

## FAKE PROFILE IDENTIFICATION USING ARTIFICIAL NEURAL NETWORK

Ch Bhavani<sup>1</sup>, P Manasa<sup>2</sup>, J. Anusha<sup>3</sup>

(1), (2),(3) Assistant Professor, Department of Electronics & Communication Engineering, KITS(S), Huzurabad, Karimangar, T.S.

Mail ID: bhavanipriya06@gmail.com

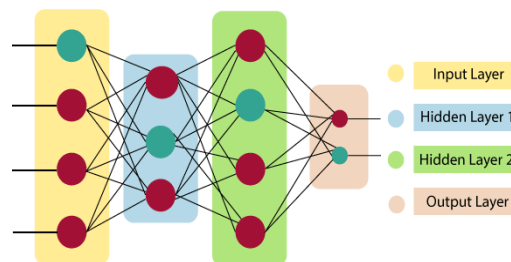
### ABSTRACT

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 Face book 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 Support Vector Machine 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.

### Introduction:

Artificial Neural Networks (ANN) are algorithms based on brain function and are used to model complicated patterns and forecast issues. The Artificial Neural Network (ANN) is a deep learning method that arose from the concept of the human brain Biological Neural Networks. The development of ANN was the result of an attempt to replicate the workings of the human brain. The workings of ANN are extremely similar to those of biological neural networks, although they are not identical. ANN algorithm accepts only numeric and structured data.

To understand the concept of the architecture of an artificial neural network, we have to understand what a neural network consists of. In order to define a neural network that consists of a large number of artificial neurons, which are termed units arranged in a sequence of layers. Let us look at various types of layers available in an artificial neural network.



Artificial Neural Network primarily consists of three layers:

#### Input layer:

As the name suggests, it accepts inputs in several different formats provided by the programmer.

#### Hidden Layer:

The hidden layer presents in-between input and output layers. It performs all the calculations to find hidden features and patterns.

#### Output Layer:

The input goes through a series of transformations using the hidden layer, which finally results in output that is conveyed using this layer.

The artificial neural network takes input and computes the weighted sum of the inputs and

includes a bias. This computation is represented in the form of a transfer function.

$$\sum_{i=1}^n W_i * X_i + b$$

It determines weighted total is passed as an input to an activation function to produce the output. Activation functions choose whether a node should fire or not. Only those who are fired make it to the output layer.

### **Literature Survey:**

Sybil rank was designed in late 2012, to efficiently identify fake profiles through a ranking graph-based system [5]. The algorithm uses a seed selection method combined with early terminated random walks to propagate trust [5]. Its computational cost is measured in  $O(n \log n)$ . Profiles are ranked according to the number of interactions, tags, wall posts, and friends over time. Profiles that have a high rank are considered to be real with fake profiles having a low rank in the system. Unfortunately, this technique was found to be mostly unreliable because it failed to take into account the possibility that real profiles can be ranked low and fake profiles can be ranked high.

Sarcode and Mishra proposed a different approach which is a sequence of steps to detect fake profiles [6]. They used the Facebook graph API tool to gain access to numerous profiles and wrote a script to extract the viewed information.

Later on, this extracted information forms the attributes the classifier will use in their algorithm. First, the data is in JSON format, which is further parsed to a structured format (CSV) that is easier readable by machine learning techniques.

These separated values will later make the classifier more efficient. The authors tried unsupervised and also supervised machine learning techniques. In this case, supervised machine learning techniques had a higher accuracy rate of almost 98%. For supervised machine learning, they split up the dataset into training and testing sets. They used 80% of the samples to train the classifier and the rest to test it. After the algorithm runs, there is feedback provided to the profile, requiring it to submit identification to prove it is not a fake profile [6].

Profiles are processed on mass to extract features. Resilient Back Propagation algorithm in neural networks algorithm combined with support vector machines is used in the classification of fake profiles. Sybil Frame uses multi-stage level classification. Approaches include content-based and structure based. Content-based approach explores the dataset and extracts information used to calculate prior information about nodes and edges.

