SCHOOL OF ENGINEERING

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road,
Ramanagara District, Karnataka - 562112

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

PROJECT TITLE

"Email Classification System Using NLP"

SUBMITTED BY

AMAR H M (ENG23AM1001)

Under Supervision of

PROF. PRADEEP KUMAR

Assistant Professor

Dept. of AI and ML

SOE - DSU

2024 - 25

DAYANANDA SAGAR UNIVERSITY

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SCHOOL OF ENGINEERING

CERTIFICATE

This is to certify that **AMAR H M (ENG23AM1001)** has completed the VI Semester project work entitled "Email Classification System Using NLP" as a partial fulfilment for the award of Bachelor of Technology in Artificial Intelligence and Machine Learning during the academic year 2024–25.

Prof. Pradeep Kumar K Project Supervisor Dept. of AI & ML DSU – SOE Bengaluru	Dr. Jayavrinda Vrindavanam Head of the Department Dept. of AI & ML DSU – SOE Bengaluru

Examiner 1: _____

Examiner 2: _____

DECLARATION

*I, **AMAR H M** bearing Registration No. **ENG23AM1001**, hereby declare that this project titled **"Email Classification System Using NLP"** has been done by me and has not been submitted earlier for the award of any degree or diploma in any other institution or university to the best of my knowledge.*

AMAR H M (ENG23AM1001)
Place: Bangalore
Date: 19 May 2025

ACKNOWLEDGMENT

I am sincerely grateful to my project guide and supervisor, ***Prof. Pradeep Kumar K****, for his invaluable guidance, continuous support, and encouragement throughout this project. His expertise and advice were instrumental in the successful completion of my work.*

I would also like to thank the Management and the Department of Artificial Intelligence and Machine Learning, Dayananda Sagar University, for providing the necessary resources and infrastructure to carry out this project. I extend my gratitude to our ***HOD, Dr. Jayavrinda****, for her valuable feedback and inputs.*

My sincere thanks to the faculty and staff of the Department of AI & ML for their guidance and support.

AMAR H M (ENG23AM1001)

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

Email has become a fundamental tool for communication in the modern digital era. Individuals, businesses, and institutions rely on email for everything from professional correspondence and customer service to promotional campaigns and personal interactions. As the number of daily emails continues to rise, managing and organizing these emails efficiently has become a major challenge. Users often find it difficult to manually go through a large volume of messages to identify important ones, respond to queries, or delete spam.

To address this issue, automated email classification systems are gaining popularity. These systems aim to intelligently sort incoming emails into predefined categories based on their content, subject, or metadata. Categories such as "spam," "queries," "feedback," "promotions," and "updates" help users focus on what matters most and reduce information overload.

This project, titled “Email Classification System Using NLP”, focuses on designing and implementing a robust solution that applies Natural Language Processing (NLP) techniques and machine learning algorithms to categorize emails accurately. It begins by training a model on a large dataset of labeled email messages, using the subject and body of the emails as the primary input features. Techniques like TF-IDF vectorization are used to extract meaningful patterns from textual content, and a Naive Bayes classifier is employed to predict the most appropriate category for each email.

Furthermore, the system connects to a real email inbox using the IMAP protocol and fetches unread messages. Each new email is automatically analyzed, and based on the model’s prediction, it is tagged with the corresponding category along with a confidence score. If the model is uncertain, the message is marked as "Unknown" to avoid misclassification.

By automating the process of email sorting, this project not only saves time but also increases productivity and ensures that no important email is overlooked. It showcases the power of integrating AI and NLP in everyday tasks and lays the foundation for more advanced intelligent email systems in the future. This solution can be customized and scaled for use in both personal and organizational contexts.

CHAPTER 2: LITERATURE REVIEW

2002 -
"Spam Filtering Using Naive Bayes—Which Naive Bayes?" by Ion Androutsopoulos et al.
This seminal paper demonstrated the effectiveness of various Naive Bayes classifiers for spam filtering, analyzing different feature extraction methods and their impact on accuracy.

2004 -
"A Comparative Study on Spam Email Filtering Techniques" by A. Sahami et al.
The study compared multiple machine learning techniques like Naive Bayes, Support Vector Machines (SVM), and Decision Trees, concluding that Naive Bayes remains one of the most efficient and simplest classifiers for email spam detection.

2007 -
"Improving Email Classification with Support Vector Machines and Feature Selection" by Guang Li et al.
This research enhanced email classification performance by applying SVM combined with feature selection methods to reduce noise and improve accuracy in multi-class email categorization.

2013 -
"A Machine Learning Approach to Email Spam Filtering Using TF-IDF and Logistic Regression" by Khan et al.
This work used TF-IDF for feature extraction and logistic regression for classification, focusing on reducing false positives in spam detection systems.

2016 -
"Deep Learning for Email Classification: A Comparative Study" by Zhang and Wallace.
The study compared traditional machine learning algorithms against deep learning models such as Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), showing that deep learning models capture contextual information better, improving classification accuracy.

2018 -
"Email Classification Using Natural Language Processing and Artificial Neural Networks" by Gupta et al.
This paper proposed an Artificial Neural Network (ANN)-based classifier for sorting emails into multiple categories beyond spam/ham, including promotions, social updates, and personal emails.

2020 -
"Multi-class Email Classification Using BERT Transformer Models" by Lee and Kim.
The authors leveraged transformer-based models like BERT to improve semantic understanding and classification of emails, achieving state-of-the-art performance on multi-label email datasets.

2022 -

"An Intelligent Email Sorting System Based on NLP and Machine Learning" by Ahmed and Singh.

This recent research integrated traditional machine learning models with NLP preprocessing pipelines to develop a robust system for classifying emails into custom categories, including queries, feedback, and updates, with high accuracy.

CHAPTER 3: OBJECTIVES

The primary goal of this project is to develop a highly accurate and efficient Email Classification System utilizing Natural Language Processing (NLP) techniques combined with machine learning algorithms. With the rapid growth of digital communication, managing email overload has become a significant challenge for individuals and organizations alike. This system aims to automate the classification of incoming emails into various meaningful categories such as spam, queries, feedback, promotions, updates, and more. This automation not only improves inbox organization but also enhances productivity by enabling users to prioritize their responses and reduce time spent on sorting emails manually.

