MALICIOUS ACCOUNT DETECTION USING ANN

By

ARYAN PANDEY – 20BLC1087

ANANYA B – 20BLC1016

SAKSHI AGNIHOTRI – 20BEC1330

UTKARSH JAIN – 20BEC1160

ABSTRACT

Social media is a platform which has gained the most attention in the past few decades because of this high usage in social media by having many subscribers, followers, etc., the user's personal information is extracted by the attackers. They steal information, spread false news, and do malicious activities. This paper will summarize the solution for this problem using artificial neural network. It will be efficiently used to block/detect the fake accounts on social media comments section. The main objective is to sort malicious and genuine accounts which is a classification or clustering problem. With the provided solution automatically, accounts will be sorted instead of manual methods.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
1	**INTRODUCTION**	8
2	**RELATED WORK**	9-10
	2.1 Sequential Model	9
	2.2 Artificial Neural Network	10
3	**MALICIOUS ACCOUNT DETECTION USING ANN**	11-15
	3.1 Feature Detection	12
	3.2 Feature Reduction	12
	3.3 Dataset	13
	3.3.1 Training Data	13
	3.3.2 Testing Data	13
	3.4 Finding Fake Accounts	14
4	**SIMULATION/IMPLEMENTION RESULTS**	16-20
	4.1 Result and Analysis	16
	4.1.1 Feature Importance through Graph and Analysis	16
	4.1.2 Feature Significance Comparison	18
	4.1.3 Model Summary	19
	4.1.4 Test Summary	20
5	**CONCLUSION AND FUTURE WORK**	21
	5.1 Conclusion	21
	5.2 Future Works	21
7	**REFERENCES**	22-23

LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
1	Model Summary Table	19

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
1	Process Flow in Sequential Model	10
2	Neural Network Internetwork connection with Dropout and Dense layer	11
3	Depiction of Neural Network with Dropout Layers	12
4	Sample of Training Dataset	13
5	Sample of Testing Dataset	14
6	Process Flow of Detecting the Malicious Account from the real one	15
7	Block Diagram of Hidden Layers	15
8	Count of Malicious and Not Malicious account	16
9	Count of private and public account	16
10	Count of profile pic is there or not	17
11	Histogram of Density v/s Length of Username	17
12	Heat Map of Correlation Matrix	18
13	Model Loss Progression During Training/Validation	19
14	Heatmap of Predicted Values while Testing	20

CHAPTER I

INTRODUCTION

Online interpersonal organization are one of the most populous and quick data propagation application in the ongoing period. Considerations of various explores; consequently, broad investigates have been done on the ID of not normal records which has been pulled by Eliminating counterfeit records. Proposed work aims to build up a framework that depends on information mining strategies which may assist with finding the phony record in web-based media networks. These loopholes are widely abused since the detection of fake human accounts is laborious. Such kind of forging of identity can be widely used for other purposes like:

- People will giving their accurate details in the privacy policies of social media is not the expectation being kept. When someone blackmails or harasses a person for some specific reason is an example of cyber-bullying.
- Groups and people who hammer out their identities on social media are seeking to spread havoc in the society.
- To increase the popularity by making websites more gamified which helps in increasing the social rating sand popularity with more likes, followers and remarks, the process has been developed.
- These days fake accounts can be created very easily and can be done by anyone since it is effortless these days to buy Instagram and Twitter followers and likes online.

Many fake accounts are made by bots. To identify bots on social media, ML has been used but again ML is used for the identification of bots. Counterfeit accounts can be identified by different various methodologies by the likes neural network and support vector machine. A malicious profile detection method has been presented by the paper using algorithms like deep neural networks. The data is composed from Instagram Network. The feature set consists of attributes like profile pic, username length, full name words, full name length, username, description length, external URL, Private, #posts, #followers, #follows, fake.

CHAPTER II

RELATED WORKS

Some features which are extracted include frequency of friend requests, and the percentage of the accepted requests. These features are added to a classifier and this classifier has been trained based on machine learning techniques. Some of the most interesting features which are tested are average clickers in every session, the session length of using that social media site, the number of friend requests and the accepting the friend request rate, the number of posts or reel view rate and the shared contents.

B. Viswanath, et al. [1] utilizes PCA to identify features that best explains the prominent user behaviour. PCA does so by projecting high-dimensional data in low-dimensional subspace. To distinguish between anomalies and noise, bounds on L^ (2) norm in the residual subspace such that an operator specified fraction of the unlabelled training data is within the bound. The result obtained is 99% true positive and 0.7% false positive.

2.1 Sequential Model:

A Sequential model is appropriate for a plain stack of layers where each layer has exactly one input tensor and one output tensor. The model needs to know what input shape it should expect. For this reason, the first layer in a Sequential model needs to receive information about its input shape. There are several possible ways to do this:

Step 1: Pass an ‘input_shape’ argument to the first layer. This is a shape tuple. In ‘input_shape’, the batch dimension is not included.

