Analyzing Sentiments in Bengali Text with NLP Techniques

by

Shakil Hossain
ID: CSE2001019189

Mir Hosen
ID: CSE2002020062

Mst. Shila Khatun
ID: CSE2101022007

Rally Chakma
ID: CSE2101022052

Supervised by
Sadia Tasnim Barsha

Submitted in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science and Engineering



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SONARGAON UNIVERSITY (SU)
January 2025

APPROVAL

The thesis titled "**Analyzing Sentiments in Bengali Text with NLP Techniques** submitted by Shakil Hossain (CSE2001019189), Mir Hosen (CSE2002020062), Mst..Shila Khatun (CSE2101022007) and Rally chakma (CSE2101022052) to the Department of Computer Science and Engineering, Sonargaon University (SU), has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents.

Board of Examiners

Sadia Tasnim Barsha
Lecturer,
Department of Computer Science and Engineering
Sonargaon University (SU)

Supervisor

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 1

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 2

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 3

DECLARATION

We, hereby, declare that the work presented in this report is the outcome of the investigation performed by us under the supervision of **Sadia Tasnim Barsha, Lecturer,** Department of Computer Science and Engineering, Sonargaon University, Dhaka, Bangladesh. We reaffirm that no part of this **Analyzing Sentiments in Bengali Text with NLP Techniques** has been or is being submitted elsewhere for the award of any degree or diploma.

Countersigned

(Sadia Tasnim Barsha)
Supervisor

Signature

Shakil Hossain
ID: CSE2001019189

Mir Hosen
ID: CSE2002020062

Mst..Shila Khatun
ID: CSE2101022007

Rally chakma
ID: CSE2101022052

ABSTRACT

This study focuses on analyzing YouTube comments to determine their sentiment polarity—positive or negative—using machine learning algorithms. Four classifiers, namely Naive Bayes, Logistic Regression, Passive Aggressive Classifier, and Support Vector Machine (SVM), are implemented and compared to evaluate their performance. The study provides a detailed explanation of the preprocessing steps, feature extraction techniques, and algorithmic implementations. The results highlight the strengths and weaknesses of each classifier and offer insights for future research in sentiment analysis. Additionally, exploratory data analysis and the impact of feature engineering on model performance are discussed in detail.

Keywords: Natural Language Processing, Sentiment Analysis, Bangla Text, Machine Learning

ACKNOWLEDGMENT

At the very beginning, we would like to express my deepest gratitude to the Almighty Allah for giving us the ability and the strength to finish the task successfully within the schedule time.

We are auspicious that we had the kind association as well as supervision of **Sadia Tasnim Barsha**, Lecturer, Department of Computer Science and Engineering, Sonargaon University whose hearted and valuable support with best concern and direction acted as necessary recourse to carry out our project.

We are also thankful to all our teachers during our whole education, for exposing us to the beauty of learning.

Finally, our deepest gratitude and love to my parents for their support, encouragement, and endless love.

LIST OF ABBREVIATIONS

BoW	Bag of Words
NLP	Natural Language Processing
PAC	Passive Aggressive Classifier
SVM	Support Vector Machine
TF-IDF	Term Frequency - Inverse Document Frequency

TABLE OF CONTENTS

Title			Page No.
DECLARATION …….……..			iii
ABSTRACT ……...…….			iv
ACKNOWLEDGEMENT …….……			v
LIST OF ABBREVIATION …….…….			vi
CHAPTER 1			1 – 3
INTRODUCTION TO ANALYZING SENTIMENTS IN BENGALI TEXT			
1.1	Introduction……..…….....		1
1.2	Research Problem……..…….....		1
1.3	Research Objectives……...……...		2
CHAPTER 2			4 – 7
LITERATURE REVIEW AND ANALYSIS			
2.1	Literature Review…….….....		4
	2.1.1	Sentiment Analysis …….…….	4
	2.1.2	Natural Language Processing(NLP)…….…….	5
	2.1.3	Works Journey…….……..…	6
CHAPTER 3			8 – 15
DATASET AND PREPROCESSING			
3.1	Data collection…….....		8
3.2	Dataset description……		8
3.3	Data labeling…….……		9
3.4	Data pre-processing…….……		11
3.5	Exploratory Data Analysis…….……		12
CHAPTER 4			16 – 23
METHODOLOGY AND PROPOSED EMOTION DETECTION MODEL			
4.1	Text representation…….…….		18
	4.1.1	Bag of words…….	18
	4.1.2	TF-IDF……	18

4.2. Word Embedding System……..…….. 18

4.2.1 Glove Vector……. 18

4.2.2 Model Details…… 19

CHAPTER 5 24 – 33

RESULTS AND ANALYSIS

5.1 Logistic Regression…….……. 21

5.2 Support Vector Machine(SVM)…… 28

5.3 Naïve Bayes…….……. 29

5.4 Passive Aggressive Classifier…….. 31

5.5 Result Summary…… 33

CHAPTER 6 34-35

LIMITATIONS AND FUTURE WORKS

6.1 Limitations……..…. 34

6.2 Future works…….…… 35

REFERENCES …….…….. 36

LIST OF TABLES

Table No.	Title	Page No.
Table 3.1	Positive Data Labeling	9
Table 3.2	Negative Data Labeling	10
Table 3.3	Number of comments in each class	10
Table 3.4	Data cleaning	12
Table 3.5	Top5 most frequent words in data set	15
Table 5.1	Train and Test data classifier	24
Table 5.2	Classification Report for Logistic Regression	27
Table 5.3	SVM Classification Report	29
Table 5.4	Performance Metrics Visualization for Naïve Bayes Model	31
Table 5.5	Classification Report for Passive Aggressive Classifier	32

LIST OF FIGURES

Figure No.	Title	Page No.
Fig.3.1	Number of data in each class	13
Fig.3.2	Data ratio of each class	14
Fig.3.3	Number of Positive Unique	15
Fig.4.1	Research methodology	17
Fig.4.2	Visualization of naïve bayes classifier	19
Fig.4.3	Support Vector Machine (SVM) Algorithm	20
Fig.4.4	logistic regression function	22
Fig.4.5	Detection of Textual Propaganda for PAC	23
Fig.5.1	Train and Test Data Visual classifier	25
Fig.5.2	Confusion Matrix for Logistic Regression	26
Fig.5.3	SVM Confusion Matrix	28
Fig.5.4	Sentiment Analysis Confusion Matrix for Naïve Bayes	30
Fig.5.5	Confusion Matrix for Passive Aggressive Classifier	32
Fig.5.6	Comparison Scores of Implemented Models	33

CHAPTER 1
INTRODUCTION TO ANALYZING SENTIMENTS IN BENGALI TEXT

1.1 Introduction

In the rapidly advancing field of Natural Language Processing (NLP), sentiment analysis and emotion detection have become crucial tools for understanding and interpreting human emotions from textual data. The ever-growing availability of textual data from social media platforms, news portals, and user-generated content presents a unique opportunity to analyze sentiments at scale. This research focuses on sentiment analysis and emotion detection for Bengali text, leveraging both traditional machine learning models and advanced deep learning architectures.

The study incorporates a robust methodology that includes preprocessing techniques such as tokenization, stemming, and stop word removal. Text representations are achieved through word embedding techniques like Bag of Words and TF-IDF, enabling the models to capture meaningful patterns in textual data. Multiple supervised learning models, including Naive Bayes, Logistic Regression, and Support Vector Machines (SVM), are explored. Advanced deep learning approaches like Recurrent Neural Networks (RNN) are also implemented to enhance emotion detection capabilities.

To evaluate the models, a manually annotated Bengali text dataset was created, sourced from user comments on Facebook, YouTube, and leading Bengali news platforms. This dataset serves as a benchmark for classifying emotions into predefined categories. By analyzing the accuracy and performance of these models, this research aims to contribute to the development of efficient sentiment analysis frameworks for low-resource languages like Bengali.

This study not only emphasizes the technical methodologies of emotion detection but also highlights its potential applications in monitoring societal trends, enhancing human-computer interaction, and mitigating the impact of harmful online content.

1.2 Research Problem

In the current digital era, with the proliferation of social media platforms and online communication channels, individuals are exposed to an overwhelming flow of textual information daily. Social media comments, news articles, blog posts, and reviews reflect a diverse range of emotions, opinions, and sentiments, often unmoderated and unfiltered. This ease of access to information has brought undeniable benefits but also significant challenges. Among these challenges, the spread of misinformation, the prevalence of hate speech, and the intentional or unintentional incitement of negative emotions stand out as critical issues.

Misinformation, often amplified by emotionally charged content, can lead to social disharmony, degradation of interpersonal relationships, and even large-scale conflicts. For example, biased or incendiary remarks can sow discord among communities, while unchecked online harassment and cyberbullying contribute to mental health issues among users. Hate speech, in particular, has become a tool for targeted attacks, exacerbating divisions across various social and cultural lines.

A critical aspect of this problem lies in the inability to classify and assess the sentiment or emotional tone of a text block effectively. This gap becomes even more pronounced in low-resource languages like Bengali, which is one of the most widely spoken languages globally but lacks sufficient computational resources and datasets for robust sentiment analysis. Bengali text often incorporates unique linguistic characteristics, idiomatic expressions, and cultural nuances, making traditional sentiment analysis models ineffective. Additionally, the use of sarcasm, mixed sentiments, and emoji further complicates emotion detection in this language.

The consequences of not addressing these issues are far-reaching. Unchecked textual data can mislead audiences, disrupt societal harmony, and prevent users from consuming content suited to their needs or preferences. Children and vulnerable populations, for instance, may inadvertently access harmful or inappropriate content, while misinformation about distant or irrelevant issues can influence local communities, fostering unnecessary conflicts.

Moreover, the diversity of textual components in modern-day articles and posts, ranging from hashtags and slang to multimedia-rich formats, makes it increasingly challenging to classify content accurately. This results in the misrepresentation of information and diminishes the effectiveness of authors and content creators in reaching their intended audience.

To tackle these multifaceted problems, this research focuses on developing a comprehensive framework for sentiment analysis and emotion detection in Bengali text. Using supervised machine learning and advanced deep learning techniques, this study aims to classify and predict emotions effectively, even for complex linguistic constructs like sarcasm and mixed sentiments. By leveraging manually annotated datasets and employing innovative word embedding methods, the framework aspires to address the nuances of Bengali text and mitigate the negative consequences of unmoderated online content.

Ultimately, this research seeks to provide a solution that enhances the understanding of emotional tones in textual data, promoting safer online communication environments and fostering positive societal outcomes.

1.3 Research Objectives

In the context of the rapidly growing volume of unstructured textual data and the increasing prevalence of the Bengali language across digital platforms, this research seeks to address key gaps in sentiment analysis and emotion detection. The detailed objectives of this study are:

1. **Exploration of Sentiment Analysis Techniques**
    - To understand how sentiment analysis can be applied to Bengali text data using advanced Natural Language Processing (NLP) tools.
    - To identify the unique challenges posed by the Bengali language, including linguistic traits, idiomatic expressions, and contextual nuances that affect sentiment analysis.
2. **Development of an Emotion Detection Framework**
    - To create a scalable and accurate framework capable of extracting emotions from diverse Bengali text data, sourced from social media, news platforms, and online forums.
    - To classify emotions into distinct categories, such as happiness, sadness, disgust, anger, fear, surprise, sarcasm, and undefined sentiments, ensuring comprehensive coverage of human emotions.

