## EMAIL INTELLIGENCE USING MACHINE LEARNING AND NLP

**KARTHIKRAJ G**

Student ID – 210419104057

Under the supervision of

**PREMLATHA T**

**JULY 2022**

**Abstract**

Email intelligence is an automatic process based on continuous machine learning used to gain statistical insights into the emails stored in the database. Email Intelligence or Email Analysis is one of the most powerful and comprehensive tool, providing the most accurate research-based data for many companies that has a large email database. It is done with the help of machine learning and Natural Language Processing. This project takes in the input as files in the same format as how the emails are stored in the back end. The model classifies whether the email is spam or ham. It gives a short summary of contents present in the email, as well as the user is can select to focus on spam, ham or both. The model is built upon various machine learning models for spam classification and the best out of those is selected. Natural Language Processing is used for pre-processing the email as well as for summarizing the content of the email.

**Table of Contents**

**[1.](#_Toc70512430)****[Background](#_Toc70512430)** [1](#_Toc70512430)

**[2.](#_Toc70512431)****[Related Research](#_Toc70512431)** [2](#_Toc70512431)

**[3.](#_Toc70512432)****[Research Questions](#_Toc70512432)** [2](#_Toc70512432)

**[4.](#_Toc70512433)****[Aim and Objectives](#_Toc70512433)** [2](#_Toc70512433)

**[5.](#_Toc70512434)****[Significance of Study](#_Toc70512434)** [2](#_Toc70512434)

**[6.](#_Toc70512435)****[Scope of Study](#_Toc70512435)** [3](#_Toc70512435)

**[7.](#_Toc70512436)****[Research Methodology](#_Toc70512436)** [3](#_Toc70512436)

**[8.](#_Toc70512437)****[Required Resources](#_Toc70512437)** [8](#_Toc70512437)

**[9.](#_Toc70512438)****[Business Value proposition](#_Toc70512438)** [9](#_Toc70512438)

**[10.References](#_Toc70512439)** [9](#_Toc70512439)

# **Background**

In recent years, internet has been created several platforms for making human life become more secure. Among these; e-mail is a substantial platform for user communication. for user communication. spam filtering is a process to detect unsolicited message and prevent from entering into user’s inbox.

Now days, various systems have been existed to generate anti-spam technique for preventing unsolicited email. Most of the anti-spam methods have some inconsistency between false negatives (missed spam) and false positives (rejecting good emails) which act as a barrier for most of the system to make successful anti-spam system. Therefore, an intelligent and effective spam-filtering system is the prime demand for web.

Among various approach. Spam filtering has two major section; “Knowledge engineering” and “Machine learning”. Knowledge engineering is an arrangement of guidelines to determine the spam emails. In contrast, Machine learning is more efficient than knowledge engineering. It does not require any predefined rules. Naive Bayes, Support Vector Machines, Neural Networks, K-nearest neighbour, and artificial immune system are some prominent technique of Machine learning for spam filtering those are works by matching the regular expression, keywords from message text and so on.

Since machine learning is more efficient most of the spam filtration systems are built on this.

The email summarization is the process of getting the short summary of the content present in the email. Text Summarization is process of shortening source text into a small version by preserving the content of information and the meaning of the context. The surplus availability of information in unstructured format have intensively necessitated the research in the area of automatic text summarization. Due to the inability of people to assimilate vast amounts of information, efficient methods of summarizing text are important with the explosion of data available on the Web and other media in the form of unstructured text. Most of the text summarization methods typically engages with multiple approaches either to identify the most relevant sentences in the text or to remove those sentences which are redundant and irrelevant. Automatic Text Summarization shortens the volume of information by creating a summary from one or more paragraph of sentences or from text documents without losing any of the main contents in it. The focus of the summary may be either generic, which captures the important semantic concepts of the documents, or query-based, which captures the sub-concepts of the user query and provides personalized and relevant abstracts based on the matching between input query and document collection.

# **Related Research**

# **Research Questions**

# **Aim and Objectives**

The primary aim corresponding to this research is to perform Email analysis on the email files present in the back end of the database and obtain informative inference out of the performed analysis.

The research objectives are formulated based on the aim of this study which are as follows:

* 1. To read the email files and extracting required information from those files.
  2. To use model to classify the emails into spam or ham.
  3. To use pre-trained model to summarize the content in the email.
  4. To provide options to the user on whether they have to focus on spam, ham or both.

