JSS Science & Technology University

SPAM MESSAGE DETECTION

Presented by

ADITHYA DEEPTHI KUMAR (01JST21CS005)
KUSHAL NUNNA (01JST21CS063)
ULLEKH PUTTASWAMY (01JCE21CS118)
SURENDRA A (01JST21CS153)

Guided by

Dr.R GURU Ph.D
Associate Professor
Dept of Computer Science & Engineering,
SJCE,JSS STU,Mysuru

Submitted for the partial fulfillment of
BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

Content

1. Introduction

In the digital age, spam messages disrupt communication and threaten privacy and security. This project aims to detect and mitigate spam messages through a structured approach. By collecting and preprocessing SMS datasets, extracting and selecting relevant features, and developing and deploying robust machine learning models, we strive to enhance user experience and security. The project concludes with comprehensive documentation and reporting, summarizing findings and providing recommendations. Our ultimate goal is to create a safer digital ecosystem and contribute to a more positive online experience for all users.

1.1 Objectives

1.Develop Accurate Model:

- Classify SMS messages in real-time
- High precision and recall in spam detection

2.Generalization:

- Handle new, unseen messages effectively
- Robust against variations in content and style

3.Integration:

- Seamless incorporation into communication platforms via mobile app
- Ethical considerations to ensure minimal false positives/negatives
- Foster user trust and privacy

2.Literarture Review

Sl.no	Authors	Title	Findings
1.	Nikhil Kumar, Sanket Sonowal, Nishant(2020)	Email Spam Detection Using Machine Learning Algorithms	The literature review finds that various machine learning algorithms, including Naïve Bayes, support vector machines, and ensemble methods, show promise for effective email spam detection, with a need for continuous refinement to improve performance and adaptability.
2.	"Thashina Sultana, KA Sapnaz, Fathima Sana, Jamedar Najath(2020)"	Email based Spam Detection	"It highlights the effectiveness of using machine learning techniques, particularly Naïve Bayes Classifier, for spam detection, emphasizing the need for continual advancements to protect email users and organizations from the detrimental effects of spam emails."

3. Design of the Project

Data Acquisition and EDA

Dataset Selection:

- Use the UCI Machine Learning repository's SMS Spam Collection and user-contributed data.

Data Analysis:

- Clean and manipulate raw SMS data.
- Extract features that differentiate spam from legitimate messages.

Data Exploration:

- Use histograms to analyze message lengths.
- Create word clouds to identify common words in spam and non-spam messages

3. Design of the Project

Data Preprocessing

Cleaning:

- Remove irrelevant characters, punctuation, extra spaces, and special symbols.

Normalization:

- Convert text to lowercase.

Tokenization:

- Split messages into individual words or tokens.

Stop Word Removal:

- Remove common words that don't help in spam identification.

Stemming/Lemmatization:

- Reduce words to their root form

3. Design of the Project

Feature Engineering

Bag-of-Words (BoW):

- Represent messages as frequency vectors.

TF-IDF:

- Adjust BoW by considering word importance based on frequency and rarity.

Model Selection

Naive Bayes:

- Suitable for text classification, assuming feature independence.

3. Design of the Project

Model Training and Evaluation

Split Data:

- Divide data into training (70-80%) and testing (20-30%) sets.

Train Model:

- Train with the training set and optimize using hyperparameter tuning.

Evaluation:

- Assess performance using metrics like accuracy and precision

3. Design of the Project

Deployment Design (Website & App)

Website/App Interface:

- Design a user-friendly interface for SMS input.

API Integration:

- Use a web service API to send SMS text for analysis.

Spam/Ham Classification:

- Perform TF-IDF conversion and classify using the trained model.

Result Display:

- Show classification results to the user (spam/ham)

4.Implementation

Tech Stack

Programming languages:

- Python: Data manipulation, analysis, and machine learning
- Dart: Web, server, and mobile app development.

Libraries and Modules:

- Pandas: Data manipulation and analysis.
- NumPy: Numerical operations on arrays and matrices.
- Matplotlib: Visualizations (pie charts, histograms).
- Seaborn: High-level statistical graphics.
- NLTK: Text preprocessing (tokenization, stemming).
- WordCloud: Visualizing frequent words.

4.Implementation

- Scikit-learn (sklearn):

  1. LabelEncoder: Encoding text labels to numerical form.

  2.train_test_split: Splitting dataset into training and testing sets.

  3.CountVectorizer, TfidfVectorizer: Converting text data into feature vectors.

  4.GaussianNB,MultinomialNB,BernoulliNB:Naive Bayes algorithms for classification.

  5.accuracy_score,confusion_matrix, precision_score: Evaluating model performance.

- Pickle: Saving and re-using models.