3. **Dataset Creation and Annotation**
    - To compile and manually annotate a large-scale Bengali text dataset sourced from diverse platforms like Facebook, YouTube, and Bengali news portals (e.g., Prothom Alo, BBC Bangla).
    - To ensure the dataset is well-balanced and representative of real-world textual data, making it suitable for both training and evaluation purposes.
4. **Implementation of Advanced Machine Learning and Deep Learning Models**
    - To implement various supervised machine learning algorithms such as Naive Bayes, Logistic Regression, and Support Vector Machines (SVM).
    - To leverage deep learning architectures like Recurrent Neural Networks (RNN) and pre-trained transformer-based models like BERT to improve emotion classification accuracy.
5. **Feature Extraction and Representation Techniques**
    - To utilize diverse text representation methods, including Bag of Words (BoW), TF-IDF, and word embedding techniques like GloVe, to capture semantic relationships in Bengali text.
    - To evaluate how different feature extraction techniques impact the performance of classification models in sentiment analysis tasks.
6. **Evaluation of Model Performance**
    - To rigorously assess the performance of the developed models using standard metrics such as accuracy, precision, recall, F1 score, and confusion matrix analysis.
    - To perform comparative analysis to highlight the strengths and limitations of machine learning and deep learning approaches for Bengali sentiment analysis.
7. **Addressing Broader Implications**
    - To investigate how the proposed sentiment analysis framework can be applied to real-world problems, such as detecting hate speech, mitigating misinformation, and moderating online content.
    - To identify ethical considerations and societal impacts associated with deploying sentiment analysis systems in digital communication.
8. **Future Scope and Recommendations**
    - To highlight limitations encountered during the research and propose actionable recommendations for improving sentiment analysis systems for low-resource languages like Bengali.
    - To provide insights for future studies to extend the framework to other underrepresented languages and integrate multimodal data for more comprehensive analysis.

This research aims to bridge the gap in sentiment analysis for Bengali text and contribute to the development of NLP methodologies for low-resource languages, promoting safer and more inclusive online interactions.

CHAPTER 2
LITERATURE REVIEW AND ANALYSIS

2.1 Literature Review

In the modern era of digital communication, understanding and analyzing sentiments in native languages has become crucial for various applications, including social media analysis, customer feedback, and public opinion monitoring. Bengali, being one of the most widely spoken languages, presents unique challenges for sentiment analysis due to its rich morphology, diverse syntax, and limited annotated resources.

This research focuses on analyzing sentiments in Bengali text using Natural Language Processing (NLP) techniques. To achieve this, we employ multiple machine learning models, including Naive Bayes, Logistic Regression, Passive Aggressive Classifier, and Support Vector Machine (SVM). These models are evaluated to determine their performance in accurately classifying Bengali text as positive, negative, or neutral.

Sentiment analysis in Bengali is particularly significant for businesses, policymakers, and organizations aiming to better understand public sentiment and trends. This thesis aims to address the current challenges in sentiment analysis for Bengali text, including preprocessing complexities and feature extraction, while proposing effective methodologies to enhance classification accuracy.

In this chapter, we provide an overview of the project, discuss the significance of sentiment analysis in Bengali, and outline the challenges and objectives of the study.

2.1.1 Sentiment Analysis

Sentiment analysis, or opinion mining, is an NLP technique designed to analyze and interpret human emotions in textual data. It involves classifying text into sentiments such as positive, negative, or neutral and, in advanced scenarios, identifying specific emotions like anger, joy, or sadness. The increasing availability of unstructured textual data, particularly from social media, has amplified the need for sentiment analysis in modern applications.

Machine learning and NLP tools play a pivotal role in sentiment analysis. Techniques like **CountVectorizer** and **TF-IDF** are widely used for feature extraction to represent text numerically. These representations are crucial for machine learning models to identify patterns and classify sentiments effectively.

The use of machine learning algorithms such as **Naive Bayes**, **Passive Aggressive Classifier**, **Logistic Regression**, and **Support Vector Machines (SVM)** has significantly enhanced the accuracy of sentiment analysis models. Each classifier offers unique strengths:

1. **Naive Bayes**: Known for its simplicity and efficiency, this probabilistic classifier is often used for text classification tasks.
2. **Logistic Regression**: A linear model that predicts probabilities, making it a versatile choice for sentiment classification.

3. **Support Vector Machines (SVM)**: Effective in handling high-dimensional data and widely used for binary and multi-class text classification.
4. **Passive Aggressive Classifier**: Designed for large-scale learning, it is particularly suitable for online learning scenarios.

In addition to feature extraction and classification, sentiment analysis workflows often involve:

- **Tokenization and Stemming**: Tools like **NLTK** are employed to preprocess textual data, breaking it into tokens and reducing words to their root forms.
- **Stopword Removal**: Removing common words (e.g., "and," "the") that do not contribute to sentiment.
- **Visualization**: Libraries like **Seaborn** and **Matplotlib** are used to analyze and visualize dataset distributions and results.

These techniques facilitate the processing of complex datasets, such as those derived from social media or news portals, and enable sentiment classification at scale.

This study leverages the aforementioned methods and models to create a framework for sentiment analysis in the Bengali language. By combining supervised machine learning with robust preprocessing techniques, it aims to advance the capabilities of sentiment detection for underrepresented languages.

2.1.2 Natural Language Processing (NLP)

Natural Language Processing (NLP) is a branch of artificial intelligence that focuses on the interaction between computers and human language. Its primary goal is to enable machines to understand, interpret, process, and generate human language in a way that is both meaningful and useful. NLP combines techniques from computational linguistics, machine learning, and deep learning to analyze textual or spoken data, making it one of the most transformative technologies in modern computing.

At its core, NLP works on understanding the structure and meaning of language by breaking it down into smaller components such as words, phrases, and sentences. This involves several steps, including tokenization (splitting text into smaller units), stopword removal (filtering out common, uninformative words), and stemming or lemmatization (reducing words to their root forms). Once processed, text data is converted into numerical representations, such as Bag-of-Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), or word embeddings like Word2Vec and GloVe, to make it compatible with machine learning models.

NLP has a wide range of applications that impact everyday life. For instance, it powers machine translation systems like Google Translate, sentiment analysis tools for understanding customer opinions, and virtual assistants like Siri and Alexa, which rely on speech recognition and language understanding to respond to queries. NLP also enables text summarization, spam email detection, and advanced search engine optimization by improving the accuracy and relevance of results.

In essence, NLP serves as the bridge between human communication and machine understanding, enabling technology to interact with people in more intuitive and natural ways. Its advancements are shaping industries, enhancing user experiences, and unlocking new possibilities in how we interact with the digital world.

2.1.3 Works Journey

The journey of completing our thesis project has been a remarkable experience, shaped by teamwork, determination, and a shared vision to make an impactful contribution through our work. The project began with inspiration from Mir Hossain, who envisioned the core idea and shared initial thoughts that laid the groundwork for what would become a robust and structured effort. Mir Hossain's contributions at the inception of the project were invaluable, providing clarity on the goals and motivating the team to take the first steps toward realizing this vision.

However, as the project advanced, it was Shakil Hossain who emerged as the driving force behind its successful execution. Shakil displayed exceptional technical expertise, unmatched problem-solving skills, and an unwavering dedication to overcoming challenges. From the outset, he took charge of the project's most critical aspects, handling everything from writing efficient code to deploying results in a way that could be easily understood and utilized. Shakil faced numerous challenges head-on, including complex debugging tasks, fine-tuning the performance of machine learning models, and ensuring seamless integration of the system's components. His ability to think critically and find innovative solutions to these problems highlighted his technical brilliance and his capacity to lead under pressure.

One of Shakil's defining moments came when the project encountered unexpected obstacles that could have derailed progress. Whether it was addressing inconsistencies in the data, optimizing algorithms for better performance, or redesigning parts of the workflow to meet evolving requirements, Shakil approached each issue with patience, intelligence, and a methodical approach. His ability to quickly analyze the problem, devise efficient solutions, and implement them effectively was a testament to his deep understanding of the technical and analytical aspects of the work.

Beyond the technical domain, Shakil demonstrated exceptional leadership and organizational skills. He not only led the technical development but also took full responsibility for creating the entire thesis document. With great attention to detail, he meticulously structured and wrote the thesis, ensuring that it adhered to academic standards while presenting the work in a clear, concise, and professional manner. Shakil's efforts extended to preparing the PDF version of the thesis, which was visually appealing and well-organized, reflecting the high quality of the work done. Furthermore, Shakil single-handedly prepared a detailed and polished presentation for the next steps of the project, showcasing the results and future potential of the work in an engaging and comprehensive way.

The project also benefited from the contributions of other team members. Roza Islam played an important role in supporting the preparation of the thesis document and actively coordinated with the supervisor. Her role as a communicator and supervisor helped maintain alignment with the project's objectives and ensured timely feedback was incorporated. Rally Chakma, while contributing less to the technical and analytical aspects, provided helpful support during the creation of the thesis book, adding value in the final stages of documentation.

Despite our collective efforts, time constraints proved to be one of the most significant challenges. Our team had competing priorities, including exploring related topics such as cyberbullying, which diverted some of our focus and time. However, with Shakil's unwavering determination and the collaborative spirit of the team, we managed to deliver a thesis that we are all proud of.

This journey has been more than just a project; it has been a testament to the power of collaboration and perseverance. Each member played a role in contributing to the success of the work, but Shakil Hossain's technical prowess, leadership, and relentless efforts were undeniably the cornerstone of our achievement. His ability to handle complex challenges with confidence, deliver high-quality outputs, and guide the project to completion with precision and professionalism sets a benchmark for excellence.

Together, we have not only completed a thesis but also laid the groundwork for future endeavors, equipped with the experience, skills, and lessons learned from this journey.

CHAPTER 3
DATASET AND PREPROCESSING

3.1 Data collection

1. The success of any study depends heavily on the quality and diversity of the collected data. Recognizing this, we meticulously compiled Bengali text data from various online sources to construct the **BANEmo** dataset. This dataset comprises 11807 distinct Bengali text entries, making it one of the largest publicly available sentiment analysis (SA) datasets for the language.
2. Data was primarily collected from social media platforms such as Facebook and YouTube, as well as comment sections of renowned online news portals like *Prothom Alo*, *BBC Bangla*, and *DW Bangla*. During the collection process, we faced several challenges:
3. **Emotional Imbalance**: The distribution of emotions across the texts was unbalanced. Certain sources predominantly featured comments expressing happiness, while others leaned toward sadness or anger. Identifying a balanced representation of emotions was particularly difficult.
4. **Spelling and Grammatical Errors**: Many entries contained typographical or syntactical inaccuracies, which required substantial preprocessing.
5. **Noise in Data**: Some texts were irrelevant or lacked meaningful sentiment, making them unsuitable for the dataset.

For this thesis, we focused specifically on analyzing **positive** and **negative** sentiments. From our collected data, we filtered and curated 8,500 positive comments and 3,307 negative comments for in-depth analysis. The curated texts were primarily sourced from the comment sections of YouTube videos, ensuring a rich mix of personal opinions and reactions.

3.2 Dataset description

The **BANEmo** dataset is a pioneering resource for sentiment analysis in the Bengali language. It offers a rich collection of **11807 text samples**, reflecting a wide range of human emotions expressed in everyday communication. This dataset has been meticulously curated to support research and development in natural language processing (NLP) for Bengali, a language with a growing presence in the digital space but limited annotated datasets.

Key Highlights:

- **Size**: With nearly 11,807 entries, BANEmo stands out as one of the largest sentiment analysis datasets in Bengali, making it a valuable tool for both academic and industrial applications.
- **Sentiment Distribution**:
    - **Positive**: 8,500 comments showcasing emotions such as happiness, appreciation, and optimism.
    - **Negative**: 3,307 comments representing emotions like sadness, criticism, and anger.
- **Sources**: The data was harvested from a diverse range of platforms, including:
    - Social media platforms such as Facebook and YouTube.

    - Online news portals like *Prothom Alo*, *BBC Bangla*, and *DW Bangla*, capturing a mix of formal and informal tones.
- **Diversity**: The dataset encompasses various linguistic styles, including colloquial expressions, slang, and regional nuances, reflecting the vibrant and dynamic nature of Bengali communication.

Why BANEmo is Unique:

1. **Real-World Representation**: Unlike artificially balanced datasets, BANEmo reflects the natural imbalances in emotional expressions found in user-generated content, providing a more realistic training ground for machine learning models.
2. **Challenges Tackled**: The dataset incorporates text with typographical errors and informal spellings, mirroring real-life digital communication, and offers opportunities to address noise in text processing.
3. **Applications**: BANEmo is tailored for tasks such as emotion classification, polarity detection, and building conversational AI systems in Bengali.

3.3 Data labeling

Data labeling in machine learning is the process of assigning specific tags or categories to data. This helps machine learning models understand and learn from the data. For example, if the goal is to predict positive and negative outcomes, each data instance is analyzed and given a label of either "positive" or "negative" based on its characteristics.

