ISSN (Online): 2456-0448
International Journal of Innovative Research in Management, Engineering, and Technology
Vol. 7, Issue 3, March 2022

Designing Malware Detection Scheme To Handling New Edge Cases

[1] Ms.A.Sunitha M.E,[2] Ms.Arockia Jayachrista, .J, [3] Ms.Uma .E

[1] Assistant Professor Of Computer Science and engineering, St.Joseph College Of Engineering,Sriperumudur,Chennai

[2] Student Of Computer Science and engineering, St.Joseph College Of Engineering,Sriperumudur,Chennai

[3] Student Of Computer Science and engineering, St.Joseph College Of Engineering,Sriperumudur,Chennai.

Abstract ***With the raised quality of online social networks, spammers realize these platforms are simple to lure users into malicious activities by posting spam messages in the comments section of the videos. In this work, YouTube comments have been taken and spam detection is performed. To stop spammers, Google Safe Browsing and YouTube Bookmaker tools detect and block spam YouTube. These tools will block malicious links, however they cannot protect the user in real-time as early as possible. Thus, industries and researchers have applied completely different approaches to form spam free social network platform. The survey for the spam comments detection methodology has been carried out The bag-of-words model does exactly we want, that is to convert the phrases or sentences and counts the number of times a similar word appears. In the world of computer science, a bag refers to a data structure that keeps track of objects like an array or list does, but in such cases the order does not matter and if an object appears more than once, we just keep track of the count rather we keep repeating them. The most notable procedures and of their suitability to the issue of spam if we still wanted to reduce very common words and highlight the rare ones, what we would need to do is record the relative importance of each word rather than its raw count. This is known as term frequency inverse document frequency (TF-IDF), which measures how common a word or term is in the document.***

1. INTRODUCTION

In the previous years, informal online communities like Face book and YouTube have become progressively common platform in an individual person's day to day life. People use social media as a virtual community platform to stay in touch with friends and family and to also share thoughts and ideas in blogs. Due to this developing pattern, these platforms pull in an enormous number of clients and are easy targets for spammers. YouTube has become the most well-known informal community among youngsters. For example, many makeup tutorials have been started by bloggers who are referred to as "beauty guru" or "beauty influencers" in which majority of the audiences are teenage girls. These days, 200 million clients produce 400 million new YouTube content (videos) every day. This extensive environment provided by YouTube also creates an opportunity for spammers to create irrelevant content directed to users. These irrelevant or unsolicited messages are aimed to attack users by luring them into clicking links to view malicious sites containing malware, phising and scams. One of the most highlighted features of YouTube is the comments section below every video posted by a user. This feature allows users to share opinions and ideas. In this project, the prediction of the spam comments present in the comments section of Youtube videos using the concept called machine learning, it is also known as subset of artificial intelligence, is done. Supervised learning approach depends on a very large number of labeled datasets.The proposed classification algorithm (Logistic Regression) is used in order to predict the spam comment. The purpose of project is to introduce briefly the techniques of machine learning and to outline the prediction technique. Being much more superior to the conventional data analysis techniques, machine learning can open a new opportunity to explore and increase the prediction accuracy. Spam remarks are regularly completely immaterial to the given video and are normally created via mechanized bots camouflaged as a client. The comments section is target by spammers to post completely irrelevant messages, comments, links and ideas. AI is the strategy for extraction, changing, stacking and anticipating the significant data from enormous information to remove a few examples and furthermore change it into justifiable structure for additional utilization. Grouping and expectation are two sorts of dissecting information which portray principal classes of information and forecast of patterns in future information. The noxious spam remarks will ruin the positive perspective of the contents present in the videos posted. The contingency for anticipating the spam remarks has started but has yet not been concluded and built up for an exact forecast of spam remarks.

AIM & OBJECTIVES

YouTube, the world's largest video sharing site, was founded in 2005 and acquired by Google in 2006. YouTube has grown tremendously as a video content platform, with the recent shift in online content to video. At present, more than 400 hours of video are uploaded and 4.5 million videos are watched every minute on YouTube. It is easy for users to watch and upload videos without any restrictions. This great accessibility has increased the number of personal media, and some of them have become online influencers. YouTube creators can monetize if they have more than 1,000 subscribers and 4,000 hours of watch time for the last 12 months. Accordingly, spam comments are being created to promote their channels or videos in popular videos. Some creators closed the comment function due to aggression such as political comments, abusive speech, or derogatory comments not related to their videos.