Structure-based approach correlates nodes using Markov random field and loopy belief propagation which employs previous information. The content-based approach is used in the first stage of Sybil Frame and Structure-based approach is used in the second stage of Sybil Frame technique. Clickstreams are analyzed, and Friend recommendations are examined in stage I. Vote Trust uses a voting-based system that pulls user activities to find fake profiles using trust-based vote signment and global votes total. It is considered as the first line of defense due to limitations which include real accounts that were already compromised being sold.

### **Proposed System:**

In our solution, we use machine learning, namely an artificial neural network to determine what are the chances that a friend request is authentic or not. Each equation at each neuron (node) is

put through a Sigmoid function to keep the results between the interval of 0.0 and 1.0. At the output end, this could easily be multiplied by 100 to give us the possible percentage that it is a malicious request. Our solution would be only one deep neural network, meaning it only has a single hidden layer. Each input neuron would be a different, previously chosen feature of each profile converted into a numerical value (e.g., gender as a binary number, female 0 and male 1) and if needed, divided by an arbitrary number (e.g., age is always divided by 100) to minimize one feature having more influence on the result than the other.

The neurons represent nodes. Each node would be responsible for exactly one decision-making process. Each object has a weight and bias that in turn would help in the decision-making process. The output would be the possibility in the percentage that the friend request is not from a real person.

## **ALGORITHM**

### **Naive Bayes Algorithm:**

In machine learning, naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naive) independence assumptions between the features. Naive Bayes has been studied extensively since the 1950s.

It was introduced under a different name into the text retrieval community in the early 1960s, and remains a popular (baseline) method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or legitimate, sports or politics, etc.) with word frequencies as the features.

With appropriate pre-processing, it is competitive in this domain with more advanced methods including support vector machines. It also finds application in automatic medical diagnosis.[3] Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem.

Maximum-likelihood training can be done by evaluating a closed-form expression, 718 which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers. In the statistics and computer science literature, naive Bayes models are known under a variety of names, including simple Bayes and independence Bayes. All these names reference the use of Bayes' theorem in the classifier's decision rule, but naive Bayes is not (necessarily) a Bayesian method.

## **REQUIREMENT ANALYSIS**

### **Preliminary Investigation:**

The first and foremost strategy for development of a project starts from the thought of designing a mail enabled platform for a small firm in which it is easy and convenient of sending and receiving messages, there is a search engine, address book and also including some entertaining games. When it is approved by the organization and our project guide the first activity, i.e. preliminary investigation begins. The activity has three parts:

- Request Clarification
- Feasibility Study
- Request Approval

### **Request Clarification:**

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system

requires.

Here our project is basically meant for users within the company whose systems can be interconnected by the Local Area Network (LAN). In today's busy schedule man need everything should be provided in a readymade manner. So, taking into consideration of the vastly use of the net in day-to-day life, the corresponding development of the portal came intoexistence.

### **Feasibility Report:**

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

- Operational Feasibility
- Economic Feasibility
- Technical Feasibility

### **Operational Feasibility:**

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

### **Economic Feasibility:**

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer-based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at any time. The Virtual Private Network is to be developed using the existing resources of the organization. So, the project is economically feasible.

### **Technical Feasibility:**

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and Web Logic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

### **Request Approval:**

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible.

Inputs are supposed to be designed effectively so that the errors occurring while feeding

are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.

Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided within an option to select an appropriate input from various alternatives related to the field in certain cases.

Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

### **Output Design:**

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rest with the administrator only.

The application starts running when it is executed for the first time. The server has to be started and then the internet explorer is used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

## **SYSTEM DESIGN**

### **MODULES:**

**Admin Module:** Admin will login to application by using username as 'admin' and password as 'admin' and then perform below actions.

a) **Generate ANN Train Model:** Admin will upload profile dataset to ANN algorithm to build train model. This train model can be used to predict fake or genuine account by taking new account test data.

b) **View ANN Train Dataset:** Using this module admin can view all dataset used to train ANN model.

**User Module:** Any user can use this application and enter test data of new account and call ANN algorithm. ANN algorithm will take new test data and applied train model to predict

whether given test data contains fake or genuine details.

