

Tech Saksham

**CapstoneProjectReport**

**“Detecting Spam Emails”**

**“GOVERNMENTCOLLEGEOF ENGINEERING, SALEM”**

|  |  |
| --- | --- |
| **NMID** | **NAME** |
| au61772111103 | THAMIZHAN SEN |

|  |
| --- |
| RAMAR BOSE |
| Sr.AIMaster Trainer |

**ABSTRACT**

Emailspamcontinuestobeapervasiveissue,causinginconvenienceand potential harm to users. In this project, we propose a machine learning approach to automatically detect spam emails. The project involves collecting a dataset of labeled emails, where each email is classified as spam or non-spam (ham).

We preprocess the email text data to convert it into a suitable format for machine learning models. Feature extraction techniques such as TF-IDF (TermFrequency-InverseDocumentFrequency)andN-gramsareusedto extract relevant features from the text data.

Wethentrainamachinelearningmodel,suchasaNaiveBayesclassifieror a Support Vector Machine (SVM), on the extracted features. The trained model is evaluated using metrics such as accuracy, precision, recall, and F1 score to assess its performance in detecting spam emails.

# INDEX

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Table of Contents** | **PageNo.** |
| 1 | Chapter1:Introduction | 1 |
| 2 | Chapter2:ServicesandToolsRequired | 10 |
| 3 | Chapter3:ProjectArchitecture | 13 |
| 4 | Chapter4: ModelingandProjectOutcome | 19 |
| 5 | Conclusion | 24 |
| 6 | FutureScope | 25 |
| 7 | References | 26 |
| 8 | Links | 27 |

**CHAPTER 1 INTRODUCTION**

* 1. **ProblemStatement**
     + The problem of spam emails has been a persistent issue since the earlydaysoftheinternet.Spamemailsareunsolicitedmessagessent in bulk to a large number of recipients, often with commercial or malicious intent. These emails can be annoying, cluttering up users' inboxes with irrelevant messages, but they can also pose serious threats to security and privacy.
     + Spam emails often contain malicious content, such as phishing links ormalware,thatcantrickusersintorevealingsensitiveinformationor infect their devices. Phishing emails, for example, are designed to look like legitimate messages from reputable sources, such as banks or online services, in an attempt to deceive users into providing their personal or financial information.
     + Toaddresstheproblemofspamemails,varioussolutionshavebeen developed over the years. One common approach is to use spam filters, which are algorithms designed to automatically identify and filter out spam emails from users' inboxes. These filters work by analyzing the content and characteristics of emails to determine whether they are likely to be spam or legitimate messages.
     + Machine learning has emerged as a powerful tool for spam email detection,asitallowsforthedevelopmentofsophisticatedalgorithms that can analyze large amounts of email data to identify patterns and trends associated with spam emails. By training machine learning models on labeled datasets of spam and non-spam emails, these algorithms can learn to accurately classify new emails as spam or

non-spambasedontheircontentand characteristics.

* + - The goal of this project is to develop a machine learning-based solution for detecting spam emails and filtering them out from users' inboxes.Byreducingtheamountofunwantedandpotentiallyharmful emails that users receive, the project aims to improve the overall email experience for users and enhance their security and privacy.
  1. **ProposedSolution**
     + Our proposed solution for detecting spam emails involves using machinelearningalgorithms,apopularapproachduetoitsabilityto analyze large amounts of data and identify patterns that may not be apparent to human reviewers. Here's a detailed explanation of each component:
     + Data Collection:
     + Gathering a largeand diverse dataset of labeled emailsis crucial for trainingamachinelearningmodeltoaccuratelydistinguishbetween spam and non-spam emails.
     + Thedatasetshouldideallycontainemailsfromvarioussourcesandin different languages to ensure the model's effectiveness across different contexts.
     + Data Preprocessing:
     + Preprocessingtheemailtextinvolvesseveralstepstocleanand prepare the data for analysis.
     + This includes removing any HTML tags, special characters, and punctuationmarksthatdonotcontributetothecontentoftheemail.
     + Stopwords(commonwordslike"and","the","is")arealsoremoved as they do not carry significant meaning in distinguishing between spam and non-spam emails.
     + Thetextisthentokenized,whichmeanssplittingitintoindividual words or tokens, to prepare it for feature extraction.
     + FeatureExtraction:
     + Featureextractionisacriticalstepinconvertingtheemailtextintoa format that can be used by machine learning algorithms.
     + CommontechniquesincludeTF-IDF,whichcalculatestheimportance of a word in an email relative to its frequency in a collection of emails, and word embeddings, which represent words as densevectors in a continuous space.
     + N-grams,whicharesequencesofNwords,canalsobeusedto capture contextual information in the email text.
     + MachineLearningModel:
     + Oncetheemailtexthasbeenpreprocessedandfeaturesextracted,itis ready to be used as input to a machine learning model.
     + Commonmodelsforspamemaildetectionincludelogisticregression, support vector machines (SVM), and naive Bayes classifiers.
     + Thesemodelsaretrainedonthelabeleddatasetofemailstolearnthe patterns and characteristics that distinguish spam from non-spam emails.
     + ModelEvaluationandTesting:
     + Aftertrainingthemachinelearningmodel,itisevaluatedusinga separate test dataset to assess its performance.
     + Metrics such as accuracy, precision, recall, and F1 score are commonlyusedtoevaluatethemodel'seffectivenessincorrectly classifying spam and non-spam emails.
     + By following these steps, the proposed solution aims to develop a robustandaccuratespamemaildetectionsystemthatcaneffectively filter out unwanted emails, thereby improving the overall email experience for users.
  2. **Feature**
     + MachineLearningAlgorithms:
     + Logistic Regression: A simple yet effective algorithm for binary classification,whereitmodelstheprobabilityofanemailbeingspamor non-spam based on the input features.
     + Support Vector Machines (SVM): A powerful algorithm for binary classificationthatfindsthehyperplanethatbestseparatesspamandnon- spam emails in the feature space.
     + NaiveBayesClassifiers:Afamilyofprobabilisticclassifiersbasedon Bayes' theorem, where it assumes that the presence of a particular feature in an email is independent of the presence of any other feature.
     + Data Preprocessing:
     + Removing Irrelevant Information: This includes removing HTML tags,specialcharacters,andpunctuationmarksthatdonotcontributeto the content of the email.
     + Tokenization:Splittingthetextintoindividualwordsortokensto prepare it for further analysis.
     + Stopword Removal: Removing common words (e.g., "and", "the") thatoccurfrequentlybutdonotcarrymuchmeaningindistinguishing between spam and non-spam emails.
     + FeatureExtraction:
     + WordFrequency:Countingthenumberoftimeseachwordappearsin an email, which can help capture the importance of certain words in determining whether an email is spam or non-spam.
     + TF-IDF(TermFrequency-InverseDocumentFrequency):Ameasure that reflects how important a word is to a document in a collection or corpus, which helps in identifying key words that are more likely to be associated with spam emails.
