Kathmandu University

Department of Computer Science & Engineering

Dhulikhel, Kavre

A Project Report on

"Reelfeel"

[Code no: COMP 313]

(For partial fulfillment of III Year / II Semester in Computer Science)

Submitted by:

Samwarta Chitrakar (06)

Sujan Mainali (28)

Krishna Pathak (30)

Sahil Ratna Tuladhar (53)

Submitted to:

Mr. Nabin Ghimire

(Department of Computer Science and Engineering)

Submission date: 18th June 2024

Bona fide Certificate

This project work on

"ReelFeel"

is the bona fide work of

Samwarta Chitrakar (06)

Sujan Mainali (28)

Krishna Pathak (30)

Sahil Ratna Tuladhar (53)

who carried out the project work under my supervision.

Project Supervisor

_____

Dr. Rajani Chulyadyo

Department of Computer Science and Engineering

Date: 18th June 2024

Acknowledgement

We would like to express our sincere gratitude to everyone who contributed to the successful completion of our project, “Reelfeel”.

First and foremost, we are extremely grateful to Dr. Rajani Chulyadyo, our project supervisorfrom the Department of Computer Science and Engineering at Kathmandu University. Her continuous support, guidance, and invaluable insights have been instrumented in shaping thisproject.

We also extend our thanks to the Department of Computer Science to give us this opportunity to showcase our work, and providing us the necessary resources to realize this project.

Finally, we are grateful for our friends who contributed ideas and perspectives that enriched the project.

Thank you everyone for shaping this project and enhancing our learning experience.

Team Reelfeel
Samwarta Chitrakar (06)
Sujan Mainali (28)
Krishna Pathak (30)
Sahil Ratna Tuladhar (53)
CS 2020

Abstract

Reelfeel explores the development of a sentiment analysis program designed to analyze movie reviews and gauge audience reception. This project leverages natural language processing techniques to identify positive and negative sentiment within textual reviews. By pre-processing the review data, including tokenization, lemmatizing, and stop-word removal, the program extracts key features that influence sentiment.

We explore the application of machine learning algorithms to train the program on a large corpus of labeled movie reviews. The trained program can then be employed to analyze unlabeled reviews, generating sentiment scores that reflect the overall positive or negative reception of a movie. This project will be developed using Python which will be used for the backend and React which will be used for the frontend development.

Keywords: Python, React, Sentiment analysis

Acronyms/Abbreviation

OS	Operating System
HTML	Hyper Text Markup Language
CSS	Cascaded Style Sheets
ML	Machine Learning
BoW	Bag-of-Words
TF-IDF	Term Frequency-Inverse Document Frequency
TF	Term Frequency
IDF	Inverse Document Frequency
NLP	Natural Language Processing
GloVe	Global Vectors for Word Representation

List of Figures

Table of Contents

CHAPTER 1: Introduction

1.1 Background

Sentiment analysis, a branch of natural language processing, has emerged as a valuable tool for understanding public opinion and emotional responses conveyed in textual data. With the exponential growth of online platforms and social media, the volume of user-generated content, including movie reviews, has surged, presenting a rich source of data for sentiment analysis. Movie reviews encapsulate viewers' subjective impressions, ranging from praise to criticism, thereby offering insights into audience preferences and perceptions.

The advent of sentiment analysis has facilitated the extraction of sentiment polarity from textual data, enabling automated categorization of opinions as positive, negative, or neutral. This technique has found widespread applications across various domains, including marketing, product development, and customer feedback analysis. In the realm of film criticism, sentiment analysis holds promise for gauging audience reactions, predicting box office success, and informing content creators about the strengths and weaknesses of their productions.

1.2 Objectives

- Develop a sentiment analysis model for classifying movie reviews into positive or negative sentiments.
- Assess machine learning algorithms and sentiment lexicons to improve sentiment analysis accuracy for movie reviews.
- Explore how sentiment analysis can inform marketing strategies and content creation in the entertainment industry.
- Evaluate sentiment analysis' potential to democratize audience feedback for stakeholders in the film industry.

1.3 Motivation and Significance

The motivation behind this project stems from the growing influence of online platforms in shaping consumer perceptions and behavior, particularly in the realm of entertainment. As the proliferation of digital media platforms facilitates the dissemination of movie reviews and opinions, there arises a need to harness this wealth of data to extract actionable insights for stakeholders in the film industry.

The significance of this endeavor lies in its potential to democratize audience feedback and empower filmmakers, distributors, and marketers with data-driven insights into audience preferences and sentiment dynamics. By systematically analyzing movie reviews, this project seeks to bridge the gap between subjective viewer opinions and objective evaluative criteria, thereby facilitating more informed decision-making processes within the film industry ecosystem.

Furthermore, the project's outcomes hold relevance for academia, offering a case study in the application of sentiment analysis techniques to cultural artifacts and creative expressions. By shedding light on the intersection of technology and culture, this research contributes to the broader discourse on the role of computational methods in understanding human emotions and cultural phenomena. In the end, this study aims to deepen our comprehension of the intricate relationship between art, commerce, and audience reception in the digital era by clarifying the subtle aspects of audience feeling in movie reviews.

1.4 Techniques Utilized

- **Data Visualization Techniques**

In this project we have used different techniques and methods to generate visualization of data which includes:

Matplotlib

Matplotlib is a versatile and widely-used plotting library in Python that provides a comprehensive set of tools for creating static, animated, and interactive visualizations. In a sentiment analysis project, Matplotlib is invaluable for creating detailed and customizable

visual representations of data.

Seaborn

Seaborn is a powerful Python visualization library built on top of Matplotlib, designed to make it easier to create aesthetically pleasing and informative statistical graphics. It excels in providing high-level abstractions for drawing attractive plots, which are particularly useful in a sentiment analysis project.

- **Data Preprocessing Techniques**

To make sure that the data is ready to be processed by the ML models we implement various pre-processing techniques.

Lemmatization

Lemmatization is the process of grouping together different inflected forms of the same word. It's used in computational linguistics, natural language processing (NLP) and chatbots. (Gillis, 2023) Lemmatization links similar meaning words as one word, making tools such as chatbots and search engine queries more effective and accurate. (Gillis, 2023)

How Lemmatization Works:

Part-of-Speech Tagging: Lemmatization starts by identifying the part of speech (POS) of each word in a sentence, such as nouns, verbs, adjectives, etc. This is important because the lemma of a word depends on its role in the sentence. For instance, the lemma of "better" is "good" when it's an adjective, but as a verb, it remains "better."

Morphological Analysis: Using linguistic rules and morphological analysis, lemmatization examines the word's structure and its inflections. It analyzes the word to find its root form based on its part of speech.

Dictionary Lookup: Lemmatization often use a dictionary of words and their lemmas. The word is matched against this dictionary to find its base form. For example, "running" would be matched to "run," "geese" to "goose," and "mice" to "mouse."

- **Feature Extraction Techniques**

Feature Extraction techniques involves transforming raw text data into numerical features that can be used by machine learning algorithms

Bag-of-words

The Bag of Words (BoW) model is a fundamental technique in natural language processing (NLP) and text mining. It represents text data as a collection (or "bag") of words, disregarding grammar and word order but keeping multiplicity. A document is represented as a sparse vector where each dimension corresponds to a unique word in the corpus, and the value in each dimension represents the frequency of that word in the document.