Step 2: Pass instead a ‘batch_input_shape’ argument, where the batch dimension is included. This is useful for specifying a fixed batch size.

Step 3: Some 2D layers, such as Dense, support the specification of their input shape via the argument ‘input_dim’, and some 3D temporal layers support the arguments input dim and ‘input_length’.

Multiple Sequential instances can be merged into a single output via a Merge layer. The output is a layer that can be added as first layer in a new Sequential model. For instance, a model with two separate input branches getting merged as shown in the Figure 1.

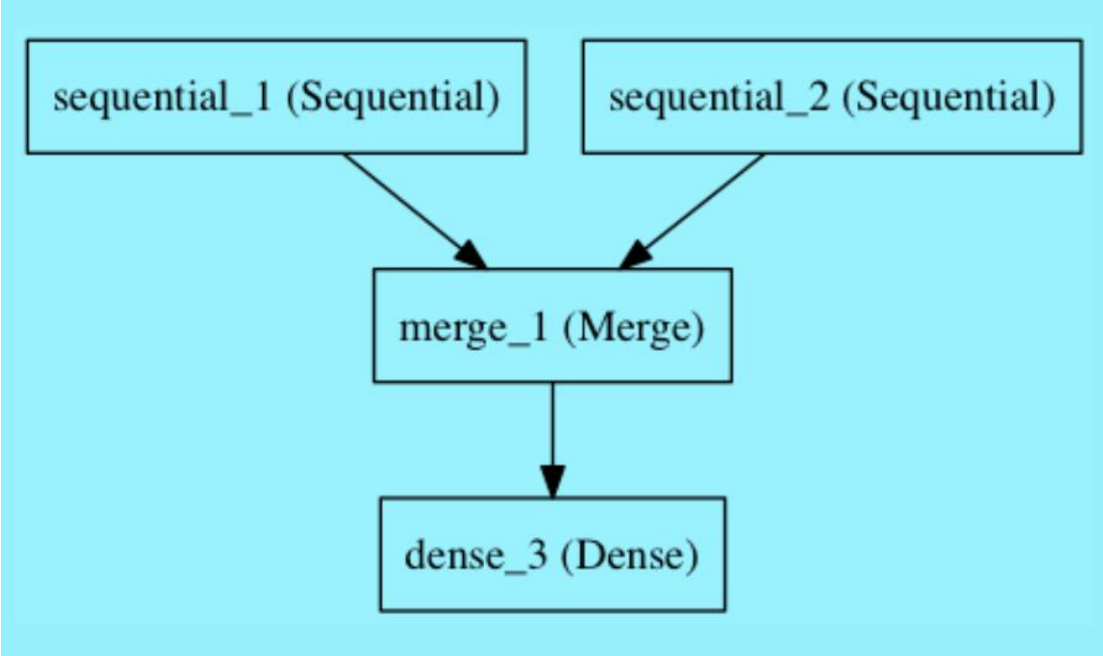

Figure 1: Process Flow in Sequential Model

2.2 Artificial Neural Networks:

Y. Boshmaf et al. [3] Even though feature-based detection scales to large OSNs, it could be circumvented. Attackers used to change content and activity patterns of their actions to avoid spam detection techniques. [6]. Feature-based detection does not provide any formal security guarantees and often results in a high false positive rate in practice. It involved how different machine learning techniques along with feature reduction methods are employed to efficiently solve the problem of malicious accounts over OSNs.

CHAPTER III

MALICIOUS ACCOUNT DETECTION USING ANN

In any neural network, a layer that is densely connected to its preceding layer means that every neuron in the layer is connected to every other neuron in the layer above it. Dropout is a regularization technique for neural network models. It is a simple way to prevent neural networks from Overfitting. It is a technique where randomly selected neurons are ignored during training as shown in the below Figure 2:

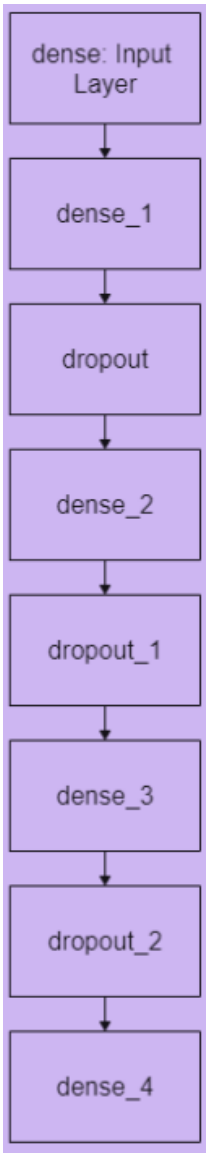

Figure 2: Neural Network Internetwork connection with Dropout and Dense layer

3.1 Feature Detection:

This is a classifying technique used to determine if the account is fake based on the user activity history and the present activity status. This also include user logs and their profiles. Some features which are extracted include frequency of friend requests, and the percentage of the accepted requests. These features are added to a classifier and this classifier has been trained based on machine learning techniques. Some of the most interesting features which are tested are average clickers in every session, the session length of using that social media site, the number of friend requests and the accepting the friend request rate, the number of posts or reel view rate and the shared contents.

