DETECTION OF MALICIOUS SOCIAL BOTS USING LEARNING AUTOMATA WITH URL FEATURES IN TWITTER NETWORK

[1]Dr. Madhavi Pingili [2]Mrs. M.Jhansi Lakshmi [3]M. Divya, [4]D. Abhishek, [5]K. Naresh, [6]P. Sparsha, ,

[1] Associate Professor and HOD, Dept. of IT, CMR Engineering College, UGC Autonomous,
Kandlakoya, Medchal Road, Medchal Dist, Hyderabad-501 401
[2]Guide-Assistant Professor, Dept. of IT, CMR Engineering College, UGC Autonomous, Kandlakoya
e-mail : mettu.jhansilakshmi@cmrec.ac.in
[3,4,5,6]B.Tech Scholars, Dept. of IT, CMR Engineering College, UGC Autonomous, Kandlakoya,
Medchal Road, Medchal Dist, Hyderabad-501 401

ABSTRACT: Malicious social bots generate fake tweets and automate their social relationships either by pretending like a follower or by creating multiple fake accounts with malicious activities. Moreover, malicious social bots post shortened malicious URLs in the tweet in order to redirect the requests of online social networking participants to some malicious servers. Hence, distinguishing malicious social bots from legitimate users is one of the most important tasks in the Twitter network. To detect malicious social bots, extracting URL-based features (such as URL redirection, frequency of shared URLs, and spam content in URL) consumes less amount of time in comparison with social graph-based features (which rely on the social interactions of users).
Furthermore, malicious social bots cannot easily manipulate URL redirection chains. In this article, a Learning Automata-based Malicious Social Bot Detection (LA-MSBD) algorithm is proposed by integrating a trust computation model with URL based features for identifying trustworthy participants (users) in the Twitter network. The proposed trust computation model contains two parameters, namely, direct trust and indirect trust. Moreover, the direct trust is derived from Bayes' theorem, and the indirect trust is derived from the Dempster Shafer Theory (DST) to determine the trustworthiness of each participant accurately. Experimentation has been performed on two Twitter data sets, and the results illustrate that the proposed algorithm achieves improvement in precision, recall, F-measure, and accuracy compared with existing approaches for MSBD.

Keywords: Malicious bots, Phishing, Twitter network, URL Features, Spam detection.

I. INTRODUCTION

Malicious social bots generate fake tweets and automate their social relationships either by pretending like a follower or by creating multiple fake accounts with malicious activities.

Social media has played a more important role in our daily life. With billions of users producing and consuming information every day, it is a natural extension that people turn to this medium to read and disseminate news. Social media bots are programs that vary in size depending on their function, capability, and design and can be used on social media platforms to do various useful and malicious tasks while stimulating human behaviour. Some social media bots provide useful services, such as weather updates and sports scores. These good social media bots are clearly identified as such and the people who interact with them know that they are bots.

However a large number of social media bots are malicious bots disguised as human users. These bots cause users to lose trust that social media platforms can deliver news honestly, as they become suspicious that the stories they see at the top of their feeds were "pushed" there by manipulative bots. With so many people turning to social media, malicious users like bots have begun to sway the conversations in whatever direction their creators want.

These malicious bots have been used for malicious tasks such as spreading false information about political candidates, inflating the perceived popularity of celebrities, deliberately pushing down the messages of protestors and activists, illicitly advertising by spamming the social web with links to commercial websites and influencing financial markets in an attempt to manipulate the direction of stock prices. Furthermore, these bots can change the results of common analyses performed on social media.

Some of the common attack methods of social media bots are: sleeper bots-they remain dormant for long periods of time, then wake up to launch their attack of thousands of posts in a short period of time(perhaps as a spam attack),and then return to a dormant state, trend jacking - use of top trending topics to focus on an intended audience for targeting purposes, watering hole attack-attacker guesses or observes which websites an organization often uses and infects one or more of them with malware, hashtag hijacking-use of hashtags to focus an attack(e.g. spam ,malicious links)on a specific audience using the same hashtag and click farming or like farming-inflate fame or popularity on a website through liking or reposting of content via click farms.

Bot detection is an important task in social media. Twitter, a popular social media platform, is plagued by automated accounts. Some studies estimated that around 15% of the accounts on Twitter Operates Automatically or Semi-automatically. One reason which might have stimulated the rise of the number of bots is the characteristics of Twitter. In this article, the malicious behavior of participants is analyzed by considering features extracted from the posted URLs (in the tweets), such as URL redirection, frequency of shared URLs, and spam content in URL, to distinguish between legitimate and malicious tweets.
To protect against the malicious social bot attacks, our proposed LA-based malicious social bot detection (LA-MSBD) algorithm integrates a trust computational model with a set of URL-based features for the detection of malicious social bots. However, our work is different from other existing works in the sense that we focus on detecting malicious social bots based on the LA model with the trust computational model. The LA has also been successfully applied in various areas, such as Internet of Things (IOT), cloud computing, social networks, wireless networks, and image processing.

II. LITERATURE SURVEY

G. Lingam, R. R. Rout, and D. V. L. N. Somayajulu et. al [1] Adaptive deep Q-learning model for detecting social bots and influential users in online social networks. In an online social network a botmaster establishes a social relationship among legitimate participants to reduce the probability of social bot detection. Social bots generate fake tweets and spread malicious information by manipulating the public opinion. Therefore, the detection of social bots in an online social network is an important task.

A.K. Jain and B. B. Gupta et. al [2] A machine learning based approach for phishing detection using hyperlinks information. The proposed approach incorporates various new outstanding hyperlink specific features to detect phishing attack. The proposed approach has divided the hyperlink specific features into 12 different categories and used these features to train the machine learning algorithms.

P. Shi, Z. Zhang, and K.-K.-R. Choo et. al [3] Detecting malicious social bots based on clickstream sequences. With the significant increase in the volume, velocity and variety of user data (e.g., user-generated data) in online social networks, there have been attempted to design new ways of collecting and analyzing such big data.

S. Madisetty and M. S. Desarkar et. al [4] A neural network-based ensemble approach for spam detection in Twitter. As the social networking sites get more popular, spammers target these sites to spread spam posts. Twitter is one of the most popular online social networking sites where users communicate and interact on various topics.

D. R. Patil and J. B. Patil et. al [5] Malicious URLs detection using decision tree classifiers and majority voting technique. Researchers all over the world have provided significant and effective solutions to detect malicious URLs. Still due to the ever changing nature of cyber-attacks, there are many open issues.

H. Gupta, M. S. Jamal, S. Madisetty and M. S. Desarkar et. al [6] A framework for real-time spam detection in Twitter. With the increased popularity of online social networks, spammers find these platforms easily accessible to trap users in malicious activities by posting spam messages. In this work, we have taken Twitter platform and performed spam tweets detection.

M. Al-Janabi, E. D. Quincey and P. Andras et. al [7] Using supervised machine learning algorithms to detect suspicious URLs in online social networks. The increasing volume of malicious content in social networks requires automated methods to detect and eliminate such content. This paper describes a supervised machine learning classification model that has been built to detect the distribution of malicious content in online social networks

T. Wu, S. Liu, J. Zhang, and Y. Xiang et. al [8] Twitter spam detection based on deep learning. Twitter spam has long been a critical but difficult problem to be addressed. So far, researchers have developed a series of machine learning-based methods and blacklisting techniques to detect spamming activities on Twitter.

H. B. Kazemian and S. Ahmed et. al [9] Comparisons of machine learning techniques for detecting malicious webpages. The conventional method of detecting malicious webpages is going through the black list and checking whether the webpages are listed. Black list is a list of webpages which are classified as malicious from a user's point of view.

C.-M. Chen, D. J. Guan, and Q.-K. Su et. al [10] Feature set identification for detecting suspicious URLs using Bayesian classification in social networks. Social Network Services (SNSs) are increasing popular. Communicating with friends forms a social network that can be used to promptly share information with friends.

III. PROPOSED METHOD

A Learning Automata-based Malicious Social Bot Detection (LA-MSBD) model is proposed by integrating a trust computation model with URL-based features for identifying trustworthy participants (users) in the Twitter network. In this we will be giving 13 parameters as input to the SVM algorithm which process these inputs and return a single digit either 0 or 1.The results illustrate that the proposed algorithm achieves improvement in precision, recall, Fmeasure, and accuracy compared with existing approaches for MSBD.

IV. SOFTWARE DESIGN

4.1 System Architecture:

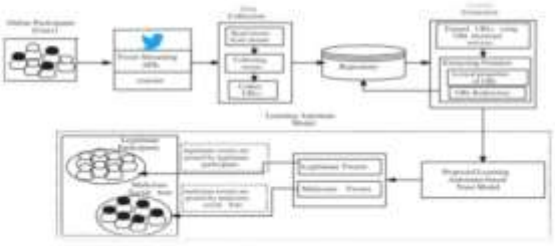


Fig.1: System Architecture

The methodology of review consists following steps

1. Data Collection
2. Feature Extraction
3. Classification
4. Testing
5. URL feature prediction

4.1.1: Data Collection: The dataset was composed of URL's of twitter data set along with their respective identity such as malicious or non malicious. Based upon these data the given input can be classified as either malicious or non-malicious. The dataset which we use in this project is the twitter data set collected.