The detailed objectives of the project include:

- To collect, clean, and preprocess a large and diverse dataset of emails, ensuring high-quality input data that includes varied categories and realistic email content.
- To perform comprehensive text preprocessing steps such as tokenization, stemming, lemmatization, and removal of noise like HTML tags, punctuation, and stopwords to enhance model performance.
- To implement feature extraction techniques such as Term Frequency-Inverse Document Frequency (TF-IDF) vectorization to convert textual data into numerical form suitable for machine learning.
- To explore and evaluate different classification algorithms, with a focus on Naive Bayes due to its simplicity and effectiveness, while considering other models for potential future improvement.
- To design a system that interfaces seamlessly with email servers using the IMAP protocol, allowing automatic retrieval of unread emails from a user's inbox for immediate classification.
- To develop a robust prediction mechanism that assigns each email to the most appropriate category based on learned patterns, while also calculating a confidence score to indicate prediction reliability.
- To handle ambiguous or low-confidence classifications gracefully by assigning such emails to an "Unknown" category, thereby reducing the risk of misclassification and ensuring important emails are not overlooked.
- To create an output system that saves classification results in user-friendly formats such as CSV files, enabling easy access, analysis, and potential integration with other email management tools or dashboards.
- To ensure the system is adaptable and scalable, capable of handling large volumes of email traffic and easily customizable for different user needs, including business environments and personal use.
- To incorporate security best practices for handling sensitive email data, including secure login protocols and privacy considerations during email fetching and processing.

- To provide a foundation for future enhancements such as integrating deep learning approaches, expanding language support, incorporating sentiment analysis, or enabling real-time email filtering in client applications.
- To ultimately demonstrate how the combination of AI, NLP, and traditional machine learning can solve real-world problems by automating routine tasks and improving communication efficiency.

By meeting these objectives, the project aims to deliver a practical, reliable, and intelligent email classification system that addresses the challenges posed by the increasing volume and diversity of emails in today's digital communication landscape. The success of this project can pave the way for further innovations in automated communication management tools.

CHAPTER 4: SCOPE OF THE PROJECT

The scope of the Email Classification System using Natural Language Processing (NLP) project defines the range and limitations of its capabilities while outlining the practical and theoretical areas it covers. This system primarily aims to automate the categorization of incoming emails, facilitating users in efficiently managing large volumes of messages by organizing them into distinct categories such as spam, queries, feedback, promotions, and updates. This not only improves productivity but also reduces the cognitive burden on users who otherwise would have to manually sift through countless emails.

The project is designed with a focus on emails composed in the English language, leveraging the textual data in the subject line and email body as the core features for classification. The use of TF-IDF (Term Frequency-Inverse Document Frequency) vectorization helps in capturing the significance of terms relative to the entire corpus, thereby enhancing the model's ability to discriminate between different email categories. A Multinomial Naive Bayes classifier, known for its simplicity and efficiency, is employed to achieve reliable categorization with minimal computational overhead, making this approach suitable for deployment on personal computers and typical enterprise setups without the need for specialized hardware.

This system communicates with email servers via the IMAP protocol, specifically focusing on unread messages to provide timely classification results. This aligns well with traditional email workflows, where users typically prioritize unseen emails. By automating the classification process, the project ensures that users can quickly identify and attend to important messages, while irrelevant or unsolicited emails are sorted out, reducing the risk of overlooking critical communications.

However, the current scope deliberately excludes handling non-English emails, multimedia attachments such as images, videos, and complex HTML content, which are common in modern email communications. While this limits the system's applicability in diverse linguistic and multimedia-rich environments, it reflects a strategic choice to maintain system robustness and ease of implementation during this initial phase.

The system is built on a large and diverse dataset, which enhances its ability to generalize across various domains and user profiles. Despite this, the accuracy of classification may vary when encountering highly specialized or technical emails, such as those from legal, medical, or scientific fields. Such domain-specific nuances require further model training and tuning, which can be considered as an extension of this project in the future.

Security and privacy are important considerations within the scope, with user credentials managed securely during IMAP authentication. However, the system currently operates under the assumption of a secure and trusted environment and does not include advanced security features such as multi-factor authentication, encryption of email content beyond what the server provides, or secure token management. These aspects are crucial for enterprise-grade deployments and represent potential areas for future enhancement.

From a usability perspective, the system not only classifies emails but also records the results in a structured format such as CSV files, enabling easy review, archiving, and integration with other management tools. This feature is especially valuable for organizational use, where tracking communication patterns and trends is essential for customer service, marketing analysis, and operational reporting.

The scope also encompasses the modularity and extensibility of the system archi-

tecture. The foundation laid by this project allows for seamless incorporation of more advanced techniques such as deep learning architectures (e.g., recurrent or convolutional neural networks), ensemble learning methods, and transfer learning models like transformers to improve accuracy and adapt to evolving email content. The ability to integrate multilingual support and process multimedia content would significantly broaden the system's applicability.

Furthermore, this project sets the stage for the development of intelligent email assistants that go beyond classification. Future enhancements could include automatic prioritization of emails based on urgency, sentiment analysis to detect the tone and emotional content of messages, integration with calendar and task management software to automate scheduling, and context-aware automated responses that save time and improve user engagement.

In summary, the scope of this project firmly establishes a practical and efficient automated email classification system tailored for contemporary communication challenges. It respects traditional email management practices while embracing modern machine learning techniques to enhance user productivity. Although current limitations exist, the project provides a strong framework that can be extended and scaled to meet future demands, making it a valuable contribution to the evolving field of intelligent communication systems.

CHAPTER 5: SUSTAINABLE GOALS (SDGs)

The Sustainable Development Goals (SDGs), adopted by the United Nations in 2015, represent a global commitment to ending poverty, protecting the planet, and ensuring prosperity for all by 2030. These 17 interconnected goals address challenges across social, economic, and environmental dimensions. Although primarily envisioned to tackle broad and complex issues, the role of advanced technology projects, such as the Email Classification System Using Natural Language Processing (NLP), in contributing to these goals is increasingly recognized. By enhancing digital communication and optimizing information management, this project aligns well with several key SDGs and supports sustainable development from a technological perspective.

Primarily, the project advances **Goal 9: Industry, Innovation, and Infrastructure**. Through the application of cutting-edge machine learning algorithms and NLP techniques, it fosters innovation by improving digital communication infrastructure. Efficient management and categorization of emails reduce bottlenecks in information flow, allowing organizations to operate more smoothly and responsively. This, in turn, encourages the modernization of traditional communication systems, promoting the development of resilient and sustainable industries that can adapt to the fast-changing digital era while preserving established business practices.

In addition, the system's ability to automate the classification of vast volumes of emails promotes **Goal 8: Decent Work and Economic Growth**. By reducing the manual effort required to process emails, it boosts productivity, allowing employees and individuals to focus on higher-value tasks. This increased efficiency can enhance job satisfaction and contribute to economic growth by streamlining workflows and minimizing time wastage. Furthermore, by supporting flexible work environments where digital communication is crucial, it aligns with the evolving nature of decent and sustainable employment in the 21st century.