SCOPE:

Research on detecting spam content and users focus on various fields. Many studies focused on spam on websites (e.g., portal sites and blogs).As YouTube gains popularity as a video sharing platform, spammers target it with low quality content or promotions. Since spammers that harm the YouTube community are increasing, detecting them becomes an interesting source to research. So, we divide the literature of detecting spam into two sections, spam on websites and spam on YouTube.

II.LITERATURE REVIEW

TITLE NAME: A YouTube Spam Comments Detection Scheme Using Cascaded Ensemble Machine Learning Model
AUTHOR: HAYOUNG OH
YEAR: 2021
ABSTRACT: This paper proposes a technique to detect spam comments on YouTube, which have recently seen tremendous growth. YouTube is running its own spam blocking system but continues to fail to block them properly. Therefore, we examined related studies on YouTube spam comment screening and conducted classification experiments with six different machine learning techniques (Decision tree, Logistic regression, Bernoulli Naïve Bayes, Random Forest, Support vector machine with linear kernel, Support vector machine with Gaussian kernel) and two ensemble models (Ensemble with hard voting, Ensemble with soft voting) combining these techniques in the comment data from popular music videos - Psy, Katy Perry, LMFAO, Eminem and Shakira.

TITLE NAME: Sentiment Analysis for Youtube Videos with User Comments : Review
AUTHOR: Rawan Fahad Alhujaili, Wael M.S. Yafooz
YEAR: 2021
ABSTRACT: Sentiment analysis is a process that discovers the user opinions and views against any service or a product. YouTube is one of the most popular videos sharing platforms obtaining millions of views. These receive several comments, containing valuable information that helps in improving the rating levels of the uploaded content. These comments are utilized by using natural language processing techniques and machine learning techniques. There are many attempts had been proposed scholarly with two (positive or negative), three (two with neutral) or multiple (happy, sad, fear, surprise and anger) classes. However, it is challenging to choose the best accurate model. Therefore, there had been attempts to use sentiment analysis on YouTube comments in identifying the polarity as well. This research paper investigates the sentiment analysis methods and techniques that can be used on the YouTube content. Additionally, it explains and categorizes these approaches which are useful in researches in data mining and sentiment analysis.

TITLE NAME: Emotion Classification on Youtube Comments using Word Embedding
AUTHOR: Julio Savigny, Ayu Purwarianti
YEAR: 2020
ABSTRACT: Youtube is one of the most popular video sharing platform in Indonesia. A person can react to a video by commenting on the video. A comment may contain an emotion that can be identified automatically. In this study, we conducted experiments on emotion classification on Indonesian Youtube comments. A corpus containing 8,115 Youtube comments is collected and manually labelled using 6 basic emotion label (happy, sad, angry, surprised, disgust, fear) and one neutral label. Word embedding is a popular technique in NLP, and have been used in many classification tasks. Word embedding is a representation of a word, not a document, and there are many methods to use word embedding in a text classification task. Here, we compared many methods for using word embedding in a classification task, namely average word vector, average word vector with TF-IDF, paragraph vector, and by using Convolutional Neural Network (CNN) algorithm. We also study the effect of the parameters used to train the word embedding. We compare the performance of the classification with a baseline, which was previously state-of-the-art, SVM with Unigram TFIDF. The experiments showed that the best performance is achieved by using word embedding with CNN method with accuracy of 76.2%, which is an improvement from the baseline.

TITLE NAME: A Comparative Analysis of Common YouTube Comment Spam Filtering Techniques
AUTHOR: Abdullah O. Abdullah, Mashhood A. Ali, Murat Karabatak, Abdulkadir Sengur
YEAR: 2020
ABSTRACT: Ever since its development in 2005, YouTube has been providing a vital social media platform for video sharing.