4.1.2: Feature Extraction: The feature extraction technique plays an important role. The features are the main parameter that are involved for classification of URL. Texture extraction is determined as the example of information or course of action of the structure with random interval.

4.1.3: Classification: In a typical classification system image captured by a camera and then processed. In Supervised classification, most importantly preparing occurred through known gathering of pixels. The numbers of clusters decided by users. When trained pixels are not available, the supervised classification is used that is KNN.

4.1.4: Testing: In the testing phase the URL's are being tested.

4.1.5: URL feature prediction: Finally, we get the URL is either malicious or non malicious.

4.2 Data Flow Diagram:

The flow of execution of the current project is shown in the data flow diagram. The stages are present in the step wise process so that we can follow that and know about the complete details of the project. A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both. A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. A Data Flow Diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement. They are often elements of a formal methodology such as Structured Systems Analysis and Design Method (SSADM).

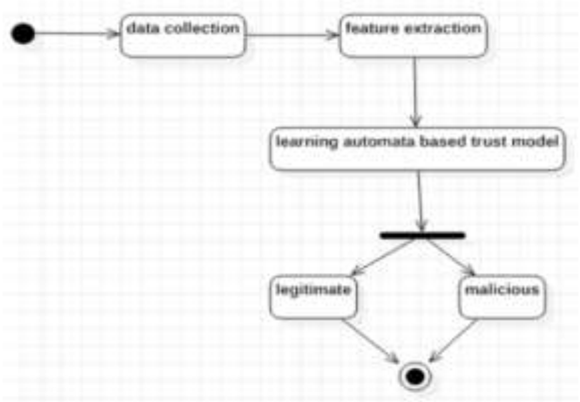


Fig.2: Data Flow Diagram

4.3 UML Diagrams:

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.UML was created by the Object Management Group (OMG) and UML1.0 specification draft was proposed to the OMG in January1997. There are several types of UML diagrams and each one of them serves a different purpose regardless of whether it is being designed before the implementation or after (as part of documentation). UML has a direct relation with object oriented analysis and design. After some standardization, UML has become an OMG standard. The two broadest categories that compass all other types are:
1. Behavioral UML diagram
2. Structural UML diagram.

As then it suggests, some UML diagrams try to analyze and depict the structure of a system or process, whereas other describe the behavior of the system, its actors, and its building components. The different types are broken down as follows:
1. Sequence diagram
2. Use case Diagram
3. Activity diagram
4. Class diagram

4.3.1 Sequence Diagram: A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for and existing systems.

The sequence diagram is a good diagram to use to document a system's requirements and to flush out a system's design. The reason the sequence diagram is so useful is because it shows the interaction logic between the objects in the system in the time order that the interactions take place. In sequence diagrams, combined fragments are logical groupings, represented by a rectangle, which contain the conditional structures that affect the flow of messages. A combined fragment contains interaction operands and is defined by the interaction operator.

A lifeline represents an individual participant in a sequence diagram. A lifeline will usually have a rectangle containing its object name. If its name is "self", that indicates that the lifeline represents the classifier which owns the sequence diagram.

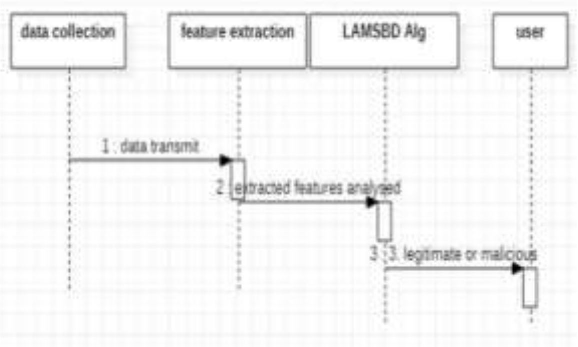


Fig.3: Sequence Diagram

4.3.2 Use case Diagram: A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use case in which the user is involved. A use case diagram is used to structure of the behavior thing in a model. The use cases are represented by either circles or ellipses.

Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally. A use case is a written description of how users will perform tasks on your website. It outlines, from a user's point of view, a system's behavior as it responds to a request. Each use case is represented as a sequence of simple steps, beginning with a user's goal and ending when that goal is fulfilled.