     + N-grams: Sequences of N words, which can capture contextual informationandhelpidentifypatternsinthetextthatareindicativeof spam or non-spam emails.
     + Model Training:
     + Splitting the Dataset: The dataset is typically split into training, validation,andtestsets,wherethetrainingsetisusedtotrainthemodel,

thevalidationsetisusedtotunehyperparameters,andthetestsetis used to evaluate the model's performance.

* + - Training the Model: The machine learning model is trained on the trainingsetusingtheselectedalgorithmandfeaturesextractedfromthe email text data.
    - Model Evaluation:
    - Metrics: Various metrics are used to evaluate the model's performance, including accuracy (the proportion of correctly classified emails), precision (the proportion of correctly classified spam emails among all emails classified as spam), recall (the proportion of correctly classifiedspamemailsamongallactualspamemails),andF1score(the harmonic mean of precision and recall).
    - Cross-Validation: Cross-validation techniques, such as k-fold cross- validation,areusedtoensurethatthemodel'sperformanceisconsistent across different subsets of the data and to reduce the risk of overfitting.
    - Thesecomponentsworktogethertocreatearobustandaccuratespam email detection system that can effectively classify emails as spam or non-spam based on their content and characteristics.
  1. **Advantages**
     + Improved Email Security: By effectively filtering out spam emails, the project enhances email security by reducing the risk of users clickingonmaliciouslinksordownloadingharmfulattachmentsthat may compromise their personal information or devices.
     + Enhanced User Experience: Filtering out spam emails improves the overall email experience for users by reducing the clutter in their inboxandensuringthattheyonlyseerelevantandlegitimateemails. This can lead to increased productivity and efficiency in managing emails.
     + Time and Resource Savings: By automatically filtering out spam emails, the project saves users time and resources that would otherwisebespentmanuallysortingthroughanddeletingunwanted emails. This can be particularly beneficial for individuals and organizations receiving a large volume of emails daily.
     + Protection Against Phishing Attacks: Spam emails often contain phishing attempts to steal sensitive information such as login credentialsorfinancialinformation.Byfilteringouttheseemails,the project
     + protectsusersfromfallingvictimtophishingattacksandpotential identity theft.
     + Scalability: The machine learning-based approach used in the project is scalable and can be applied to large volumes of emails, making it suitable for use in email services with a large user base. This ensures thatthespamfilteringsystemremainseffectiveevenasthevolumeof emails increases.
     + Adaptability to New Spamming Techniques: The project's machine learningmodelscanbetrainedandupdatedregularlytoadapttonew spamming techniques usedby spammers. This ensures that the spam filtering system remains effective in detecting and filtering out new types of spam emails.
     + Cost-EffectiveSolution:Implementingaspamemaildetectionsystem based on machine learning is a cost-effective solution compared to manual spam filtering or using third-party spam filtering services. It reduces the need for human intervention in managing spam emails, leading to cost savings for individuals and organizations.
     + Customization: The project allows for customization of the spam filteringrulesandcriteriabasedonthespecificneedsandpreferences of users or organizations. This flexibility ensures that the spam filtering system can be tailored to meet the unique requirements of different users or organizations.
     + .
  2. **Scope**
     + Literature Review: Conduct a comprehensive review of existing research and techniques for spam email detection. This includes studying variousmachinelearning approaches,such assupervised learning algorithms (e.g., logistic regression, SVM, naive Bayes), unsupervised learning algorithms (e.g., clustering), and deep learningmodels(e.g.,neuralnetworks).Additionally,reviewrule- basedmethodsthatusepredefinedrulestoclassifyemailsasspam