# **Significance of Study**

With increase in number of companies that use email as standard basis of communication to its clients, there overall email database is also growing rapidly. When dealing with such large database consists emails, they have to spend a lot of time and effort to go through each and every email. But by using this project, the companies can save up lot of time and effort, because the model itself will provide us with the important aspects of an email such as whether it is spam or ham, and the short summary for the corresponding email.

It is not only useful for the companies, it is useful to the clients as well. There is also an another import feature provided by this project, which is to select on which to focus, Spam or ham or considering both. If the user is concerned about spam being misclassified as ham, or vice versa, the user can make the model to give more weightage to that particular class.

# **Scope of Study**

* To help users or companies that work with large amount of emails on a daily basis
* To eliminate the involvement of human interaction in the process of extracting information before feeding into models, as this project does that automatically
* To provide customized service to each user

# **Research Methodology**

The flow of entire research project will be discussed below.

## Introduction

The research/project will be carried out in various stages Basic parts are Dataset loading, Information extraction, pre-processing,model building, evaluation and deployment.

## Dataset Description

## The data set for this project is not in the form of csv or any other kind of tabular data. The data set consists of email files as such that are being stored in the back end. It is simply a large string in same format as a email without any visual formatting put into in.

## Dataset Loading

## The data is present in different folders and all files in the ham folder belongs to the ham mails and all files in the spam folder belongs to spam mails. These files are changed into readable format and then it is loaded one by one and stored into a Dataset.

## Data Pre-processing

There are several steps involved in pre-processing of the database, because data cannot be directly fed into the model for training. First some basic information is formed by doing the following steps:

* Extracting the body from the whole email
* Checking whether the mail address is disposable or not
* Checking whether the root address and the sender address is the same

**7.4.1 Lower Case**

If the text is in the same case, it is easy for a machine to interpret the words because the lower case and upper case are treated differently by the machine. for example, words like Ball and ball are treated differently by machine. So, we need to make the text in the same case and the most preferred case is a lower case to avoid such problems.

**7.4.2 Remove punctuations**

One of the other text processing techniques is removing punctuation. There are total 32 main punctuations that need to be taken care of. we can directly use the string module with a regular expression to replace any punctuation in text with an empty string.

**7.4.3 Lemmatizing**

Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item. Lemmatization is similar to stemming but it brings context to the words. So it links words with similar meanings to one word..

**7.4.4 Remove Stopwords**

Stop words are the most commonly occurring words in a text which do not provide any valuable information. Stop words like they, there, this, where, etc are some of the stopwords. NLTK library is a common library that is used to remove stopwords and include approximately 180 stopwords which it removes. If we want to add any new word to a set of words then it is easy using the add method.

**7.4.5 Remove Extra Spaces**

Most of the time text data contain extra spaces or while performing the above pre-processing techniques more than one space is left between the text so we need to control this problem. regular expression library performs well to solve this problem. It also removes the empty lines in the email.

## Email Summarization

Summarization is performed using a pre-trained model known as T5 model along with a pre-trained tokenizer which takes in a String with multiple sentences and returns the tokens for those sentences. The T5 model takes these tokens as input and returns the summary based on values for the hyper parameters. Both the lemmatized as well as non lemmatized are summarized to give the general summary of the content inside the mail.

## Count Vectorizering

In this process we will use count vectorizer to vectorize the final pre-processed data-frame which returns a count vector matrix. In the count vector matrix, the columns are the alphabetical arrangement of the top n number of words used in the entire corpus of data, where is defined by the max features parameter.

## Model Building

In this process we will train multiple models with the final Count vector matrix, and the model with the most accuracy is selected. The various models used are:

**7.7.1 Multinomial Naive Bayes**

Multinomial Naive Bayes algorithm is a probabilistic learning method that is mostly used in Natural Language Processing (NLP). The algorithm is based on the Bayes theorem and predicts the tag of a text such as a piece of email or newspaper article. It calculates the probability of each tag for a given sample and then gives the tag with the highest probability as output. Naive Bayes classifier is a collection of many algorithms where all the algorithms share one common principle, and that is each feature being classified is not related to any other feature. The presence or absence of a feature does not affect the presence or absence of the other feature.