TF-IDF (Term Frequency-Inverse Document Frequency)

TF-IDF (Term Frequency-Inverse Document Frequency) is an extension of the BoW model that addresses some of its limitations. TF-IDF reflects the importance of a word in a document relative to a collection of documents (corpus). It consists of two main components:

Term Frequency (TF): Measures how frequently a term occurs in a document. It is calculated as the ratio of the number of occurrences of a word to the total number of words in the document.

Inverse Document Frequency (IDF): Measures the rarity of a term across the entire corpus. It is calculated as the logarithm of the ratio of the total number of documents to the number of documents containing the term.

$$\mathrm{TF} = \frac{\text{number of times the term appears in the document}}{\text{total number of terms in the document}}$$

$$IDF = \log\left(\frac{number\ of\ documents\ in\ the\ corpus}{number\ of\ documents\ in\ the\ corpus\ containing\ the\ term}\right)$$

$$TF - IDF = TF * IDF$$

GloVe Embedding (Global Vectors for Word Representation)

GloVe (Global Vectors for Word Representation) is a popular unsupervised learning algorithm for obtaining dense vector representations (embeddings) of words. Unlike BoW and TF-IDF, which represent words as sparse vectors, GloVe embeddings capture semantic relationships between words based on their co-occurrence statistics in a corpus. GloVe achieves this by learning word embeddings using a global word-word co-occurrence matrix from the corpus, which encodes how frequently pairs of words co-occur across all documents. These embeddings are learned such that the dot product of two-word vectors indicates their co-occurrence probability.

GloVe embeddings offer several advantages:

Semantic Meaning: They capture semantic relationships between words, making them suitable for tasks requiring understanding of word similarity and context.

Dimensionality Reduction: They represent words in a continuous vector space of lower dimensionality compared to the high-dimensional sparse vectors of BoW and TF-IDF.

Pre-trained Models: Pre-trained GloVe embeddings are available for various languages and domains, allowing for transfer learning and easy integration into NLP applications without the need for extensive data preprocessing or training from scratch.

CHAPTER 2: Related Works

Sentiment analysis, a crucial aspect of Natural Language Processing (NLP), has garnered significant attention due to its practical applications across various domains. In this section, we review related works and projects that leverage sentiment analysis techniques to extract valuable insights from textual data, focusing on product reviews, Twitter feeds, and stock prices.

2.1 Amazon Product Reviews Analysis:

Analyzing product reviews is a fundamental task in sentiment analysis. Researchers use datasets from well-known online platforms like e-commerce websites to understand customer sentiments. For example, a study utilized Amazon review data to gauge customer satisfaction and identify areas for product improvement. By applying sentiment classification techniques, reviews are categorized as positive, negative, or neutral, offering insights into customer opinions about different products and brands. (Team, 2024)

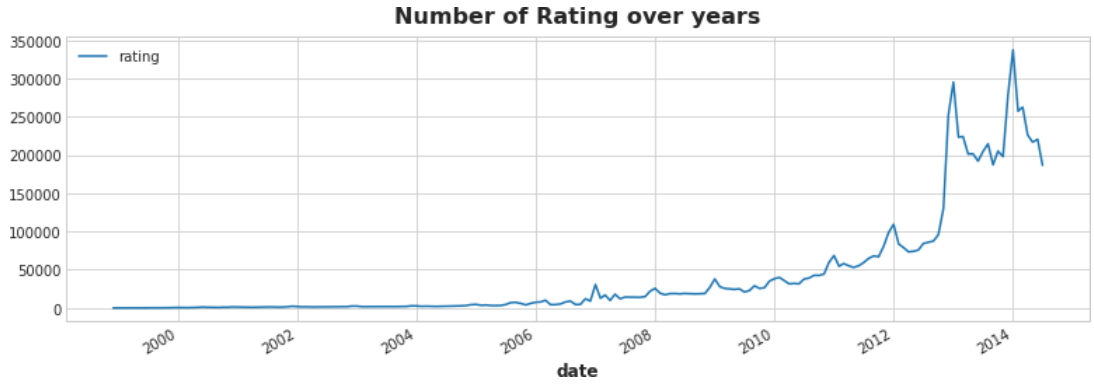

Fig 2.2.1: Amazon product Sentiment Analysis (Team, 2024)

2.2 Twitter Feed Analysis:

Twitter is a rich source of user-generated content, offering ample opportunities for sentiment analysis. Researchers have examined tweets on various topics using sentiment analysis methods. For instance, a study focused on tweets from airline travelers to evaluate sentiments towards major airlines, helping to assess customer experiences and airline reputations. Similarly, another study investigated customer help requests on Twitter for retail brands, identifying common issues and sentiment trends. By employing machine learning algorithms, tweets are categorized into positive, negative, or neutral sentiments, aiding in understanding customer concerns and improving service quality. (Team, 2024)

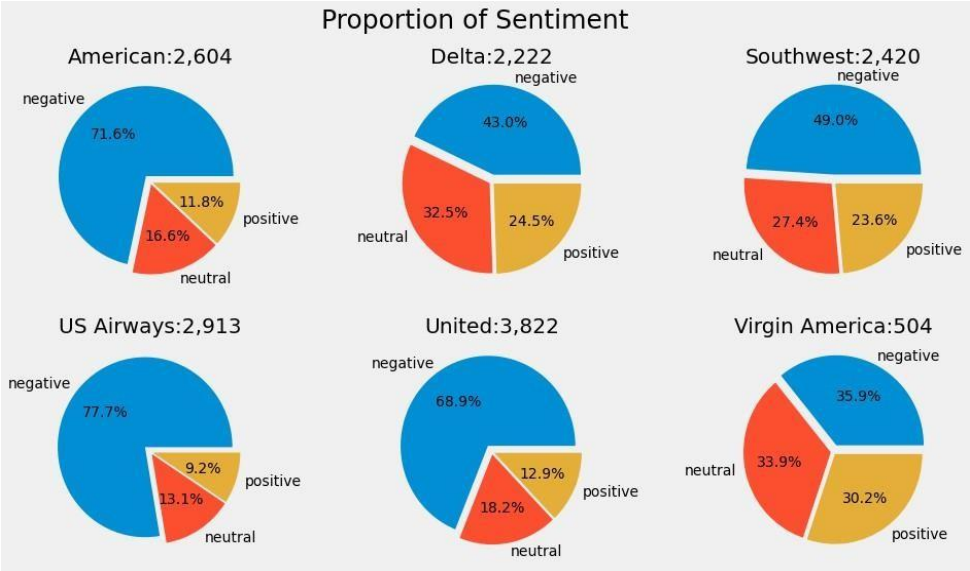

Fig 2.2.1: Twitter Sentiment Analysis (Team, 2024)

