Akshay Raut (33262)

Kunal Sherkar (33268)

Yash Sonavane (33272)

ML MINI PROJECT

"Sentiment Analysis of Movie Reviews"

PROBLEM STATEMENT/DEFINITION:

The problem at hand is to develop a **sentiment analysis model for IMDb movie reviews**. The primary goal of this project is to design and implement a machine learning model that can automatically classify these reviews into sentiment categories, specifically as positive, negative.

OBJECTIVE:

1. **Data Collection**: Gather a substantial dataset of IMDb movie reviews and their corresponding sentiment labels to create a reliable training and testing dataset.
2. **Data Preprocessing:** Clean and preprocess the textual data to remove noise, special characters, and irrelevant information, making it suitable for analysis.
3. **Model Training:** Train the selected model on the labeled data to learn the relationships between textual features and sentiment labels.
4. **Sentiment Classification:** Develop a system that can automatically classify IMDb movie reviews into sentiment categories: positive, negative, or neutral.
5. **Model Evaluation**: Assess the performance of the sentiment analysis model using metrics like accuracy, precision, recall, and F1 score. Ensure that the model provides reliable sentiment predictions.

ABSTRACT

The **sentiment analysis of IMDb movie reviews** is a compelling machine learning mini-project that leverages natural language processing techniques to classify the sentiment (positive or negative) of user-generated movie reviews. This project aims to build a model capable of automatically assessing the sentiment expressed in these reviews, providing valuable insights into audience reactions to films. The project workflow encompasses data collection, preprocessing, feature extraction, model development, and evaluation. By exploring this project, students can gain practical experience in text classification, and machine learning while also uncovering the sentiments hidden within the vast repository of IMDb movie reviews. This mini-project not only equips learners with valuable skills in data analysis and text mining but also allows them to gain insights into the world of movie reviews and their impact on the film industry.

ACKNOWLEDGEMENT

We are wholeheartedly acknowledging our deep gratitude to all those who have helped us build this mini project. We wholeheartedly thank **Mr. Vinit Tribhuvan Sir** for the guidance and helping us throughout the process and constantly motivating us to do better. With their guidance we were able to complete our mini project and understand it up to the mark. I'm also thankful to **Dr. A. S. Ghotkar, HOD of the IT Department**, for her invaluable advice and support throughout the mini project. We would like to thank the principal and everyone else who helped me write this report. This project represents a collective effort, and we thank each and every member of our team for their hard work and dedication. Together, we've achieved our goals and created a project.

1. INTRODUCTION

1.1. Purpose and Problem Statement:

Purpose: To explore the application of machine learning techniques in the domain of sentiment analysis.

Problem Statement: The problem at hand is to develop a sentiment analysis model for IMDb movie reviews. The primary goal of this project is to design and implement a machine learning model that can automatically classify these reviews into sentiment categories, specifically as positive, negative**.**

1.2. Scope and Objectives:

1.2.1. Scope:

Model Selection: A machine learning or deep learning model will be chosen for sentiment analysis. Various models will be evaluated to determine the most effective one for the task.

Model Training: The selected model will be trained on the labelled data to learn the relationships between textual features and sentiment labels. This step will involve fine-tuning the model for optimal performance.

1.2.2. Objectives:

Feature Engineering: Convert the textual data into numerical features, making it suitable for machine learning models.

Model Selection and Development: Choose and train a sentiment analysis model capable of accurately categorizing movie reviews into positive, negative, or neutral sentiments.

1.3. Definitions, Acronyms and Abbreviations

IMDb: Internet Movie Database
ML: Machine Learning
NLP: Natural Language Processing
TF-IDF: Term Frequency-Inverse Document Frequency
RNN: Recurrent Neural Networks
BERT: Bidirectional Encoder Representations from Transformers
SVM: Support Vector Machines
CSV: Comma-Separated Values
API: Application Programming Interface
HTML: HyperText Markup Language
NumPy: Numerical Python
Pandas: Python Data Analysis Library

NLTK: Natural Language Toolkit
Seaborn: Seaborn is a Python data visualization library based on Matplotlib that provides a high-level interface for creating informative and attractive statistical graphics.
re: Regular Expressions

1.4. References

1.4.1. Research Papers

- **Movies Reviews Sentiment Analysis and Classification:** https://www.kaggle.com/competitions/sentiment-analysis-on-movie-reviews
- **Sentiment Analysis of Twitter Messages for Predicting Movie Box Office:** https://norma.ncirl.ie/3418/1/kapardhikumarguda.pdf

1.4.2. Websites

- Internet Movie Database (IMDb): https://www.imdb.com/
- Rotten Tomatoes: https://www.rottentomatoes.com/
- MovieLens: https://movielens.org/
- Kaggle: https://www.kaggle.com/

2. Literature Survey

2.1. Introduction

The literature survey titled "Sentimental Analysis of Movie Reviews Using Machine Learning Algorithms" likely serves as a comprehensive exploration of sentiment analysis in the context of movie reviews. It may cover a range of methodologies and approaches to sentiment analysis, focusing on machine learning algorithms.