Unfortunately, YouTube users may have malicious intentions, such as disseminating malware and profanity. One way to do so is using the comment field for this purpose. Although YouTube provides a built-in tool for spam control, yet it is insufficient for combating malicious and spam contents within the comments. In this paper, a comparative study of the common filtering techniques used for YouTube comment spam is conducted. The study deploys datasets extracted from YouTube using its Data API. According to the obtained results, high filtering accuracy (more than 98%) can be achieved with low-complexity algorithms, implying the possibility of developing a suitable browser extension to alleviate comment spam on YouTube in future

TITLE NAME: SpamHD: Memory-Efficient Text Spam Detection using Brain-Inspired Hyperdimensional Computing

AUTHOR: Rahul Thapa; Bikal Lamichhane; Dongning Ma; Xun Jiao

YEAR:

ABSTRACT:Brain-inspired hyperdimensional Computing (HDC) leverages the mathematical properties of high-dimensional vectors (hypervectors) which show remarkable agreement with how brain functions. Hypervectors (HVs) are high-dimensional (e.g., 10,000 dimensions), holographic, and (pseudo)random with independent and identically distributed (i.i.d) components. Recently, HDC has demonstrated promising capability in a wide range of applications such as robotics, bio-medical signal processing, and genome sequencing. Text spam detection is a classic natural language processing (NLP) task that is usually solved using machine learning methods associated with data preprocessing techniques such as tokenization. In this paper, we develop a memory-efficient text spam detection approach called SpamHD based on HDC methods. In addition to the conventional tokenization-based approach, we also develop a tokenization-free HDC approach with N-gram encoding. Experimental results on three real-world spam datasets (Hotel review, SMS text, and YouTube comments) show that SpamHD is able achieve similar or even outperform baseline tokenization-based learning methods, but with significantly less storage requirements (30X-115X model size reduction). Further, we perform a design space exploration for SpamHD by tuning the number of dimensions of HVs and encoding methods, and evaluate the impact of such design parameters on accuracy and memory requirements.

III.SYSTEM ANALYSIS

EXISTING SYSTEM:

YouTube has its own spam filtering system, though there are still spam comments that are not being caught. In this paper, we review related studies on YouTube spam comments and propose the Cascaded Ensemble Machine Learning Model aware YouTube Spam Comments Detection Scheme to improve the performance of the model. In previous studies, various machine learning techniques were applied to each dataset to detect spam comments and compare their performance. Therefore, in this paper, we propose an ensemble machine learning method that combines the results of several models to produce the final result.

DISADVANTAGE:

- The user evaluation can collect desired information such as usage problems or fitness of recommendation to users who have been invited for evaluation.
- This is a method for determining whether the prediction result is reliable in the case of the data scarcity problem.

PROPOSED SYSTEM:

In this paper on spam or normal labeled datasets. The dataset consists of 6,431,471 crawled comments of which 481,334 comments were spam in the 6,407 videos that were most viewed between October 31, 2011 and January 17, 2012 in the United States. This dataset was mixed with English and non-English comments, so we extracted only English comments for the experiment. In addition, to make it similar to the data size used in the experiment of 3, we extracted 1,000 spam comments and normal comments, and compared them with 5,000 samples. In the experiment, we used an ANN (Artificial Neural Network) technique with the techniques used in 3. Finally, we plotted the Precision, Recall, F1-score, and ROC curves by adding 1,000 data points from 1,000 to 5,000.

ADVANTAGE:

- The advantages are that it is very efficient in terms of storage space and computation time and handles noise and missing data well.
- Offline evaluation has the advantage of standardizing evaluation methods and evaluation items, and there are various evaluation items such as accuracy, coverage, confidence, and novelty.