**7.7.1 K Nearest Neighbor(KNN)**

K Nearest Neighbor(KNN) is a very simple, easy to understand, versatile and one of the topmost machine learning algorithms. KNN used in the variety of applications such as finance, healthcare, political science, handwriting detection, image recognition and video recognition. In Credit ratings, financial institutes will predict the credit rating of customers. In loan disbursement, banking institutes will predict whether the loan is safe or risky. In political science, classifying potential voters in two classes will vote or won’t vote. KNN algorithm used for both classification and regression problems. KNN algorithm based on feature similarity approach..

**7.7.1 SVM classifier**

Support Vector Machines (SVM) are machine learning algorithms that are used for classification and regression purposes. SVMs are one of the powerful machine learning algorithms for classification, regression and outlier detection purposes. An SVM classifier builds a model that assigns new data points to one of the given categories. Thus, it can be viewed as a non-probabilistic binary linear classifier.

**7.7.1 Decision Tree Classifier**

Decision Tree algorithm belongs to the family of supervised learning algorithms. Unlike other supervised learning algorithms, the decision tree algorithm can be used for solving regression and classification problems too. The goal of using a Decision Tree is to create a training model that can use to predict the class or value of the target variable by learning simple decision rules inferred from prior data(training data)

**7.7.1 Random Forest Classifier**

Random forest has nearly the same hyper parameters as a decision tree or a bagging classifier. Fortunately, there’s no need to combine a decision tree with a bagging classifier because you can easily use the classifier-class of random forest. With random forest, you can also deal with regression tasks by using the algorithm’s regressor. Random forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model.

**7.7.1 Ada Boost Classifier**

Adaptive Boosting combines multiple classifiers to increase the accuracy of classifiers. AdaBoost is an iterative ensemble method. AdaBoost classifier builds a strong classifier by combining multiple poorly performing classifiers so that you will get high accuracy strong classifier. The basic concept behind Adaboost is to set the weights of classifiers and training the data sample in each iteration such that it ensures the accurate predictions of unusual observations.

## Model Controls – Hyper parameter Tuning

Class\_weightage plays an important part role in this project. It is a hyper parameter that is used to give individual classes specific weightage according to user’s choice. When the user wants to focus mainly on not classifying a spam as non spam, then the spam class is given more weightage, so that it doesn’t miss classify spam. And the same is done for ham class also. But by default we have set it to give spam class to have a little higher weightage than the ham class.

## Model Evaluation

The R-square value of all model with test and train are compared with each other. R-squared (R2) is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a [regression](https://www.investopedia.com/terms/r/regression.asp) model. Whereas correlation explains the strength of the relationship between an independent and dependent variable, R-squared explains to what extent the variance of one variable explains the variance of the second variable. The Closer to 1 it is, the better the model it is.

In this project both Decision tree as well as adaboost gave 100% accuracy, but Decision tree is a much less complex model compared to Adaboost, therefore Decision tree is chosen as the main model for classification of spam or ham.

## Model deployment and consumption

Once the model has been finetuned, we will save the model file and use FLASK or Streamlit to consume the model as an API and will connect it with the web UI swhich would likely to be developed in Angular v9. It can implemented to a local host where we can directly upload the email file and the model runs in the background to give us the Summary and classification of the given mail.

# **Required Resources**

The research will need below hardware and software resources throughout the implementation.

## Software Requirements

* + - Package Manager: Anaconda Navigator 1.9.12
    - Presentation Layer: Jupyter lab 0.35.4
    - Language: Python 3.6.X
    - Python Libraries for machine learning: Pandas and NumPyfor data processing, pytorch, scikit-learn for interpretable models, Simpletransformers for sequence 2 sequnce model,

## Hardware Requirements

A laptop with below configuration will be used.

* + - Operating System. Windows 10 : 64-bit
    - Processor: Intel® Core™ i5 8th gen 8650U Processor
    - Memory: 4 GB

# **Business Value Proposition**

The concepts used in this project such as Natural Language Processing, Text Summarization, Text Pre-processing, Lemmatization and other processes can also be implemented for various other ideas to reduces the manual work done by the employees in areas such as [Media monitoring](https://www.frase.io/blog/20-applications-of-automatic-summarization-in-the-enterprise/" \l "1_Media_monitoring" \o "1. Media monitoring) [Newsletters](https://www.frase.io/blog/20-applications-of-automatic-summarization-in-the-enterprise/" \l "2_Newsletters" \o "2. Newsletters) [Social media marketing](https://www.frase.io/blog/20-applications-of-automatic-summarization-in-the-enterprise/" \l "7_Social_media_marketing" \o "7. Social media marketing) [Question answering and bots](https://www.frase.io/blog/20-applications-of-automatic-summarization-in-the-enterprise/" \l "8_Question_answering_and_bots" \o "8. Question answering and bots), etc.