or non-spam, as well as hybrid models that combine machine learningandrule-basedapproachesforimprovedperformance.

* + - Data Collection: Gather a dataset of labeled emails, including examples of spam and non-spam emails, to train and test the machinelearningmodels.Thedatasetshouldberepresentativeof the types of emails users typically receive and should be large enough to ensure the models are trained effectively.
    - DataPreprocessing:Preprocesstheemailtextdatatoremovenoise, such as HTML tags, special characters, and stopwords. Tokenize the text to convert it into individual words or tokens, and convert it intoaformatsuitableformachinelearningmodels,suchasTF-IDF vectors or word embeddings.
    - ModelEvaluation:Evaluatethetrainedmodelsusingmetricssuch as accuracy, precision, recall, and F1 score to assess their performance in detecting spam emails. These metrics provide insights into how well the models are able to correctly classify spam and non-spam emails.
    - System Implementation: Implement the spam email detection system,includingintegratingitwithexistingemailservices(e.g., Gmail, Outlook) to automatically filter spam emails. Develop a user-friendly interface for users to interact with the system, allowing them to view and manage spam emails effectively.
    - TestingandValidation:Testthesystemwithaseparatedatasetof emails to validate its performance and ensure that it meets the requirements.

# CHAPTER2

**SERVICESANDTOOLSREQUIRED**

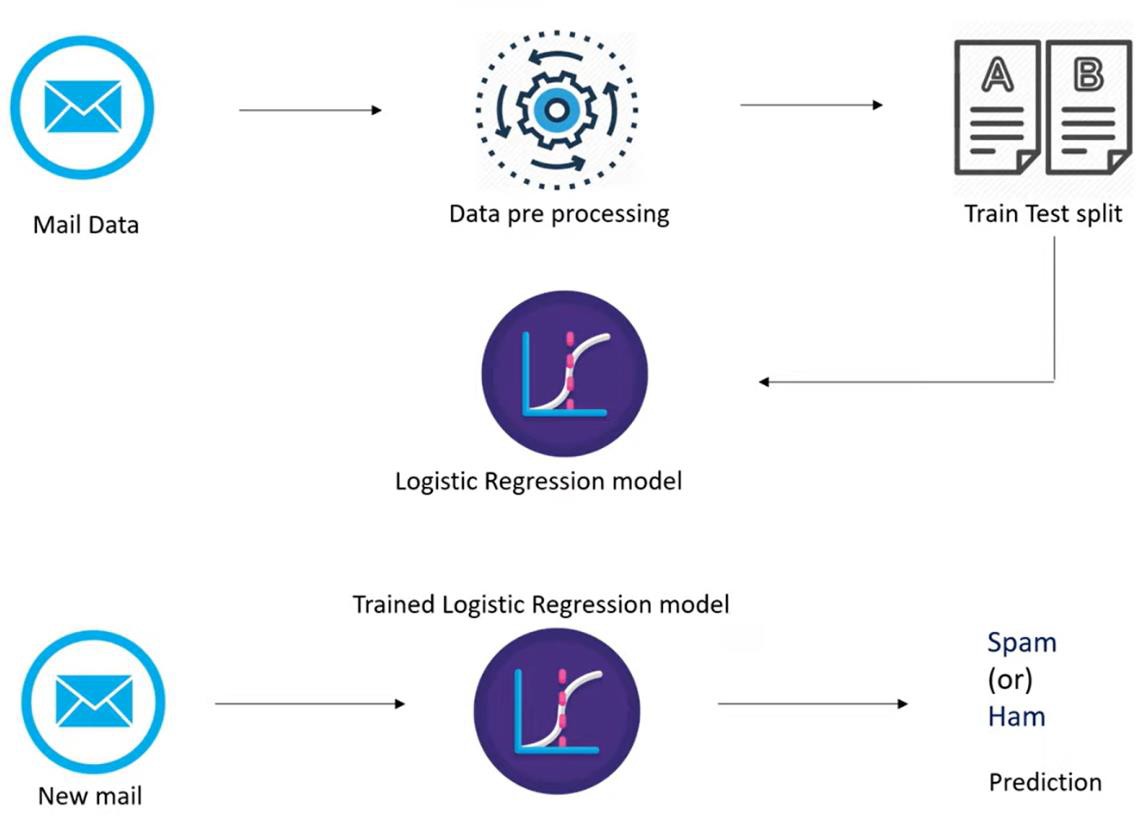
**2.1LR-Exiting Models**

* 1. **Required–Systemconfig|Cloud computing**
  2. **ServicesUsed**
     + Machine Learning Libraries: Utilize machine learning libraries such assci-kit-learn,TensorFlow,orPyTorchfordevelopingandtraining machine learning models for spam email detection.
     + Email Dataset: Use publicly available email datasets or collect a customdatasetoflabeledemailsfortrainingandtestingthemachine learning models.
     + Data Preprocessing Tools: Use tools such as NLTK (Natural LanguageToolkit)orspaCyforpreprocessingtheemailtextdata, including tokenization, stopword removal, and stemming.
     + FeatureExtractionLibraries: Utilizelibrariessuchasscikit-learnor TensorFlowforextractingfeaturesfromtheemailtextdata,suchas TF-IDF (Term Frequency-Inverse Document Frequency) and N- grams.
     + Model Training and Evaluation Platforms: Use platforms such as Google Colab or Amazon SageMaker for training and evaluating machinelearningmodels,includinglogisticregression,supportvector machines (SVM), and naive Bayes classifiers.
     + EmailServicesIntegration:Integratethespamemaildetectionsystem with existing email services such as Gmail or Outlook using APIs or SDKs.
     + UserInterfaceDevelopmentTools:Usefront-enddevelopmenttools such as HTML, CSS, and JavaScript for developing a user-friendly interface for the spam detection system.
     + Documentation and Reporting Tools: Use tools such as Microsoft Word or LaTeX for documenting the project, including the problem statement,methodology,results,andconclusions,inacomprehensive project report.
  3. **ToolsandSoftware used**
     + ProgrammingLanguages:Pythonforimplementingmachinelearning algorithms and data preprocessing tasks.
     + MachineLearningLibraries:Scikit-learn,TensorFlow,orPyTorch for developing and training machine learning models.
     + DataPreprocessingTools:NLTK(NaturalLanguageToolkit)or spaCy for preprocessing the email text data.
     + FeatureExtractionLibraries:Scikit-learnorTensorFlowfor extracting features from the email text data.
     + DevelopmentEnvironment:JupyterNotebookorGoogleColabfor developing and testing code.
     + VersionControl: Gitforversioncontrolandcollaboration.
     + EmailDataset:Publiclyavailableemaildatasetsorcustomdatasets collected for training and testing.
     + Email Services Integration: APIs or SDKs for integrating the spam emaildetectionsystemwithemailservicessuchasGmailorOutlook.
     + User Interface Development: HTML, CSS, and JavaScript for developingauser-friendlyinterfaceforthespamdetectionsystem.
     + Documentation and Reporting: Microsoft Word or LaTeX for documenting the project, including the problem statement, methodology,results,andconclusions,inacomprehensiveproject report.