4.Implementation

Loading the Dataset:

```
df = pd.read_csv('spam.csv', encoding='latin1')
```

Random dataset sample

```
df.sample(5)
```

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
6660	spam	new medz how t cornel o save on your medlcatio...	NaN	NaN	NaN
516	spam	Boltblue tones for 150p Reply POLY# or MONO# e...	NaN	NaN	NaN
4432	ham	Can u look 4 me in da lib i got stuff havent f...	NaN	NaN	NaN
4599	ham	Hi did u decide wot 2 get 4 his bday if not il...	NaN	NaN	NaN
549	ham	Ok give me 5 minutes I think I see her. BTW yo...	NaN	NaN	NaN

Shape of the Dataset

```
df.shape
```

```
(7515, 5)
```

Importing the data set

4.Implementation

Data Cleaning

```
df.drop(columns=['Unnamed: 2','Unnamed: 3','Unnamed: 4'],inplace=True)
```

```
df.sample(5)
```

	v1	v2
4881	ham	As usual u can call me ard 10 smth.
5708	spam	free for business or personal cwfqt start a bu...
1447	ham	Donâ□°Ã□Ã·t give a flying monkeys wot they thi...
5407	ham	Yup he msg me: is tat yijue? Then i tot it's m...
7091	spam	greetings you are receiving this letter becaus...

Excluding last three columns

4.Implementation

```
from sklearn.preprocessing import LabelEncoder
encoder = LabelEncoder()
```

```
df['target'] = encoder.fit_transform(df['target'])
```

```
df.head()
```

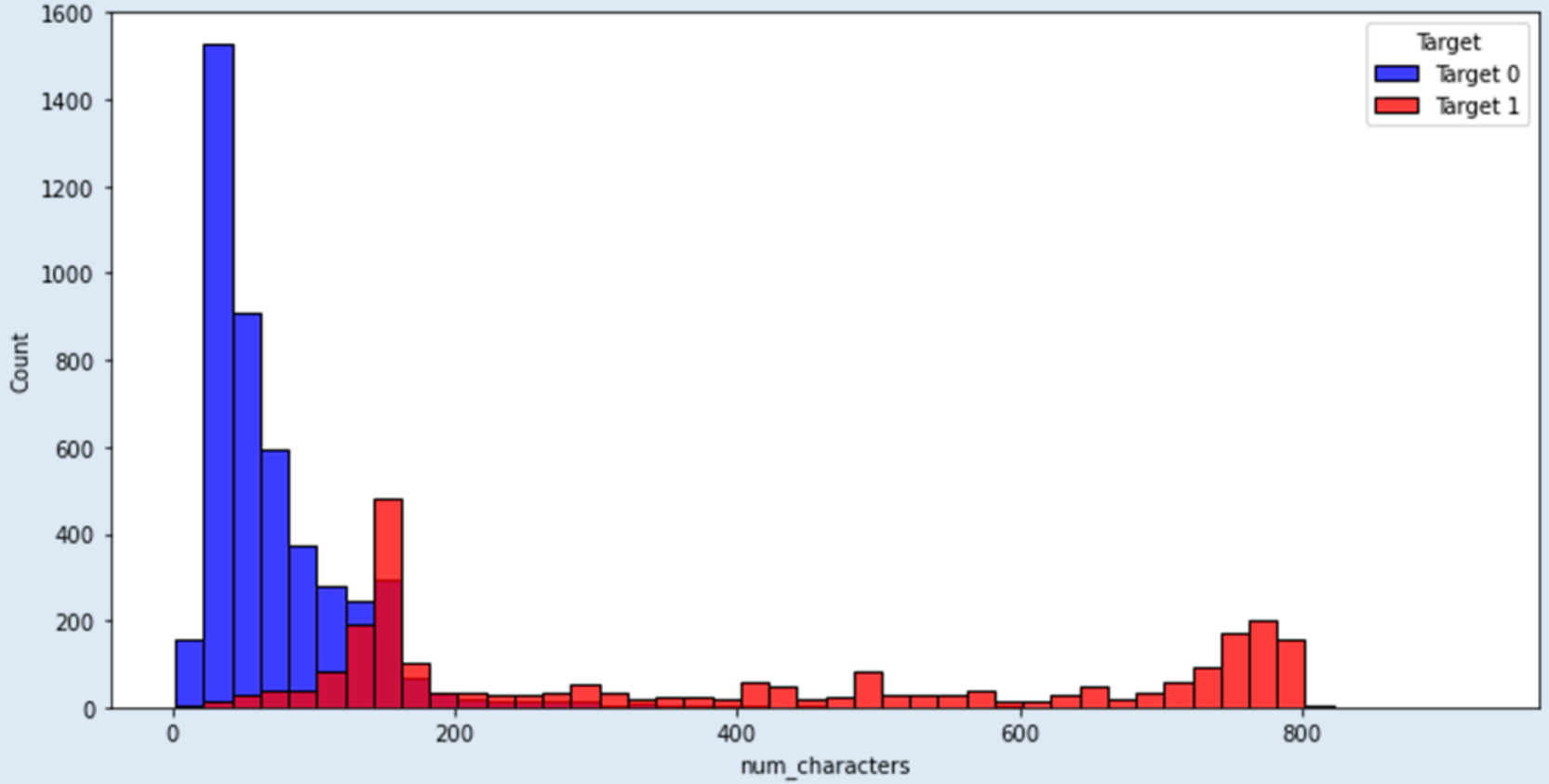
	target	text
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