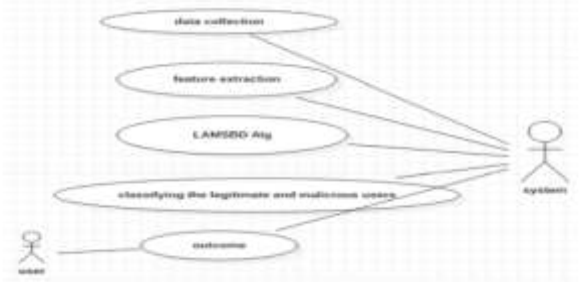


Fig.4: Use case Diagram

Use case:

(a) User need to collect the data.
(b) The URL'S are stored in the dataset.
(c) User need to upload the url.
(d) Few algorithms are used to find the malicious URL's.

4.3.3Activity Diagram: Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent.

An activity diagram shows business and software processes as a progression of actions. These actions can be carried out by people, software components or computers. Activity diagrams are used to describe business processes and use cases as well as to document the implementation of system processes.

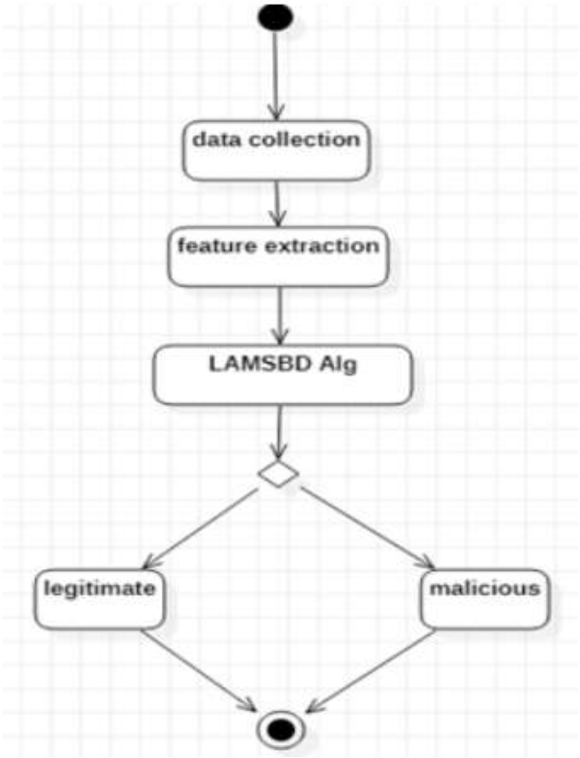


Fig.5: Activity Diagram

4.3.4 Class Diagram: A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity. Class diagram mainly consists of classes Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

There are 3 main sections in class diagram Upper section: Contains the name of the class. This section is always required,• whether you are talking about the classifier or an object. Middle section: Contains the attributes of the class.• Bottom section: Includes class operations (methods).•

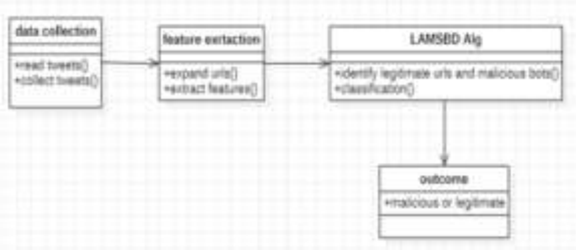


Fig.6: Class Diagram

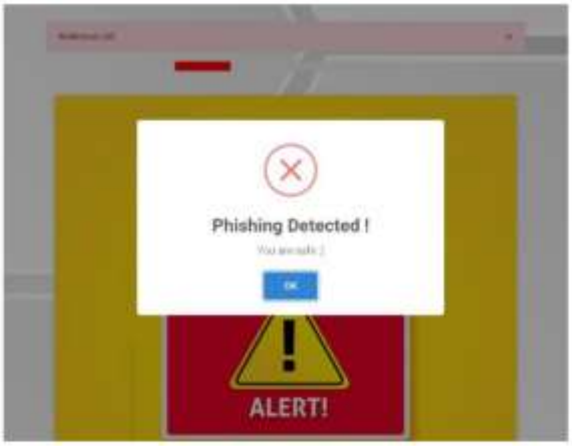
V. OUTPUT SCREENS

Fig.7: Home page

This is the home page in this page we will upload the URL to detect whether the given URL is either malicious or not.

Fig.8: Inserting an URL

Here we upload the URL took from different resources.



Phishing Detected !

Fig.9: Malicious URL Detected

In the figure the given URL is detected as malicious that may install a malicious bot in our device.

