Retraction

Retraction: Detect fake identities using improved Machine Learning Algorithm (J. Phys.: Conf. Ser. 1916 012056)

Published 23 February 2022

This article (and all articles in the proceedings volume relating to the same conference) has been retracted by IOP Publishing following an extensive investigation in line with the COPE guidelines. This investigation has uncovered evidence of systematic manipulation of the publication process and considerable citation manipulation.

IOP Publishing respectfully requests that readers consider all work within this volume potentially unreliable, as the volume has not been through a credible peer review process.

IOP Publishing regrets that our usual quality checks did not identify these issues before publication, and have since put additional measures in place to try to prevent these issues from reoccurring. IOP Publishing wishes to credit anonymous whistleblowers and the Problematic Paper Screener [1] for bringing some of the above issues to our attention, prompting us to investigate further.

[1] Cabanac G, Labbé C and Magazinov A 2021 arXiv:2107.06751v1

Retraction published: 23 February 2022
Detect fake identities using improved Machine Learning Algorithm

R Suganya¹, S Muthulakshmi¹, B Venmuhilan¹, K Varun Kumar¹ and G Vignesh¹
¹Dept. of Information technology, Sri Krishna College of Technology, Coimbatore, Tamil Nadu, India.
r.suganya@skct.edu.in

Abstract. Nowadays, Social networks are controlling the world. Number of users using the social media platforms are continuously increasing every year. The main advantage of Online social media is anyone from anywhere in the world can connect to any users easily and able to communicate with them. This paved the way of malicious attacks like fake identities and spams. The Survey of Facebook and Twitter states that the actual users present in their platforms is lesser than the number of accounts created. It means there is an increase in fake profiles over the year. With these accounts, their creators can influence the users and distribute false information. They exploit the users in order to achieve their hateful aims, for example, creating hateful posts/tweets. The increasing number of fake accounts in social media platforms is becoming a major risk for the internet users. So, there is a need for identifying these fake identities. The existing machine learning algorithms are SVM linear, naïve bayes, random forest and Decision tree algorithm which are used to detect fake identities provides less accuracy and has become inefficient. We implement Ada-boosting algorithm with SVM to detect and identify bots. The proposed model is Ada-boost SVM classifier provides higher accuracy in detecting the fake accounts and show better results in the detection of identity bots.

Keywords: Machine Learning, Support Vector Machine (SVM), Social networks, Data Pre-processing, Social Media Platforms (SMPs);

1. Introduction

Online Social Network is a popular media for sharing and receiving data. It is used for business, education, Digital marketing, content providing, daily news and updates etc. Millions of users are accessing social media every second [1]. They use it to interact the content and information provided by the platform users. People from all over the world use the social media to communicate and share their innovative works and ideas. It also provides a platform to prove their talents and to search out new people with same interest anywhere in the world [2]. There are different kinds of Social media like Instagram, LinkedIn, Twitter, Facebook, Snapchat etc. As the technology is developing, Social media is becoming the daily necessity for most of human beings.

There are many advantages in Online social media network, they also have some disadvantages too. There is risk for every user in the social media because of problems like Malware attacks and privacy breach attacks [3]. Privacy breach attack is when the attackers access the private information without the permission of the user and the data can be misused. Nowadays, Social media becoming the platform for spreading malicious software by interacting and advertising among user accounts. Another major
problem is Fake accounts or bots [4]. A Fake account is representation of individual or organization who pretends to be someone else. There are two types of fake Accounts. (i) Duplicate accounts- they are the backup account created by the user for extra uses without indulging in other user’s activities. (ii) False accounts or undesirable accounts – Some false accounts are created for business and organization benefits. But mostly these accounts are mainly used to spam and contravene the Social Media terms and service. Many false accounts have been created not only to spam but also to influence the users in a wrong way for their promotions. Some accounts are created to support some political parties or some corporate organization for their development. Fake accounts are continuously increasing every year as shown (in Figure 1).