SPAM:1 ,HAM:0

Assigning binary values to spam and ham messages

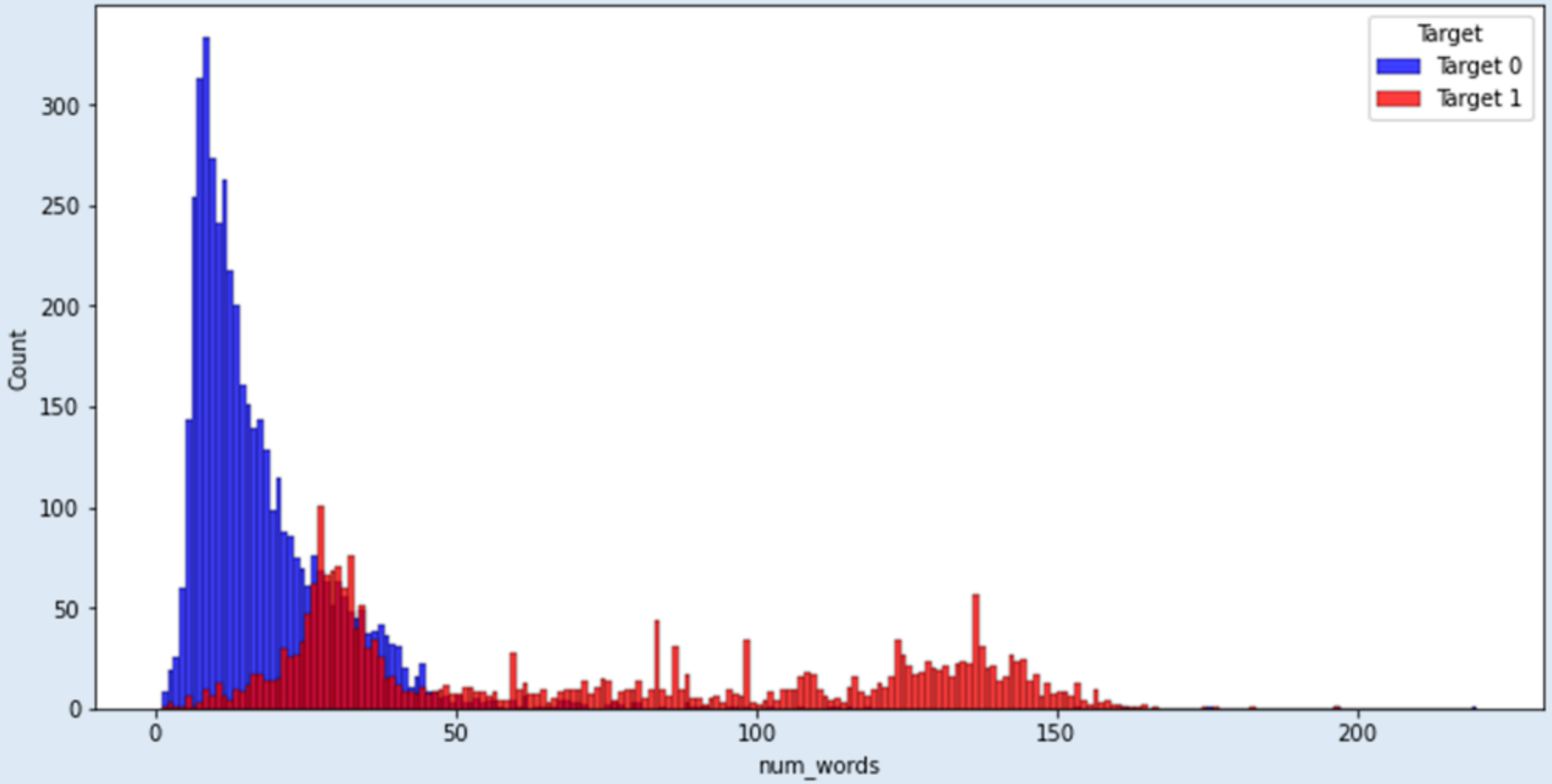
4.Implementation

