FAKE PROFILE DETECTION USING MACHINE LEARNING

Dr. N .Ananthi[1] Sharan Deepak RB[2] Shrinithi Sellam V P[3] Sneha N[4]

[1] Professor, Dept. of Information Technology, Easwari Engineering College, Chennai

[2,3,4] Student, Dept. of Information Technology, Easwari Engineering College, Chennai

Abstract— ***The social network, a crucial part of our life is plagued by online impersonation and fake accounts. According to the 'Community Standards Enforcement Report' published by a private social media organization on March 2018, about 583 million fake accounts were taken down just in quarter 1 of 2018 and as many as 3-4% of its active accounts during this time were still fake. In this project, we propose a model that could be used to classify an account as fake or genuine. This model uses Random Forest Algorithm as a classification technique and can process a large dataset of accounts at once, eliminating the need to evaluate each account manually. The community of concern to us here is Fake Accounts and our problem can be said to be a classification or a clustering problem.***

Keywords— Online Social Network; Machine Learning; Random Forest; Regression; Classification; Linear Regression;

I. INTRODUCTION

Nearly everyone nowadays is a member of at least one of the online social networking websites, making social networking platforms an indispensable aspect of daily life. Since there is always a big number of users on these platforms, spammers and fake users have been drawn to online social networking. Users establish false profiles in order to distribute false information like rumors, hate speech, bullying language, and other things. Although many fraudulent accounts still exist, researchers have proposed a number of methods to reduce this problem utilizing machine learning and deep learning-based models. However, these fraudulent profiles are unacceptable for a reputable social networking site. An attacker creates a fake profile in order to connect to a victim to cause malicious activities. And also to spread fake news and spam messages.

II. LITERATURE SURVEY

1) In today's social networks, fake and clone profiles pose a very real threat. Therefore, a detection technique is absolutely important to find these frauds that exploit people's faith to gather personal information and create fake profiles. Many authors have researched this issue and suggested ways to spot these kinds of social network profiles. Following are some of these techniques discussed.

P. Sowmya and M. Chatterjee[1] have proposed a prototype to Profile Cloning detection using two methods. One using Similarity Measures and the other using C4.5 decision tree algorithm

S. Revathi and D. M. Suriakala[2] put an effort to accomplish an idea of profile cloning recognition in Online Social Networks (OSN) utilizing Network Theory.

W. Shahid, Y. Li, D. Staples, G. Amin, S. Hakak and A. Ghorbani[3] provided a comprehensive survey of the state of art methods for detecting malicious users and bots based on different features proposed in our novel taxonomy.

E. Van Der Walt and J. Eloff[4] researched and discussed in this paper applies these same engineered features to a set of fake human accounts in the hope of advancing the successful detection of fake identities created by humans on SMPs.

G. Sansonetti, F. Gasparetti, G. D'aniello and A. Micarelli [5] performed (i) an offline analysis realized through the use of deep learning techniques and (ii) an online analysis that involved real users in the classification of reliable/unreliable user profiles

III. EXISTING SYSTEM

Naive Bayes algorithm having less accuracy. Because of Privacy Issues the social media dataset is very limited and a lot of details are not made public. One of the easiest ways to determine that a Facebook account is fake is by examining the photo. It's often the case that fake accounts use a profile photo that they've downloaded from somewhere else online.

Just fire up Google Image search, then download the profile photo from the Facebook page that you suspect is fake. Drag and drop that photo into the Google Image Search bar and click the Search button. If the photo is from a fake Facebook account, you should see loads of matches all across cyberspace.

IV. PROPOSED SYSTEM

The threat posed by fake profiles to society has grown significantly. Hackers can easily access information like phone numbers, email addresses, school or college names, corporate names, locations, etc. in social networks to build fake profiles. Then they attempt to carry out a number of attacks, including phishing, spamming, cyberbullying, etc. They even make an effort to defame the organisation or the legal owner. In order to make users' social lives more secure, a detection system that can identify fake profiles has been presented. The selection of the profile that has to be categorized is the first step in classification. Following the selection of the profile, the useful features are extracted in order to do classification. A trained classifier is then fed the retrieved characteristics. As new data is given into the classifier, it is regularly trained. After that, the classifier decides if the profile is real or fake. The classification algorithm's output is then confirmed, and the classifier is given feedback. The classifier gets better and better at predicting the fake profiles as the amount of training data grows.