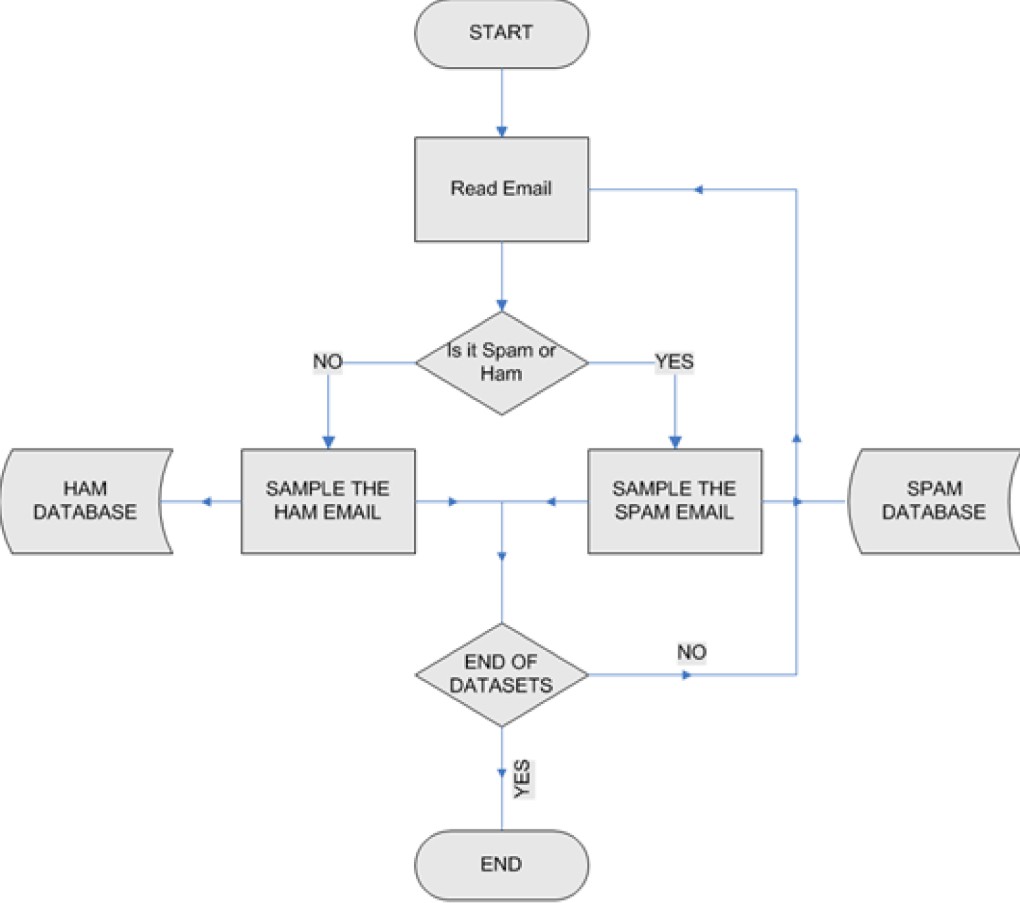
# CHAPTER 3 PROJECTARCHITECTURE

**3.1Architecture**

1. **Systemflowdiagram**



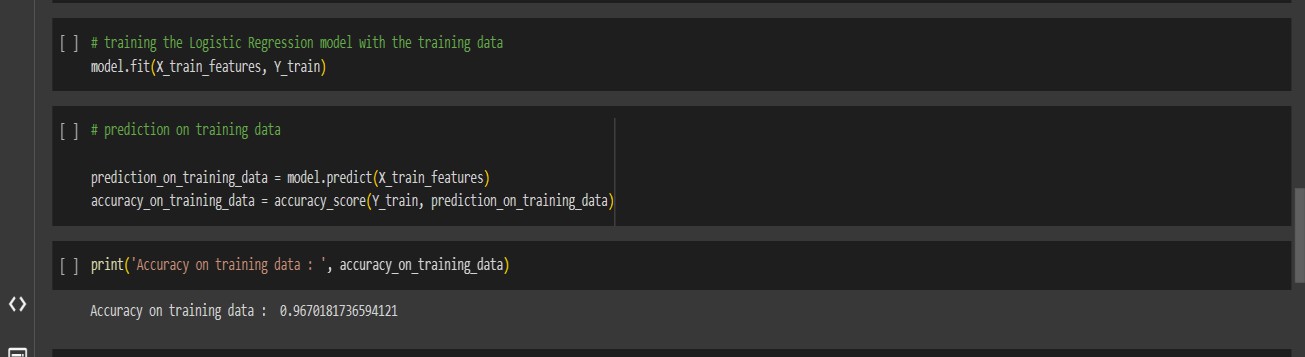
1. **Dataflowdiagram**



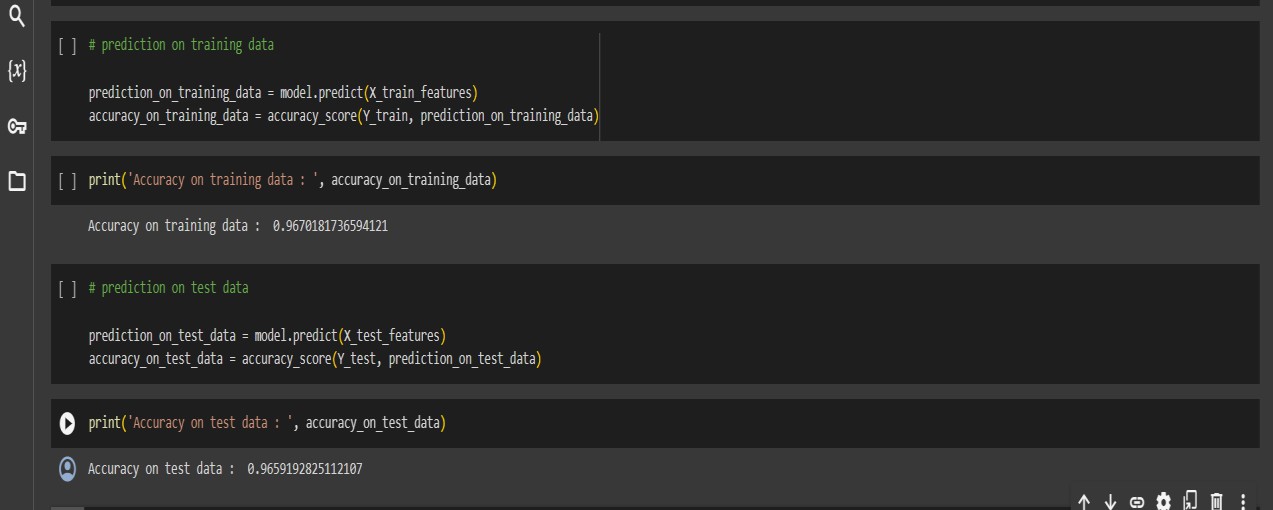
## Modules

* + Data Collection Module: Responsible for collecting a dataset of labeledemails,includingspamandnon-spamemails,fortrainingand testing the machine learning models.
  + DataPreprocessingModule:Handlespreprocessingtaskssuchas removing noise, tokenization, and converting text into a format suitable for machine learning models.
  + FeatureExtractionModule:Extractsrelevantfeaturesfromtheemail text data, such as word frequency, TF-IDF, and N-grams, to use as input to the machine learning models.
  + Machine Learning Model Module: Implements machine learning algorithms such as logistic regression, support vector machines (SVM),ornaiveBayesclassifiersforclassifyingemailsasspamor non-spam.
  + Model Training and Evaluation Module: Handles the training of machine learning models using the collected dataset and evaluates theirperformanceusingmetricssuchasaccuracy,precision,recall, and F1 score.
  + EmailIntegrationModule:Integratesthespamemaildetectionsystem with existing email services using APIs or SDKs for real-time detection and filtering of spam emails.
  + UserInterfaceModule:Developsauser-friendlyinterfaceforusersto interact with the spam detection system, allowing them to view and manage spam emails effectively.
  + DocumentationandReportingModule:Documentstheentireproject, including the problem statement, methodology, results, and conclusions, in a comprehensive project report.

## Trainingmodeldiagram



* + **Predictingthemodel’s diagram**



Here’sahigh-levelarchitecturefortheproject:

* + - Data Collection:

Collectadatasetoflabeledemails,includingspamandnon-spam emails, for training and testing.

