

A COMPARATIVE STUDY ON FAKE PROFILE IDENTIFICATION USING DIFFERENT MACHINE LEARNING TECHNIQUES

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree in Computer Science and
Engineering

By

K Sai Suraj(38110260)

S Mujafar(38110335)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY

JEPIAAR NAGAR, RAJIV GANDHI SALAI,

CHENNAI – 600119, TAMILNADU



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with “A” grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **K Sai Suraj(38110260)** and **S Mujafar(38110335)** who carried out the project entitled “**A Comparative study on fake profile identification using different machine learning techniques**” under my supervision from November 2021 to March 2022.

Internal Guide

Dr. P.Asha M.E., Ph.D.

Head of the Department

Dr.S.Vigneshwari M.E., Ph.D

Dr.L.Lakshmanan M.E., Ph.D

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **K Sai Suraj (Reg No:38110260)** and **S Mujafar (Reg No: 38110335)** hereby declare that the Project Report entitled “**A Comparative study on fake profile identification using different machine learning techniques**” done by us under the guidance of **Dr. P. Asha M.E., Ph.D.** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in 2018-2022.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E.,Ph.D., Dean**, School of Computing **Dr.S.Vigneshwari M.E., Ph.D.** and **Dr.L.Lakshmanan M.E., Ph.D.** , Heads of the Department of Computer Science and Engineering for providing us necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and a deep sense of gratitude to my Project Guide **Dr. P. Asha M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

Abstract

In the present generation, On-Line social networks (OSNs) have become increasingly popular, which impacts people's social lives and impel them to become associated with various social media sites. Social Networks are the essential platforms through which many activities such as promotion, communications, agenda creation, advertisements, and news creation have started to be done. Adding new friends and keeping in contact with them and their updates has become easier. Researchers have been studying these online social networks to see the impact they make on the people. Some malicious accounts are used for purposes such as misinformation and agenda creation. Detection of malicious account is significant. The methods based on machine learning-based were used to detect fake accounts that could mislead people. The dataset is pre-processed using various python libraries and a comparison model is obtained to get a feasible algorithm suitable for the given dataset. An attempt to detect fake accounts on the social media platforms is determined by various Machine Learning algorithms. The classification performances of the algorithms Random Forest, Neural Network and Support Vector Machines are used for the detection of fake accounts.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	LIST OF FIGURES	viii
	LIST OF ABBREVIATIONS	ix
1	INTRODUCTION	1
2	LITERATURE SURVEY	3
3	METHODOLOGY	7
	3.1. EXISTING SYSTEM	7
	3.2. PROPOSED SYSTEM	8
	3.3. ADVANTAGES OF PROPOSED SYSTEM	8
	3.4. ARCHITECTURE	9
	3.5. SYSTEM REQUIREMENTS	10
	3.5.1 HARDWARE REQUIREMENTS	10
	3.5.2. SOFTWARE REQUIREMENTS	10
	3.6. SOFTWARE ENVIRONMENT	11
	3.7. MODULES	15
4	RESULTS AND DISCUSSION	26
5	CONCLUSION AND FUTURE ENHANCEMENT	32
	5.1. CONCLUSION	32
	5.2 FUTURE ENHANCEMENT	32
	REFERENCES	33
	APPENDICES	35
	A. SOURCE CODE	35
	B. SCREENSHOTS	41
	C. PUBLICATION	44

LIST OF FIGURES

Figure No.	Figure Name	Page No.
3.1.	SYSTEM ARCHITECTURE	18
3.2	DATA FLOW DIAGRAM	28
3.3	USE CASE DIAGRAM	31
3.4	CLASS DIAGRAM	32
3.5	SEQUENCE DIAGRAM	33
3.6	ACTIVITY DIAGRAM	34
4.1	LOGIN PAGE	35
4.2	LOGIN SUCCESS	36
4.3	DATA COLLECTION PAGE	36
4.4	TRAINING FINISHED	37
4.5	TESTING PAGE	37
4.6	USER CHECK PAGE	38
4.7	USER CHECK PAGE AFTER ENTERING ID	38
4.8	FINAL RESULT	39
4.9	USER CHECK PAGE WITH ANOTHER ID	39
4.10	FINAL RESULT SHOWING THAT ITS FAKE/BOT USER	40

LIST OF ABBREVIATIONS

ABBREVIATIONS	EXPANSION
ML	MACHINE LEARNING
DFD	DATA FLOW DIAGRAM
DT	DECISION TREE
OSN	ONLINE SOCIAL NETWORK
GUI	GRAPHICAL USER INTERFACE

CHAPTER 1

INTRODUCTION

It has become quite unpretentious to obtain any kind of information from any source across the world by using the Internet. The increased demand of social sites permits users to collect abundant amount of information and data about users. Huge volumes of data available on these sites also draw the attention of fake users. Twitter has rapidly become an online source for acquiring real-time information about users. Twitter is an Online Social Network (OSN) where users can share anything and everything, such as news, opinions, and even their moods. Several arguments can be held over different topics, such as politics, current affairs, and important events. When a user tweets something, it is instantly conveyed to his/her followers, allowing them to outspread the received information at a much broader level [2].

With the evolution of OSNs, the need to study and analyze users' behaviors in online social platforms has intensified . Many people who do not have much information regarding the OSN can easily be tricked by the fraudsters. There is also a demand to combat and place a control on the people who use OSNs only for advertisements and thus spam other people's accounts. Recently, the detection of spam in social networking sites attracted the attention of researchers. Spam detection is a difficult task in maintaining the security of social networks. It is essential to recognize spams in the OSN sites to save users from various kinds of malicious attacks and to preserve their security and privacy. These hazardous maneuvers adopted by spammers cause massive destruction of the community in the real world. Twitter spammers have various objectives, such as spreading invalid information, fake news, rumors, and spontaneous messages.

Spammers achieve their malicious objectives through advertisements and several other means where they support different mailing lists and subsequently dispatch spam messages randomly to broadcast their interests. These activities cause disturbance to the original users who are known as non-spammers. In addition, it also decreases the reputation of the OSN platforms. Therefore, it is essential to design a scheme to spot spammers so that corrective efforts can be taken to counter their malicious activities. Several research works have been carried out in the domain of Twitter spam detection. To encompass the existing state-of-the-art, a few surveys have also been carried out on fake user identification from Twitter. Tingmin et al. Provide a survey of new methods and techniques to identify Twitter spam detection.

The survey presents a comparative study of the current approaches. On the other hand, the authors have conducted a survey on different behaviors exhibited by spammers on Twitter social network. The study also provides a literature review that recognizes the existence of spammers on Twitter social network. Despite all the existing studies, there is still a gap in the existing literature. Therefore, to bridge the gap, we review state-of-the-art in the spammer detection and fake user identification on Twitter. Moreover, this survey presents taxonomy of the Twitter spam detection approaches and attempts to offer a detailed description of recent developments in the domain.

The aim of this project is to identify different approaches of spam detection on Twitter and to present a taxonomy by classifying these approaches into several categories. For classification, we have identified four means of reporting spammers that can be helpful in identifying fake identities of users. Spammers can be identified based on: (i) fake content, (ii) URL based spam detection, (iii) detecting spam in trending topics, and (iv) fake user identification .

CHAPTER 2

LITERATURE SURVEY

1) Statistical features-based real-time detection of drifted Twitter spam

AUTHORS: C. Chen, Y. Wang, J. Zhang, Y. Xiang, W. Zhou, and G. Min

Twitter spam has become a critical problem nowadays. Recent works focus on applying machine learning techniques for Twitter spam detection, which make use of the statistical features of tweets. In this labeled tweets data set, however, we observe that the statistical properties of spam tweets vary over time, and thus, the performance of existing machine learning-based classifiers decreases. This issue is referred to as “Twitter Spam Drift”. In order to tackle this problem, we first carry out a deep analysis on the statistical features of one million spam tweets and one million non-spam tweets, and then propose a novel Lfun scheme. The proposed scheme can discover “changed” spam tweets from unlabeled tweets and incorporate them into classifier's training process. A number of experiments are performed to evaluate the proposed scheme. The results show that our proposed Lfun scheme can significantly improve the spam detection accuracy in real-world scenarios.

2) Automatically identifying fake news in popular Twitter threads

AUTHORS: C. Buntain and J. Golbeck

Information quality in social media is an increasingly important issue, but web-scale data hinders experts' ability to assess and correct much of the inaccurate content, or "fake news," present in these platforms. This paper develops a

method for automating fake news detection on Twitter by learning to predict accuracy assessments in two credibility-focused Twitter datasets: CREDBANK, a crowdsourced dataset of accuracy assessments for events in Twitter, and PHEME, a dataset of potential rumors in Twitter and journalistic assessments of their accuracies.

We apply this method to Twitter content sourced from BuzzFeed's fake news dataset and show models trained against crowdsourced workers outperform models based on journalists' assessment and models trained on a pooled dataset of both crowdsourced workers and journalists. All three datasets, aligned into a uniform format, are also publicly available. A feature analysis then identifies features that are most predictive for crowdsourced and journalistic accuracy assessments, results of which are consistent with prior work. We close with a discussion contrasting accuracy and credibility and why models of non-experts outperform models of journalists for fake news detection in Twitter.

3) A performance evaluation of machine learning-based streaming spam tweets detection

AUTHORS: C. Chen, J. Zhang, Y. Xie, Y. Xiang, W. Zhou, M. M. Hassan, A. AlElaiwi, and M. Alrubaian

The popularity of Twitter attracts more and more spammers. Spammers send unwanted tweets to Twitter users to promote websites or services, which are harmful to normal users. In order to stop spammers, researchers have proposed a number of mechanisms. The focus of recent works is on the application of machine learning techniques into Twitter spam detection. However, tweets are retrieved in a streaming way, and Twitter provides the Streaming API for

developers and researchers to access public tweets in real time. There lacks a performance evaluation of existing machine learning-based streaming spam detection methods. In this paper, we bridged the gap by carrying out a performance evaluation, which was from three different aspects of data, feature, and model.

A big ground-truth of over 600 million public tweets was created by using a commercial URL-based security tool. For real-time spam detection, we further extracted 12 lightweight features for tweet representation. Spam detection was then transformed to a binary classification problem in the feature space and can be solved by conventional machine learning algorithms. We evaluated the impact of different factors to the spam detection performance, which included spam to nonspam ratio, feature discretization, training data size, data sampling, time-related data, and machine learning algorithms. The results show the streaming spam tweet detection is still a big challenge and a robust detection technique should take into account the three aspects of data, feature, and model.

4) A model-based approach for identifying spammers in social networks

AUTHORS: F. Fathaliani and M. Bouguessa

In this paper, we view the task of identifying spammers in social networks from a mixture modeling perspective, based on which we devise a principled unsupervised approach to detect spammers. In our approach, we first represent each user of the social network with a feature vector that reflects its behaviour and interactions with other participants. Next, based on the estimated users feature vectors, we propose a statistical framework that uses the Dirichlet distribution in order to identify spammers. The proposed approach is able to

automatically discriminate between spammers and legitimate users, while existing unsupervised approaches require human intervention in order to set informal threshold parameters to detect spammers. Furthermore, our approach is general in the sense that it can be applied to different online social sites. To demonstrate the suitability of the proposed method, we conducted experiments on real data extracted from Instagram and Twitter.

