Abstract

In this project, we employ machine learning, specifically a man-made neural network, to evaluate the probability that a social account is authentic or not. We also describe the relevant classes and libraries. We also examine the sigmoid function and the selection and application of the weights. Finally, we take into account the social network page's parameters, which are crucial to the offered solution.

1. Introduction

Facebook became the most popular social media platform in 2017 after reaching a total user base of 2.46 billion. Users' information is used by social media networks to generate cash. The usual user is unaware that when they use the service of a social media network, their rights are forfeited. Social media firms stand to earn greatly at the expense of the user. Facebook generates income from adverts and user data whenever a user posts new content, such as locations, images, likes, and dislikes. More specifically, the average American user produces around $26.76 every three months. When several users are involved, that sum soon grows.

In the current digital era, the growing reliance on technology has made the average person more susceptible to crimes like data breaches and potential identity theft. These attacks can happen suddenly and frequently without warning to those who have experienced a data breach. Social networks currently have little reason to strengthen their data security. These hacks frequently target Twitter and Facebook, among other social media platforms. Banks and other financial organisations will also be on their radar.

Social media platforms getting hacked seem to be a newsworthy topic every day. About 50 million Facebook users were impacted by a recent knowledge breach. Facebook outlines a number of clearly stated provisions that detail how they use user data. The policy does very little, if anything, to stop the ongoing invasion of privacy and security. The built-in security measures on Facebook appear to be overcome by fake profiles.  
  
Bots and phoney profiles represent the opposite threat of personal information being collected for illegal reasons. Bots are computer programmes that collect data about users without their knowledge. Web scraping is the term for this activity. Even worse, this action is legal. On social networks, bots are frequently concealed or seem as phoney friend requests to get private data.

This paper's proposed approach aims to highlight the risks posed by a bot that poses as a false social media presence. An algorithm would be available to implement this solution. Python is the language we decided to utilise. The algorithm would be prepared to determine whether a user is receiving a friend request from a legitimate person, a bot, or a phoney friend request that is gathering information. Since we might need a training dataset from the social media companies to build our model and then determine if the profiles are phoney or real, our algorithm would function with their help. The method might potentially be used as a standard layer on the user's web browser as a browser plug-in.

2. Literature Survey (list of papers referred )

Sybil rank, a ranking graph-based method, was created in late 2012 to effectively identify fraudulent profiles. To spread trust, the algorithm combines an early terminated random walk technique with seed selection. The computational expense is expressed in O (nlogn). According to the quantity of interactions, tags, wall postings, and friends throughout time, profiles are ranked. High-ranking profiles are taken into account to be genuine, whereas low-ranking ones are deemed to be false. Unfortunately, it was discovered that this system was largely unreliable because it overlooked the possibility that actual accounts may be ranked poorly and false profiles could be ranked well.

A unique method for identifying bogus profiles was proposed by Sarode and Mishra. They created a script to retrieve the seen data and used the Facebook graph API tool to gain access to many profiles. This gathered data is later used to create the features that the classifier will incorporate into their algorithm. The information is first in JSON format, which is then further converted to a structured format (CSV) that is simpler for machine learning techniques to understand. Later on, the classifier will function more effectively thanks to these comma separated values. The authors experimented with both supervised and unsupervised machine learning methods. Supervised machine learning algorithms performed better in this example, with an accuracy rate of approximately 98%. The dataset for supervised machine learning is divided into training and testing sets. They used 80% of the samples to coach the classifier and the rest to test it. After the algorithm runs, there's feedback provided to the profile, requiring it to submit identification to prove it's not a fake profile  
  
Sybil Frame classifies at multiple levels. There are both content-based and structure-based methodologies. The dataset is analysed using a content-based technique, which extracts data necessary to compute historical information about nodes and edges. Using a Markov random field and loopy belief propagation, which uses prior knowledge, the structure-based technique correlates nodes. The first step of the Sybil Frame technique uses a content-based approach, and the second stage uses a structure-based approach.

3. System Implementation

3.1 Project Modules :  
3.1.1 Admin Module:  
The administrator will enter the application with the credentials "admin" and "admin" and then carry out the following tasks.

To develop a train model using the ANN algorithm, the administrator will upload the profile dataset. By using test data from new accounts, this train model is frequently used to determine whether an account is real or false.

The admin can access all the datasets used to train the ANN model by using this module.

3.1.2 User Module:  
Anyone can use this application, input test data from the most recent account, and invoke the ANN algorithm. A new set of test data will be used, and an ANN algorithm will be employed to forecast if the test data comprises real accounts or phishing accounts.  
  
