Application of the Pattern Growth Approach for detecting spam mail

Teacher: Võ Nguyễn Lê Duy

Group 10:
Lương Anh Huy – 22520550
Phạm Đông Hưng – 22520521
Phan Công Minh - 22520884

Table of contents

01

Introduction

02

Dataset

03

Methodology

04

Result & Analysis

05

Conclusion

01

Introduction

Introduction

- ☆ **What is spam mail?**
- ☆ **Why detecting spam mails is essential?**
- ☆ **Approach for detecting spam mails**

What is spam mail?

E-mail is an effective way of communication as it saves a lot of time and cost. Because of this, websites expose to various types of unwanted and malicious risks - especially spam mails. Spam mails is the practice of frequently sending unwanted data or bulk data in a large quantity to some email accounts.

Introduction

- ☆ **What is spam mail?**
- ☆ **Why detecting spam mails is essential?**
- ☆ **Approach for detecting spam mails**

Why detecting spam mails is essential?

- **Spam Mail** has become an increasing problem in recent years. As the usage of web expanding, problem of spam mails are also expanding.
- So, if the email should be more secure and effective, appropriate email filtering is an essential process.

Introduction

- ☆ **What is spam mail?**

- ☆ **Why detecting spam mails is essential?**

- ☆ **Approach for detecting spam mails**

Approach for detecting spam mails

- Several algorithms are used for classification of spam mails which are extensively utilize and analyze out of which **Logistic Regression**, **Support Vector Machine**, **Naïve Bayes**, **Decision Tree**, **Neural network** classifiers are well known classifiers.

- In this article, **Pattern Growth Approach** - one of the Data Mining algorithms is used for detecting Spam Mails.

Table of contents

01

Introduction

02

Dataset

03

Methodology

04

Result & Analysis

05

Conclusion

02

Dataset

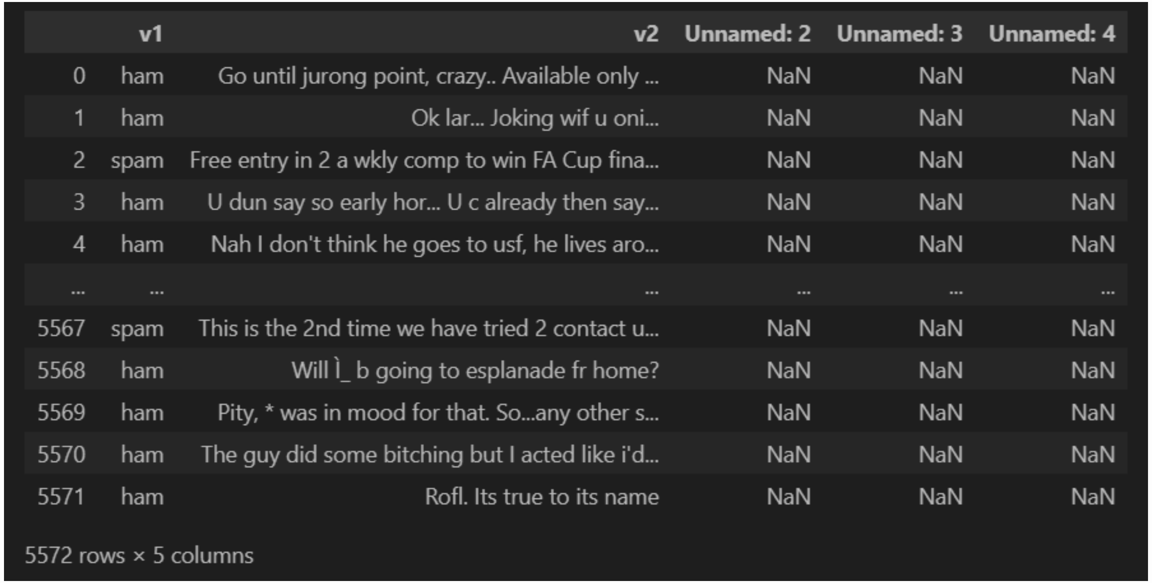
Dataset

- https://raw.githubusercontent.com/mohitgupta-1O1/Kaggle-SMS-Spam-Collection-Dataset-/master/spam.csv

- The Spam Mails Dataset on Kaggle is a valuable resource for those interested in spam email detection and analysis. This dataset consists of a substantial collection of email text messages, categorized as either spam or non-spam (ham).