5) Spam detection of Twitter traffic: A framework based on random forests and non-uniform feature sampling

AUTHORS: C. Meda, E. Ragusa, C. Gianoglio, R. Zunino, A. Ottaviano, E. Scillia, and R. Surlinelli

Law Enforcement Agencies cover a crucial role in the analysis of open data and need effective techniques to filter troublesome information. In a real scenario, Law Enforcement Agencies analyze Social Networks, i.e. Twitter, monitoring events and profiling accounts. Unfortunately, between the huge amount of internet users, there are people that use microblogs for harassing other people or spreading malicious contents. Users' classification and spammers' identification is a useful technique for relieve Twitter traffic from uninformative content.

This work proposes a framework that exploits a non-uniform feature sampling inside a gray box Machine Learning System, using a variant of the Random Forests Algorithm to identify spammers inside Twitter traffic. Experiments are made on a popular Twitter dataset and on a new dataset of Twitter users. The new provided Twitter dataset is made up of users labeled as spammers or legitimate users, described by 54 features. Experimental results demonstrate the effectiveness of enriched feature sampling method

CHAPTER 3

METHODOLOGY

3.1 EXISTING SYSTEM

- Tingminet *al.* provide a survey of new methods and techniques to identify Twitter spam detection. The above survey presents a comparative study of the current approaches.
- On the other hand, S. J. Somanet. *al.* conducted a survey on different behaviors exhibited by spammers on Twitter social network. The study also provides a literature review that recognizes the existence of spammers on Twitter social network.
- Despite all the existing studies, there is still a gap in the existing literature. Therefore, to bridge the gap, we review state-of-the-art in the spammer detection and fake user identification on Twitter

DISADVANTAGES EXISTING SYSTEM

- Because of Privacy Issues the Facebook dataset is very limited and a lot of details are not made public.
- Having less accuracy
- More complex

3.2 PROPOSED SYSTEM

The proposed framework, the sequence of processes that need to be followed for continues detection of fake profiles with active learning from the feedback of the result given by the classification algorithm. This framework can easily be implemented by social networking companies. 1. The detection process starts with the selection of the profile that needs to be tested. 2. After the selection of the profile, the suitable attributes (i.e. features) are selected on which the classification algorithm is implemented. 3. The attributes extracted is passed to the trained classifier. The classifier gets trained regularly as new training data is feed into the classifier. 4. The classifier determines whether the profile is fake or genuine. 5. The classifier may not be 100% accurate in classifying the profile so; the feedback of the result is given back to the classifier. 6. This process repeats and as the time proceeds, the no. of training data increases and the classifier becomes more and more accurate in predicting the fake profiles.

3.3 ADVANTAGES OF PROPOSED SYSTEM

- The social networking sites are making our social lives better but nevertheless there are a lot of issues with using these social networking sites.
- The issues are privacy, online bullying, potential for misuse, trolling, etc. These are done mostly by using fake profiles.
- In this project, we came up with a framework through which we can detect a fake profile using machine learning algorithms so that the social life of people become secured.

3.4 ARCHITECTURE DIAGRAM



Fig 3.1 System Architecture

DESCRIPTION

Proposed system is equipped with various Machine Learning tasks and the architecture followed is as shown below. The proposed system collects the dataset which are preprocessed by providing a framework of algorithms using which we can detect fake profiles in Facebook by comparing the accuracy of three machine learning algorithms and the algorithm with very high efficiency is found for the given dataset. The different ways in which an algorithm can model a problem is based on its interaction with the experience or environment for the model preparation process that helps in choosing the most appropriate algorithm for the given input data in order to get the best result.

3.5 SYSTEM REQUIREMENTS:

3.5.1 HARDWARE REQUIREMENTS:

- System : Pentium Dual Core.
- Hard Disk : 120 GB.
- Monitor : 15'' LED
- Input Devices : Keyboard, Mouse
- Ram : 4 GB.

3.5.2 SOFTWARE REQUIREMENTS:

- Operating system: Windows 7/10.
- Coding Language :Python
- Tool : Pi-champ
- Database : MYSQL

3.6 SOFTWARE ENVIRONMENT

Python:

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

- **Python is Interactive** – You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** – Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python Features

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.

- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy-to-maintain.
- **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** – You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** – Python provides interfaces to all major commercial databases.
- **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** – Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below –

- It supports functional and structured programming methods as well as OOP.

- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

Getting Python

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python <https://www.python.org>.

Windows Installation

Here are the steps to install Python on Windows machine.

- Open a Web browser and go to <https://www.python.org/downloads/>.
- Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.
- To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
- Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages.

Flask Framework:

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Http protocol is the foundation of data communication in world wide web. Different methods of data retrieval from specified URL are defined in this protocol.

The following table summarizes different http methods –

Sr.No	Methods & Description
1	GET Sends data in unencrypted form to the server. Most common method.
2	HEAD Same as GET, but without response body

3	POST Used to send HTML form data to server. Data received by POST method is not cached by server.
4	PUT Replaces all current representations of the target resource with the uploaded content.
5	DELETE Removes all current representations of the target resource given by a URL

3.7 MODULES:

- ❖ Admin Module
- ❖ Data Collection
- ❖ Train and Test
- ❖ Machine Learning Technique
- ❖ Detection of Fake Profiles

MODULE DESCRIPTIONS:

Admin Module:

In the first module, we develop the Online Social Networking (OSN) system module. We build up the system with the feature of Online Social Networking System, Twitter. Where, this module is used for admin login with their authentication.

Data Collection:

We will be using a Python Library called *Tweepy* to connect to the Twitter API and collect the data. We download tweets containing certain key words, to incorporate the words or hash tags that contain relevant keyword related to fake users.

Some of the most important fields are:

- *text*, which contains the text included in the tweet.
- *created_at*, which is a timestamp of when the tweet was created.
- *user*, which contains information about the user that created the tweet, like the username and user id.

Train and Test:

- ❖ We present the proposed framework for metadata features are extracted from available additional information regarding the tweets of a user, whereas content-based features aim to observe the message posting behavior of a user and the quality of the text that the user uses in posts.

Machine Learning Technique:

- ❖ The number of features, which are associated with tweet content, and the characteristics of users are recognized for the detection of spammers. These features are considered as the characteristics of machine learning process for categorizing users, i.e., to know whether they are spammers or not.
- ❖ In order to recognize the approach for detecting spammers on Twitter, the labelled collection in pre-classification of fake user and legitimate user

has been done. Next, those steps are taken which are needed for the construction of labeled collection and acquired various desired properties.

- ❖ In other words, steps which are essential to be examined to develop the collection of users that can be labelled as fake user or legitimate user. At the end, user attributes are identified based on their behavior, e.g., who they interact with and what is the frequency of their interaction.
- ❖ In order to confirm this instinct, features of users of the labelled collection has been checked. Two attribute sets are considered, i.e., content attributes and user behavior attributes, to differentiate one user from the other.

Detection of Fake User:

The proposed system collects the dataset which are preprocessed by providing a framework of algorithms using which we can detect fake profiles in Facebook by comparing the accuracy of three machine learning algorithms and the algorithm with very high efficiency is found for the given dataset. The different ways in which an algorithm can model a problem is based on its interaction with the experience or environment for the model preparation process that helps in choosing the most appropriate algorithm for the given input data in order to get the best result.

DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

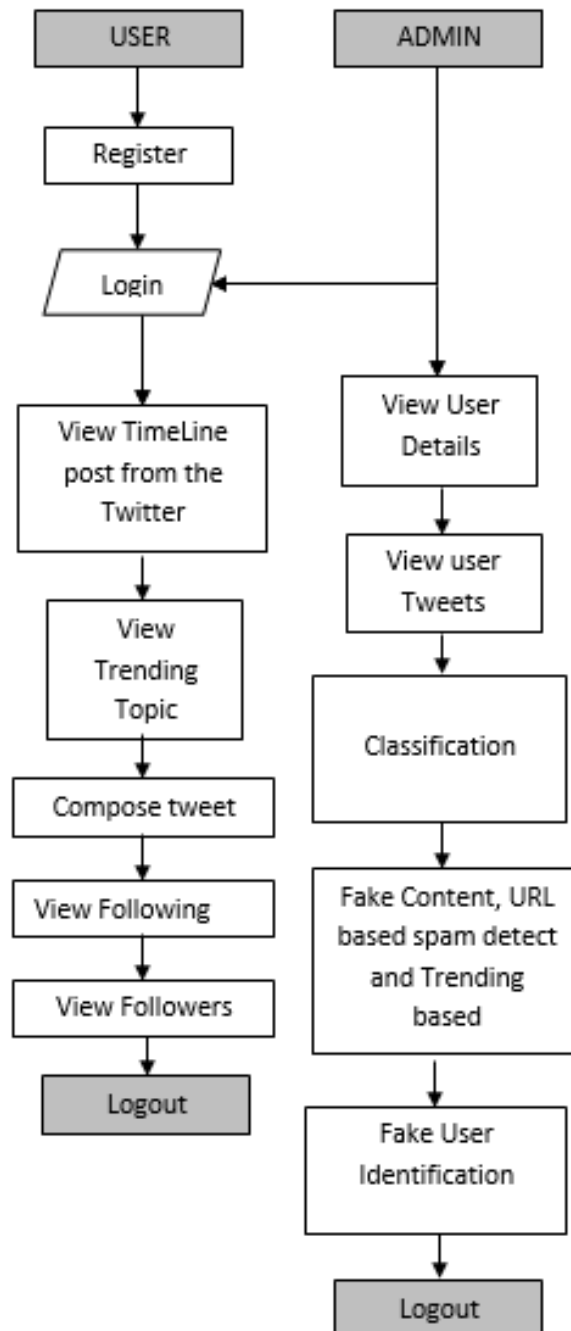


Fig 3.2 Data Flow Diagram

UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.

