Detection of Social Media Scam Using Machine Learning

Guru SS[1], K.P.Arun Kumar[2], Kethireddy Sathwik[3]
[1, 2, 3] Dept of computer science
[1, 2, 3] Prathyusha engineering college

Abstract- *The modern way of keeping in touch with people we know and being aware of current is through social media. The demand form it has grown worldwide and also the scam is increased the way proportional to it. In India, the use of social media is quite high and mostly among the young people. In this paper, we discussed about the scam that happens in TWITTER. The most used tweets are processed as data and made as clean tweets using regular expression process. Stream clustering methods have been repeatedly used for spam filtering in order to categorize input messages or tweets into spam and non-spam clusters. Followed by data processing, the machine learning algorithms are used. We use four algorithms, and among that naive bayes algorithm gives the best result. A set of incremental Naive Bayes (INB) classifier is trained for micro clusters whose population exceeds a Threshold. The compared result implied the superiority of our method to the rivals in almost the datasets. After the algorithm, the final result page. In the final result page, Once we enter the word, it differentiates the ham word and the spam word. These were the modules used in the project. By this, the user gets some idea about the scam happening in social media.*

I. INTRODUCTION

1.1 OVERVIEW

Social networking sites have become very popular in recent years. Users use them to find new friends, updates their existing friends with their latest thoughts and activities. Among these sites, Twitter is the fastest growing site. The demand form it has grown worldwide and also the scam is increased the way proportional to it. In India, the use of social media is quite high and mostly among the young people. In this paper, we discussed about the scam that happens in TWITTER. Its popularity also attracts many spammers to infiltrate legitimate user's accounts with a large amount of spam messages. In this paper, we discuss some user-based and content- based features that are different between spammers and legitimate users. Then, we use these features to facilitate spam detection. Using the API methods provided by Twitter, we crawled active Twitter users, their followers/following information and their most recent 100 tweets. Then, we evaluated our detection

scheme based on the suggested user and content-based features. Stream clustering methods have been repeatedly used for spam filtering in order to categorize input messages or tweets into spam and non-spam clusters. Followed by data processing, the machine learning algorithms are used. We use four algorithms, and among that naïve bayes algorithm gives the best result. A set of incremental Naïve Bayes (INB) classifier is trained for micro clusters whose population exceeds a Threshold. The compared result implied the superiority of our method to the rivals in almost the datasets. After the algorithm, the final result page. Our results show that among the four classifiers we evaluated, the Random Forest classifier produces the best results. Our spam detector can achieve 95.7% precision and 95.7% F-measure using the Naïve bayes Algorithm.

1.2 OBJECTIVE

To enhance the assigning accuracy of former methods in spam detection in Twitter using advanced methods. This project aims at classifying and finding the accuracy of the detection.

1.3 LITERATURE SURVEY

[1] Chen et al., "A performance evaluation of machine learning-based streaming spam tweets detection," IEEE Trans. Computer. Social Syst., Vol. 2, no. 3, pp. 65-76, Sep. 2017. Evaluating the solution with four different machine learning algorithms namely - Support Vector Machine, Neural Network, Random Forest and Gradient Boosting. With Neural Network, we are able to achieve an accuracy of 91.65% and surpassed the existing solution by approximately 18%.

[2] H. Tajalizadeh and R. Boostani, "A Novel Clustering Framework for Stream Data Un nouveau cadre de classifications pour les données de flux," Can. J. Elect. Comput. Eng., Vol. 42, no. 1, pp. 27–35, Sep. 2018. This project proposes a content-based approach to filter spam tweets. We have used the text in the tweet and machine learning and compression algorithms to filter those undesired tweets.

[3] I. Inuwa-Dutse, M. Liptrott, and I. Korkontzelos, “Detection of spam posting accounts on Twitter,” Neurocomputing, Vol. 315, no.2, pp. 496–511, Nov. 2018. This system evaluated the detection scheme based on the suggested user and content- based features. Our results show that among the four classifiers we evaluated, the Random Forest classifier produces the best results. Our spam detector can achieve 95.7% precision and 95.7% F-measure using the Random Forest classifier.

[4] T. Wu, S. Liu, J. Zhang, and Y. Xiang, “Twitter spam detection based on deeplearning,” in Proc. Australas. Comput. Sci. Week Multiconf., Geelong, Australia, Vol. 544, no.2, pp. 101-121, Oct. 2017. This project system proposed a

taxonomy based on different feature selection methods and analyses, namely content analysis, user analysis, tweet analysis, network analysis, and hybrid analysis. Then, we present numerical analyses and comparative studies on current approaches, coming up with open challenges that help researchers develop solutions in this topic.

II. SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

Stream clustering methods have been repeatedly used for spam fifiltering in order To categorize input messages/tweets into spam and non-spam clusters. These methods assume each cluster contains a number of neighbor small (micro) clusters, where each micro-cluster has a symmetric distribution. Nonetheless, this assumption is not necessarily correct and big micro clusters might have asymmetric distribution. To enhance the assigning accuracy of former methods in their online phase, we suggest replacing by machine learning classififiers. Our results show that among the four classifiers we evaluated, the Random Forest classifier produces the best.

2.1.1 DISADVANTAGES

- Accuracy of the system is low

2.2 PROPOSED SYSTEM

We discuss some user-based and content-based features that are different between spammers and legitimate users. Then, we use these features to facilitate spam detection. Using the API methods provided by Twitter, we crawled active Twitter users, their followers/following information and their most recent 100 tweets. Then, we evaluated our detection scheme based on the suggested user and content-based features. Our results show that among the four classifiers we evaluated, the Random forest classifier produces the best results**.**

2.2.1 ADVANTAGES

- Accuracy of the system is enhanced in this method.
- Random classifier gives the best results.

III. SYSTEM REQUIREMENTS

The requirements specification is a technical specification of requirements for the software products. It is the first step in the requirements analysis process it lists the requirements of a particular software system including functional, performance and security requirements. The requirements also provide usage scenarios from a user, an operational and an administrative perspective. This describes the projects target audience and its user interface, hardware and software requirements.

3.1 HARDWARE REQUIREMENTS

- Hard Disk : 500GB and Above
- RAM : 4GB and Above
- Processor : I3 and Above

3.2 SOFTWARE REQUIREMENTS

- Operating System : Windows 7 , 8, 10 (64 bit)
- Software : Python 3.7
- Tools : Anaconda (Jupyter Note Book IDE)

3.3 SOFTWARE DESCRIPTION

Python 3.7 - Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. Python is a programming language that lets you work quickly and integrate systems more efficiently.

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.

- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

JUPYTER NOTEBOOK

Jupyter Lab is a web-based interactive development environment for Jupyter notebooks, code, and data. Jupyter Lab is flexible configure and arrange the user interface to support a wide range of workflows in data science, scientific computing, and machine learning. Jupyter Lab is extensible and modular: write plugins that add new components and integrate with existing ones. The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. The uses include data cleaning and transformation, numerical simulation, statistical modelling, data visualization, machine learning, and much more.

ANACONDA PROMPT

Anaconda command prompt is just like command prompt, but it makes sure that you are able to use anaconda and conda commands from the prompt, without having to change directories or your path. When you start Anaconda command prompt, you'll notice that it adds/("prepends") a bunch of locations to your PATH.

IV. SYSTEM DESIGN

System design is the process of planning a new system or to replace the existing system. Simply, system design is like the blueprint for building, it specifies all the features that are to be in the finished product.

4.1 SYSTEM ARCHITECTURE

System architecture is the conceptual model that defines the structure, behaviour and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviour of the system.

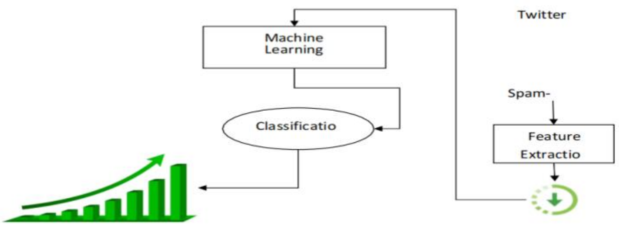


Figure □□□ Architecture Diagram for social media spam detection.

4.2 UML DIAGRAM

4.2.1 USECASE DIAGRAM

Use Cases are used to describe the visible interactions that the system will have with users and external systems. They are used to describe how a user would perform their role using the system. In our project, user can take up the test only after login and all the scores are stored in database which can be shown to user in graphical format.