IV.SYSTEM REQUIREMENTS

ISSN (Online): 2456-0448
International Journal of Innovative Research in Management, Engineering, and Technology
Vol. 7, Issue 3, March 2022

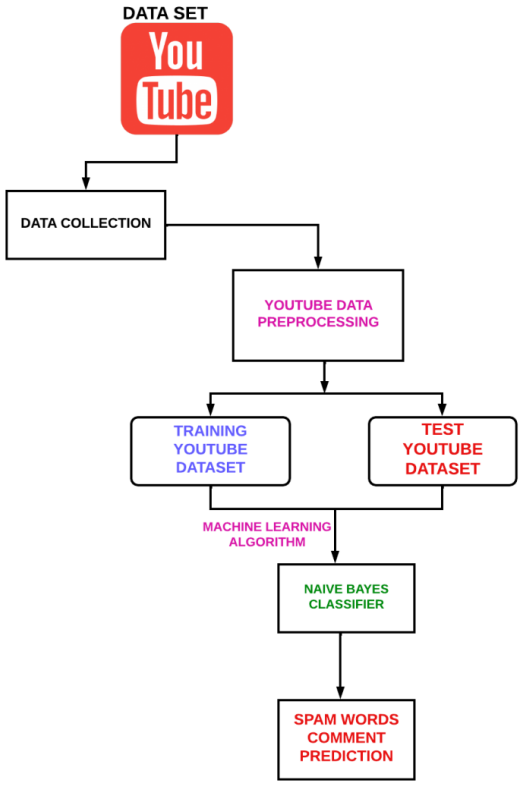
H/W SYSTEM CONFIGURATION

- Processor - I3, I5,I7
- Speed - 2.1 GHz
- RAM - 4 Gb
- Hard Disk - 260 GB

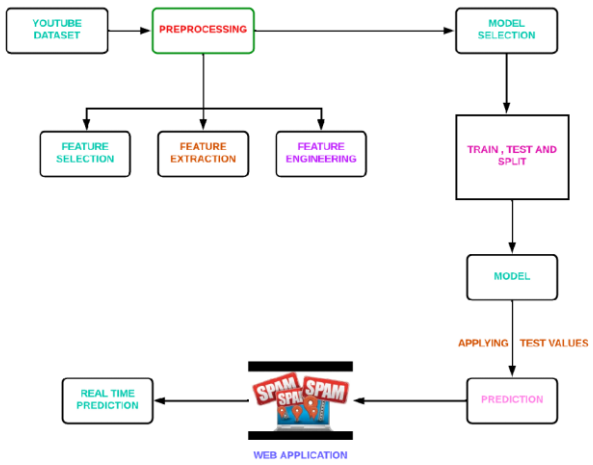
S/W SYSTEM CONFIGURATION

- Operating System - Windows 7/8/10
- Front End - Html,Css
- Scripts - python language
- Tool - python3 IDE

ARCHITECTURE DIAGRAM:



ISSN (Online): 2456-0448
International Journal of Innovative Research in Management, Engineering, and Technology
Vol. 7, Issue 3, March 2022



V.MODULES NAME:

- YOUTUBE DATASET.
- PREPROCESSING
- FEATURE SELECTION
- FEATURE EXTRACTION AND FEATURE ENGINEERING
- NAÏVE BAYES CLASSIFIER

MODULES DESCRIPTION:

Youtube Dataset

The benefit of using these words based on their entropy score in the characteristic-set is that we have been capable of lessen uncertainty in the prediction final results as those phrases have an exceptional effect of frequency count in spam and non-spam YouTube.

PREPROCESSING

Before starting with preparation preprocessing of the messages must be done. First all the characters must be in lowercase. The word which is both in uppercase and lowercase must be considered as same words and not as two different words. Then tokenization must be done for each message in the data set.

FEATURE SELECTION

The main advantage of using the words present in the dataset is that it is capable of reducing uncertainty in the prediction of the final results as those phrases have a remarkable effect of frequency count in spam and ham comments in YouTube.

FEATURE EXTRACTION AND FEATURE ENGINEERING

- Attribute significance is a supervised characteristic that ranks attributes in a step by step manner with their significance in predicting an aim. Here Count Vectorizer is used which convert a “collection of text documents to a matrix of token counts . This undergoes the following technique:
- N-grams: N-grams is used to improve the accuracy. It is dealt with single word but when there are two mutual words the complete meaning will be changed. So, the variation of accuracy is better occurred when text is split into token of two or more words rather than being a single word.
- Analyzer: “Whether the feature should be made of word or character n-grams. Option ‘char_wb’ creates character n-grams only from text inside word boundaries; n-grams at the edges of words are padded with space.”

