Emily Kim

Professor Erik Grimmelmann

CSC447 – Introduction to Machine Learning

05.15.2024

Final Report

Introduction

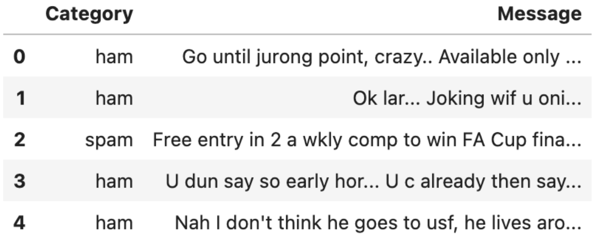
Addressing the challenge of differentiating between spam and non-spam emails is the focus of my project's classification task. Even though current algorithms can categorize emails, spam continues to find its way into our inboxes, suggesting that further improvement is necessary. I focused on developing a classification system that is more accurate and reliable, reducing the amount of spam emails that reach an individual's user's inbox. Throughout my project, I used six distinct models: Naive Bayes (multinomial naive bayes), SVM, Logistic Regression, Random Forest, Gradient Boosting, and Decision Tree. These models played a crucial role in my efforts to identify the most accurate model for accurate email classification.

Pre-processing

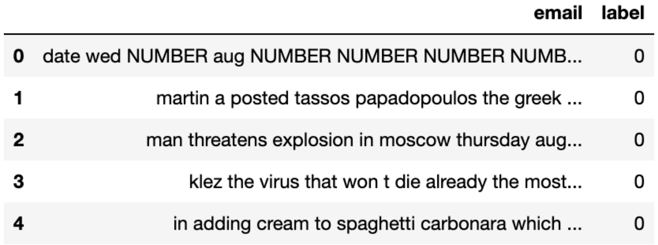
Combining Data:

To prepare my data for a model, a crucial step involves data cleaning. Initially, I identified 2 datasets: spam.csv and spam_or_not_spam.csv. Here are the specifics of each dataset:

- spam.csv
  - Represented as df2 in my notebook
  - 2 columns, 5k rows
  - Columns: Category, Message

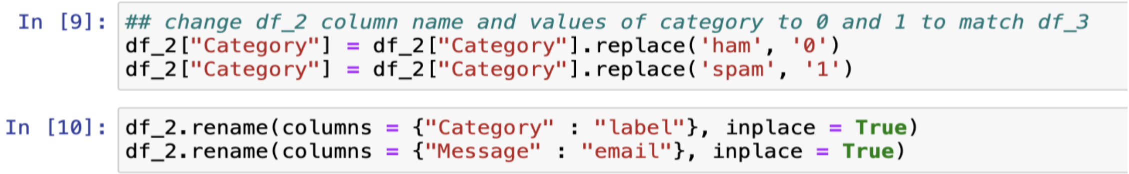
  - Column Category: ham = not-spam, spam = spam
  - 

	Category	Message
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

- spam_or_not_spam.csv.
  - Represented as df3 in my notebook
  - 2 columns, 3k rows
  - Columns: Email, Label
  - Column Label: 0 = not-spam, 1 = spam
  - 

	email	label
0	date wed NUMBER aug NUMBER NUMBER NUMBER NUMB...	0
1	martin a posted tassos papadopoulos the greek ...	0
2	man threatens explosion in moscow thursday aug...	0
3	klez the virus that won t die already the most...	0
4	in adding cream to spaghetti carbonara which ...	0

Both datasets exhibited similar formatting. They both have 2 columns, one indicating spam or not spam, another column providing the string of email messages. To combine it, I've decided to first address the column indicating spam/not-spam. I've decided to indicate spam/not-spam using 1 and 0, respectively. So, in df2 I replaced "ham" with "0" and "spam" with "1". Then, I relabeled the column names in df2 to be "label" and "email". Figure 1 below shows the code for this.