Exploratory Data Analysis



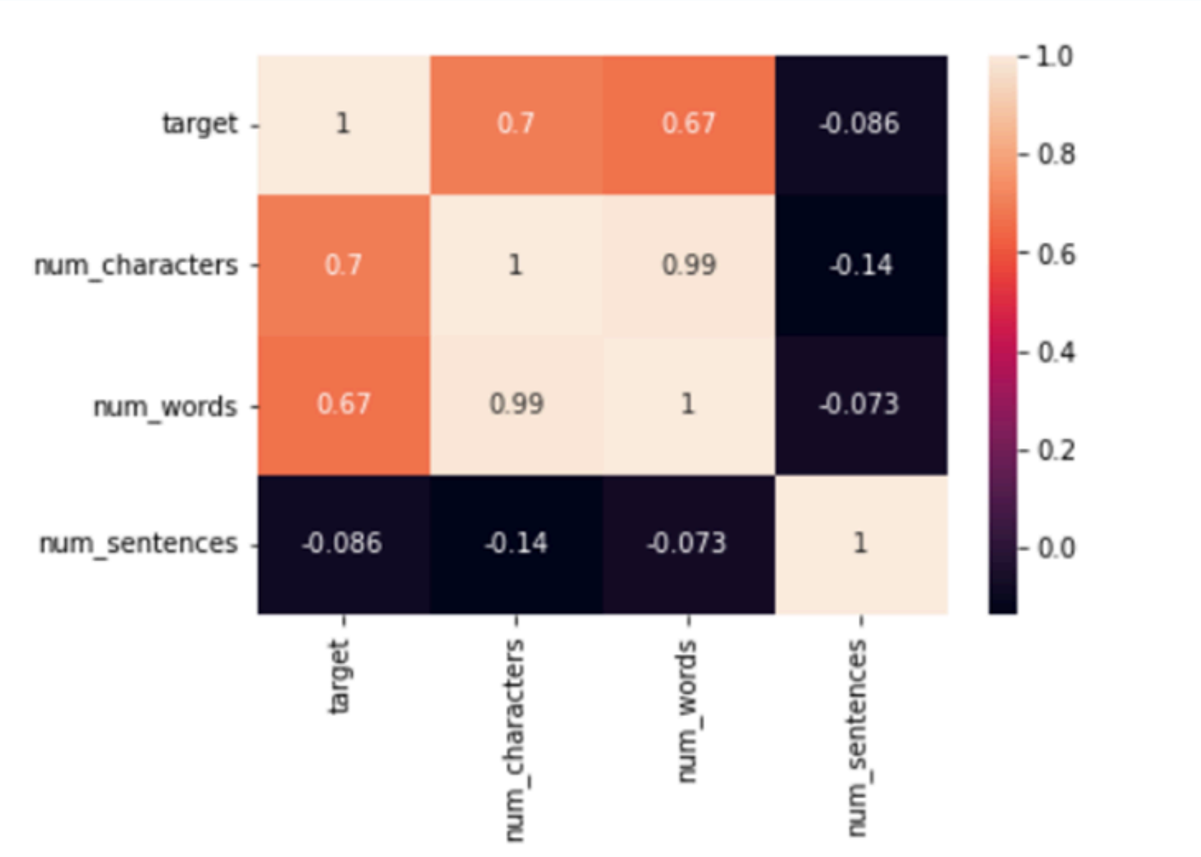
Box Plot (count vs characters)

4.Implementation



Box Plot (count vs words)