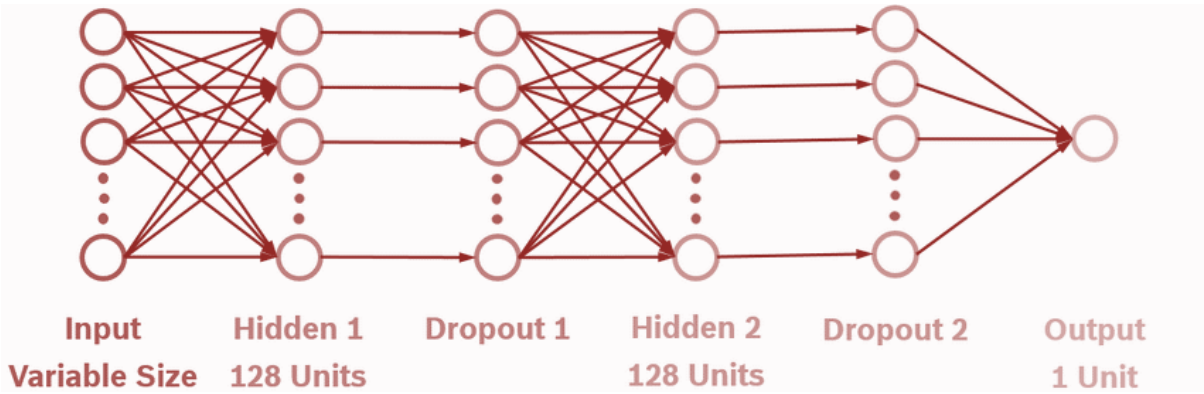

Figure 3: Depiction of Neural Network with Dropout Layers

3.2 Feature Reduction:

Usually, we come across many data sets with humongous high dimensional data. Most of the machine learning algorithms undergo over fitting. As there are many types of data and huge number of feature selection so there is a huge possibility of classifying the wrong data and giving low accuracy results. In high dimension data there is huge noise leading to give low accuracy data. So, dimension reduction plays an important role. Always it must be noted to select a subset and then apply classifying techniques to avoid noise.

3.3 Dataset: -

We used Exploratory Data Analysis because it refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies and basically to summarize their main characteristics, often using statistical graphics and other visualization methods. We have split the data into 70-30 ratio in which the 70 is training and 30 is testing as shown below in Figure 4 and Figure 5 respectively.

3.3.1 Training Data: -

It has 12 columns and 576 rows in testing data and they comprise of the basic information which are needed for training the model. Few of the important fields which are needed are followers, posts, profile pic, full name words, username, any external link is available or not. The below figure 4, shows the sample of our data:

profile pic	nums/length username	fullname words	nums/length fullname	name==username	description length	external URL	private	#posts	#followers	#follows	fake
1	0.27	0	0	0	53	0	0	32	1000	955	0
1	0	2	0	0	44	0	0	286	2740	533	0
1	0.1	2	0	0	0	0	1	13	159	98	0
1	0	1	0	0	82	0	0	679	414	651	0
1	0	2	0	0	0	0	1	6	151	126	0
1	0	4	0	0	81	1	0	344	669987	150	0
1	0	2	0	0	50	0	0	16	122	177	0
1	0	2	0	0	0	0	0	33	1078	76	0
1	0	0	0	0	71	0	0	72	1824	2713	0
1	0	2	0	0	40	1	0	213	12945	813	0
1	0	2	0	0	54	0	0	648	9884	1173	0
1	0	2	0	0	54	1	0	76	1188	365	0
1	0	2	0	0	0	1	0	298	945	583	0
1	0	2	0	0	103	1	0	117	12033	248	0
1	0	2	0	0	98	1	0	487	1962	2701	0
1	0	3	0	0	46	0	0	254	50374	900	0
1	0	3	0	0	0	0	0	59	7007	289	0
1	0.29	3	0	0	48	0	0	1570	1128	694	0
1	0	2	0	0	63	1	0	378	34670	1878	0
1	0	2	0	0	106	1	0	526	2338	776	0
1	0	2	0	0	40	0	0	228	3516	999	0
1	0	1	0	0	35	1	1	35	1809	416	0
1	0	2	0	0	30	0	0	281	427	470	0