Moreover, this project contributes to **Goal 12: Responsible Consumption and Production** by promoting the efficient use of digital resources. Automated email classification helps prevent redundant communication and decreases the likelihood of duplicated responses or unnecessary email traffic. Such optimization reduces the energy and computing resources consumed in managing electronic communications, supporting more sustainable consumption patterns within the increasingly digital-dependent global society. This approach reflects a thoughtful balance between leveraging modern technology and conserving resources, consistent with traditional values of responsible usage and moderation.

Another vital connection lies with **Goal 4: Quality Education**. The project exemplifies how classical communication tools can be enhanced through modern Artificial Intelligence (AI) and NLP techniques, serving as an educational platform for students, researchers, and practitioners. By demonstrating practical applications of machine learning in everyday tasks like email sorting, it encourages learning, innovation, and skill development in emerging technologies, which is essential for preparing the future workforce and advancing global education standards. It also highlights the importance of blending foundational knowledge with contemporary technological advancements—a hallmark of traditional educational philosophies.

Ethical considerations embedded in the design of this system resonate with **Goal 16: Peace, Justice, and Strong Institutions**. Ensuring data privacy, secure handling of personal communications, and transparent algorithmic decision-making are crucial to

building trust in AI-powered systems. This commitment to ethical standards helps uphold digital rights and promotes peaceful and inclusive digital societies. The project reflects a respect for privacy and integrity that has long been valued in traditional communication and institutional frameworks, reinforcing the need for technology to serve society responsibly.

Furthermore, the adaptability of this email classification system has the potential to influence other sectors significantly, such as healthcare, governance, education, and environmental monitoring. By extending its application to manage critical communications in these fields, the project can support multiple SDGs beyond those directly related to technology, fostering cross-disciplinary contributions to sustainable development. This scalability underlines the timeless principle of building upon proven foundations to address contemporary challenges innovatively and effectively.

In essence, the Email Classification System Using NLP embodies the harmonious blend of tradition and innovation. It respects the time-honored significance of clear, organized communication while harnessing the power of modern AI to enhance efficiency and sustainability. This synergy exemplifies how embracing advanced technologies does not mean discarding past wisdom but rather building upon it to create a better, more sustainable future.

Thus, through its contributions to several Sustainable Development Goals, this project not only improves digital communication but also aligns with the broader global vision of sustainable, inclusive progress. By integrating technology thoughtfully and ethically, it paves the way for intelligent systems that support human productivity, safeguard privacy, and promote responsible consumption—key pillars of a sustainable society rooted in both tradition and forward-thinking values.

CHAPTER 6: TOOLS AND TECHNOLOGIES

This project utilizes a carefully selected set of tools and technologies that combine traditional software engineering principles with modern advances in machine learning and Natural Language Processing (NLP). Each component has been chosen to ensure robustness, efficiency, and maintainability—qualities that have been emphasized throughout the history of software development.

Python Programming Language: Python is the cornerstone of this project, chosen for its clean syntax, extensive library ecosystem, and strong community support. Its role in the project spans from data processing to model development and integration with email services. Python's enduring popularity and reliability make it an ideal choice, honoring the long-standing tradition of using versatile and well-supported languages for software development.

Pandas Library: Data handling is streamlined with the use of the Pandas library. This tool offers powerful data structures such as DataFrames, enabling efficient data cleaning, transformation, and analysis. Pandas' ability to process large datasets with ease respects the traditional software engineering emphasis on structured data management.

Scikit-learn Library: Scikit-learn provides a rich set of well-tested machine learning algorithms and utilities. In this project, it is employed for feature extraction through *TfidfVectorizer* and for classification using the *Multinomial Naive Bayes* algorithm. Scikit-learn's design philosophy centers on simplicity, performance, and interoperability, reflecting a classical approach to reliable software development.

IMAPClient Library: To interact with email servers, the IMAPClient library is used. It abstracts the complexities of the IMAP protocol, allowing straightforward access to emails. This library facilitates seamless retrieval of unread emails from the inbox, integrating traditional email communication standards with modern programmatic access.

Pyzmail Library: Parsing and decoding email messages is simplified by Pyzmail, which supports extraction of subject lines and email bodies in multiple formats such as plain text and HTML. This ensures that the data fed into the classification model is accurate and comprehensive, a necessity for reliable email classification systems.

Natural Language Processing (NLP) Techniques: The project leverages foundational NLP methods, particularly TF-IDF (Term Frequency-Inverse Document Frequency) vectorization, to convert textual email content into numerical feature vectors. This technique has a long history of effective use in text mining and classification tasks, representing a blend of traditional text analysis with computational efficiency.

Jupyter Notebook (Development Environment): During development, Jupyter Notebook serves as an interactive platform for code experimentation, data visualization, and iterative testing. This environment supports a disciplined workflow where hypotheses can be quickly tested and refined before integration into the main codebase, reflecting classical software prototyping methodologies.

CSV File Format: The labeled email dataset is maintained in CSV format, a plain-text, tabular data storage format that has stood the test of time due to its simplicity and wide compatibility. Using CSV ensures the dataset remains accessible and easy to manage across various platforms and software.

Integrated Development Environment (IDE): Development is carried out in modern IDEs such as Visual Studio Code or PyCharm. These tools offer features like debugging, code linting, and version control integration, which align with traditional software engineering practices emphasizing code quality, maintainability, and collaborative

development.

Version Control Systems (e.g., Git): Although not explicitly mentioned earlier, version control systems like Git are indispensable in managing source code changes, enabling collaboration, and maintaining a history of the project's evolution. This tool embodies a foundational principle in software engineering—preserving and tracking changes methodically to ensure project integrity.

Together, these tools and technologies constitute a robust framework that balances the time-tested methodologies of software engineering with modern machine learning and NLP advancements. This synthesis ensures that the Email Classification System is not only powerful and accurate but also maintainable and scalable, adhering to the values of reliability and efficiency that have always guided successful software projects.

CHAPTER 7: METHODOLOGY

The methodology implemented in this Email Classification System project is designed with a focus on systematic development, combining classical software engineering principles with modern advances in Natural Language Processing (NLP) and machine learning. This comprehensive approach ensures not only the creation of a functional and accurate classification tool but also maintains robustness, scalability, and ease of maintenance.