Overview

- The dataset contains 5572 emails in total and 5 columns.

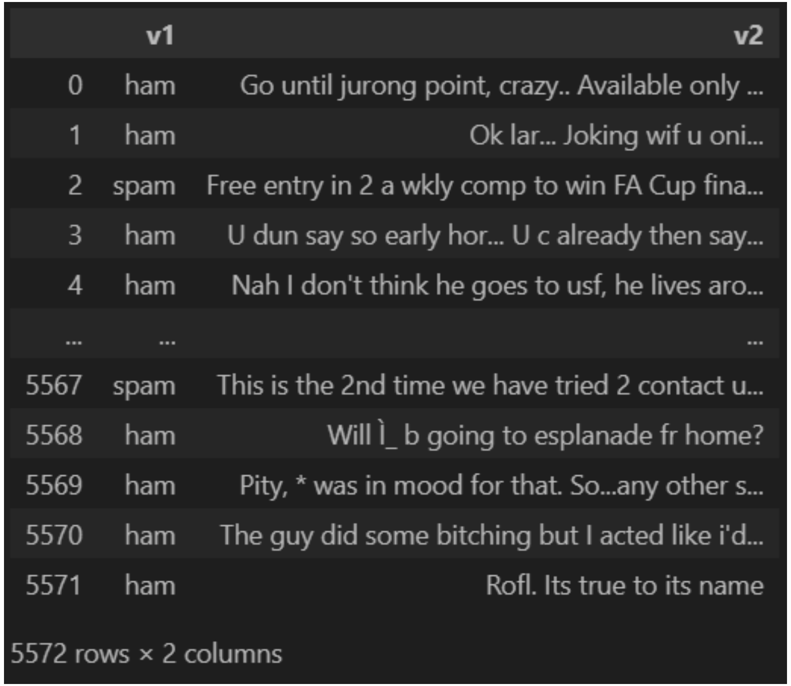


	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	ham	Go until jurong point, crazy.. Available only ...	NaN	NaN	NaN
1	ham	Ok lar... Joking wif u oni...	NaN	NaN	NaN
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	NaN	NaN	NaN
3	ham	U dun say so early hor... U c already then say...	NaN	NaN	NaN
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN
...	...	...	...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...	NaN	NaN	NaN
5568	ham	Will Ì_ b going to esplanade fr home?	NaN	NaN	NaN
5569	ham	Pity, * was in mood for that. So...any other s...	NaN	NaN	NaN
5570	ham	The guy did some bitching but I acted like i'd...	NaN	NaN	NaN
5571	ham	Rofl. Its true to its name	NaN	NaN	NaN

5572 rows × 5 columns

Overview

- In which we consider two columns: the label and the content of the message - which are the main contents used for analysis and evaluation of the model.



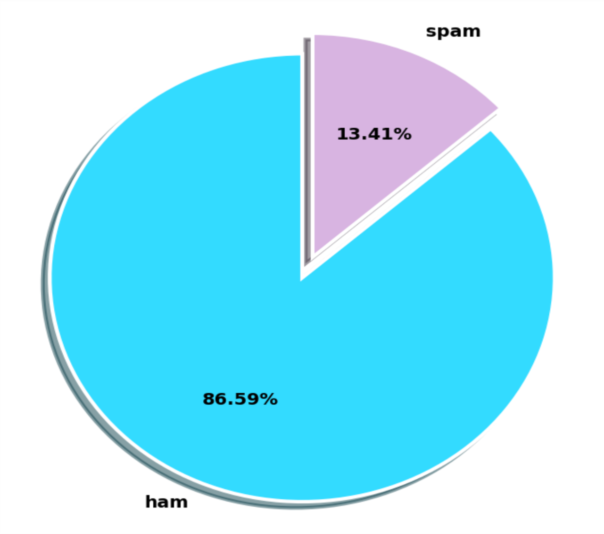
	v1	v2
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...
...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...
5568	ham	Will Ì_ b going to esplanade fr home?
5569	ham	Pity, * was in mood for that. So...any other s...
5570	ham	The guy did some bitching but I acted like i'd...
5571	ham	Rofl. Its true to its name