4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

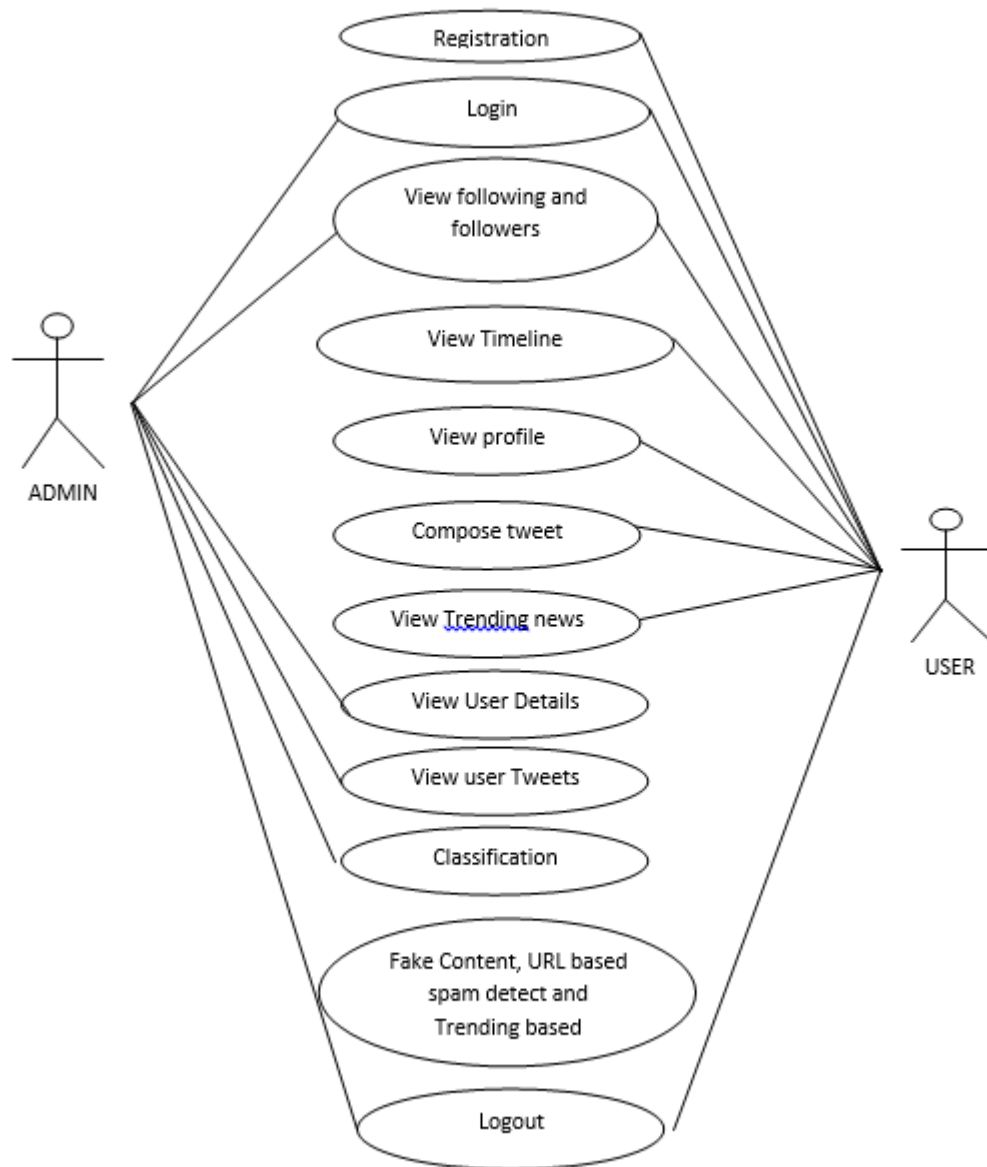


Fig 3.3 Use Case Diagram

CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or

methods), and the relationships among the classes. It explains which class contains information.

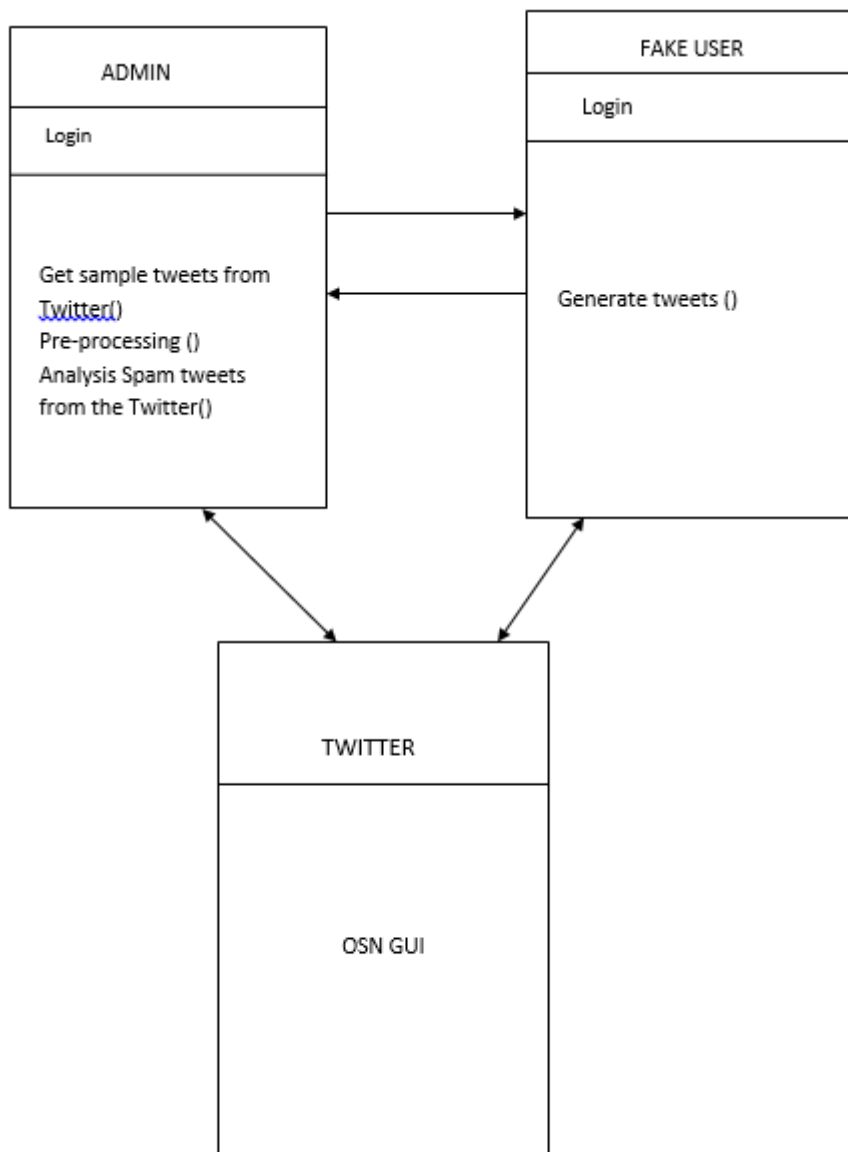


Fig 3.4 Class Diagram

SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

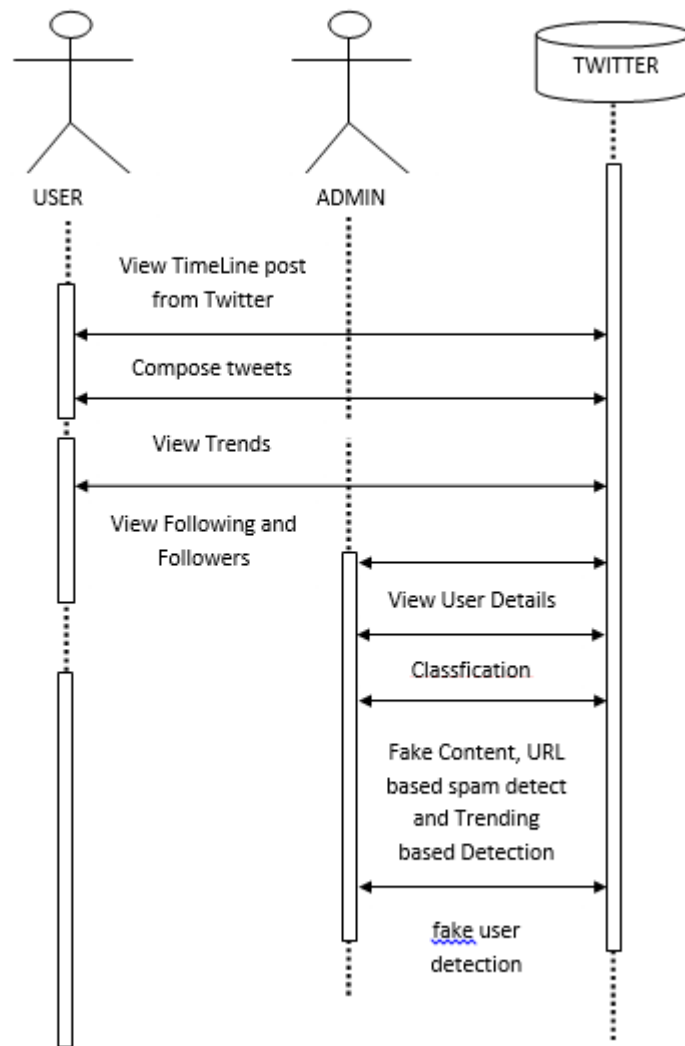


Fig 3.5 Sequence Diagram

ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

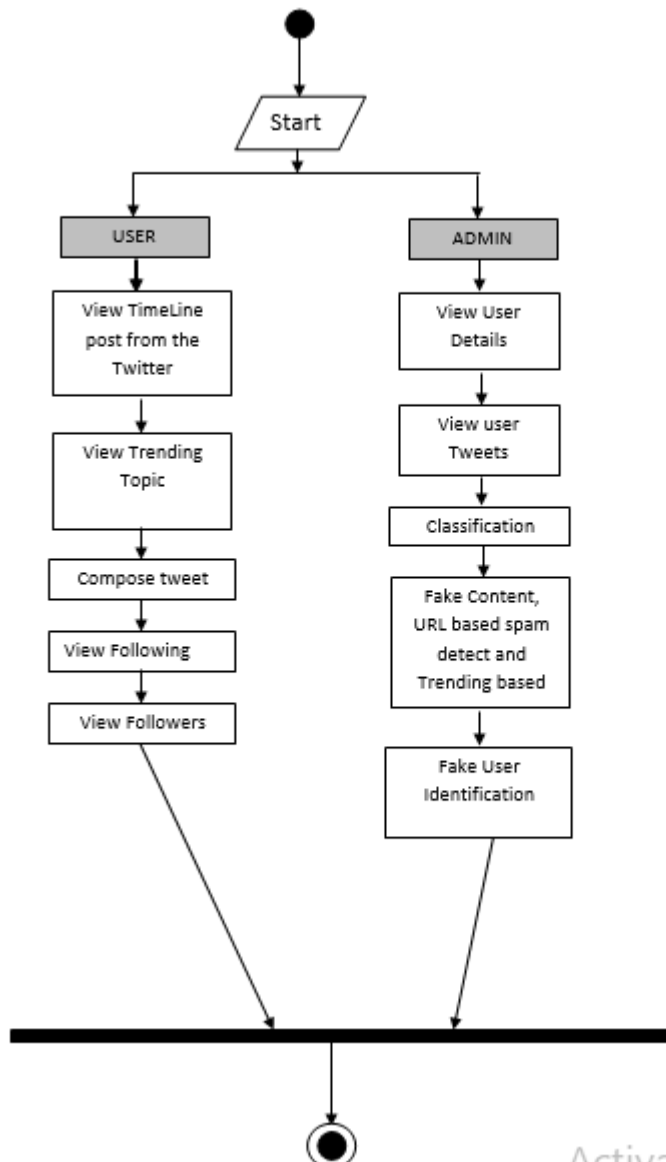


Fig 3.6 Activity Diagram

CHAPTER 4

RESULTS AND DISCUSSION

So by the final results we can detect whether the user is legitimate or fake/bot user. In the below figures we show two cases where the user is legitimate and other is fake user and it predicted or detected correctly. In Fig.1 Where I have given my own id it predicted as legitimate user and in the next Fig.3 I have given bot user id and it predicted as fake/bot user.