4.Implementation



Correlation between features and labels

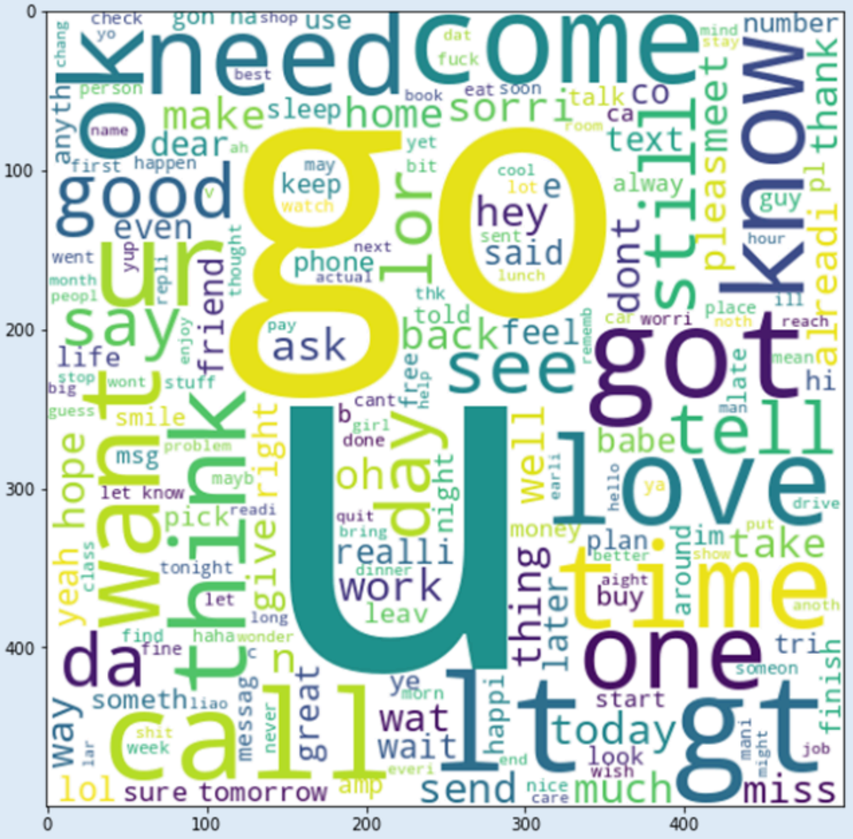
4.Implementation

Data Processing

- Convert text to lower case.
- Tokenize the text.
- Remove special characters.
- Eliminate stop words and punctuation.
- Apply stemming to words.

4.Implementation

Word Cloud for Spam messages



Word Cloud for Ham messages

4.Implementation

Model Training

```
from sklearn.feature_extraction.text import CountVectorizer,TfidfVectorizer
cv = CountVectorizer()
tfidf = TfidfVectorizer(max_features=3000)
```

```
X = tfidf.fit_transform(df['transformed_text']).toarray()
```

```
X.shape
```

```
(7099, 3000)
```

```
y = df['target'].values
```

Train:80%, Test Set :20%

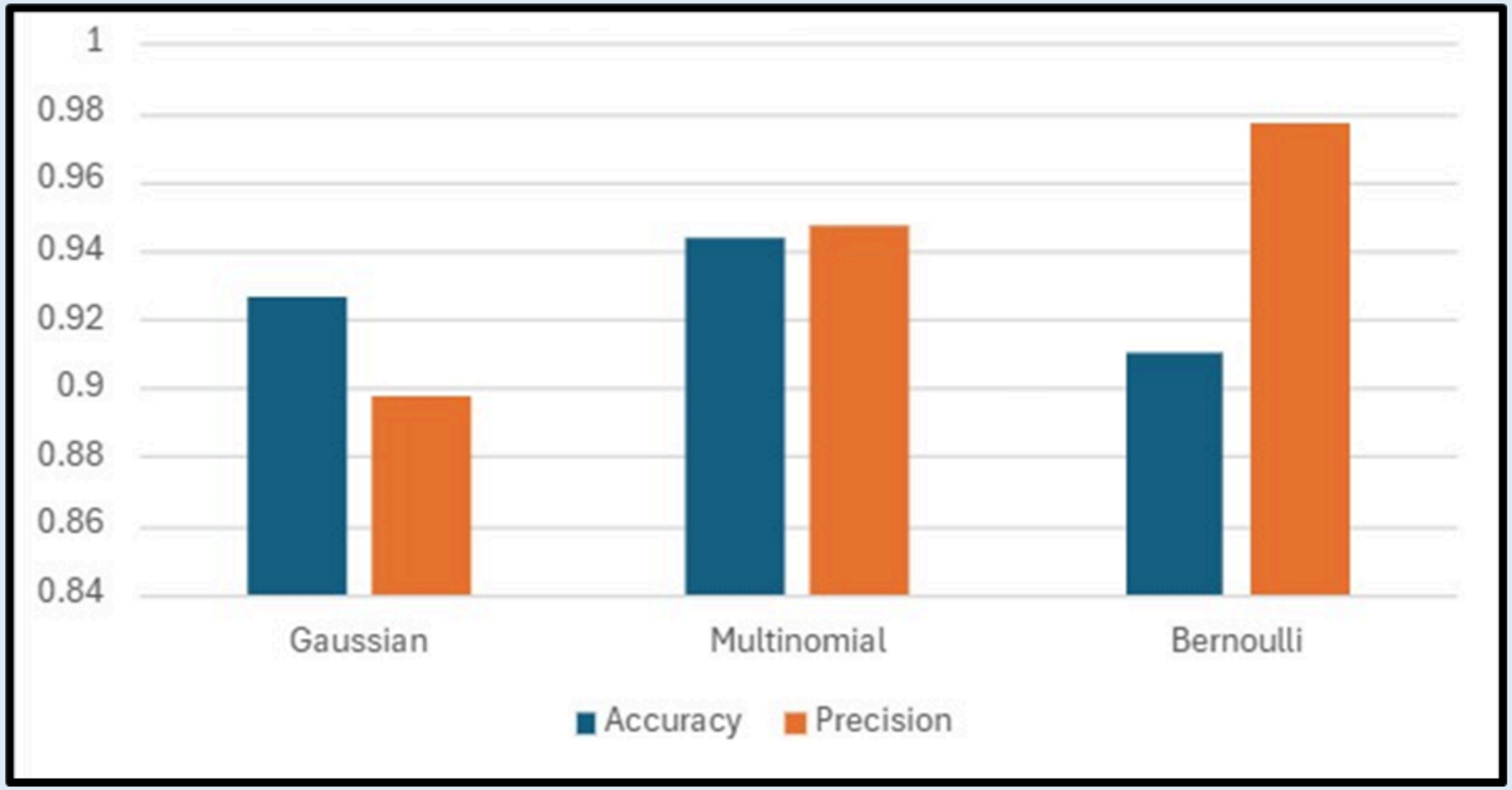
```
from sklearn.model_selection import train_test_split
```