**DIAGRAMS**

**Architecture Diagram**

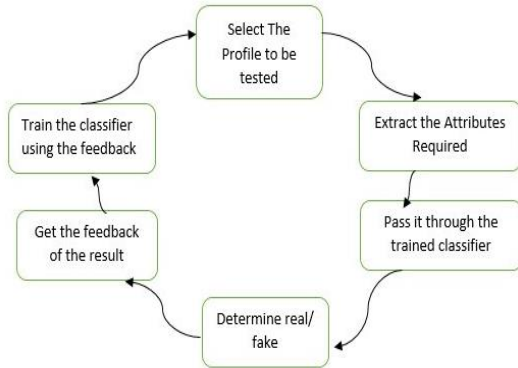


Fig. 1. Architecture Diagram

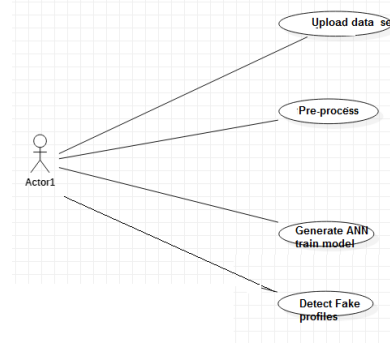


Fig 2. Use case Diagram

**USE CASE DIAGRAM**

The use case diagram is used to represent all the functional use cases that are involved in the project. The above diagram represents the main two actors in the project, they are

**SEQUENCE DIAGRAM**

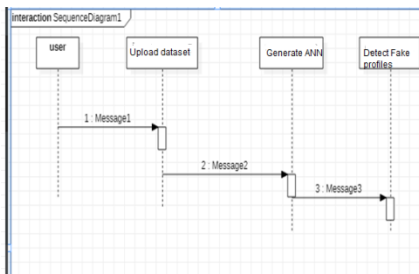


Fig 3. Sequence diagram

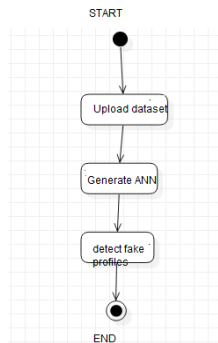


Fig. 4. Activity Diagram

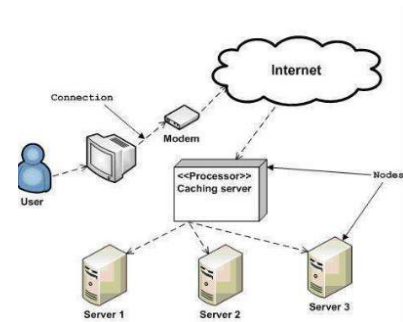


Fig. 5. Deployment Diagram

The above diagram represents the sequence of flow of actions in the system.

Here the use send a message to upload dataset again from upload dataset another message send to ANN then from ANN next message send to detect fake profiles.

**TESTING**

**MANUAL TESTING:**

The various types of testing are

**White Box Testing:**

- It is also called as glass box testing. It is a test case design method that uses the control structure of the procedural design to drive the test case.
- Using white box testing method, the software engineer can drive the test cases that
  1. Guarantee that all independent parts within a module have been exercised.
  2. Exercise all logical decisions on their true and false sides.

**Black Box Testing:**

- It's also called as behavioral testing. It focuses on the functional requirements of the software.
- It is a complementary approach that is likely to uncover a different class of errors than white box errors.
- A black box testing enables a software engineering to derive a set of input conditions that will fully exercise all functional requirements of a program.

**Alpha Testing:**

Alpha testing is the software prototype stage when the software is the first able to run. It will not have all the intended functionality, but it will have core functions and will be able to accept the input and generate outputs. An alpha test usually takes place in the developer's offices on a separate system.

**Beta Testing:**

The beta test is a "live" application of the software in an environment that cannot be controlled by the developer. The beta test is conducted at one or more customer sites by the end user of the software.

**Win Runner And Load Runner:**

We use WinRunner as a load testing tool operating at the GUI layer as it allows us to record and playback user actions from a vast variety of user applications as if a real user had manually executed those actions.