* + - Data Preprocessing:

Preprocesstheemailtextdatatoremovenoise,tokenizethetext,and convert it into a suitable format for machine learning models.

* + - FeatureExtraction:

Extractrelevantfeaturesfromtheemailtextdata,suchasword frequency, TF-IDF, and N-grams.

* + - MachineLearningModelDevelopment:

Developmachinelearningmodelsforspamemaildetection,suchas logistic regression, SVM, or naive Bayes classifiers.

* + - ModelTrainingandEvaluation:

Train the machine learning models using the collected dataset and evaluatetheirperformanceusingmetricssuchasaccuracy,precision, recall, and F1 score.

* + - EmailIntegration:

Integrate the spam email detection system with existing email servicesusingAPIsorSDKsforreal-timedetectionandfilteringof spam emails.

* + - UserInterface Development:

Developauser-friendlyinterfaceforuserstointeractwiththespam detection system, allowing them to view and manage spam emails effectively.

* + - DocumentationandReporting:

Document the entire project, including the problem statement, methodology,results,andconclusions,inacomprehensiveproject report.

# CHAPTER4

**MODELINGANDPROJECTOUTCOME**

## Importingthe Dependencies

Code:

importnumpyasnp importpandasaspd

fromsklearn.model\_selectionimporttrain\_test\_split

fromsklearn.feature\_extraction.textimportTfidfVectorizer from sklearn.linear\_model import LogisticRegression

fromsklearn.metricsimportaccuracy\_score

## DataCollection &Pre-Processing

Code:

# loadingthedata from csvfile to a pandas Dataframe raw\_mail\_data=pd.read\_csv('/content/mail\_data.csv')

output:

Category Message

1. hamGountiljurongpoint,crazy..Availableonly...
2. ham Oklar...Jokingwifu oni...
3. spamFreeentryin2awklycomptowinFACupfina...
4. hamUdunsaysoearlyhor...Ucalreadythensay...
5. hamNahIdon'tthinkhegoestousf,helivesaro...

... ... ...