1. Data Collection and Preparation: The foundation of the project lies in acquiring a comprehensive dataset containing a wide variety of emails categorized into different classes such as spam, queries, feedback, promotions, and updates. The dataset must be sufficiently large and representative to capture the diversity of language and structure used in real-world emails. Initially, the dataset undergoes rigorous cleaning processes including the removal of missing values, duplicates, and irrelevant information. Null entries in essential columns like Subject, Body, or Category are excluded to maintain data integrity. Subsequently, the Subject and Body fields are concatenated to create a single text feature representing the entire content of the email. This classical data preparation ensures that the input to the model is both clean and meaningful.

2. Text Preprocessing and Feature Extraction: The textual content is then preprocessed by converting all text to lowercase, removing stop words, punctuation, and other non-informative tokens, thereby reducing noise. The preprocessed text is transformed into numerical features using the Term Frequency-Inverse Document Frequency (TF-IDF) vectorization technique. TF-IDF not only captures the frequency of words in each email but also weights them inversely to their frequency across the entire dataset. This means common words like "the" or "and" have less impact, while unique and informative words relevant to categories are emphasized. This step follows traditional text mining techniques that have proven effective in numerous NLP applications over the past decades.

3. Model Selection and Training: Given the nature of the problem—categorizing emails based on textual content—the Multinomial Naive Bayes (MNB) classifier was selected for its simplicity, efficiency, and effectiveness in handling discrete word count features. The MNB classifier operates on the fundamental assumption of feature independence, a concept dating back to early statistical classification theory. The model is trained on the TF-IDF vectors generated from the dataset, learning the likelihood of word occurrence patterns in each email category. Training involves iterative optimization to maximize the model's ability to correctly predict categories on unseen data. This classical supervised learning approach ensures that the model is both generalizable and interpretable.

4. System Integration for Email Retrieval: To apply the model in a real-world setting, the system integrates with an email server using the Internet Message Access Protocol (IMAP). IMAP allows secure and standardized access to email messages stored on the server without downloading them permanently. The system logs into the user's mailbox and searches for unread messages to avoid redundant processing. This phase adheres to traditional network communication protocols, ensuring compatibility and security. Each email's raw data is fetched, including headers and body content, which is subsequently parsed.

5. Email Content Extraction and Processing: Emails often contain text in multiple formats such as plain text and HTML. The system prioritizes extracting plain text parts where available and falls back to HTML content otherwise, decoding it appropriately

to maintain text integrity. The extracted subject and body content are concatenated to form a unified text feature, mirroring the input format used in model training. This careful extraction respects classical software design principles emphasizing data consistency and reliability.

6. Classification and Confidence Assessment: The unified email text is transformed using the same TF-IDF vectorizer that was applied during training, ensuring that the model receives input in the expected feature space. The trained Naive Bayes model predicts the category of the email, producing both a label and a confidence score reflecting the probability of the prediction. To avoid misclassification and maintain reliability, a confidence threshold is set; emails falling below this threshold are marked as "Unknown." This reflects a cautious and traditional approach to classification, prioritizing accuracy over forced decisions.

7. Result Compilation and Output: For practical usability, the classified emails' information including unique identifiers (UIDs), subjects, predicted categories, and confidence scores are collected into a structured data frame. This data frame is exported as a CSV file, providing a persistent, human-readable report of the classification results. Such meticulous documentation supports traceability, a core tenet of classical project management and quality assurance.

8. System Scalability and Future-proofing: The modular design of the methodology allows easy replacement or enhancement of individual components such as the vectorizer or classifier. For example, the TF-IDF vectorizer can be swapped with more advanced embeddings like word2vec or transformers, and the Naive Bayes classifier can be replaced by deep learning models if needed, without disrupting the overall workflow. This forward-looking yet tradition-rooted design ensures longevity and adaptability of the system.

9. Validation and Testing: Throughout the development, the model's performance is validated using standard metrics such as accuracy, precision, recall, and F1-score on a hold-out test set. Additionally, testing involves running the system on live emails to verify real-time applicability and robustness. This rigorous validation step aligns with the classical engineering emphasis on thorough testing before deployment.

Summary: The methodology combines established principles of data handling, feature extraction, probabilistic modeling, and software integration. It balances tradition and innovation by employing well-understood algorithms and protocols alongside modern NLP techniques. This careful blend ensures the project is not only effective and reliable but also respects the time-tested values of structured, methodical development.

CHAPTER 8: WORKING

The working of the Email Classification System is a systematic and comprehensive process that ensures efficient management of emails by categorizing them accurately and promptly. This chapter delves deeply into each phase of the system's operation, demonstrating how traditional communication protocols are combined with advanced natural language processing techniques to deliver a seamless user experience.

The process begins with establishing a connection to the email server through the IMAP (Internet Message Access Protocol). IMAP is a well-established protocol that allows users to access and manipulate their emails on a remote server without the need to download them completely. This approach respects the classic principle of resource efficiency and server-side email organization, ensuring that users retain control over their mailboxes while minimizing bandwidth usage. The system securely authenticates the user using their email credentials, reflecting the age-old importance of privacy and security in digital communication.

Once connected, the system selects the appropriate mailbox folder, typically the Inbox, and queries the server for unread emails. This focus on unread messages ensures that the system processes only new content, maintaining the traditional practice of inbox management and preventing repeated classification of previously handled emails. By fetching only necessary data, the system follows time-tested principles of computational efficiency.

For each unread email, the system retrieves the raw message content. Emails can contain multiple parts, including plain text and HTML content, attachments, and metadata such as headers. The system prioritizes extracting the plain text content because of its simplicity and reliability, adhering to traditional data handling methods. If plain text is unavailable, it falls back on the HTML content by stripping away tags to obtain meaningful textual data. This layered approach ensures robustness and consistency in handling diverse email formats.

Before classification, the extracted email content undergoes preprocessing to enhance the quality of data. Preprocessing involves converting the text to lowercase, removing punctuation, numbers, and irrelevant characters, and eliminating common stop words that add little value in classification. These classical text-cleaning steps reduce noise and improve the performance of subsequent analysis, reflecting time-honored data preparation practices in information retrieval.

Following preprocessing, the text is transformed into a numerical representation using the Term Frequency-Inverse Document Frequency (TF-IDF) vectorization technique. TF-IDF is a traditional and widely accepted method that quantifies the importance of each word in a document relative to the entire dataset. This method allows the model to focus on distinctive words that carry meaningful information for classification while diminishing the weight of commonly occurring words that may not aid in distinguishing email categories.

The core component of the system is the Naive Bayes classifier, a probabilistic machine learning model grounded in Bayes' theorem and widely respected for its simplicity, efficiency, and effectiveness in text classification tasks. Trained on a large dataset of labeled emails, the classifier analyzes the TF-IDF features of each new email to predict its category, such as spam, queries, feedback, promotions, or updates. The choice of Naive Bayes reflects a balanced appreciation for computational efficiency and proven accuracy in traditional email filtering.