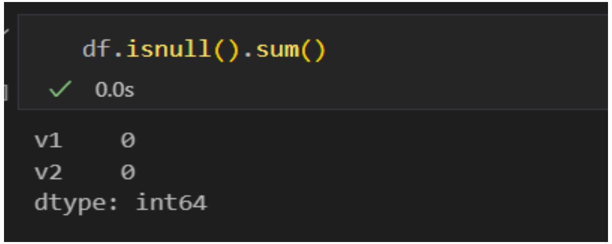
5572 rows × 2 columns

Overview

- Out of those, 747 emails (13.41%) are classified as spam and 4825 (86.59%) are classified as ham

- No missing value for v1 and v2 column





```
df.isnull().sum()
✓ 0.0s

v1    0
v2    0
dtype: int64
```

Preprocessing

- **Droping and Renaming Columns**
  - Keeping the two necessary columns from the original dataframe, v1 and v2, as the **label** for the message and **message** content, respectively
- **Cleaning message**
  - Convert all letters to lowercase for consistency
  - Remove all non-letter characters, including numbers and punctuation, to focus on word analysis
  - Remove excess leading and trailing spaces as well as extra spaces between words in the message

Preprocessing

- **Removing duplicate data**
  - Using the **drop_duplicates** method to remove duplicate data rows, keeping only the first copy of each duplicate data row, thereby reducing the possibility of model distortion due to duplicate data.
- **Removing stopwords**
  - Stopwords are words that do not have much meaning and are often removed during natural language processing.
  - Each message is divided into a list of words and only words that are not in the stopwords list are kept.
- **Label encoding**
  - Use LaberEncoder to convert labels from text to numbers, which is necessary to perform classification because most machine learning models require numeric input.

Data spliting

- Spliting the dataset into and test sets, where 30% of the data is reserved for testing
- Sampling 10% of the training data to reduce computation time or to work with a smaller subset for quick prototyping
- Spliting the sample data by label. The ‘message’ columns for rows with a spam label (1) and a ham (non-spam) label (0) is split into lists of words (tokens), and then converted to a list of lists, representing transactions

Data spliting

- Generating unique items and creating a binary matrix for spam and ham transactions:
  - A set of unique items is created from all of the spam transactions. This set is sorted to maintain a consistent order
  - A binary matrix is formed by going through each transaction and marking ‘1’ if the unique item is present in that transaction, else ‘0’. This is sorted in a DataFrame df_te_spam with columns representing unique items

Table of contents

01

Introduction

02

Dataset

03

Methodology

04

Result & Analysis

05

Conclusion

03

Methodology

What is FP Growth?

- It is an efficient algorithm for mining the complete set of frequent itemsets without candidate generation, making it faster than algorithms like Apriori. It achieves this efficiency by adopting a compact structure called the FP-tree (Frequent Pattern tree) which retains the itemset association information.

Input and Output of FP Growth

- **Input**: A dataset comprising transactions and minimum support threshold.
- **Output**: A complete set of frequent itemsets that satisfied the minimum support threshold

What is Association Rules Mining?

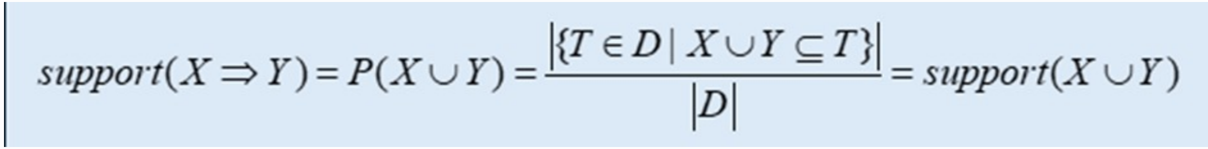
- **Association Rules Mining** is a rule-based machine learning method for discovering interesting relations between variables in large datasets. An example of an association rule in a retail setting could be {Diapers} => {Beer}, which suggests that customers who purchase diapers are also likely to purchase beer.