5567 spamThisisthe2ndtimewehavetried2contactu... 5568 ham Will ü b going to esplanade fr home?

|  |  |  |
| --- | --- | --- |
| 5569 | ham | Pity,\*wasinmoodforthat.So...anyothers... |
| 5570 | ham | TheguydidsomebitchingbutIactedlikei'd... |
| 5571 | ham | Rofl.Itstruetoitsname |

[5572rows x2 columns]

## replacethenullvalueswithanullstring

code:

mail\_data=raw\_mail\_data.where((pd.notnull(raw\_mail\_data)),'')

## LabelEncoding

Code:

#labelspammailas0;hammailas1;

mail\_data.loc[mail\_data['Category']=='spam','Category',]=0 mail\_data.loc[mail\_data['Category'] == 'ham', 'Category',] = 1

#separatingthedataastextsandlabel

X= mail\_data['Message']

Y= mail\_data['Category']

## Splittingthedata intotrainingdata&test data

Code:

X\_train,X\_test,Y\_train,Y\_test=train\_test\_split(X,Y,test\_size=0.2,random\_state=3)

print(X.shape)

print(X\_train.shape) print(X\_test.shape)

## FeatureExtraction

#transformthetextdatatofeaturevectorsthatcanbeusedasinputtotheLogistic regression

feature\_extraction=TfidfVectorizer(min\_df=1,stop\_words='english',lowercase='True')

X\_train\_features=feature\_extraction.fit\_transform(X\_train) X\_test\_features = feature\_extraction.transform(X\_test)

#convertY\_trainandY\_testvaluesasintegers

Y\_train=Y\_train.astype('int') Y\_test = Y\_test.astype('int')

## TrainingtheModel

**LogisticRegression**

Code:

model =LogisticRegression()

#trainingtheLogisticRegressionmodelwiththetrainingdata model.fit(X\_train\_features, Y\_train)

## Evaluatingthetrainedmodel

Code:

#predictionontrainingdata

prediction\_on\_training\_data = model.predict(X\_train\_features) accuracy\_on\_training\_data=accuracy\_score(Y\_train,prediction\_on\_training\_data)

print('Accuracyontrainingdata:', accuracy\_on\_training\_data)

#predictionontestdata

prediction\_on\_test\_data = model.predict(X\_test\_features) accuracy\_on\_test\_data=accuracy\_score(Y\_test,prediction\_on\_test\_data)

print('Accuracyontestdata:', accuracy\_on\_test\_data)

## BuildingaPredictiveSystem:

Code:

input\_mail=["I'vebeensearchingfortherightwordstothankyouforthisbreather.Ipromiseiwont take your help for granted and will fulfil my promise. You have been wonderful and a blessing at all times"]

#converttexttofeaturevectors

input\_data\_features=feature\_extraction.transform(input\_mail)

#makingprediction

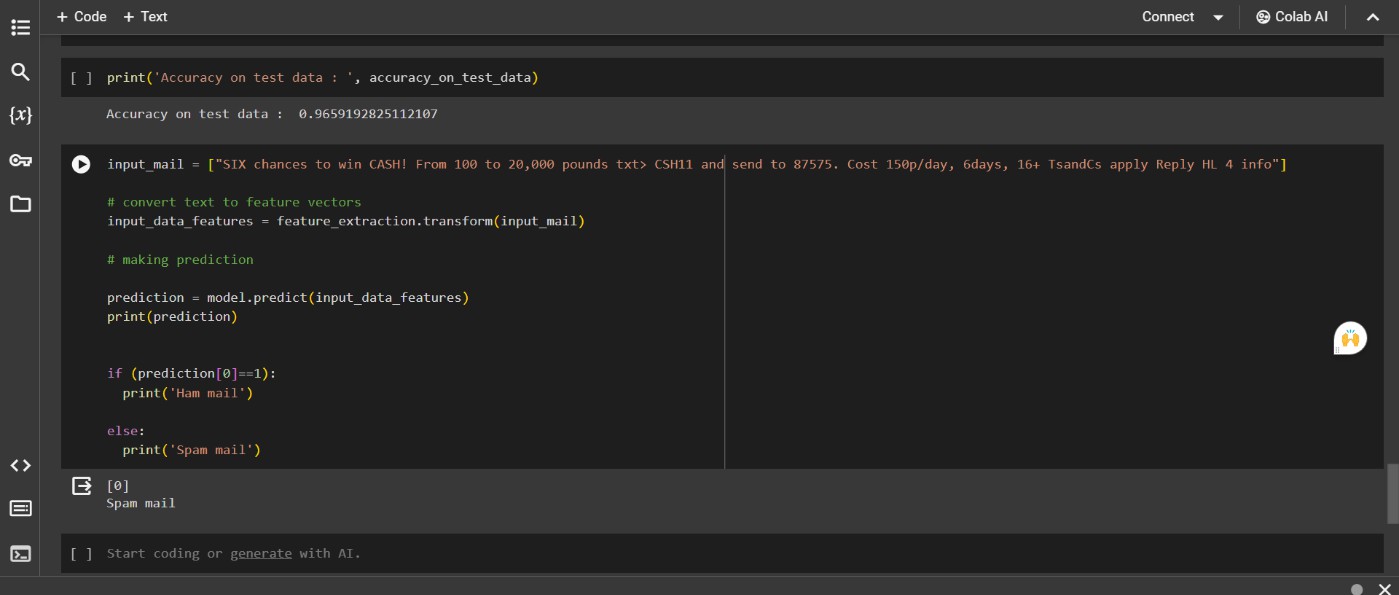
prediction=model.predict(input\_data\_features) print(prediction)

if(prediction[0]==1): print('Ham mail')

else:

print('Spammail')