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=2)
```

Tfidf Vectorization

4.Implementation

Model Testing and Selection



Comparative Study of Naive Bayes Algorithms

4.Implementation

The Multinomial Naive Bayes model's combination of high accuracy and competitive precision makes it the best choice for SMS spam detection, ensuring a robust and reliable classification system.

Multinomial Naive Bayes with tfidf vectorization is the best model.

```
import pickle
pickle.dump(tfidf,open('vectorizer.pkl','wb'))
pickle.dump(mnb,open('model.pkl','wb'))
```

Saving the Multinomial Naive Bayes Model

5. Test Cases and Results

Model Performance

- Accuracy Score: 0.9437, indicating robust and reliable spam classification.
- Confusion Matrix: The confusion matrix provides a more detailed insight into the accuracy of the model:

  True Positive (TP): 895 (Correctly predicted spam)

  True Negative (TN): 445 (Correctly predicted ham)

  False Positive (FP): 25 (Incorrectly predicted as spam)

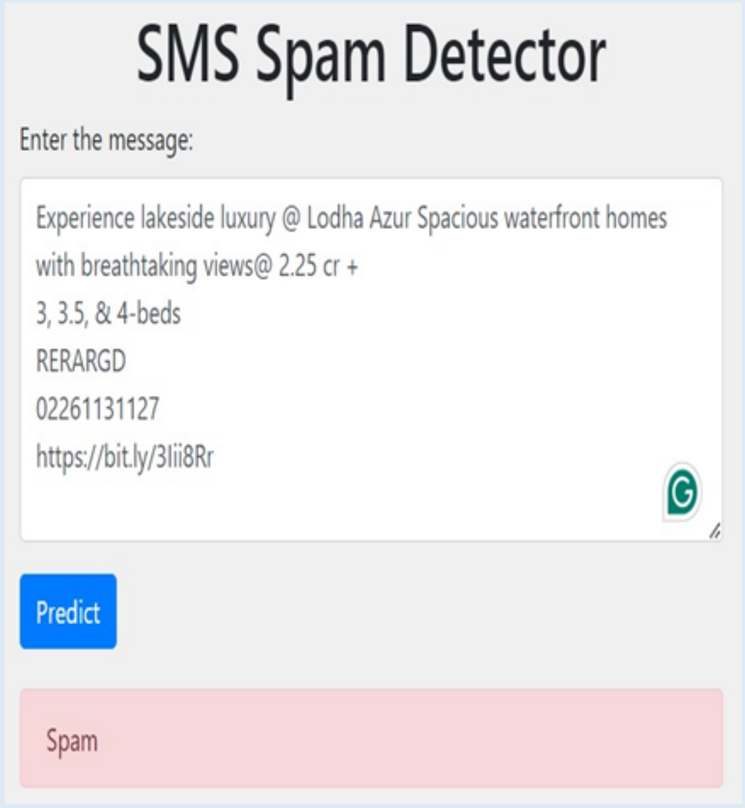
  False Negative (FN): 55 (Incorrectly predicted as ham)
- Precision Score: 94.68%, indicating high reliability in accurately identifying spam while minimizing false positives