Figure 4: Sample of Training Dataset

3.3.2 Testing Data: -

It has 12 columns like training dataset and 120 sample data which would be tested against our trained model and help us know the accuracy of the model. It would determine if the account is malicious or not. The below figure 5, shows the sample of our data:

profile pic	nums/length username	fullname words	nums/length fullname	name==username	description length	external URL	private	#posts	#followers	#follows	fake
1	0.33	1	0.33	1	30	0	1	35	488	604	0
1	0	5	0	0	64	0	1	3	35	6	0
1	0	2	0	0	82	0	1	319	328	668	0
1	0	1	0	0	143	0	1	273	14890	7369	0
1	0.5	1	0	0	76	0	1	6	225	356	0
1	0	1	0	0	0	0	1	6	362	424	0
1	0	1	0	0	132	0	1	9	213	254	0
1	0	2	0	0	0	0	1	19	552	521	0
1	0	2	0	0	96	0	1	17	122	143	0
1	0	1	0	0	78	0	1	9	834	358	0
1	0	1	0	0	0	0	1	53	229	492	0
1	0.14	1	0	0	78	1	1	97	1913	436	0
1	0.14	2	0	0	61	0	1	17	200	437	0
1	0.33	2	0	0	45	0	1	8	130	622	0
1	0.1	2	0	0	43	0	0	60	192	141	0
1	0	2	0	0	56	0	1	51	498	337	0
1	0.33	2	0	0	86	0	1	25	96	499	0
1	0	1	0	0	97	0	1	188	202	605	0
1	0	3	0	0	46	0	1	590	175	199	0
1	0	2	0	0	39	0	1	251	223	694	0
1	0.5	1	0	0	0	0	1	0	189	276	0
1	0	0	0	0	28	0	1	58	486	862	0
1	0.22	2	0	0	63	0	1	46	464	367	0

Figure 5: Sample of Testing Dataset

3.4 Finding Fake Accounts:

Steps taken to find a malicious and a genuine account: -

Step 1: Choosing /Choice of the profile should be characterized.

Step 2: After choosing the profile the helpful data /highlights are separated and grouped

Step 3: The most important step now is the use of the classifier. The new information is now taken care of the classifier.

Step 4: Now the classifier decides if the taken profile is authentic or not.

Step 5: After characterization calculation which will profound neural organization and furthermore support vector machine is then confirmed and criticism is taken care of once again into the classifier

Step 6: As the number of training data sets or preparation to information builds, the classifier increases its preciseness in foreseeing the phony profiles.

We find results up to 96% accuracy which is efficient. To summarize a large amount of data where the goal is to see patterns, we use the Correlation Matrix. It helps us to know which variable is having a high or low correlation in respect to another variable. As shown below in the figure 6, the detailed flow of our process is depicted through the flow chart:

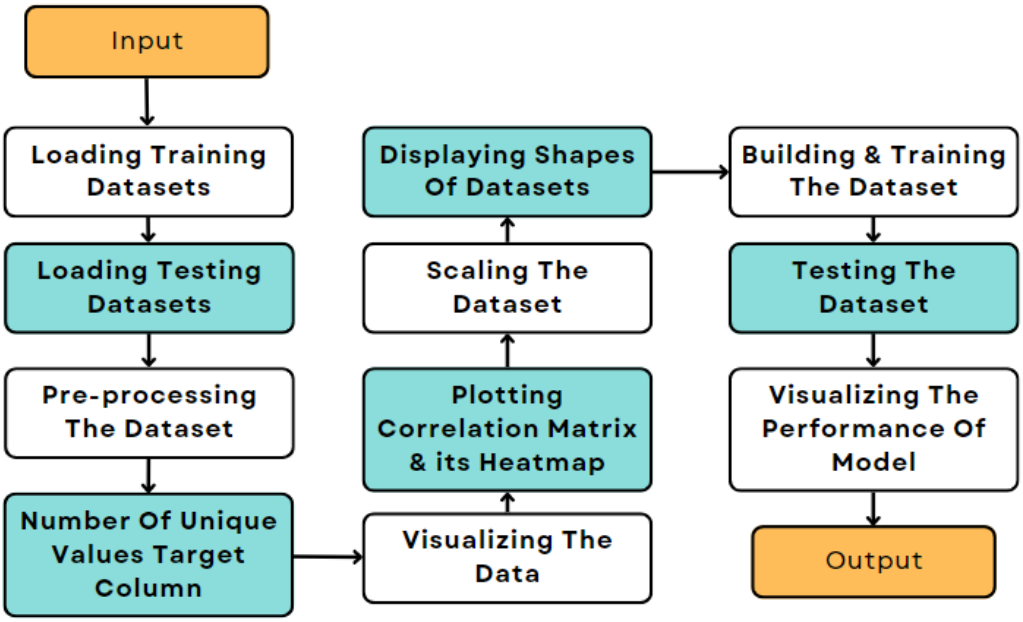

Figure 6: Process Flow of Detecting the Malicious Account from the real one

Hidden layers make a neural network as superior to ML algorithms. These layers are place in between the input and output. These layers are not visible to the external system, thus making them private for neural networks. A shown below in figure 7, 'Σ' is used to define weighted sum where 'σ(Σ)' is used to define the activation function.