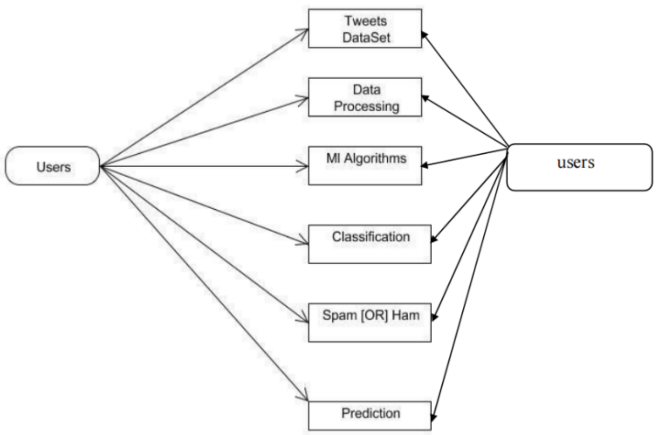


Figure 4.2.1 Use case Diagram for social media spam detection.

4.2.2 CLASS DIAGRAM

The class diagram is a static diagram. It represents he static view of an application. 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.

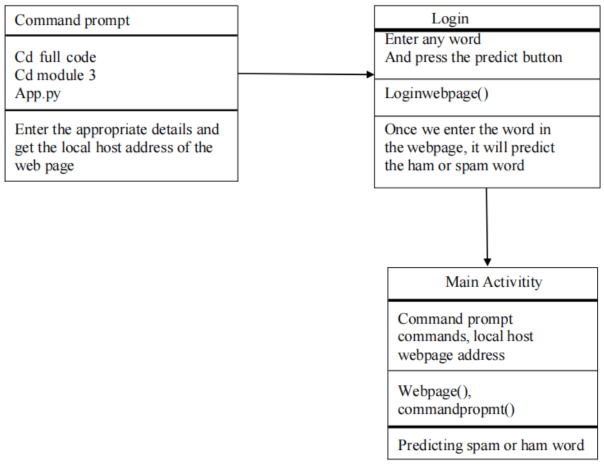

Command prompt
Cd full code
Cd module 3
App.py
Enter the appropriate details and get the local host address of the web page
Login
Enter any word
And press the predict button
Loginwebpage()
Once we enter the word in the webpage, it will predict the ham or spam word
Main Activitity
Command prompt commands, local host webpage address
Webpage(), commandpropmt()
Predicting spam or ham word

Figure 4.2.2 Class Diagram for social media spam detection.

4.2.3 ACTIVITY DIAGRAM

Activity diagram is a graphical representation of workflows of stepwise activities and actions with support for choice, iteration and concurrency. An activity diagram shows the overall flow of control. The most important shape types:

- Rounded rectangles represent activities.
- Diamonds represent decisions.
- Bars represent the start or end of concurrent activities.
- A black circle represents the start of the workflow.
- An encircled circle represents the end of the workflow.

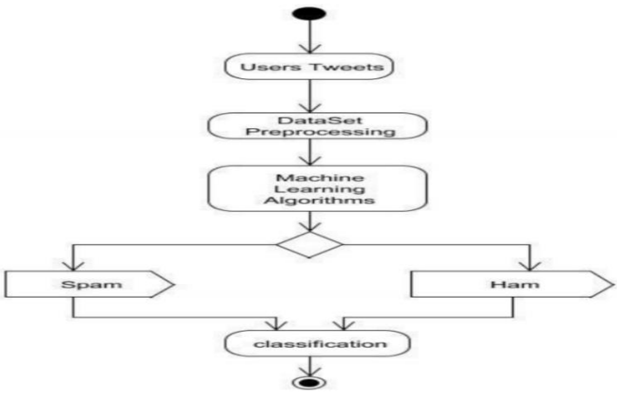


Figure 4.2.3 Activity Diagram for social media spam detection.

4.2.4 SEQUENCE DIAGRAM

A Sequence diagram 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. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

- Represents the details of uml usecase.
- It’s easy to learn it to the level of understanding and it’s nicely adopted by business.
- It’s quite unequivocal so people don’t have different interpretations of the same model.

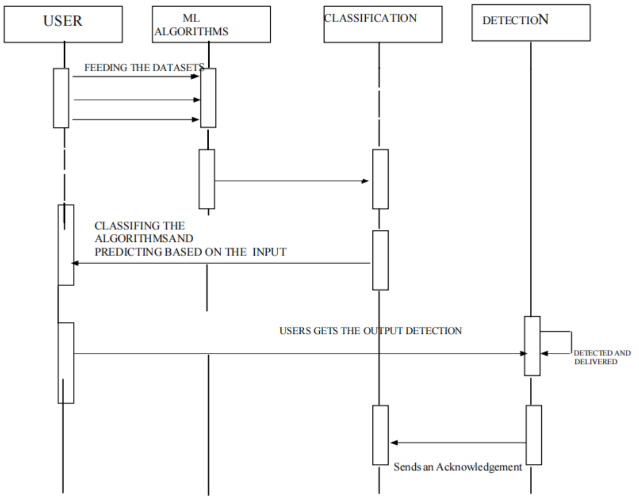


Figure 4.2.4 Sequence Diagram for social media spam detection.