Fake accounts or bots are responsible for spreading fake news and rumours. Some bots are created specially to defame a person/celebrity’s character. They also changing the thoughts of a user by influencing them through fake news and information. For Instance: Twitter is a social network used to share current affairs and news [5]. Users discuss about their opinions. In twitter, Hashtag used to find and post a particular subject. It becomes a trending topic when hashtag for the topic used by the greatest number of users. Sometimes the hashtag trending is done by the bots as the large number of fake accounts uses that particular hashtag. To increase the sales of products and to generate large revenue, paid promotions are being done [6-10]. Bots are also used to increase the sales of products by promoting the particular organization. So, it is necessary to detect and identify the fake users and to secure the normal accounts from malicious attacks. A software that can send tweets automatically to users are called Twitter Bots. Bots are designed for doing malicious activities like spamming [11].

![Figure 1. Shows increase in no of Fake accounts and Duplicate Accounts over the years](image)

2. Existing System

Three kinds of users in Social media Platforms are sybils, trusted and honest. Multiple accounts controlled by an adversary are called as Sybil accounts. Sybils also have some connections to honest users and the honest and Sybil regions are connected in a scattered manner [12]. Fake trustworthy impression created by the Sybil communities on genuine accounts of the Social Network. The difference between fake accounts and the normal account was noted with respect to account properties like external URL ratio, tweet content and tweeting behaviour.

There are various Machine Learning algorithms are used in detecting fake accounts [13-16]. Among them, Decision trees and random forest algorithm are the most common machine learning algorithms by the existing systems. Decision tree and random forest detects the bots with an accuracy of 87.85%, 91.49% [17]. Though these algorithms are able to detect and identify the bots and normal accounts, they provide the final output with less accuracy. If the machine learning model is improvised with different ML algorithms, it can increase the accuracy level overall outcome. Many existing systems uses only a
few attributes in detecting fake identities [18-20]. Accuracy of the decision making depends on the number of attributes. So, if the number of attributes is less, the accuracy will also reduce.

3. Proposed System

In this paper, we intend to use Machine learning classification algorithms to determine whether the identity of the account is real or fake using our newly developed algorithm Adaptive SVM classifier (Ada-boost SVM). The proposed algorithm (Ada-boost SVM) is able to correctly classify the accounts of our training dataset into bots and non bots with high accuracy.

Figure 2 shows the proposed system architecture of fake account detection. It is real-time data analytics. The input data containing the Twitter dataset with some features. The training dataset contains data pre-processing. It includes two steps: Feature Extraction and Machine learning technique. Later, these two are arranged in a model, which is used for selecting the number of features. After that apply the Support Vector machine for classifying our data and Ada-boost used for boosting the classifiers (It is called Adaptive Boosting as the weights are re-assigned to each instance). Therefore, it increases the accuracy of the classifiers. After applying the algorithms, it predicts whether the model account is fake or not. Finally, evaluate and visualize the results.

4. Modules

The modules are:

- Data Aggregation: Current social media Data sets which are collected from the Kaggle.
- Data Pre-processing: the process of fixing or removing, incomplete or duplicate date within a dataset and clean the irrelevant data and fill the empty rows & columns.
- Machine learning algorithm: Used Ada-boost SVM algorithm to classify the bots and non-bots.

4.1. Data Aggregation

Nowadays, it becomes more important to acquire large amounts of data and label data for machine learning as it becomes more widely used. Collecting the data for training the ML model is the fundamental step in the machine learning pipeline. The predictions done by ML systems can only be as good as the datasets on which they have been trained. The information processed into the system is data.
The quality and the correctness of the algorithm will determine the efficiency and the accuracy of the algorithm. So the output depends on the datasets.

4.2. Data Pre-processing
Real-world untrained data and images are often incomplete, inconsistent and lacking in certain behaviours or trends. They are also possibly contain many errors [20-25]. So, once the data is collected and pre-processed into a format the machine learning algorithm can use them for the model.