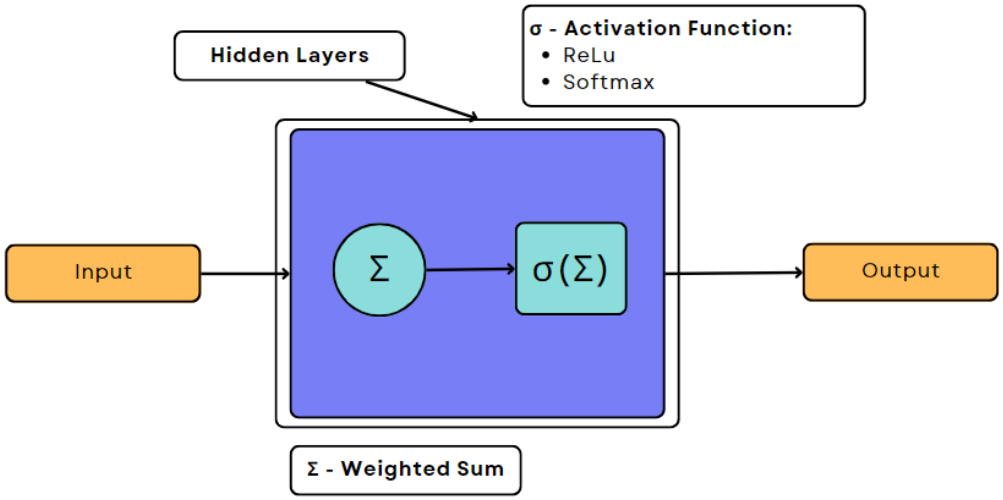

Figure 7: Block Diagram of Hidden Layers

CHAPTER IV

SIMULATION/IMPLEMENTATION RESULTS

4.1 Result and Analysis: -

4.1.1 Feature Importance through Graph and Analysis: -

Visualizing the data, the column data of all features is very necessary and understand it completely. Some features are more necessary and they shouldn't be dropped out of our training as they support the model and increase the accurate rate of the model. One of the most important columns was "Fake" as shown below in Figure 8 the count is defined as the number of 0's and 1's in the table value. 0's mean that account was not malicious and 1's mean that the account is malicious.

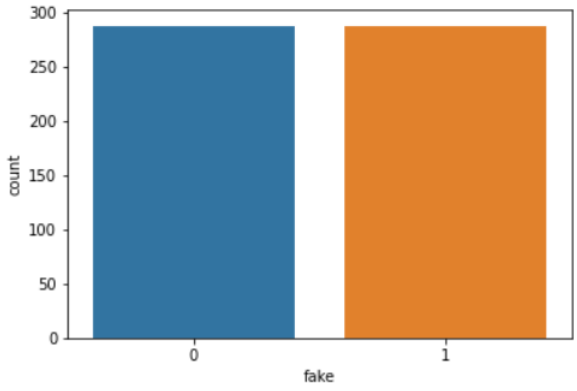

Figure 8: Count of Malicious and Not Malicious account

More plots are plotted for visualizations as it helps us know the importance and the below plot tells us if the account is private or not, '1' being private and '0' being public.

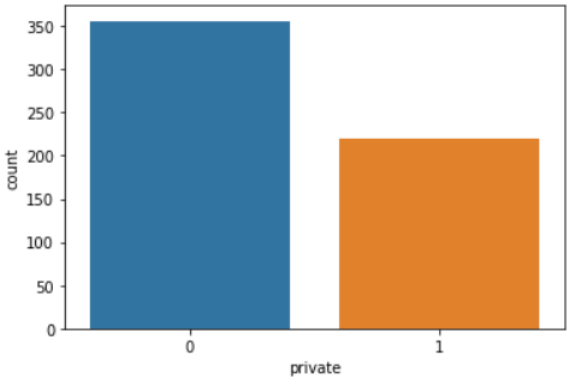

Figure 9: Count of private and public account

The Figure 10 below tells if the account has a profile picture or not. 1 being true and 0 being false.