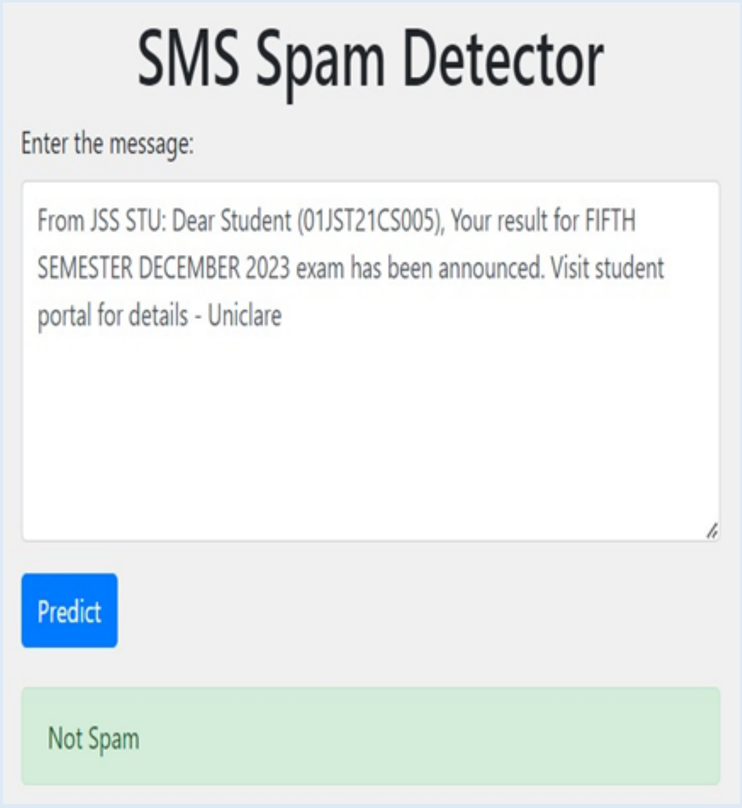
5. Test Cases and Results

Deployment

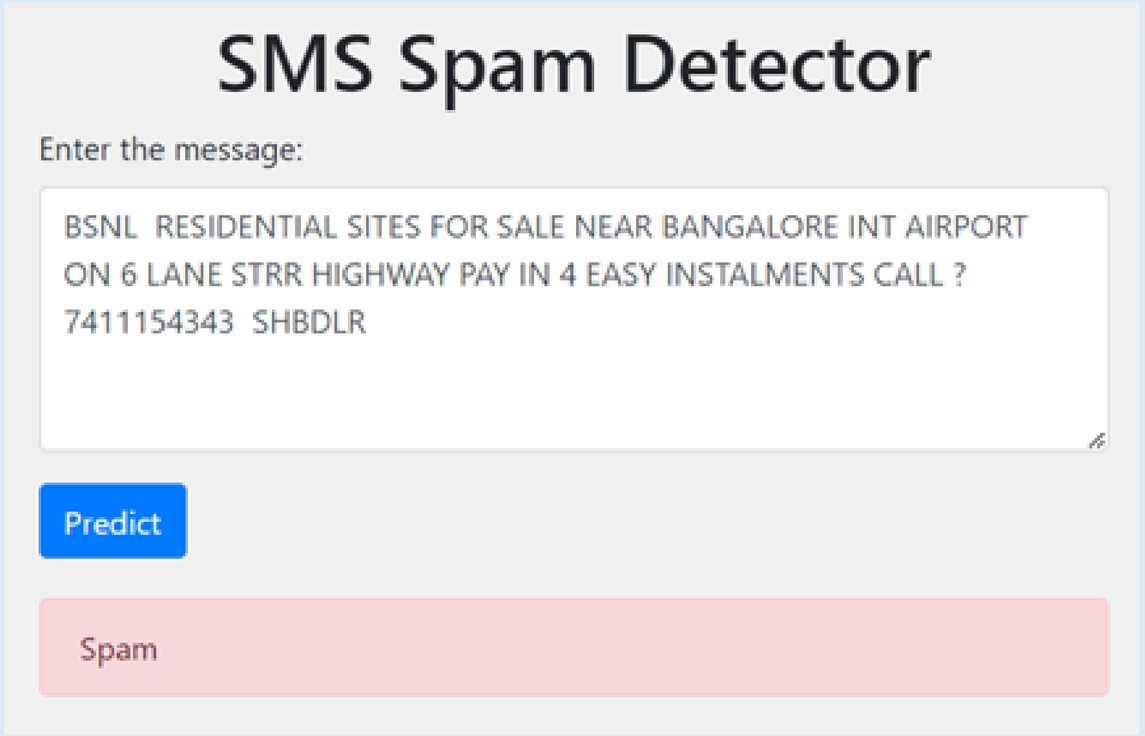
- Web Application: Enables users to input SMS messages and receive instant, user-friendly classifications without needing technical knowledge.

:

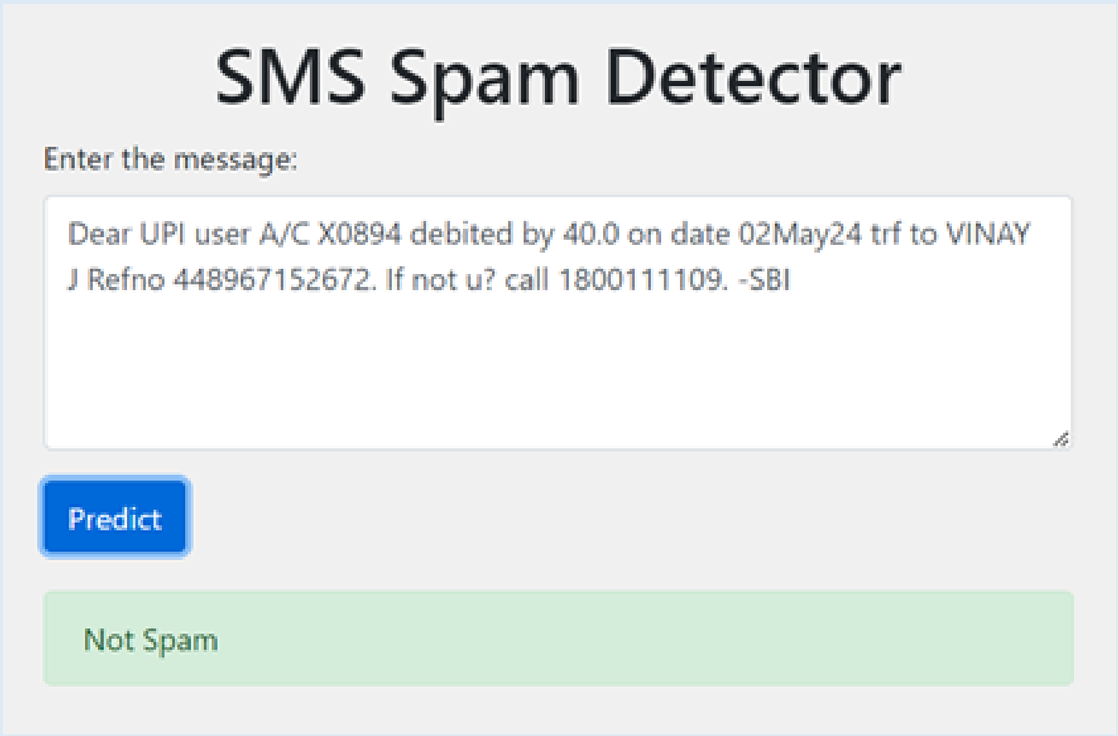




5. Test Cases and Results



Sample Spam message detection



Sample Ham message detection

5. Test Cases and Results

- Mobile App: Developed with Dart and Flutter for Android,offering real-time SMS analysis and immediate notifications on message nature.

Mobile App screenshot

6. Future Enhancements

- Ongoing Monitoring: Continuously refine the model to maintain its effectiveness in combating spam.
- Language and Cultural Variations: Investigate methods to enhance model effectiveness across different languages and cultural contexts.
- Robustness Against Attacks: Enhance model's defenses against adversarial attacks and deliberate spam crafting.
- Deep Learning Integration: Explore deep learning and adversarial learning to improve spam detection accuracy and adapt to evolving spamming tactics.

7.Conclusion