**SYSTEM TESTING:**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the

software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**TYPES OF TESTS****Unit Testing:**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration Testing:**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successful unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional test:**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input : identified classes of valid input must be accepted.
- Invalid Input : identified classes of invalid input must be rejected.
- Functions : identified functions must be exercised.
- Output : identified classes of application outputs must be exercised.
- Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

### **System Test:**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

### **White Box testing**

White Box testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software or at least its purpose . It is purpose it is used to test areas that cannot be reach from a black level.

### **Black Box Testing:**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

### **Integration Testing:**

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

### **Acceptance Testing:**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

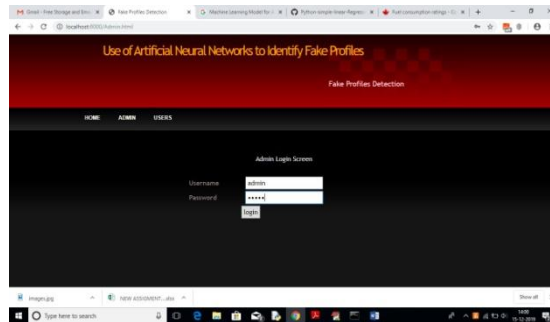
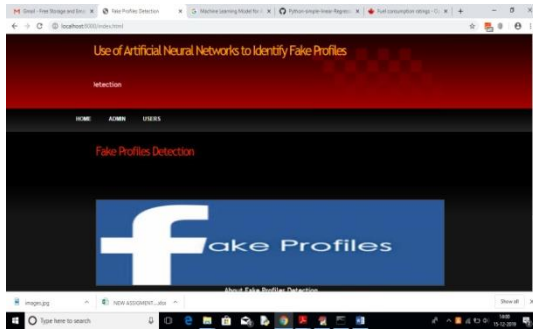
## **RESULTS**

### **Screen Captures**

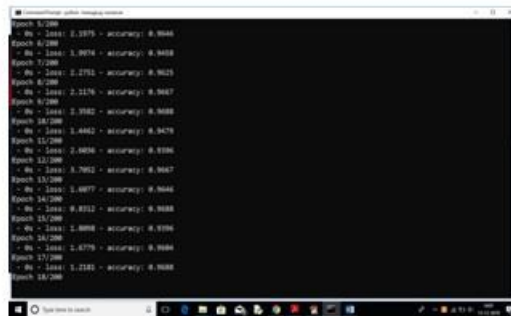
#### **User Login Screen:**



Deploy this application on DJANGO server and then run in browser enter URL as [‘http://localhost:8000/index.html’](http://localhost:8000/index.html) to get below screen  
 In the above screen click on the ‘ADMIN’ link to get below login screen



In above screen enter admin and admin as username and password to login as admin.  
 After login will get below screen



In the above black console, we can see all ANN details.

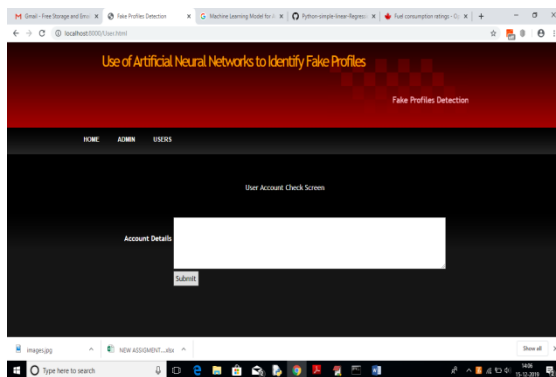
In the above screen click on ‘Generate ANN Train Model’ to generate the training model on the dataset. After clicking on that link, you can see the serverconsole to check ANN processing details with accuracy.