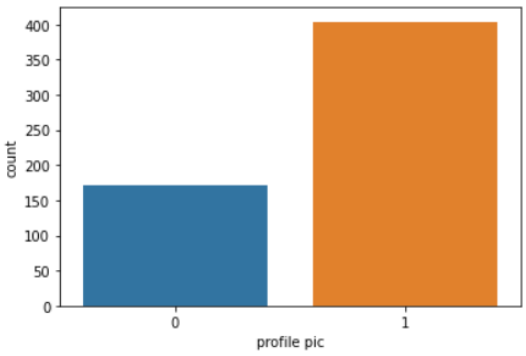

Figure 10: Count of profile pic is there or not

The plot below depicts the comparison between the number of users and the length of their usernames. Through this we can differentiate between an original and malicious account by comparing the length of their username with the density of users that have the same length of username.

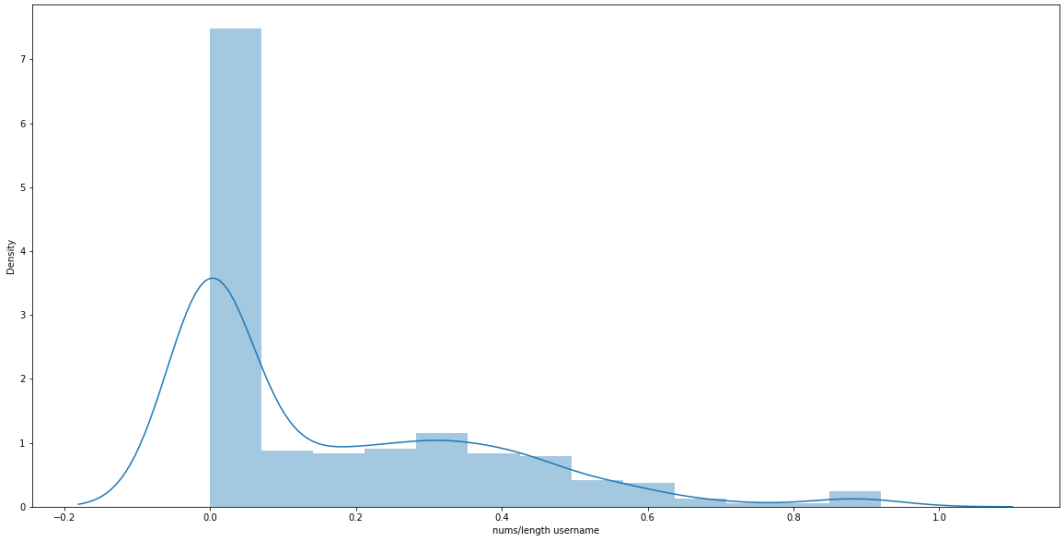

Figure 11: Histogram of Density v/s Length of Username

4.1.2 Feature Significance Comparison: -

Through the heat map below we can easily filter out the features that we need in order to train and test our model at maximum efficiency. Heat Map help us distinguish between the important features with some noise features very easily.

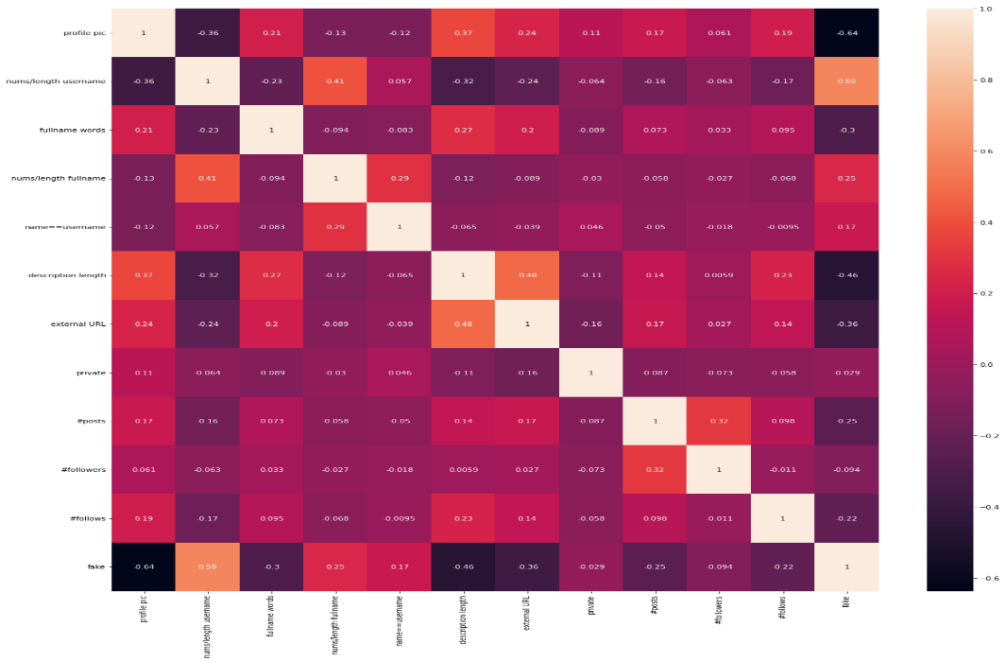

Figure 12: Heat Map of Correlation Matrix

Hidden layers are only necessary in artificial neural networks when non-linear data separation is necessary. To obtain the optimum decision boundary, we must employ hidden layers. Even if we choose not to use hidden layers in this scenario, the classification accuracy will suffer. Therefore, using concealed layers is preferable. Progression of Model Losses A number reflecting how poorly the model predicted on a single case is provided during training and validation. If the model's prediction is exact, there will be no loss; if not, there will be a bigger loss. Finding a set of weights and biases that, on average, have low loss across all examples is the aim of training a model.