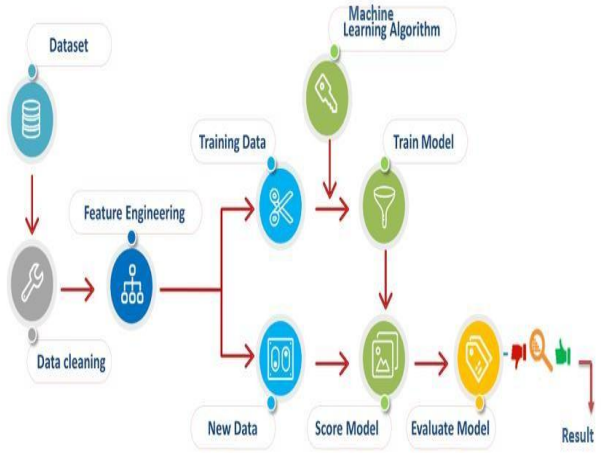

Fig.1.Architecture Diagram of proposed system

A. Collecting Dataset

Data Collection is one of the most important tasks in building a machine learning model. We collect the specific dataset based on requirements from internet. The dataset contains some unwanted data also. So first we need to pre-process the data and obtain perfect data set for algorithm

B. Data pre-processing

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

C. Data cleaning

Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.

D. Data transformation

Data transformation may include smoothing, aggregation, generalization, transformation which improves the quality of the data.

E. Data selection

Data selection includes some methods or functions which allow us to select the useful data for our system.

V. ALGORITHM

Random forest, like its name implies, 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 becomes our model's prediction.

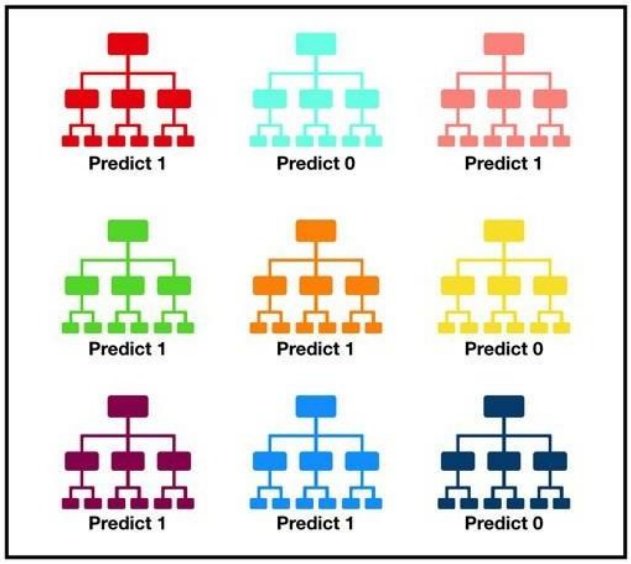

Fig.2. Prediction: 1

The fundamental concept behind random forest is a simple but powerful one — the wisdom of crowds. In data science speak, the reason that the random forest model works so well is:

A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.

The low correlation between models is the key. Just like how investments with low correlations (like stocks and bonds) come together to form a portfolio that is greater than the sum of its parts, uncorrelated models can produce ensemble predictions that are more accurate than any of the individual predictions.

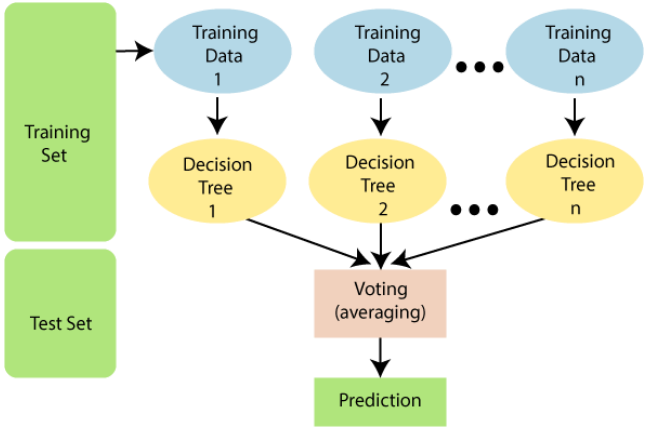

Fig.3. Random Forest

The reason for this wonderful effect is that the trees protect each other from their individual errors (as long as they don't constantly all err in the same direction). While some trees may be wrong, many other trees will be right, so as a group the trees are able to move in the correct direction.