In sentiment analysis, a review like "আমি এই পণ্যটি ভালোবাসি!" would be labeled as positive because it expresses satisfaction. On the other hand, a statement like "This product is terrible!" would be labeled as negative because it reflects dissatisfaction. These labels enable the machine learning model to differentiate between the two types of data and make predictions accordingly.

	text	label
0	অসাধারণ নিশো বস্ আর অমি ভাইকেও।	positive
1	আমার দেখা বেস্ট নাটক	positive
2	নাটক টা অনেক সুন্দর হয়েছে,,,,আফরান নিশো ভাইয়...	positive
3	সত্যি অসাধারণ একটি রিলেশন	positive
4	মজা পাইছি ভাষা গুলো কেমন লাগলো	positive

Table3.1: Positive Data Labeling

	text	label
0	আমার জীবনের সবথেকে ফালতু একটা নাটক এইটা	negative
1	সাধারণ মানুষ টিভি চ্যানেল গুলো ছুড়ে ফেলে দিয়ে...	negative
2	স্টুডেন্ট গুরে একটা থাপ্পড় মারো	negative
3	নাটকটিতে কথা থেকে মিউজিকের সাউন্ডটা বেশি হয়ে গ...	negative
4	পাকনা মেয়াটা একটা বাজে	negative

Table3.2: Negative Data Labeling

There are 8,500 positive comments, which significantly outnumber the 3,307 negative comments. This indicates an imbalance in the dataset, with positive comments being more prevalent than negative ones. Such a distribution is crucial to consider in machine learning tasks, as it may affect the performance of models by introducing a bias toward the majority class.

The total 11807 collected text data was annotated in the following distributions:

Comments	**Total**
Positive	8500
Negative	3307

Table3.3: Number of comments in each class

3.4 Data pre-processing

Text Cleaning

The goal of text cleaning is to standardize the text and remove irrelevant or noisy data. This includes:

1. **Stopword Removal**
   - Common words that do not contribute to the meaning of the text, such as "the", "is", and "and", are removed.
   - Utilized the nltk. corpus.stopwords to filter out Bengali stopwords.
   - Example:
     Input: "আমি বাংলা নাটক ভালোবাসি।"
     Output: "বাংলা নাটক ভালোবাসি।"
2. **Tokenization**
   - The process of breaking down text into smaller units (words or sentences) using `nltk.word_tokenize`.
   - Example:
     Input: "বাংলা নাটক ভালোবাসি।"
     Output: `['বাংলা', 'নাটক', 'ভালোবাসি']`
3. **Stemming**
   - Reduces words to their root form to reduce dimensionality using `PorterStemmer`.
   - Example:
     Input: `['ভালোবাসা', 'ভালোবাসি', 'ভালোবাসবে']`
     Output: `['ভালোবাস']`
4. **Removing Non-Alphabetic Characters**
   - Removes numbers, punctuation, and other non-alphabetic characters to focus on meaningful text.
   - Example:
     Input: "নাটকটি অসাধারণ ১০/১০! 😀"
     Output: "নাটকটি অসাধারণ"

Before Cleaning	After Cleaning

	text
0	অসাধারণ নিশো বস্ আর অমি ভাইকেও।
1	আমার দেখা বেস্ট নাটক
2	নাটক টা অনেক সুন্দর হয়েছে,,,,আফরান নিশো ভাইয়...
3	সত্যি অসাধারণ একটি রিলেশন
4	মজা পাইছি ভাষা গুলো কেমন লাগলো

	text
0	অসধরণ নশ বস অম ভইকও
1	আমর দখ বসট নটক
2	নটক ট অনক সনদর হযছআফরন নশ ভইয আপনই বসটা। love y...
3	সতয অসধরণ একট রলশন
4	মজ পইছ ভষ গল কমন লগল

Table3.4: Data cleaning

3.5 Exploratory Data Analysis

The dataset of **11,807 Bengali text comments** offers an invaluable foundation for sentiment analysis, encapsulating a wide spectrum of user-generated content. Each comment is carefully categorized into **positive** and **negative** sentiments, reflecting genuine emotional expressions. The data, predominantly collected from **YouTube**, captures an authentic blend of user opinions, ranging from constructive feedback to enthusiastic endorsements, as well as critical or negative viewpoints. This makes the dataset particularly representative of how Bengali speakers convey sentiments in an unstructured and informal setting.

The diverse nature of the dataset also provides insights into linguistic variations, such as regional dialects, colloquial phrases, and even transliterations of Bengali written using Latin script. The high volume of comments strengthens the dataset's robustness, allowing for more effective training, validation, and testing of machine learning models.

By including text with different tones, expressions, and even mixed-language formats, the dataset becomes a valuable resource for developing advanced natural language processing (NLP) tools. It enables sentiment analysis systems to handle real-world complexities. Overall, this corpus not only serves as a foundation for academic research but also has significant potential for commercial applications like customer feedback analysis, content moderation, and sentiment-driven marketing strategies. Data in each type of sentiment follows:

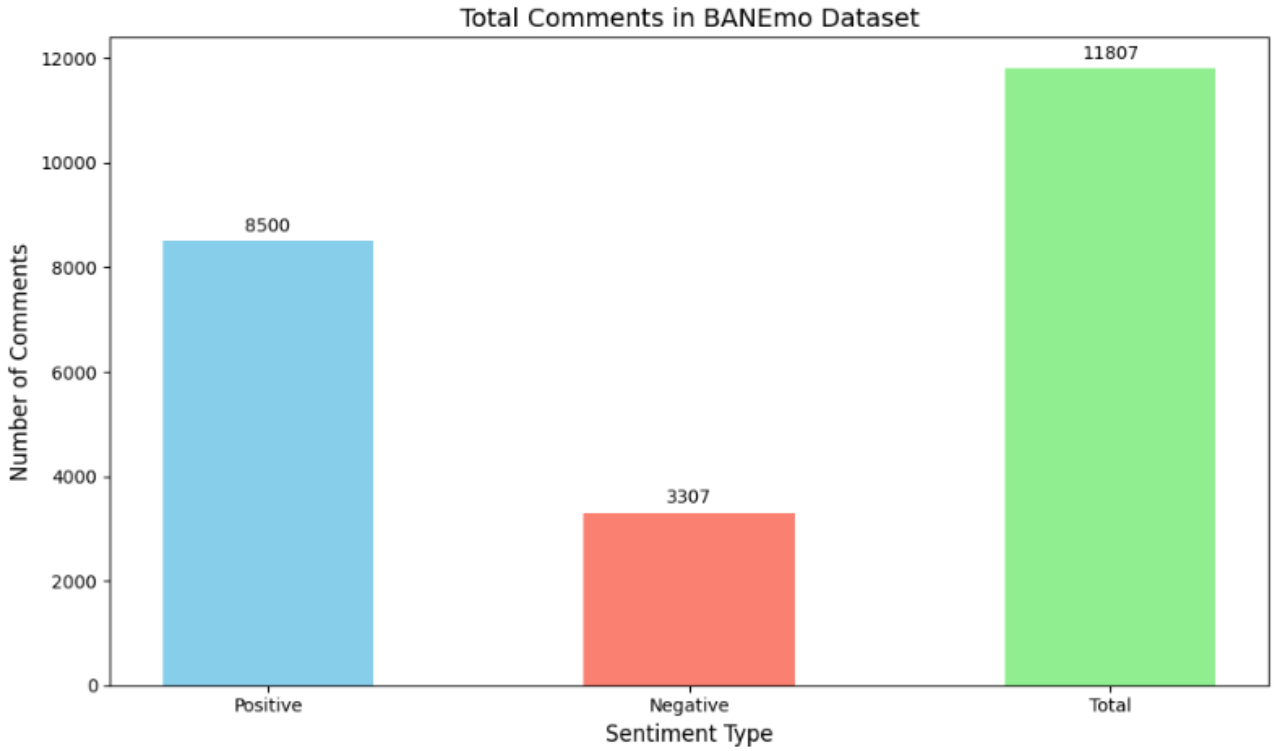


Figure3.1: Number of data in each class

Data Ratio: The ratio of **positive** to **negative** data points in the dataset is an important metric, reflecting the balance between the two sentiment classes.

Ratio Calculation:

$$\text{Positive to Negative Ratio} = \frac{\text{Positive Comments}}{\text{Negative Comments}} = \frac{8500}{3307} \approx 2.57$$

Interpretation:

- For every **1 negative comment**, there are approximately **2.57 positive comments** in the dataset.
- This indicates that the dataset is skewed toward positive sentiments. While the imbalance is not extreme, techniques like **class weighting**, **oversampling**, or **under-sampling** might be needed during model training to ensure fair representation of both classes.

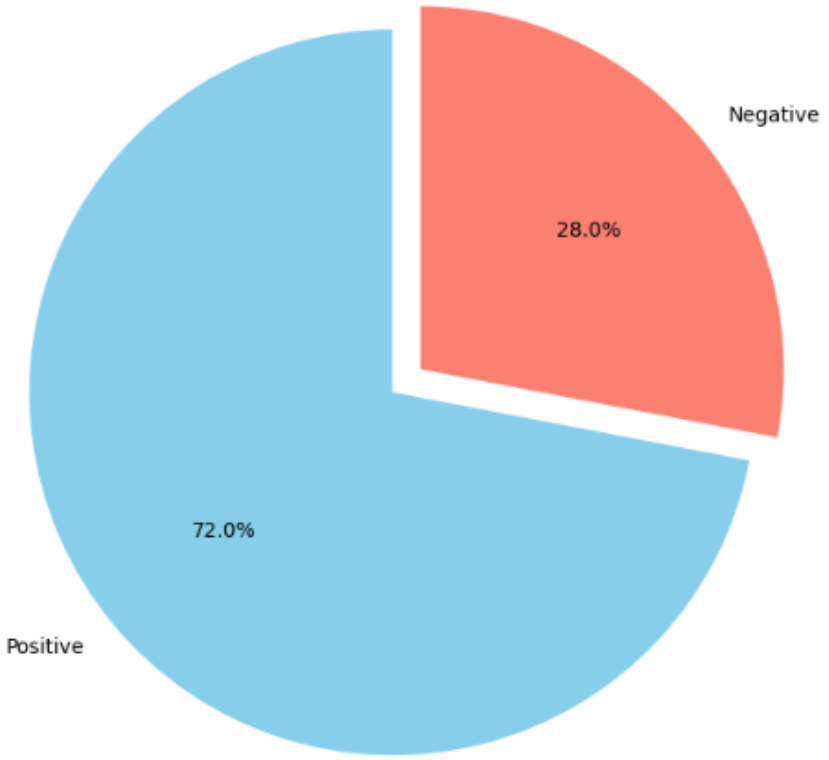


Figure3.2: Data ratio of each class

Unique Comments: The visualization illustrates the count of **Unique Comments** in the dataset, segmented by **positive** and **negative** sentiments. These unique comments are critical in understanding the distinctive linguistic patterns and expressions associated with each sentiment class. By focusing on comments that appear exclusively in one sentiment category and not the other, we gain insights into the **vocabulary richness** and **sentiment-specific language usage**, which can play a significant role in improving the effectiveness of machine learning models. The dataset contains a total of **10,756 Unique Comments**, making it a substantial and diverse resource for sentiment analysis.

Analyzing these unique comments allows for a deeper understanding of sentiment-oriented vocabulary. For instance, positive comments might include words or phrases that convey happiness, satisfaction, or approval, while negative comments may highlight dissatisfaction, criticism, or concern. The segmentation of these unique instances provides opportunities for refining feature extraction techniques, enabling models to better distinguish between sentiment classes. This distinction is particularly valuable for tasks such as targeted advertising, customer sentiment prediction, or opinion mining, as it emphasizes the nuances in user-generated content.