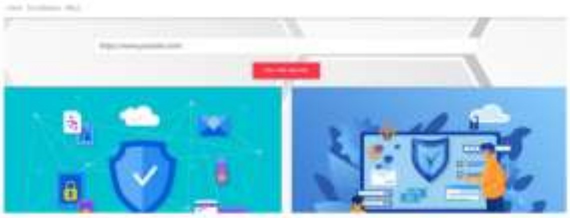
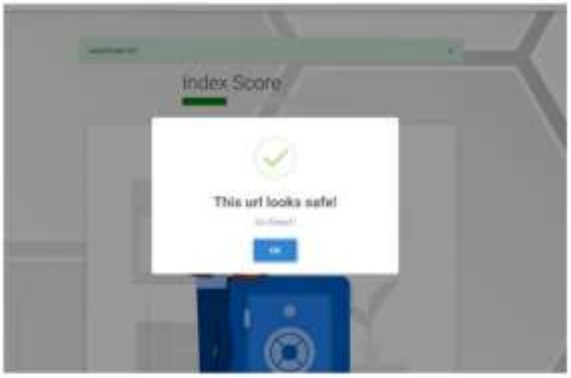


Fig.10: Uploading an URL

Here we upload the URL took from different resources.



This url looks safe!

Fig.11: Phishing Detected

The given URL is non Malicious and the user can use the URL.

VI. CONCLUSION & FUTURE SCOPE

This article presents an LA-MSBD algorithm by integrating a trust computational model with a set of URL-based features for MSBD. In addition, we evaluate the trustworthiness of tweets (posted by each participant) by using the Bayesian learning and DST. Moreover, the proposed LAMSBD algorithm executes a fifinite set of learning actions to update action probability value (i.e., probability of a participant posting malicious URLs in the tweets).

The proposed LA-MSBD algorithm achieves the advantages of incremental learning. Two Twitter data sets are used to evaluate the performance of our proposed LA-MSBD algorithm. The experimental results show that the proposed LA-MSBD algorithm achieves upto 7% improvement of accuracy compared with other existing algorithms. For The Fake Project and Social Honeypot data sets, the proposed LA-MSBD algorithm has achieved precisions of 95.37% and 91.77% for MSBD, respectively. Furthermore, as a future research challenge, we would like to investigate the dependence among the features and its impact on MSBD.

FUTURE SCOPE:

The Regression methods or the algorithms using the regression will be providing more accuracy than the classification algorithms. As we used SVM algorithm using classification in the proposed system we will be using SVM algorithm using Regression in the future so that the accuracy will be increased.

VII. REFERENCES

[1] G. Lingam, R. R. Rout, and D. V. L. N. Somayajulu, “Adaptive deep Q-learning model for detecting social bots and influential users in online social networks,”

Appl. Intell., vol. 49, no. 11, pp. 3947–3964, Nov. 2019.

[2] A.K. Jain and B. B. Gupta, “A machine learning based approach for phishing detection using hyperlinks information,” J. Ambient Intell. Hum. Comput., vol. 10, no. 5, pp. 2015–2028, May 2019.

[3] P. Shi, Z. Zhang, and K.-K.-R. Choo, “Detecting malicious social bots based on clickstream sequences,” IEEE Access, vol. 7, pp. 28855–28862, 2019.

[4] S. Madisetty and M. S. Desarkar, “A neural network-based ensemble approach for spam detection in Twitter,” IEEE Trans. Comput. Social Syst., vol. 5, no. 4, pp. 973–984, Dec. 2018.

[5] D. R. Patil and J. B. Patil, “Malicious URLs detection using decision tree classifiers and majority voting technique,” Cybern. Inf. Technol., vol. 18, no. 1, pp. 11–29, Mar. 2018.

[6] H. Gupta, M. S. Jamal, S. Madisetty, and M. S. Desarkar, “A framework for real-time spam detection in Twitter,” in Proc. 10th Int. Conf. Commun. Syst. Netw. (COMSNETS), Jan. 2018.

[7] M. Al-Janabi, E. D. Quincey, and P. Andras, “Using supervised machine learning algorithms to detect suspicious URLs in online social networks,” in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining, Jul. 2017.

[8] T. Wu, S. Liu, J. Zhang, and Y. Xiang, “Twitter spam detection based on deep learning,” in Proc. Australas. Comput. Sci. Week Multiconf. (ACSW), 2017.

[9] H. B. Kazemian and S. Ahmed, “Comparisons of machine learning techniques for detecting malicious webpages,” Expert Syst. Appl., vol. 42, no. 3, pp. 1166–1177, Feb. 2015.

[10] C.-M. Chen, D. J. Guan, and Q.-K. Su, “Feature set identification for detecting suspicious URLs using Bayesian classification in social networks,” Inf. Sci., vol. 289, pp. 133–147, Dec. 2014.