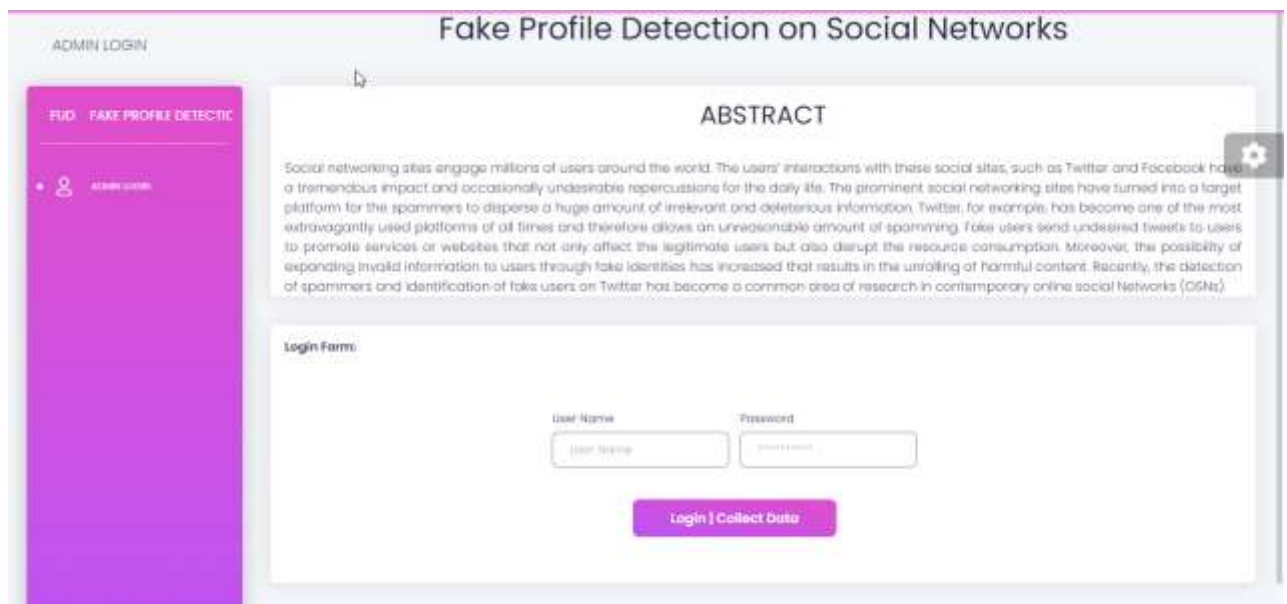


Fig 4.1 Login Page

So in the above figure 4.1 it's a login page where we enter our username and password and then we login into the website. Here we are admin so we use admin login.

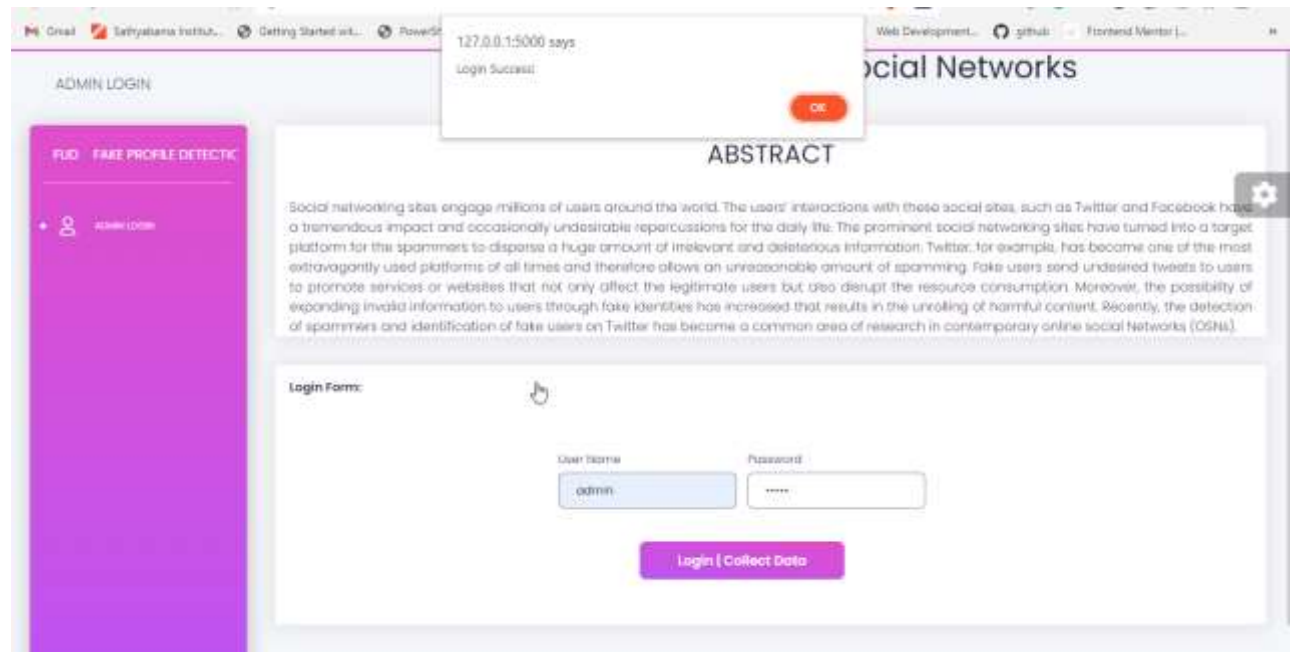


Fig 4.2 Login Success

So in the above image using admin login credentials we logged in and it showed login success.

	created_at	username	tweet_id	text	favorite_count	retweet_count
0	Thu Apr 07 21:17:05 +0000 2022	7artistai	151217565835093760	Advertisements for #btc \n#CashApp. a #mobile #payment #app from #Block introduces a #service that allows users to... https://t.co/nugHawG13	2	0
1	Sat Apr 09 10:08:24 +0000 2022	Andriovikasnews	1512734932674976822	Covid boosters provisionally approved for Australian adolescents https://t.co/yG0Ubnvax #NationalNews... https://t.co/7nBzPF8Ful	0	0
2	Sat Apr 09 10:09:03 +0000 2022	ArtyAnti	1512734327418855434	RT @perkytee_ The Polar Collection computer-generated art pieces using algorithms \n\nOpensea: https://t.co/hK0tQw4lw \n\nMintable: https://t.co/7nBzPF8Ful	0	1
3	Sat Apr 09 10:05:48 +0000 2022	ArtyAnti	151273350250416227	RT @perkytee_ Art0012 from The Polar Collection \n\nI Computer-Generated Art \n\nPrice: .001eth (gasless) \n\n https://t.co/0h6iDQDQs \n\nnftNF...	0	1
4	Sat Apr 09 10:05:06 +0000 2022	ArtyAnti	1512733335054821729	RT @perkytee_ Art002 from The Polar Collection @Mintable \n\nI Computer-Generated Abstract Art \n\nPrice: .001eth	0	1

Fig 4.3 Data Collection Page

This is the Data collection page here we collect all the data and then we click on train | test button so the model gets trained on the data

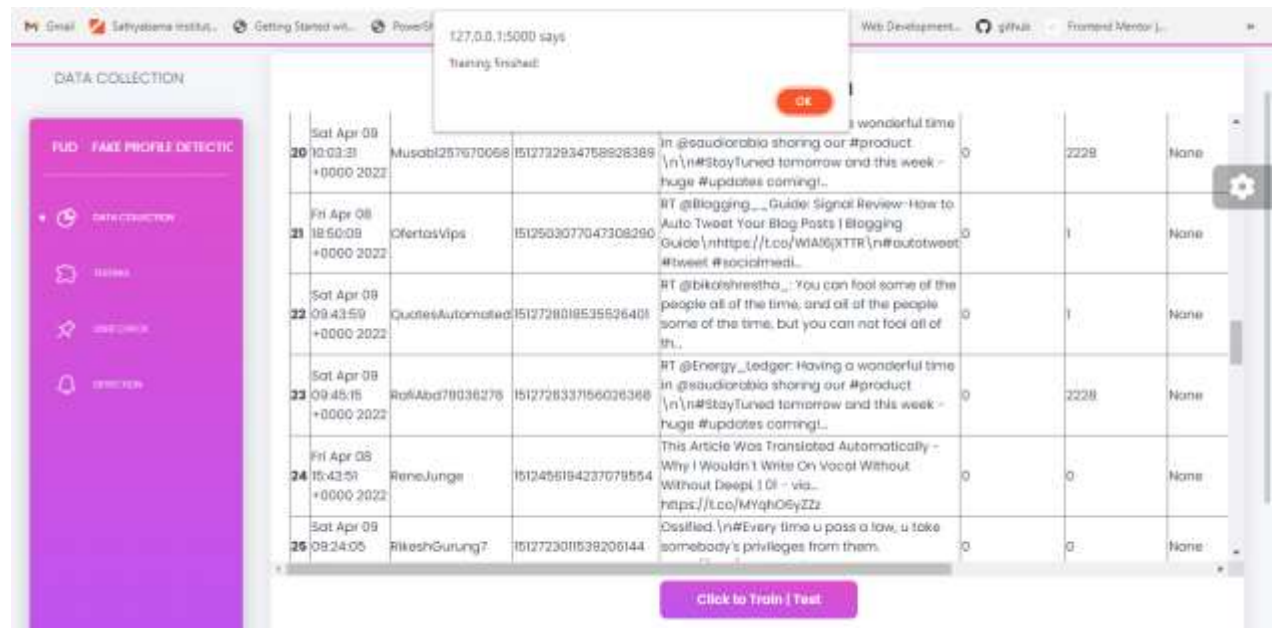


Fig 4.4 Training Finished

So here it says that training is finished so the model got its training on data finished so we can move on to next stage

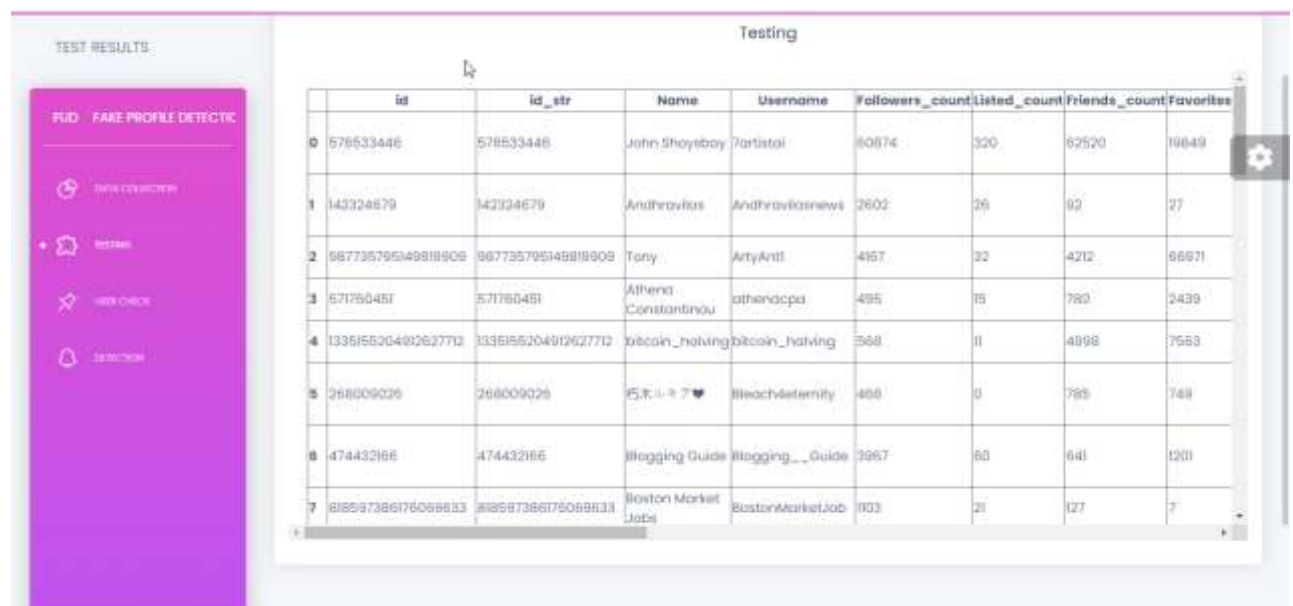


Fig 4.5 Testing Page

So this is the testing page where we take the data we collected and test on it using our model so after doing this testing we move on to the next stage which is user check