There needs to be some actual signal in our features so that models built using those features do better than random guessing.
The predictions (and therefore the errors) made by the individual trees need to have low correlations with each other.

A. Attribute Similarity

Based on the similarity of attribute values between the profiles, attribute similarity is determined. The following characteristics are taken into account while calculating similarity: Name, ScreenName, Language, Location, and Time zone. Cosine similarity and Levenshtein distance are used as two similarity measures to assess how similar the qualities are to one another. Levenshtein distance is used to detect similarities between two sequences, and cosine similarity is used to find similarities between words. Cosine similarity formula is given by equation

$$\cos\theta = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

where Ai and Bi are two non-zero vectors. A cosine similarity of 1 exists between two vectors when they have the same orientation, 0 when they are at 90°, and -1 when they are diametrically opposing. A similarity metric used to compare two sequences is the Levenshtein distance. The Levenshtein distance between any two sequences is the least amount of insert, delete, or substitution operations necessary to transform one sequence into another. The equation describes the Levenshtein distance between two strings, a, and b, with lengths I and j, respectively.

$$lev_{a,b}(i,j) = : \begin{cases} \max(i,j) \\ min\begin{cases} lev_{a,b}(i-j,j)+1 \\ lev_{a,b}(i,j-1)+1 \\ lev_{a,b}(i-1,j-1)+1_{ai\neq bj} \end{cases} \end{cases} \quad \text{If } \min(i,j)=0, \text{ else}$$

B. Evaluation Metrics

The system's performance is measured using a variety of evaluation metrics based on the four main standard indications.
• True Positive (TP): Records that are accurately discovered using anticipated vectors are referred to as true positives.
• True Negative (TN): Records that were appropriately identified as Neutral but were actually True Negatives.
False positives (FP) are records that the system mistakenly thought were being discovered but are really being listed in the other vectors.
• False Negative (FN): False negative records are ones that the system did not detect.
The evaluation indicators taken into account are: 1. Accuracy, which measures the proportion of correct results to total inputs.
2. Precision, which indicates the percentage of correctly detected positives,
3. Recall, which displays the percentage of real positives that were accurately detected.
4. F1 Score, which computes the score by accounting for both precision and recall. The harmonic mean of precision and recall yields the F1 score. The best value is 1 and the poorest value is 0 if the F1-score is 1.