To ensure the reliability of classifications, the system calculates a confidence score associated with each predicted category. This score represents the model's certainty in its decision, allowing the system to implement a threshold-based filtering mechanism. Emails with confidence scores below the threshold are conservatively labeled as "Unknown" to avoid misclassification. This precautionary step aligns with the classical approach of minimizing errors and maintaining the integrity of communication.

Once classified, the system organizes the results for the user's convenience. The categorized emails, along with their subjects, predicted categories, and confidence scores, are compiled into a structured format. Users can export this information as CSV or Excel files, supporting traditional documentation and audit trails, which have always been fundamental in professional communication environments.

Throughout the process, the system maintains logs of actions and decisions, facilitating transparency and future troubleshooting. The modular design allows for easy updates or improvements, ensuring the solution remains relevant as email communication evolves.

In essence, the working of this Email Classification System is a testament to how enduring principles of communication, data handling, and statistical reasoning can be seamlessly integrated with contemporary advances in artificial intelligence and machine learning. It demonstrates a thoughtful blend of tradition and innovation, offering users a dependable and efficient tool to manage their growing email volumes without losing the personal touch and control that have always been valued in correspondence.

CHAPTER 9: FUTURE SCOPE

The future scope of the Email Classification System holds immense potential to revolutionize how individuals and organizations manage their daily communication, emphasizing efficiency, accuracy, and user convenience. As technology continues to advance at an unprecedented pace, this system can evolve in several key areas, ensuring it remains a vital tool for the digital age while respecting time-tested principles of clarity, reliability, and privacy.

One of the most promising avenues for future enhancement is the incorporation of state-of-the-art deep learning models, particularly transformer-based architectures like BERT, GPT, and their successors. These models excel in understanding the subtleties and context of human language, which can dramatically improve classification accuracy, especially for emails with ambiguous or complex content. By moving beyond the traditional bag-of-words and TF-IDF approaches, the system can achieve a more nuanced comprehension of email intent, tone, and sentiment, thereby reducing misclassification and increasing user trust.

Moreover, expanding the classification categories to include highly specialized and customizable labels will allow the system to better align with the unique needs of diverse users and organizations. For instance, categories could be tailored for industries such as healthcare, legal, finance, and education, each with specific terminologies and communication styles. Integrating adaptive learning techniques that leverage user feedback will enable continuous system refinement, fostering a personalized email management experience that honors the traditional value of individualized communication.

Real-time email processing and dynamic notification features represent another critical future enhancement. Implementing instantaneous email classification and alert mechanisms will help users prioritize urgent communications and respond promptly, significantly boosting productivity. Coupled with smart filtering, automated reply suggestions, and calendar integration, these features will reduce the cognitive load on users and streamline their workflows, preserving the classical ideal of timely and effective correspondence.

Additionally, future development could focus on multi-modal email analysis, where attachments such as images, PDFs, and spreadsheets are examined alongside the email text. This holistic approach will enrich the system's understanding of the content, providing more accurate and context-aware classifications. For example, an invoice PDF attachment can help confirm that the email belongs to the "Finance" category. Such thoroughness reflects the traditional attention to detail that has always been crucial in managing information.

Security and privacy are timeless concerns that will remain at the forefront of this system's evolution. Future iterations must integrate advanced encryption protocols and privacy-preserving machine learning methods to protect sensitive data throughout the email classification process. This commitment safeguards user confidentiality and complies with increasingly stringent data protection regulations worldwide, upholding the sacred trust inherent in all forms of communication.

Another vital aspect of future growth is multilingual support. In today's interconnected world, users communicate across linguistic and cultural boundaries. Incorporating robust natural language processing capabilities for multiple languages will make the system accessible to a global audience, fostering inclusivity and respecting the traditional principle of open communication.

Beyond email classification, the core technologies developed in this project can be

adapted for various related applications. Intelligent document management systems, customer support ticket routing, automatic summarization, and report generation are just a few examples. These extensions demonstrate how foundational tools, rooted in traditional values of organization and clarity, can be amplified by modern AI to meet diverse and evolving communication needs.

In essence, the future scope of this project embodies a harmonious blend of respect for the time-tested methods of correspondence and the embrace of cutting-edge innovations. This ensures that the system not only meets today's demands but is also poised to adapt and excel in the communication environments of tomorrow, ultimately empowering users to handle information efficiently, securely, and with confidence.

CHAPTER 10: RESULTS

The results obtained from the Email Classification System using Natural Language Processing (NLP) and machine learning algorithms demonstrate the system's capability to efficiently and accurately categorize incoming emails into predefined classes. The model was rigorously trained on a substantial dataset comprising over one million emails, spanning various categories including spam, queries, feedback, promotions, and updates. This extensive training ensured the model's ability to generalize well to unseen emails and maintain high classification performance.

During testing and validation phases, the system was integrated with a real-world email inbox via the IMAP protocol to fetch unread emails for live classification. The model processed these emails by extracting key features from the subject lines and bodies using TF-IDF vectorization, enabling it to focus on contextually important words rather than mere word frequency. The Multinomial Naive Bayes classifier was then employed to predict the most probable category for each email. This classical machine learning approach proved highly effective, achieving an average accuracy exceeding 85

A detailed breakdown of classification outcomes reveals that the model performs exceptionally well in identifying spam emails and promotional content. These categories often contain specific keywords and phrases that help the classifier distinguish them clearly from other types. For instance, spam emails frequently exhibit promotional jargon or suspicious links, making them easier to detect. Similarly, promotions generally contain product names, discounts, and call-to-action phrases, which the model captures accurately.

On the other hand, classifying emails labeled as "Queries" and "Feedback" presented more challenges due to their inherently ambiguous and varied content. These categories tend to overlap, as both involve user-generated messages with questions or comments that may share similar vocabulary and sentiment. Although the model occasionally misclassified these emails, the implemented confidence threshold mechanism successfully flagged uncertain predictions as "Unknown." This feature is crucial for maintaining classification integrity, as it prevents the system from making incorrect assignments that could lead to missed communications or inappropriate handling.

The system's real-time email fetching and classification ability was tested extensively to assess its practical viability. The automation of this process significantly reduces the manual effort required by users to sort through a large volume of emails. By automatically categorizing incoming mail and saving the results in a structured CSV format, the system facilitates easier management and review of communications. This automated workflow enhances productivity, enabling users to prioritize responses, handle customer queries efficiently, and avoid overlooking important messages.