4.2.5 COLLABORATION DIAGRAM

A collaboration diagram, also called as communication diagram or interaction diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). The concept is more than a decade old although it has been refined as the modeling paradigms have evolved.

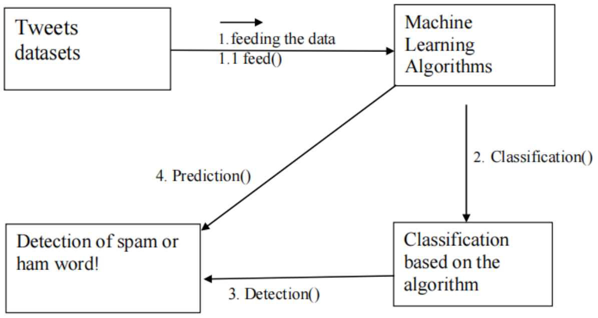

Tweets datasets
1. feeding the data
1.1 feed()
Machine Learning Algorithms
2. Classification()
4. Prediction()
Detection of spam or ham word!
3. Detection()
Classification based on the algorithm

Fig 4.2. 4 Collaboration diagram for social media spam detection.

V. SYSTEM IMPLEMENTATION

5.1 LIST OF MODULES

- Data Preprocessing for spam detection.
- Classification of spam datasets based on ML Algorithms.
- Performance Statistics of spam detection.

5.2 MODULE DESCRIPTION

5.2.1 Data Preprocessing for spam detection

The first and foremost step in data processing is collecting the dataset. We have collected a dataset based on Twitter spam data. The dataset is a CSV file format data which consists of n number of Twitter spam Data. We need to select or extract the features from the collected dataset. Then the Data Cleaning should be initiated. Thus, in this module data preprocessing will be completed.

5.2.2 Classification of spam datasets based on ML algorithms

We evaluated our detection scheme based on the suggested user and content-based features. Our results show that among the three classifiers.

- Decision Trees Classifier
- Support Vector Classifier
- Random Forest Algorithm
- Naive Bayes Algorithm

DECISION TREE CLASSIFIER:

Decision tree classifiers are used successfully in many diverse areas. Their most important feature is the capability of capturing descriptive decision making knowledge from the supplied data. Decision tree can be generated from training sets. The procedure for such generation based on the set of objects (**S**), each belonging to one of the classes **C**1, **C**2, …, **C**k .

Types of Decision Tree

Categorical Variable Decision Tree: Decision Tree which has a categorical target variable then it called a **Categorical variable decision tree.**

Continuous Variable Decision Tree: Decision Tree has a continuous target variable then it is called **Continuous Variable Decision Tree.**

SUPPORT VECTOR CLASSIFIER:

A support vector machine is a supervised learning algorithm that sorts data into Two categories. It is trained with a series of data already classified into two categories, building the model as it is initially trained. The task of an SVM algorithm is to determine which category a new data point belongs in. This makes SVM a kind of non-binary linear classifier. An SVM algorithm should not only place objects into categories, but have the margins between them on a graph as wide as possible.

RANDOM FOREST ALGORITHM:

Random forests or **random decision forests** are an ensamble learning method for classification, regression and other tasks that operates by constructing a multitude of decision tree at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees.

NAIVE BAYES ALGORITHM

In statistics, naïve bayes classifiers are a family of simple probabilistic classifiers based on applying bayes's theorem with strong independence assumptions between the features. They are among the simplest Bayesian network models, but coupled with kernel destiny estimation, they can achieve higher accuracy levels.

Performance Statistics of spam datasets

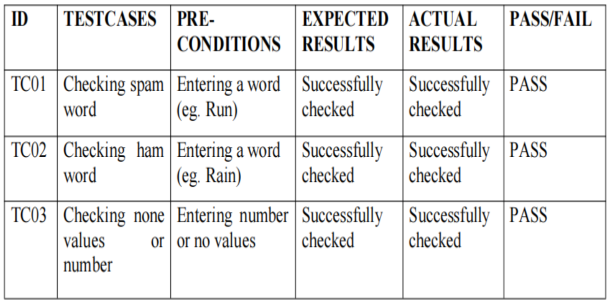
This project results show that among the four classifiers we evaluated, the Naive Bayes algorithm produces the best results**.** The bar graph that shows us the accuracy level of the classifier and the algorithm we used in the project.