VI. LEARNING CURVE AND CONFUSION MATRIX

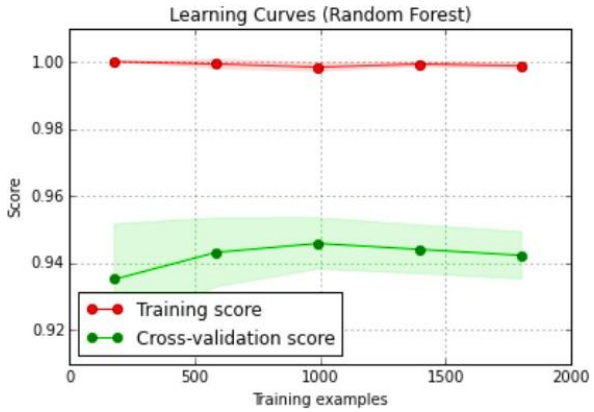

Fig.4. Learning Curve

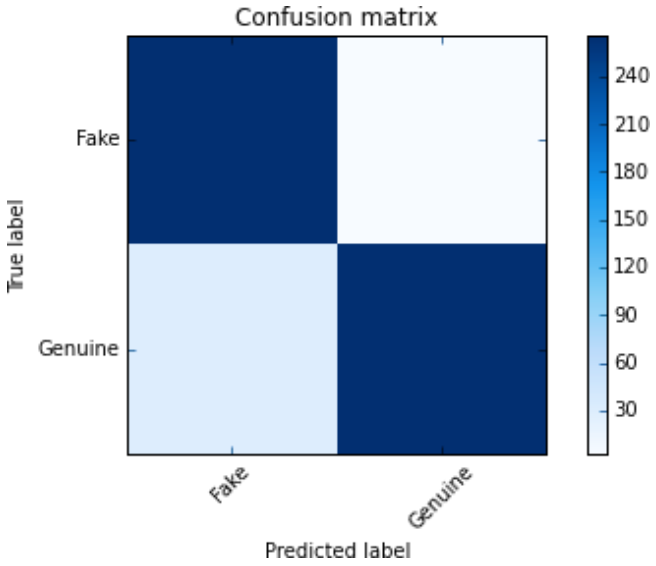

Fig.5. Confusion Matrix

VII. CONCLUSION

Finally, we draw the conclusion that further research is necessary to identify or detect the fake profiles generated by fake users. The bots are unable to distinguish between phony profiles produced by humans and those created by software. The Machine Learning Module is currently moving forward in time. Using data sets containing fake profiles, we can easily distinguish between actual and fraudulent profiles before identifying them as such. When the true existing data set is released, the model will then successfully detect whether the profile was created by a human and is authentic or fake. The detection of real and false social media profiles uses a variety of machine learning modules. For the detection of fake profiles, both supervised and unsupervised machine learning is used. These modules are successfully detecting fake accounts.

REFERENCES

[1] REFERENCES
[2] [1] P. K. Roy and S. Chahar, "Fake Profile Detection on Social Networking Websites: A Comprehensive Review," in IEEE

Transactions on Artificial Intelligence, vol. 1, no. 3, pp. 271-285, Dec. 2020, doi: 10.1109/TAI.2021.3064901.

[3] [2] E. Van Der Walt and J. Eloff, "Using Machine Learning to Detect Fake Identities: Bots vs Humans," in IEEE Access, vol. 6, pp. 6540-6549, 2018, doi: 10.1109/ACCESS.2018.2796018.

[4] [3] K. A. Qureshi, R. A. S. Malick, M. Sabih and H. Cherifi, "Complex Network and Source Inspired COVID-19 Fake News Classification on Twitter," in IEEE Access, vol. 9, pp. 139636-139656, 2021, doi: 10.1109/ACCESS.2021.3119404.

[5] [4] G. Sansonetti, F. Gasparetti, G. D'aniello and A. Micarelli, "Unreliable Users Detection in Social Media: Deep Learning Techniques for Automatic Detection," in IEEE Access, vol. 8, pp. 213154-213167, 2020, doi: 10.1109/ACCESS.2020.3040604.

[6] [5] M. Fazil, A. K. Sah and M. Abulaish, "DeepSBD: A Deep Neural Network Model With Attention Mechanism for SocialBot Detection," in IEEE Transactions on Information Forensics and Security, vol. 16, pp. 4211-4223, 2021, doi: 10.1109/TIFS.2021.3102498.

[7] [6] Akshima, D. Chang, A. Goel, S. Mishra and S. K. Sanadhya, "Generation of Secure and Reliable Honeywords, Preventing False Detection," in IEEE Transactions on Dependable and Secure Computing, vol. 16, no. 5, pp. 757-769, 1 Sept.-Oct. 2019, doi: 10.1109/TDSC.2018.2824323.

[8] [7] M. F. Hashmi, B. K. K. Ashish, A. G. Keskar, N. D. Bokde, J. H. Yoon and Z. W. Geem, "An Exploratory Analysis on Visual Counterfeits Using Conv-LSTM Hybrid Architecture," in IEEE Access, vol. 8, pp. 101293-101308, 2020, doi: 10.1109/ACCESS.2020.2998330.

[9] [8] W. Shahid, Y. Li, D. Staples, G. Amin, S. Hakak and A. Ghorbani, "Are You a Cyborg, Bot or Human?—A Survey on Detecting Fake News Spreaders," in IEEE Access, vol. 10, pp. 27069-27083, 2022, doi: 10.1109/ACCESS.2022.3157724.

[10] [9] G. Suarez-Tangil, M. Edwards, C. Peersman, G. Stringhini, A. Rashid and M. Whitty, "Automatically Dismantling Online Dating Fraud," in IEEE Transactions on Information Forensics and Security, vol. 15, pp. 1128-1137, 2020, doi: 10.1109/TIFS.2019.2930479.

[10] F. Zhang and S. Wang, "Detecting Group Shilling Attacks in Online Recommender Systems Based on Bisecting K-Means Clustering," in IEEE Transactions on Computational Social Systems, vol. 7, no. 5, pp. 1189-1199, Oct. 2020, doi: 10.1109/TCSS.2020.3013878.