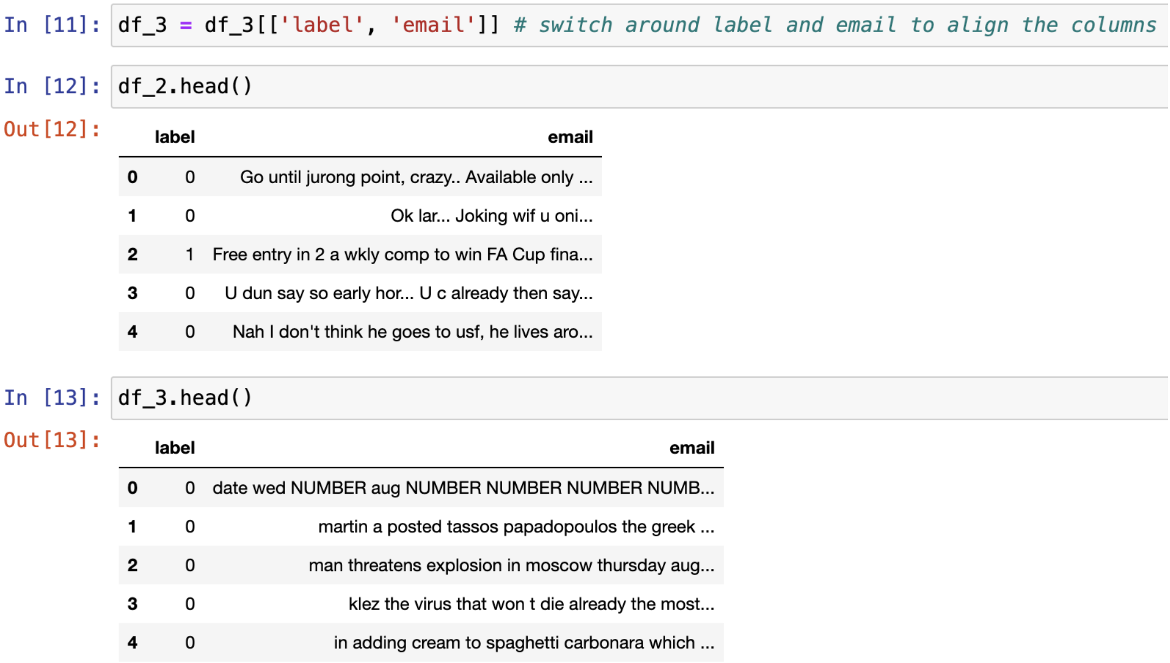


```
 In [9]: ## change df_2 column name and values of category to 0 and 1 to match df_3
         df_2["Category"] = df_2["Category"].replace('ham', '0')
         df_2["Category"] = df_2["Category"].replace('spam', '1')

In [10]: df_2.rename(columns = {"Category" : "label"}, inplace = True)
         df_2.rename(columns = {"Message" : "email"}, inplace = True)
```

Figure 1. Code for changing column names and values

To prepare for merging, I matched the same order of columns as df2. I re-arranged df3's columns to be "label" then "email", instead of its original "email" then "label", as shown in figure 2 below.



```
In [11]: df_3 = df_3[['label', 'email']] # switch around label and email to align the columns

In [12]: df_2.head()
Out[12]:
```

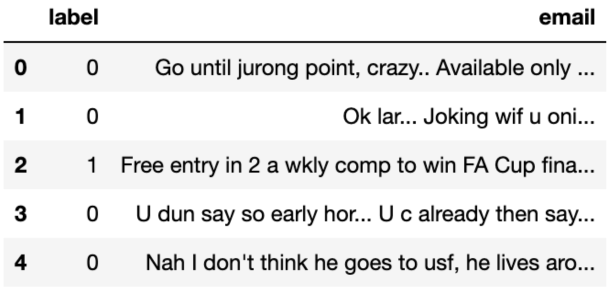
	label	email
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

```
In [13]: df_3.head()
Out[13]:
```

	label	email
0	0	date wed NUMBER aug NUMBER NUMBER NUMBER NUMB...
1	0	martin a posted tassos papadopoulos the greek ...
2	0	man threatens explosion in moscow thursday aug...
3	0	klez the virus that won t die already the most...
4	0	in adding cream to spaghetti carbonara which ...

Figure 2. Changing the order of columns in df3

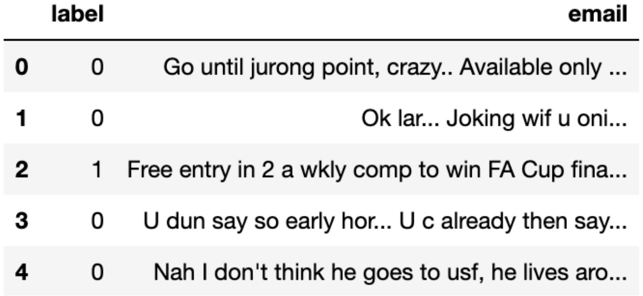
The final combined dataset is seen below:



	label	email
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

Figure 3. Combined dataset

Lastly, I prepared the new data, df_combine, to be converted to CSV file for further Exploratory Data Analysis (EDA) and data preprocessing. The resulting dataset CSV file is characterized by the following details:

- 2 columns, 8k rows
- Columns: Label, Email
- 

	label	email
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

The EDA is delved into further details in the next portion. I identified 513 emails out of 8,029 to be in foreign languages. Contemplating the option of translation, I decided against it due to potential inaccuracies that could disrupt the model's outcomes. Consequently, I collectively opted to omit these rows from our dataset. To standardize the process, I converted all words to lowercase, tokenized the strings, and eliminated stop words and punctuation.