NAÏVE BAYES CLASSIFIER

Naive Bayes classifier is based on Bayes’ theorem from which it gets its name. It is a simple to understand probabilistic model which gives really quick predictions Naive Bayes work on dependent events and the probability of an event occurring in the future that can be detected from the previous occurring of the same event. This technique can be used to classify spam you tube words probabilities play the main rule here. If some words occur often in spam but not in ham, then this incoming youtube is probably spammed. Naive Bayes classifier technique has become a very popular method in spam words filter. Every word has certain probability of occurring in spam or ham in its database. If the total of words probabilities exceeds a certain limit, the filter will mark the dataset to either category So in the Training Stage Naive Bayes create a Lookup table in which they store all the possibility of probability which we are going to use in the Algorithm for predicting the result. And In the testing phase let suppose you have given a test point to the algorithm to predict the result, they fetch the values from the lookup table in which they store all the possibility of probability and use that value to predict the result.

ISSN (Online): 2456-0448
International Journal of Innovative Research in Management, Engineering, and Technology
Vol. 7, Issue 3, March 2022

VI.SOFTWARE TESTING

6.1 UNIT TESTING

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

6.2 INTEGRATION TESTING

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

6.3 FUNCTIONAL TEST

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

Functional testing is centered on the following items:

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

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

6.4 SYSTEM TEST

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

6.5 WHITE BOX TESTING

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

6.5 BLACK BOX TESTING

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

2.3 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

2.3.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

1. 2.3.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null

changes are required for implementing this system.

2.3.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

VII.SOURCE CODE:

TEST.PY

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB

df= pd.read_csv("YoutubeSpamMergedData.csv")
df_data = df[["CONTENT","CLASS"]]
# Features and Labels
df_x = df_data['CONTENT']
df_y = df_data.CLASS
# Extract Feature With CountVectorizer
corpus = df_x
cv = CountVectorizer()
X = cv.fit_transform(corpus) # Fit the Data
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, df_y, test_size=0.33, random_state=42)
#Naive Bayes Classifier
from sklearn.naive_bayes import MultinomialNB
clf = MultinomialNB()
clf.fit(X_train,y_train)
score = clf.score(X_test,y_test)
print(score)
```

APP.PY

```
from flask import Flask,render_template,url_for,request
import pandas as pd
import pickle
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.externals import joblib

app = Flask(__name__)

@app.route('/')
def home():
        return render_template('home.html')

@app.route('/predict',methods=['POST'])
def predict():
        df= pd.read_csv("YoutubeSpamMergedData.csv")
        df_data = df[["CONTENT","CLASS"]]
        # Features and Labels
        df_x = df_data['CONTENT']
        df_y = df_data.CLASS
  # Extract Feature With CountVectorizer
        corpus = df_x
```

```
        cv = CountVectorizer()
        X = cv.fit_transform(corpus) # Fit the Data
        from sklearn.model_selection import train_test_split
        X_train, X_test, y_train, y_test = train_test_split(X, df_y, test_size=0.33, random_state=42)
        #Naive Bayes Classifier
        from sklearn.naive_bayes import MultinomialNB
        clf = MultinomialNB()
        clf.fit(X_train,y_train)
        clf.score(X_test,y_test)
        #Alternative Usage of Saved Model
        # ytb_model = open("naivebayes_spam_model.pkl","rb")
        # clf = joblib.load(ytb_model)

        if request.method == 'POST':
                comment = request.form['comment']
                data = [comment]
                vect = cv.transform(data).toarray()
                my_prediction = clf.predict(vect)
        return render_template('result.html',prediction = my_prediction)


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

HOME.HTML

```
<!DOCTYPE html>
<html>
<head>
        <title>Home</title>
        <!-- <link rel="stylesheet" type="text/css" href="../static/css/styles.css"> -->
        <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/styles.css') }}">
  <link      rel="stylesheet"      href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"      integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
</head>
<body>

        <header>
                <div class="container">
                <div id="brandname">
                        ML App
                </div>
                <h2>Spam Detection For Youtube Comments</h2>

        </div>
        </header>

        <div class="ml-container">

                <form action="{{ url_for('predict')}}" method="POST">
                <p>Enter Your Comment Here</p>
                <!-- <input type="text" name="comment"/> -->
                <textarea name="comment" rows="4" cols="50"></textarea>
                <br/>
```

```
                <input type="submit" class="btn-info" value="predict">

        </form>

        </div>



</body>
</html>
```