The "Spam Message Detection" project addresses the pervasive issue of unsolicited SMS messages in today's digital age. While mobile communication has enabled seamless information exchange, it has also introduced security risks through spam messages. Leveraging machine learning, the project aims to automatically identify and filter spam, safeguarding users from scams and phishing attempts.

Through meticulous methodology, including data preprocessing and model deployment, the project enhances the security and reliability of mobile communication platforms. By analyzing SMS content and patterns, the system discerns between genuine messages and spam, fostering a safer digital communication environment worldwide.

8. References

[1]Kumar, Nikhil & Sonowal, Sanket & Nishant,. (2020). Email Spam Detection Using Machine Learning Algorithms. 108-113. 10.1109/ICIRCA48905.2020.9183098.

[2]Machine learning for email spam filtering: review, approaches and open research problems Emmanuel Gbenga Dada, Joseph Stephen Bassi, Haruna Chiroma, Shafi'i Muhammad Abdulhamid, Adebayo Olusola Adetunmbi, Opeyemi Emmanuel Ajibuwa.

[3]Shirani-Mehr, H. (2013) —SMS Spam Detection using Machine Learning Approach.‖ p. 4.

[4]Abdulhamid, S. M. et al., (2017) —A Review on Mobile SMS Spam Filtering Techniques.‖ IEEE Access 5: 15650–15666.


thank you