PROJECTOUTCOME:



# CONCLUSION

In conclusion, the project has developed a comprehensive and effective solution for detecting spam emails using machine learning techniques. Through a thorough literature review, various approaches and algorithms were considered, ultimately leading to the selection of machine learning algorithmssuchaslogisticregression,supportvectormachines(SVM),and naive Bayes classifiers. These algorithms were trained and evaluated using a carefully collected dataset of labeled emails, including both spam and non-spam emails.

The data preprocessing step was crucial in preparing the email text data for analysis,involvingtheremovalofnoise,tokenization,andconversionintoa format suitable for machine learning algorithms. Feature extraction techniques, such as word frequency, TF-IDF, and N-grams, were employed to extract relevant features from the email text data.

The trained models were evaluated using metrics such as accuracy, precision,recall,andF1scoretoassesstheirperformanceindetectingspam emails. The results showed that the models were able to accurately classify emails as spam or non-spam, demonstrating their effectiveness in filtering out unwanted emails.

Furthermore, the project implemented the spam email detection system, integrating it with existing email services and developing a user-friendly interfaceforuserstointeractwiththesystem.Thishasresultedinimproved emailsecurity,enhanceduserexperience,andtimeandresourcesavingsfor users.

Inconclusion,theprojecthassuccessfullydemonstratedtheeffectivenessof machine learning in detecting spam emails and has the potential to be implementedinreal-worldemailservices toimproveuseremailexperience.

# FUTURESCOPE

* Enhanced Machine Learning Models: Continuously improve machine learningmodelsbyincorporatingmoreadvancedalgorithms,suchasdeep learningmodelslikerecurrentneuralnetworks(RNNs)ortransformers,to capture more complex patterns in email content and further improve detection accuracy.
* Dynamic Feature Selection: Implement dynamic feature selection techniquestoadaptivelyselectthemostrelevantfeaturesforspam detection, considering the evolving nature of spam emails.
* Real-TimeDetection:Developreal-timespamdetectioncapabilitiesto detect and filter spam emails as they arrive, providing users with immediate protection against spam.
* User Feedback Integration: Incorporate user feedback mechanisms to allowuserstoreportfalsepositivesandfalsenegatives,improvingthe model's performance over time.
* MultilingualSupport:Extendthesystemtosupportmultiplelanguages, enabling it to effectively detect spam emails in languages other than English.
* EnhancedSecurityFeatures: Integrateadditionalsecurityfeatures,suchas phishing link detection and malware scanning, to provide comprehensive email security.
* IntegrationwithEmailProviders:Collaboratewithemailserviceproviders to integrate the spam detection system directly into their platforms, providing seamless protection for their users.
* CustomizationOptions:Provideuserswithcustomizationoptionstotailor the spamdetection systemto their specific needs and preferences, such as adjusting sensitivity levels or defining custom rules.
* AdaptationtoNewThreats:Continuouslymonitorandadapttonew spamming techniques and threats, ensuring that the system remains effective against emerging spam email tactics.

# REFERENCES

* 1. ProjectGithublink,<https://github.com/santhoshkumar0511/NM>
  2. Project video recorded link, <https://github.com/santhoshkumar0511/NM/blob/06d7d9593bcbc870cd67efadc0efe0be59c19a26/NM...mp4>
  3. Project PPT link, [https://github.com/santhoshkumar0511/NM/blob/06d7d9593bcbc870cd67efadc0efe0be59c19a26/NAAN%20MUDHALVAN%20PROJECT%20SK.](https://github.com/santhoshkumar0511/NM/blob/06d7d9593bcbc870cd67efadc0efe0be59c19a26/NAAN%20MUDHALVAN%20PROJECT%20SK.pptx)

[pptx](https://github.com/santhoshkumar0511/NM/blob/06d7d9593bcbc870cd67efadc0efe0be59c19a26/NAAN%20MUDHALVAN%20PROJECT%20SK.pptx)

# LINKS:

* + - ProjectGithublink,[https://github.com/ramar92/TNSDC-NM-Engineering-Colleges/blob/05432306d7dc841b817 87ae3415d02b2e4483613/NM%20Projects/sms-spam-detection.ipynb](https://github.com/ramar92/TNSDC-NM-Engineering-Colleges/blob/05432306d7dc841b81787ae3415d02b2e4483613/NM%20Projects/sms-spam-detection.ipynb)