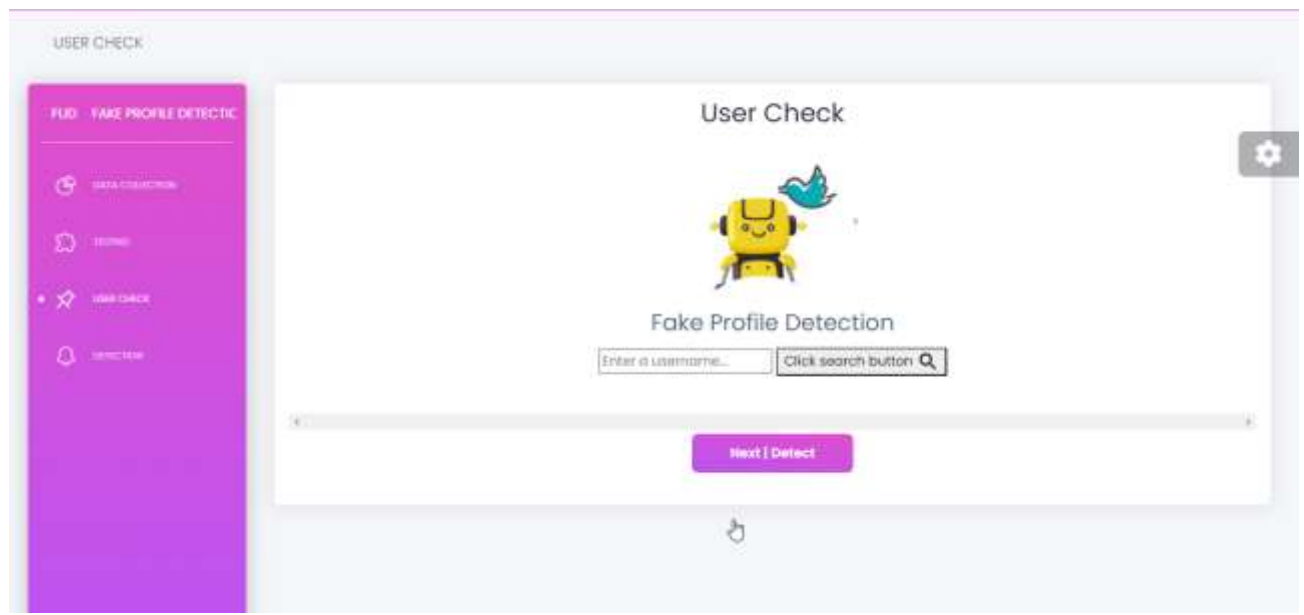


Fig 4.6 User check page

So in this page we enter username of the user that we want to check whether it's a legitimate user or fake user.



Fig 4.7 User check page after entering id

Here we entered 1510sai as username which is my twitter username so when we click the next| detect button we get our final results

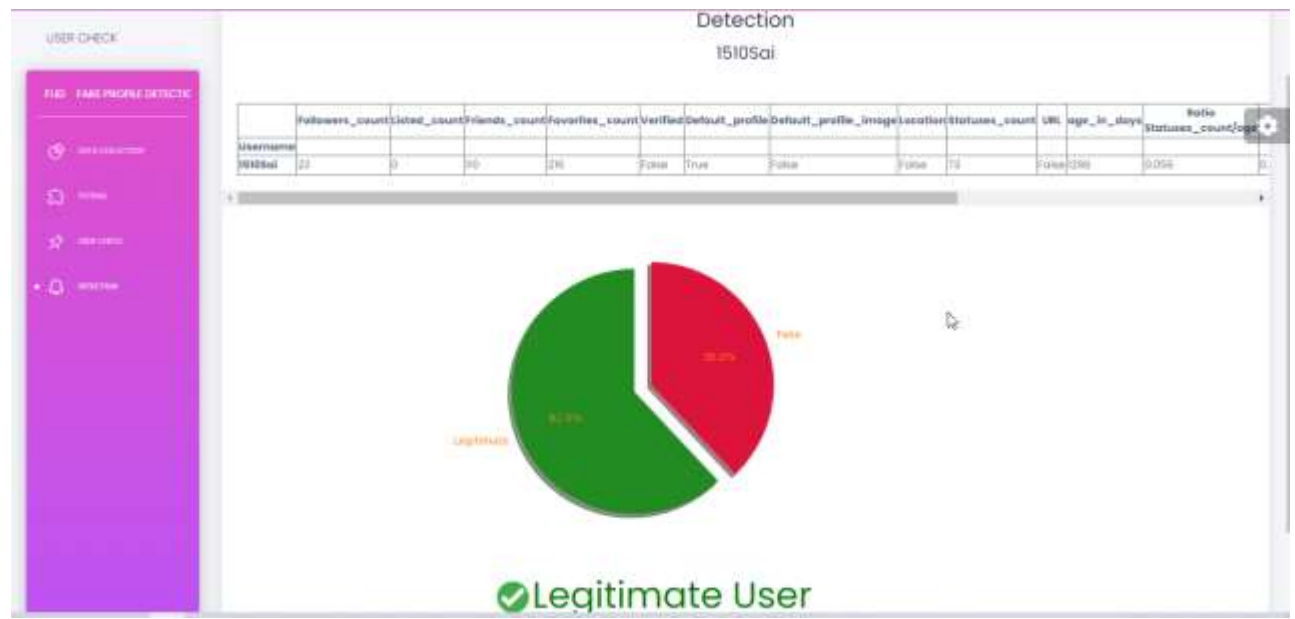


Fig 4.8 Final Result

So here's we got the output that 1510sai is a legitimate user so this is the final stage of our project. Like this we can check for various username and find out whether they are fake/ Legitimate user.

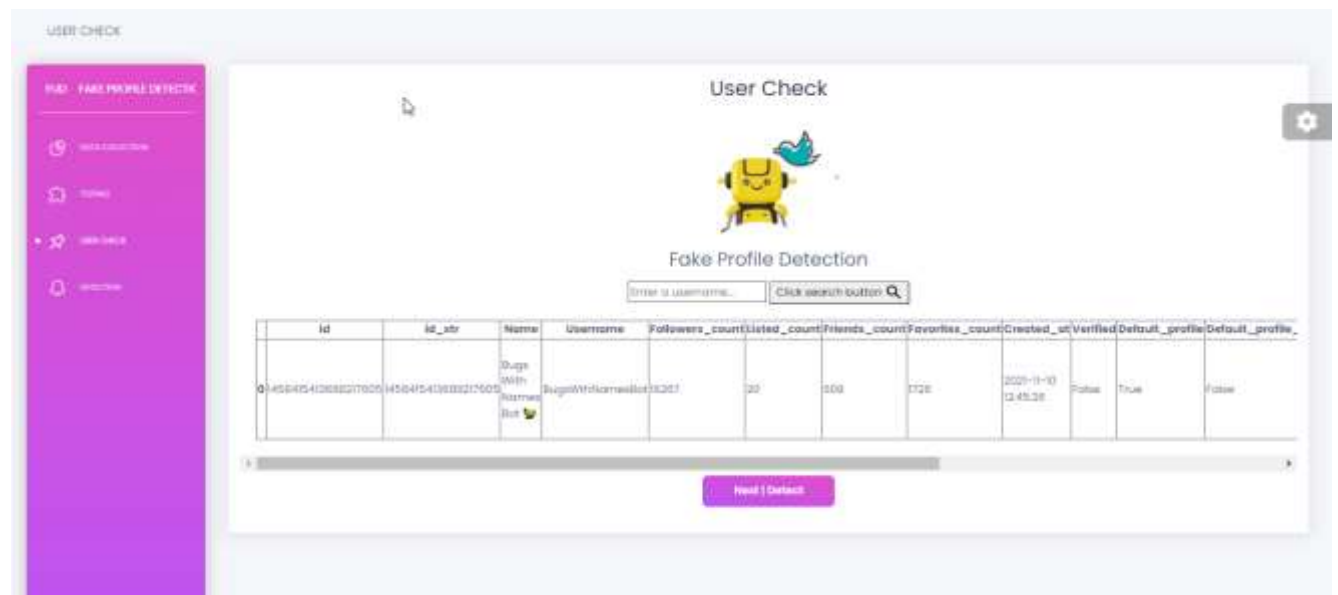


Fig 4.9 User check page with another id

So here unlike previous we enter another id called BugswithNamesBot so now we click next|detect button

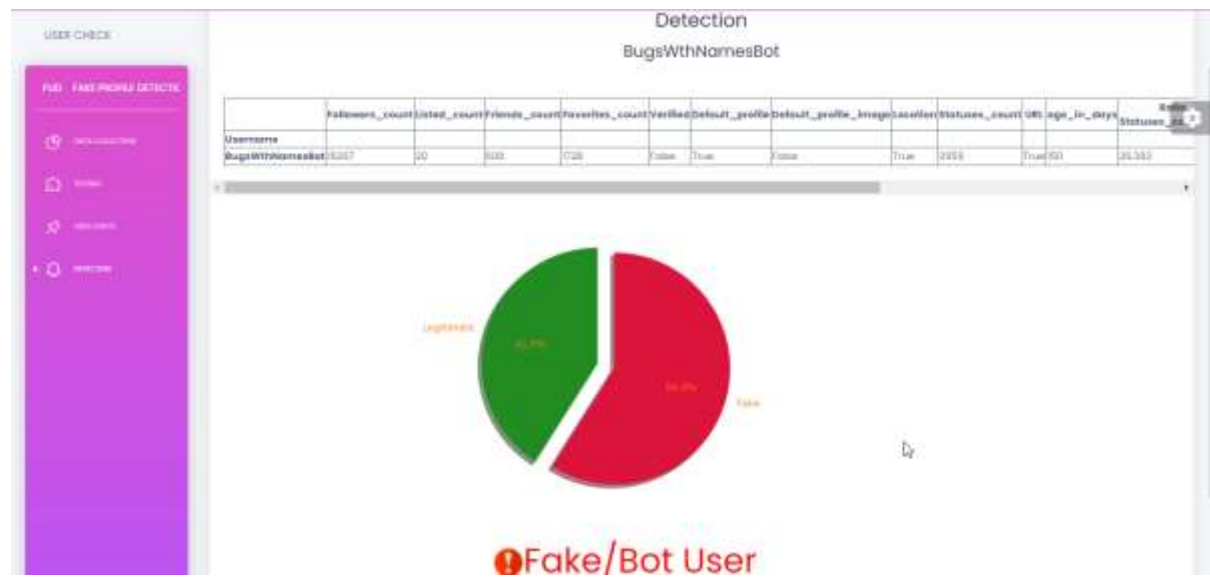


Fig 4.10 Final result showing that its Fake/bot user

So here we got the result for the previous entered id so we get the correct output which detected it as the fake/bot user which is correct since it's a bot.

