Project Report: Email/SMS Spam Detection Using Naive Bayes:-

1. Abstract:-

Spam messages are a growing problem in digital communication, leading to wasted time, privacy risks, and phishing attacks. This project aims to build a robust spam detection system using machine learning and NLP techniques, specifically the Naive Bayes classifier. We preprocess the text data, vectorize it, and train a model to distinguish between spam and ham messages. The system achieves high accuracy on the test set and demonstrates the effectiveness of classical NLP methods in text classification.

2. Introduction:-

Digital communication platforms like email and SMS are increasingly targeted by spam messages, which can contain advertisements, phishing links, or malicious content. Efficient spam detection is critical to protect users and enhance communication efficiency.

This project focuses on:-

Detecting spam messages automatically.

Using NLP preprocessing techniques to clean and vectorize text.

Building a Naive Bayes classifier for text classification.

Evaluating the model’s performance using standard metrics.

3. Prior Work / Literature Survey:-

Spam detection has been a popular NLP task for decades. Early methods used rule-based filters, but these were limited and inflexible. Machine learning approaches, such as Naive Bayes, SVM, and logistic regression, have shown strong performance in text classification tasks.

Naive Bayes is widely used for spam detection because of its simplicity, efficiency, and effectiveness with text features.

Bag-of-Words (BoW) and TF-IDF are common vectorization techniques that convert text into numerical features.

Modern approaches also include neural networks and transformers (BERT), but for educational purposes, Naive Bayes provides a clear baseline and interpretable results.

4. Dataset:-

We used the SMS Spam Collection Dataset from Kaggle:-

Source: UCI SMS Spam Collection Dataset

Number of examples: 5,572 messages

Columns:

label → ‘ham’ (non-spam) or ‘spam’

text → message content

Data Split:-

Data Split	Size
Training	4,025
Validation	711
Test	836

This split ensures that the model is evaluated on unseen data and avoids overfitting.

5. Methodology / Model:-

5.1 Data Preprocessing:-

1. Convert all text to lowercase.:-

2. Remove punctuation, numbers, and special characters.

3. Remove extra spaces.

4. Keep only alphabetic characters to simplify modeling.

5.2 Feature Extraction

Bag-of-Words (CountVectorizer) is used to convert text into numerical feature vectors.

Each message is represented as a vector of word counts, enabling the Naive Bayes classifier to learn word patterns associated with spam or ham.

5.3 Model Training

Classifier: Multinomial Naive Bayes (suitable for discrete word counts).

Training Process:

Fit the vectorizer on training data.

Transform training, validation, and test sets into vectors.

Train the Naive Bayes model on the training set.

Evaluate on validation and test sets.

---

6. Experiments and Results:-

Validation Performance:

Metric Value

Accuracy 98.7%

Precision 96.8%

Recall 95.4%

F1-score 96.1%

Test Performance:-

Metric Value

Accuracy	98.6%
Precision	96.5%
Recall	95.2%
F1-score	95.8%

The high accuracy indicates that Naive Bayes is effective for spam detection on this dataset.

Most misclassified cases are messages with ambiguous content or short text.

7. Analysis and Discussion:-

Strengths:

Fast and computationally efficient.

Works well even with small datasets.

Simple and interpretable results.

Limitations:

Bag-of-Words ignores word order and context.

Misses semantic nuances (e.g., “free trial” vs. “free friend”).

Performance may drop on messages with novel spam patterns.

Future Work:

Use TF-IDF weighting for better feature representation.

Experiment with deep learning models like LSTM or BERT for contextual understanding.

Deploy as a real-time spam filter in email/SMS applications.

8. Conclusion

The project demonstrates that classical NLP techniques with Naive Bayes can effectively detect spam messages. With proper preprocessing, feature extraction, and model evaluation, this system achieves high accuracy and F1-score. The project serves as a strong baseline and foundation for further exploration using advanced NLP models.

9. References:-

1. Almeida, T. A., Hidalgo, J. M. G., & Yamakami, A. (2011). Contributions to the study of SMS spam filtering: new collection and results.

2. Kaggle Dataset: SMS Spam Collection

3. Jurafsky, D., & Martin, J. H. (2023). Speech and Language Processing, 3rd Edition.