RESULT.HTML:

```
<!DOCTYPE html>
<html>
<head>
        <title></title>
   <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/styles.css') }}">
   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
</head>
<body>
        <header>
                <div class="container">
                <div id="brandname">
                        ML App
                </div>
                <h2>Spam Detection For Youtube Comments</h2>
        </div>
        </header>
        <p style="color:blue;font-size:20;text-align: center;"><b>Results for Comment with a Accuracy of 91.95%</b></p>
        <div class="results">
        {% if prediction == 1%}
        <h2 style="color:red;">Spam</h2>
        {% elif prediction == 0%}
        <h2 style="color:blue;">Not a Spam (It's a Ham)</h2>
        {% endif %}
        </div>
</body>
</html>
```

VIII.CONCLUSION:

In this paper, we proposed a technique to detect spam comments on YouTube, which have recently seen tremendous growth using a Cascaded Ensemble Machine Learning Model. It examined related studies on YouTube spam comment screening and conducted classification experiments with six different machine learning techniques (Decision tree, Logistic regression, Bernoulli Naïve Bayes, Random Forest, Support vector machine with linear kernel, Support vector machine with Gaussian kernel) and two ensemble models (Ensemble with hard voting, Ensemble with soft voting) combining these techniques in the comment data. The experimental results showed that the ESM-S model proposed in this paper had the best performance in four of five evaluation measures. We proposed a new model, combining various techniques that improved the performance results unlike previous studies that used one model for detection.

IX.References

[1] S. Aiyar and N. P. Shetty, ‘‘N-gram assisted Youtube spam comment detection,’’ *Proc. Comput. Sci.*, vol. 132, pp. 174–182, Jan. 2018, doi: 10.1016/j.procs.2018.05.181.

[2] A. Kantchelian, J. Ma, L. Huang, S. Afroz, A. Joseph, and J. D. Tygar, ''Robust detection of comment spam using entropy rate,'' in *Proc. 5th ACM Workshop Secur. Artif. Intell. (AISec)*, 2012, pp. 59–70, doi: 10.1145/2381896.2381907.

[3] A. Madden, I. Ruthven, and D. Mcmenemy, ''A classification scheme for content analyses of Youtube video comments,'' *J. Documentation*, vol. 69, no. 5, pp. 693–714, Sep. 2013, doi: 10.1108/JD-06-2012-0078.

[4] A. Severyn, A. Moschitti, O. Uryupina, B. Plank, and K. Filippova, ''Opin ion mining on Youtube,'' in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2014, pp. 1–10, doi: 10.3115/v1/P14- 1118.

[5] M. Z. Asghar, S. Ahmad, A. Marwat, and F. M. Kundi, ''Sentiment analysis on Youtube: A brief survey,'' 2015, *arXiv:1511.09142*. [Online]. Available: http://arxiv.org/abs/1511.09142

[6] T. C. Alberto, J. V. Lochter, and T. A. Almeida, ''TubeSpam: Comment spam filtering on Youtube,'' in *Proc. IEEE 14th Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2015, pp. 138–143, doi: 10.1109/ICMLA.2015.37.

[7] A. U. R. Khan, M. Khan, and M. B. Khan, ''Naïve multi-label classification of Youtube comments using comparative opinion mining,'' *Proc. Comput. Sci.*, vol. 82, pp. 57–64, Jan. 2016, doi: 10.1016/j.procs.2016.04.009.

[8] J. Savigny and A. Purwarianti, ''Emotion classification on Youtube comments using word embedding,'' in *Proc. Int. Conf. Adv. Infor mat., Concepts, Theory, Appl. (ICAICTA)*, Aug. 2017, pp. 1–5, doi: 10.1109/ICAICTA.2017.8090986.

[9] S. Sharmin and Z. Zaman, ''Spam detection in social media employing machine learning tool for text mining,'' in *Proc. 13th Int. Conf. Signal Image Technol. Internet-Based Syst. (SITIS)*, Dec. 2017, pp. 137–142, doi: 10.1109/SITIS.2017.32.