2.3 Stock Prices and Sentiment Analysis:

In addition to product reviews and Twitter feeds, sentiment analysis plays a crucial role in assessing information in financial markets. Platforms like Twitter generate thousands of pieces of investor sentiment every second, providing valuable insights into market dynamics. By analyzing sentiments related to listed companies, data scientists can predict stock price movements, as stock prices often track with investor sentiment. This integration of sentiment analysis with stock market data offers opportunities for making informed investment decisions based on market sentiment trends. These studies showcase the diverse applications and

methodologies of sentiment analysis, emphasizing its importance in extracting meaningful insights from textual data in e-commerce, social media, and financial market. (Team, 2024)

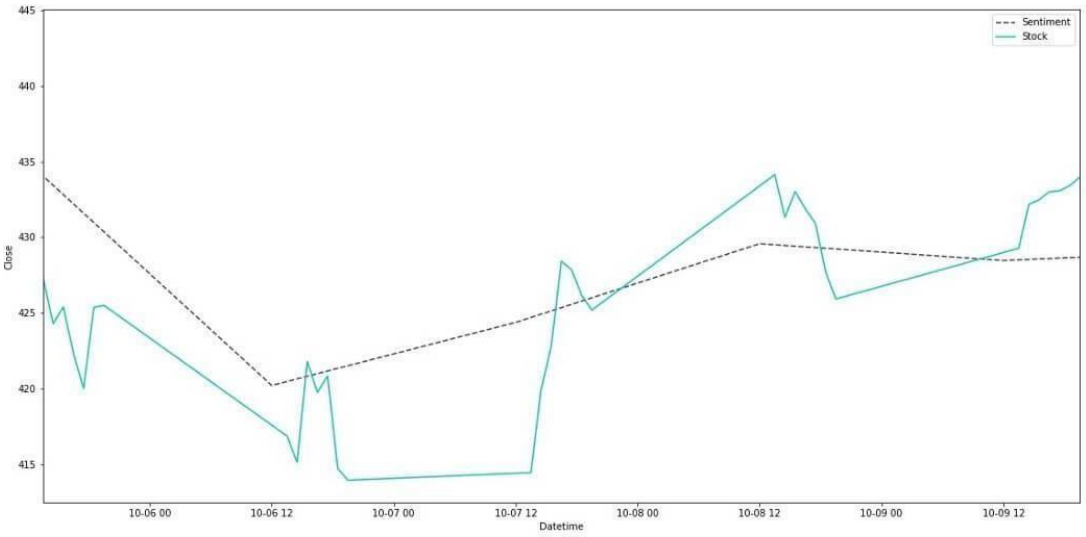

Fig 2.3.1: Stock Market Sentiment Analysis (Team, 2024)

Chapter 3: Methodology

3.1. Literature Review:

3.1.1. Introduction

Sentiment analysis, a critical subfield of Natural Language Processing (NLP), involves the extraction of subjective information from text data. Its applications range from understanding customer feedback to gauging public opinion on social media platforms. This literature review explores the evolution, core techniques, applications, challenges, recent advancements, and future directions in sentiment analysis.

3.1.2. Historical Background

With the advent of the internet era, sentiment analysis algorithms started to change. Early efforts were mostly concerned with "bag-of-words" models and were quite basic. These models, which lacked context and subtlety, distinguished between positive and negative terms. (Archive Technologies, 2023)

A big advancement was signaled by the development of machine learning in the latter half of the 20th century. Sentiment analysis by algorithms started to consider linguistic nuances, word placement, and context. This innovative move made it possible to comprehend text analysis moods considerably better. (Archive Technologies, 2023)

The amount of data that was available for sentiment analysis increased dramatically in the 2000s with the rise of social media. As a result of learning from enormous datasets and changing with each encounter, algorithms grew increasingly complex. (Archive Technologies, 2023)

3.1.3. Applications of Sentiment Analysis

Product Reviews:

Sentiment analysis is extensively used in e-commerce to analyze customer reviews. Platforms like Amazon use sentiment analysis to gauge customer satisfaction and identify areas for product improvement. For example, analyzing sentiment in product reviews can help businesses understand consumer preferences and enhance their offerings.

Social Media:

Twitter and Facebook are rich sources of user-generated content, making them ideal for sentiment analysis. Analyzing tweets and posts can provide insights into public opinion, brand reputation, and crisis management. Studies have shown the effectiveness of sentiment analysis in monitoring social media for real-time feedback and trend analysis.

Financial Markets:

Sentiment analysis is used to predict stock prices and market trends by analyzing news articles, social media, and financial reports. By understanding the sentiment of investors and market participants, analysts can make more informed predictions about market movements.

Other Domains:

Beyond the mentioned areas, sentiment analysis finds applications in healthcare (analyzing patient feedback), politics (predicting election outcomes), and entertainment (movie and game reviews).

3.1.4. Recent Advancements and Trends

Transfer Learning

Pre-trained models like BERT, GPT, and their derivatives have revolutionized sentiment analysis by leveraging large-scale language models. These models, fine-tuned on specific tasks, have set new benchmarks in accuracy and efficiency. (Sachin, 2024)

Real-Time Sentiment Analysis

Advancements in real-time processing enable sentiment analysis to provide immediate insights, which is particularly useful for monitoring social media and live feedback systems. (Repustate Team, 2022)

Multimodal Sentiment Analysis

Integrating textual analysis with other data types, such as images and videos, offers a more comprehensive understanding of sentiment. This multimodal approach is gaining traction, especially in social media analysis.

3.2. Procedure:

For the completion of our project, we have followed the following procedures and methods:

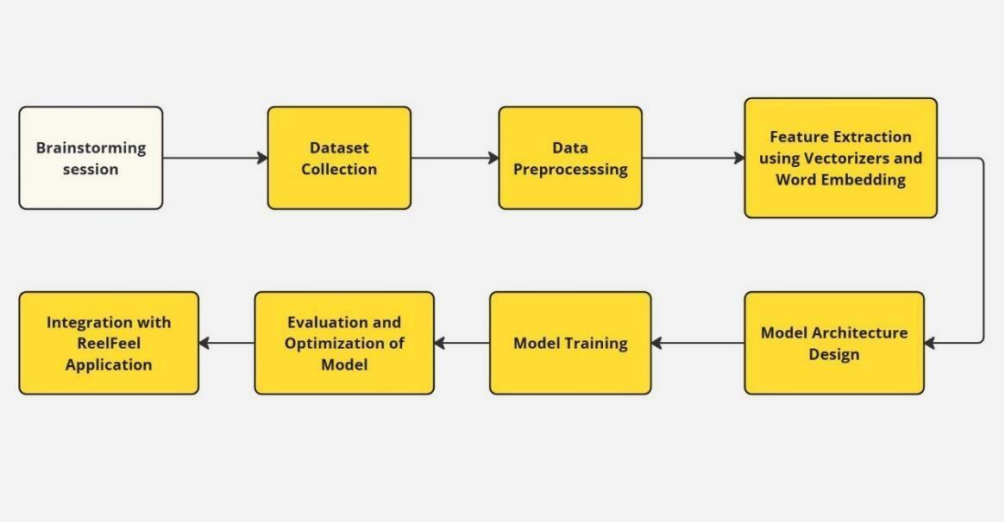

Fig 3.2.1: Development and Workflow Diagram

For every project, a particular list of procedures and methods needs to be followed to ensure efficiency and optimism. We decided to start our project by brainstorming ideas for our website. After selecting an idea, we collected datasets on which our analysis models were trained and tested. The content of the datasets was then preprocessed to make it ready for training on the model. We then used vectorizers and word embeddings to extract features from the preprocessed data. Next, the architecture of the model was designed and trained on the collected dataset. Further optimization and evaluation of the model were conducted to make it as accurate as possible. Finally, after training and testing the model and achieving the desired accuracy, the model was implemented by integrating it with the frontend of the application for a user-friendly experience.