VI. TESTING

Testing is the process of executing a program or application with the intent of finding software bugs, and to verify that the software product is fit for use.

6.1 UNIT TESTING

Unit testing focuses verification efforts on the smallest unit of software design in the module. This is also known as module testing. The module of the system is tested separately. This testing is carried out during programming stage itself. In this testing step each module is found to working satisfactorily as regard to the expected output from the module. In this project, all statements are executed properly.



ID	TESTCASES	PRE-CONDITIONS	EXPECTED RESULTS	ACTUAL RESULTS	PASS/FAIL
TC01	Checking spam word	Entering a word (eg. Run)	Successfully checked	Successfully checked	PASS
TC02	Checking ham word	Entering a word (eg. Rain)	Successfully checked	Successfully checked	PASS
TC03	Checking none values or number	Entering number or no values	Successfully checked	Successfully checked	PASS

6.1.1 TEST CASES FOR SPAM DETECTION

6.2 INTEGRATION TESTING

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before verification testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

6.3 SYSTEM TESTING

The listed tests were conducted in the software at the various development stages. Unit testing was conducted. The errors were debugged was performed. The integration testing will be performed once the system is integrated. The results were analyzed, and the appropriate alterations were made. The test results proved to be positive and henceforth this project is feasible, and test approved.

VII. RESULTS AND DISCUSSION

7.1 RESULTS

We have used four machine learning algorithms and predicted the spam and the ham words. Firstly, the spam and ham words are differentiated and uploaded as a dataset. Using this mechanism, we can effectively find out the illegal activities and fake news which are getting trending in the twitter. So, using this project people can differentiate the good and evil and make an awareness for the people and be alert on the fake news.

7.2 DISCUSSION

Twitter data and machine learning algorithms or approaches can be leveraged and applied on various platforms. As the situation rapidly evolves, several topics are consistently dominant or trending on twitter. Trending topics are sometimes believed to be true even though it is baseless. Hence, this project is much useful for the users to make them aware from this evil issues. We have implemented this project using machine learning algorithm, so it is well secured, effectively work on the project. This would be very useful for the common mob to aware themselves from the fake news and illegal activities.

VIII. CONCLUSION AND FUTURE ENHANCEMENT

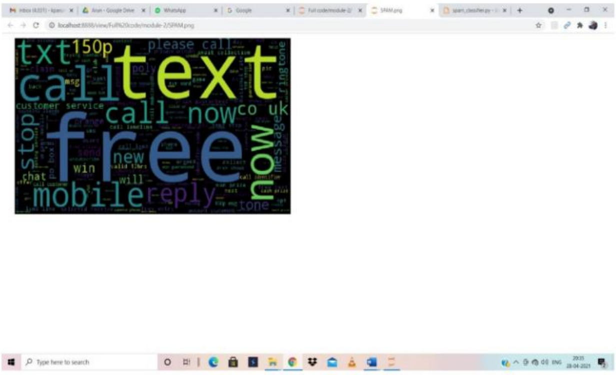
8.1 CONCLUSION

This project infers that through this we are detecting the social media Scam using machine learning, will be very effective in finding the social media spam and also aware and help the people to stay away from it. Trending topics are sometimes believed to be true even though it is baseless. Hence, this project is much useful for the users to make them aware from this evil issues. We have implemented this project using machine learning algorithm, so it is well secured, effectively work on the project. This would be very useful for the common mob to aware themselves from the fake news and illegal activities.

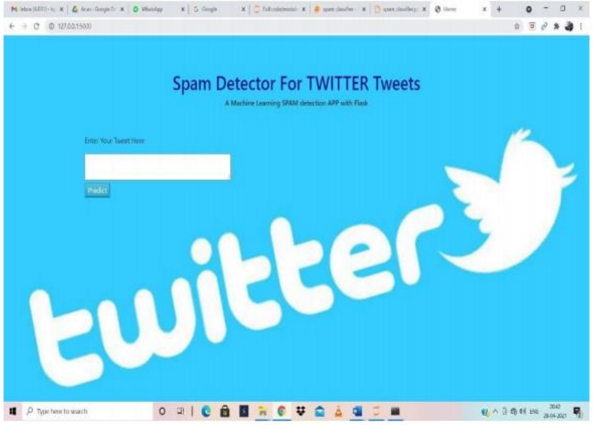
8.2 FUTURE ENHANCEMENT

In this project, the Naive bayes algorithm gives a best results. Here, rather creating a web pages and evaluating, the technologies can be used to give a popup warning message when detected the spam to the user. Also, from where the fake news is originated can also be founded in the future.