Pre-processing techniques and actions are:
- Data cleaning: These techniques remove data that are incorrectly added or classified. Sometime the collected data may possibly contain a lot of unwanted information and null values and also there is a possibility that the data might be in an unorganized format. So, clean all the data. Replace the unwanted data with the approximate data. Fill the null values with some static values are the some steps in pre-processing of data. Some collected datasets which may contain only garbage values. Now, the data is organized and ready for next process.
- Data imputations: For balancing or filling in missing data most ML frameworks include methods and APIs. Techniques generally include attributing missing values with standard deviation, mean, median of the data in the given field.
- Oversampling: Imbalance in the dataset or Bias can be rectified by generating more observations/samples with methods like repetition, bootstrapping and then attach them to the under-represented classes.
- Data integration: Integrating multiple datasets to get a large entity can overcome incompleteness in a single dataset.
- Data normalization: The size of a dataset impacts the processing and memory needed for iterations during training. Normalization decreases the size by reducing the order and magnitude of data.

4.3. Machine Learning Algorithm
Machine learning algorithms are implemented to train and test the classifiers. It is based on several Social Media features such as attribute similarity, account friend similarity and so on. A combination of SVM and Ada-boost are used for the proposed system to improve the accuracy of the existing system and also to maintain a high accuracy in detecting the bots. Different algorithms used are listed below:

4.3.1. SVM: SVM stands for Support Vector Machine.
Support Vector Machine (SVM) is a Supervised Machine learning algorithm. It is mainly used to classify data into different classes. In SVM, data is divided into segments and it provides data analysis for regression and classification analysis. SVM can be used for regression, but it is mostly used for classification. The coordinate representations of individual observation are called Support Vectors. (Figure 3).
- Hyperplane - boundary line which helps to classify data points.
- Support Vectors - data points which lie closer to boundary line.
- Margin – It is a distance between the data points and the boundary line.

Support Vector Machine is a robust method to classify huge datasets like dataset of social network with millions of profiles. It is a binary classification algorithm that observes the maximum separation hyperplane between two classes. SVM creates a hyperplane to divide the dataset into classes by taking all the data points. Then it checks the distance between the boundary line and the nearest data points from each class. As SVM is a supervised learning algorithm, there is enough training examples are given which divides two classes fairly well, and classifies new examples. Given a bias weight b and weight vector w, we formulize the classification methodology as:

\[ wTx + b > 0 \Rightarrow \text{positive class} \]
\[ wT \mathbf{x} + b < 0 \Rightarrow \text{negative class} \]

This equation 1 gives a separator and it is instinctive that depends upon the choice of the above-mentioned parameters. Kernel in SVM is used to provide more efficient and modest way to convert data into higher dimensions. The kernel is:

\[ k(x, x_i) = \text{sum}(x \ast x_i) \] (1)

The kernel determines the similarity between support vectors and the new data. Similarity can be linear combination of inputs. The objective of an SVM algorithm is to maximize the margin. The hyperplane becomes optimal, when the margin reaches its maximum.

4.3.2. Ada Boost Classifier.
Ada-boost (Adaptive boosting) classifier merges weak learning algorithms to create optimal strong learning algorithm. Bootstrapping is not followed by Ada-boost. Some algorithms poorly classify the attributes. By merging various classifiers, it can improve the accuracy level of the overall classifiers by choosing data at each iteration and allocating weights at final stage. Based on the accuracy of the previous training, it chooses the training set. It creates a Decision tree with one depth called decision stumps. Initially, it assigns equal weights to every training data. It creates N decision stumps for n number of features. Equation 2 shows it Calculates the performance of the stump. Classifier with more accuracy is assigned with high weight to get the final output.

\[ H(x) = \text{sign} \left( \sum_{t=1}^{T} \alpha_t h_t(x) \right) \] (2)

Whereas,
- \( h_t(x) \) – week classifier’s output t for x
- \( \alpha_t \) - weight alloted to classifier.
- \( \alpha_t \) can be calculated as: \( \alpha_t = \frac{1}{2} \times \ln \left( \frac{1 - E}{E} \right) \). \( \alpha_t \) is based on the error rate E. Each training data has equal weights initially.

4.3.3. Ada Boost Classifier.
The component classifiers gain from boosting when applying boosting technique to robust classifiers. Algorithm: Ada-Boost SVM

1. Input: input contains the training data \{(x1, y1), ..., (xn, yn)}\, \sigma, \sigma \text{ min}, \sigma \text{ step}.
2. Initialize: initialize the weight, \( w_i = 1/N \) (i = 1, ..., N) for all training data.
3. Do While (\( \sigma > \sigma \text{ min} \))
   (1) On the weighted training set, train the classifier.
   (2) Calculate the total error of \( h_t \).
   (3) If \( er > \frac{1}{2} \), reduce \( \sigma \) value by \( \sigma \text{ step} \) and goto (1).

