of Software Engineering from Amirkabir University in 2007 and her M.Sc. from Shiraz University in 2010. Her current research interests include Text Mining, Text Summarization, and Natural Language Processing.

Farzaneh Rahmani received her B.Sc. degree in Computer Engineering from Bahonar University, Kerman, Iran and her M.Sc. degree in Computer Engineering from Tarbiat Modares University and Ph.D. degree from ICT Research Institute (ITRC), Tehran, Iran. Her research interests include Social Networks Analysis, Machine Learning, Natural Language Processing, and Computer Vision.

Leila Rabiei received her B.Sc. degree in Computer Engineering from Islamic Azad University of Tehran, Iran, and her M.Sc. degree in Computer Engineering from Iran University of Science and Technology, Tehran, Iran. She is currently works as a researcher and project manager in the Development and Innovation Center for AI in ICT Research Institute (ITRC), Tehran, Iran. Her research interests include Big Data Analysis, Data Mining and Social Networks Analysis.

[DOI: 10.61186/itrc.15.1.35]