Visualizations/EDA:

Initial exploration included shape and unique values. I had used a bar graph to visually represent the imbalance between spam and non-spam emails within my dataset, examining their respective ratios. As shown in figure 4 below, I had 6471 non-spam and 1039 spam emails in my dataset.

There was a higher number of non-spam emails compared to spam emails. While the dataset’s imbalance – that has been the larger portion of non-spam emails than spam – may provide difficulties for classification models, accuracy tuning still takes priority for this project. This distribution mirrors our own inboxes, where spam emails are, in fact, less common. In other words, in my personal email experience, spam emails are few compared to the abundance of non-spam emails I may receive in a day.

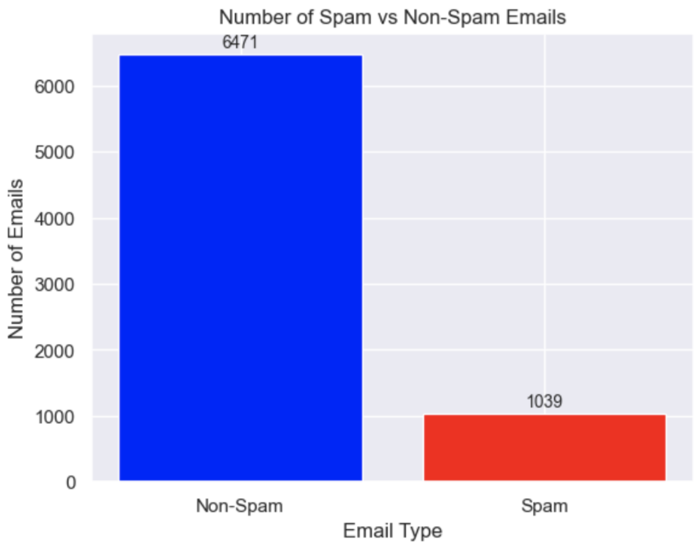


Figure 4. Spam vs Non-spam emails distribution

Additionally, I used another bar graph to compare the average length of spam and non-spam emails. As shown below in figure 5, spam emails had an average of 893 characters, significantly longer than the 518 characters found in non-spam emails. This suggests that spam emails tend to be more verbose in their content, potentially using wordiness as a strategy to draw attention or avoid. These insights broaden my comprehension of the unique qualities that distinguish these email categories.

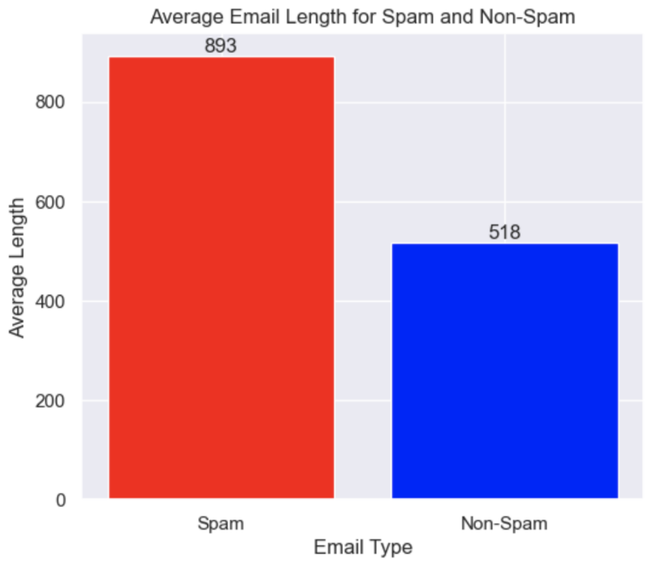
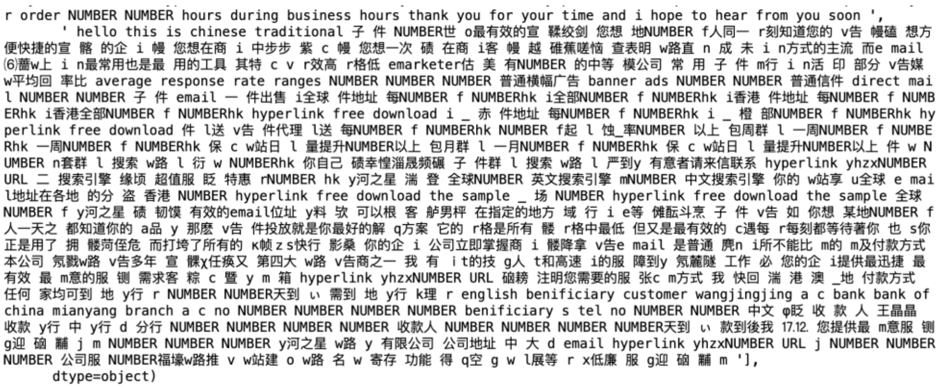