Figure 3. Support Vector Machine (SVM)
(4) weight of classifier \( h_t = \frac{1}{2} \ln \left( \frac{1 - E}{E} \right) \).
(5) Update the weight of the training samples.

Proposed Ada-Boost SVM algorithm can be described as follows: The data collected and processed is trained with Ada-boost SVM algorithm. The classifier is based on several Social Media attributes such as id, followers, friends list, verified status, description, location etc. Create a condition statement which helps to distinguishes the attributes between bots and non bots. Print and visualize the results.

Flask API (Application Programming Interface) is used in the framework for having a communication between the front end and back end. An application programming interface is used to send requests and receive responses to and from the application. In our application, the scores obtained from the module is passed to the machine learning model via the flask API. The request in this scenario is to identify the fake accounts from the given data. The response in this scenario is to classify the bots and non bots with high accuracy. Python is used to implement the Application programming interface part.

5. Results
The performance of the machine learning model can be determined by a concept called confusion matrix. A confusion matrix is represented in the form of a table in which there will be four different values present in the matrix such as true positive, false positive, true negative, and false negative. This can be used on the test dataset for which the true values are already known.

- True positives (TP) - the value in which the examples are correctly determined as positive.
- True negatives (TN) - the value in which the examples are correctly determined as negative.
- False positives (FP) - the value in which the examples are negative but are actually determined as positive.
- False negatives (FN) - the value in which the examples are positive but are actually determined as negative.

The confusion matrix determines the performance of the classification model by calculating recall, accuracy, precision, f-measure and error-rate. The following are the formulae for each of the above-mentioned performance measures:

\[
\begin{align*}
TP &= TP/(TP+FN) \\
FP &= FP/(FP+TN) \\
Accuracy &= (TP+TN)/(TP+TN+FP+FN) \\
Precision &= TP/(TP+FP) \\
Recall &= TP/(TP+FN) \\
F-measure &= 2 \times (Precision \times Recall) / (Precision + Recall) \\
Error-rate &= 1 - accuracy
\end{align*}
\]

Table 1. This table shows the accuracy of different algorithms

| Algorithm            | Accuracy (%) |
|----------------------|--------------|
| Multinomial Naïve Bayes | 69.76        |
| Decision Tree        | 87.85        |
| Random Forest        | 91.49        |
| SVM                  | 92.84        |
| Ada-Boost SVM        | 97.16        |
From above inferred (Table1), Ada-boost SVM is the classification technique which is determined as the efficient classifier for Identifying Fake Accounts. Ada-boost SVM have a Precision - 0.96984925, recall - 0.97379913, F-score - 0.96501048.

6. Conclusion
Fake Identities are continuously increasing in online social media platforms. Therefore, it is necessary to create an efficient new methods and algorithms which detect Fake profiles in online social media platforms. In this project, we have been able to explore the problems encountered by the existing systems, to design the proposed algorithm which obtains higher accuracy than the existing classifiers. Adaptive SVM (Ada-boost SVM) algorithm is the best learning model with an accuracy of 97.16% when it is compared to Decision Tree, Multinomial Naïve Bayes and Random Forest. Finally, the research work concluded by detecting, eliminating fake bot accounts which are created in social media platforms in an efficient way.