This literature survey delves into the field of sentiment analysis with a specific focus on movie reviews. It explores the methodologies, techniques, and algorithms used to extract sentiment and subjective information from textual data. With a significant emphasis on machine learning algorithms, we aim to understand how these methods are applied to the complex and often nuanced world of movie reviews.

2.2. Detailed Literature Survey

This literature survey delves into the field of sentiment analysis with a specific focus on movie reviews. It explores the methodologies, techniques, and algorithms used to extract sentiment and subjective information from textual data. With a significant emphasis on machine learning algorithms, we aim to understand how these methods are applied to the complex and often nuanced world of movie reviews.

Sentiment analysis, a subfield of natural language processing, poses unique challenges when applied to movie reviews. Not only does it require the accurate classification of text into sentiment categories (e.g., positive, negative, neutral), but it also necessitates the comprehension of context, sarcasm, and the influence of cultural factors on sentiment expression.

This survey draws on the research and findings of various academic studies and industry reports, providing an overview of key concepts, datasets, methodologies, and models. Additionally, it explores the limitations and areas of improvement within the domain of sentiment analysis for movie reviews.

By the end of this survey, we aim to have a deeper understanding of the state of the art in sentiment analysis within the film industry, offering valuable insights for future research and applications in the field. Through a comprehensive analysis of existing literature, we seek to shed light on the evolution of sentiment analysis techniques, challenges faced, and the potential for innovation and refinement in this critical area of machine learning.

2.3. Findings of Literature Survey

Data Sources and Datasets:
Researchers frequently leverage IMDb and similar movie review platforms as valuable sources of data for sentiment analysis.

The size and diversity of movie review datasets significantly impact the performance of sentiment analysis models, with larger and more diverse datasets often leading to improved results.

Text Preprocessing and Feature Engineering:
Text preprocessing steps, such as stopword removal, stemming, and the handling of special characters, are essential for improving the quality of textual data.

Feature engineering techniques, such as TF-IDF and word embeddings, play a critical role in transforming text data into numerical representations suitable for machine learning models.

Machine Learning Algorithms:
Various machine learning algorithms are applied to sentiment analysis, including support vector machines, logistic regression, and Naïve Bayes, all of which have been tested for their effectiveness.

Deep learning models, particularly recurrent neural networks (RNNs) and transformer-based models like BERT, have gained popularity for their ability to capture complex relationships within text data.

Model Evaluation and Metrics:
Researchers use a variety of evaluation metrics, such as accuracy, precision, recall, and F1 score, to assess the performance of sentiment analysis models.

The choice of evaluation metric depends on the specific goals of the sentiment analysis task, with a focus on minimizing false positives or false negatives, depending on the application.

3. System Architecture and Design

3.1. Detailed Architecture

Our system is designed to predict house prices based on the area using a linear regression model. The architecture of the system can be broken down into the following components:

- Data Collection: We obtain a dataset containing information on house prices and corresponding areas.
- Data Pre-processing: In this phase, we clean and pre-process the data. This involves handling missing values, outlier detection, and data normalization.
- Model Building: We create a linear regression model that can learn the relationship between house prices and area.
- Model Training: We train the model on the pre-processed data to optimize its parameters.
- Model Evaluation: We evaluate the model's performance using various metrics.

3.2 Dataset Description

IMDB dataset having 50,000 movie reviews for natural language processing or Text analytics.

This is a dataset for binary sentiment classification containing substantially more data than previous benchmark datasets. We provide a set of 25,000 highly polar movie reviews for training and 25,000 for testing. So, predict the number of positive and negative reviews using either classification or deep learning algorithms.

3.3 Detailed Phases

1. Data Collection:

- **Data Sources:** IMDb movie reviews and associated sentiment labels.
- **Data Retrieval:** Accessing the IMDb dataset, either through APIs or web scraping.
- **Data Storage:** Store the collected data in a structured format, such as a database or CSV files.

2. Data Preprocessing:

- **Text Cleaning:** Remove special characters, punctuation, and formatting issues.
- **Tokenization:** Split text into individual words or tokens.
- **Stop word Removal:** Eliminate common words that do not carry much sentiment information.
- **Stemming or Lemmatization**: Normalize words to their root form.
- **Data Split:** Divide the dataset into training, validation, and test sets.

3. Text Representation:

- **Feature Extraction:** Convert text data into numerical representations.
- **Vectorization Techniques**: Utilize techniques such as TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings (Word2Vec, GloVe) to create feature vectors.

4. Model Development:

- **Model Selection:** Choose a machine learning or deep learning algorithm for sentiment analysis. Common choices include logistic regression, support vector machines, recurrent neural networks (RNNs), or transformer-based models.
- **Model Training:** Train the selected model using the training dataset. Fine-tune hyperparameters for optimal performance.

5. Sentiment Analysis:

- **Sentiment Classification:** Deploy the trained model to classify movie reviews into sentiment categories (e.g., positive and negative).
- **Real-time Analysis (Optional):** If needed, create an interface where users can input movie reviews, and the model provides real-time sentiment predictions.