Figure 5. Average email length spam vs non-spam emails

When I had reviewed the unique values, I discovered that there was foreign language within my dataset, as depicted below.



```
r order NUMBER NUMBER hours during business hours thank you for your time and i hope to hear from you soon ',
       ' hello this is chinese traditional 子 件 NUMBER世 o最有效的宣 鞣绞剑 您想 地NUMBER f人同一 r刻知道您的 v告 幔磕 想方
便快捷的宣 髂 的企 i 幔 您想在商 i 中步步 紫 c 幔 您想一次 碛 在商 i客 幔 越 碓蕉嗟恼 查表明 w路直 n 成 未 i n方式的主流 而e mail
⑹蔷w上 i n最常用也是最 用的工具 其特 c v r效高 r格低 emarketer估 美 有NUMBER 的中等 模公司 常 用 子 件 m行 i n活 印 部分 v告媒
w平均回 率比 average response rate ranges NUMBER NUMBER NUMBER 普通横幅广告 banner ads NUMBER NUMBER 普通信件 direct mai
l NUMBER NUMBER 子 件 email 一 件出售 i全球 件地址 每NUMBER f NUMBERhk i全部NUMBER f NUMBERhk i香港 件地址 每NUMBER f NUMB
ERhk i香港全部NUMBER f NUMBERhk hyperlink free download i _ 赤 件地址 每NUMBER f NUMBERhk i _ 橙 部NUMBER f NUMBERhk hy
perlink free download 件 l送 v告 件代理 l送 每NUMBER f NUMBERhk NUMBER f起 l 蚀_率NUMBER 以上 包周群 l 一周NUMBER f NUMBE
Rhk 一周NUMBER f NUMBERhk 保 c w站日 l 量提升NUMBER以上 包月群 l 一月NUMBER f NUMBERhk 保 c w站日 l 量提升NUMBER以上 件 w N
UMBER n套群 l 搜索 w路 l 衍 w NUMBERhk 你自己 碛幸惶淄晟频碾 子 件群 l 搜索 w路 l 严到y 有意者请来信联系 hyperlink yhzxNUMBER
URL 二 搜索引擎 缘顷 超值服 贬 特惠 rNUMBER hk y河之星 湍 登 全球NUMBER 英文搜索引擎 mNUMBER 中文搜索引擎 你的 w站享 u全球 e mai
l地址在各地 的分 盗 香港 NUMBER hyperlink free download the sample _ 场 NUMBER hyperlink free download the sample 全球
NUMBER f y河之星 碛 韧馍 有效的email位址 y料 欤 可以根 客 舻男枰 在指定的地方 域 行 i e等 傩酝斗烹 子 件 v告 如 你想 某地NUMBER f
人一天之 都知道你的 a品 y 那麽 v告 件投放就是你最好的解 q方案 它的 r格是所有 髅 r格中最低 但又是最有效的 c遇每 r每刻都等待著你 也 s你
正是用了 拥 髅菏侄危 而打垮了所有的 κ帧ｚs快行 影桑 你的企 i 公司立即掌握商 i 髅降拿 v告e mail 是普通 腮n i所不能比 m的 m及付款方式
本公司 氖戮w路 v告多年 宣 髁χ任痪又 第四大 w路 v告商之一 我 有 ｉt的技 g人 t和高速 i的服 障到y 氖麓隧 工作 必 您的企 i提供最迅捷 最
有效 最 m意的服 铡 需求客 粽 c 暨 y m 箱 hyperlink yhzxNUMBER URL 硇耪 注明您需要的服 张c m方式 我 快回 湍 港 澳 _地 付款方式
任何 家均可到 地 y行 r NUMBER NUMBER天到 ぃ 需到 地 y行 k理 r english benificiary customer wangjingjing a c bank bank of
china mianyang branch a c no NUMBER NUMBER NUMBER NUMBER benificiary s tel no NUMBER NUMBER 中文 φ贬 收 款 人 王晶晶
收款 y行 中 y行 d 分行 NUMBER NUMBER NUMBER NUMBER 收款人 NUMBER NUMBER NUMBER NUMBER天到 ぃ 款到後我 17.12. 您提供最 m意服 铡
g迎 硇 鳎 j m NUMBER NUMBER NUMBER y河之星 w路 y 有限公司 公司地址 中 大 d email hyperlink yhzxNUMBER URL j NUMBER NUMBER
NUMBER 公司服 NUMBER福壕w路推 v w站建 o w路 名 w 寄存 功能 得 q空 g w l展等 r x低廉 服 g迎 硇 鳎 m '],
      dtype=object)
```

Figure 6. Non-English texts in the dataset

Thus, I addressed the foreign language. I generated a ratio of how many emails were English and non-English that I would have to possibly work with. These are the insights I've discovered: there were 496 non-English emails out of the 8,029 emails in the dataset. Only 467 of them being non-spam and 29 being spam. After analyzing the pie chart, it became clear that non-English emails accounted for a mere 6.2% of the dataset, indicating their insignificance in the overall composition. Therefore, I made the decision to drop the non-English emails, as including it and translating them can be inaccurate, which can skew the final results of my model.