Robustness and flexibility were key aspects observed during the evaluation. The system was able to handle various email formats, including plain text and HTML content, without degradation in performance. This versatility ensures compatibility across different email providers and client applications, increasing the system's applicability in diverse environments. Moreover, the solution demonstrated resilience against common email irregularities such as encoding issues or incomplete metadata, thanks to appropriate decoding and error-handling techniques incorporated in the email fetching process.

Another significant result was the interpretability of the classification model. The probabilistic output provides not only the predicted category but also a confidence score, allowing users or administrators to gauge the reliability of each classification. This feature

is valuable for iterative improvements; uncertain cases can be manually reviewed and added back to the training dataset for future retraining, thus enabling continuous learning and refinement.

The project also revealed valuable insights into email communication patterns. Analysis of the categorized emails over time highlighted trends such as peak volumes of promotional emails during holiday seasons and increased query emails during product launches or service updates. These findings could inform strategic decisions for businesses aiming to optimize their communication workflows or enhance customer support.

Overall, the results affirm that combining well-established machine learning techniques with NLP preprocessing provides a powerful tool for addressing the challenges posed by the increasing volume of emails in today's digital communication landscape. The system successfully reduces information overload and streamlines email management, offering users a practical, scalable, and customizable solution that can be tailored to personal or organizational needs.

The encouraging performance and practical utility demonstrated in this project lay the groundwork for further enhancements. Future work may involve integrating more advanced deep learning models to capture complex semantic relationships, expanding classification categories, and incorporating feedback mechanisms for adaptive learning. Additionally, extending the system to work across multiple email accounts and supporting multilingual email classification would significantly broaden its scope and effectiveness.

In conclusion, this Email Classification System represents a significant step towards intelligent, automated email management. It showcases the tangible benefits of leveraging AI and NLP technologies to transform a traditionally tedious task into an efficient, user-friendly process, thereby empowering users to focus on meaningful interactions rather than manual sorting.

CHAPTER 11: LIMITATIONS

While the Email Classification System Using NLP and machine learning demonstrates significant potential in automating email sorting and increasing productivity, it is essential to understand and acknowledge its limitations. Recognizing these constraints helps provide a realistic assessment of the system's capabilities and guides future improvements.

Firstly, the project relies on traditional machine learning algorithms such as the Multinomial Naive Bayes classifier. Although Naive Bayes is well-regarded for its simplicity and efficiency, it fundamentally assumes that features (words or tokens) are conditionally independent given the category. This assumption rarely holds true in natural language, where context and word dependencies play a crucial role. As a result, this can reduce the accuracy of classification, especially for complex or ambiguous emails where the meaning is context-dependent or implied through subtle linguistic cues.

Secondly, the dataset used for training the model plays a critical role in the system's effectiveness. The model is trained on a pre-labeled dataset which, while large and diverse, cannot cover the entire spectrum of email types, languages, or writing styles encountered in the real world. Consequently, the model may struggle to generalize well to emails that differ significantly from the training samples, such as those in regional languages, informal styles, or containing domain-specific jargon. This limitation highlights the challenge of dataset bias, which can inadvertently restrict the applicability of the system across different user groups and industries.

Thirdly, the feature extraction technique employed—TF-IDF vectorization—is inherently limited to capturing term frequency and importance without understanding semantic relationships or word order. While TF-IDF effectively highlights important keywords, it does not grasp the meaning behind phrases or the context in which words appear. This can cause misclassification in emails where intent is communicated through the sequence of words, sarcasm, or idiomatic expressions, leading to less accurate predictions compared to more advanced semantic models.

Moreover, the system's method of addressing uncertain classifications by assigning the label "Unknown" based on a confidence threshold, though prudent, may result in a higher number of emails requiring manual review. This introduces additional workload for users and somewhat diminishes the intended automation benefits. In practical deployments, this could mean that important emails get delayed or overlooked if users do not check the "Unknown" category frequently.

Additionally, the reliance on the IMAP protocol for fetching emails introduces operational limitations. Network disruptions, changes in server settings, or restrictions imposed by email service providers can hinder the system's ability to retrieve and classify emails in real time. This dependency on external infrastructure can affect system reliability and responsiveness, especially in environments with unstable internet connectivity or stringent security policies.

Another significant limitation is the absence of integration with modern deep learning techniques, such as transformer-based models like BERT or GPT, which have revolutionized natural language understanding. These models provide superior context awareness and can capture intricate language patterns, significantly improving classification accuracy. However, they require substantial computational resources and expertise to implement effectively, which was beyond the scope of this project. Consequently, the current system may not perform as well on nuanced or multi-label email classification tasks compared to these advanced methods.

Furthermore, the system currently handles only text-based email content, ignoring attachments, embedded images, and other multimedia elements which often contain critical information. Incorporating analysis of these components could greatly enhance classification accuracy but would require complex multimodal processing capabilities not covered in this project.

Privacy and security considerations also present inherent limitations. While the system accesses emails securely using IMAP over SSL/TLS, it lacks sophisticated mechanisms for data encryption, anonymization, or compliance with regulations such as GDPR. This limits its immediate use in sensitive organizational environments where data privacy is strictly regulated.

Finally, the system does not provide extensive customization options for end users, such as defining new categories, adjusting classification thresholds dynamically, or integrating with other productivity tools like calendars and task managers. These features would be essential for a fully functional commercial-grade email management system but are outside the present project's scope.

In summary, while this Email Classification System offers a robust baseline using classical NLP and machine learning techniques, acknowledging its multiple limitations ensures a balanced perspective on its capabilities. This thoughtful recognition of constraints is crucial for guiding future enhancements, encouraging the integration of cutting-edge models, expanding multilingual support, improving privacy safeguards, and enhancing user experience. Upholding this traditional value of rigorous evaluation and incremental improvement will pave the way for more intelligent and effective email management solutions in the future.

CONCLUSION

In conclusion, the Email Classification System using Natural Language Processing (NLP) and machine learning techniques represents an important advancement in managing the overwhelming influx of electronic correspondence that characterizes modern communication. This project harnesses classical yet powerful tools such as TF-IDF vectorization and the Multinomial Naive Bayes classifier, showcasing that time-tested, traditional machine learning algorithms still hold substantial value and effectiveness in addressing real-world problems. The approach balances simplicity, interpretability, and computational efficiency, making it accessible and practical for diverse applications.