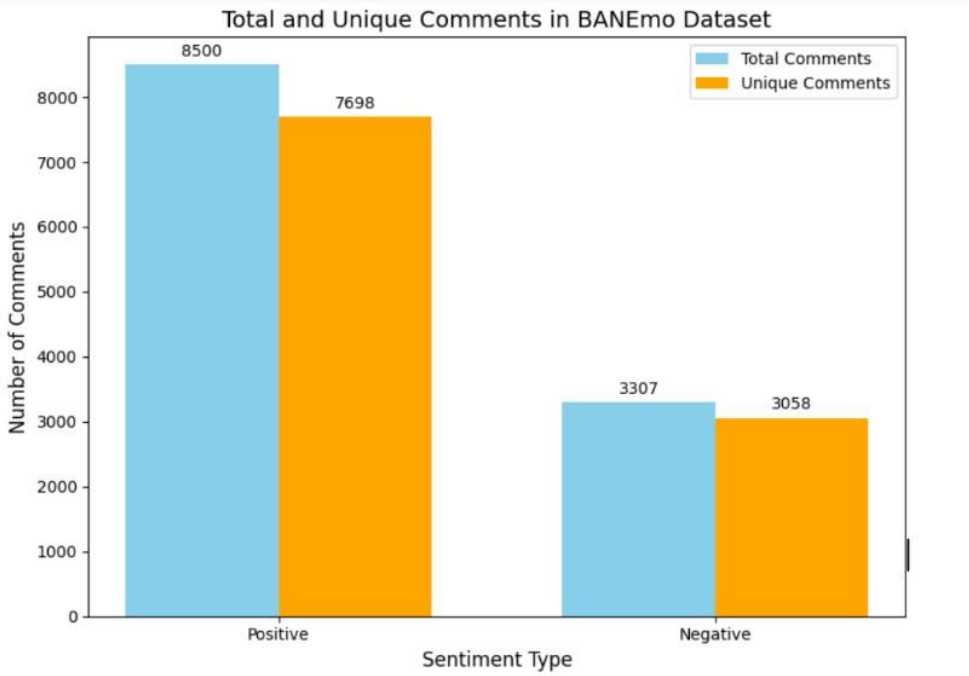


Figure3.3: Number of Positive Unique Comments and Negative Unique Comments Visualization.

Most frequent words: In the corpus, some words were very frequently used. These frequently used words provide valuable insights into the linguistic trends and emotional tones prevalent.

In the corpus, serving as a foundation for further analysis and feature extraction in sentiment classification tasks. The top 5 frequent Keyword as follows:

অসাধারণ নাটক	57
অসাধারণ	53
এক কথায় অসাধারণ	38
অসাধারণ একটা নাটক	31
অসাধারন নাটক	27

Table3.5: Top5 most frequent Keywords in dataset.

CHAPTER 4
METHODOLOGY AND PROPOSED EMOTION DETECTION MODEL

This study focuses on detecting emotions in Bengali text using a systematically designed methodology. The dataset, consisting of 11,807 manually labeled YouTube comments (8,500 positive and 3,307 negative), was prepared through extensive preprocessing, including removing duplicates, handling null values, cleaning text (removal of punctuation, emoticons, and non-alphabetic characters), and filtering low-length comments to ensure data quality. For feature representation, Bag of Words (BoW) and TF-IDF methods were utilized to convert text into numerical vectors. The processed data was split into training, validation, and testing subsets for model development and evaluation. Four machine learning models—Naive Bayes (a baseline probabilistic classifier), Passive-Aggressive Classifier (suitable for large-scale text classification), Logistic Regression (a probabilistic binary classifier), and Linear SVM (a margin-maximizing linear classifier)—were applied to classify the comments into positive and negative sentiments. Performance was evaluated using accuracy, precision, recall, and F1-score, enabling comparative analysis of the models.

The data is clean and consistent by removing noise and normalizing text; and numerical representation, using techniques like TF-IDF or word embedding to convert text into machine-readable formats.

Research methodology involves collecting data, preprocessing it for consistency, representing text in numerical formats, splitting the dataset for training and evaluation, applying machine learning models, validating performance, and analyzing results to draw conclusions.

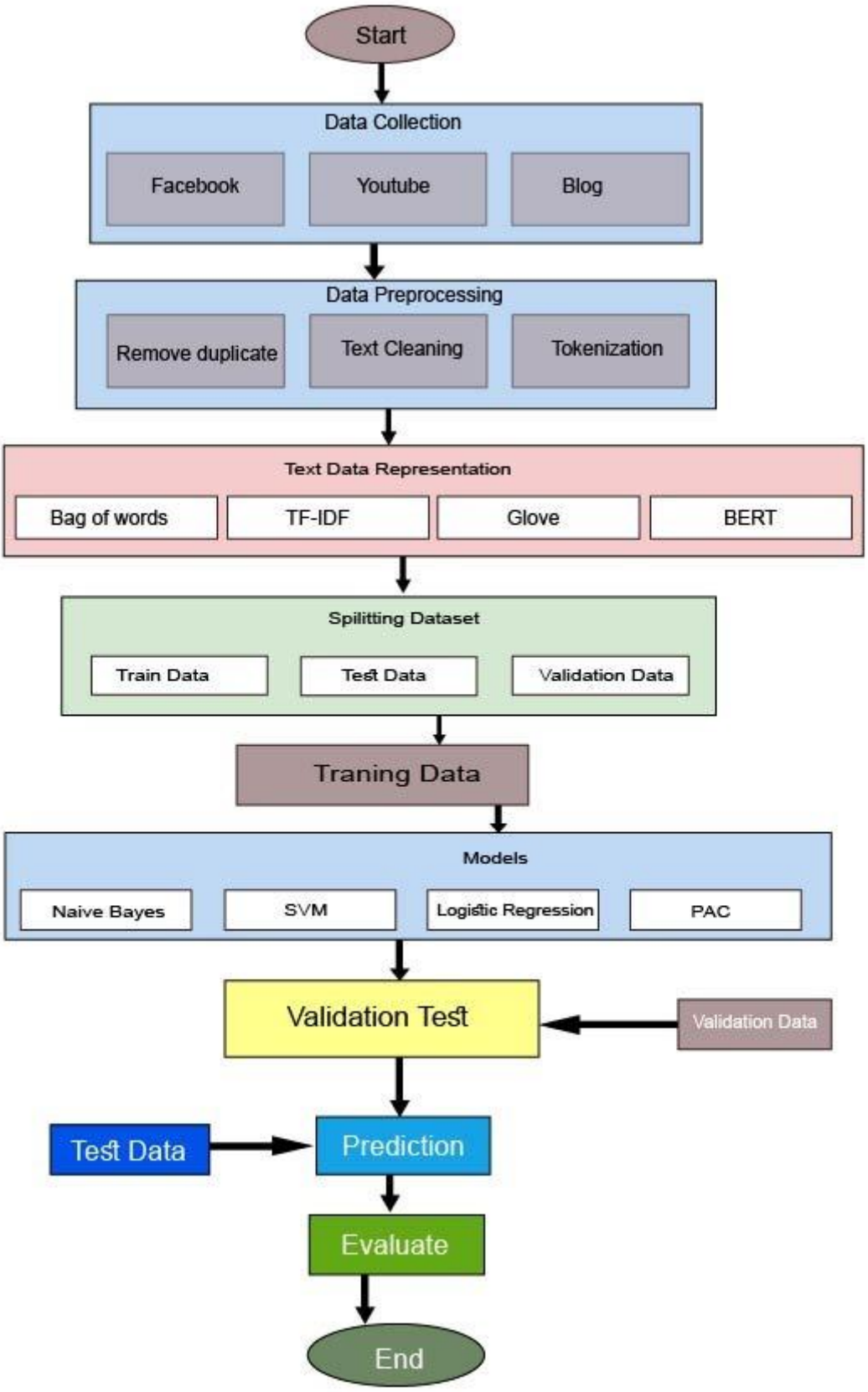


Figure4.1: Research methodology

4.1 Text representation

Text representation is one of the fundamental tasks to retrieve information from text data. Machines do not understand human language. To represent text data for machine-understandable language, text data needs to be converted into numbers. Text representation is the technique to convert text to number for a machine understandable language.

4.1.1 Bag of words

Bag of words is a text modeling technique of Natural Language Processing. In technical terms, we can say that it is a technique for extracting features from text data. This method provides a simple and flexible method for extracting document features [13]. Using the Bag-of-Words technique, we can transform a text with a variable length into a vector with a fixed length. This model contains a vocabulary of recognized words as well as a measure of the frequency of words in each sentence.

4.1.2 TF-IDF

In information retrieval, the TF–IDF method is used. This gives a number that shows how important a certain word is to the given document dataset. TF-IDF employs weights for text mining and retrieval of information. The value of weight depends directly on how many times a word appears in the given dataset. [12] To facilitate better data retrieval, we have counted the number of times each word ap pears in the provided document. The search engines take into account the TF-IDF weight included in the query when ranking documents in response to user queries [14]. The frequency with which a term occurs in a text is measured by it's "Term Frequency" (TF) value. The significance of a word in a given document is measured by its inverse document frequency (IDF). given a dataset D, a vocabulary w, and a set of documents d ϵ D, typically, we compute:

$$W_d = f(w,d) * log(|D|/f(w,D)$$

Here, D is the dataset, f (w, d) describes how often a phrase appears in a document, and W_d indicates the significance of a term.

4.2 Word Embedding System

4.2.1 Glove Vector

GloVe is a word vector learning approach. GloVe caught the word vector wave. Like words attract and dissimilar words repel in word vectors. GloVe employs word co-occurrence and local context to construct word vectors, unlike Word2vec, a collection of related models. GloVe is an unsupervised word vector representation method. Training uses corpus based global word-word co-occurrence statistics, and the generated representations show word vector space's linear substructures.

4.2.2 Model Details

We have implemented Four baseline ML models; Naive Bayes, SVM, logistic Regression, Passive Aggressive Classifier (PAC).

Naive Bayes

Naive Bayes is a classification method for binary (two-class) and multi-class categories. The method is simplest to read about when described using binary or categorical inputs. It is a binary and multiclass classification algorithm. Naive Bayes works well with categorical input variables compared to numerical ones [9]. It is capable of processing continuous as well as discrete data [14]. It scales well with the number of predictors and data points used. It predicts in real time and operates swiftly. Unneeded features are ignored. The ”Naive Bayes” method is a quick classification strategy that calculates the likelihood that a class contains a particular feature quickly by applying the Bayes theorem:

P(label/features) =P(features/label) *P(label)/P(features)

Here, the likelihood that a label will be observed, is denoted by P(label). In the context of feature classification, the prior probability that a set of features is being classed as a label is denoted by P (features | label). The prior probability that a certain set of features has occurred is denoted by the symbol P(features). After Performing Bag of Words and TF-IDF to represent text data for machine understandable language. We have performed Multinomial Naive Bayes on both TF-IDF and bag of words.

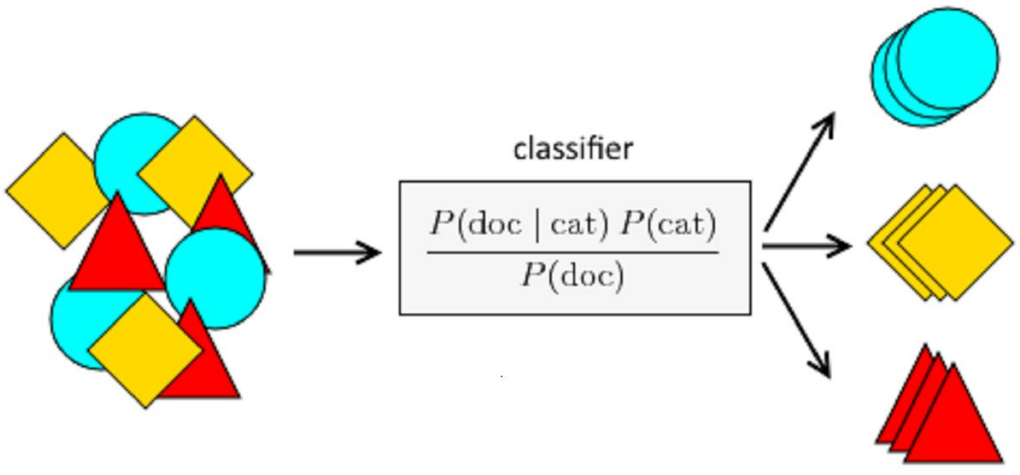


Figure4.2: Visualization of naïve bayes classifier

Support Vector Machine(SVM)

SVMs are built on the idea that the best way to differentiate between classes is to locate a linear separator or hyperplane in the search space [15]. Multiple hyperplanes may effectively divide the groups; the one used is the one along which the largest normal distance is found for any of the data points [9]. SVMs, or support vector machines, are one sort of supervised learning model with related learning algorithms for conducting classification and regression analysis on data [14]. The RBF kernel was chosen for this study among the several widely-used SVM kernels.

$$K(x, x') = e^{-\gamma||x - x'||^2}$$

Here, ||x − x′||2 is the squared Euclidean distance between two feature vectors (2 points). In our case, in BoW we have used the SVM RBF function kernel with a gamma value of 0.1 which gives better accuracy than other kernels and gamma values. In TF-IDF we have also used the SVM RBF function kernel with a gamma value of 0.5 which provides better performance.