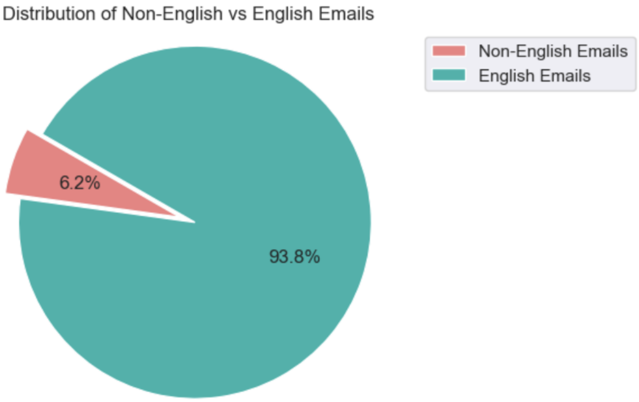


Figure 7. Distribution of non-English vs English emails

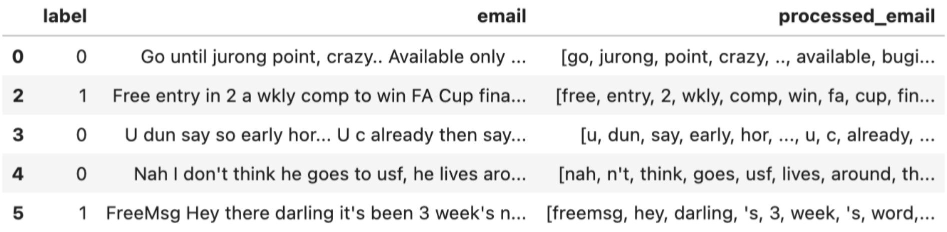
Continued Data Preprocessing:

I then wanted to gain a better insight as to what words in the email message would indicate spam and not-spam. To do this, it required extra data preprocessing:

1. Conversion to lowercase
    a. Many of the email texts had contained capitalized letters, ensuring uniformity for consistent word recognition.
2. Tokenization
    a. Separate the text into a list of individual words.
3. Removal of Stopwords
    a. stopwords such as “is”, “like”, “the”, are common English terminology to generate sentences. When generating a word cloud to see the most frequent words used, these stopwords can throw off the result and it won’t help with generating better insights.
4. Removal of Punctuation
    a. Punctuation is removed as it adds no value to the string.

The outcome of this were then distilled into a new CSV file which was used for the rest of my project, especially in the next visualize that will be shown and the modeling portion:

- preprocessed_english.csv
    - 3 columns, 8k rows
    - Columns: Label, Email, processed_email
    -



	label	email	processed_email
0	0	Go until jurong point, crazy.. Available only ...	[go, jurong, point, crazy, .., available, bugi...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	[free, entry, 2, wkly, comp, win, fa, cup, fin...
3	0	U dun say so early hor... U c already then say...	[u, dun, say, early, hor, ..., u, c, already, ...
4	0	Nah I don't think he goes to usf, he lives aro...	[nah, n't, think, goes, usf, lives, around, th...
5	1	FreeMsg Hey there darling it's been 3 week's n...	[freemsg, hey, darling, 's, 3, week, 's, word,...

Most Frequent Words:

After preprocessing, I wanted to explore the most frequent words in both email categories. The resulting word clouds are displayed below.