CHAPTER 5

CONCLUSION

5.1. CONCLUSION

In this paper, we performed a review of techniques used for detecting spammers on Twitter. In addition, we also presented a taxonomy of Twitter spam detection approaches and categorized them as fake content detection, URL based spam detection, spam detection in trending topics, and fake user detection techniques. We also compared the presented techniques based on several features, such as user features, content features, graph features, structure features, and time features. Moreover, the techniques were also compared in terms of their specified goals and datasets used. It is anticipated that the presented review will help researchers find the information on state-of-the-art Twitter spam detection techniques in a consolidated form. Despite the development of efficient and effective approaches for the spam detection and fake user identification on Twitter [34], there are still certain open areas that require considerable attention by the researchers. The issues are briefly highlighted as under : False news identification on social media networks is an issue that needs to be explored because of the serious repercussions of such news at individual as well as collective level [25]. Another associated topic that is worth investigating is the identification of rumor sources on social media. Although a few studies based on statistical methods have already been conducted to detect the sources of rumors, more sophisticated approaches, e.g., social network based approaches, can be applied because of their proven effectiveness.

5.2. FUTURE ENHANCEMENT

We can improve this project in many various ways so one thing we can do is we can upload this into some external server and host it online so as a website people can use it for their own purposes. We can also make it as a chrome

extension and when people are in twitter on some users page it detects whether it's a real user or a fake/bot user so its very useful for them we can also try to make it as an app for mobile users. So there are a lot of exciting ways to this project and enhance it in future in any way we want.

REFERENCES

- [1] C. Chen, S. Wen, J. Zhang, Y. Xiang, J. Oliver, A. Alelaiwi, and M. M. Hassan, "Investigating the deceptive information in Twitter spam," *Future Gener. Comput. Syst.*, vol. 72, pp. 319–326, Jul. 2017.
- [2] I. David, O. S. Siordia, and D. Moctezuma, "Features combination for the detection of malicious Twitter accounts," in *Proc. IEEE Int. Autumn Meeting Power, Electron. Comput. (ROPEC)*, Nov. 2016, pp. 1–6.
- [3] M. Babcock, R. A. V. Cox, and S. Kumar, "Diffusion of pro- and anti-false information tweets: The black panther movie case," *Comput. Math. Org. Theory*, vol. 25, no. 1, pp. 72–84, Mar. 2019.
- [4] S. Keretna, A. Hossny, and D. Creighton, "Recognising user identity in Twitter social networks via text mining," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2013, pp. 3079–3082.
- [5] C. Meda, F. Bisio, P. Gastaldo, and R. Zunino, "A machine learning approach for Twitter spammers detection," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2014, pp. 1–6.
- [6] W. Chen, C. K. Yeo, C. T. Lau, and B. S. Lee, "Real-time Twitter content polluter detection based on direct features," in *Proc. 2nd Int. Conf. Inf. Sci. Secur. (ICISS)*, Dec. 2015, pp. 1–4.
- [7] H. Shen and X. Liu, "Detecting spammers on Twitter based on content and social interaction," in *Proc. Int. Conf. Netw. Inf. Syst. Comput.*, pp. 413–417, Jan. 2015.
- [8] G. Jain, M. Sharma, and B. Agarwal, "Spam detection in social media using convolutional and long short term memory neural network," *Ann. Math. Artif. Intell.*, vol. 85, no. 1, pp. 21–44, Jan. 2019.

- [9] M. Washha, A. Qaroush, M. Mezghani, and F. Sedes, “A topic-based hidden Markov model for real-time spam tweets filtering,” *Procedia Comput. Sci.*, vol. 112, pp. 833–843, Jan. 2017.
- [10] F. Pierri and S. Ceri, “False news on social media: A data-driven survey,” 2019, arXiv:1902.07539. [Online]. Available: <https://arxiv.org/abs/1902.07539>
- [11] S. Sadiq, Y. Yan, A. Taylor, M.-L. Shyu, S.-C. Chen, and D. Feaster, “AAFA: Associative affinity factor analysis for bot detection and stance classification in Twitter,” in *Proc. IEEE Int. Conf. Inf. Reuse Integr. (IRI)*, Aug. 2017, pp. 356–365.
- [12] M. U. S. Khan, M. Ali, A. Abbas, S. U. Khan, and A. Y. Zomaya, “Segregating spammers and unsolicited bloggers from genuine experts on Twitter,” *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 551–560, Jul./Aug. 2018.

APPENDICES

A) SOURCE CODE

```
from flask import Flask, render_template, request, jsonify, session
from retrieve_tweet import data_collection,download_user,download_user_bulk
import joblib
import pandas as pd
import os
import glob
import matplotlib.pyplot as plt
import matplotlib
matplotlib.rcParams['text.color'] = 'tab:orange'
from analisis_data_profil import preprocess, preprocess_bulk
app = Flask(__name__)
app.secret_key = 'twitter'
app.config['SEND_FILE_MAX_AGE_DEFAULT'] = 1

#SESSION_TYPE = 'filesystem'

model = joblib.load('finalized_model_without.sav')
path = os.getcwd()+"\\collectindividual'
```

```

def prediction_bulk(df):
    skrip = preprocess_bulk(df)
    pred_proba=model.predict_proba(skrip)
    y_pred=model.predict(skrip)
    percentage=pred_proba[:,1]
    joins=' '.join(map(str, percentage))
    perc=float(joins)*100
    percent=(str(perc)+"%")
    #print(y_pred)
    df = df.drop(df.columns[[17,18,19,20,21,22]], axis = 1)
    return df,percent,y_pred

```

```

def prediction(df):
    skrip,tab = preprocess(df)
    pred_proba=model.predict_proba(skrip)
    y_pred=model.predict(skrip)
    percentage=pred_proba[:,1]
    joins=' '.join(map(str, percentage))
    perc=float(joins)*100
    percent=perc
    print(y_pred)
    return percent,tab,y_pred

```

```

@app.route("/")

```

```

def login():
    return render_template('login.html')

@app.route("/collect")
def collect():
    df = data_collection()
    df=df.sort_values(by=['username']).reset_index(drop=True)
    df.to_csv('collect.csv')
    return render_template('collect.html',df=df.to_html())

@app.route("/test")
def test():
    df = pd.read_csv("collect.csv")
    res = pd.DataFrame()
    for ind in df.index:
        uname = df['username'][ind]
        print(uname)
        download_user_bulk(df['username'][ind])
    all_files = glob.glob(path + "/*.csv")
    li = []
    for filename in all_files:
        df = pd.read_csv(filename, index_col=None, header=0)
        li.append(df)
        fr,per,stat=prediction_bulk(df)

```

```

pr = per

st = stat

fr['Percentage'] = pr

if st == False :

    Result = "Legitimate User"

    fr['State'] = Result

else:

    Result = "Fake/Bot User"

    fr['State'] = Result

res = res.append(fr)

res=res.reset_index(drop=True)

return render_template('test.html',result=res.to_html())

```

```

@app.route("/check", methods=['POST','GET'])

```

```

def check():

```

```

    dl_user = request.form.get('chat_in')

```

```

    if dl_user == None:

```

```

        return render_template('check.html')

```

```

    else:

```

```

        session['username'] = dl_user

```

```

        download = download_user(dl_user)

```

```

        df=pd.read_csv('coba.csv')

```

```

        return render_template('check.html',df=df.to_html())

```

```

@app.route("/detect")
def detect():
    all_files = glob.glob(path + "/*.csv")
    for filename in all_files:
        if filename.endswith('.csv'):
            os.unlink(filename)
            #print(filename)
    my_var = session.get('username', None)
    print(my_var)
    df=pd.read_csv('coba.csv')
    prediksi,tab,y_pred=prediction(df)
    tab['Username']=my_var
    tab=tab.set_index('Username')
    tab=tab
    labels = 'Legitimate', 'Fake'
    size1 = 100-float(prediksi)
    size2 = prediksi
    sizes = [size1, size2]
    colors = ['forestgreen', 'crimson']
    explode = (0, 0.15)
    fig1, ax1 = plt.subplots()
    ax1.pie(sizes, explode=explode, labels=labels, autopct='%1.1f%%', colors
=colors,
            shadow=True, startangle=90)
    ax1.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.

```

```
plt.savefig('static/images/graph/'+my_var+'.png',transparent=True)

img = 'static/images/graph/'+my_var+'.png'

return
render_template('detect.html',prediction=prediksi,tab=tab.to_html(),y_pred=y_pred,my_var=my_var,img=img)

if __name__ == "__main__":
    app.run(debug=True)
```


B.SCREENSHOTS

ADMIN LOGIN

FUD - FAKE PROFILE DETECTIC

ABSTRACT

Social networking sites engage millions of users around the world. The users' interactions with these social sites, such as Twitter and Facebook, have a tremendous impact and occasionally undesirable repercussions for the daily life. The prominent social networking sites have turned into a target platform for the spammers to dispose a huge amount of irrelevant and deleterious information. Twitter, for example, has become one of the most extravagantly used platforms of all times and therefore allows an unreasonable amount of spamming. Fake users send undesired tweets to users to promote services or websites that not only affect the legitimate users but also disrupt the resource consumption. Moreover, the possibility of expanding invalid information to users through fake identities has increased that results in the unrolling of harmful content. Recently, the detection of spammers and identification of fake users on Twitter has become a common area of research in contemporary online social networks (OSNs).