In above screen we can see ANN got 98% accuracy to train all Facebook profile. Now click on ‘View Ann Train Dataset’ link to view all dataset details



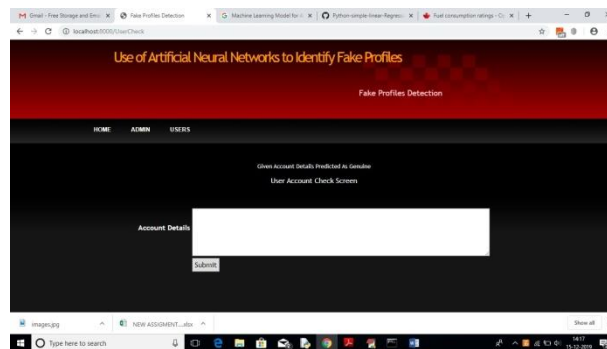
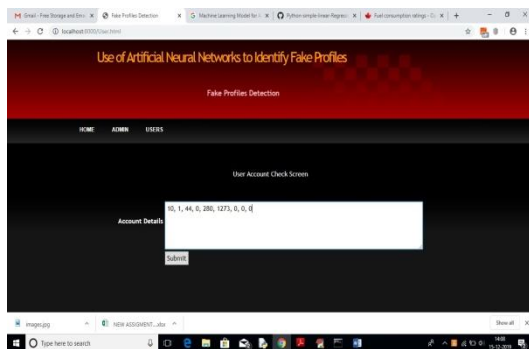
Account	Age	Gender	User Age	Link Description	Status Count	Friend Count	Location	Location IP	Profile Status
12	1	24	0	2170	205	0	0	0	0
12	0	24	0	2114	201	0	0	0	0
12	0	19	0	4024	97	0	0	0	0
12	1	16	0	4076	202	0	0	0	0
12	0	16	0	2174	44	0	0	0	0
12	0	44	0	3653	128	0	0	0	0
12	1	25	0	1182	168	0	0	0	0
12	1	16	0	4174	170	0	0	0	0
12	0	20	0	1942	168	0	0	0	0
12	0	24	0	1947	712	0	0	0	0
12	1	40	0	2741	274	0	0	0	0
12	1	16	0	1669	169	0	0	0	0
12	1	17	0	2473	117	0	0	0	0
12	0	24	0	4111	128	0	0	0	0
12	0	24	0	1441	201	0	0	0	0
12	0	20	0	1669	169	0	0	0	0
12	1	17	0	167	0	0	0	0	0
12	0	20	0	1493	148	0	0	0	0
12	1	24	0	1417	201	0	0	0	0
12	1	16	0	1742	171	0	0	0	0
12	1	22	0	276	40	0	0	0	0
12	1	44	0	1439	121	0	0	0	0
12	1	20	0	2444	201	0	0	0	0

In above screen we can see all train data and scroll down to view all records. Now ANN train model is ready and you can logout and click on ‘User’ link to get below screen.



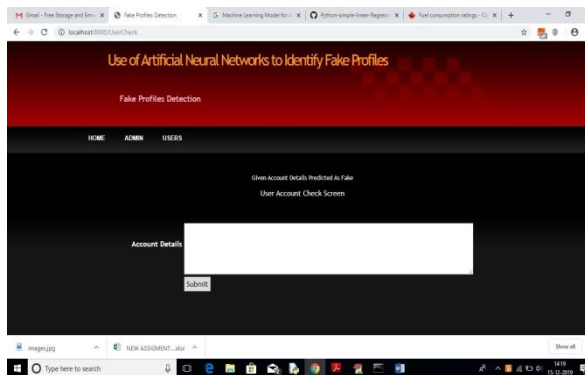
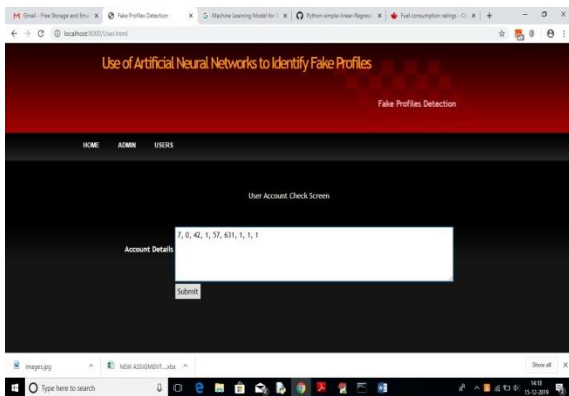
In above screen enter some test account details to get prediction/identification from ANN. You can use below records to check

- 10, 1, 44, 0, 280, 1273, 0, 0
- 10, 0, 54, 0, 5237, 241, 0, 0
- 7, 0, 42, 1, 57, 631, 1, 1
- 7, 1, 56, 1, 66, 623, 1, 1



For the above input will get the below result.