Figure 13: Model Loss Progression During Training/Validation

4.1.3 Model Summary:

A model Summary is employed to describe how well a classification system performed on a test data. It displays and summarises a classification algorithm's performance through factors like Precision, Recall, Fi-Score, Support and it also displays the accuracy as shown in the below table which is a summary for the sequential model. The accuracy we achieved is 70% also shown below in the Table 1.

	precision	recall	Fi-score	Support
0	0.75	0.60	0.67	60
1	0.67	0.80	0.73	60
Accuracy			0.70	120
Macro Avg.	0.71	0.70	0.70	120
Weighted Avg.	0.71	0.70	0.70	120

Table 1: - Model Summary Table

4.1.4 Test Summary:

This is due to the difficulty in reproducing machine learning findings. Think about what happens when you train a neural network. The weights are initially initialised randomly when training begins, and gradient descent-based algorithms will undoubtedly alter the weights depending on where you start, but the weights are always different after training. Some weight combinations will be superior than others since they are different from one another. Granted, if a set of hyper parameters worked well during training, they are likely to work well during testing as well, with outcomes that are similar. For repeatability purposes, you frequently establish a random seed at the program's beginning. Additionally, you can save the model weights after training so that you don't need to retrain a neural network that performed well. It is prepared to foresee the future.