3.3. Methodology:

3.3.1. Data Collection

This process involved gathering data relevant to training and testing a sentiment analysis model. We explored several supervised datasets, each containing labeled data, which we used to train the model. For both training and testing, we used a single dataset: the IMDB dataset of 50,000 movie reviews. This dataset included 50,000 labeled records, with each record consisting of two columns: one for the review and one for the sentiment label.

3.3.2. Data Preprocessing

The dataset collected was first stored in the MongoDB database to ensure efficient management of the data. The data was then retrieved from the database and a preprocessing function was developed to prepare the data for analysis. This process involved cleaning and preparing the data for analysis. The preprocessing function involved various steps: Removing Stop words, removing whitespaces, removing HTML tags, changing the text into lowercase, removing punctuations and performing lemmatization. This step is crucial as it removes unnecessary data from the dataset which do not contribute while training the model.

3.3.3. Feature Extraction using word embedding and Vectorizer

This step involved converting the data present in textual format into a numerical form so that

the machine learning models are able to understand the data. As most ML models deals with numerical data, this step is really important so that the data can be utilized to its full capacity to train the model. In this project, we have used two different approaches to vectorize the text and analyze which method was more accurate which includes: Vectorizing using Bag-of-Words and TF-IDF transformers and using GloVe word embedding

3.3.4. Model Architecture Design

This step involves choosing a ML model and implementing it. For our project, we have used a neural network model to perform the analysis. We have implemented three different models of neural network including: Simple Neural Networks, Convolutional Neural Network and Recurrent Neural Network. Each model was created by implementing various input, hidden and output layers of the model and stating optimizer and loss functions to evaluate the model. These models were then compared with each other based on their capabilities to capture within the text provided.

3.3.5. Model Training

In this step, we train the ML model that we have selected by fitting and evaluating the model using training data. The dataset is first divided into training and testing data by using a train_test_split function that allows the data to be divided as the user wishes. It involved optimizing the parameters through back propagation and adjusting the weights of the neural network to minimize the loss function.

3.3.6. Evaluation and Optimization of the Model

In this step, it involved careful assessment of the results obtained from training the model. The results were analyzed and the parameters of the models were adjusted iteratively to refine the model's performance. This step allowed us to improve the model's effectiveness in analyzing the sentiments in diverse movie reviews.

3.3.7. Integration with ReelFeel Application

In the final step of the project, we deployed the optimized and trained model to work on real movie reviews by integrating the model with the frontend and backend framework of the

application for better user-experience. This integration will equip ReelFeel with the ability to conduct thorough sentiment analysis of movie reviews.

3.4. Features of ReelFeel Application:

3.4.1. Movie Review Sentiment Analysis

The Reelfeel application includes a Movie Review Sentiment Analysis feature that displays the count of positive and negative reviews for any movie. This allows users to quickly gauge overall sentiment at a glance, enhancing their movie selection process by providing a clear visual summary of viewer opinions.

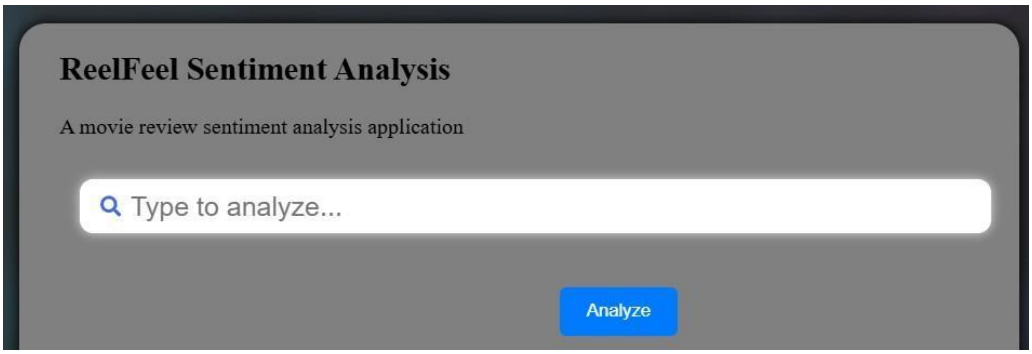

Fig 3.4.1.1: Input section for movie to be analyzed

Fig 3.4.1.2: Number of positive and negative reviews

3.4.2. Emotion Analysis

Reelfeel's Emotion Analysis feature visualizes the emotional tone of movie reviews. It presents a graph displaying the percentages of various emotions: anger, disgust, fear, joy, and surprise—within each review. This detailed breakdown helps users understand the nuanced emotional responses a movie evokes, offering deeper insight beyond simple positive or negative sentiment.

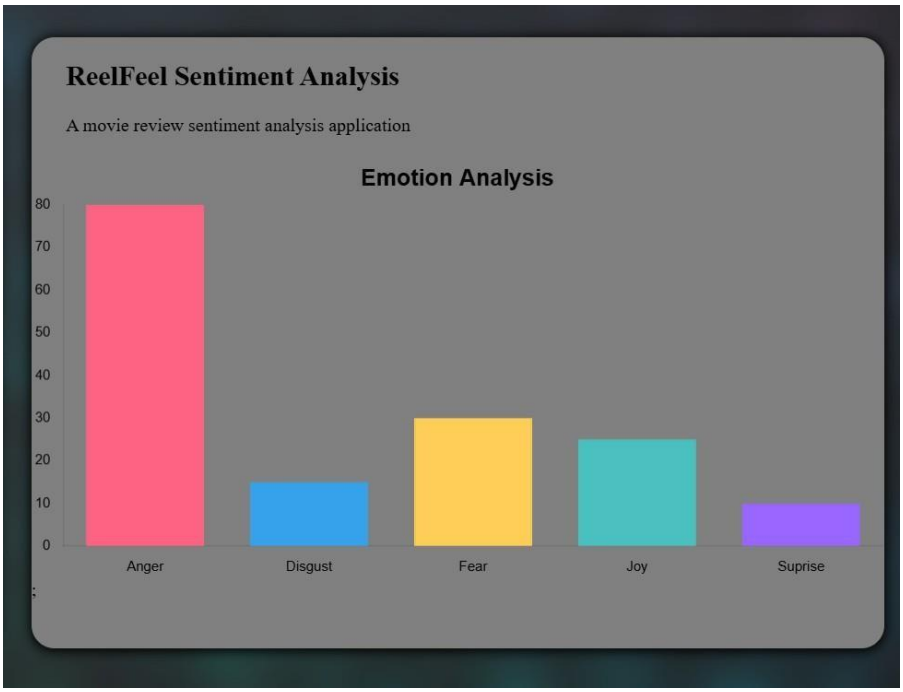

Fig 3.4.2.1: Bar graph for Emotion Analysis

3.4.3. Word Cloud

Reelfeel's Word Cloud feature highlights the most frequently used words in the review dataset. This visual representation emphasizes common terms, with more prominent words appearing larger. It helps users quickly identify recurring themes and keywords in the reviews, providing an at-a-glance understanding of the main points and topics discussed by reviewers.