# **References**

An empirical study on email classification using supervised machine learning in real environments- Wenjuan Li and Weizhi Meng

Department of Computer Science, City University of Hong Kong, Hong Kong SAR, Infocomm Security Department, Institute for Infocomm Research, Singapore

Spam in Q1 2014: US Once Again the Prime Target for Malicious Emails, Online, May, 2014. Available: http://www.kaspersky.com/about/news/spam/2014/Spam-in-Q1 -2014-US-Once-Again-the-Prime-Target-for-Malicious-Emails.

Z. Duan, Y. Dong, and K. Gopalan, “DMTP: Controlling spam through message delivery differentiation,” Computer Networks 51(10), pp. 2616-2630, 2007.

A.G. West, A.J. Aviv, J. Chang, and I. Lee, “Spam mitigation using spatio-temporal reputations from blacklist history,” Proceedings of ACSAC, pp. 161-170, 2010.

Text Summarization Approaches Using Machine Learning & LSTM – Neeraj Kumar Sirohi, Dr. Mamta Bansal, Dr. SN Rajan

Research Scholar, Shobhit Institute of Engineering & Technology, Meerut, India.

Shobhit Institute of Engineering & Technology, Meerut, India.

IMS Engineering College, Ghaziabad, India.

Hou, L., Hu, P. & Bei, C. (2017). Abstractive document summarization via neural model with joint attention. Paper presented at the Natural Language Processing and Chinese Computing, Dalian, China.

Mohan, M. J., Sunitha, C., Ganesh, A., & Jaya, A. (2016). A study on ontology based abstractive summarization. Procedia Computer Science, 87, 32–37. <https://doi.org/10.1016/j.procs.2016.05.122>.

Chitrakala, S., Moratanch, N., Ramya, B., Revanth Raaj, C. G. & Divya, B. (2018). Concept-based extractive text summarization using graph modelling and weighted iterative ranking. In N.R. Shetty, L.M. Patnaik, N.H. Prasad & N. Nalini (Eds.), Emerging research in computing, information, communication and applications: ERCICA 2016 (pp.149–160). Singapore: Springer Singapore.

Text Summarization using Deep Learning- Kasimahanthi Divya, Kambala Sneha, Baisetti Sowmya, G Sankara Rao : Department of Computer Science and Engineering, Gayathri Vidya Parishad College of Engineering for Women (Affiliated by Jawaharlal Nehru Technological University, Kakinada), Visakhapatnam, Andhra Pradesh, India

1. Lloret, M. Palomar, "Text summarization in progress: a literature review" in Springer, Springer, pp. 1-41, 2012

K. Spärck Jones, "Automatic summarizing: The state of the art", Information Processing & Management, vol. 43, pp. 1449-1481, nov 2007

1. Khan, N. Salim, "A review on abstractive summarization methods", Journal of Theoretical and Applied Information Technology, vol. 59, no. 1, pp. 64- 72, 2014

E. Baralis, L. Cagliero, A. Fiori, P. Garza, "MWI-Sum: A Multilingual Summarizer Based on Frequent Weighted Itemsets", ACM Transactions on Information Systems, vol. 34, no. 1, pp. 1-35, 2015

Multi-document Summarization via Deep Learning Techniques: A Survey CONGBO MA, The University of Adelaide WEI EMMA ZHANG, The University of Adelaide MINGYU GUO, The University of Adelaide HU WANG, The University of Adelaide QUAN Z. SHENG, Macquarie University

CONGBO MA, WEI EMMA ZHANG, MINGYU GUO, HU WANG, and QUAN Z. SHENG. 2021. Multi-document Summarization via Deep Learning Techniques: A Survey. 1, 1 (December 2021), 35 pages. https://doi.org/10. 1145/

Xiaojun Wan and Jianwu Yang. 2008. Multi-document Summarization Using Cluster-based Link Analysis. In Proceedings of the 31st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008). Singapore, 299–306.