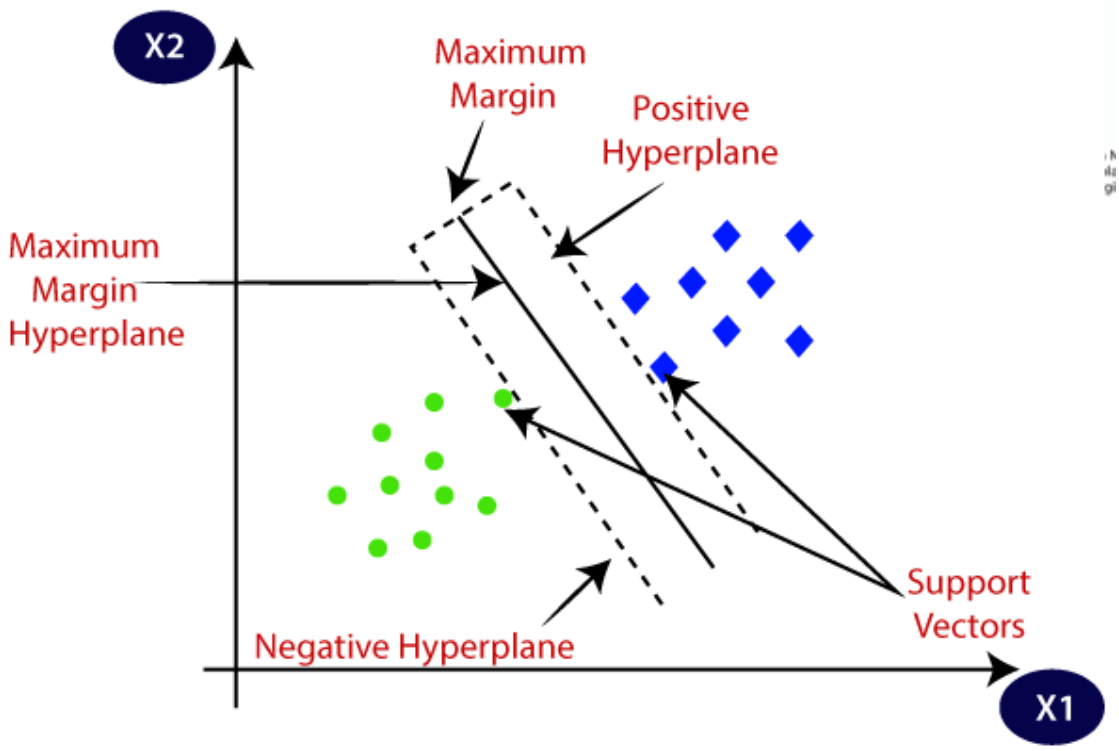


Figure4.3: Support Vector Machine (SVM) Algorithm

Logistic Regression

Logistic Regression is a widely used machine learning algorithm for binary classification tasks. It predicts the probability of a data point belonging to one of two classes using the logistic (sigmoid) function. The sigmoid function maps any real-valued input into a range between 0 and 1, making it ideal for classification tasks where outputs are probabilities.

Logistic regression estimates the probability P (y = 1|x), where y is the target variable (y ∈ {0, 1} and x is the input feature vector. The model outputs probabilities between 0 and 1, which can then be thresholder (e.g., at 0.5) to make class predictions

The model predicts the probability $P(y = 1|x)$ using the sigmoid function:

$$P(y = 1|x) = \frac{1}{1 + e^{-z}}$$

where:

$$z = w \cdot x + b$$

- w: Weight vector.
- x: Input feature vector.
- b: Bias term.
- e: Base of the natural logarithm.

For binary classification, the decision boundary is determined by:

$$\hat{y} = \begin{cases} 1 & \text{if } P(y = 1|x) \geq 0.5 \\ 0 & \text{if } P(y = 1|x) < 0.5 \end{cases}$$

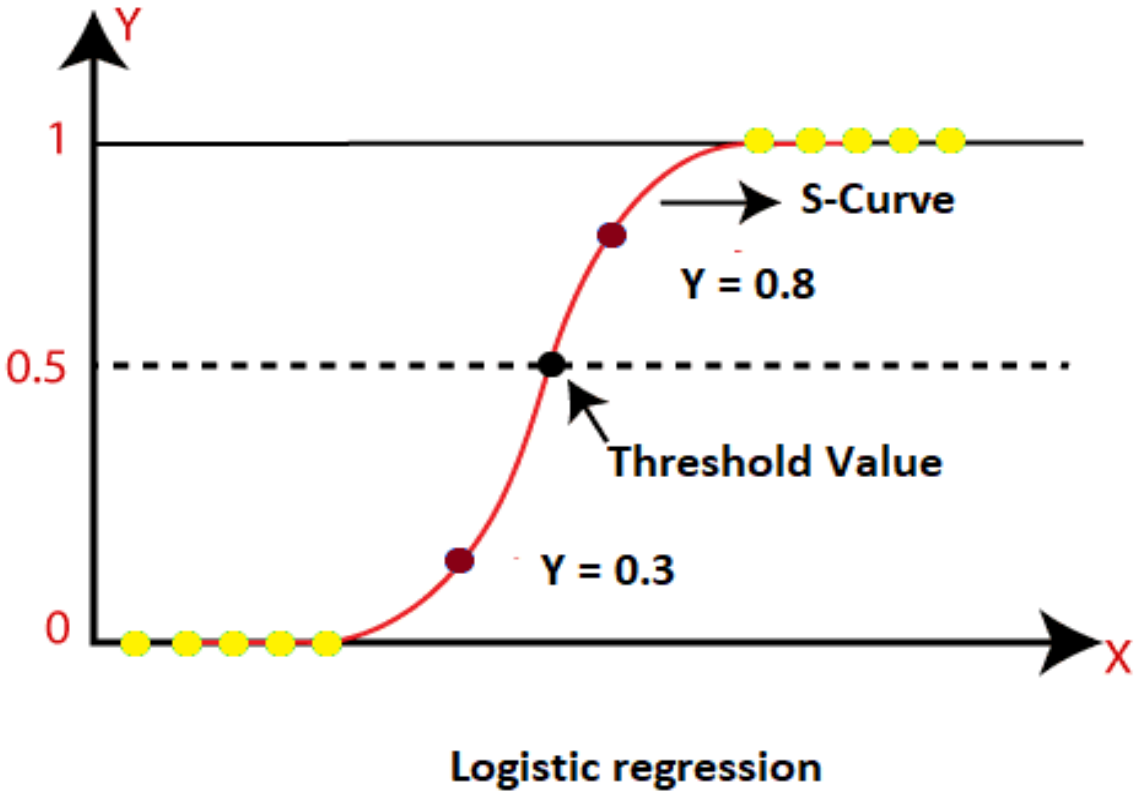


Figure4.4: logistic regression function

Passive-Aggressive Classifier

The Passive-Aggressive Classifier is an online learning algorithm primarily designed for text classification tasks, especially in scenarios involving large-scale, high-dimensional data. Its name reflects its unique learning strategy—remaining passive when the prediction is correct and acting aggressively to update the model when the prediction is incorrect or the decision boundary margin is insufficient. This behavior makes it efficient for tasks where data arrives in a stream or requires real-time processing.

The core principle of the Passive-Aggressive Classifier is to minimize a specific loss function called **hinge loss**, which is commonly used in margin-based classifiers like Support Vector Machines (SVM). When a prediction error occurs, the model updates the weights to correct the mistake and ensure future predictions are more accurate for similar data points. However, unlike traditional gradient-based methods, it uses a dynamically computed learning rate based on the magnitude of the hinge loss, eliminating the need for manual tuning of hyper parameters.

In practice, the model predicts the output label by taking the dot product of the weight vector and the input feature vector, followed by a sign function to determine the class label. If the prediction matches the actual label and satisfies the margin condition, the model does not update. When the prediction is incorrect, or the margin condition is violated, the algorithm adjusts the weights and bias proportionally to the extent of the error. This targeted update mechanism ensures the model

is both computationally efficient and effective in handling sparse, high-dimensional datasets, which are typical in text-based applications.

The Passive-Aggressive Classifier is particularly useful in applications like spam detection, sentiment analysis, and news categorization, where data is often text-heavy and requires fast processing. While it performs well with balanced datasets, it may struggle with imbalanced data, necessitating techniques like class weighting to improve performance. Additionally, it does not inherently produce probabilistic outputs, which can limit its interpretability in certain use cases. However, its simplicity, scalability, and ability to adapt quickly to new data make it a valuable tool in the realm of machine learning.

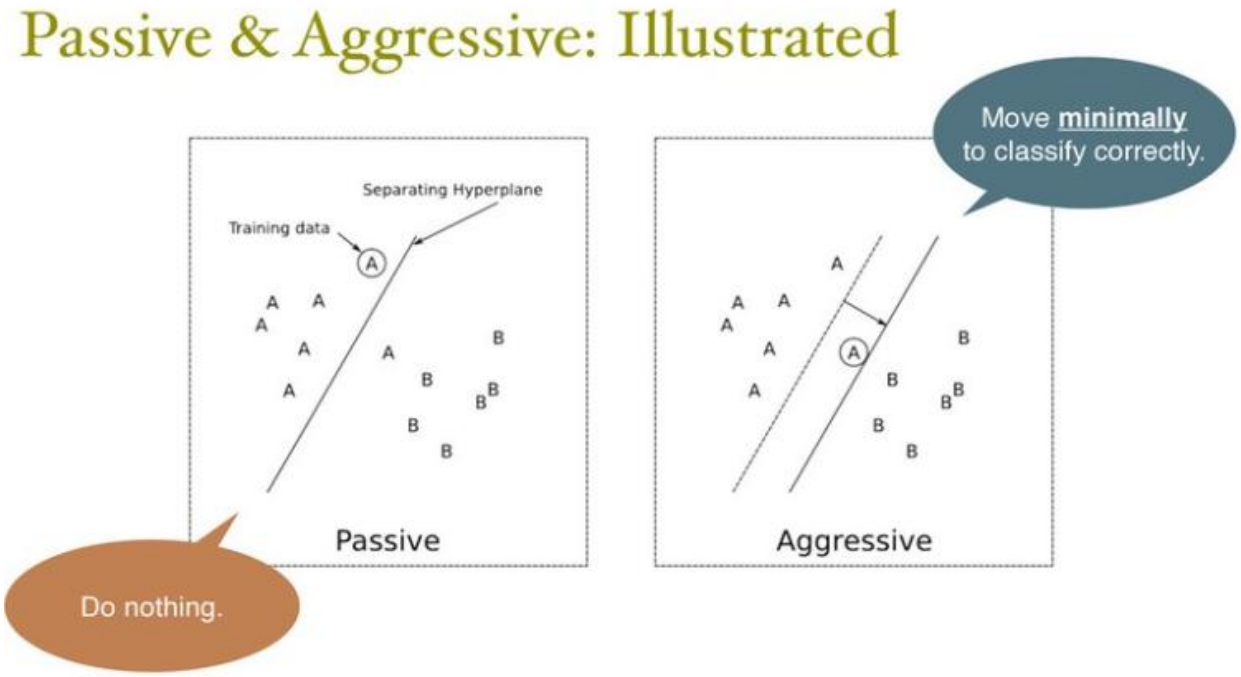


Figure4.5: Detection of Textual Propaganda for PAC

CHAPTER 5
RESULTS AND ANALYSIS

At the beginning of the sentiment analysis task we randomly split the whole dataset into 2 parts for training and testing purposes. We split into an 80:20 ratios to the whole dataset randomly. Training set: The training data set is fed to the model. The model learns from it and learns any hidden characteristics or patterns. We used 80% data of the whole dataset for training. At the beginning of the sentiment analysis task, we randomly split the dataset into **training** and **testing** sets, following an **80:20 ratios**. This split ensures a balanced approach to training the model and evaluating its performance. Below is the data distribution:

Training Set

The training set, comprising **80% of the dataset**, includes **6787 positive** and **2658 negative** comments. This data is used to train the model, enabling it to identify patterns and characteristics of different sentiments in the comments.

Testing Set

The testing set, representing **20% of the dataset**, contains **1713 positive** and **649 negative** comments. This set is reserved for evaluating the model after training is complete. It provides an unbiased measure of the model's performance on unseen data.

Data Split	Positive	Negative	Total
Train	6787	2658	9445
Test	1713	649	2362

Table5.1: Train and Test data classifier

Total Data Counts:

- Positive: $6787 + 1713 = 8500$
- Negative: $2658 + 649 = 3307$
- Overall Total: $9445 + 2362 = 11,807$

Percentage Split:

- Training: $\frac{9445}{11807} \times 100 = 80.0\%$
- Testing: $\frac{2362}{11807} \times 100 = 20.0\%$

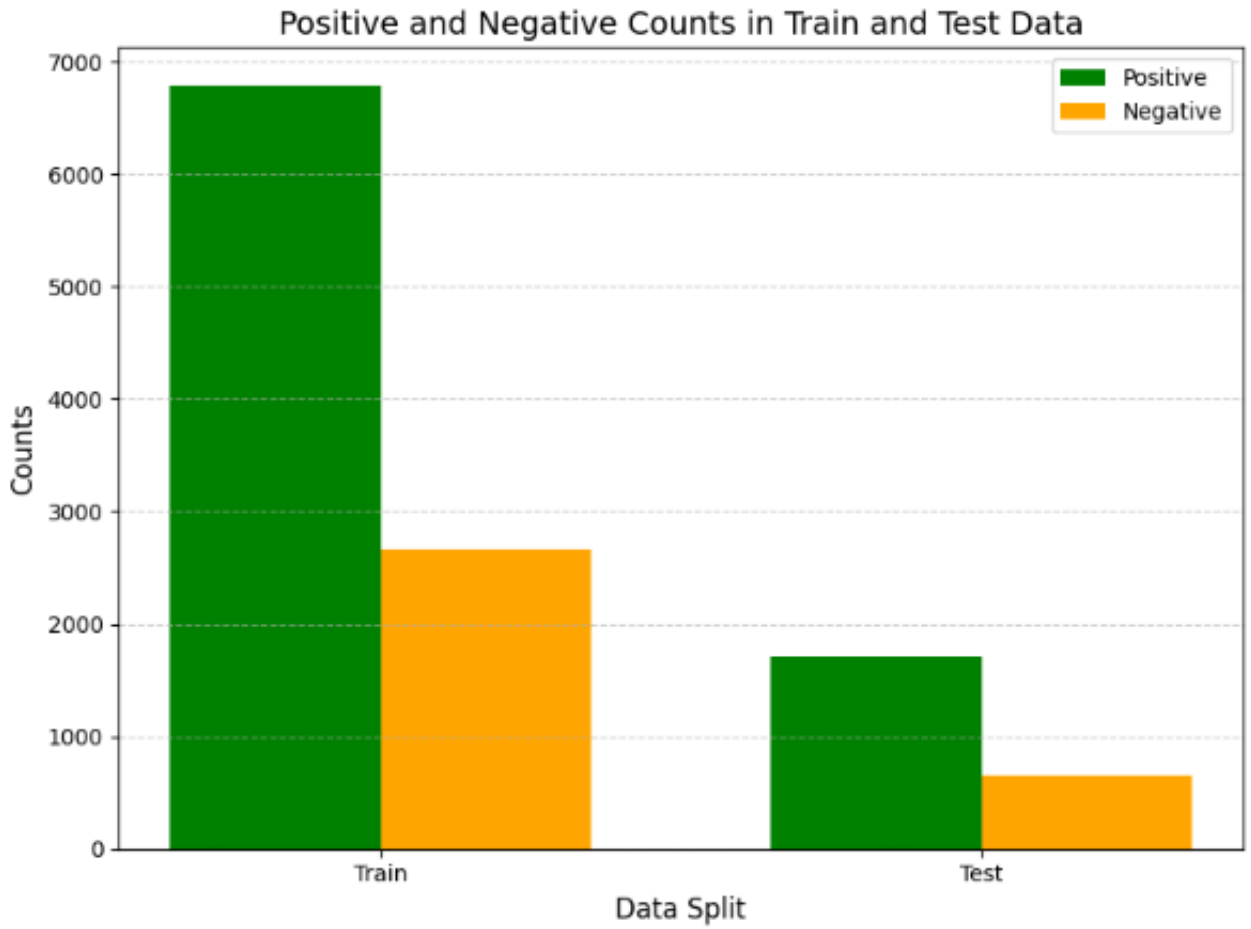


Figure5.1: Train and Test Data Visual classifier

5.1 Logistic Regression

The results indicate the performance of a logistic regression model for classifying comments into positive and negative sentiments. The model achieved an accuracy of 93.14%, meaning it correctly predicted the sentiment of comments in most cases. Precision for the negative class is 93%, indicating that most predictions of negative comments were accurate, while precision for the positive class is also 93%. Recall for the negative class is slightly lower at 81%, showing that the model struggled to identify some negative comments correctly. In contrast, recall for the positive class is much higher at 98%, indicating strong performance in identifying positive comments.

The F1-score, which balances precision and recall, is 0.87 for the negative class and 0.95 for the positive class. The macro-average F1-score across both classes is 0.91, while the weighted average F1-score, adjusted for class imbalance, is 0.93.

Accuracy: 0.9314140558848434
Classification Report:

	precision	recall	f1-score	support
negative	0.93	0.81	0.87	649
positive	0.93	0.98	0.95	1713
accuracy			0.93	2362
macro avg	0.93	0.89	0.91	2362
weighted avg	0.93	0.93	0.93	2362

Table5.2: Classification Report for Logistic Regression

The confusion matrix provides further insight. Out of 649 actual negative comments, 530 were correctly identified, while 120 were misclassified as positive. Out of 1,713 actual positive comments, 1,700 were correctly identified, and only 41 were misclassified as negative. This suggests that the model is highly effective in identifying positive comments but less effective in capturing negative ones, as shown by the disparity in recall between the two classes.

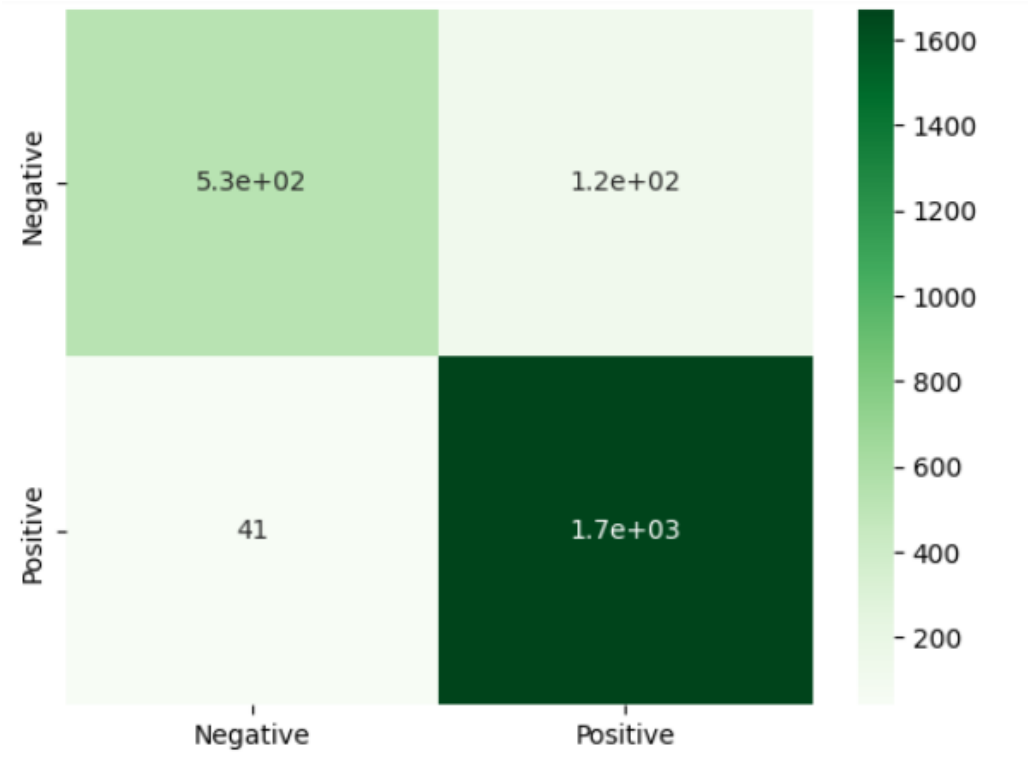


Figure5.2: Confusion Matrix for Logistic Regression

5.2 Support Vector Machine (SVM)

The classification report further elaborates on the model's performance metrics. For the negative class, the precision of 0.89 implies that when the model predicts a comment as negative, 89% of the time it is correct. The recall of 0.87 shows that the model successfully captures 87% of all actual negative comments. The F1-score of 0.88 balances the trade-off between precision and recall, demonstrating consistent performance for the negative class. In the case of the positive class, the precision of 0.95 means the model's positive predictions are 95% accurate. A recall of 0.96 indicates that 96% of all true positive comments are correctly identified. The F1-score for the positive class is 0.96, reflecting its excellent capability in handling positive sentiments.

The overall accuracy of the SVM model is 94%, signifying that it correctly classifies the sentiments of 94% of the total comments in the dataset. The macro-average F1-score of 0.92, which treats both classes equally regardless of their size, underscores robust performance across both categories. The weighted average F1-score, accounting for the imbalance in class distribution, is 0.94, which aligns with the overall accuracy and confirms the model's reliability and precision.

	precision	recall	f1-score	support
negative	0.89	0.87	0.88	649
positive	0.95	0.96	0.96	1713
accuracy			0.94	2362
macro avg	0.92	0.92	0.92	2362
weighted avg	0.94	0.94	0.94	2362

Table5.3: SVM Classification Report

The SVM model's results in the sentiment analysis task highlight its reliability and precision in understanding textual sentiment. By accurately classifying 560 negative comments and 1600 positive comments, the model demonstrates its ability to effectively handle the challenges of sentiment polarity. While there were 84 instances of false positives (negative comments misclassified as positive) and 68 instances of false negatives (positive comments misclassified as negative), these errors represent a small fraction of the overall dataset. This low error rate underscores the model's robustness and adaptability to tasks requiring high levels of accuracy.

In sentiment analysis, such performance is crucial for tasks where even minor errors can have significant consequences, such as analyzing customer feedback to improve products or services. The SVM model's effectiveness in managing these tasks ensures that businesses can rely on it for actionable insights. For example, understanding whether customer reviews lean positive or negative can help companies make data-driven adjustments to enhance satisfaction and loyalty.

Furthermore, the model's high accuracy rate provides assurance for its scalability across different datasets and industries. Sentiment analysis applications can extend to fields such as political opinion tracking, brand reputation management, and even mental health sentiment studies. The combination of low misclassification rates and high true positive and negative identification rates ensures that the SVM model is not only reliable but also versatile in diverse applications.

By consistently minimizing errors while maintaining a strong focus on accurate sentiment classification, the SVM model proves to be a valuable tool for processing large volumes of text data. As sentiment analysis continues to grow in importance for decision-making, models like this are paving the way for more efficient and impactful applications in both academic and professional settings.

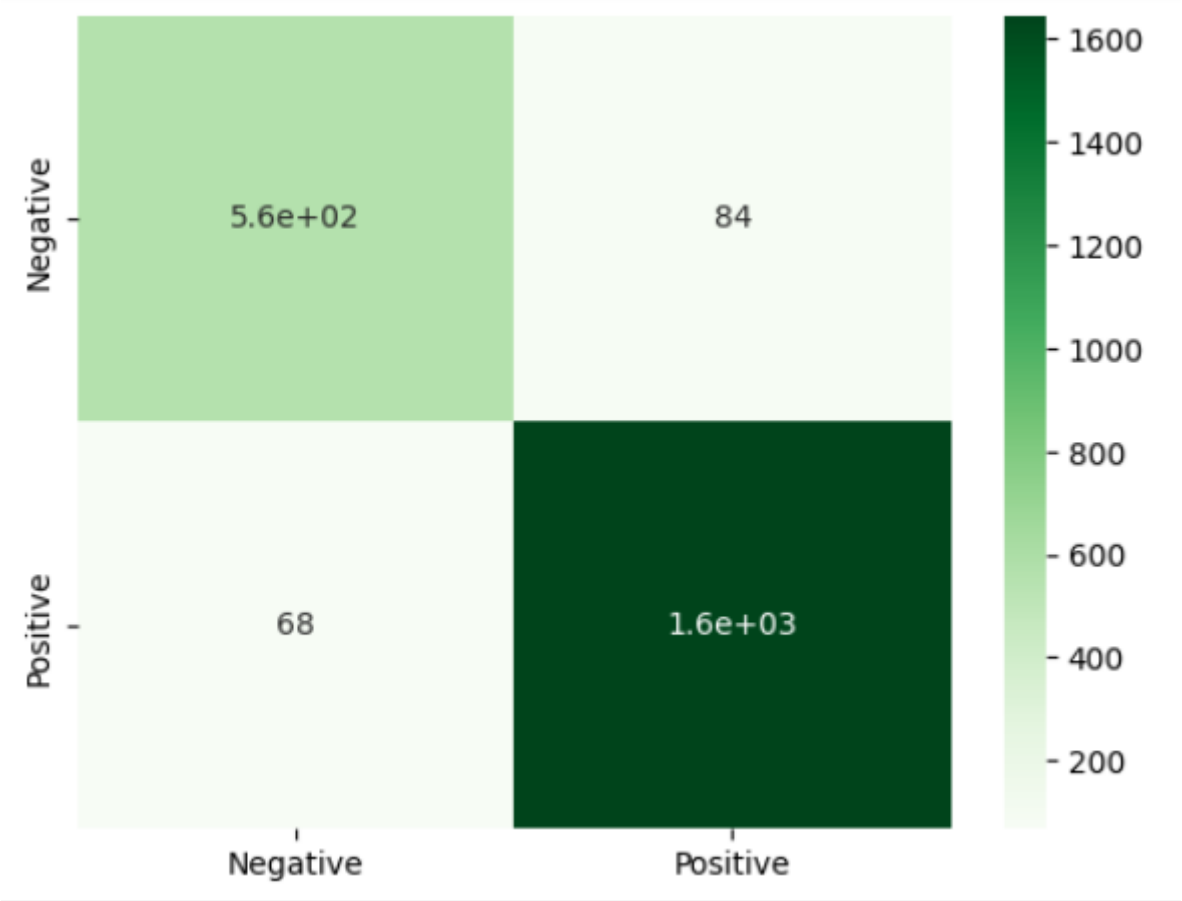


Figure5.3: SVM Confusion Matrix

5.3 Naïve Bayes

The results demonstrate the performance of a Naïve Bayes model in classifying comments into positive and negative sentiments. The overall accuracy of the model is 93.14%, reflecting its effectiveness in correctly predicting sentiment in most cases.

For the negative class, the precision is 93%, meaning that most of the comments predicted as negative were indeed negative. However, the recall for the negative class is lower at 81%, indicating that some negative comments were not correctly identified. In contrast, the positive class shows both high precision and recall at 93% and 98%, respectively, demonstrating strong performance in identifying positive comments.

The F1-score, which balances precision and recall, is 0.87 for the negative class and 0.95 for the positive class. The macro-average F1-score, which treats both classes equally, is 0.91, while the weighted average F1-score, adjusted for the class distribution, is 0.93.

Accuracy: 0.9314140558848434
Classification Report:

	precision	recall	f1-score	support
negative	0.93	0.81	0.87	649
positive	0.93	0.98	0.95	1713
accuracy			0.93	2362
macro avg	0.93	0.89	0.91	2362
weighted avg	0.93	0.93	0.93	2362

Table5.4: Performance Metrics Visualization for Naïve Bayes Model

The confusion matrix further reveals the classification details. Out of 649 actual negative comments, 530 were correctly classified as negative, but 120 were misclassified as positive. For the 1,713 actual positive comments, 1,700 were correctly classified as positive, with only 41 misclassified as negative.

These results indicate that while the Naïve Bayes model is highly effective in identifying positive comments, it has some difficulty in correctly identifying all negative comments. This is evident from the lower recall and F1-score for the negative class compared to the positive class.

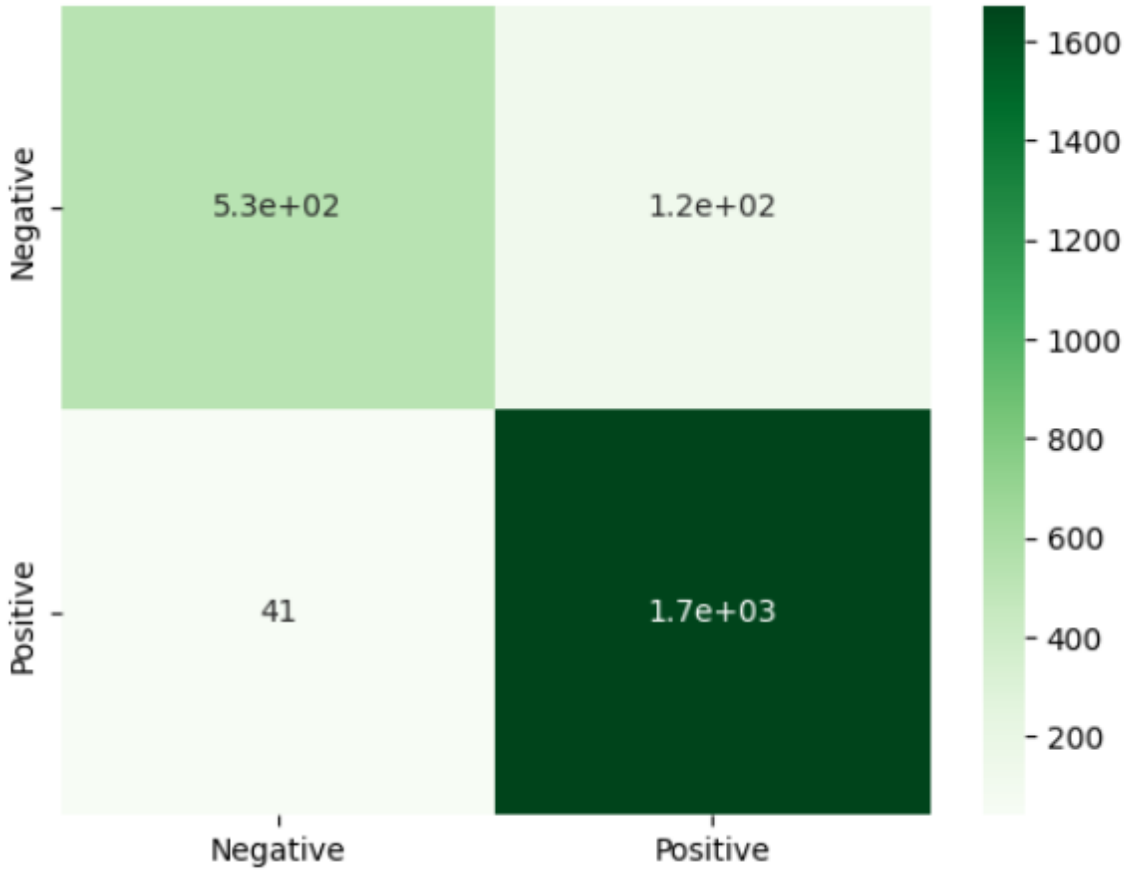


Figure5.4: Sentiment Analysis Confusion Matrix for Naïve Bayes

5.4 Passive Aggressive Classifier

A classification report outlines the metrics for two classes: "negative" and "positive." The "negative" class achieves a precision of 0.85, indicating that 85% of the predicted negative instances were correct. Its recall is 0.86, meaning that 86% of the actual negative instances were correctly identified by the model. The F1-score, which combines precision and recall, is 0.85, reflecting a balanced performance for this class. For the "positive" class, the precision is higher at 0.95, showing a strong ability to correctly predict positive instances. The recall for the positive class is 0.94, slightly lower than its precision but still very high, indicating that the model effectively identifies the majority of actual positives. The F1-score for the positive class is 0.94, suggesting excellent performance.

The overall accuracy of the model is 92%, indicating that a substantial proportion of predictions, across both classes, are correct. When averaging the metrics across the two classes, the macro average scores for precision, recall, and F1-score are all 0.90. The weighted average, which takes into account the imbalance in class support (649 instances for "negative" and 1,713 for

"positive"), results in slightly higher scores of 0.92 for precision, recall, and F1-score. These metrics demonstrate that the model performs exceptionally well, particularly for the "positive" class, which has more instances.

The weighted average, which takes into account the imbalance in class support (649 instances for "negative" and 1,713 for "positive"), results in slightly higher scores of 0.92 for precision, recall, and F1-score. These metrics demonstrate that the model performs exceptionally well, particularly for the "positive" class, which has more instances.

	precision	recall	f1-score	support
negative	0.85	0.86	0.85	649
positive	0.95	0.94	0.94	1713
accuracy			0.92	2362
macro avg	0.90	0.90	0.90	2362
weighted avg	0.92	0.92	0.92	2362

Table5.5: Classification Report for Passive Aggressive Classifier

The **confusion matrix** provides a comprehensive visualization of the model's performance by comparing its predictions with the true labels of the dataset. The matrix is divided into four quadrants, each offering insights into the classifier's strengths and weaknesses. In the **top-left cell**, we see that **560 negative instances** were correctly classified as negative, reflecting the model's ability to accurately identify negative sentiments. The **top-right cell** shows that **88 negative instances** were misclassified as positive, indicating some difficulty in distinguishing negative sentiments that may share characteristics with positive ones.

In the **bottom-right cell**, the model correctly identified **1,600 positive instances** as positive, which signifies a strong performance in recognizing positive sentiments. However, in the **bottom-left cell**, **110 positive instances** were incorrectly classified as negative, suggesting a weakness in detecting subtle positive cues or distinguishing them from negative comments. This misclassification could indicate that the model is possibly overfitting or struggling with ambiguous cases where sentiment may not be clear-cut.

The **heatmap** accompanying the matrix visually represents the frequency of instances in each cell using varying intensities of green. Darker shades of green correspond to higher counts, allowing for quick identification of where the model performs well (e.g., the bottom-right cell for positive predictions) and where it struggles (e.g., the top-right and bottom-left cells for misclassifications). This heatmap not only provides a more intuitive understanding of the classifier's performance but also helps in diagnosing areas that require improvement. By focusing on the misclassified instances, further fine-tuning of the model or exploring alternative techniques, such as adjusting the class weights or incorporating different feature engineering methods, could help in enhancing the model's overall accuracy and reliability.