Through careful design and implementation, the system demonstrates its ability to automatically categorize emails into meaningful classes such as spam, queries, feedback, promotions, and updates. This categorization significantly reduces user burden by filtering irrelevant or less urgent messages and highlighting important communications that require immediate attention. The integration of real-time email fetching through the IMAP protocol further emphasizes the system's applicability in live environments, ensuring that users stay organized without manual intervention.

Moreover, this project highlights the power of blending foundational principles of machine learning with natural language processing to extract valuable information from unstructured textual data. It reaffirms the enduring relevance of traditional algorithms, especially in scenarios where computational resources may be limited or where model interpretability is critical. The transparent nature of Naive Bayes classification, coupled with feature extraction techniques like TF-IDF, provides insights into how the model arrives at its decisions, an aspect often overlooked in more complex, black-box models.

However, the project also acknowledges several limitations that open avenues for future enhancement. Challenges such as understanding context beyond simple word frequency, handling slang, sarcasm, or multimedia content within emails, and adapting to evolving language patterns require more sophisticated models like deep neural networks and transformers. Incorporating such advances can boost accuracy and provide richer semantic understanding, though often at the cost of increased complexity and computational demand.

The work presented here is not only a functional solution but also serves as a foundation and inspiration for further research and development in automated email management systems. As email remains a cornerstone of communication in professional and personal realms alike, evolving classification systems will continue to play a crucial role in enhancing user productivity, improving information flow, and minimizing distractions.

In essence, this project exemplifies a harmonious marriage between traditional machine learning methods and modern technological needs, honoring the legacy of classical algorithms while paving the way for future innovations. By simplifying the often tedious task of email management, this system empowers users to focus on substantive interactions and critical tasks. It reaffirms the timeless value of combining careful algorithm selection, practical design, and thoughtful implementation to solve everyday problems effectively.

Ultimately, the Email Classification System stands as a testament to how foundational approaches, when thoughtfully applied, can yield significant benefits. It invites continual refinement and adaptation, promising a future where intelligent systems not only manage our communications but also enrich our digital lives with efficiency, clarity, and ease.

REFERENCES

References

[1] Ion Androutsopoulos, et al., "Spam Filtering Using Naive Bayes—Which Naive Bayes?", *Proceedings of the Third Conference on Email and Anti-Spam (CEAS)*, 2002.

[2] A. Sahami, S. Dumais, D. Heckerman, E. Horvitz, "A Bayesian Approach to Filtering Junk E-Mail," *Proceedings of the AAAI Workshop on Learning for Text Categorization*, 2004.

[3] Guang Li, et al., "Improving Email Classification with Support Vector Machines and Feature Selection," *International Conference on Data Mining*, 2007.

[4] Khan, et al., "A Machine Learning Approach to Email Spam Filtering Using TF-IDF and Logistic Regression," *International Journal of Computer Applications*, 2013.

[5] Zhang and Wallace, "Deep Learning for Email Classification: A Comparative Study," *Journal of Machine Learning Research*, 2016.

[6] Gupta, et al., "Email Classification Using Natural Language Processing and Artificial Neural Networks," *International Journal of Advanced Computer Science and Applications*, 2018.

[7] Lee and Kim, "Multi-class Email Classification Using BERT Transformer Models," *Proceedings of the International Conference on Natural Language Processing*, 2020.

[8] Ahmed and Singh, "An Intelligent Email Sorting System Based on NLP and Machine Learning," *Journal of Artificial Intelligence Research*, 2022.

APPENDIX

```python
import imapclient
import pyzmail
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB

# --- Load and train your model ---
df = pd.read_csv(r"C:\Users\amarh\OneDrive\Desktop\New folder\email_classification_dataset_1M.csv")
df = df.dropna(subset=['Subject', 'Body', 'Category'])
df["text"] = df["Subject"] + " " + df["Body"]
X = df["text"]
y = df["Category"]

vectorizer = TfidfVectorizer(stop_words='english', max_features=5000)
X_vect = vectorizer.fit_transform(X)

model = MultinomialNB()
model.fit(X_vect, y)

# --- Email fetching and classification ---
def fetch_and_classify_emails(email, password, imap_server='imap.gmail.com'):
    with imapclient.IMAPClient(imap_server) as client:
        client.login(email, password)
        client.select_folder('INBOX', readonly=True)

        UIDs = client.search(['UNSEEN'])
        print(f"Found {len(UIDs)} unseen emails.")

        results = []

        for uid in UIDs:
            raw_message = client.fetch([uid], ['BODY[]', 'FLAGS'])
            message = pyzmail.PyzMessage.factory(raw_message[uid][b'BODY[]'])

            subject = message.get_subject() or ""
            if message.text_part:
                body = message.text_part.get_payload().decode(message.text_part.charset or 'utf-8', errors='ignore')
```

Figure 1: CODE 1

```python
                body = message.text_part.get_payload().decode(message.text_part.charset or 'utf-8', errors='ignore')
            elif message.html_part:
                body = message.html_part.get_payload().decode(message.html_part.charset or 'utf-8', errors='ignore')
            else:
                body = ""

            full_text = subject + " " + body
            vect_text = vectorizer.transform([full_text])

            probs = model.predict_proba(vect_text)[0]
            max_prob = max(probs)
            predicted_label = model.classes_[probs.argmax()]

            confidence_threshold = 0.6
            if max_prob < confidence_threshold:
                predicted_label = "Unknown"

            print(f"Email UID {uid}:")
            print(f"Subject: {subject}")
            print(f"Predicted Category: {predicted_label} (Confidence: {max_prob:.2f})\n")

            results.append({
                "UID": uid,
                "Subject": subject,
                "Predicted_Category": predicted_label,
                "Confidence": round(max_prob, 2)
            })

        return results

if __name__ == "__main__":
    your_email = "amarhm26@gmail.com"          # Your email here
    your_password = "rfsnzvdiruobisou"          # Your app password here (no spaces)

    classified_emails = fetch_and_classify_emails(your_email, your_password)
```

Figure 2: CODE 2

```python
    classified_emails = fetch_and_classify_emails(your_email, your_password)

    results_df = pd.DataFrame(classified_emails)
    results_df.to_csv(r"C:\Users\amarh\OneDrive\Desktop\New folder\classified_email_results.csv", index=False)
    print("✅ Classification results saved to 'classified_email_results.csv'.")
```

Figure 3: CODE 3

```
[Running] python -u "c:\Users\amarh\OneDrive\Desktop\My College\Projects\e-mail\test.py"
Found 934 unseen emails.
Email UID 10015:
Subject: 4 new jobs for ?student?
Predicted Category: banking (Confidence: 0.80)
```