[10] A. O. Abdullah, M. A. Ali, M. Karabatak, and A. Sengur, ''A comparative analysis of common Youtube comment spam filtering techniques,'' in *Proc. 6th Int. Symp. Digit. Forensic Secur. (ISDFS)*, Mar. 2018, pp. 1–5, doi: 10.1109/ISDFS.2018.8355315.

[11] E. Poche, N. Jha, G. Williams, J. Staten, M. Vesper, and A. Mahmoud, ''Analyzing user comments on Youtube coding tutorial videos,'' in *Proc. IEEE/ACM 25th Int. Conf. Program Comprehension (ICPC)*, May 2017, pp. 196–206, doi: 10.1109/ICPC.2017.26.

[12] A. Aziz, C. F. M. Foozy, P. Shamala, and Z. Suradi, ''Youtube spam comment detection using support vector machine and k-nearest neighbor,'' Tech. Rep., 2018, doi: 10.11591/ijeecs.v12.i2.pp607-611.

[13] R. K. Das, S. S. Dash, K. Das, and M. Panda, ''Detection of spam in Youtube comments using different classifiers,'' in *Advanced Computing and Intelligent Engineering*, 2020, pp. 201–214, doi: 10.1007%2F978- 981-15-1081-6_17.

[14] N. M. Samsudin, C. F. B. Mohd Foozy, N. Alias, P. Shamala, N. F. Othman, and W. I. S. Wan Din, ''Youtube spam detection framework using Naïve Bayes and logistic regression,'' *Indonesian J. Electr. Eng. Comput. Sci.*, vol. 14, no. 3, p. 1508, Jun. 2019, doi: 10.11591/ijeecs.v14.i3.pp1508- 1517.

[15] G. Kaur, A. Kaushik, and S. Sharma, ''Cooking is creating emotion: A study on hinglish sentiments of Youtube cookery channels using semi supervised approach,'' *Big Data Cognit. Comput.*, vol. 3, no. 3, p. 37, Jul. 2019, doi: 10.3390/bdcc3030037.

[16] E. Ezpeleta, M. Iturbe, I. Garitano, I. V. de Mendizabal, and U. Zurutuza, ''A mood analysis on Youtube comments and a methodfor improved social spam detection,'' in *Proc. HAIS*, 2018, pp. 514–525, doi: 10.1007/978-3- 319-92639-1_43.

[17] N. Hussain, H. Turab Mirza, G. Rasool, I. Hussain, and M. Kaleem, ''Predilection decoded: Spam review detection techniques: A system atic literature review,'' *Appl. Sci.*, vol. 9, no. 5, p. 987, Mar. 2019, doi: 10.3390/app9050987.

[18] L. Song, R. Y. K. Lau, R. C.-W. Kwok, K. Mirkovski, and W. Dou, ''Who are the spoilers in social media marketing? Incremental learning of latent semantics for social spam detection,'' *Electron. Commerce Res.*, vol. 17, no. 1, pp. 51–81, Mar. 2017, doi: 10.1007/s10660-016-9244-5.

[19] S. Jain and D. M. Patel. (2019). *Analyzing User Comments of Learn ing Videos From Youtube Using Machine Learning*. [Online]. Available: http://www.ijirset.com/upload/2019/august/50_Analyzing_DJ.PDF

[20] P. Bansal. (2019). *Detection of Offensive Youtube Comments, a Perfor mance Comparison of Deep Learning Approaches*. [Online]. Available: https://core.ac.uk/reader/301313034

[21] G. Shi, F. Luo, Y. Tang, and Y. Li, ‘‘Dimensionality reduction of hyper spectral image based on local constrained manifold structure collaborative preserving embedding,’’ *Remote Sens.*, vol. 13, no. 7, p. 1363, Apr. 2021, doi: 10.3390/rs13071363.

[22] W. Li and Q. Du, ‘‘Laplacian regularized collaborative graph for dis criminant analysis of hyperspectral imagery,’’ *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7066–7076, Dec. 2016, doi: 10.1109/TGRS.2016.2594848.