Fig 3.4.3.1: Word Cloud Page

3.4.4. Sentiment Meter

Reelfeel's Sentiment Meter feature displays a progression chart indicating the percentage of positive and negative emotions in reviews. This visual tool allows users to easily see the balance of sentiments, providing a clear and dynamic overview of how viewers feel about a movie. It aids in making informed viewing decisions based on the collective emotional response.

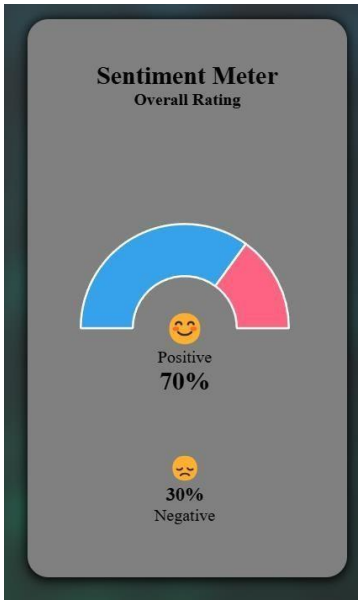

Fig 3.4.4.1: Sentiment Meter using a progression chart

3.4.5. Test Page

Reelfeel's Test Page allows users to input a movie review and instantly test its sentiment. The feature analyzes the review to determine if it is positive or negative and displays the results on a progression chart. This interactive tool provides immediate feedback, showing the percentage breakdown of the sentiment, helping users understand the emotional tone of their review.

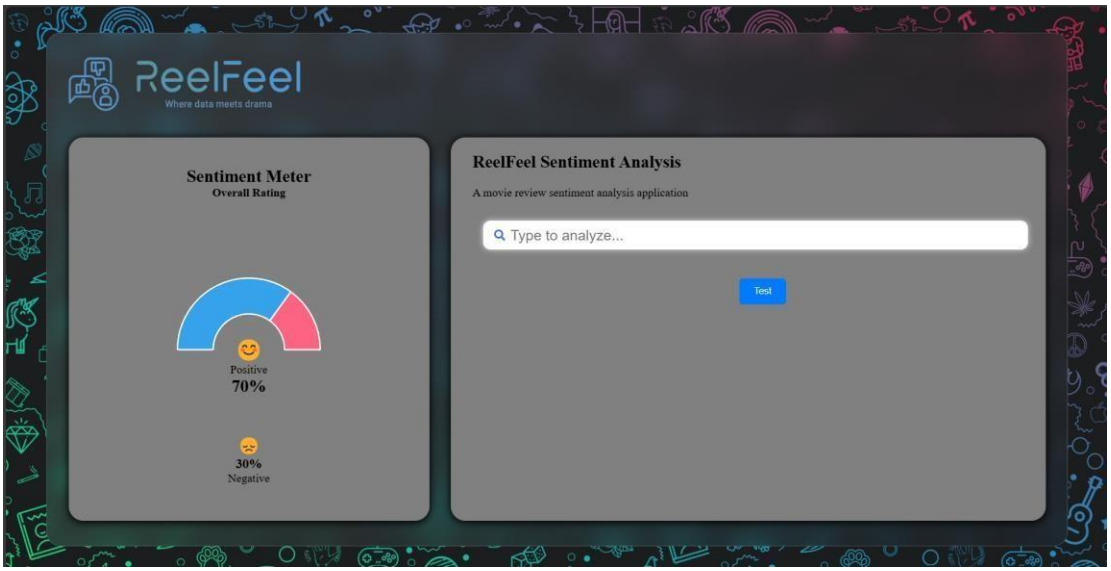

Fig 3.4.5.1: Test Page

3.5 Use-case Diagram:

The use case diagram illustrates the interactions between the Admin, Tester, Developer, and the System for managing datasets and machine learning models. The Admin is responsible for gathering known data and responses, preparing datasets, deleting datasets, and selecting appropriate algorithms and validation methods. The Tester interacts with the system to examine it and make necessary updates until the results are satisfactory. The Developer's role involves using the refined model with new data to make predictions. This structured approach ensures a collaborative effort in building, validating, and deploying machine learning models within the system.