Input and Output of Association Rules Mining

- **Input**: A complete set of frequent itemsets and minimum confidence threshold
- **Output**: Association rules that satisfy the minimum confidence threshold

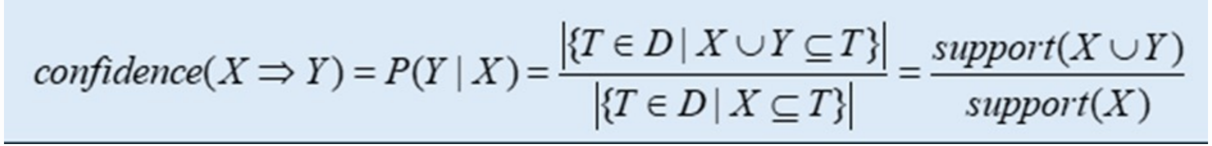
Related formula in Association Rules Mining

- **Support:** Frequency (Probability) of the entire rule with respect to database



$$support(X \Rightarrow Y) = P(X \cup Y) = \frac{\left|\{T \in D \mid X \cup Y \subseteq T\}\right|}{|D|} = support(X \cup Y)$$

- **Confidence:** Indicates the strength of implication in the rule



$$confidence(X \Rightarrow Y) = P(Y \mid X) = \frac{\left|\{T \in D \mid X \cup Y \subseteq T\}\right|}{\left|\{T \in D \mid X \subseteq T\}\right|} = \frac{support(X \cup Y)}{support(X)}$$

Process of algorithm

- **Data representation**
    - The data is represented in the form of transactions and corresponding items
    - Each message is split into individual words, and each transaction contains a set of words appearing in each message

Process of algorithm

- **Finding frequent patterns**
    - FP Growth algorithm is used to find frequent patterns in the data, for both the spam and ham groups
    - This help identify sets of words that frequently appear together in the messages

Process of algorithm

- **Generating association rules**
  - Based on the frequent patterns found, association rules are created to describe the relationships between words
  - These rules consist of a set of condition (antecedents) and a consequent, representing the relation ship between words in messages

Process of algorithm

- **Classifying new messages**
  - Based on the created rules, each message in the test set is classified by comparing the words in the message with the conditions in the association rules
  - A message is classified as spam if it contains the conditions of spam association rules with higher confidence compared to ham association rules, and vice versa

Table of contents

01

Introduction

02

Dataset

03

Methodology

04

Result & Analysis

05

Conclusion

04

Result & Analysis

Classification report

Method	Accuracy	Precision	Recall	F1-Score	CEL
FP Growth & Association Rules	0.906	0.78	0.82	0.79	0.6967
Logistic Regression	0.972	0.98	0.89	0.93	0.0913
Multinomial Naive Bayes	0.964	0.90	0.93	0.92	0.1526

Overview

- Logistic Regression and Multinomial Naive Bayes are popular algorithms for classification, and they work well on both discrete and continuous data.

- In essence, the FP-Growth algorithm and Association Rules (AR) are not typically used for classification problems. They still have characteristics appropriate for email classification, but they also have limitations compared to other classification algorithms.

Advantages of FP Growth and AR

- **Ability to handle large dataset**: FP-Growth can efficiently process large datasets to find frequent itemsets

- **Generating association rules from these frequent itemsets about the co-occurrence of words in emails**: These rules can be used to infer the likelihood of an email being spam or ham.

Disadvantages of FP Growth and AR

- **Poor performance in handling imbalanced data**: The spam email dataset often has an imbalance between classes, with spam emails usually being much fewer than non-spam emails. Frequent pattern growth combined with association rules can struggle with handling this imbalance, leading to poor classification performance.

Disadvantages of FP Growth and AR

- **Need to tweak the parameters**: To determine which itemsets or rules to choose, support and confidence thresholds need to be established first. The choice of those is not always obvious and can significantly affect the mining results.

- **Large Number of Rules and Difficult to Manage**: When data is large and complex, the number of frequent itemsets and association rules can become very large and difficult to control. This makes it difficult to find rules that have real value in classification task.

Table of contents

01

Introduction

02

Dataset

03

Methodology

04

Result & Analysis

05

Conclusion

05

Conclusion

Conclusion

Although Frequent Pattern Growth and Association Rules is not a common method in the field of classification in general, they still shows relative effectiveness when it comes to classification tasks, particularly in this case of text processing and feature extraction.

Challenges and Limitations

- Lack of training on multiple datasets to capture the diversity in semantics and context of words and sentences.

- Not flexible, with lower classification accuracy when encountering new, unseen data.

Suggested Development Directions

- Implement more effective data preprocessing steps, possibly deriving some new features from the existing ones to enhance data analysis.
- Train on multiple datasets to increase the model's learning capability.
- Develop a classification model for other languages, such as Vietnamese.

Thanks for listening!