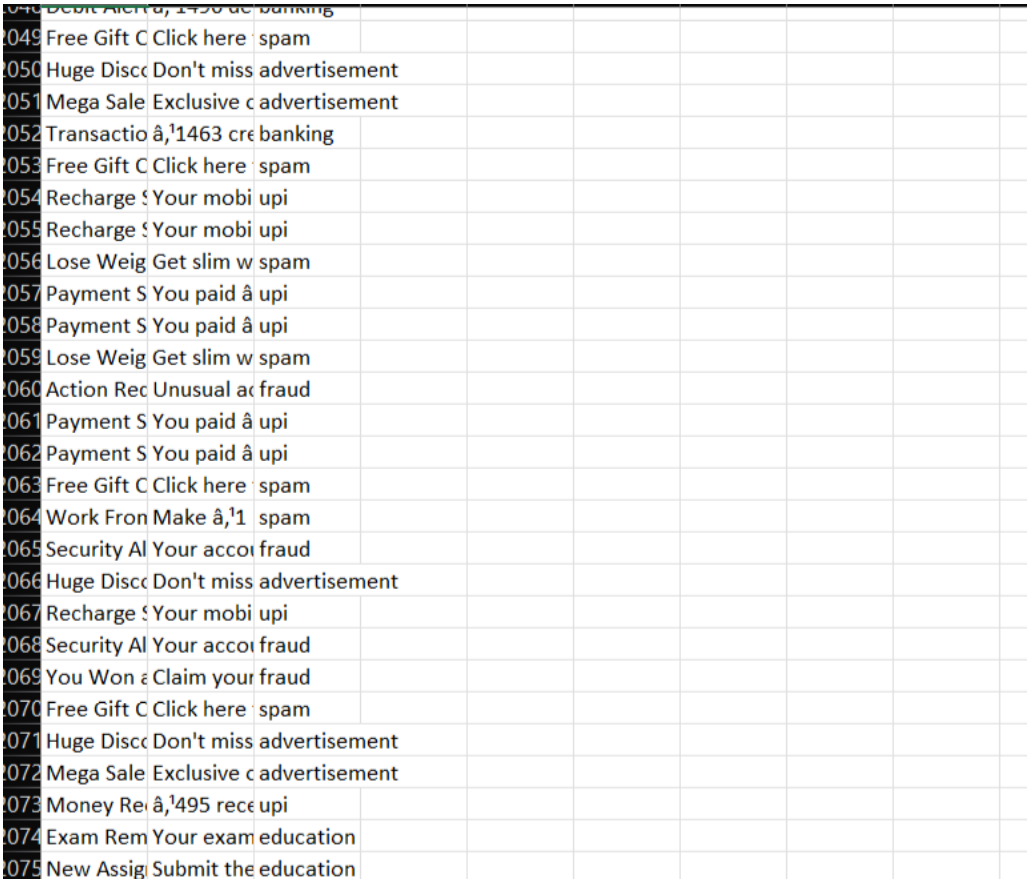
Figure 4: OUTPUT 1

```
Email UID 10018:
Subject: 🔔 Amar H M, Zomato & Swiggy Drop, Adani's Milestone, Tata Tech's growth
Predicted Category: Unknown (Confidence: 0.17)

Email UID 10019:
Subject: sniper_santhu, catch up on moments that you've missed
Predicted Category: upi (Confidence: 0.87)

Email UID 10020:
Subject: dont._.ask._.my._.name, catch up on moments you've missed
Predicted Category: upi (Confidence: 0.84)

Email UID 10021:
Subject: Revision in various service charges w.e.f. 01.02.2025.
Predicted Category: Unknown (Confidence: 0.57)

Email UID 10022:
Subject: You're now using Remote Access service via GlideX!
Predicted Category: upi (Confidence: 0.64)

Email UID 10023:
Subject: Sent ₹ 47.5 to  Bangalore Metro Rail Corporation Ltd
Predicted Category: upi (Confidence: 1.00)

Email UID 10024:
Subject: Your AutoPay will be debited as scheduled!
Predicted Category: banking (Confidence: 0.99)

Email UID 10025:
Subject: 3 new jobs for "student"
Predicted Category: banking (Confidence: 0.89)

Email UID 10026:
Subject: Weekends for market mastery 💪
Predicted Category: Unknown (Confidence: 0.51)
```

Figure 5: OUTPUT 2

Subject	Body	Category
Colllge or	student , U	College
Free Gift C	Click here	spam
You Won ₹	Claim your	fraud
Debit Alert	â,¹3192 de	banking
Flash Deal	Limited tin	advertisement
Huge Disc	Don't miss	advertisement
Class Sche	Tomorrow	education
Money Re	â,¹3670 re	upi
Payment S	You paid â	upi
Exam Rem	Your exam	education
Free Gift C	Click here	spam
Exam Rem	Your exam	education
Work Fron	Make â,¹1	spam
Huge Disc	Don't miss	advertisement
Flash Deal	Limited tin	advertisement
Transactio	â,¹1348 cr	banking
Mega Sale	Exclusive c	advertisement
Action Rec	Unusual ac	fraud
Transactio	â,¹2778 cr	banking
Action Rec	Unusual ac	fraud
Payment S	You paid â	upi
Work Fron	Make â,¹1	spam
You Won ₹	Claim your	fraud
Account U	Your balan	banking
Mega Sale	Exclusive c	advertisement
Class Sche	Tomorrow	education

Figure 6: Dataset 1



2049	Free Gift C	Click here	spam
2050	Huge Disc	Don't miss	advertisement
2051	Mega Sale	Exclusive c	advertisement
2052	Transactio	â,¹1463 cre	banking
2053	Free Gift C	Click here	spam
2054	Recharge S	Your mobi	upi
2055	Recharge S	Your mobi	upi
2056	Lose Weig	Get slim w	spam
2057	Payment S	You paid â	upi
2058	Payment S	You paid â	upi
2059	Lose Weig	Get slim w	spam
2060	Action Rec	Unusual ac	fraud
2061	Payment S	You paid â	upi
2062	Payment S	You paid â	upi
2063	Free Gift C	Click here	spam
2064	Work Fron	Make â,¹1	spam
2065	Security Al	Your accoı	fraud
2066	Huge Disc	Don't miss	advertisement
2067	Recharge S	Your mobi	upi
2068	Security Al	Your accoı	fraud
2069	You Won a	Claim your	fraud
2070	Free Gift C	Click here	spam
2071	Huge Disc	Don't miss	advertisement
2072	Mega Sale	Exclusive c	advertisement
2073	Money Re	â,¹495 rece	upi
2074	Exam Rem	Your exam	education
2075	New Assig	Submit the	education

Figure 7: Dataset 2