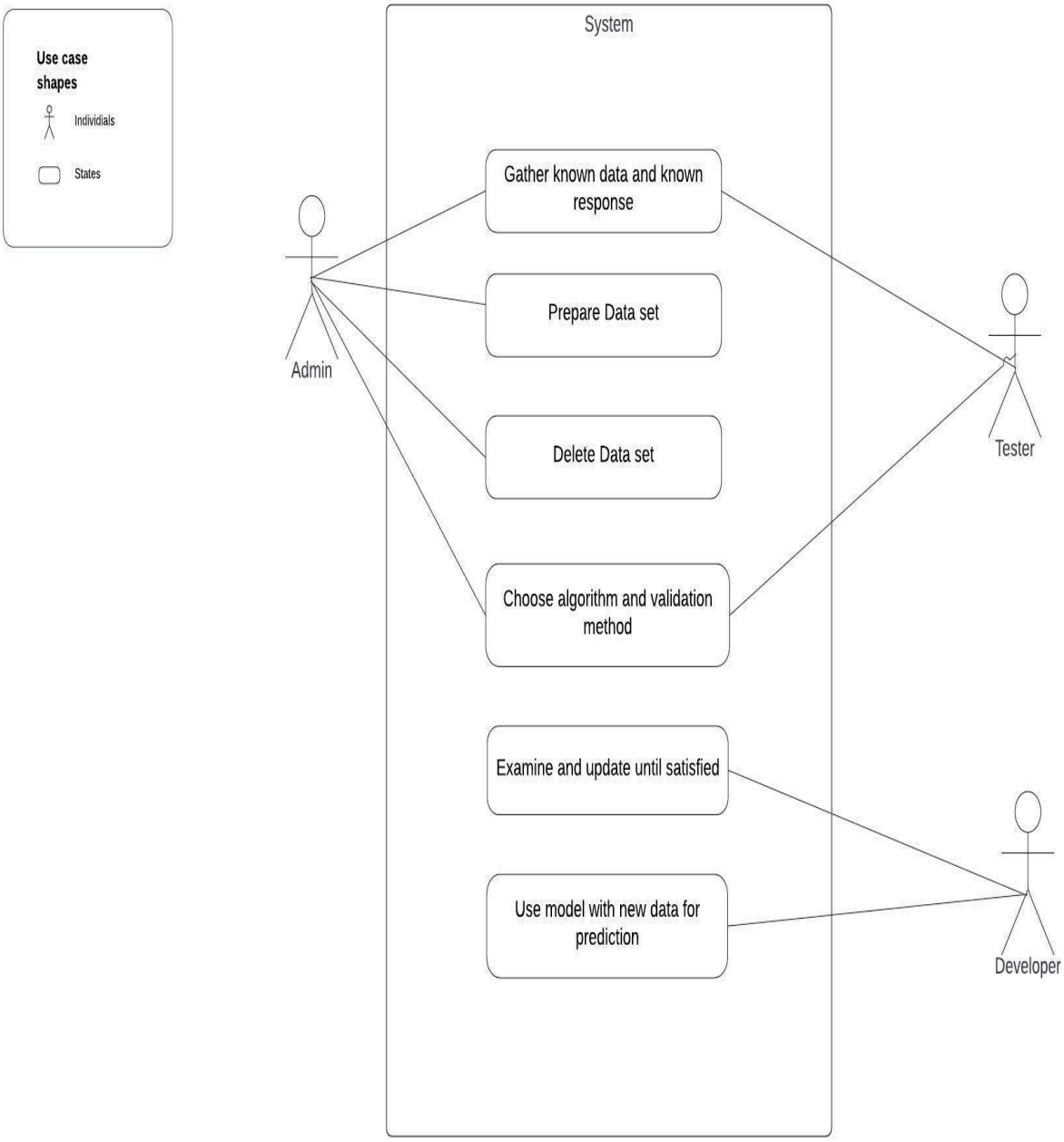

Fig 3.5.1: Use-case Diagram

Chapter 4: Design Implementation and Achievements

4.1. Dataset:

The dataset for my sentiment analysis project is the IMDB Dataset of 50K Movie Reviews which we obtained from Kaggle. This particular dataset consists of 50,000 movie reviews from the Internet Movie Database (IMDB), where each review is labeled as either positive or negative. So, this dataset is considered as a supervised dataset. It contains equal number of positive and negative reviews. The initial Database is first stored in MongoDB database, and it is then retrieved as per the requirement for preprocessing, training and testing. It encompasses a range of topics and writing styles, enabling the model to generalize well across different domains and contexts

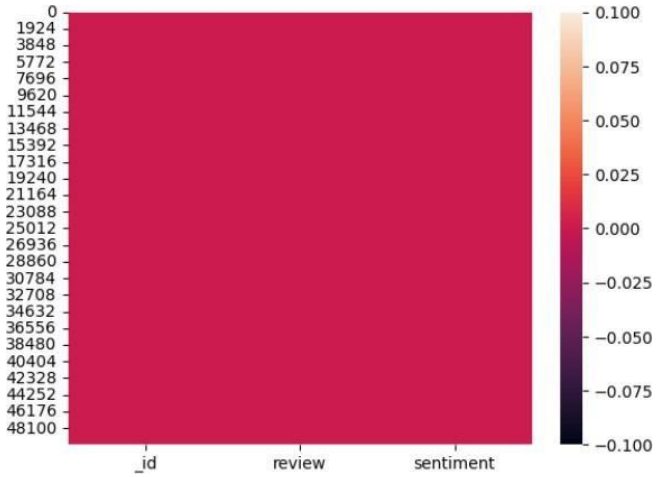

Fig 4.1.1: Heat Map of Missing Data

This Figure shows that there were no null or empty value present for any of the columns present in the dataset.

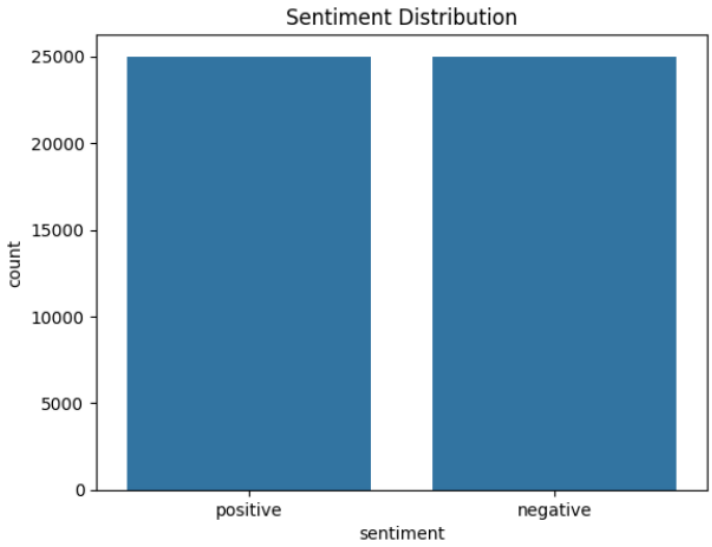

Fig 4.1.2: Bar plot of different Sentiments

This Figure shows that there are equal number of positive and negative sentiment reviews in the dataset.

4.2. Data Preprocessing:

It is an important step where the original raw data is taken as an input and is converted into a form that is suitable for analysis for the models. The primary aim of this step is to remove unnecessary data present in the dataset which results in wastage of resources. In this project, we have created our own text preprocessing function which is then applied to the raw dataset which involves:

- **Lowercasing**: Converting all text into lowercase to maintain uniformity
- **Removing Punctuation and Special Characters**: Removing punctuations and special characters that do not contribute to determine sentiment
- **Removal of Stop words**: Removing words that do not carry significant meaning for sentiment
- **Lemmatization**: Reducing words to their base word or root form using knowledge regarding English language, so that different words having similar meaning are not treated separately

- **Removing HMTL Tags and URLs**: Stripping out any HTML tags and URLs that may be present in the text

The preprocessed data is then stored in the MongoDB database, so that we do not need to preprocess the data every time we run the program

4.3. Data Vectorization:

Data vectorization refers to the process of converting textual data into numerical vectors which can be understood and dealt with by the machine learning models. In this project we have implemented two different methods of Data Vectorization:

- **Bag of words (BoW) and TF-IDF vectorization**:

The first method involves representing the entire corpus of data using a bag of words approach. This bag of words is then converted into a sparse matrix representation using a TF-IDF transformer. In this representation, the sparse matrix consists of rows representing the reviews and columns representing the entire vocabulary of the corpus. The values in the cells indicate the importance of the word in relation to the review and the entire corpus.

- **GloVe word Embeddings**:

The second method involved extracting features from the text by using a gloVe word embedding which also created a sparse matrix where rows represented the vocabulary of the corpus and the columns represented the dimensions using which the words are represented. Based on these dimension values, the relation and co-occurrence of the words were determined.

We applied BoW and TF-IDF in a model where a single input is taken as an input and then it is classified as positive or negative.

GloVe word embedding was used in a model where an entire dataset is taken as an input and then it classifies each review as positive or negative and then provides a sentiment score for the dataset. It has been implemented in three different neural network model: Simple Neural Network, Convolutional Neural Network and Recurrent Neural Network with LSTM.

4.4. Model Development:

It is the step the involves building, training, and evaluating ML models to accurately predict the sentiment of the textual data. In this project, the data was divided into training and testing sets by using the train_test_split. We have implemented a train-test split of 80% - 20% respectively. Three different models were developed using different layers of the neural network. Each neural network basically consisted of Input Layer, Hidden Layers and the Output Layer. The model was then compiled and evaluated using the Adam optimizer, binary cross-entropy loss function and the accuracy as an evaluation metric.