IX. OUTPUT SCREENSHOTS



SPAM DATASET

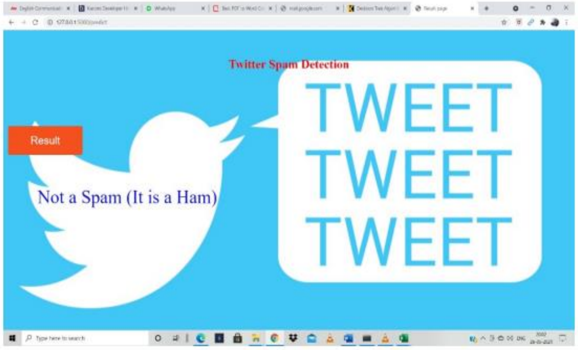


Spam Detector For TWITTER Tweets

A Machine Learning SPAM detection APP with Flask

Enter Your Tweet Here

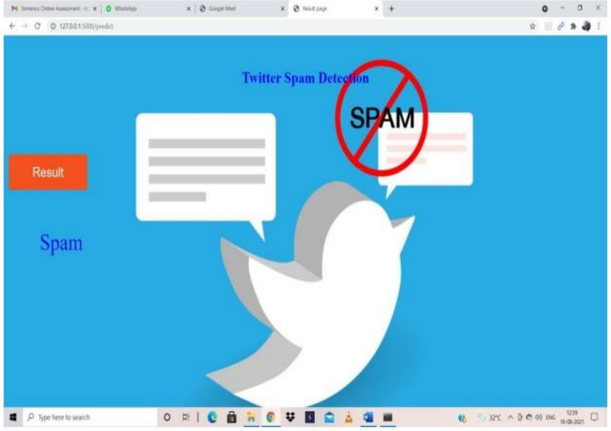
INPUT PAGE FOR SPAM DETECTION



Twitter Spam Detection

Not a Spam (It is a Ham)

RESULT PAGE FOR HAM WORD



Twitter Spam Detection

Spam

RESULT PAGE FOR SPAM WORD

REFERENCES

[1] C. Chen et al., “A performance evaluation of machine learning-based streaming spam tweets detection,” IEEE Trans. Computer. Social Syst., Vol. 2, no. 3, pp. 65-76, Sep. 2015.

[2] X. Zheng, Z. Zeng, Z. Chen, Y. Yu, and C. Rong, “Detecting spammers on social networks,” Neurocomputing, Vol. 159, pp. 27–34, Jul. 2015.

[3] H. Tajalizadeh and R. Boostani, “A Novel Clustering Framework for Stream Data Un nouveau cadre de classifications pour les données de flux,” Can. J. Elect. Comput. Eng., Vol. 42, no. 1, pp. 27–33, Oct 2018.

[4] I. Inuwa-Dutse, M. Liptrott, and I. Korkontzelos, “Detection of spamposting accounts on Twitter,” Neurocomputing, Vol. 315, no. 2, pp. 496–511, Nov2018.

[5] S. Sedhai and A. Sun, “Semi-supervised spam detection in Twitter stream,” IEEE Trans. Comput. Social Syst., Vol. 5, no. 1, pp. 169–175, Mar. 2018.

[6] S. Sedhai and A. Sun, “HSpam14: A collection of 14 million tweets for hashtag- oriented spam research,” in Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Santiago, Chile, Vol. 12, no.2, pp. 223–232, Dec2016.

[7] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, “Finding deceptive opinion spam by any stretch of the imagination,” in Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol., Vol. 1, no.4, pp. 309–319, Jan2017.

[8] T. Wu, S. Liu, J. Zhang, and Y. Xiang, “Twitter spam detection based on deep learning,” in Proc. Australas. Comput. Sci. Week Multiconf., Geelong, Australia, Vol. 685, no. 5, pp. 205-216, june2018.

[9] G. Roffo, S. Melzi, U. Castellani, and A. Vinciarelli, “Infinite latent feature selection: A probabilistic latent graph-based ranking approach,” in Proc. IEEE Int. Conf. Comput. Vis., Vol.55, no.3, pp. 1398–1406, Oct. 2017.