References
[1] Majed Alrubaian, Muhammad Al-Qurishi, Mohammad Mehedi Hassan, and Atif Alamri, A Credibility Analysis System for Assessing Information on Twitter, IEEE(Aug 2018) https://ieeexplore.ieee.org/document/7551232
[2] Van Der Walt, Jan Eloff and Estée. Using machine learning to detect fake identities: bots vs humans. IEEE (2018) https://ieeexplore.ieee.org/document/8265147
[3] Tri Hadiah Mulawati, Perdana, Rizal Setya, and Reddy Alejandro. Bot spammer detection in Twitter using tweet similarity and time interval entropy. (Research Gate)
[4] Karatas, Arzum, and Serap Şahin. A Review on Social Bot Detection Techniques and Research Directions. (2015)
[5] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, Fame for sale: efficient detection of fake twitter followers. IEEE (2014) pp. 620-627 https://arxiv.org/abs/1509.04098
[6] S, D., & H, A. (2019). AODV Route Discovery and Route Maintenance in MANETs. 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS). doi:10.1109/icaccs.2019.8728456
[7] H. Anandakumar and K. Umamaheswari, An Efficient Optimized Handover in Cognitive Radio Networks using Cooperative Spectrum Sensing, Intelligent Automation & Soft Computing. pp. 1–8, Sep. 2017. doi:10.1080/10798587.2017.1364931
[8] S. Gurajala, J.S. White, B. Hudson, B. R. Voter, and J. N. Matthews, Profile characteristics of fake Twitter accounts.
[9] M.Tsikerdekis , S. Zeadally. Multiple Account Identity Deception Detection in Social Media Using Nonverbal Behavior, (IEEE) 2014.
[10] Naman Singh, Tushar Sharma, Abha Thakral, Tanupriya Chaudhury, Detection of Fake profile in Online Social Network using Machine Learning.
[11] M Egele, G Stringhini, C Kruegel, and G Vigna, Compa: Detecting compromised accounts on social networks.
[12] K Sreenivasa Rao, Seeram Gutha, Dr B Devena Raju, Detecting Fake Account on Social Media Using Machine Learning Algorithms. https://www.researchgate.net/publication/340816200_DETECTING_FAKE_ACCOUNT_ON_SOCIAL_MEDIA_USING_MACHINE_LEARNING_ALGORITHMS
[13] Sarah Khaleed, Neamat El-Tazi, Hoda M. O. Mokhtar, Detecting Fake Accounts on Social Media. https://ieeexplore.ieee.org/document/8621913
[14] Ilham Aydin, Mehmet SEVİ, Mehmet Umut SALUR, Detection of Fake Twitter Account with Machine Learning Algorithms, IEEE (2019)
[15] Myo Myo Swe and Nyein Nyein Myo, Fake accounts on Twitter using Blacklist, IEEE (2018)
[16] Manuel Egele, Gianluca Stringhini, Christopher Kruegel, and Giovanni Vigna Towards Detecting Compromised Accounts on Social Networks, IEEE Exchanges on Reliable and Secure Processing (Volume: 14, Issue: 4, July-Aug. 1 2017).

[17] O. Dekel, O. Shamir, and L. Xiao. Machine Learning to classify with missing and corrupted features, Machine Learning, 2010.

[18] Nitin T Simon Dr. Susan Elias, Detection of Fake Followers using Feature Ratio in Self-Organizing Maps, 2017 IEEE.

[19] J. P. Dickerson, V. Kagan, and V. S. Subrahmanian, Using sentiment to detect bots on Twitter: Are humans more opinionated than bots? in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM), Aug. 2014, pp. 620–627.

[20] Xiaoyun Wang, Chun-Ming Lai, Yunfeng Hong, ChoJuiHsieh, S. Felix Wu Multiple Accounts Detection on Facebook Using Semi-Supervised Learning on Graphs.

[21] hubham Patil, Akshay Ingale, Pradeep Ranher, Purushottam Mahakal and Mr. Y. L Hakim Detect Fake Identities of Bots vs Human using Machine Learning, IJSRD -International Journal for Scientific Research & Development| Vol. 7, Issue 02, 2019 | ISSN (online): 2321-0613.

[22] Xiao, D. M. Freeman, and T. Hwa, Detecting clusters of fake accounts in online social networks, in Proc. 8th ACM Workshop Artif. Intell. Secur., 2013, pp. 91–101.

[23] A. K. Jain and B. Gupta, Phishing detection: Analysis of visual similarity-based approaches, Security and Communication Networks, 2017.

[24] P. Galán-García, J. G. De La Puerta, C. L. Gómez, I. Santos, and P. G. Bringas, Supervised machine learning for the detection of troll profiles in Twitter social network: Application to a real case of cyberbullying, 2015.

[25] Odukoya Oluwatojini Helen, A Social Network Spam Detection Model, International Journal of Scientific & Engineering Research, Volume 8, ISSN 2229-5518