While creating a model using GloVe embedding, we used an Embedding layer that is used to assign the embedding matrix generated as weights in the network and in the SNN created using TF-IDF vectorizer, concepts of Principal Component Analysis (PCA) has been implemented to reduce the dimensionality of features.

4.5. Discussion on Achievements:

In this project, we have used 3 different versions of the neural network model to analyze its accuracy and capabilities among each other. The models include: Simple Neural Networks, Convolutional Neural Network and Recurrent Neural Network.

The charts below represent the accuracy and loss of the different architecture throughout the training process.

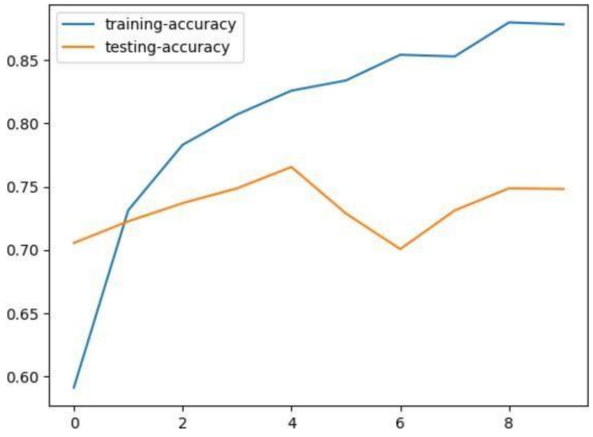

Fig 4.5.1: SNN model Accuracy

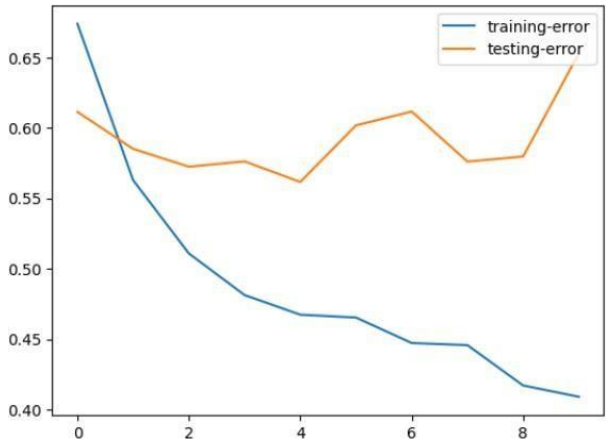

Fig 4.5.2: SNN model Loss

Here, we can observe that overfitting of the SNN model takes place as there is a big gap between the training and testing accuracies.

Fig 4.5.3: CNN model Accuracy

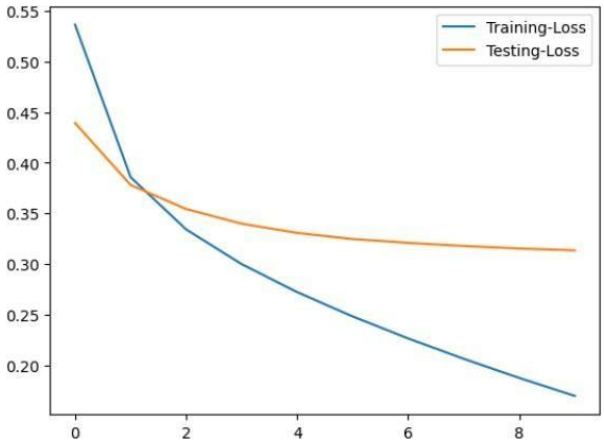

Fig 4.5.4: CNN model Loss

Here, we observe that the CNN model is less over fit in comparison to SNN network as the training and testing accuracies are a bit closer.

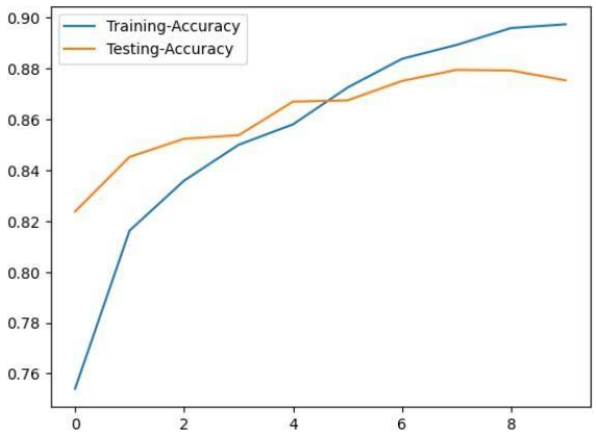

Fig 4.5.5: RNN model Accuracy

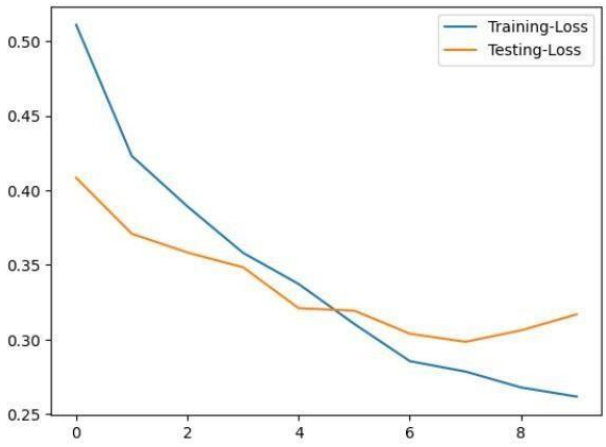

Fig 4.5.6: RNN model Loss

Here, we observe that in the RNN with LSTM model, overfitting of training the model is least as the training and testing accuracies are almost similar.

Model Architecture	Accuracy
Simple Neural Network	75%
Convolutional Neural Network (CNN)	86%
Recurrent Neural Network (RNN) with LSTM	87%

The comparison of the three sentiment analysis model architectures on IMDB movie data set revealed varying accuracies. The simple neural network achieved can accuracy of 75% while having a loss of 60%, while the convolutional neural network (CNN) improved significantly with an accuracy of 86% and loss of 31%. The Recurrent neural network (RNN) with Long Short-Term Memory (LSTM) improved slightly than CNN and outperformed both the other architectures with an accuracy of 87% and a loss of 31%.

These results highlight the importance of architecture selection, stating the RNN-LSTM model's effectiveness in capturing the relationships expressed among words in the movie reviews.

4.6 System requirement specification

4.6.1 Software Specification

React's component-based architecture promotes code reusability, making it efficient for building complex user interfaces. Its virtual DOM (Document Object Model) optimizes rendering performance, ensuring that only the necessary parts of a webpage are updated, resulting in a responsive and fast user experience. React's extensive ecosystem, including tools like React Router for routing and Redux for state management, simplifies the development process and enhances maintainability. Additionally, its strong community support ensures access to a wealth of resources, libraries, and documentation. With Facebook maintaining React and its use by prominent companies like Airbnb and Netflix, it's a reliable and future-proof choice for building modern, interactive web applications.