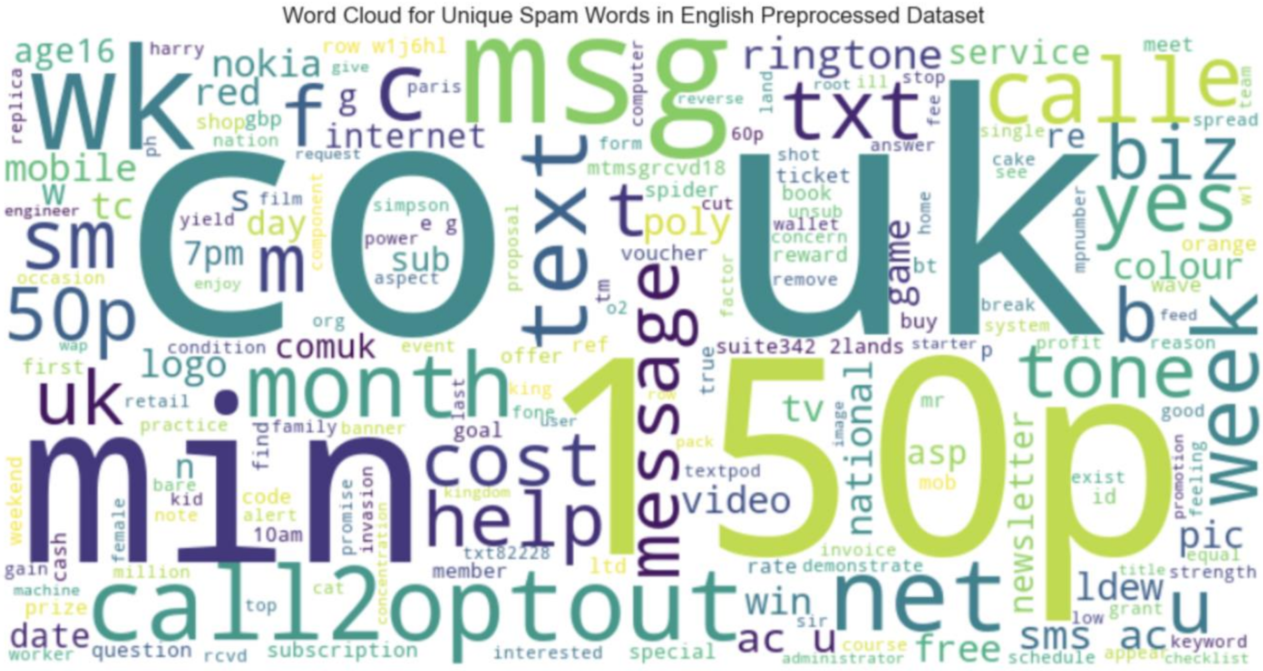


Figure 8. Word cloud for spam emails

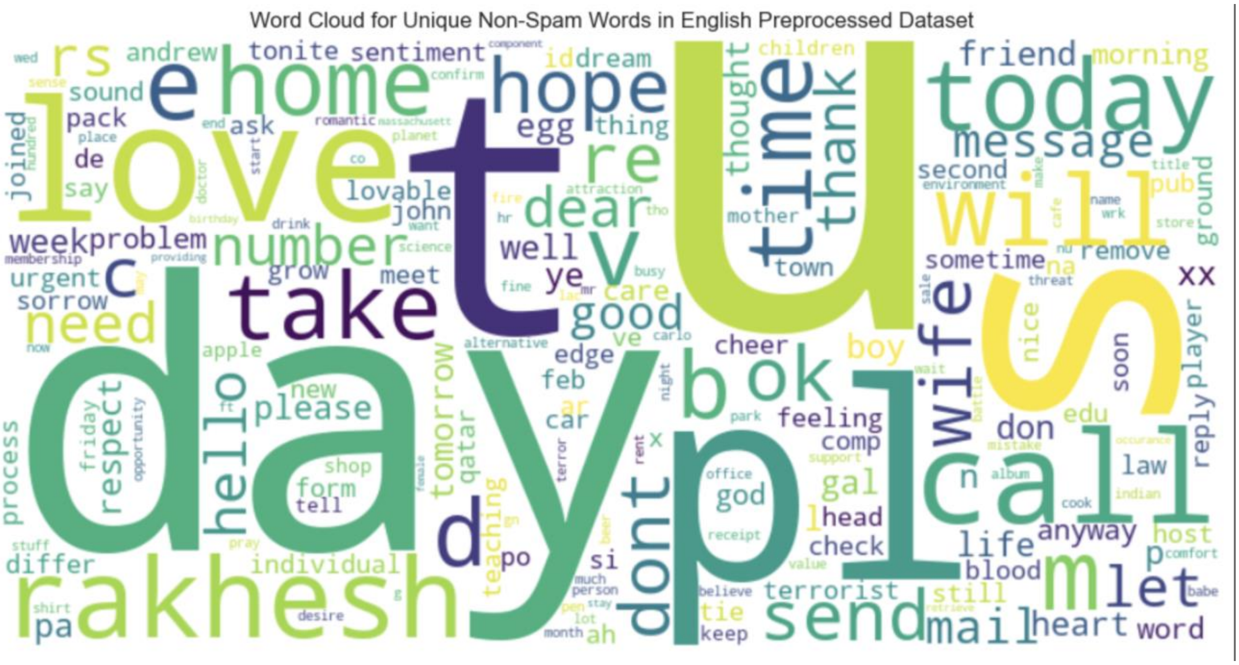


Figure 9. Word cloud for non-spam emails

In figure 8, it illustrates the frequency of distinct occurring words found in spam emails. Noteworthy words include “free”, “win”, “rewards”, “org”, “call”, “msg”, and “offer”. This suggests a common pattern in spam emails, where enticing targets with promises of freebies or urging them to click on links or make calls to claim prizes is a common strategy used by the scammers.