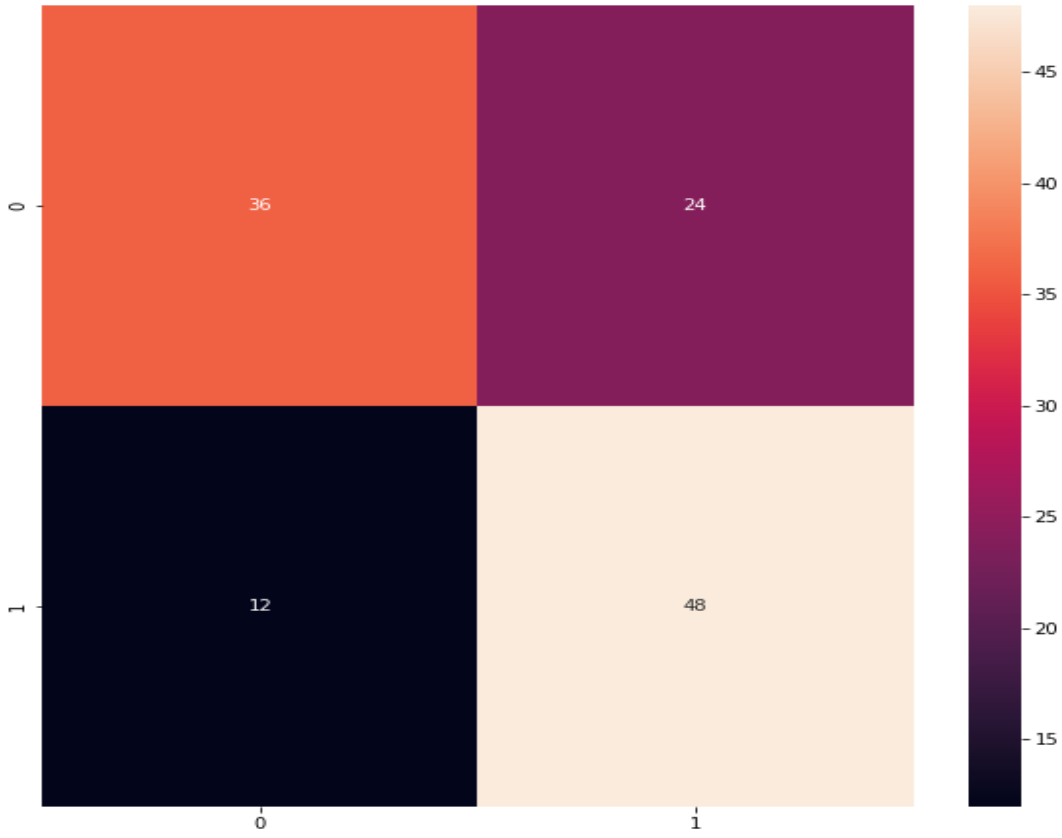

Figure 14: Heatmap of Predicted Values while Testing

CHAPTER V

CONCLUSION AND FUTURE WORKS

5.1 Conclusion:

This Paper, presented a machine learning pipeline for detecting bogus accounts in online social networks. Instead of making a prediction for each individual account, this system classifies groups of fake accounts to determine if they were created by the same actor. The system used in production to find and restrict over 1 million accounts, and evaluation of in-sample and out-of-sample data revealed strong performance. Testing of the framework on clusters formed by simple grouping based on registration date and IP address in this paper. In the future the aim is to test our model on clusters formed by grouping on other features, such as ISP, and other time periods, such as week or month. Another promising area of research is the use of more advanced clustering algorithms such as k-means or hierarchical clustering. While these approaches may be beneficial, they present challenges when applied at scale: kmeans may require too many clusters (i.e., a large value of k) to produce useful results, and data clustering may be too time-consuming for classifying millions of accounts in an Online Social Network. One important direction for future work in modelling is to apply feature sets used in other spam detection models, enabling multi-model ensemble prediction. Another method is to harden the system against adversarial attacks, such as a botnet that diversifies all features or an attacker who learns from failures.

5.2 Future works:

Because we have limited data to train the classifier, our approach faces a high variance problem, as shown by the learning curve below. High variance problems are typically mitigated by increasing the dataset size, which should not be a major concern for Social Networks. Organizations with relatively large datasets

REFERENCES

[1] Kuncham Sreenivasa Rao, Bandrapalli Deevena Raju, Sreeram Gutha, “DETECTING FAKE ACCOUNT ON SOCIAL MEDIA USING MACHINE LEARNING ALGORITHMS”, International Journal of Control and Automation · April 2020.

[2] Avik Kumar Ghosh, P. Visalakshi, Hiren J Doshi, “Fraudulent Users Detection on Social Media Platforms Using Machine Learning Techniques”, Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4170 – 4174.

[3] Y. Boshmaf, D. Logothetis, G. Siganos, J. Lería, J. Lorenzo, M. Ripeanu, et al., "Íntegro: Leveraging victim prediction for robust fake account detection in large scale osns", *Computers & Security*, vol. 61, pp. 142-168, 2016.

[4] Sowmya P, Madhumita Chatterjee, “Detection of Fake and Clone accounts in Twitter using Classification and Distance Measure Algorithms”, International Conference on Communication and Signal Processing, July 28 - 30, 2020, India.

[5] Sarah Khaled, Neamat El-Tazi, Hoda M. O. Mokhtar, “Detecting Fake Accounts on Social Media”, 2018 IEEE International Conference on Big Data (Big Data).

[6] Neha M. Yadav, Prof. Dr. P. N. Chatur, “Compromised Account Detection and Prevention by Profiling Social Behavior and FASS Key Concept”, 2017 International Conference on Recent Trends in Electrical, Electronics and Computing Technologies.

[7] E. Van Der Walt and J. Eloff, “Using Machine Learning to Detect Fake Identities: Bots vs Humans,” IEEE Access, vol. 6, pp. 6540– 6549, 2018.

[8] Hridoy Sankar Dutta, Vishal Raj Dutta, Aditya Adhikary, and Tanmoy Chakraborty, “HawkesEye: Detecting Fake Retweeters Using Hawkes Process and Topic Modeling”, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 15, 2020.

[9] Y. Zhang and J. Lu, “Discover millions of fake followers in weibo,” Social Network Analysis and Mining, vol. 6, no. 1, p. 16, 2016.

[10] Karishma Anklesaria, Zeel Desai, Vikram Kulkarni, Harish Balasubramaniam, "A Survey on Machine Learning Algorithms for Detecting Fake Instagram Accounts ",2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN).

[11] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro, "Aiding the Detection of Fake Accounts in Large Scale Social Online Services", In USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI), 2012.

[12] I. Sen, A. Aggarwal, S. Mian, S. Singh, P. Kumaraguru, and A. Datta, "Worth its weight in likes: Towards detecting fake likes on instagram." ,in WebSci, 2018, pp. 205–209.

[13] Yeh-Cheng Chen, Shyhtsun Felix Wu, "Fake Buster:A Robust Fake Account Detection by Activity Analysis", 2018 9th International Symposium on Parallel Architectures, Algorithms and Programming (PAAP).

[14] Pradeep Kumar Roy and Shivam Chahar, "Fake Profile Detection on Social Networking Websites: A Comprehensive Review, "IEEE TRANSACTIONS ON ARTIFICIAL INTELLIGENCE, VOL. 1, NO. 3, DECEMBER 2020.

[15] P. V. Savyan and S. M. S. Bhanu, "Behaviour profiling of reactions in Facebook posts for anomaly detection," in *Proc. 9th Int. Conf. Adv.Comput*, 2017, pp. 220–226.