4.6.1.1 Front End Tools: React

React is a popular JavaScript library for building user interfaces. (React, n.d.) Developed and maintained by Facebook, react has gained widespread adoption in web development due to its component-based architecture and efficient virtual DOM rendering. (Meta, n.d.) React allows developers to create reusable UI components that can be composed to build complex user interfaces. Its declarative syntax and one-way data flow make it easier to reason about and maintain large-scale applications. (React, n.d.) Additionally, React's vibrant ecosystem, including tools like Redux for state management and React Router for routing, makes it a versatile choice for building interactive and responsive web applications. (Abramov, n.d.)

4.6.1.2 Back End Tool: Python

Python is a versatile and widely-used programming language known for its simplicity and readability. Created in the late 1980s, Python has since gained immense popularity due to its ease of learning and its extensive library support. (Guido Van Rossum, 2009) Its clean and concise syntax makes it an ideal choice for both beginners and experienced developers, enabling them to write efficient code for a wide range of applications, from web development and data analysis to artificial intelligence and scientific research. (Lutz, 2013) Its simplicity and readability make it an excellent choice for developing various machine learning related

applications. (Lutz, 2013) Popular python libraries such as NLTK (natural language toolkit) provides tools for text processing, tokenization, and basic sentiment analysis with VADER. (Steven Bird, 2009) Scikit-learn offers various machine learning algorithms for classification, including sentiment analysis. (Fabian Pedregosa, 2011) Python's active developer community also ensures that it stays up-to-date with the evolving machine language landscape, making it an essential tool for those looking to innovate in the world of machine language technology.

4.6.2 Hardware Specification

The hardware specification required to develop this project is a 1.8 GHz or faster 64-bit processor; Quad-core or better recommended. ARM processors are not Supported. Minimum of 4 GB of RAM. Many factors impact resources used, we Recommend 16 GB RAM for typical professional solutions.
Windows 365: Minimum 2 vCPU and 8 GB RAM. 4 vCPU and 16 GB of RAM are recommended.
Hard disk space: Minimum of 850 MB upto 210 GB of available space, Depending on features installed installing Windows and Visual Studio on a Solid-state drive (SSD) to increase performance.
A video card that supports a minimum display resolution of WXGA (1366 by 768); Visual Studio will work best at a resolution of 1920 x 1080 or higher

Chapter 5: Conclusion and Recommendation

Leveraging the power of machine learning, this project categorizes user-typed sentences and reviews from a given CSV file, specifically focusing on movie reviews, into positive, negative, or neutral sentiments. It provides a platform to delve into the emotions expressed by users, offering valuable insights into their sentiments and opinions. This tool enables users to understand the underlying emotional tone of reviews, aiding in making more informed decisions.

The ability to analyze sentiments in both individual sentences and bulk data from CSV files broadens the scope of the project. By providing a deeper understanding of user sentiments, this project helps in identifying trends and making data-driven decisions. This project can be highly viable in the movie industry by providing insights into audience reactions

and reviews. Movie producers, distributors, and marketers can leverage this tool to gauge public opinion, identify trends in viewer feedback, and make informed decisions regarding marketing strategies, movie sequels, and more. By analyzing sentiments in movie reviews, industry stakeholders can better understand audience preferences and improve their content to meet viewer expectations.

5.1 Limitations

While the project achieves its primary objectives, several limitations must be acknowledged:

1. **Contextual Understanding:** The sentiment analysis may not accurately identify sarcasm or irony in movie reviews.
2. **Language and Cultural Nuances:** The system may struggle with understanding slang, idioms, and cultural nuances specific to movie reviews.
3. **Limited Sentiment Categories:** Classifying sentiments into only two categories (positive, negative) might oversimplify complex emotions.

5.2 Future Enhancement

To enhance the project's effectiveness and scope, several future enhancements are proposed:

1. **Granular Sentiment Levels:** Introduce more granular sentiment categories (e.g., very positive, slightly positive) to capture detailed emotions.
2. **Contextual Analysis:** Incorporate contextual analysis to better understand the background and intention behind the text.
3. **Sarcasm Detection:** Improve the model to better detect sarcasm and irony in sentences.

By addressing these limitations and incorporating these future enhancements, the project can significantly improve its accuracy, usability, and impact. This will provide a more comprehensive understanding of user sentiments and enable more sophisticated and actionable insights, making it a valuable tool for the movie industry.

CHAPTER 6: Project Planning and Scheduling

For our engineering project this semester we were allocated a time of about 12 weeks. The following shows the GANTT chart to show the time allocated for different tasks to be completed in our project. We started by dividing the project into individual tasks to increase our efficiency and utilize as much time as we can. We have allocated 9 weeks for coding purposes as it is the most important aspect of our project. In the end weeks, we aim to test and debug our website to check for errors and fix them within the same period before releasing it. The following shows the GANTT chart to show the time allocated for different tasks to be completed in our project.

Week	1	2	3	4	5	6	7	8	9	10
Planning and Research										
Data Collection and Preprocessing										
Model Selection and Training										
Model Evaluation and Testing										
Model Deployment and Documentation										

Figure 5.1: GANTT Chart

Index:

Task Completed:

References

Abramov, D. (n.d.). *Redux*. Retrieved from https://redux.js.org

Archive Technologies. (2023, October 5). *Sentiment Analysis Explained: From Theory to Real-World Applications*. Retrieved from archive: https://archive.com/blog/sentiment-analysis

Fabian Pedregosa, G. V. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning*, 2825-2830.

Gillis, A. S. (2023, March). *DEFINITION*. Retrieved from TechTarget: https://www.techtarget.com/searchenterpriseai/definition/lemmatization

Guido Van Rossum, F. L. (2009). *Python 3 Reference Manual.* CreateSpace, 100 Enterprise Way, Suite A200, Scotts Valley, CA.

Lutz, M. (2013). *Learning Python.* O'Reilly Media, Inc.

Meta. (n.d.). *Meta OpenSource*. Retrieved from React: https://opensource.fb.com/projects/react/

Meta Platforms. (2023). *Tutorial: Intro to React*. Retrieved from React - A JavaScript Library for building user interfaces: https://legacy.reactjs.org/tutorial/tutorial.html

React. (n.d.). *React - A JavaScript library for building user interfaces*. Retrieved from React: https://reactjs.org/

Repustate Team. (2022, July 20). *How To Use Real-Time Sentiment Analysis For Live Social Feeds*. Retrieved from Repustate: https://www.repustate.com/blog/real-time-sentiment-analysis

Sachin, S. (2024, March 11). *Transfer Learning for NLP: Adapting Pre-trained Language Models to New Tasks*. Retrieved from Linkedin: https://www.linkedin.com/pulse/transfer-learning-nlp-adapting-pre-trained-language-models-sachin-tcqfc

Steven Bird, E. K. (2009). *Natural Language Processing with Python – Analyzing Text with the Natural Language Toolkit.* O'Reilly Media, Inc.

Team, I. (2024, June 3). *Top 10 Sentiment Analysis Project Ideas and Datasets*. Retrieved from Interview Query: https://www.interviewquery.com/p/sentiment-analysis-projects-and-datasets