Login Form:

User Name: admin Password: *****

Login | Collect Data

FIG B1. Login Page

DATA COLLECTION

FUD - FAKE PROFILE DETECTIC

Data Collection

	created_at	username	tweet_id	text	favorited
0	Wed Mar 02 03:06:02 +0000 2022	1000daysOfWriter	149885717471505536	RT @Blogging__Guide: Signal Review-How to Auto Tweet Your Blog Posts Blogging Guide https://t.co/WA55XTTR n#autotweet #tweet #socialmedi...	0
1	Wed Mar 02 17:23:17 +0000 2022	3183capital	1499072868033857896	Closed Buy on #MT4 #FOREX #AUTOMATED EA: JPN225 26481.0 for +103.0 pips, total for today +259.0 pips	0
2	Wed Mar 02 17:23:17 +0000 2022	3183capital	1499072868575023109	Closed Sell on #MT4 #FOREX #AUTOMATED EA: JPN225 26354.0 for -257.0 pips, total for today +11.0 pips	0
3	Wed Mar 02 17:16:00 +0000 2022	ActualPhoneServ	1499071032568659970	AA1501\1A320\1N679AW\1n81d3c\1nElev Angle: 19.2\1n#AAL501 #AAL #A320 #N678AW #a81d3c\1nAutomated Photo Crop #Robotflow #AI Co... https://t.co/q7menCQal	0
4	Wed Mar 02 14:52:00 +0000 2022	Animuncula	1499034719486089744	RT @badDinDAI: The #generated #spelloftheday is Speak with Objects! The descriptions on these were so fun that I'll probably post more descr...	0
5	Wed Mar 02 15:03:14 +0000 2022	AsimweAile	149905272338542083	@TheGtsbaBoats #every one should be mindful of the importance of impacting the future of a girl child through ski... https://t.co/RuWAXT53AX	0
6	Wed Mar 02 03:06:06 +0000 2022	Blogging__Guide	149885715055425264	Signal Review-How to Auto Tweet Your Blog Posts Blogging Guide https://t.co/WA55XTTR n#autotweet #tweet #socialmedi...	0

Click to Train | Test

FIG B2. Data Collection Stage

TEST RESULTS

FUD / FAKE PROFILE DETECTIC

DATA COLLECTION

TESTING

USER CHECK

DETECTION

Testing

	id	id_str	Name	Username	Followers_count	Listed_count	Friends_count	Fa
0	148008104655249152	148008104655249152	100 Days of Writers (Bot)	100DaysOfWriter	2033	7	1	17
1	1451496699440345089	1451496699440345089	3183 capital	3183capital	81	0	221	0
2	1485613318905513473	1485613318905513473	ActualPlaneServer	ActualPlaneServ	22	1	38	27
3	144212411150553853	144212411150553853	Bone Dice	Animuncula	3584	42	3	67

FIG B3. Testing Page

USER CHECK

FUD / FAKE PROFILE DETECTIC


DATA COLLECTION

TESTING

USER CHECK

DETECTION

User Check



Fake Profile Detection

Enter a username...

Click search button 🔍

Next | Detect

FIG B4. We give user id as input

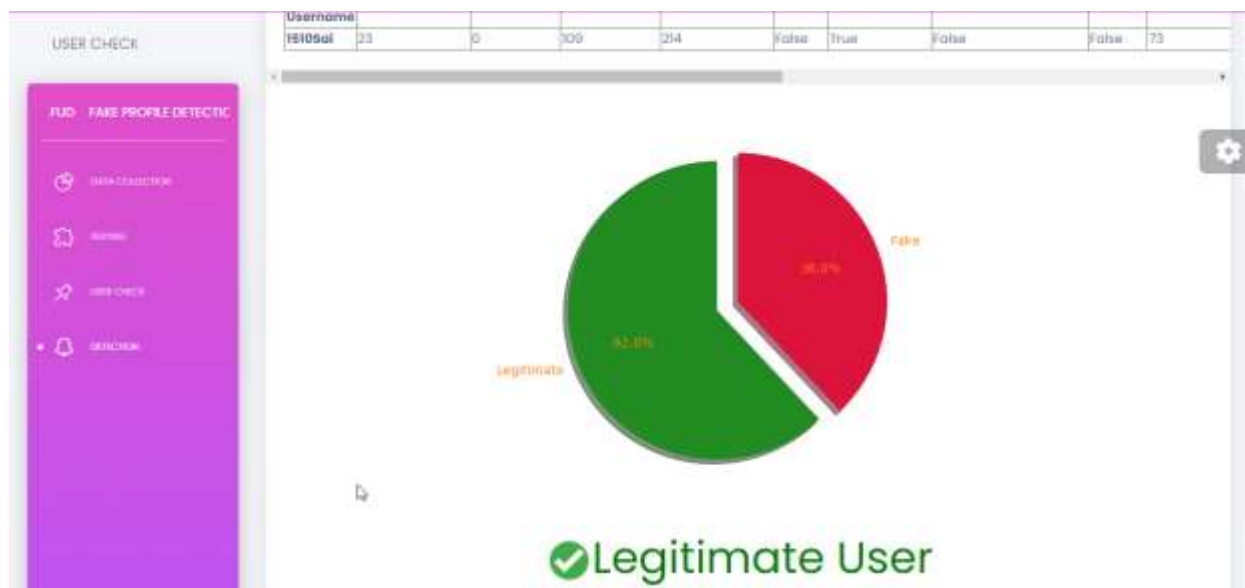


FIG B5. Final Output Detection Page

C.PUBLICATION

A COMPARATIVE STUDY ON FAKE PROFILE IDENTIFICATION USING DIFFERENT MACHINE LEARNING TECHNIQUES

Student name : K Sai Suraj

S Mujafar

Guide name : Dr. P Asha M.E.,PhD

ABSTRACT:

Today, On-Line social networks (OSNs) are turning out to be an ever-increasing number of famous, influencing individuals' prosperity and associating with different informal communities. An organization is an enormous site that goes through numerous exercises, like publicizing, correspondence, booking, promoting, and data creation. Add new companions and keep up the great stuff. Specialists have concentrated on what person to person communication destinations mean for individuals. Some terrible records are utilized for exercises like falsehood and timetables. It is essential to know the awful record. AI strategies were utilized to observe fake records that could confound individuals. The informational index was first handled utilizing diverse python libraries and contrasted with track down the fitting calculation for the given information Endeavors to recognize fake records on open locales are controlled by various calculations. Calculation positioning classes are Default Forest, Neural Network, and Vector Machines, which are utilized to look for fake records.

Keywords: Fake user, online social networks, Machine Learning, Ensemble Approach.

INTRODUCTION

Recently, the On-Line social networks (OSNs) is turning out to be an ever increasing number of well known, influencing individuals' lives and empowering them to associate with various web-based media locales. Networks associate many individuals with key exercises, like promoting, interchanges, stage improvement, publicizing, and data creation. Making new companions, staying in contact, and refreshing them are simple. Analysts have been investigating online media locales to perceive what they mean for individuals. Some terrible records are utilized to distort or create some issues. It is vital to know the awful record. Machine-based techniques have been utilized to look for misdirecting reports. The information is pre-delivered utilizing diverse python libraries, and correlations are made to get the calculation conceivable to match the information gave. Attempting to distinguish fake locations on a public site is portrayed by different Machine Learning calculations. Posting of normal timberland exercises, Neural organization, and Vector Machines calculations used to recognize deceitful records.

LITERATURE SURVEY

Fake Profile Identification in Online Social Networks

Sk.Shama, K.SivaNandini, P.Bhavya Anjali, K. Devi Manaswi/ 2019

Innovation is blasting today. Cell phones are turning out to be increasingly shrewd. Innovation is tied in with observing new companions, observing companions is the Internet of Things in everybody's day to day existence, and it's not difficult to track down their own advantages. In any case, the expansion of long-range interpersonal communication locales leads to a great deal of issues, like deception of their personalities and web-based learning. Island clients have pointless information during swimming, which has been sent by fake clients. Studies have shown that 20-40 percent of online media profiles like Facebook are phony. Along these lines, understanding phony profiles on a public site prompts an answer utilizing a class. It doesn't contain inside and out data about the Facebook app.

Detecting Fake User Accounts on Different Social Media Networks

Sachin Ingle, Satish Borade ,SagarAwasare/ 2019

This will find fake user from multiple Social Networking platform by matching their friends network, matching profile details and their writing styles. In this way, we are matching different user accounts on multiple social network platforms and finding matched account. In this way we can detect fake user. Along with this we are improving efficiency of previous works in this area. To work on these disadvantages, we are proposing a system that can help to detect fake user from multiple social networking platforms and detect those users who have two accounts on different SMN's site with different name means same user creating multiple accounts but different name for hiding identity.

Implications of Various Fake Profile Detection Techniques in Social Networks

Dr. Sanjeev Dhawan, Ekta/ 2018

The motivation behind this page is to cause to notice Facebook and distinguish fakequalifications. Facebook is broadly utilized on informal organizations, where clients can share messages, recordings, and recordings, and clients can add more companions to their profiles. Yet, it is hard to know whether a renewed individual is genuine. It very well may be an awful business. Distinguish culprits or profiles of different specialized cheats wanted. In this article, we have attempted to investigate various methods of correlation dependent on various projects for drawing various aspects. This application is just for programming dependent on client data or client wallets. It does exclude different choices, for example, the Facebook application.

Statistical features-based real-time detection of drifted Twitter spam

C. Chen, Y. Wang, J. Zhang, Y. Xiang, W. Zhou, and G. Min / 2017

The proposed program can distinguish "change" tweets in unwritten tweets and carry out them in preparing. A few examinations have been directed to assess the proposed program. The outcomes show that our Lfun program can extraordinarily further develop spammers who know the reality of the world. There are likewise hindrances to our Lfun program. The upside of "old" spam is that it kills the impacts of "spam drift" so we can post genuine spam tweets before very long.

Automatically identifying fake news in popular Twitter threads

C. Buntain and J. Golbeck / 2018

In request to discover the bogus data on Twitter, we discussed the valid and solid motivations behind why non-proficient models are more than fakemodels. The fundamental exercises are restricted to different exercises among CREDBANK and PHEME, which can change the exchange of force. Assuming the essential designations of our models are altogether unique, contrasts in construction and capacity might be because of more restricted issues than plan abilities.

A performance evaluation of machine learning-based streaming spam tweets detection

C. Chen, J. Zhang, Y. Xie, Y. Xiang, W. Zhou, M. M. Hassan, A. AlElaiwi, and M. Alrubaian / 2015

The huge spam issue on Twitter has effectively grabbed the eye of analysts. A few scientists have concentrated on the idea of spam, trailed by various Twitter spam identification exercises. Accordingly, we examine the means engaged with setting it up in two phases: 1) portrayal, and 2) spam discovery on Twitter. On this page we give an essential outline of ML calculations for looking for spam tweets. To do this examination, we initially gathered a sum of 600 million

tweets. We currently create around 6.5 million messages utilizing Trend Micro's Web Reputation System

A model-based approach for identifying spammers in social networks

F. Fathaliani and M. Bouguessa / 2017

In this article, we will take a gander at how to recognize a spam sender on a public site blended in with highlights dependent on the making of an uncontrolled strategy for distinguishing spammers. In our methodology, we initially address every informal community client with a vector that mirrors their conduct and associations with different members. The exploration introduced in this article shows that our uncontrolled techniques work better (now and again better) than numerous different strategies. Notwithstanding, our methodology permits us to investigate the advantages of genuine without neglected data and study hall access.

Spam detection of Twitter traffic: A framework based on random forests and non-uniform feature sampling

C. Meda, E. Ragusa, C. Gianoglio, R. Zunino, A. Ottaviano, E. Scillia, and R. Surlinelli / 2016

This work offers a wide scope of choices to browse before the Black Box Machine Learning Machine, utilizing an arbitrary woods memory calculation choice to recognize the spam sender in the rush hour gridlock load. The exploration depends on the most famous Twitter information bundles and the new Twitter client information bundles. Luckily, there are numerous Internet clients who use microblogging to disturb and spread terrible things. Distinguishing clients and recognizing spammers is a valuable method for decreasing traffic to Twitter's unknown substance.

