EMAIL SPAM CLASSIFIER

Enrollment No. (s) - 19103034, 19103044, 19103211
Name of Student (s) - Roshni Singh, Tanishk Gupta, Deepika Khullar
Submitted to - Dr. Neetu Sardana

Information Retrieval and Semantic Web

PROJECT REPORT

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING & INFORMATION TECHNOLOGY

JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

NOVEMBER 2022

Table of Contents

Introduction

Spam emails are unsolicited email messages that are sent by people we don't know. They include promotional emails that we did not ask for, counterfeit notes that attempt to trick us into giving out sensitive personal information, and fraudulent messages from hacked email accounts. Junk emails waste a bunch of time and effort that could've been used for something more productive but that's not even the worst part. Spam is also a popular means of transferring harmful malware and electronic viruses. And in an age where hacking tools and techniques grow increasingly sophisticated by the minute, spam-instigated security attacks become a perpetual threat. In this world of developing companies, there are lots of emails/spam which is not important, considering our interests and subscriptions. So for this problem, we are making an Email Spam Classifier that will help us to separate genuine mail from spam. We will be using a public data set of emails from Kaggle to test our spam classifier.

Problem Statement

Spam is also a popular means of transferring harmful malware and electronic viruses. As per data port spam statistics, email spam costs businesses $20.5 billion every year. Scams and fraud comprise only 2.5% of all spam emails; however, phishing statistics indicate that identity theft makes up 73% of this figure. Accurate prediction of email spam is highly valuable. This project presents our attempt to develop a model to accurately and quickly predict whether emails are spam or not using the dataset from Kaggle.

Literature Study

1. Study on the Effect of Preprocessing Methods for Spam Email Detection

Link: http://socj.telkomuniversity.ac.id/ojs/index.php/indojc/article/view/284

This research studies the effect of preprocessing steps on the performance of supervised spam classifier algorithms. Experiments were conducted on two widely used supervised spam classifier algorithms: Naïve Bayes and Support Vector Machine. The evaluation is performed on the Ling-spam corpus dataset and uses evaluation metrics: accuracy. This study recommends that the use (or no use) of the appropriate pre-processing methods on each classifier will result in better accuracy. This depends on the classifier used. For the Naïve Bayes classifier, the combination of stop words removal and stemming gives better results than other combinations. However, for a Support Vector Machine (SVM) classifier, the preprocessing stage often does not provide an increase in classification results. This difference is caused by the characteristics of these two classifiers.

2. Analysis of Naïve Bayes Algorithm for Email Spam Filtering across Multiple Datasets

Link: https://iopscience.iop.org/article/10.1088/1757-899X/226/1/012091/meta

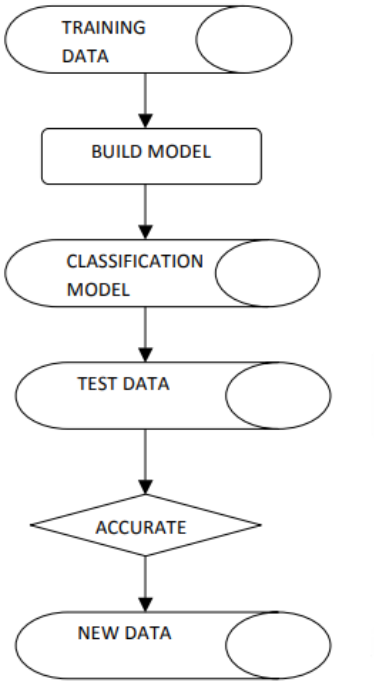
In this research, Naïve Bayes algorithm for e-mail spam filtering on two datasets and test its performance, i.e., Spam Data and SPAMBASE datasets. The performance of the datasets is evaluated based on their accuracy, recall, precision, and F-measure.

Dataset Used

The data used for this project was taken from the Kaggle website. The dataset contains a randomly selected collection of emails in plain text format, which have been labeled as HAM or SPAM. The data is used to build a model for classifying emails into HAM and SPAM & used to check the accuracy of the model built with the training data. The data set contains 5575 emails with 4829 ham and 748 spam emails.
Link: Dataset

Implementation

We will first extract a dataset of 'Email Spam Collection Dataset', then will apply Data cleaning, data preprocessing, EDA, and feature engineering, then we will build our model using Naive Bayes Classifier (Gaussian NB, Multinomial NB, Bernoulli NB), Logistic regression, Decision Tree Classifier, K Neighbours Classifier, Random Forest Classifier, AdaBoost Classifier, Bagging Classifier, ExtraTrees Classifier, Gradient Boosting Classifier, XGB Classifier and SVC (Support Vector Classifier). Evaluation is done using accuracy_score, and precision_score (The most important metric is Precision as it is a high-precision model, we want the least false positive). After that, we will integrate our model into the website.



Results

	Algorithm	Accuracy	Precision	Accuracy_scaling_x	Precision_scaling_x	Accuracy_scaling_y	Precision_scaling_y	Accuracy_num_chars	Precision_num_chars
0	KN	0.905222	1.000000	0.905222	1.000000	0.905222	1.000000	0.905222	1.000000
1	NB	0.970986	1.000000	0.970986	1.000000	0.970986	1.000000	0.970986	1.000000
2	RF	0.974855	0.982759	0.974855	0.982759	0.974855	0.982759	0.974855	0.982759
3	SVC	0.975822	0.974790	0.975822	0.974790	0.975822	0.974790	0.975822	0.974790
4	ETC	0.974855	0.974576	0.974855	0.974576	0.974855	0.974576	0.974855	0.974576
5	LR	0.958414	0.970297	0.958414	0.970297	0.958414	0.970297	0.958414	0.970297
6	xgb	0.971954	0.943089	0.971954	0.943089	0.971954	0.943089	0.971954	0.943089
7	AdaBoost	0.960348	0.929204	0.960348	0.929204	0.960348	0.929204	0.960348	0.929204
8	GBDT	0.947776	0.920000	0.947776	0.920000	0.947776	0.920000	0.947776	0.920000
9	BgC	0.957447	0.867188	0.957447	0.867188	0.957447	0.867188	0.957447	0.867188
10	DT	0.928433	0.820000	0.928433	0.820000	0.928433	0.820000	0.928433	0.820000

As a conclusion, we decided to move forward with the Naive Bayes Classifier as it gives the highest precision, with a high level of accuracy.

Conclusion

Given a set of words, we used feature selection to obtain words that allow us to distinguish between spam and ham emails. We also compared the accuracy of various classifiers in predicting the class attribute. We see that the Naive Bayes method gives the highest precision no matter how many attributes are used and which method is used.

Contribution

Name	**Enrollment No.**	**Work**
Roshni Singh	19103034	Research, Code, Report
Tanishk Gupta	19103044	Research, Code, Report
Deepika Khullar	19103211	Research, Code, Report

References

1. https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=email+spam+classifier&btnG=
2. https://www.researchgate.net/profile/Sudhakar-Pandiarajan/publication/290651051_Comparative_Study_on_Email_Spam_Classifier_using_Data_Mining_Techniques/links/582dd43608aef19cb813dd23/Comparative-Study-on-Email-Spam-Classifier-using-Data-Mining-Techniques.pdf

3. http://socj.telkomuniversity.ac.id/ojs/index.php/indojc/article/view/284/117

4. https://iopscience.iop.org/article/10.1088/1757-899X/226/1/012091/pdf

5. https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset