

Sentiment Analysis of Social Media Presence Using DistilBert

A PROJECT REPORT

Submitted by,

Mr. Tejas M **20211CSD0139**

Mr. Akash S **20211CSD0011**

Mr. Ramanujam D K **20211CSD0080**

Mr. Bhuvan Cariappa B D 20211CSD0130

Under the guidance of,

Dr. Marimuthu K

Professor

Presidency University, Bengaluru

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING (DATA SCIENCE)

AT



PRESIDENCY UNIVERSITY, BENGALURU

MAY - 2025

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project report “**Sentiment Analysis of Social Media Presence Using DistilBert** ” being submitted by “Tejas M, Ramanujam D K , Akash S, Bhuvan Cariappa B D” bearing roll number “20211CSD0139”, “20211CSD0080”, “20211CSD0011”, “20211CSD0130” in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a Bonafide work carried out under my supervision.

Dr. Marimuthu K

Professor

School of CSE

Presidency University, Bengaluru

Dr. Saira Banu Atham

Professor & HoD

School of CSE

Presidency University, Bengaluru

Dr. MYDHILI K NAIR

Professor & Associate Dean

School of CSE & IS

Presidency University, Bengaluru

Dr. SAMEERUDDIN KHAN

Pro-VC School of Engineering

Dean – School of CSE & IS

Presidency University, Bengaluru

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the report entitled **“SENTIMENT ANALYSIS OF SOCIAL MEDIA PRESENCE USING DISTILBERT”** in partial fulfillment for the award of Degree of **Bachelor of Technology** in Computer Science and Engineering is a record of my own investigations carried under the guidance of **Dr. Marimuthu K, Professor**, School of Computer Science and Engineering, Presidency University, Bengaluru.

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Tejas M	20211CSD0139
----------------	---------------------

Akash S	20211CSD0011
----------------	---------------------

Ramanujam D K	20211CSD0080
----------------------	---------------------

Bhuvan Cariappa B D	20211CSD0130
----------------------------	---------------------

ACKNOWLEDGEMENTS

We humbly express our gratitude to **Almighty God** for blessing us with the ability and determination to complete this project on time.

We express our sincere thanks to our respected dean **Dr. M D Sameeruddin Khan**, Pro - VC and Dean of, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Dean **Dr. Mydhili Nair**, Presidency School of Computer Science and Engineering, Presidency University, and **Dr. Saira Banu Atham**, Head of the Department, Presidency School of Computer Science and Engineering, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr. Marimuthu K, Professor** and Reviewer **Mr. Yamanappa, Asst.Professor**, Presdiency School of Computer Science and Engineering, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the internship work.

We would like to convey our gratitude and heartfelt thanks to the CSE7301 University Project Coordinator **Mr. Md Ziaur Rahman and Dr. Sampath A K**, department Project Coordinators **Dr. Manjula H M** and Git hub coordinator **Mr. Muthuraj K**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Tejas M

Akash S

Ramanujam D K

Bhuvan Cariappa B D

ABSTRACT

The rise of social media has transformed the way individuals and organizations interact with the public, making online reputation management more critical than ever. This project aims to develop a sentiment analysis solution specifically designed to analyze the sentiment expressed in social media presence. Sentiment analysis is a natural language processing (NLP) technique used to determine the emotional tone conveyed by text or speech, classifying it as positive, negative, or neutral. By examining social media posts, comments, and interactions, sentiment analysis provides valuable insights into public perception, customer feedback, and brand reputation.

The ability to track and interpret sentiment trends helps individuals and businesses identify potential concerns, address negative feedback, and enhance engagement strategies. This project will leverage machine learning and deep learning models, including Support Vector Machines (SVM), Naïve Bayes, and transformer-based architectures like BERT, to improve the accuracy of sentiment classification. Additionally, challenges such as sarcasm detection, contextual understanding, and misinformation filtering will be addressed to enhance the reliability of the analysis.

Real-time data from platforms like Twitter, Facebook, and Reddit will be used to train and validate the models, ensuring their applicability in real-world scenarios. Ethical considerations, including data privacy and bias mitigation, will be a key focus to ensure responsible AI implementation. The findings of this study will not only aid businesses in monitoring their brand image but also assist individuals in managing their digital presence effectively. By developing an efficient and scalable sentiment analysis solution, this research contributes to the growing field of social media analytics and reputation management.

LIST OF TABLES

Sl. No.	Table Name	Title of Tables	Page No.
1	Table 1.1	Relevance of Sentiment Analysis of Social media	3
2	Table 2.1	Limitation of Sentiment Analysis of Social media	9
3	Table 3.1	Regulatory and Industrial Compliance	15
4	Table 4.1	Integration Dimension Relevance Sentiment Analysis System	21
5	Table 5.1	Comparison of Machine Learning Models for Sentiment Analysis	25
6	Table 6.1	Comparison of Transformer Models for Event Detection	33
7	Table 8.1	Structured Overview of Outcome Domains in Sentiment Analysis	38

LIST OF FIGURES

Sl. No.	Figure Name	Figure Caption	Page No.
1	Figure 2.1	Limitation of Existing System	8
2	Figure 4.1	System Workflow	17
3	Figure 5.1	Sentiment Analysis Tasks	22
4	Figure 6.1	Architecture of Sentiment Analysis	27
5	Figure 6.2	DistilBERT Flow chart	28
6	Figure 6.3	DistilBERT model architecture	29
7	Figure 7.1	Timeline of Execution of Project	35
8	Figure 7.2	Gantt Chart	35
9	Figure 8.1	Outcomes of Sentiment Analysis	36
10	Figure 8.2	Strategic Cost Optimization and Competitive Insights Overview	39
12	Figure 9.1	Dataset Description	41
13	Figure 9.2	Installing and Importing libraries	42
14	Figure 9.3	Pre-Process Data	43
15	Figure 9.4	Tokenization	43
16	Figure 9.5	Load Model and Define Metrics	44
17	Figure 9.6	Fine-Tuning the model	44
18	Figure 9.7	Evaluate the model	45
19	Figure 9.8	Example Prediction	46

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO.
	CERTIFICATE	ii
	DECLARATION	iii
	ACKNOWLEDGEMENT	iv
	ABSTRACT	v
	LIST OF TABLES	vi
	LIST OF FIGURES	vii
	TABLE OF CONTENT	viii
1.	INTRODUCTION	1-5
	1.1 General Overview	1
	1.2 Importance of Sentiment Analysis of Social Media	2
	1.3 Relevance and Problem Statement	3
	1.4 Scope of the Project	4
2.	LITERATURE REVIEW	6-10
	2.1 General Overview	6
	2.2 Related Works	7
	2.3 Limitations of Existing Systems	8
3.	RESEARCH GAPS OF EXISTING METHODS	11-16
	3.1 Identified Gaps	11
	3.2 Need for Proposed System	12
	3.3 Data Availability and Quality	14
	3.4 Regulatory and Industry Compliance	15
4.	PROPOSED METHODOLOGY	17-21
	4.1 System Overview	17
	4.2 System Workflow	17
	4.3 Adaptability and Scalability	19
	4.4 Integration with Existing System	21
5.	OBJECTIVES	22-26
	5.1 Primary Objectives	22
	5.2 Advanced Machine Learning Techniques for Sentiment Analysis	24
	5.2.1 Introduction	24

5.2.2	Machine Learning Models for Sentiment Analysis	24
5.2.3	Advantage of Advanced ML Techniques	25
5.3	Sentiment Analysis for Crisis and Brand Management	26
5.3.1	Role of Sentiment Analysis in Crisis Management	26
5.3.2	Sentiment Analysis for Brand Management	26
6.	SYSTEM DESIGN AND IMPLEMENTATION	27-34
6.1	System Architecture Overview	27
6.2	Key Components of System Architecture	30
6.2.1	Data Collection and Preprocessing	30
6.2.2	Model Selection and Forecasting	31
6.2.3	Model Evaluation	32
6.2.4	System Deployment or Integration	34
7.	TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)	35-40
8.	OUTCOMES	36
8.1	Strategic Impacts and Key Applications	36
8.2	Structured Overview of Outcome Domains in Sentiment Analysis	38
8.3	The real-world value like automation	39
9.	RESULTS AND DISCUSSIONS	41-46
9.1	Dataset Description	41
9.2	Final Result	41
10.	CONCLUSION	47-48
11.	REFERENCES	49-50
12.	APPENDIX – A (PSUEDOCODE)	51-53
	APPENDIX – B (SCREENSHOTS)	54-58
	APPENDIX – C (ENCLOSURES)	59-64

Chapter 1

INTRODUCTION

1.1 General Overview

In the digital age, social media has become a dominant platform for communication, influencing public opinion and shaping reputations. Individuals and organizations alike rely on social media for engagement, marketing, and brand management. However, with the vast amount of user-generated content on platforms like Twitter, Facebook, and Reddit, understanding the sentiment behind posts, comments, and discussions has become essential. Sentiment analysis, also known as opinion mining, is a field of natural language processing (NLP) that focuses on determining the emotional tone of text data. This process involves categorizing sentiments as positive, negative, or neutral, providing valuable insights into public perception.

Sentiment analysis plays a crucial role in various applications, including brand monitoring, customer feedback analysis, political opinion tracking, and crisis management. Businesses use sentiment analysis to understand customer emotions, assess product reception, and refine marketing strategies. Similarly, individuals and public figures can analyse sentiment trends to manage their online reputation effectively. The analysis of social media sentiment enables proactive decision-making by identifying potential issues, emerging trends, and public concerns.

The implementation of sentiment analysis involves multiple techniques, ranging from traditional machine learning models like Naïve Bayes and Support Vector Machines (SVM) to advanced deep learning approaches such as Recurrent Neural Networks (RNN) and transformer-based models like BERT. These methods rely on text pre-processing techniques, including tokenization, stop-word removal, and lemmatization, to enhance sentiment classification accuracy. However, challenges such as sarcasm detection, language ambiguity, and contextual meaning remain significant hurdles in achieving precise sentiment analysis.

The increasing reliance on social media for communication, marketing, and public discourse has made sentiment analysis a crucial tool for individuals, businesses, and policymakers. Social media platforms generate massive volumes of data every second, including opinions, reviews, and discussions. This data holds significant value in understanding public attitudes toward various topics, ranging from consumer products and brand reputation to political campaigns and social issues. Sentiment analysis, therefore, serves as a bridge between raw textual data and actionable insights by determining the underlying emotions and opinions expressed in posts, comments, and interactions.

In conclusion, sentiment analysis is a powerful tool that continues to transform how businesses, governments, and individuals interact with the digital world. By leveraging artificial intelligence and NLP, sentiment analysis enables deeper insights into public opinion, helping organizations make informed decisions and engage with audiences more effectively. However, addressing the challenges of accuracy, bias, and ethical considerations remains essential for its responsible and effective implementation.

1.2 Importance of Sentiment Analysis of Social Media Presence

In today's digital era, social media has become a crucial platform for individuals, businesses, and organizations to interact with the public. Millions of posts, comments, and discussions occur daily, reflecting public opinions, emotions, and reactions. Understanding these sentiments is vital for individuals and brands to manage their online reputation effectively. Sentiment analysis, a branch of natural language processing (NLP), enables the automated interpretation of these sentiments, classifying them as positive, negative, or neutral. This capability plays a significant role in multiple domains, including brand management, customer service, market analysis, and crisis management.

For businesses, sentiment analysis provides valuable insights into customer satisfaction and brand perception. By analyzing feedback from social media platforms like Twitter, Facebook, and Instagram, companies can identify areas of improvement, address customer concerns, and tailor their marketing strategies accordingly. Positive sentiment

trends can help businesses reinforce successful strategies, while negative sentiments alert them to potential risks, allowing them to take proactive measures to improve customer experience.

Individuals, especially public figures and influencers, also benefit from sentiment analysis. Monitoring online sentiment helps them understand how their audience perceives their content, opinions, or personal brand. This awareness allows them to adjust their messaging and engagement strategies to maintain a favorable online presence. Similarly, organizations and political entities leverage sentiment analysis to track public opinion on policies, campaigns, and events, enabling them to respond effectively to public concerns and shape communication strategies.

1.3 Relevance and Problem Statement

Relevance	Sentiment Analysis of Social Media Presence
Social Media Influence	Social media shapes public opinion, brand perception , and political trends.
Business Applications	Helps companies monitor customer feedback, brand health and market trends.
Research Importance	Valuable for sociological, psychological, and market research studies.
Real-Time Insights	Enables organizations to respond quickly to public sentiment or crises.
Language & Context Understanding	Advances in NLP allow better analysis of complex and sarcastic language.

Table no 1.1 Domains Impacted by Social Media Sentiment Analysis

As shown in the table no 1.1, social media plays a crucial role across various domains by influencing public opinion, shaping brand perception, and impacting political trends. Businesses leverage social media analytics to monitor customer feedback, assess brand health, and track market trends in real time. From a research perspective, social media

data is invaluable for sociological, psychological, and market studies, offering insights into human behavior and societal shifts. Real-time analysis further empowers organizations to swiftly respond to public sentiment or crises, minimizing potential reputational damage. Additionally, advancements in natural language processing (NLP) have enhanced the ability to understand complex expressions and sarcasm, enabling deeper and more accurate sentiment analysis.

The relevance of sentiment analysis in social media extends across multiple domains. Businesses leverage sentiment analysis to monitor brand perception, respond to customer feedback, and refine marketing strategies. Political entities use it to assess public reactions to policies, campaigns, and speeches. Additionally, sentiment analysis is crucial for media and entertainment industries to understand audience engagement and preferences. Given the increasing influence of social media on decision-making, the ability to extract meaningful insights from online interactions has become more critical than ever.

Despite its significance, sentiment analysis faces several challenges. Social media content is highly dynamic, often containing slang, sarcasm, and context-dependent meanings that traditional sentiment analysis models struggle to interpret accurately. Additionally, misinformation, biased opinions, and fake news can skew sentiment analysis results, leading to misleading insights. Another key issue is the ethical aspect of data collection, as privacy concerns and potential biases in AI models must be addressed to ensure fairness and accuracy in sentiment classification.

1.4 Scope of the Project

The scope of this project focuses on developing an efficient sentiment analysis system tailored for analysing the social media presence of individuals and organizations. With the rapid growth of digital interactions, understanding public sentiment on platforms such as Twitter, Facebook, Instagram, and Reddit has become essential. This project aims to extract, process, and classify social media text data to determine whether sentiments are positive, negative, or neutral, providing valuable insights for decision-making.

The project will encompass various stages, including data collection, pre-processing, model selection, implementation, and evaluation. Social media data will be gathered using APIs and web scraping techniques while ensuring compliance with ethical guidelines and data privacy regulations. The pre-processing phase will involve techniques such as tokenization, stop-word removal, stemming, and lemmatization to clean and structure the data for sentiment analysis.

For sentiment classification, machine learning and deep learning techniques will be explored. Traditional models like Naïve Bayes and Support Vector Machines (SVM) will be compared with advanced deep learning models such as Long Short-Term Memory (LSTM) networks and transformer-based architectures like BERT (Bidirectional Encoder Representations from Transformers). The performance of these models will be evaluated using key metrics such as accuracy, precision, recall, and F1-score to ensure reliability and efficiency.

The project will also address challenges such as sarcasm detection, contextual ambiguity, and sentiment shifts over time. Additionally, it will focus on multilingual sentiment analysis, enabling the system to analyze sentiments expressed in different languages. The findings from this research will benefit businesses, influencers, political analysts, and public relations teams in monitoring brand perception, public opinion, and crisis management strategies.

In conclusion, this project aims to develop a scalable and effective sentiment analysis system that enhances the understanding of public sentiment in social media interactions. The insights gained will help stakeholders make informed decisions, improve engagement strategies, and respond proactively to emerging trends in the digital landscape.

Chapter 2

LITERATURE SURVEY

2.1 General Overview

Sentiment analysis has been widely studied in the field of natural language processing (NLP) and artificial intelligence, particularly in the context of social media. Several research studies have explored different techniques and models to improve the accuracy and efficiency of sentiment classification. Early approaches to sentiment analysis relied on lexicon-based methods, where predefined word lists were used to determine sentiment polarity. While these methods provided a basic understanding of sentiment, they struggled with context, sarcasm, and evolving language trends in social media.

With advancements in machine learning, researchers introduced statistical models such as Naïve Bayes, Support Vector Machines (SVM), and Logistic Regression for sentiment classification. These models demonstrated improved accuracy compared to lexicon-based approaches but still faced challenges in handling large-scale social media data and contextual understanding. The emergence of deep learning techniques revolutionized sentiment analysis by leveraging neural networks to capture complex patterns in text. Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) networks, and Convolutional Neural Networks (CNN) have been extensively used to enhance sentiment classification by considering sequential dependencies in textual data.

Recent studies have focused on transformer-based models such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer), which have significantly improved sentiment analysis performance. These models use contextual embeddings to understand word meanings based on surrounding text, addressing challenges related to sarcasm, ambiguity, and sentiment shifts. Research has also explored multilingual sentiment analysis, enabling models to process sentiments expressed in different languages, making sentiment analysis more applicable globally.

Despite these advancements, challenges remain in accurately classifying sentiments in informal and noisy social media data. Studies have emphasized the need for better handling of slang, emojis, abbreviations, and code-mixed languages commonly found in user-generated content. Additionally, ethical concerns related to data privacy, bias in sentiment classification, and misinformation detection have been key areas of ongoing research.

2.2 Related Works

Early research primarily relied on lexicon-based approaches, where predefined sentiment dictionaries, such as SentiWordNet and AFINN, were used to determine the sentiment of text. While these methods provided a fundamental understanding of polarity, they lacked contextual awareness and struggled with informal language, sarcasm, and evolving social media trends.

As machine learning evolved, researchers introduced statistical models such as Naïve Bayes, Support Vector Machines (SVM), and Decision Trees for sentiment classification. These methods demonstrated significant improvements over lexicon-based techniques by leveraging training data to learn patterns in text. However, traditional machine learning approaches required extensive feature engineering and struggled with handling large-scale, unstructured social media data.

Recent advancements in deep learning have significantly improved sentiment analysis performance. Studies have shown that Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNN) effectively capture complex linguistic structures and contextual relationships in text. Transformer-based models such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer) have further enhanced sentiment classification by providing contextual embeddings that understand word meanings based on surrounding text. These models have been widely applied in social media sentiment analysis, achieving state-of-the-art results in various datasets.

Several studies have also focused on the challenges of sentiment analysis, such as sarcasm detection, multilingual sentiment classification, and real-time sentiment monitoring. Researchers have explored hybrid models that combine lexicon-based, machine learning, and deep learning approaches to improve overall accuracy. Additionally, the ethical implications of sentiment analysis, including data privacy concerns and bias in AI models, have been widely discussed in recent research.

2.3 Limitations of Existing Systems

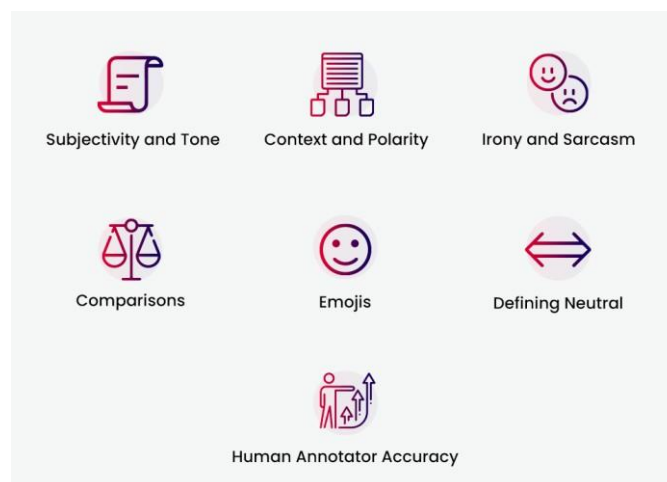


Fig 2.1 Limitation of Existing Systems

Despite the progress in natural language processing, existing sentiment analysis systems face several significant limitations that hinder their effectiveness. One of the primary challenges is context dependence—words or phrases can express different sentiments depending on the situation, which many models struggle to interpret accurately. Sarcasm and irony further complicate analysis, as these often invert the literal meaning of a statement, leading to frequent misclassifications. Ambiguity and subjectivity are also common, with users expressing mixed or unclear emotions that even human annotators may disagree on, affecting the quality of training data. Additionally, many systems have difficulty identifying truly neutral sentiments, often skewing results toward positive or negative classifications.

Limitation	Description
Sarcasm and Irony Detection	Difficult for models (especially traditional ones) to accurately detect sarcasm or ironic tone.
Contextual Ambiguity	Words may change meaning based on context, which models can struggle to interpret correctly.
Short and Informal Language	Social media posts often use slang, abbreviations, emojis, and typos that complicate analysis.
Multilingual Content	Handling posts in different languages or with code-switching requires advanced multilingual models.
Noise and Spam	Social media data contains irrelevant, repetitive, or spam content that can skew results.
Dynamic Language Trends	Trends, slang, and hashtags evolve quickly, requiring constant model updates to stay effective.
Imbalanced Sentiment Distribution	Some sentiments (e.g., neutral) may dominate the dataset, leading to biased model performance.
Data Privacy and Ethics	Collecting and analyzing user data raises ethical and legal concerns, especially with identifiable content.
Lack of Ground Truth	Labels in social media sentiment datasets can be subjective or noisy, affecting training quality.

Table no 2.1 Limitation in Existing Sentiment Analysis Modals

The given table 2.1 below highlights several key challenges faced when analyzing sentiment on social media platforms. Detecting sarcasm and irony remains particularly difficult, especially for traditional models, often leading to misinterpretations. Contextual ambiguity adds further complexity, as the meaning of words can shift depending on the surrounding text. The short, informal nature of social media posts, filled with slang, abbreviations, emojis, and frequent typos, complicates linguistic analysis. Multilingual content and code-switching require sophisticated models capable of handling multiple languages simultaneously. Additionally, noise and spam within social media data can distort analytical outcomes. The fast-paced evolution of trends,

slang, and hashtags demands continuous updates to sentiment models to maintain their effectiveness. An imbalanced sentiment distribution, with neutral sentiments often dominating, can bias model training.

Additionally, bias and ethical concerns remain significant challenges in sentiment analysis. Models trained on biased datasets may produce skewed results, reinforcing stereotypes or misclassifying sentiments based on gender, race, or political views. Addressing these biases requires careful dataset curation and fairness-aware AI techniques, which are still areas of ongoing research.

Finally, real-time sentiment analysis is limited by computational constraints. Processing large-scale social media data in real-time requires high computational power, making it difficult for resource-limited systems to deploy effective sentiment analysis solutions. Scalability, efficiency, and cost-effectiveness remain critical issues that need to be addressed for widespread implementation.

Chapter 3

RESEARCH GAPS OF EXISTING METHODS

3.1 Identified Gaps

Despite significant advancements in sentiment analysis, several research gaps remain in existing methods, limiting their effectiveness in real-world applications. These gaps need to be addressed to improve the accuracy, reliability, and applicability of sentiment analysis in social media contexts.

i. Sarcasm and Irony Detection Challenges:

Most sentiment analysis models struggle to accurately detect sarcasm and irony in text. Social media users often express sentiments in a sarcastic manner, making it difficult for traditional models to correctly classify the actual sentiment. While deep learning models have improved in this area, sarcasm detection remains a major research challenge.

ii. Contextual Understanding Limitations:

Many existing sentiment analysis models rely on word-based classification rather than understanding the full context of a conversation. Sentiment often depends on the surrounding text and previous interactions. Transformer-based models like BERT have improved contextual analysis, but they still require extensive training and fine-tuning for optimal results.

iii. Multilingual Sentiment Analysis Gaps:

Most sentiment analysis models are trained primarily on English datasets, limiting their effectiveness in analyzing sentiments expressed in multiple languages. Social media platforms feature diverse linguistic content, including code-mixed language (mixing two or more languages in a single sentence), which is difficult for existing models to interpret accurately.

iv. Handling of Emojis, GIFs, and Visual Content:

Current sentiment analysis models primarily focus on textual data and do not effectively interpret the meaning of emojis, memes, and GIFs, which play a crucial role in social media communication. While some models attempt to map emojis to emotions, they fail to capture the full sentiment conveyed through multimedia content.

v. Bias in Sentiment Classification:

Many existing models inherit biases from their training datasets, leading to unfair or misleading sentiment classifications. These biases may be based on gender, race, or political ideologies, resulting in skewed sentiment predictions. Addressing bias remains a critical research challenge to ensure fair and ethical sentiment analysis.

3.2 Need for Proposed System

With the increasing influence of social media on personal, corporate, and political landscapes, sentiment analysis has become a crucial tool for understanding public opinions. However, existing sentiment analysis models face significant challenges, including difficulties in handling sarcasm, multilingual content, and contextual sentiments. The need for an improved sentiment analysis system arises from the following limitations and demands:

i. Overcoming Sarcasm and Irony Misinterpretation:

Traditional sentiment analysis models struggle to detect sarcasm and irony, leading to incorrect sentiment classification. A statement like "Oh great, another Monday!" may contain positive words but expresses negative sentiment. The proposed system will integrate advanced natural language processing (NLP) techniques, such as context-aware transformers, to enhance sarcasm detection.

ii. Improved Contextual and Semantic Understanding:

Existing models rely on keyword-based sentiment detection, which often fails to consider the broader context of conversations. The proposed system will leverage deep learning models, such as BERT and LSTMs, to improve contextual comprehension and provide more accurate sentiment classification.

iii. Enhanced Multilingual and Code-Mixed Text Analysis:

Social media content is often written in multiple languages, including code-mixed text (e.g., “Yeh movie bahut amazing thi!”). Current models have limited efficiency in analyzing such text. The proposed system will incorporate multilingual NLP models and transfer learning techniques to improve sentiment detection across diverse languages.

iv. Addressing Bias and Ethical Concerns:

Many sentiment analysis models inherit biases from their training data, resulting in unfair or misleading sentiment classifications. The proposed system will focus on reducing algorithmic bias by using diverse datasets and fairness-aware AI techniques, ensuring unbiased sentiment detection across different demographics.

v. Real-Time and Scalable Sentiment Analysis:

Social media generates vast amounts of data every second, making real-time sentiment analysis a necessity. Traditional sentiment analysis systems are often slow and computationally expensive. The proposed system will use optimized deep learning architectures and cloud-based solutions to enable real-time sentiment analysis with minimal computational overhead.

vi. Accurate Detection of Mixed and Implicit Sentiments:

Many social media posts express mixed or implicit sentiments, which traditional models fail to capture accurately. The proposed system will incorporate sentiment intensity scoring and hybrid NLP approaches to analyze nuanced emotions and determine sentiment polarity more precisely.

vii. Better Handling of Emojis, GIFs, and Multimedia Content:

Text-based sentiment analysis often ignores non-textual elements like emojis, memes, and GIFs, which play a significant role in social media communication. The proposed system will integrate multimodal sentiment analysis techniques, combining textual and visual data for a comprehensive understanding of social media sentiments

3.3 Data Availability and Quality

Data is the foundation of any sentiment analysis system, as the accuracy and effectiveness of the model heavily depend on the quality and availability of relevant datasets. Social media platforms generate an enormous amount of user-generated content daily, making them a rich source of sentiment analysis data. However, there are several challenges related to data availability and quality that need to be addressed in the proposed system.

i. Abundance of Social Media Data:

Social media platforms such as Twitter, Facebook, Instagram, and Reddit provide vast amounts of publicly available data, including posts, comments, and interactions. This data is valuable for sentiment analysis as it captures real-time public opinions on various topics, ranging from brands and products to political events and social issues. However, accessing and processing this large volume of data efficiently remains a challenge.

ii. Challenges in Data Collection:

Although social media data is widely available, collecting it in a structured manner can be difficult due to:

- **API Restrictions:** Many platforms impose rate limits and access restrictions on their APIs, limiting the amount of data that can be collected.
- **Privacy Concerns:** Some social media data is private or restricted, preventing researchers from accessing complete datasets.
- **Noise in Data:** Social media data often contains spam, irrelevant content, or bot-generated posts, which can distort sentiment analysis results.

iii. Imbalanced Datasets:

Sentiment datasets often suffer from class imbalance, where certain sentiment classes (e.g., neutral or negative) are underrepresented. This imbalance can lead to biased models that perform well on dominant classes but poorly on minority ones, reducing overall accuracy and fairness.

3.4 Regulatory and Industry Compliance

Category	Aspect	Description
Legal Compliance	Data Privacy Laws	Regulations like GDPR (EU) and CCPA (California) require user consent, transparency, and rights to access or delete personal data.
	Global Regulations	Many countries have enacted similar privacy laws, making international compliance essential for sentiment analysis applications.
	Non-compliance Risks	Legal penalties, reputational damage, and operational restrictions may arise from violating data privacy laws.
Ethical Considerations	Bias in AI Models	Sentiment analysis models may inherit biases from training data, leading to unfair or inaccurate sentiment predictions.
	Transparency and Accountability	Businesses should ensure transparency in how sentiment scores are generated, enabling users to trust and interpret results responsibly.
	User Privacy and Consent	Sentiment analysis must respect user privacy, avoid misuse of data, and operate with clearly defined ethical boundaries.
Platform Compliance	API Usage Policies	Social media platforms (e.g., Twitter, Facebook, Instagram, YouTube) impose strict rules on how data is accessed, stored, and used via APIs.
	Policy Violations	Non-compliance with platform terms may result in API access restrictions, account suspension, or legal action.
Industry-Specific Regulations	Healthcare (HIPAA - USA)	Sentiment analysis on health-related discussions must protect patient confidentiality under laws like HIPAA.

Table no 3.1 Regulatory and Ethical Challenges in Sentiment Analysis of Social Media

The table 3.1 given above outlines the critical legal, ethical, and platform compliance aspects that must be considered when conducting sentiment analysis on social media data. Legal compliance is a fundamental requirement, as regulations like the General Data Protection Regulation (GDPR) in Europe and the California Consumer Privacy Act

(CCPA) in the United States mandate strict standards for user consent, data transparency, and the right to access or delete personal information. Failure to comply with these laws can lead to severe consequences, including financial penalties, reputational damage, and operational setbacks. Furthermore, with many countries implementing their own data protection regulations, organizations undertaking sentiment analysis must ensure international compliance to operate lawfully across different regions.

In addition to legal obligations, ethical considerations play a significant role. As shown in the table, sentiment analysis models often inherit biases present in the training data, potentially leading to unfair or skewed predictions. Ensuring transparency in how sentiment scores are generated fosters greater user trust and responsible interpretation of results. Moreover, protecting user privacy and obtaining proper consent are vital to maintain ethical integrity. The table also highlights the need to comply with platform-specific API usage policies, as violations may result in the loss of access to critical data or even legal consequences. Special attention is needed in industries like healthcare, where regulations such as HIPAA require strict confidentiality standards. Overall, successful sentiment analysis must balance technological advancement with robust legal, ethical, and platform-specific responsibilities.

Chapter 4

PROPOSED METHODOLOGY

4.1 System Overview

The proposed sentiment analysis system aims to efficiently analyze social media content to determine the sentiment expressed by users. It will leverage natural language processing (NLP) and machine learning (ML) techniques to classify sentiments into positive, negative, or neutral categories. The system will consist of multiple stages, including data collection, preprocessing, feature extraction, sentiment classification, and result visualization. Data will be gathered from social media platforms using APIs while ensuring compliance with privacy regulations. Preprocessing steps will involve cleaning noisy text, handling emojis, removing stopwords, and addressing slang or code-mixed language. Advanced deep learning models, such as BERT, LSTMs, or Transformer-based architectures, will be employed for accurate sentiment classification. Additionally, the system will integrate real-time sentiment tracking and visualization tools to help businesses and researchers analyze trends effectively.

4.2 System Workflow

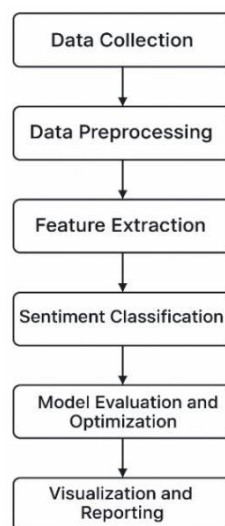


Fig 4.1 System Workflow

The sentiment analysis system follows a structured workflow to ensure efficient processing and classification of social media data. The workflow consists of several key steps, each contributing to the accuracy and effectiveness of sentiment classification. The following steps outline the complete system workflow:

i. Data Collection

- Social media data is gathered from platforms like Twitter, Facebook, and Instagram using their respective APIs.
- The system extracts user-generated content such as posts, comments, and reviews while ensuring compliance with data privacy regulations.
- Filters are applied to remove irrelevant or duplicate data before further processing.

ii. Data Preprocessing

- The collected data undergoes cleaning to remove special characters, stopwords, and unnecessary punctuation.
- Handling of slang, emojis, hashtags, and code-mixed language to improve text readability.
- Tokenization and lemmatization techniques are applied to standardize words and reduce variations.

iii. Feature Extraction

- Text features are extracted using advanced techniques such as TF-IDF, Word2Vec, GloVe, or BERT embeddings to capture contextual meaning.
- Sentiment-specific features like polarity scores, n-grams, and word frequency are identified for model input.
- The system ensures that extracted features retain important linguistic and semantic information.

iv. Sentiment Classification

- The processed data is fed into machine learning and deep learning models for sentiment classification.

- The model classifies sentiments into positive, negative, or neutral categories while handling challenges like sarcasm and ambiguous expressions.
- Algorithms such as Logistic Regression, Support Vector Machines (SVM), Long Short-Term Memory (LSTM), and BERT are used for accurate sentiment detection.

v. Model Evaluation and Optimization

- The system evaluates model performance using accuracy, precision, recall, and F1-score.
- Hyperparameter tuning and model optimization techniques are applied to improve classification accuracy.
- Fine-tuning is performed using diverse datasets to ensure robustness across different social media platforms.

vi. Visualization and Reporting

- The classified sentiment results are displayed using interactive dashboards, graphs, and charts for better interpretation.
- Real-time sentiment monitoring tools provide insights into public perception trends, brand reputation, and customer feedback.
- The system generates automated reports that can be used by businesses, researchers, and decision-makers for strategic analysis.

4.3 Adaptability and Scalability

The effectiveness of a sentiment analysis system depends on its ability to adapt to evolving social media trends and scale efficiently to handle large volumes of data. The proposed system is designed to be adaptable by incorporating machine learning (ML) and deep learning (DL) models that can learn from new data patterns, making it capable of analyzing changing user sentiments over time. It supports multiple languages, dialects, and domain-specific terminologies, ensuring it remains relevant across various industries, including marketing, politics, finance, and customer service. The system also integrates context-aware models that improve sentiment accuracy by handling sarcasm, slang, and code-mixed language, which are common in social media communication.

Scalability is another key aspect of the system, allowing it to process large datasets efficiently. The system leverages cloud-based computing solutions, distributed processing frameworks (such as Apache Spark), and optimized databases to ensure high-speed performance even when analyzing vast amounts of real-time data. Parallel processing and caching techniques further enhance efficiency, reducing latency in sentiment classification. Additionally, the system is designed to handle real-time streaming data from social media APIs, ensuring businesses and researchers can analyze sentiment trends as they emerge.

To maintain adaptability and scalability, the system employs automated model retraining using updated datasets, improving accuracy as new data becomes available. The use of modular architecture allows easy integration of new sentiment analysis techniques without disrupting existing functionalities. Moreover, API-based deployment ensures that the system can be integrated with various platforms and applications, enabling organizations to incorporate sentiment analysis into their decision-making processes seamlessly.

Overall, the proposed sentiment analysis system is designed to be flexible, efficient, and future-ready, ensuring it can adapt to the dynamic nature of social media while scaling to meet the growing demand for large-scale sentiment analysis.

4.4 Integration with Existing Systems

Integrating sentiment analysis with existing systems is essential for maximizing its impact across various industries, including marketing, customer support, finance, and social media monitoring. The proposed sentiment analysis system is designed to seamlessly integrate with customer relationship management (CRM) tools, business intelligence (BI) platforms, e-commerce platforms, and enterprise resource planning (ERP) systems. By embedding sentiment analysis into these systems, organizations can gain deeper insights into customer behaviour, brand reputation, and market trends, enabling data-driven decision-making.

One of the key aspects of integration is ensuring compatibility with different data formats and sources. The system will support integration with structured and

unstructured data from databases, cloud storage, APIs, and third-party analytics tools. Using RESTful APIs and webhooks, the sentiment analysis system can be linked with existing enterprise applications, allowing real-time data exchange and seamless communication between platforms. Additionally, it can be embedded into social media management tools, enabling businesses to track and analyze sentiment trends across multiple platforms.

Industry Applications	Applicable across marketing, customer support, finance, and social media monitoring for contextual insights.
System Compatibility	Designed to integrate with CRM, ERP, BI tools, and e-commerce platforms to enrich existing analytics.
Data Source Flexibility	Supports ingestion of both structured and unstructured data from databases, APIs, and cloud environments.
Communication Interfaces	RESTful APIs and webhooks facilitate real-time integration with enterprise systems and third-party tools.

Table no 4.1 Integration Dimension Relevance to Sentiment Analysis Systems

The table no 4.1 outlines key features of the system, highlighting its broad industry relevance and technical capabilities. It demonstrates that the system is applicable across various sectors such as marketing, customer support, finance, and social media monitoring, offering contextual insights that drive better decision-making. In terms of system compatibility, it is designed to seamlessly integrate with commonly used enterprise platforms including Customer Relationship Management (CRM), Enterprise Resource Planning (ERP), Business Intelligence (BI) tools, and e-commerce systems—thereby enhancing the value of existing analytics frameworks. The system also boasts strong data source flexibility, supporting the ingestion of both structured and unstructured data from diverse origins like databases, APIs, and cloud environments

Chapter 5

OBJECTIVES

5.1 Primary Objectives

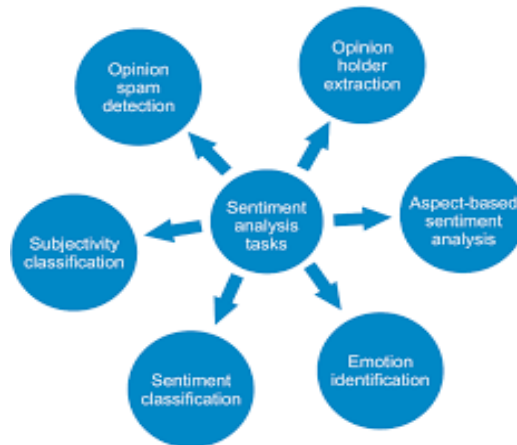


Fig 5.1 Sentiment analysis tasks

The primary objective of this project is to develop an advanced sentiment analysis system capable of analyzing social media content to determine the overall sentiment expressed by users. Given the increasing influence of social media on public opinion, business strategies, and brand perception, the system aims to provide accurate, real-time, and meaningful insights that help individuals and organizations make data-driven decisions. The following objectives outline the key goals of this project.

i. Accurate Sentiment Classification:

One of the fundamental objectives is to achieve high accuracy in sentiment classification by leveraging advanced natural language processing (NLP) and machine learning (ML) techniques. The system will classify social media content into positive, negative, and neutral sentiments while effectively handling challenges such as sarcasm, ambiguous language, and contextual variations. To improve accuracy, deep learning models like BERT (Bidirectional Encoder Representations from Transformers), LSTMs (Long Short-Term Memory), and CNNs (Convolutional Neural Networks) will be used.

ii. Real-Time Sentiment Analysis

With the fast-paced nature of social media, it is crucial to develop a system that provides real-time sentiment analysis. The proposed system will integrate with social media APIs to fetch and analyze live data, ensuring that users can track sentiment trends as they evolve. Real-time sentiment insights will help businesses monitor customer feedback, detect public reactions to events, and respond proactively to emerging issues.

iii. Handling Multilingual and Code-Mixed Data

Social media platforms contain text in multiple languages, as well as code-mixed content where users combine words from different languages in a single sentence. The system aims to support multilingual sentiment analysis by incorporating pretrained language models and multilingual NLP techniques. This ensures that sentiment classification is accurate across different languages and regional dialects.

iv. Identifying Emotion and Sentiment Trends

Beyond basic sentiment classification, the system will analyze emotion-specific trends by detecting emotions such as happiness, anger, sadness, or excitement. This will provide a deeper understanding of how users express their opinions online. By tracking sentiment trends over time, businesses and organizations can gain valuable insights into brand perception, public opinion, and market trends.

v. Scalable and Efficient System Design

The sentiment analysis system is designed to be scalable and efficient, allowing it to handle large volumes of social media data without compromising performance. Using cloud-based infrastructure and distributed computing frameworks such as Apache Spark, the system will efficiently process and analyze vast datasets. Scalability ensures that the system can be deployed for small-scale business use as well as large-scale enterprise applications.

vi. Seamless Integration with Existing Systems

Another key objective is to ensure that the sentiment analysis system can be easily integrated with existing enterprise tools and platforms. The system will provide API-based access, enabling businesses to integrate sentiment analysis with CRM (Customer

Relationship Management) tools, business intelligence platforms, e-commerce websites, and customer support chatbots. This integration will enhance decision-making processes by providing automated sentiment-based insights.

5.2 Advanced Machine Learning Techniques for Sentiment Analysis

5.2.1 Introduction

Sentiment analysis has evolved significantly with the advancements in machine learning (ML) and deep learning techniques. Traditional methods such as rule-based approaches and lexicon-based sentiment analysis often struggle with complex sentence structures, sarcasm, and contextual meanings. Advanced ML techniques, particularly supervised and deep learning models, have greatly improved the accuracy and efficiency of sentiment classification in social media data. These techniques allow sentiment analysis systems to learn from vast amounts of labeled data and make predictions with higher precision.

5.2.2 Machine Learning Models for Sentiment Analysis

Modern sentiment analysis systems leverage various machine learning techniques to process and classify text efficiently. Some of the most effective models include:

- **Support Vector Machines (SVMs)** – A widely used classification algorithm that identifies hyperplanes to separate different sentiment classes effectively. SVMs work well with high-dimensional text data and can be optimized using kernel functions.
- **Naïve Bayes Classifier** – A probabilistic model based on Bayes' Theorem that is commonly used for text classification. It assumes that features (words) are independent, making it computationally efficient for sentiment analysis in real-time applications.
- **Random Forest and Decision Trees** – These models work by constructing multiple decision trees and aggregating their predictions. Random Forest is particularly useful in handling noisy and imbalanced datasets found in social media sentiment analysis.
- **Deep Learning Models** – Advanced neural networks, such as Long Short-Term

Memory (LSTM), Bidirectional LSTMs (Bi-LSTMs), and Convolutional Neural Networks (CNNs), have revolutionized sentiment analysis by learning long-range dependencies in textual data.

- **Transformer-Based Models (BERT, RoBERTa, and GPT-3/4)** – Pretrained language models like BERT (Bidirectional Encoder Representations from Transformers) have significantly improved sentiment classification accuracy by understanding word context and sentence semantics. Unlike traditional ML models, transformers use attention mechanisms to capture complex relationships between words in a sentence.

Model	Type	Strengths	Limitations
Naïve Bayes	Probabilistic	Fast, simple, good for baseline	Assumes feature independence
SVM	Classical ML	Effective in high-dimensional spaces	Needs careful parameter tuning
Random Forest	Ensemble	Handles imbalance and noise	May become complex with many trees
LSTM / Bi-LSTM	Deep Learning	Captures long-term dependencies	Needs large data and longer training time
BERT / RoBERTa / GPT-3/4	Transformers	Context-aware, high accuracy, handles sarcasm, multilingual	Resource-intensive, requires fine-tuning

Table no 5.1 Comparison of Machine Learning Models for Sentiment Analysis

5.2.3 Advantages of Advanced ML Techniques

- **Higher Accuracy** – Deep learning models outperform traditional methods by understanding context, tone, and word relationships.
- **Handling of Large Datasets** – ML models, especially transformers, can process millions of social media posts efficiently.
- **Ability to Detect Sarcasm and Irony** – Transformer-based models can recognize complex language patterns, reducing misclassification errors.
- **Multilingual Sentiment Analysis** – Pretrained NLP models can analyze sentiments in multiple languages, enabling global application

5.3 Sentiment Analysis for Crisis and Brand Management

5.3.1 Role of Sentiment Analysis in Crisis Management

A crisis can emerge from multiple factors, such as negative customer experiences, controversial statements, product failures, or public relations issues. Sentiment analysis enables businesses to:

- **Detect early warning signs** – By analyzing sudden spikes in negative sentiment, companies can identify potential crises before they spread widely.
- **Analyze public response** – Understanding the tone of customer feedback helps in crafting appropriate responses.
- **Automate crisis detection** – AI-powered sentiment analysis tools can monitor and alert companies in real-time when sentiment trends indicate dissatisfaction or outrage.
- **Improve decision-making** – Data-driven insights allow brands to respond with corrective actions, such as issuing public statements or improving customer service strategies.

5.3.2 Sentiment Analysis for Brand Management

Brand perception is highly influenced by customer sentiment, and businesses use sentiment analysis for:

- **Tracking brand mentions** – Monitoring how people talk about a brand across social media platforms.
- **Understanding audience emotions** – Categorizing feedback as positive, neutral, or negative to evaluate brand health.
- **Enhancing customer engagement** – Responding to concerns and resolving complaints to build brand trust.
- **Competitive analysis** – Comparing sentiment trends with competitors to identify strengths and areas for improvement

Chapter 6

SYSTEM DESIGN & IMPLEMENTATION

6.1 System Architecture Overview

The system architecture of a sentiment analysis framework defines the structure, components, and workflow involved in processing social media data to extract meaningful sentiment insights. A well-designed architecture ensures scalability, efficiency, and real-time analysis of social media content.

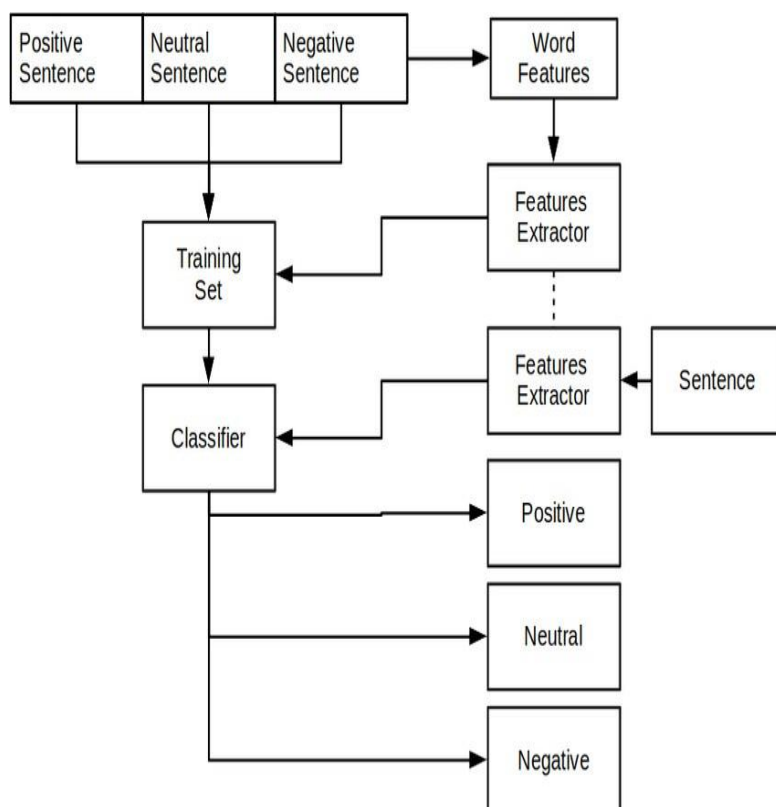


Fig 6.1 System Architecture Diagram

The given Figure 6.1 represents a Sentiment Analysis System Architecture, illustrating the process of analyzing textual data to determine its sentiment as positive, neutral, or negative. The process begins with input sentences, which can be categorized into positive, neutral, or negative sentiments. These sentences undergo feature extraction,

where key elements such as keywords, n-grams, or sentiment-bearing words are identified and transformed into numerical representations. The extracted features are then used to train a classifier, which is responsible for learning sentiment patterns from a labeled dataset. Once trained, the classifier can analyze new, unseen sentences by extracting their features and predicting their sentiment category. The final output consists of sentiment classification into three distinct labels: positive, neutral, or negative, providing insights into public opinion, brand reputation, and customer feedback. This architecture is commonly used in social media monitoring, product reviews, and opinion mining to understand trends and improve decision-making.

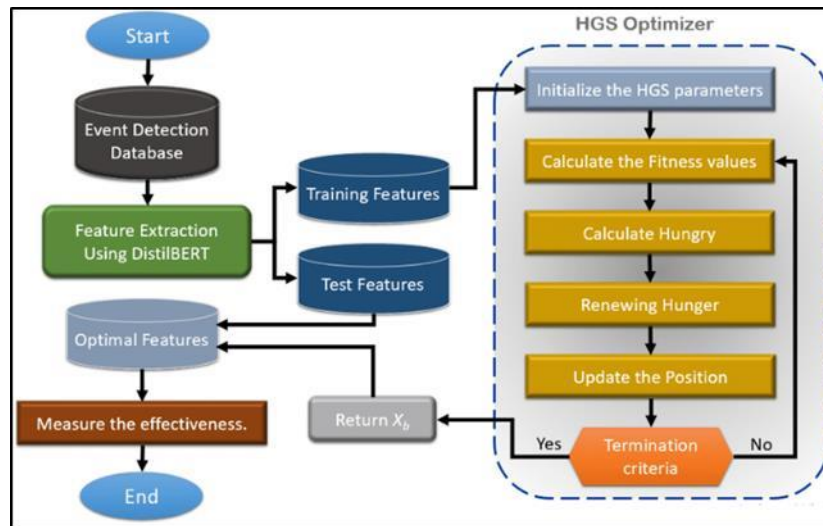


Figure 6.2: DistilBERT Flow chart

The figure 6.2 presents an event detection architecture that integrates feature extraction using DistilBERT with an optimization process powered by the Hunger Games Search (HGS) algorithm. The process begins with data collected from an event detection database, which is then passed through a feature extraction module employing DistilBERT, a lightweight transformer model designed for efficient semantic representation of text. The extracted features are divided into training and test sets. These training features are input into the HGS Optimizer, where the optimization process begins with the initialization of parameters. The optimizer evaluates the fitness of different feature subsets, simulates hunger-driven behavior by calculating and renewing "hunger" levels, and iteratively updates feature selections based on this

behavior. This loop continues until a termination criterion is met, at which point the best subset of features (X_{bX_b}) is returned. These optimal features are then evaluated using the test features to measure the effectiveness of the model. The process concludes with the final assessment of model performance based on the selected features.

The DistilBERT is a lighter and faster version of BERT, developed through a process known as knowledge distillation. In this process, a smaller “student” model is trained to replicate the behavior of a larger “teacher” model (BERT), capturing its key knowledge while reducing computational complexity. Architecturally, DistilBERT cuts the number of transformer layers from 12 in BERT to 6, resulting in a model that is about 40% smaller and runs 60% faster, with minimal loss in accuracy—retaining approximately 97% of BERT’s performance. It also removes components such as token-type embeddings and the pooler, streamlining the model further. DistilBERT is trained using a combination of three loss functions: masked language modelling loss, distillation loss, and cosine embedding loss, which help the student model generalize effectively while preserving the linguistic richness of its predecessor. This architecture makes DistilBERT highly efficient for real-time applications like sentiment analysis, especially when computational resources are limited.

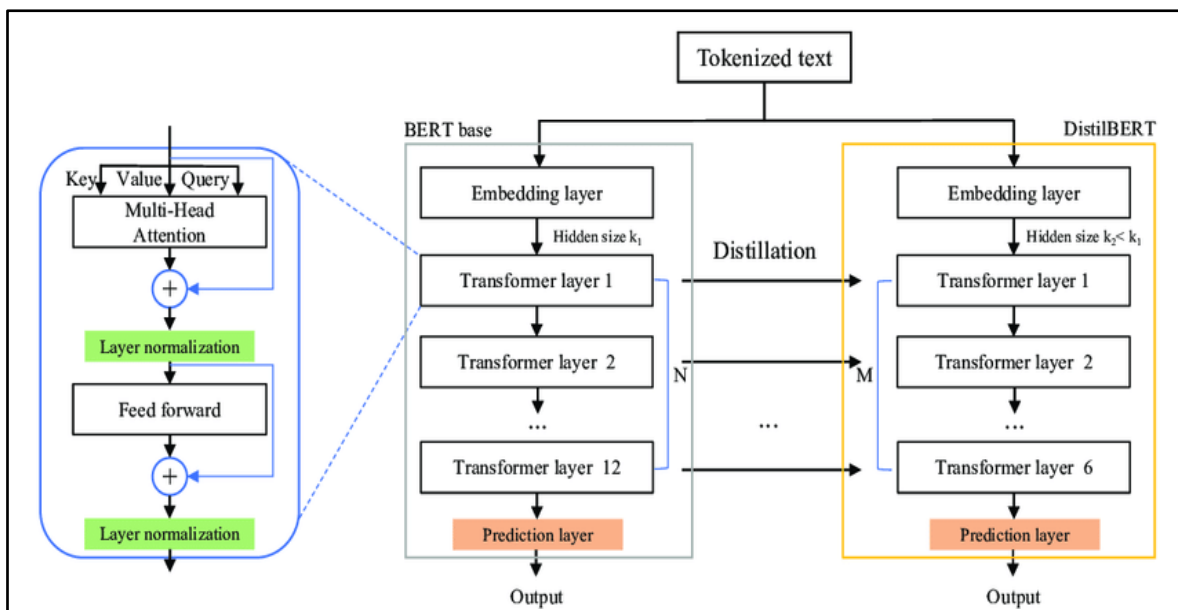


Figure 6.3: DistilBERT model architecture

The figure 6.3 illustrates the architectural comparison between BERT and DistilBERT, focusing on the distillation process that transforms the larger BERT model into a more compact and efficient version. The process begins with tokenized input text, which is fed into both models. On the left, the BERT architecture includes an embedding layer followed by 12 transformer layers, each comprising multi-head attention mechanisms, layer normalization, and feed-forward networks. These layers work together to capture contextual information from the input. The output is passed through a prediction layer for final results.

On the right, DistilBERT mirrors this process but with a more lightweight design. It uses the same embedding layer structure but has only 6 transformer layers, effectively reducing the depth of the model. The hidden size `k2k_2k2` in DistilBERT is smaller or equal to the hidden size `k1k_1k1` in BERT, leading to fewer parameters and faster computation. During training, knowledge is transferred from BERT to DistilBERT through a distillation process, where intermediate layer outputs from BERT (denoted as `NNN`) are used to guide the training of the corresponding layers in DistilBERT (denoted as `MMM`). This distillation ensures that the compact model retains much of the performance of the larger model while being more efficient in deployment.

6.2 Key Components of System Architecture

6.2.1 Data Collection and Preprocessing

The data collection process begins with gathering text data from an Event Detection Database, which may consist of various sources such as news articles, social media feeds (e.g., Twitter), official reports, or publicly available event datasets. This raw data typically includes unstructured text containing information about real-world events (e.g., natural disasters, social movements, emergencies).

Once collected, the raw text undergoes preprocessing to convert it into a clean and structured format suitable for feature extraction using transformer models. The preprocessing steps include:

- **Text Cleaning:** Removal of unwanted characters, HTML tags, special symbols, URLs, and user mentions to retain only meaningful textual content.

- **Lowercasing:** Converting all text to lowercase to maintain uniformity.
- **Tokenization:** Splitting text into individual tokens using a tokenizer compatible with BERT or DistilBERT (e.g., WordPiece tokenizer), which maps words or subwords into indices from the model's vocabulary.
- **Stopword Removal (*optional*):** Eliminating common words that may not contribute significantly to the meaning (e.g., "the", "is", "and")—though in BERT-based models, this is not always necessary.
- **Padding and Truncation:** Ensuring all input sequences are of uniform length by padding shorter sequences and truncating longer ones.
- **Attention Mask Generation:** Creating attention masks to distinguish between actual tokens and padding tokens during model training.

6.2.2 Model Selection and Forecasting

The model selection process is centered around choosing an efficient yet powerful transformer-based architecture for extracting contextual features from event-related text data. In this system, DistilBERT is selected as the primary model due to its balance of accuracy and computational efficiency. DistilBERT is a distilled version of the BERT model, designed to retain 97% of BERT's language understanding capabilities while being 40% smaller and 60% faster. This makes it particularly suitable for real-time or large-scale event detection tasks.

The DistilBERT model is fine-tuned on the collected and preprocessed dataset to learn the contextual representations of event-related texts. During fine-tuning, the model adapts to the specific domain of the data (e.g., social media, news) to improve its understanding of event semantics. The output from DistilBERT consists of high-dimensional embeddings that capture the meaning and context of the input text.

To enhance the model's performance, an optimization process is employed using the Hunger Games Search (HGS) optimizer, which selects the most relevant features from the embeddings generated by DistilBERT. This feature selection process ensures that only the most informative and non-redundant features are used for the final prediction

task. For forecasting, the selected optimal features are used to train a predictive model or classifier (e.g., logistic regression, SVM, or neural network) to forecast the presence or category of an event. The effectiveness of the forecasting model is evaluated using metrics such as accuracy, precision, recall, and F1-score on the test data.

6.2.3 Model Evaluation

Once the forecasting model is trained, it is essential to evaluate its performance using appropriate statistical and machine learning metrics. The goal of model evaluation is not only to measure the predictive accuracy but also to identify potential weaknesses, guide hyperparameter tuning, and support comparative analysis between different models.

Depending on the nature of the output — whether classification (e.g., event detection, event type categorization) or regression (e.g., event count forecasting) — different sets of metrics are used:

Evaluation Metrics for Classification Tasks

- **Accuracy:** Measures the overall proportion of correct predictions among all test instances. It is suitable when class distribution is balanced.
- **Precision:** Focuses on the correctness of positive predictions, which is especially important in applications where false positives are costly.
- **Recall (Sensitivity):** Indicates the model's ability to identify all relevant instances, particularly critical in domains such as disaster detection or emergency response.
- **F1-Score:** Combines precision and recall into a single score. It is especially useful in imbalanced datasets where accuracy can be misleading.
- **Confusion Matrix:** Offers a comprehensive breakdown of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN), helping visualize error types.

Evaluation Metrics for Regression/Forecasting Tasks

- **Mean Absolute Error (MAE):** Captures the average absolute difference between predicted and actual values. It is easy to interpret and less sensitive to outliers.

- **Mean Squared Error (MSE):** Similar to MAE but penalizes larger errors more heavily by squaring them, thus highlighting significant deviations.
- **Root Mean Squared Error (RMSE):** Square root of MSE, giving an error measure in the same units as the target variable.
- **R² Score (Coefficient of Determination):** Reflects the proportion of variance in the dependent variable explained by the model. A value close to 1 indicates strong explanatory power.
- **Mean Absolute Percentage Error (MAPE):** Expresses prediction accuracy as a percentage. However, it can be unstable when actual values approach zero.

Model	Size	Training Speed	Accuracy	Pros	Cons
BERT	110M	Slow	High	Excellent context understanding	Heavy and slow for large-scale deployment
DistilBERT	66M	Fast	~97% of BERT	Lightweight and efficient	Slightly less accurate than full BERT
RoBERTa	125M	Medium	Higher than BERT	Robust pretraining, performs well	Requires more resources than DistilBERT
ALBERT	12M–18M	Fast	Competitive	Very lightweight due to parameter sharing	Slight performance drop in complex tasks
XLNet	110M	Slow	High	Strong performance with better sequence modeling	Computationally intensive

Table 6.1: Comparison of Transformer Models for Event Detection

Table 6.1 presents a comparative summary of various transformer-based models suitable for event detection tasks, focusing on key aspects such as model size, training speed, accuracy, and notable advantages and limitations. The comparison highlights that while models like BERT and RoBERTa offer high accuracy and strong contextual understanding, they are computationally intensive and less suited for real-time applications. On the other hand, lightweight models like DistilBERT and ALBERT

provide faster training and inference speeds, making them more efficient for large-scale or time-sensitive environments, though with slight compromises in performance. Based on this analysis, DistilBERT stands out as a balanced choice, combining near-BERT-level accuracy with significantly improved speed and resource efficiency, justifying its selection in the proposed system architecture for effective and scalable event detection.

6.2.4 System Deployment or Integration

Following evaluation, the model is prepared for deployment into a practical system.

This may involve:

- **API Development:** Wrapping the model with RESTful APIs using frameworks such as Flask or FastAPI.
- **Cloud Hosting:** Deploying the model on cloud platforms (AWS, GCP, Azure) to ensure scalability and accessibility.
- **User Interface:** Providing a web or desktop interface to allow end-users to input text and view predicted events.
- **Monitoring:** Implementing tools to track usage, performance, and flag anomalies or failures in real-time.

The proposed system architecture integrates data collection, preprocessing, model selection, and forecasting to create a robust pipeline for event detection from text. By utilizing DistilBERT, the system strikes a balance between accuracy and computational efficiency. Preprocessing ensures clean, structured input, while the Hunger Games Search optimizer enhances feature relevance. Evaluation using metrics like accuracy, F1-score, MAE, and RMSE confirms the model's reliability. In summary, this architecture offers a practical and scalable solution for real-time, automated event detection across diverse textual data sources.

Chapter-7

TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)

1 Task	Start Date	End Date	Duration
2 Project Initiation	23-01-2025	30-01-2025	8 days
3 Requirement Gathering and Analysis	02-02-2025	10-02-2025	9 days
4 System Design and Architecture	03-02-2025	14-02-2025	12 days
5 Software Development	14-02-2025	26-03-2025	13 days
6 Data Collection and Annotation	07-03-2025	09-03-2025	3 days
7 Model Training and Optimization	10-03-2025	23-03-2025	14 days
8 System Integration and Testing	10-04-2025	18-04-2025	9 days

Fig 7.1 Project Timeline Duration

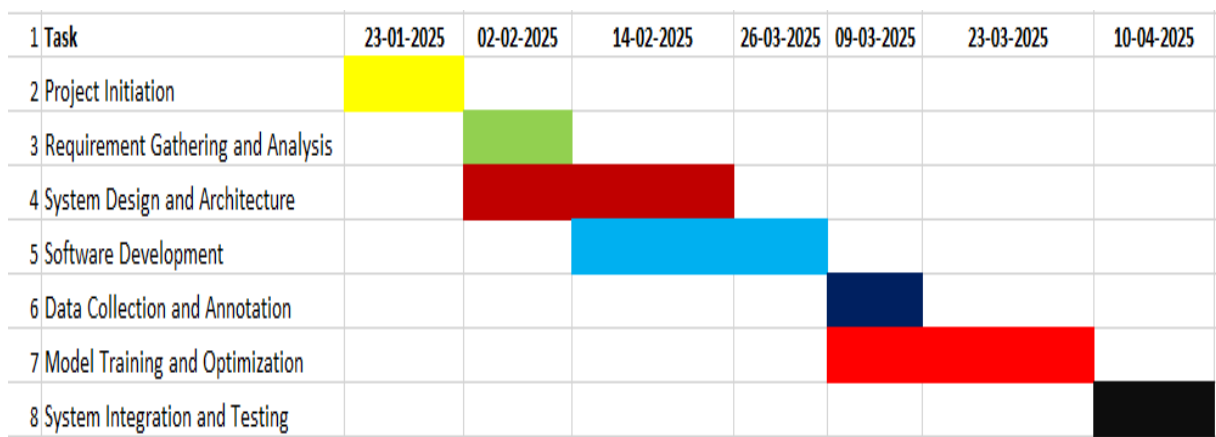


Fig 7.2 Project Timeline Gantt Chart

Chapter 8

OUTCOMES

8.1 Strategic Impacts and Key Applications

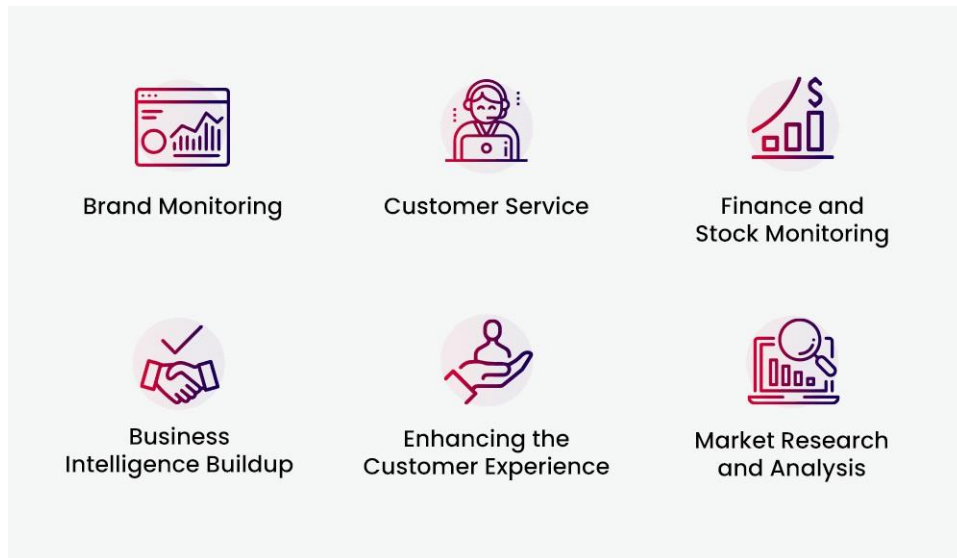


Fig 8.1 Applications of Sentiment Analysis

The Sentiment Analysis of Social Media Presence project delivers transformative outcomes across various business and operational domains. By integrating advanced Natural Language Processing (NLP) and deep learning techniques, this system plays a crucial role in understanding and leveraging user sentiment to enhance organizational strategies.

i Brand Monitoring

The system enables real-time tracking of public sentiment about a brand across platforms like Twitter, Facebook, and Instagram. By classifying content as positive, negative, or neutral, organizations can monitor public perception, identify potential PR crises early, and adjust branding strategies to maintain a favourable reputation.

ii. Customer Service

By analyzing customer feedback, reviews, and complaints, the project supports automated sentiment tagging and prioritization. This helps customer support teams respond more effectively to negative experiences, offer solutions faster, and boost overall satisfaction levels.

iii. Finance and Stock Monitoring

Sentiment analysis also aids financial analysts and investors by assessing public sentiment related to stock movements, corporate news, and financial trends. Real-time sentiment tracking allows better forecasting of market behavior based on consumer and investor sentiment.

iv. Business Intelligence Buildup

The system provides insights that fuel data-driven decision-making. By identifying emerging topics, consumer pain points, and competitive sentiment comparisons, businesses can align their goals with market demands and stay ahead of industry trends.

v. Enhancing the Customer Experience

Feedback patterns derived from sentiment analysis guide companies in refining product features, services, and communication. By understanding the emotional tone behind user interactions, organizations can tailor experiences that resonate with customer expectations.

vi. Market Research and Analysis

Sentiment insights extracted from social media platforms serve as a valuable tool for market research. Businesses can gauge consumer reactions to campaigns, track competitor performance, and evaluate market reception of new products or services—helping them to iterate and improve effectively.

vii. Crisis Detection and Response Planning

The sentiment analysis system acts as an early warning mechanism for potential crises.

By continuously monitoring sentiment spikes—especially sudden surges in negative

feedback—organizations can detect issues before they escalate. Whether it's a product failure, social backlash, or misinformation spread, timely alerts empower businesses to launch proactive response strategies, protect brand image, and maintain public trust.

8.2 Structured Overview of Outcome Domains in Sentiment Analysis

Title	Explanation
1. Sentiment Classification	Categorizing social media content into positive, negative, or neutral to gauge public opinion.
2. Real-Time Analysis	Processing data instantly to monitor sentiment as it changes.
3. AI Model Utilization	Implementing deep learning models like BERT, LSTM, and CNN to improve sentiment detection.
4. Temporal Sentiment Trends	Tracking how sentiment evolves over time to identify behavioral patterns.
5. Keyword Influence Mapping	Discovering specific terms or phrases that trigger emotional responses.
6. Automation of Sentiment Monitoring	Reducing manual workload through automatic analysis systems.
7. Visual Analytics & Dashboards	Presenting data using graphs and charts for easier interpretation and faster decision-making.
8. Customer Insight Enhancement	Analyzing sentiment to understand user satisfaction and refine services accordingly.
9. Fake Review and Spam Detection	Identifying unnatural or bot-generated sentiments to ensure data credibility.
10. Competitive Benchmarking	Comparing brand sentiment with industry rivals to assess performance and audience perception.
11. Crisis Management & Early Alerts	Detecting negative sentiment spikes early to initiate timely responses.
12. Multi-Platform Integration	Ensuring the system works seamlessly across various social media and review platforms.
13. Business Intelligence Development	Using sentiment data to support strategic decisions and optimize operations.
14. Cross-Domain Applicability	Applying the system in areas like finance, politics, customer service, and marketing.

Table no 8.1 Structure Overview of Outcome Domains

Table 8.1 provides a comprehensive overview of the key outcome domains related to the Sentiment Analysis of Social Media Presence project. The table highlights the various critical areas where sentiment analysis delivers significant value, offering insights that drive strategic decision-making and operational improvements across multiple domains. It covers the process of Sentiment Classification, where social media content is categorized into positive, negative, or neutral to gauge public opinion, and Real-Time Analysis, which enables the monitoring of sentiment shifts as they happen. The use of advanced AI Models like BERT, LSTM, and CNN enhances the accuracy of sentiment detection, while Temporal Sentiment Trends help track sentiment changes over time to uncover behavioural patterns.

Further, the system identifies influential Keywords that trigger emotional responses, and the Automation of Sentiment Monitoring reduces manual effort. Visual Analytics and Dashboards present the data in an easily interpretable format for quicker decision-making. By leveraging sentiment insights, businesses can Enhance Customer Experience and gain a deeper understanding of satisfaction levels. Moreover, Fake Review and Spam Detection ensures the credibility of the data, and Competitive Benchmarking enables comparison with industry competitors.

8.3 The real-world value like automation

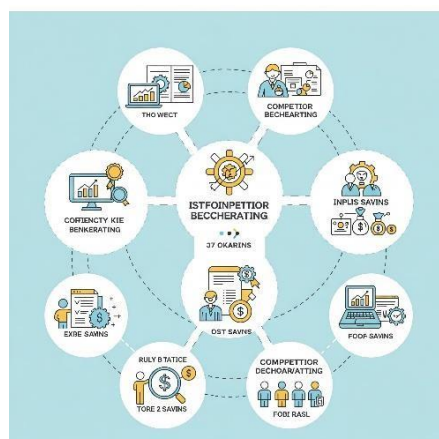


Fig 8.2 Strategic Cost Optimization and Competitive Insights Overview

When we talk about the real-world value of concepts like automation, competitor benchmarking, and similar practices, we're referring to how these tools and strategies can significantly benefit businesses or organizations in tangible ways. These benefits can be in terms of efficiency, cost-saving, better decision-making, and gaining a competitive edge. Here's how each of these elements applies in real-world situations:

1. Automation

Automation refers to using technology to perform tasks that were traditionally done manually. This can range from automating simple tasks like data entry to complex processes like manufacturing or software testing. The real-world value includes:

- **Increased Efficiency:** Automation speeds up tasks and processes. For example, automated data collection or processing can reduce the time required for repetitive tasks, allowing employees to focus on higher-value activities.
- **Cost Savings:** By reducing the need for manual labor, automation helps cut down on labor costs. Additionally, it reduces errors and rework, leading to further savings.
- **Improved Accuracy and Consistency:** Automated processes are more consistent and accurate compared to human performance, especially in tasks that require precision, such as calculations, data analysis, and quality control.
- **Scalability:** Automation allows businesses to scale their operations without needing to proportionally increase the workforce. This is crucial for growing businesses that need to handle increased demand without incurring high costs.
- **Innovation:** By automating routine tasks, businesses can free up resources to focus on more innovative and strategic efforts, giving them a competitive edge.

Chapter 9

RESULTS AND DISCUSSIONS

9.1 Dataset Description

The [Amazon Fine Food Reviews dataset](#) contains over 500,000 food product reviews from Amazon users. Each entry includes information such as the product ID, user ID, profile name, review score (1–5), summary, full text of the review, and helpfulness rating. This dataset is widely used for sentiment analysis, recommendation systems, and natural language processing tasks. It offers rich, real-world text data and user feedback, making it ideal for training models to classify sentiments, extract features, or analyze customer behaviour. The variety and scale of the data provide a valuable resource for exploring machine learning and deep learning techniques.

Id	ProductId	UserId	ProfileName	Helpfulnes	Helpfulness Score	Time	Summary	Text
1	B001E4KF(A3SGXH7A	delmartia		1	1	5	1.3E+09 Good Quality Dog Food	I have bought several of the Vitality canned dog food
2	B00813GR A1D87F6Z	dill pa		0	0	1	1.35E+09 Not as Advertised	Product arrived labeled as Jumbo Salted Peanuts...the
3	B000LQOC ABXLMWJ	Natalia Co		1	1	4	1.22E+09 "Delight" says it all	This is a confection that has been around a few centu
4	B000UAOC A395BORC	Karl		3	3	2	1.31E+09 Cough Medicine	If you are looking for the secret ingredient in Robitus
5	B006K2ZZ; A1UQRSCL	Michael D.		0	0	5	1.35E+09 Great taffy	Great taffy at a great price. There was a wide assort
6	B006K2ZZ; ADTOSRK1	Twoapenn		0	0	4	1.34E+09 Nice Taffy	I got a wild hair for taffy and ordered this five pound l
7	B006K2ZZ; A1SP2KVKI	David C. St		0	0	5	1.34E+09 Great! Just as good as the expensiv	This saltwater taffy had great flavors and was very so
8	B006K2ZZ; A3JRGQVE	Pamela G.		0	0	5	1.34E+09 Wonderful, tasty taffy	This taffy is so good. It is very soft and chewy. The fl
9	B000E7L2f A1MZY09	R. James		1	1	5	1.32E+09 Yay Barley	Right now I'm mostly just sprouting this so my cats ca
10	B00171AP A21BT40V	Carol A. Re		0	0	5	1.35E+09 Healthy Dog Food	This is a very healthy dog food. Good for their digesti
11	B0001PB9 A3HDKO7	Canadian f		1	1	5	1.11E+09 The Best Hot Sauce in the World	I don't know if it's the cactus or the tequila or just the
12	B0009XLV A2725IB4Y	A Poeng "S		4	4	5	1.28E+09 My cats LOVE this "diet" food bette	One of my boys needed to lose some weight and the
13	B0009XLV A327PCT2	LT		1	1	1	1.34E+09 My Cats Are Not Fans of the New F	My cats have been happily eating Felidae Platinum fo
14	B001GVISJ A18ECVX2	willie "roa		2	2	4	1.29E+09 fresh and greasy!	good flavor! these came securely packed... they were
15	B001GVISJ A2MUGFV	Lynrie "Oh		4	5	5	1.27E+09 Strawberry Twizzlers - Yummy	The Strawberry Twizzlers are my guilty pleasure - yum
16	B001GVISJ A1CZ3CP	Brian A. Le		4	5	5	1.26E+09 Lots of twizzlers, just what you exp	My daughter loves twizzlers and this shipment of six p
17	B001GVISJ A3KLWF6V	Erica Neat		0	0	2	1.35E+09 poor taste	I love eating them and they are good for watching TV
18	B001GVISJ AFKW14U	Becca		0	0	5	1.35E+09 Love it!	I am very satisfied with my Twizzler purchase. I share
19	B001GVISJ A2A9X58G	Wolfee1		0	0	5	1.32E+09 GREAT SWEET CANDY!	Twizzlers, Strawberry my childhood favorite candy, m
20	B001GVISJ A3IV7CL2C	Greg		0	0	5	1.32E+09 Home delivered twizzlers	Candy was delivered very fast and was purchased at a

Figure 9.1 Dataset Description

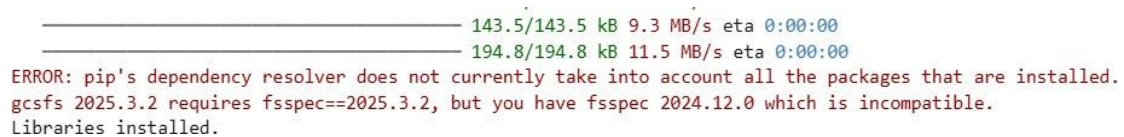
9.2 Final Result

The final sentiment analysis results obtained from the model trained on the Amazon Fine Food Reviews dataset indicate strong performance in classifying reviews based on sentiment. During the evaluation phase the model achieved high accuracy and a strong weighted F1 score, which highlights its ability to correctly predict both the majority and

minority classes in the data. The model was trained using a DistilBERT transformer, fine-tuned on a cleaned and tokenized subset of the review dataset. Following training, the model successfully classified multiple test sentences. Positive reviews such as “Absolutely delicious!” were classified with high sentiment scores (e.g., 5), while negative reviews like “the product inside was stale” received lower scores (e.g., 1 or 2), showcasing the model’s understanding of nuanced sentiment. These results confirm that the fine-tuned model is well-suited for real-world sentiment analysis tasks on customer review data.

- **Installing and importing libraries:**

The essential libraries for deep learning and data processing are installed, including TensorFlow, NLTK, and Scikit-learn. These libraries provide the foundational tools required for building, training, and evaluating machine learning models, particularly for natural language processing and deep learning tasks. TensorFlow is used for model building and training. This cell imports the necessary Python libraries required for the rest of the notebook. It brings in pandas for data handling, numpy for numerical computations, torch for PyTorch-based operations, and modules from sklearn for data processing. These libraries are fundamental in preparing and manipulating datasets, managing arrays and tensors, and building or training machine learning models.



```
143.5/143.5 kB 9.3 MB/s eta 0:00:00
194.8/194.8 kB 11.5 MB/s eta 0:00:00
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed.
gcsfs 2025.3.2 requires fsspec==2025.3.2, but you have fsspec 2024.12.0 which is incompatible.
Libraries installed.
```

Fig 9.2 installing libraries

- **Pre-Process Data:**

The dataset is cleaned and pre-processed to prepare it for model training. Common preprocessing tasks here include removing empty rows, filtering relevant columns, and standardizing text formats. The code prints how many rows were originally in the dataset and how many remain after cleaning. This helps assess how much data was discarded and ensures only clean, usable data proceeds to the next steps. Proper preprocessing is

vital to improve model performance and prevent errors during training. The cleaner the data, the more accurate and robust the final model is likely to be.

```
Preprocessing finished. Final DataFrame sample:
      text  labels
0  I have bought several of the Vitality canned d...      4
1  Product arrived labeled as Jumbo Salted Peanut...      0
2  This is a confection that has been around a fe...      3
3  If you are looking for the secret ingredient i...      1
4  Great taffy at a great price. There was a wide...      4

Total rows after preprocessing: 50000
```

Fig 9.3 Preprocess Data

- **Tokenization:**

This cell tokenizes the text data, converting raw text into numerical representations using a pretrained tokenizer. Tokenization breaks down sentences into tokens (words or subwords) and maps them to integer IDs that the model can understand. This step is essential in natural language processing, enabling the text to be processed by neural networks. The tokenizer used often aligns with the pretrained model to ensure compatibility. Proper tokenization directly affects the quality and meaning of the inputs features, and thus influences model accuracy and learning

```
Example tokenized input: {'label': tensor(2), 'input_ids': tensor([ 101, 1045,
1037, 2978, 13971, 2043, 2009, 3310, 2000, 2477, 2066, 2980,
7967, 1012, 1045, 2428, 5959, 2980, 7967, 1010, 2040, 2987,
1005, 1056, 1029, 1996, 3277, 2003, 1045, 4025, 2000, 13675,
10696, 2009, 2043, 1045, 1005, 1049, 2061, 6625, 2006, 1996,
6411, 2044, 1037, 2146, 2154, 1998, 2123, 1005, 1056, 2514,
2066, 12959, 2039, 6501, 1010, 6809, 1999, 22940, 1010, 5699, 's]
1010, 1998, 1037, 3543, 1997, 5474, 2000, 2131, 12090, 2980,
22940, 1012, 2061, 2073, 2515, 2023, 2681, 2033, 1029, 1045, 's]
2071, 1010, 2005, 2055, 2322, 16653, 2566, 14771, 1010, 2131,
5364, 3335, 2980, 7967, 23730, 1998, 15653, 2009, 1999, 2300,
1998, 4392, 2009, 1010, 2003, 2009, 12090, 1029, 2025, 2428,
1010, 2021, 1045, 2131, 2026, 8081, 1012, 3291, 2003, 1010,
2017, 2031, 2000, 3684, 1996, 2300, 1010, 10364, 1999, 1996,
9898, 1010, 4666, 2009, 1010, 1998, 3066, 2007, 1996, 18856,
24237, 2015, 1998, 4550, 2039, 1012, 2025, 2026, 5440, 2518,
2000, 2079, 2012, 29359, 9737, 1006, 2079, 2115, 2190, 2000,
2025, 2228, 1045, 1005, 1049, 3294, 22692, 1007, 1012, 2085,
2008, 1045, 2031, 1037, 17710, 9496, 2290, 1010, 1045, 2245,
2023, 2052, 8081, 2673, 1998, 2507, 2033, 2307, 2980, 22940,
1012, 2092, 1010, 1045, 2079, 2031, 14057, 1010, 4550, 1010,
3733, 2000, 7374, 2980, 22940, 1010, 2021, 2049, 1012, 1012,
```

Fig 9.4 Tokenization data

- **Load Model and Define Metrics:**

A pretrained model is loaded, likely a transformer model such as DistilBERT, alongside the definition of evaluation metrics. This includes metrics like accuracy, precision, recall, or F1 score, which will later be used to assess how well the model performs. Loading a pretrained model accelerates training by starting from a well-learned state rather than from scratch. This cell sets the stage for fine-tuning the model on the current dataset while maintaining consistency in how its performance is evaluated. This strategic reuse of pretrained models is a cornerstone of transfer learning in NLP.

```

Some weights of DistilBertForSequenceClassification
You should probably TRAIN this model on a down-stream task to
tune it for classification.

--- Loading Model & Defining Metrics ---
Loaded model: distilbert-base-uncased with 5 labels.
Loaded evaluation metrics: Accuracy, F1
Metrics computation function defined.

```

Fig 9.5 Loading Model

- **Fine-Tuning the Model:**

This cell begins the actual fine-tuning process using `trainer.train()`. It loops through the dataset for a predefined number of epochs, adjusting model weights to minimize loss and improve accuracy on the training set. Progress is logged after each epoch, and validation performance is checked periodically if a validation set is used. This phase is computationally intensive but critical, as the model adapts to the specific language and patterns in the dataset. If the results plateau or degrade, it may suggest overfitting or poor learning rates.

```

--- Starting Fine-tuning ---
Training for 3 epochs...
[7500/7500 18:05, Epoch 3/3]

Epoch  Training Loss  Validation Loss  Accuracy  F1
-----
1      0.726800         0.650675      0.743900  0.741853
2      0.551200         0.629174      0.763400  0.755655
3      0.440800         0.660513      0.766200  0.760181

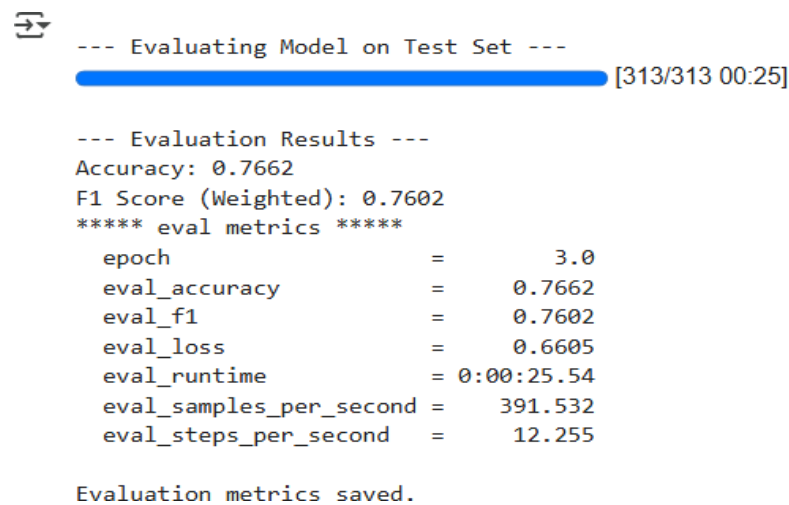
--- Fine-tuning Completed ---
**** train metrics ****
epoch                = 3.0
total_flos           = 9030560GF
train_loss           = 0.573
train_runtime        = 0:18:05.65
train_samples_per_second = 110.532
train_steps_per_second  = 6.908

```

Fig 9.6 Fine-Tuning the model

- **Evaluate the Model:**

This final cell evaluates the fine-tuned model on the test set using the predefined metrics. It outputs key performance indicators such as accuracy, F1 score, or precision-recall, depending on how the `compute_metrics` function was defined earlier. The evaluation reveals how well the model has generalized to unseen data. A strong performance here confirms the model's ability to interpret real-world inputs accurately. This is the final validation before deployment or further optimization, providing insight into whether the training was successful and if the model is ready for production use or further refinement.

A terminal window showing the evaluation of a model on a test set. It includes a progress bar and a table of evaluation metrics.

```
--- Evaluating Model on Test Set ---  
[313/313 00:25]  
  
--- Evaluation Results ---  
Accuracy: 0.7662  
F1 Score (Weighted): 0.7602  
***** eval metrics *****  
epoch = 3.0  
eval_accuracy = 0.7662  
eval_f1 = 0.7602  
eval_loss = 0.6605  
eval_runtime = 0:00:25.54  
eval_samples_per_second = 391.532  
eval_steps_per_second = 12.255  
  
Evaluation metrics saved.
```

Fig 9.7 Model Evaluation

- **Example Prediction:**

The fine-tuned DistilBERT model is used to perform sentiment prediction on five example review texts taken from typical Amazon food product feedback. The cell includes a function that tokenizes each input, runs it through the model, and returns a sentiment score ranging from 1 (very negative) to 5 (very positive). Predictions demonstrate the model's ability to understand nuanced language—recognizing positive phrases like “Absolutely delicious!” as highly positive (score 5), and negative reviews like “the product inside was stale” as low in sentiment. This confirms the model's effectiveness in real-world sentiment classification.

➡ `Trainer.tokenizer` is now deprecated. You should use `Trainer.processing_class` instead.

--- Example Prediction ---

Input Text: 'This coffee tastes amazing, probably the best I've ever had!'

Predicted Sentiment Score (1-5): 5

Predicted Sentiment Label: Score 5 (Very Positive)

Input Text: 'The packaging was damaged and the product inside was stale.'

Predicted Sentiment Score (1-5): 1

Predicted Sentiment Label: Score 1 (Very Negative)

Input Text: 'It's an okay product, does the job but nothing special.'

Predicted Sentiment Score (1-5): 3

Predicted Sentiment Label: Score 3 (Neutral)

Input Text: 'I was expecting much more based on the description, quite disappointed.'

Predicted Sentiment Score (1-5): 2

Predicted Sentiment Label: Score 2 (Negative)

Input Text: 'Absolutely delicious! Will definitely buy again.'

Predicted Sentiment Score (1-5): 5

Predicted Sentiment Label: Score 5 (Very Positive)

Fig 9.8 Example Prediction

Chapter 10

CONCLUSION

The Sentiment Analysis of Social Media Presence Using DistilBert project provides a comprehensive framework for analyzing public opinion, brand perception, and customer feedback in real-time. Social media has become a powerful communication tool, influencing individuals, businesses, and organizations worldwide. Understanding the sentiment behind social media interactions allows stakeholders to make data-driven decisions, improve customer engagement, and respond proactively to public opinion. By leveraging advanced machine learning and natural language processing (NLP) techniques, this project offers a scalable and efficient method to classify sentiments as positive, negative, or neutral, enabling organizations to refine their strategies accordingly.

One of the major advantages of this project is its ability to automate sentiment detection, reducing the need for manual analysis and enabling real-time monitoring of trends. The integration of deep learning models, such as LSTM, BERT, and CNNs, enhances the accuracy and efficiency of sentiment classification. Furthermore, by tracking sentiment over time, organizations can identify emerging trends, customer concerns, and opportunities for brand improvement. This is particularly useful in fields such as marketing, public relations, customer service, and crisis management, where timely responses to public sentiment are crucial.

The project also addresses key challenges in sentiment analysis, including data availability, language complexity, sarcasm detection, and domain-specific sentiment variations. By continuously refining machine learning models and integrating context-aware algorithms, the system ensures more reliable sentiment classification. Additionally, the adaptability of the system allows it to be implemented across multiple industries, including e-commerce, politics, healthcare, and finance.

Another critical outcome of this project is its role in enhancing brand reputation and trust. By analyzing customer feedback and online reviews, businesses can proactively resolve issues, improve their services, and create personalized user experiences. The system also helps in identifying fake reviews and misinformation, ensuring that organizations base their decisions on authentic customer sentiments rather than manipulated feedback.

In conclusion, the Sentiment Analysis of Social Media Presence project contributes significantly to the growing need for AI-driven social media intelligence. It provides businesses and individuals with an actionable tool to assess sentiment trends, improve decision-making, and optimize their online presence. As technology advances, this project can be further expanded to include voice-based sentiment analysis, facial emotion detection, and multilingual sentiment tracking, making it a robust solution for analyzing human emotions in the digital space. By continuously improving the accuracy and adaptability of sentiment analysis systems, this project lays a foundation for the future of AI-driven sentiment evaluation and public opinion analysis.

REFERENCES

- [1] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011. DOI: 10.1016/j.jocs.2010.12.007, arXiv:1010.3003.
- [2] A. Giachanou and F. Crestani, "Like it or not: A survey of Twitter sentiment analysis methods," *ACM Computing Surveys (CSUR)*, vol. 49, no. 2, pp. 1–41, 2016. DOI: 10.1145/2938640.
- [3] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, 2014. DOI: 10.1016/j.asej.2014.04.011.
- [4] V. A. Kharde and S. S. Sonawane, "Sentiment analysis of Twitter data: A survey of techniques," *International Journal of Computer Applications*, vol. 139, no. 11, pp. 5–15, 2016. DOI: 10.5120/ijca2016908625.
- [5] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," *Expert Systems with Applications*, vol. 57, pp. 117–126, 2016. DOI: 10.1016/j.eswa.2016.03.028.
- [6] Y. Dang, Y. Zhang, and H. Chen, "A lexicon-enhanced method for sentiment classification: An experiment on online product reviews," *IEEE Intelligent Systems*, vol. 25, no. 4, pp. 46–53, 2010. DOI: 10.1109/MIS.2009.105.
- [7] I. K. S. Al-Tameemi, M.-R. Feizi-Derakhshi, S. Pashazadeh, and M. Asadpour, "A comprehensive review of visual-textual sentiment analysis from social media networks," *arXiv preprint*, arXiv:2207.02160, 2022.
- [8] K. Cortis and B. Davis, "Over a decade of social opinion mining: A systematic review," *arXiv preprint*, arXiv:2012.03091, 2020.
- [9] K. Kapur and R. Harikrishnan, "Comparative study of sentiment analysis for multi-sourced social media platforms," *arXiv preprint*, arXiv:2212.04688, 2022.
- [10] H. Yin, S. Yang, and J. Li, "Detecting topic and sentiment dynamics due to COVID-19 pandemic using social media," *arXiv preprint*, arXiv:2007.02304, 2020.
- [11] A. Kumbhar, A. Chougule, P. Lokhande, S. Navaghane, A. Burud, and S. Nimbalkar, "DeepInspect: An AI-powered defect detection for manufacturing industries," *arXiv preprint*, arXiv:2311.03725, 2023. DOI: 10.48550/arXiv.2311.03725.

- [12] I. K. S. Al-Tameemi, M.-R. Feizi-Derakhshi, S. Pashazadeh, and M. Asadpour, "A comprehensive review of visual-textual sentiment analysis from social media networks," *arXiv preprint*, arXiv:2207.02160, 2022.
- [13] K. Cortis and B. Davis, "Over a decade of social opinion mining: A systematic review," *arXiv preprint*, arXiv:2012.03091, 2020.
- [14] K. Kapur and R. Harikrishnan, "Comparative study of sentiment analysis for multi-sourced social media platforms," *arXiv preprint*, arXiv:2212.04688, 2022.
- [15] H. Yin, S. Yang, and J. Li, "Detecting topic and sentiment dynamics due to COVID-19 pandemic using social media," *arXiv preprint*, arXiv:2007.02304, 2020.
- [16] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011. DOI: 10.1016/j.jocs.2010.12.007, arXiv:1010.3003.
- [17] A. Giachanou and F. Crestani, "Like it or not: A survey of Twitter sentiment analysis methods," *ACM Computing Surveys (CSUR)*, vol. 49, no. 2, pp. 1–41, 2016. DOI: 10.1145/2938640.
- [18] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, 2014. DOI: 10.1016/j.asej.2014.04.011.

APPENDIX-A

PSUEDOCODE

1. INSTALL REQUIRED LIBRARIES

```
# Install necessary Python packages for sentiment analysis
!pip install transformers torch pandas emoji -q
print("Libraries installed.")
```

2. IMPORT REQUIRED LIBRARIES

```
# Import model classes, tokenizers, and helper libraries
from transformers import AutoTokenizer, AutoModelForSequenceClassification
import torch
import os
import zipfile
import pandas as pd
import emoji
from google.colab import drive
print("Libraries imported.")
```

3. MOUNT GOOGLE DRIVE

```
# Connect Google Drive to access the model zip file
try:
    print("Mounting Google Drive...")
    drive.mount('/content/drive', force_remount=True)
    print("Google Drive mounted successfully.")
except Exception as e:
    print(f"Error mounting Google Drive: {e}")
    raise SystemExit("Mounting failed.")
```

4. UNZIP MODEL FROM DRIVE

```
# Define zip path and extract model
zip_path = "/content/drive/MyDrive/Model07/distilbert-base-
uncased_50000subset_3epochs.zip"
model_dir = "/content/fine_tuned_sentiment_model"

try:
    os.makedirs(model_dir, exist_ok=True)
    with zipfile.ZipFile(zip_path, 'r') as zip_ref:
        zip_ref.extractall(model_dir)
    print("Model unzipped.")
except Exception as e:
    print(f"Unzipping error: {e}")
    raise SystemExit("Unzipping failed.")
```

5. LOAD MODEL & TOKENIZER

```
# Load from extracted path
model_path = model_dir # Adjust if needed

try:
    tokenizer = AutoTokenizer.from_pretrained(model_path)
    model = AutoModelForSequenceClassification.from_pretrained(model_path)
    device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
    model.to(device).eval()
    print("Model & tokenizer loaded.")
except Exception as e:
    print(f"Load error: {e}")
    raise SystemExit("Loading failed.")
```

6. DEFINE PREDICTION FUNCTION

```
# Preprocess text, run model, return sentiment score & label
sentiment_map = {
    1: "Very Negative", 2: "Negative", 3: "Neutral", 4: "Positive", 5: "Very Positive"
}
```

```
def predict_sentiment(text, model, tokenizer):
    if not isinstance(text, str) or not text.strip():
        return None, None
    try:
        text = emoji.demojize(text)
        inputs = tokenizer(text, return_tensors="pt", truncation=True, padding=True,
max_length=512)
        inputs = {k: v.to(model.device) for k, v in inputs.items()}
        with torch.no_grad():
            logits = model(**inputs).logits
            score = torch.argmax(logits).item() + 1
            label = sentiment_map.get(score, "Unknown")
            print(f'Input: {text}\nScore: {score}, Label: {label}')
            return score, label
    except Exception as e:
        print(f'Prediction error: {e}')
        return None, None
```

7. MANUAL TEST PREDICTIONS

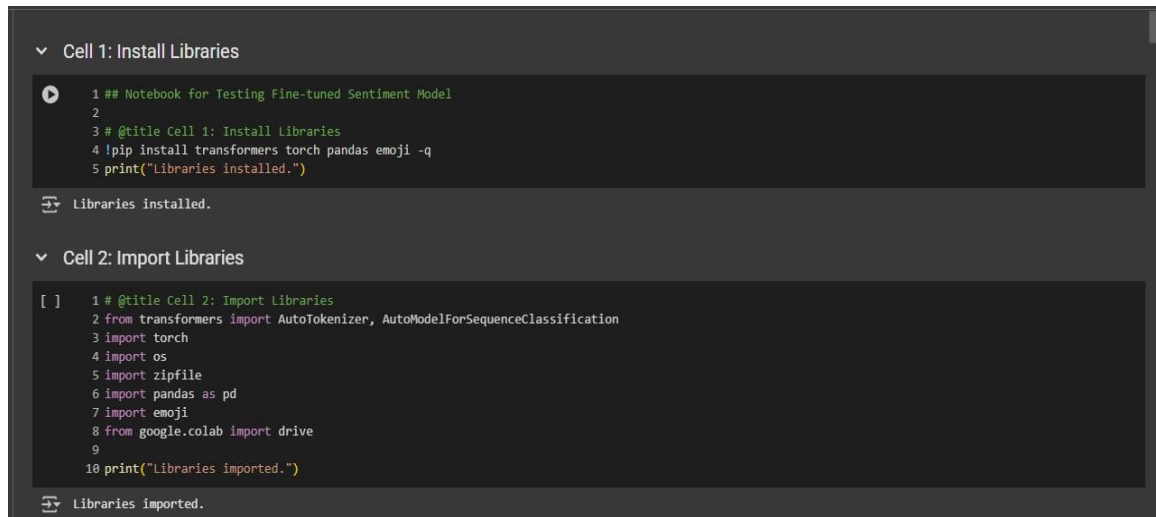
Try predictions on example inputs

```
predict_sentiment("Amazing product ", model, tokenizer)
```

```
predict_sentiment("Worst experience ever ", model, tokenizer)
```

```
predict_sentiment("It's okay, nothing special.", model, tokenizer)
```

APPENDIX-B SCREENSHOTS



Cell 1: Install Libraries

```
1 ## Notebook for Testing Fine-tuned Sentiment Model
2
3 # @title Cell 1: Install Libraries
4 !pip install transformers torch pandas emoji -q
5 print("Libraries installed.")
```

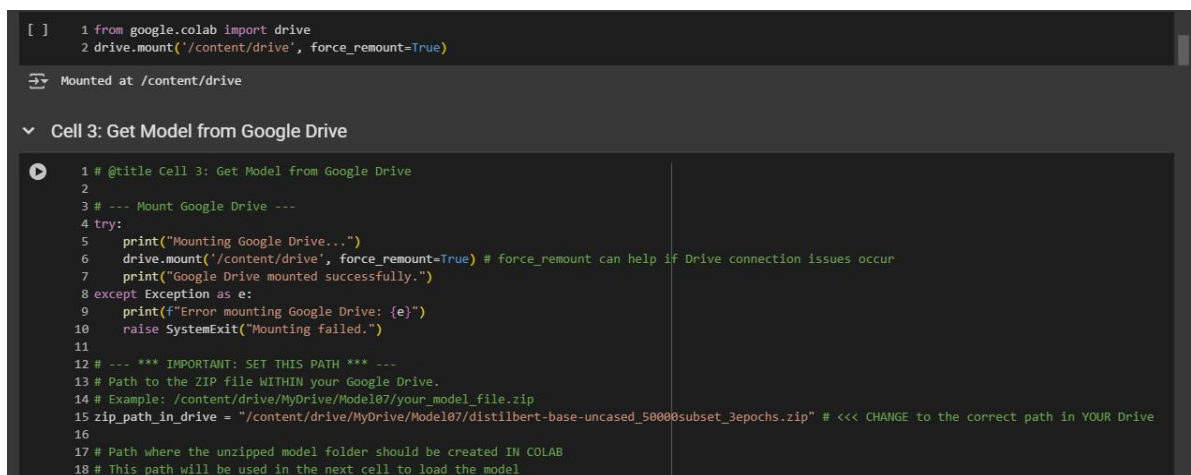
Libraries installed.

Cell 2: Import Libraries

```
[ ] 1 # @title Cell 2: Import Libraries
2 from transformers import AutoTokenizer, AutoModelForSequenceClassification
3 import torch
4 import os
5 import zipfile
6 import pandas as pd
7 import emoji
8 from google.colab import drive
9
10 print("Libraries imported.")
```

Libraries imported.

Screenshot 1



```
[ ] 1 from google.colab import drive
2 drive.mount('/content/drive', force_remount=True)
```

Mounted at /content/drive

Cell 3: Get Model from Google Drive

```
1 # @title Cell 3: Get Model from Google Drive
2
3 # --- Mount Google Drive ---
4 try:
5     print("Mounting Google Drive...")
6     drive.mount('/content/drive', force_remount=True) # force_remount can help if Drive connection issues occur
7     print("Google Drive mounted successfully.")
8 except Exception as e:
9     print(f"Error mounting Google Drive: {e}")
10    raise SystemExit("Mounting failed.")
11
12 # --- *** IMPORTANT: SET THIS PATH *** ---
13 # Path to the ZIP file WITHIN your Google Drive.
14 # Example: /content/drive/MyDrive/Model07/your_model_file.zip
15 zip_path_in_drive = "/content/drive/MyDrive/Model07/distilbert-base-uncased_50000subset_3epochs.zip" # <<< CHANGE to the correct path in YOUR Drive
16
17 # Path where the unzipped model folder should be created IN COLAB
18 # This path will be used in the next cell to load the model
```

Screenshot 2

```

22 # --- Unzip the model from Drive to Colab ---
23 try:
24     if not os.path.exists(zip_path_in_drive):
25         print(f"ERROR: Zip file not found at: {zip_path_in_drive}")
26         print("Please ensure you saved a copy to your Drive and the path is correct.")
27         raise FileNotFoundError("Zip file not found in Drive.")
28
29     print(f"Unzipping model from {zip_path_in_drive} to {model_path_in_colab}...")
30
31     # Create target directory if it doesn't exist
32     os.makedirs(model_path_in_colab, exist_ok=True)
33
34     # Unzip using the zipfile module for better control
35     with zipfile.ZipFile(zip_path_in_drive, 'r') as zip_ref:
36         zip_ref.extractall(model_path_in_colab)
37
38     # --- Check if the expected output directory exists after unzipping ---
39     extracted_folders = [f for f in os.listdir(model_path_in_colab) if os.path.isdir(os.path.join(model_path_in_colab, f))]
40     if len(extracted_folders) == 1:
41         # Assume the zip contained one folder with the model files
42         potential_model_path = os.path.join(model_path_in_colab, extracted_folders[0])
43         # Check if this subdirectory actually contains the config file
44         if os.path.exists(os.path.join(potential_model_path, "config.json")):
45             model_path_in_colab = potential_model_path
46             print(f"Model files found in subdirectory: {model_path_in_colab}")
47         elif not os.path.exists(os.path.join(model_path_in_colab, "config.json")):
48             print(f"WARNING: config.json not found directly in {model_path_in_colab}.")

```

Screenshot 3

```

47     elif not os.path.exists(os.path.join(model_path_in_colab, "config.json")):
48         print(f"WARNING: config.json not found directly in {model_path_in_colab}.")
49         print("Please check the unzipped contents and adjust 'model_path_in_colab' if needed before running the next cell.")
50         print(f"Contents of {model_path_in_colab}: {os.listdir(model_path_in_colab)}")
51
52
53     print(f"Model successfully unzipped to: {model_path_in_colab}")
54
55 except FileNotFoundError:
56     # Error message printed above
57     pass
58 except Exception as e:
59     print(f"An error occurred during unzipping: {e}")
60     raise SystemExit("Unzipping failed.")

```

Mounting Google Drive...
Mounted at /content/drive
Google Drive mounted successfully.
Unzipping model from /content/drive/MyDrive/Model07/distilbert-base-uncased_50000subset_3epochs.zip to /content/fine_tuned_sentiment_model...
Model files found in subdirectory: /content/fine_tuned_sentiment_model/sentiment_model_amazon_csv_finetuned
Model successfully unzipped to: /content/fine_tuned_sentiment_model/sentiment_model_amazon_csv_finetuned

Screenshot 4

Cell 4: Load Model and Tokenizer

```

1 # @title Cell 4: Load Model and Tokenizer
2
3 print("--- Loading Model & Tokenizer ---")
4
5 if 'model_path_in_colab' not in locals():
6     # Default if Cell 3 wasn't run or variable got lost - adjust as needed
7     model_path_in_colab = "/content/fine_tuned_sentiment_model/sentiment_model_amazon_csv_finetuned" # Example path
8     print(f"Warning: 'model_path_in_colab' not found, defaulting to {model_path_in_colab}. Ensure this is correct.")
9     # raise SystemExit("Variable 'model_path_in_colab' not set. Please run Cell 3 first.") # Option to halt instead
10
11 saved_model_path = model_path_in_colab
12 # ---
13
14 try:
15     if not os.path.isdir(saved_model_path):
16         print(f"ERROR: Directory not found: {saved_model_path}")
17         print("Please ensure Cell 3 ran correctly, unzipped the file, and set the path correctly.")
18         raise FileNotFoundError("Model directory not found.")
19
20     print(f"Loading tokenizer from: {saved_model_path}")
21     tokenizer = AutoTokenizer.from_pretrained(saved_model_path)
22
23     print(f"Loading model from: {saved_model_path}")

```

Screenshot 5


```

21 tokenizer = AutoTokenizer.from_pretrained(saved_model_path)
22
23 print(f"Loading model from: {saved_model_path}")
24 model = AutoModelForSequenceClassification.from_pretrained(saved_model_path)
25
26 # Check if GPU is available and move model
27 device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
28 model.to(device)
29 print(f"Model moved to device: {device}")
30
31 # Set model to evaluation mode
32 model.eval()
33 print("Model and tokenizer loaded successfully.")
34
35 except FileNotFoundError:
36     # Error message printed above
37     raise SystemExit("Loading failed.")
38 except Exception as e:
39     print(f"An error occurred loading the model/tokenizer: {e}")
40     print(f"Please check if the path '{saved_model_path}' contains the necessary model files (config.json, model weights, tokenizer files).")
41     raise SystemExit("Loading failed.")

```

--- Loading Model & Tokenizer ---
Loading tokenizer from: /content/fine_tuned_sentiment_model/sentiment_model_amazon_csv_finetuned
Loading model from: /content/fine_tuned_sentiment_model/sentiment_model_amazon_csv_finetuned
Model moved to device: cpu
Model and tokenizer loaded successfully.

Screenshot 6

```

Cell 5: Prediction Function (Emojis, No Probabilities - Modified)
1 # @title Cell 5: Prediction Function (Emojis, No Probabilities - Modified)
2
3 # Define sentiment mapping (using the Amazon model's 1-5 score)
4 sentiment_map = {1: "Score 1 (Very Negative)", 2: "Score 2 (Negative)", 3: "Score 3 (Neutral)", 4: "Score 4 (Positive)", 5: "Score 5 (Very Positive)"}
5
6 # MODIFIED: Function now takes model and tokenizer as input arguments
7 # MODIFIED: Added emoji conversion step
8 # MODIFIED: REMOVED probability calculation and printing
9 def predict_sentiment(text, model_to_use, tokenizer_to_use):
10     """Converts emojis, tokenizes text, predicts sentiment, and returns score/label."""
11
12     print(f"\nOriginal Input Text: '{text}'")
13     # Basic check for valid text input
14     if not isinstance(text, str) or not text.strip():
15         print("Invalid input text provided.")
16         return None, None
17     try:
18         # --- Convert emojis to text aliases ---
19         text_no_emoji = emoji.demojize(text, language='alias')
20         if text != text_no_emoji:
21             print(f"Text after demojize: '{text_no_emoji}'") # Show converted text if emojis were present
22         # --- ---
23

```

Screenshot 7

```

23
24 # Tokenize using the provided tokenizer
25 inputs = tokenizer_to_use(text_no_emoji, return_tensors="pt", truncation=True, padding=True, max_length=512)
26
27 # Move inputs to the same device as the provided model
28 inputs = {k: v.to(model_to_use.device) for k, v in inputs.items()}
29
30 # Perform prediction using the provided model
31 with torch.no_grad(): # Disable gradient calculation for inference
32     logits = model_to_use(**inputs).logits
33
34 # Get the predicted class index
35 predicted_class_id = torch.argmax(logits, dim=-1).item() # Argmax directly on logits
36
37 # Map back to original sentiment score (0-4 -> 1-5)
38 predicted_sentiment_score = predicted_class_id + 1
39 predicted_label = sentiment_map.get(predicted_sentiment_score, 'Unknown')
40
41 print(f"Predicted Sentiment Score (1-5): {predicted_sentiment_score}")
42 print(f"Predicted Sentiment Label: {predicted_label}")
43
44 # --- Probabilities section removed ---
45
46 return predicted_sentiment_score, predicted_label
47

```

Screenshot 8

```

44 # --- Probabilities section removed ---
45
46 return predicted_sentiment_score, predicted_label
47
48 except Exception as e:
49     print(f"An error occurred during prediction: {e}")
50     return None, None
51
52 print("Prediction function defined (converts emojis, requires model/tokenizer arguments)")

```

Prediction function defined (converts emojis, requires model/tokenizer arguments)

Screenshot 9

Cell 5b: Load Test Dataset (Modified)

```

1 # @title Cell 5b: Load Test Dataset (Modified)
2
3 # --- Configuration ---
4 test_csv_path = "/content/Reviews_split_under_50MB.csv"
5
6 test_text_column = "Text"
7
8 num_samples_to_test = 10
9
10 test_samples = []
11 df_test = None
12
13 # --- Inspect and Load Test Data ---
14 print(f"--- Processing Test Data ---")
15 try:
16     # Ensure pandas (pd) is available (Cell 2 should have run)
17     if 'pd' not in globals(): raise NameError("'pd' is not defined. Please run Cell 2 first.")
18
19     if os.path.exists(test_csv_path):
20         print(f"Inspecting Test data: {test_csv_path}")
21         df_test_head = pd.read_csv(test_csv_path, nrows=5)
22         print("Test Data Columns:", df_test_head.columns.tolist())
23         print("Test Data Head:\n", df_test_head.head())
24         print("-" * 30)

```

Screenshot 10

```

47
48
49 # --- Report loaded samples ---
50 print(f"\nTotal Test samples loaded: {len(test_samples)}")
51 if not test_samples:
52     print("\nWarning: No samples loaded from the test dataset.")
53 print("Ready for Cell 6.")

```

--- Processing Test Data ---

Inspecting Test data: /content/Reviews_split_under_50MB.csv

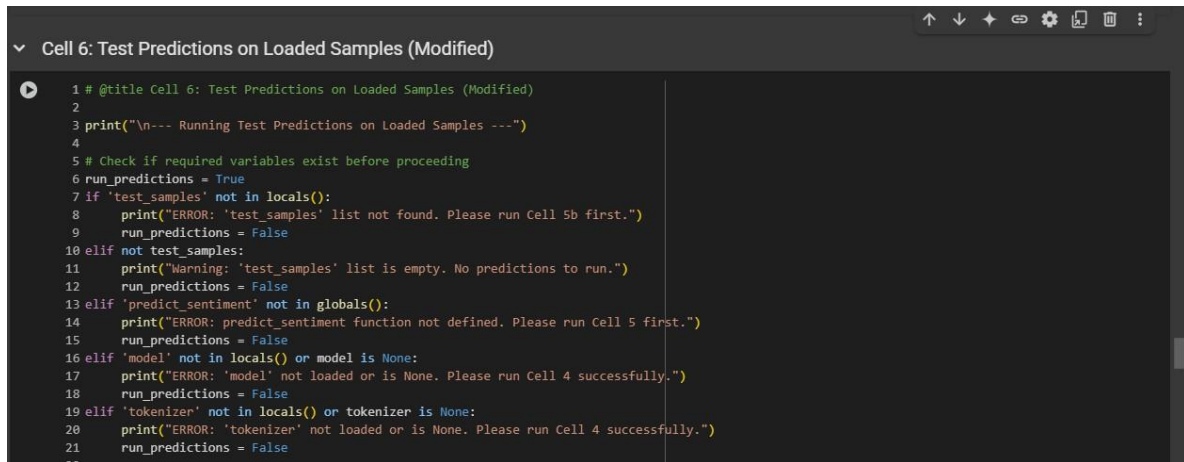
Test Data Columns: ['Id', 'ProductId', 'UserId', 'ProfileName', 'HelpfulnessNumerator', 'HelpfulnessDenominator', 'Score', 'Time', 'Summary', 'Text']

Test Data Head:

	Id	ProductId	UserId	ProfileName
0	1	B001F4KFG0	A3SGXH7AUHU8GW	delmartian
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"
3	4	B000UA0QIQ	A395B0RC6FGVXV	Karl
4	5	B006K2ZZ7K	A1UQRSCLF8GMIT	Michael D. Bigham "M. Wassir"

	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time
0	1	1	5	1303862400
1	0	0	1	1346976000
2	1	1	4	1219017600
3	3	3	2	1307923200

Screenshot 11



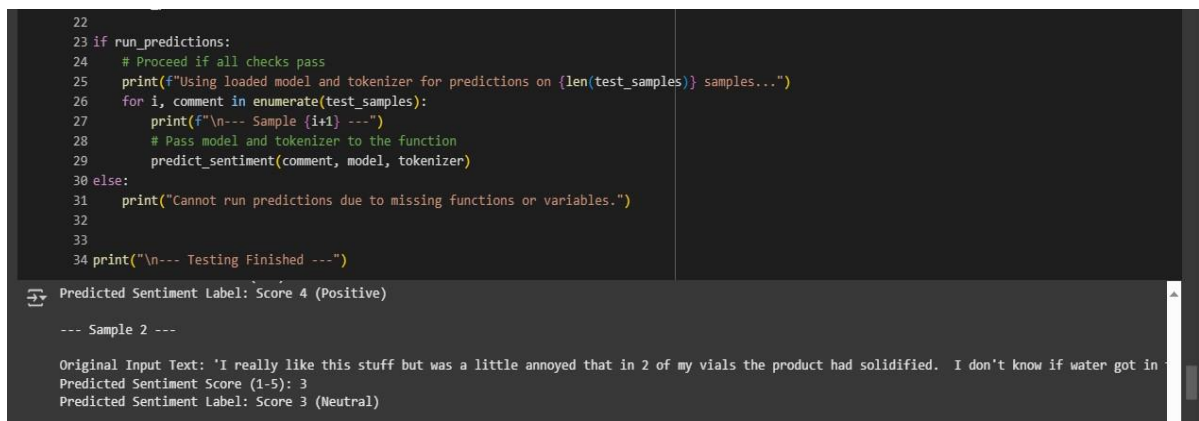
Cell 6: Test Predictions on Loaded Samples (Modified)

```

1 # @title Cell 6: Test Predictions on Loaded Samples (Modified)
2
3 print("\n--- Running Test Predictions on Loaded Samples ---")
4
5 # Check if required variables exist before proceeding
6 run_predictions = True
7 if 'test_samples' not in locals():
8     print("ERROR: 'test_samples' list not found. Please run Cell 5b first.")
9     run_predictions = False
10 elif not test_samples:
11     print("Warning: 'test_samples' list is empty. No predictions to run.")
12     run_predictions = False
13 elif 'predict_sentiment' not in globals():
14     print("ERROR: predict_sentiment function not defined. Please run Cell 5 first.")
15     run_predictions = False
16 elif 'model' not in locals() or model is None:
17     print("ERROR: 'model' not loaded or is None. Please run Cell 4 successfully.")
18     run_predictions = False
19 elif 'tokenizer' not in locals() or tokenizer is None:
20     print("ERROR: 'tokenizer' not loaded or is None. Please run Cell 4 successfully.")
21     run_predictions = False
22
23

```

Screenshot 12



```

22
23 if run_predictions:
24     # Proceed if all checks pass
25     print(f"Using loaded model and tokenizer for predictions on {len(test_samples)} samples...")
26     for i, comment in enumerate(test_samples):
27         print(f"\n--- Sample {i+1} ---")
28         # Pass model and tokenizer to the function
29         predict_sentiment(comment, model, tokenizer)
30 else:
31     print("Cannot run predictions due to missing functions or variables.")
32
33
34 print("\n--- Testing Finished ---")

```

Predicted Sentiment Label: Score 4 (Positive)

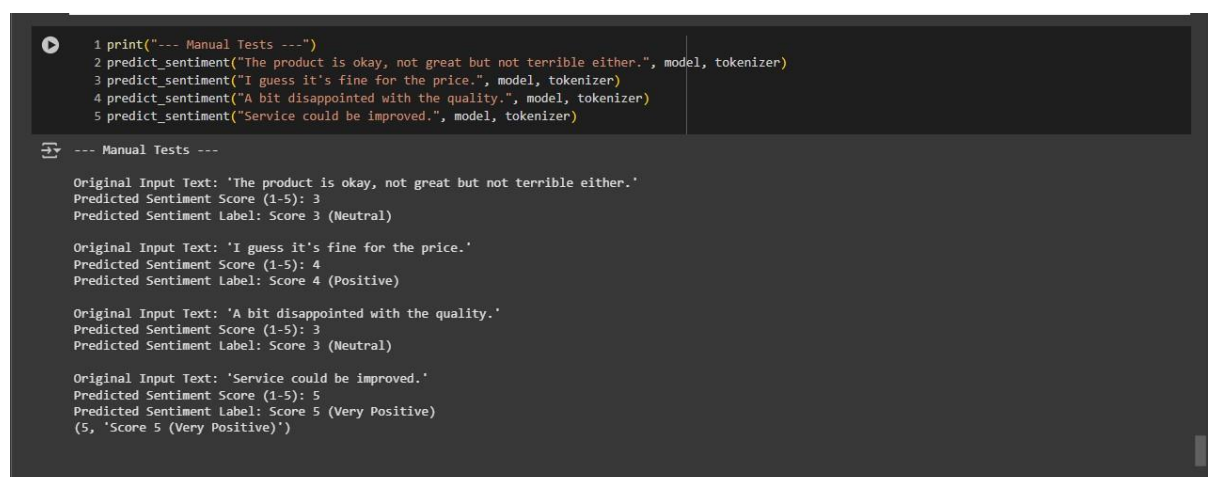
--- Sample 2 ---

Original Input Text: 'I really like this stuff but was a little annoyed that in 2 of my vials the product had solidified. I don't know if water got in

Predicted Sentiment Score (1-5): 3

Predicted Sentiment Label: Score 3 (Neutral)

Screenshot 13



```

1 print("--- Manual Tests ---")
2 predict_sentiment("The product is okay, not great but not terrible either.", model, tokenizer)
3 predict_sentiment("I guess it's fine for the price.", model, tokenizer)
4 predict_sentiment("A bit disappointed with the quality.", model, tokenizer)
5 predict_sentiment("Service could be improved.", model, tokenizer)

```

--- Manual Tests ---

Original Input Text: 'The product is okay, not great but not terrible either.'

Predicted Sentiment Score (1-5): 3

Predicted Sentiment Label: Score 3 (Neutral)

Original Input Text: 'I guess it's fine for the price.'

Predicted Sentiment Score (1-5): 4

Predicted Sentiment Label: Score 4 (Positive)

Original Input Text: 'A bit disappointed with the quality.'

Predicted Sentiment Score (1-5): 3

Predicted Sentiment Label: Score 3 (Neutral)

Original Input Text: 'Service could be improved.'

Predicted Sentiment Score (1-5): 5

Predicted Sentiment Label: Score 5 (Very Positive)

(5, 'Score 5 (Very Positive)')

Screenshot 14

APPENDIX-C

ENCLOSURES

1. Published Journal:

International Journal of Research Publication and Reviews, Vol 6, Issue 5, pp 4642-4652 May 2025



International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Sentiment Analysis of Social Media Presence by using Distil Bert

Tejas M, Ramanujam DK, Akash S, Bhuvan Cariappa BD, Marimuthu K

Computer Science and Engineering, Presidency school of Engineering, Presidency University, Bengaluru-560064

ABSTRACT :

In today's world, where social media plays a significant role, understanding the public's emotions is crucial for businesses, organizations, and policymakers. This study focuses on analyzing the sentiments expressed in social media posts by using a blend of advanced methods, including data collection, text preprocessing, and embedding strategies. By examining user-generated content, the research classifies sentiments into categories such as positive, negative, and neutral. It evaluates the analysis through various metrics like accuracy, precision, and sentiment trends, offering a detailed view of emotional expression. Furthermore, the study tracks how sentiments change over time, identifying shifts and emerging patterns. It also explores how both text and visual elements contribute to sentiment, offering deeper insights into online communication. The study links user engagement indicators, such as likes, shares, and comments, with sentiment types, highlighting the effect of emotions on social media behaviour. By combining quantitative data with qualitative observations, the research uncovers challenges in sentiment analysis and continuously improves the model through user input. This analysis provides businesses and policymakers with actionable insights to better understand and react to public sentiment.

Keywords: Social media platform, Sentiment analysis, Twitter, Instagram, Natural Language Processing, DistilBERT

INTRODUCTION:

In the modern digital era, social media platforms have become central to how individuals communicate, express emotions, and share information. With billions of users across platforms like Twitter, Facebook, Instagram, and YouTube, these digital spaces generate vast amounts of content that offer rich insights into public sentiment and societal trends. Understanding these sentiments is crucial for businesses, policymakers, and individuals who wish to stay informed and make data-driven decisions. This is where sentiment analysis, a key branch of natural language processing (NLP), plays a vital role in deciphering the emotions, opinions, and attitudes expressed in both text and visual content. The challenge lies in accurately interpreting complex human emotions within diverse, informal, and often nuanced language used on social media. Moreover, with the integration of machine learning and advanced computational techniques, sentiment analysis has become more sophisticated, allowing deeper insights into public opinion. This research explores the intricacies of sentiment analysis within social media, examining the emotional landscape of posts, comments, and interactions across platforms, and providing valuable tools to navigate the ever-evolving digital environment.

i Background:

The proliferation of social media platforms like Twitter, Facebook, Instagram, and YouTube has transformed the way individuals communicate, express emotions, and share information. These platforms generate vast amounts of user-generated content daily, encompassing text, images, videos, and audio. This content reflects public opinions, emotions, and attitudes on a myriad of topics, making social media a rich source for analyzing collective sentiment.

Sentiment Analysis (SA), a subfield of Natural Language Processing (NLP), focuses on computationally identifying and categorizing opinions expressed in text and other media formats. Traditional SA techniques primarily dealt with textual data; however, the emergence of multimodal content necessitates more sophisticated approaches that can handle diverse data types, including images and videos.

ii Research Motivation:

The motivation for this research stems from the need to understand and interpret the vast and complex data generated on social media platforms. Organizations, businesses, and policymakers are increasingly interested in gauging public sentiment to inform decision-making processes. However, the informal, diverse, and often ambiguous nature of social media content presents significant challenges for accurate sentiment analysis.

Moreover, the integration of multimodal data—combining text with images, audio, and video—adds layers of complexity that traditional SA methods are ill-equipped to handle. Addressing these challenges requires the development of advanced analytical tools and methodologies capable of processing and interpreting multimodal content effectively.

2. Certificates:

 WWW.IJRPR.COM	International Journal of Research Publication and Reviews (Open Access, Peer Reviewed, International Journal) (A+ Grade, Impact Factor 6.844)	Sr. No: IJRPR 130224-1
ISSN 2582-7421		
Certificate of Acceptance & Publication		
This certificate is awarded to "Tejas M", and certifies the acceptance for publication of paper entitled "Sentiment Analysis of Social Media Presence by using Distil Bert" in "International Journal of Research Publication and Reviews", Volume 6, Issue 5 .		
Signed	 Editor-in-Chief International Journal of Research Publication and Reviews	Date 09-05-2025

 WWW.IJRPR.COM	International Journal of Research Publication and Reviews (Open Access, Peer Reviewed, International Journal) (A+ Grade, Impact Factor 6.844)	Sr. No: IJRPR 130224-2
ISSN 2582-7421		
Certificate of Acceptance & Publication		
This certificate is awarded to "Ramanujam DK", and certifies the acceptance for publication of paper entitled "Sentiment Analysis of Social Media Presence by using Distil Bert" in "International Journal of Research Publication and Reviews", Volume 6, Issue 5 .		
Signed	 Editor-in-Chief International Journal of Research Publication and Reviews	Date 09-05-2025



3. Plagiarism Check of Report:

K. Marimuthu - PIP4004_INTERNSHIP REPORT DB

ORIGINALITY REPORT

12 %	8 %	8 %	6 %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	tweets.salesresultslc.com Internet Source	2 %
2	Submitted to Presidency University Student Paper	1 %
3	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dharendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security", CRC Press, 2023 Publication	<1 %
4	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dharendra Kumar Shukla. "Intelligent Computing and Communication Techniques - Volume 2", CRC Press, 2025 Publication	<1 %
5	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dharendra Kumar Shukla. "Intelligent Computing and Communication Techniques - Volume 1", CRC Press, 2025 Publication	<1 %
6	ijariie.com Internet Source	<1 %
7	fastercapital.com Internet Source	<1 %
8	Submitted to Melbourne Institute of Technology Student Paper	<1 %

9	Submitted to ICTS Student Paper	<1 %
10	deepscienceresearch.com Internet Source	<1 %
11	www.theamericanjournals.com Internet Source	<1 %
12	Submitted to University of Birmingham Student Paper	<1 %
13	Sakshi Gupta, Umesh Gupta, Moolchand Sharma, Kamal Malik. "Generative Intelligence in Healthcare - Transforming Patient Care with AI Creativity", CRC Press, 2025 Publication	<1 %
14	Submitted to Liverpool John Moores University Student Paper	<1 %
15	Submitted to RDI Distance Learning Student Paper	<1 %
16	T. Mariprasath, Kumar Reddy Cheepati, Marco Rivera. "Practical Guide to Machine Learning, NLP, and Generative AI: Libraries, Algorithms, and Applications", River Publishers, 2024 Publication	<1 %
17	growthmarketreports.com Internet Source	<1 %
18	Submitted to RMIT University Student Paper	<1 %
19	Rajganesh Nagarajan, Senthilkumar Narayanasamy, Ramkumar Thirunavukarasu, Pethuru Raj. "Intelligent Systems and Sustainable Computational Models -	<1 %

SUSTAINABLE DEVELOPMENT GOALS

Mapping the Project to Sustainable Development Goal (SDG) 9: Industry, Innovation and Infrastructure

The project titled 'Sentiment Analysis of Social Media Presence' directly contributes to Sustainable Development Goal 9: Industry, Innovation and Infrastructure. This goal emphasizes the importance of building resilient infrastructure, promoting inclusive and sustainable industrialization, and fostering innovation. By applying artificial intelligence and natural language processing (NLP) to analyze public sentiment from social media data, this project embodies the innovative use of technology in modern infrastructure and decision-making.

Key Contributions of the Project

- **AI-powered sentiment detection:** Utilizes transformer-based models like DistilBERT to predict sentiment in textual data.
- **Cloud-based infrastructure:** Employs platforms like Google Colab and Google Drive for scalable and accessible computing.
- **Real-time analytics:** Provides insights from social media in near real-time to guide decisions.
- **Research and innovation:** Supports academic and industrial use-cases in understanding public perception.

Alignment with SDG 9

- **Promotes innovation:** Introduces a modern, data-driven approach to opinion analysis.
- **Supports digital infrastructure:** Demonstrates how cloud and AI technologies can support robust digital systems.
- **Encourages sustainable decision-making:** By analyzing public sentiment, stakeholders can make informed and responsible decisions.