EXISTING SYSTEM

Tingmin et al. Examine new ways and strategies to distinguish Twitter spam. The above research shows a relative investigation of innovation. Then again, S. J. Somanet. al. Twitter has explored different practices of spammers via online media. The concentrate additionally incorporated a survey of the books that recognize spam on Twitter. Notwithstanding the exploration, there is as yet a hole in the current writing. So we are trying the furthest down the line innovation to distinguish spammers and recognize fakeclients on Twitter to clear the hole.

DISADVANTAGES EXISTING SYSTEM

Because of Privacy Issues the Facebook dataset is very limited and a lot of details are not made public.

Having less accuracy

More complex

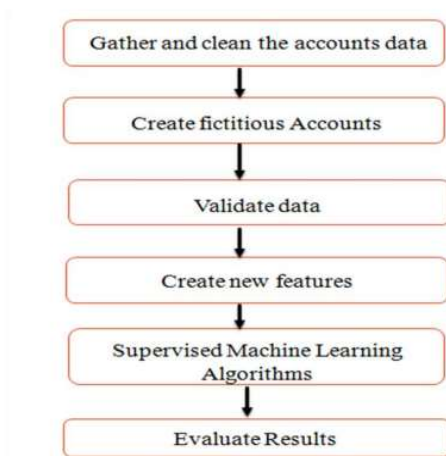
PROPOSED SYSTEM

At the arranged level, various moves should be made to follow the phony profile and gain from the criticism given by the calculation. This level can be effectively carried out by a long-range informal communication organization. 1. The intellectual cycle starts with choosing a test profile. 2. In the wake of choosing the profile, select the properties (i.e. properties) that are suitable for the space where the calculation is carried out. 3. Capacities prohibited via preparing. Plan customary trainings as you give new data and feed. 4. The rundown demonstrates whether it is bogus or valid. 5. Since the classifications can't be 100% all together; orders input thoughts. 6. This cycle is rehashed, and over the long run, no. How much data preparing is expanding and the order of phony names is turning out to be more sensible.

ADVANTAGES OF PROPOSED SYSTEM

Web-based media doesn't work on our lives, yet there are numerous issues with utilizing web-based media. Issues incorporate individual life, online badgering, misuse, and harassing. This is generally done utilizing fakeprofiles. In this venture, we utilized AI to recognize fake personalities and build up ways of securing individuals' lives.

ARCHITECTURE DIAGRAM



DESCRIPTION

The given system has an assortment of AI apparatuses, and the office was executed as depicted previously. The given framework gathers the information given before the calculation level, utilizing which we can decide the phony profile on Twitter by contrasting it and the reality of the most fit calculation on the three AI machines and the given informational index. Calculations There are various ways of recognizing issues dependent on the exhibition, experience, or climate

in which the inspecting system chooses the fitting calculation dependent on the data gave to accomplish the best outcomes.

SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

System : Pentium Dual Core.

Hard Disk : 120 GB.

Monitor : 15'' LED

Input Devices : Keyboard, Mouse

Ram : 4 GB.

SOFTWARE REQUIREMENTS:

Operating system: Windows 7/10.

Coding Language : Python

MODULES:

Data Collection

Pre-Processing

Train and Test

Machine Learning Technique

Detection of Fake Profiles

Data Collection

We will be using a Python Library called Tweepy to connect to the Twitter API and collect the data. We download tweets containing certain key words, to incorporate the words or hash tags that contain relevant keyword related to fake users. Some of the most important fields are:

text, which contains the text included in the tweet.

created at, which is a timestamp of when the tweet was created.

user, which contains information about the user that created the tweet, like the username and user id.

Pre-Processing

We scalarize the data and experiment through the calculation and the capacities put away in the x and the mark in the y.

Train and Test

We show the substance of the metadata anticipated from extra data about the client's tweets, while the substance is planned to see the nature of the client's instant message and the message utilized in the article.

Machine Learning Technique

Support Vector Machine (SVM): Support-vector machines (SVMs), and vector networks are controlled learning strategies and preparing calculations that investigate data utilized in examination and search techniques. In light of the preparation information (supervised learning), the calculation that delivers the best hyperplane records new models. Nerve organization: A muscle network is an organization of neurons, or circles, or in current speech, an organization of living things comprised of fakejobs. For fakejobs, a neural network (NN) is a gathering of regular and fake nerve cells used to process computerized information dependent on a mix of information. Typical memory: Normal calculation memory is calculation control. As the name recommends, this calculation makes backwoods with many trees. By and large, the more trees there are in the backwoods, the more wonderful the timberland looks. Thusly, an enormous number of trees fill in the woodland, which gives genuine outcomes.

Detection of Fake Profiles

Knowing a Fake Profile, a system that gathers a bunch of data created prior to giving the level of the calculation we use can recognize a phony profile on Facebook by contrasting the reality of three machine calculations, calculations, and best practices on a given informational collection. Calculations are various methods of distinguishing an issue dependent on the presentation, experience, or climate in which the examining system chooses the suitable calculation for the data gave to accomplish the best outcomes.

RESULTS

So by the final results we can detect whether the user is legitimate or fake/bot user. In the below figures we show two cases where the user is legitimate and other is fake user and it predicted or detected correctly. In Fig.1 Where I have given my own id it predicted as legitimate user and in the next Fig.3 I have given bot user id and it predicted as fake/bot user. So Algorithm is mentioned as below

1. In the first step we collected data of twitter users.
2. Then we train the model using so much data that we collected which contains various different parameters.

4. In the last step we perform detection or prediction, and here we get the final output.

[illegible]

Fig.1 Giving my twitter id as input



Fig.2 showing that user is legitimate

[illegible]

Fig.3 Giving bot id as input

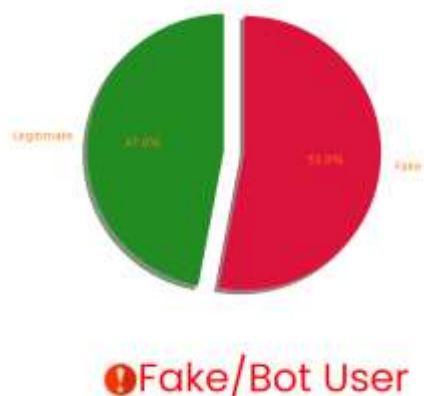


Fig.4 showing that user is fake/bot

CONCLUSION

Understanding position misrepresentation will just direct work searchers to get something authentic for the organization. To battle ladies' extortion, various AI calculations should be tended to in this article. Review techniques are utilized to give different proportions of execution to identify working environment misrepresentation. Research shows that ranger service isn't a distributed device. The normal technique is 98.27% more exact than the current strategy.

REFERENCES

- [1] C. Chen, S. Wen, J. Zhang, Y. Xiang, J. Oliver, A. Alelaiwi, and M. M. Hassan, "Investigating the deceptive information in Twitter spam," *Future Gener. Comput. Syst.*, vol. 72, pp. 319–326, Jul. 2017.
- [2] I. David, O. S. Siordia, and D. Moctezuma, "Features combination for the detection of malicious Twitter accounts," in *Proc. IEEE Int. Autumn Meeting Power, Electron. Comput. (ROPEC)*, Nov. 2016, pp. 1–6.
- [3] M. Babcock, R. A. V. Cox, and S. Kumar, "Diffusion of pro- and anti-false information tweets: The black panther movie case," *Comput. Math. Org. Theory*, vol. 25, no. 1, pp. 72–84, Mar. 2019.
- [4] S. Keretna, A. Hossny, and D. Creighton, "Recognising user identity in Twitter social networks via text mining," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2013, pp. 3079–3082.
- [5] C. Meda, F. Bisio, P. Gastaldo, and R. Zunino, "A machine learning approach for Twitter spammers detection," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2014, pp. 1–6.
- [6] W. Chen, C. K. Yeo, C. T. Lau, and B. S. Lee, "Real-time Twitter content polluter detection based on direct features," in *Proc. 2nd Int. Conf. Inf. Sci. Secur. (ICISS)*, Dec. 2015, pp. 1–4.
- [7] H. Shen and X. Liu, "Detecting spammers on Twitter based on content and social interaction," in *Proc. Int. Conf. Netw. Inf. Syst. Comput.*, pp. 413–417, Jan. 2015.

- [8] G. Jain, M. Sharma, and B. Agarwal, “Spam detection in social media using convolutional and long short term memory neural network,” *Ann. Math. Artif. Intell.*, vol. 85, no. 1, pp. 21–44, Jan. 2019.
- [9] M. Washha, A. Qaroush, M. Mezghani, and F. Sedes, “A topic-based hidden Markov model for real-time spam tweets filtering,” *Procedia Comput. Sci.*, vol. 112, pp. 833–843, Jan. 2017.
- [10] F. Pierri and S. Ceri, “False news on social media: A data-driven survey,” 2019, arXiv:1902.07539. [Online]. Available: <https://arxiv.org/abs/1902.07539>
- [11] S. Sadiq, Y. Yan, A. Taylor, M.-L. Shyu, S.-C. Chen, and D. Feaster, “AAFA: Associative affinity factor analysis for bot detection and stance classification in Twitter,” in *Proc. IEEE Int. Conf. Inf. Reuse Integr. (IRI)*, Aug. 2017, pp. 356–365.
- [12] M. U. S. Khan, M. Ali, A. Abbas, S. U. Khan, and A. Y. Zomaya, “Segregating spammers and unsolicited bloggers from genuine experts on Twitter,” *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 551–560, Jul./Aug. 2018.

PLAGIARISM REPORT

A COMPARATIVE STUDY ON FAKE PROFILE IDENTIFICATION USING DIFFERENT MACHINE LEARNING TECHNIQUES

ORIGINALITY REPORT

9%	9%	6%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	"Table of Contents", IEEE Transactions on Computational Social Systems, 2015 Publication	1%
2	www.jpinfotech.org Internet Source	1%
3	Shivangi Gheewala, Rakesh Patel. "Machine Learning Based Twitter Spam Account Detection: A Review", 2018 Second International Conference on Computing Methodologies and Communication (ICCMC), 2018 Publication	1%
4	researchhr.org Internet Source	1%
5	www.ijrte.org Internet Source	1%
6	www.pantechelearning.com Internet Source	1%
7	www.ijert.org Internet Source	1%

8	www.iosrjournals.org Internet Source	1 %
9	Submitted to Victoria University Student Paper	1 %
10	bionyt.s807.sureserver.com Internet Source	<1 %
11	jpinfotech.org Internet Source	<1 %
12	cps-vo.org Internet Source	<1 %
13	ijettjournal.org Internet Source	<1 %
14	computer.ieeeprojects.org Internet Source	<1 %
15	Faiza Masood, Ghana Ammad, Ahmad Almogren, Assad Abbas, Hasan Ali Khattak, Ikram Ud Din, Mohsen Guizani, Mansour Zuair. "Spammer Detection and Fake User Identification on Social Networks", IEEE Access, 2019 Publication	<1 %

Exclude quotes On
Exclude bibliography On

Exclude matches Off