Similarly, figure 9 focuses on non-spam emails, illustrating the frequency of words that reflect more neutral and conversational tones. Words such as “friend”, “good”, “ok”, “call”, “mail”, “love”. This highlights a clear contrast to the common words found in spam emails. However, it is interesting to note that, despite the variations in content between spam and non-spam emails, the frequency of certain words is consistent. The words “call” and “message” pop up as common elements in both categories of emails. This demonstrates that, while the overall tone and intent of the email varies greatly, the need for communication – whether to begin a call or send a message – remains consistent across diverse forms of online interaction.

Modeling

In my project, I used a range of machine learning algorithms chosen based on their perceived suitability and potential efficacy for addressing the project tasks.

1. Naive Bayes (Multinomial Naive Bayes)
    a. The Naive Bayes classifier, specifically the Multinomial Naive Bayes, was chosen for its simplicity and performance in text classification applications. It’s a classification algorithm that works particularly well for text-based data, such as classifying documents or emails into categories like spam or not spam, topics,

sentiment analysis, and so on. This approach assumes independence between features and performs well even with relatively small datasets. Its computational efficiency and ability to handle large feature spaces made it a suitable choice for this project.

2. Support Vector Machine (SVM)

    a. SVM is a powerful classifier that uses a hyperplane to divide data points and optimize class margins. It excels in high-dimensional spaces and was predicted to perform well in our text-based classification task by identifying complex decision boundaries.

3. Logistic Regression

    a. Logistic Regression is a classification machine learning algorithm that uses labeled data to predict a discrete outcome by assigning a predicted probability to each decision. Its simplicity makes it a popular choice for binary classification tasks. In this project, it was used as a baseline model to benchmark performance.

4. Random Forest

    a. Random Forest algorithm is a method that uses decision trees, and was used for its capability to handle non-linear relationships within data and reduce overfitting. By constructing multiple decision trees and collecting their outputs, this model aimed to enhance classification accuracy.

5. Gradient Boosting Models

    a. Gradient Boosting Models, including Gradient Boosting Classifier, was chosen for its ability to combine multiple weak learners (typically decision trees) into a strong learner. This helps to correct errors made by previous models, potentially leading to better predictive performance.

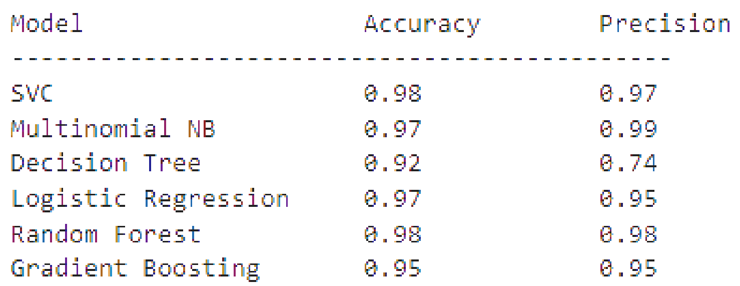
6. Decision Tree
    a. Decision trees classify the examples by organizing the data into a tree structure, starting from a root node, and branching out to some leaf, which represents the final classification of each piece of data. This method provides a more straightforward way to visualize decision-making processes, making it easier to interpret and explain.

The selection of these diverse models aimed to explore various approaches to classifying spam and non-spam emails. This strategy allowed for a comprehensive evaluation of performance across different algorithmic paradigms, helping me to find the most suitable model for my specific project.

Results

Below in figure 10 and figure 11, demonstrates the accuracy and precision scores I had received for each of the six models. My metrics included accuracy and precision.

- *Accuracy* is the count of all the predictions I got correct divided by the total number of predictions, so it represents the percentage of predictions that are correct.
    - A higher accuracy indicates better overall performance.
- *Precision* measures the accuracy of positive predictions made by a model, indicating what proportion of positive identifications were correct. It assesses the percentage of correct examples among all the times our model says “YES”.
    - A higher precision means fewer false positives.



```
Model                    Accuracy        Precision
---------------------------------------------
SVC                      0.98            0.97
Multinomial NB           0.97            0.99
Decision Tree            0.92            0.74
Logistic Regression      0.97            0.95
Random Forest            0.98            0.98
Gradient Boosting        0.95            0.95
```