In above screen we can see the result predicted as genuine account



For the above account details, we got the below result as fake for the given account data

## CONCLUSION

In this paper, we use machine learning, namely an artificial neural network to determine what are the chances that a friend request is authentic or not. Each equation at each neuron (node) is put through a Sigmoid function. We use a training data set by Facebook or other social networks. This would allow the presented deep learning algorithm to learn the patterns of bot behavior by backpropagation, minimizing the final cost function and adjusting each neuron's weight and bias. In this paper, we outline the classes and libraries involved. We also discuss the sigmoid function and how are the weights determined and used. We also consider the parameters of the social network page which are the most important to our solution.

## **BIBLIOGRAPHY**

- [1] W. D. J. Salganik, M. J. Dodds, P. S., "Experimental study of inequality and unpredictability in an artificial cultural market," in *ASONAM*, 2016, pp. 529–532.
- [2] R. Peres, E. Muller, and V. Mahajan, "Innovation diffusion and new product growth models: A critical review and research directions," *International Journal of Research in Marketing*, vol. 27, no. 2, pp. 91 – 106, 2010.
- [3] L. A. Fourt and J. W. Woodlock, "Early prediction of market success for new grocery products." *Journal of Marketing*, vol. 25, no. 2, pp. 31 – 38, 1960.
- [4] B. W. O, "Reference group influence on product and brand purchase decisions," *Journal of Consumer Research*, vol. 9, pp. 183–194, 1982.
- [5] J. J. McAuley, C. Targett, Q. Shi, and A. van den Hengel, "Imagebased recommendations on styles and substitutes," in *SIGIR*, 2015, pp. 43–52.
- [6] E. M. Rogers, *Diffusion of Innovations*. New York: The Rise of High Technology Culture, 1983.
- [7] K. Sarkar and H. Sundaram, "How do we find early adopters who will guide a resource constrained network towards a desired distribution of behaviors?" in *CoRR*, 2013, p. 1303
- [8] D. Imamori and K. Tajima, "Predicting popularity of twitter accounts through the discovery of link-propagating early adopters," in *CoRR*, 2015, p. 1512.
- [9] X. Rong and Q. Mei, "Diffusion of innovations revisited: from social network to innovation network," in *CIKM*, 2013, pp. 499– 508.
- [10] I. Mele, F. Bonchi, and A. Gionis, "The early-adopter graph and its application to web- page recommendation," in *CIKM*, 2012, pp. 1682–1686.
- [11] Y.-F. Chen, "Herd behavior in purchasing books online," *Computers in Human Behavior*, vol. 24(5), pp. 1977–1992, 2008.
- [12] Banerjee, "A simple model of herd behaviour," *Quarterly Journal of Economics*, vol. 107, pp. 797–817, 1992.
- [13] A. S. E, "Studies of independence and conformity: I. a minority of one against a unanimous majority," *Psychological monographs: General and applied*, vol. 70(9), p. 1, 1956.
- [14] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *ICLR*, 2013.
- [15] A. Bordes, N. Usunier, A. Garc'ia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multirelational data," in *NIPS*, 2013, pp. 2787–2795.
- [16] A. S. E, "Studies of independence and conformity: I. a minority of one against a unanimous majority," *Psychological monographs: General and applied*, vol. 70(9), p. 1, 1956.
- [17] M. L. S. D. X. W. L. S. Mingliang Chen, Qingguo Ma, "The neural and psychological basis of herding in purchasing books online: an event-related potential study," *Cyberpsychology, Behavior, and Social Networking*, vol. 13(3), pp. 321–328, 2010.
- [18] V. G. D. W. Shih-Lun Tseng, Shuya Lu, "The effect of herding behavior on online review voting participation," in *AMCIS*, 2017.