  
3.2 Methodology (Algorithms ) :  
  
3.2.1 Artifical Neural Networks Algorithm :

I'm applying artificial neural networks created through machine learning to evaluate the likelihood that a friend request is genuine or not. Every neuron (node) processes each equation through a Sigmoid function to keep the answers between the range of 0.0 and 1.0. This could easily be multiplied by 100 at the output end to provide us with the likelihood that the request is malicious. Our approach would consist of a single deep neural network with a single hidden layer.  
  
Formulation of Neural network :

Let’s start by understanding formulation of an easy hidden layer neural network. an easy neural network can be represented as shown in the figure below:  
  
  
  
The most amazing fact in an ANN is the linkage between nodes. The ANN algorithm only uses inputs as known values. Weights represent the nodes' connectivity during this method.  
  
Following is that the framework in which artificial neural networks work :  
  
1. To begin the algorithm, assign random weights to all or some of the links.

2. Determine the activation rate of Hidden nodes by using the inputs and the ensuing (input -> Hidden node) linkage.

3. Calculate the activation rate of the output nodes using the connections and hidden nodes' activation rates.

4. Determine the output node's error rate and recalibrate every link between hidden nodes and output nodes.

5. Cascade the error to Hidden nodes using the weights of and error discovered at output node.

6. Adjust the weights between the input nodes and the hidden nodes.

Error @ H1 is equal to W(H1O1)\*Error@O1 plus W(H1O2)\*Error@O2.

7. Continue using this approach until the convergence requirement is reached.

8. Score the activation using the ultimate linkage weights. rate of the output nodes.  
  
3.2.2 Sigmoid Activation Function :

Something that is curved in two directions is described by the term "sigmoid." We are only interested in one of the several sigmoid functions. Given that it is known as the logistic function, the mathematical formulation is simple:

f(x) = 1/1+e-x

The maximum value of the curve is determined by the constant L, hence the constant k affects how steep the transition is. The plot below displays examples of the logistic function for various values of L; thus, the plot below displays curves for various values of k.

The threshold value is where classification occurs, and the graph of the sigmoid activation function will have an S-shape.

4.Test cases :

|  |  |
| --- | --- |
| Test Case 1 | |
| Test Case Name | Empty login fields testing |
| Description | In the login screen if the username and password fields are empty |
| Output | Login fails showing the same page, asking to enter username and  password. |

Table 4.2.1 Test Case for Empty Login Fields

|  |  |
| --- | --- |
| Test Case 2 | |
| Test Case Name | Wrong login fields testing |
| Description | A unique username and password are set by administrator. On entering wrong username or password gives. |
| Output | Login fails showing the same page, and displays an error message username or password incorrect. |

Table 4.2.2 Test Case for Wrong Login Fields

|  |  |
| --- | --- |
| Test Case 3 | |
| Test Case Name | Model generation. |
| Description | Admin login to the page and will generate a model. |
| Output | A message will display on the screen that the “model was generated successfully”. |

Table 4.2.3 Test Case for Generating model

|  |  |
| --- | --- |
| Test Case 4 | |
| Test Case Name | Table View. |
| Description | Admin login to the page and will requests to show the trained datasets. |
| Output | A table will display on the screen that contains the information of the data that is used to train the model. |

Table 4.2.4 Test Case for Table View

|  |  |
| --- | --- |
| Test Case 5 | |
| Test Case Name | User Data. |
| Description | User will enter the data to check whether a account is phishing account or not. |
| Output | A message will display on the user panel whether an account is phishing account or genuine account. |

Table 4.2.5 Test Case for User Data

5. Conclusion & Future scope

5.1Conculsion :

## To evaluate the likelihood that a friend request is genuine or not, we employ machine learning, specifically an artificial neural network. At every neuron (node), a Sigmoid function is applied to every equation. We employ a training set of data from Facebook or other social networks. This makes it possible for the deep learning algorithm that is being used to learn patterns of bot behaviour to do so by using back propagation, minimising the overall cost function, and modifying the weight and bias of each neuron.

5.2Future scope :

## Each input neuron would be a unique, previously selected characteristic of each profile that was converted into a numerical value (for example, gender as a binary number, female 0 and male 1) and, if necessary, divided by an arbitrary number (for example, age is typically divided by 100) to reduce one characteristic having more influence on the result than the other. The neurons stand in for nodes. There would be one decision-making procedure assigned to each node.