Figure 10. Accuracy and precision scores of all models

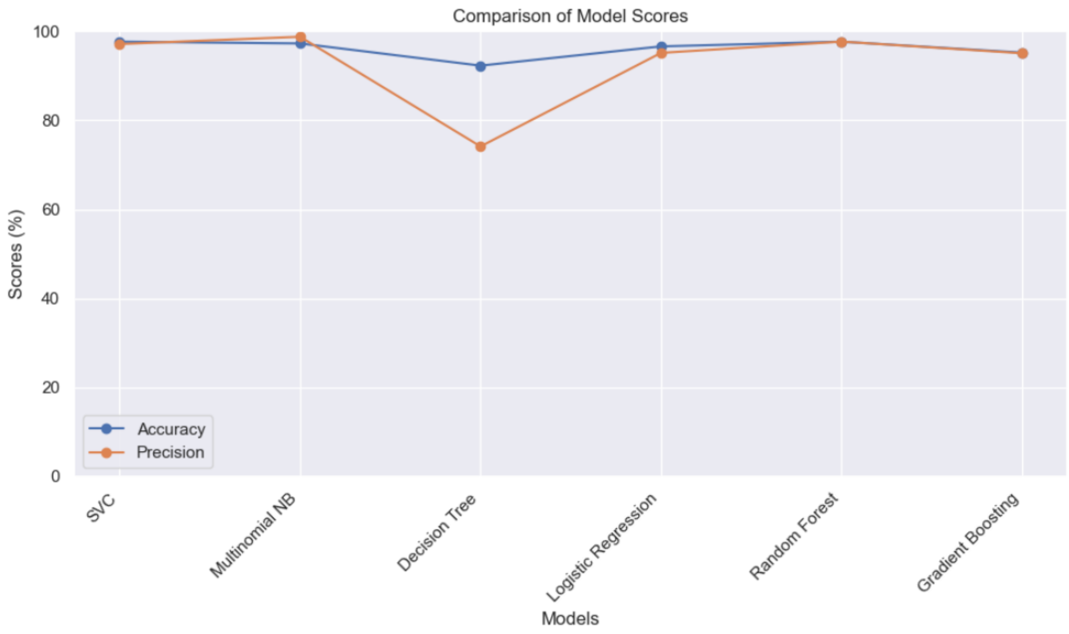


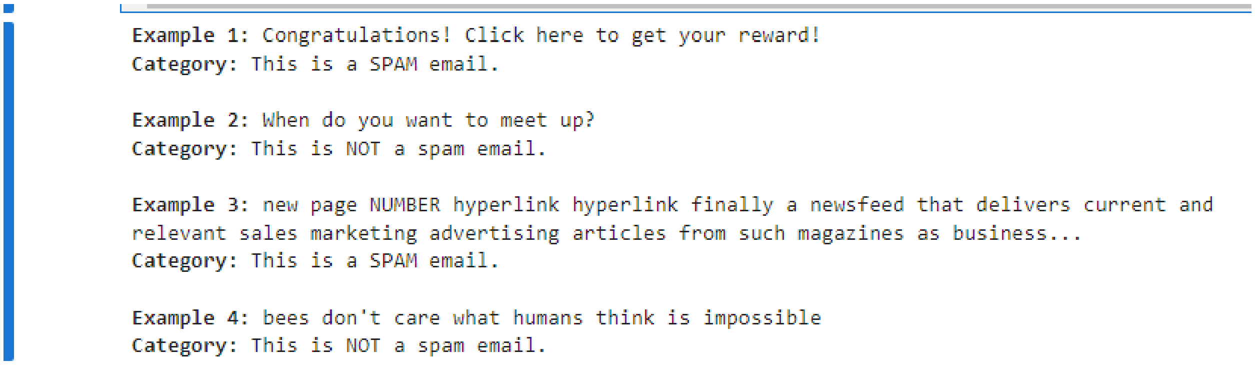
Figure 11. Visual of a comparison of the model scores

Analysis

The top three models in terms of accuracy and precision were Support Vector Classifier (SVC), Multinomial Naïve Bayes, and Random Forest. The SVC and Multinomial Naïve Bayes models had significantly high accuracy scores of 98% and 97%, respectively, as well as precision scores of 97% and 99%, demonstrating the models’ effectiveness in accurately recognizing positive cases. Similarly, the Random Forest model achieved outstanding accuracy and precision, making 98% in both categories. In contrast, the Decision Tree model was the least effective among the group, as illustrated by the dip in figure 11. It had an accuracy score of 92% and a significantly low precision score of 74%.

Predictions

Using the Random Forest Classifier model, it predicted whether a sample email text was spam or non-spam. Below are my 4 examples where it correctly classified each of the text.



```
Example 1: Congratulations! Click here to get your reward!
Category: This is a SPAM email.

Example 2: When do you want to meet up?
Category: This is NOT a spam email.

Example 3: new page NUMBER hyperlink hyperlink finally a newsfeed that delivers current and
relevant sales marketing advertising articles from such magazines as business...
Category: This is a SPAM email.

Example 4: bees don't care what humans think is impossible
Category: This is NOT a spam email.
```

Figure 12. Predictions of 4 examples