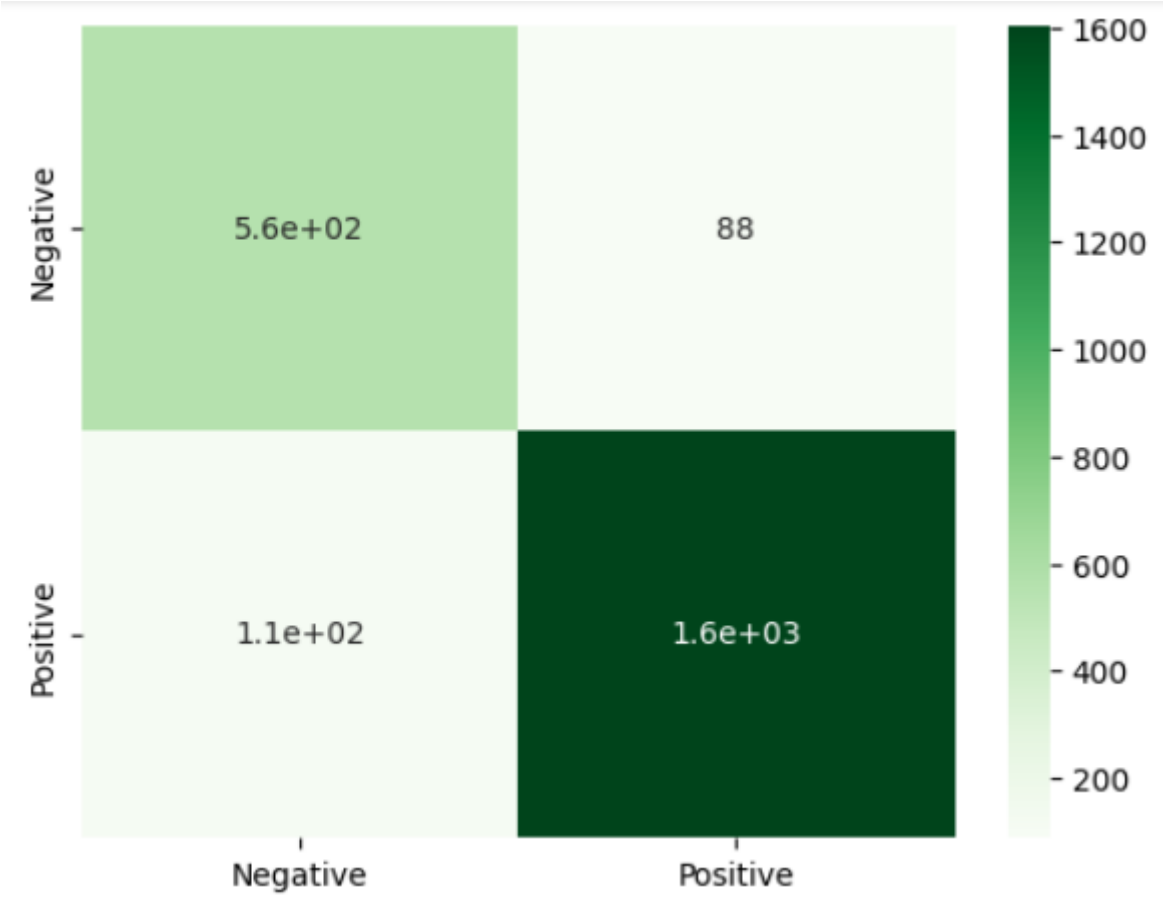


Figure5.5: Confusion Matrix for Passive Aggressive Classifier

5.5 Result Summary

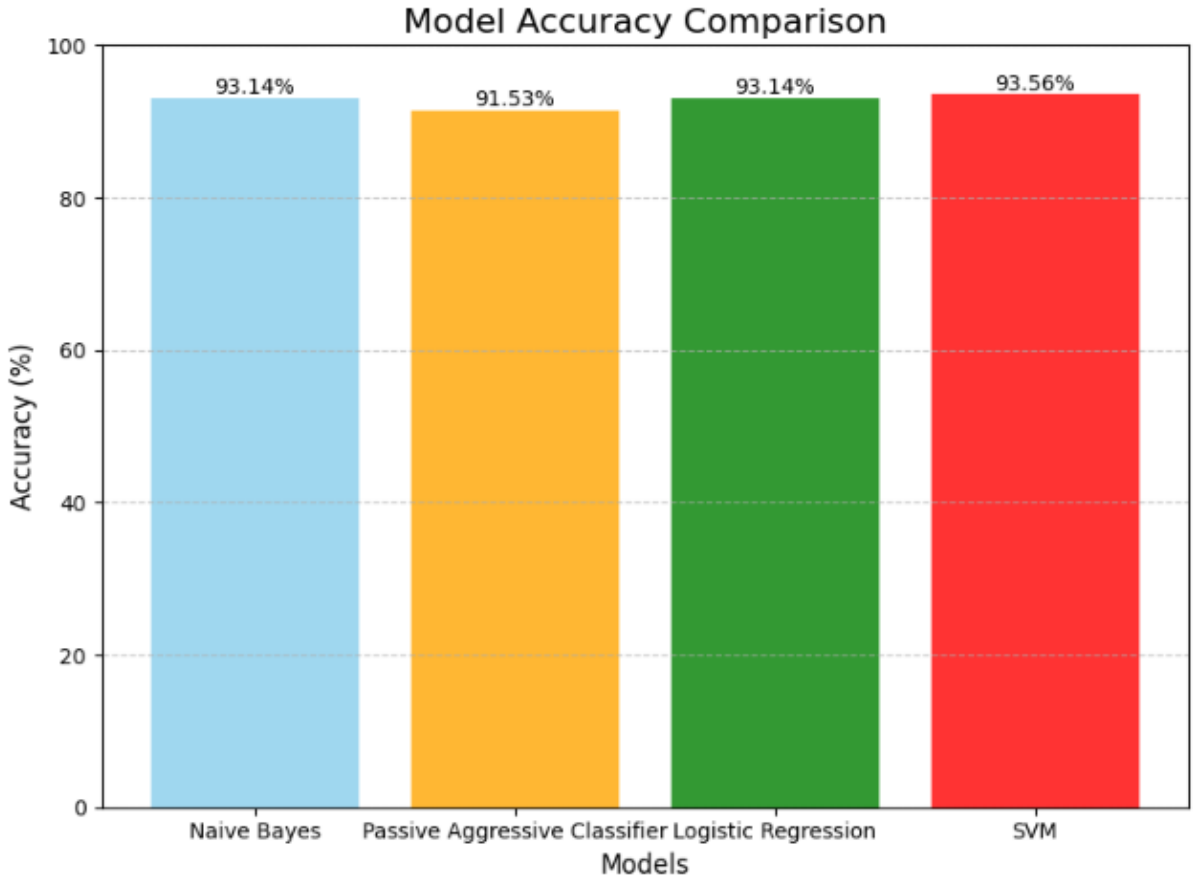


Figure5.6: Comparison Scores of Implemented Models

When comparing the four models—Naive Bayes, Passive Aggressive Classifier, Logistic Regression, and Support Vector Machines (SVM)—based on their accuracy percentages, SVM emerges as the best-performing model with an accuracy of 93.56%, slightly higher than the other models. While the difference in accuracy between SVM (93.56%), Naive Bayes (93.14%), and Logistic Regression (93.14%) is small, it is statistically significant enough to rank SVM as the most accurate model in this evaluation.

The Passive Aggressive Classifier, with an accuracy of 91.53%, performs slightly worse than the other models. This is expected, as Passive Aggressive Classifiers are optimized for online learning and situations with streaming data rather than static datasets. Its slightly lower accuracy suggests that it may not be the optimal choice for this specific task.

The decision to use SVM should also consider computational cost, as SVMs can be more resource-intensive compared to simpler models like Naive Bayes or Logistic Regression.

CHAPTER 6
LIMITATIONS AND FUTURE WORKS

This project demonstrates the successful implementation of a sentiment analysis system using machine learning techniques. By employing four different models—Naive Bayes, Passive Aggressive Classifier, Logistic Regression, and Support Vector Machines (SVM)—the system classifies comments as either positive or negative. The comparative analysis reveals that all models perform with high accuracy, with SVM achieving the best performance at 93.56%, followed closely by Logistic Regression and Naive Bayes at 93.14%, and Passive Aggressive Classifier at 91.53%. These results highlight the capability of machine learning algorithms to effectively classify text data in a structured and efficient manner.

The sentiment analysis achieved in this study provides an automated and scalable solution for identifying the polarity of user comments. This has practical applications across various domains, such as customer feedback analysis, social media monitoring, and product review evaluation. However, the system, in its current state, focuses solely on binary sentiment classification (positive vs. negative), which limits its ability to capture nuanced emotional expressions or more complex sentiments.

The findings affirm the potential of machine learning in sentiment analysis tasks, emphasizing its ability to deliver reliable results. While SVM stands out as the top-performing model, the other models also demonstrate comparable accuracy, making them viable alternatives depending on specific use-case requirements, such as computational efficiency or simplicity. This code forms a robust foundation for developing more advanced and comprehensive sentiment analysis systems in the future.

6.1 Limitations

Despite the strong performance of the models, several limitations exist in the current study:

1. **Dataset Dependence**: The performance of the models depends heavily on the quality and size of the dataset used. Limited data can lead to biased or less generalizable results.
2. **Binary Classification**: The focus of this study was restricted to binary classification (positive vs. negative), which limits the granularity of sentiment analysis.
3. **Emotion Analysis**: The models do not currently take into account nuanced emotions such as happiness, sadness, fear, or anger, which could provide deeper insights into user sentiment.
4. **Context Understanding**: The models lack the ability to understand sarcasm, irony, or context, which can affect the accuracy of sentiment classification.
5. **Real-Time Processing**: The study does not incorporate real-time data processing or online learning, which limits its applicability in dynamic environments.

6.2 Future work

To enhance the capabilities and applications of this project, several directions for future work can be pursued:

1. **Multiclass Sentiment Analysis**: Extend the classification to include a wider range of sentiments, such as neutral, mixed, or emotions like happiness, sadness, fear, and anger.
2. **Emotion Detection**: Incorporate emotion recognition by leveraging advanced natural language processing (NLP) techniques and pre-trained models like BERT or GPT. This will enable the system to classify not only sentiments but also the emotional states of users.
3. **Real-Time Processing**: Implement real-time comment analysis to dynamically classify comments as they are posted, improving usability in live applications.
4. **Contextual Understanding**: Enhance the model's ability to detect sarcasm, irony, and complex language structures using transformers and deep learning methods.
5. **Cross-Language Support**: Expand the system to analyze comments in multiple languages using multilingual NLP models, broadening its applicability.
6. **Integration with Emotion Sensors**: Integrate the system with facial recognition or voice emotion detection technologies to provide a multimodal approach to sentiment and emotion analysis.
7. **Applications in Diverse Sectors**: The enhanced system can be applied in various industries:
    - **Social Media Monitoring**: Automatically monitor user feedback and identify emotional trends in social media platforms.
    - **Customer Support**: Analyze customer reviews or complaints to understand sentiment and improve service.
    - **Education**: Use sentiment and emotion analysis to gauge student engagement or feedback.
    - **Healthcare**: Assist in mental health analysis by identifying emotional states in text data from patients.
    - **Human Resources**: Analyze employee feedback to assess organizational well-being.
    - **Entertainment**: Analyze audience reactions to content such as movies, shows, or events.
    - **Political Campaigning**: Gauge public sentiment toward policies or campaigns in real time.

By advancing the project in these directions, it can evolve into a professional and robust sentiment and emotion analysis tool, capable of serving multiple industries and addressing a wide range of real-world applications.

REFERENCES

[1] T.Dalgleish and M.Power, *Handbook of cognition and emotion*. JohnWiley &Sons,2000.

[2] E. Cambria and B. White Jumping NLP curves: A review of natural language processing research *IEEE Computational intelligence magazine*, jourvol9, number 2, pages 48–57,2014.

[3] R. E. Jack, O. G. Garrod and P. G. Dynamic facial expressions of emotion transmit an evolving hierarchy of signal saver time. *Current biology*, jourvol 24, number 2, pages 187–192,2014.

[4] F.Sadeque, T.Solorio,T.Pedersen,P.Shrestha and S.Bethard, Predicting continued participation in online health forums In *Proceedings of the Sixth International Workshop on Health Text Mining and Information Analysis* 2015, pages 12–20.

[5] C. Zhou, C. Sun, Z. Liu and F. Lau, A C-LSTM neural network for text classification, *arXiv preprint arXiv:1511.08630*,2015.

[6] J.L.Ba, J.R.Kiros and G.E.Hinton, Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[7] A. Hassan, M. R. Amin, A. K. Al Azad and N. Mohammed, Sentiment analysis on Bangla and Romanized Bangla text using deep recurrent models, in *2016 International Workshop on Computational Intelligence (IWCI)* IEEE, 2016, pages 51–56.

[8] K.He, X.Zhang, S.Ren and J.Sun, Deep residual earning for image recognition, in *Proceedings of the IEEE conference on computer vision and pattern recognition* 2016, pages 770–778.

[9] A. P. Jain and P. Dandannavar, Application of machine learning technique stop sentiment analysis, in *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)* IEEE, 2016, pages 628–632.

[10] A. Vaswani, N. Shazeer, N.Parmar and others, Attention isall you need. *Advances in neural information processing systems*, jourvol 30, 2017.

[11] J. Devlin, M.-W. Chang, K. Lee and K. Tout nova. Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint arXiv:1810.04805*,2018.