6. Model Evaluation:

- **Performance Metrics:** Assess the model's performance using evaluation metrics like accuracy, precision, recall, F1 score, and confusion matrices.
- **Cross-Validation**: Perform cross-validation to ensure robustness and prevent overfitting.

3.4 Algorithms

In a "Sentiment Analysis of IMDb Movie Reviews" project, you can employ a variety of machine learning and deep learning algorithms to perform sentiment classification. The choice of algorithm depends on the complexity of the project, the size of the dataset, and the desired level of accuracy.

1. Naïve Bayes:

- **Algorithm Type:** Probabilistic
- **Description:** Naïve Bayes is a simple yet effective algorithm for text classification. It calculates the probability of a review belonging to a specific sentiment category based on the occurrence of words within the text.

2. Logistic Regression:

- **Algorithm Type:** Linear Classifier
- **Description:** Logistic Regression is a straightforward binary classification algorithm that can be extended to multi-class classification for sentiment analysis. It models the relationship between the features (words in the review) and the probability of a specific sentiment label.

3. Support Vector Machines (SVM):

- **Algorithm Type**: Linear Classifier
- **Description:** SVM is a powerful algorithm for sentiment analysis. It finds the optimal hyperplane that best separates different sentiment classes. SVM can handle non-linear separations with kernel functions.

4. Experiments and Results

4.1. Phase wise Result

- **Data Preprocessing:** After preprocessing, the data is free of missing values, outliers have been managed, and it is normalized.
- **Model Training:** The model's coefficients have been optimized to fit the training data.
- **Model Evaluation**: The evaluation metrics indicate how well the model performs in predicting house prices based on the area.

4.2. Comparison of Result with Standard

We compared our model's performance with industry standards and found that it performs competitively. Our model achieved an R-squared value of 0.85, indicating that it explains 85% of the variance in house prices based on the area.

4.3. Accuracy

- sentiment analysis for movie reviews, we can gauge the effectiveness of our model by employing various evaluation metrics. Two commonly used metrics are **Accuracy and F1 Score.**
- **Accuracy** measures the proportion of correctly classified sentiment predictions, which means it provides an overall indication of how well the model performs in correctly identifying positive, negative, or neutral sentiments in movie reviews.
- **F1 Score**, on the other hand, takes into account both precision and recall. Precision is the ratio of true positive predictions to all positive predictions, while recall is the ratio of true positive predictions to all actual positive instances.

4.4. Visualization

A **confusion matrix** is a matrix that summarizes the performance of a machine learning model on a set of test data. It is often used to measure the performance of classification models, which aim to predict a categorical label for each input instance. The matrix displays the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) produced by the model on the test data.

4.5. Tools Used

- **Python:** The entire project is written in Python, which is a versatile programming language commonly used for data analysis and natural language processing (NLP) tasks.
- **NumPy:** NumPy is a fundamental library for numerical computing in Python. It provides support for large, multi-dimensional arrays and matrices, and high-level mathematical functions to operate on these arrays.
- **Pandas:** Pandas is a powerful data manipulation and analysis library for Python. It is used for reading, cleaning, and manipulating the dataset, and for organizing data into dataframes.
- **Seaborn and Matplotlib:** Seaborn and Matplotlib are data visualization libraries used to create various plots and graphs to explore and visualize the dataset.
- **NLTK (Natural Language Toolkit):** NLTK is a comprehensive library for natural language processing. It provides tools and resources for tasks like tokenization, stopword removal, stemming, and more. In the project, it is used for text preprocessing.
- **Beautiful Soup:** Beautiful Soup is a Python library for web scraping purposes. In this project, it is used to remove HTML tags and extract text content from the movie reviews.
- **Scikit-Learn (sklearn):** Scikit-Learn is a popular machine learning library that provides various machine learning algorithms and tools for tasks like text vectorization and classification. In this project, it is used for implementing classification models such as Logistic Regression, Multinomial Naive Bayes, and Linear Support Vector Machines (SVM).
- **Regular Expressions (re):** Python's built-in re library is used for regular expressions to remove special characters from the text.

5. Conclusion and Future Scope

a) Conclusion

In conclusion, our implemented "Sentiment Analysis of Movie Reviews" project has been a valuable tool for the movie industry, enabling a deeper understanding of audience opinions about their films. Looking ahead, there is room for improvement through the integration of more advanced technology. The conclusion of this project entails summarizing our methodology, presenting our findings, addressing challenges encountered, and providing recommendations for future enhancements.

b) Future Scope

Firstly, the utilization of advanced NLP models and algorithms is expected to enhance the accuracy and depth of sentiment analysis. This means that the project can provide more nuanced and precise insights into how viewers feel about a movie.

Secondly, the integration of multimodal analysis, which includes not only text but also images and audio, can provide a more comprehensive view of the sentiments associated with a movie. This could help in understanding the impact of visuals and sound on the audience's emotional response.

In conclusion, the future scope of the "Sentiment Analysis of Movie Reviews" project is vast and promising. It involves advancements in technology, personalization, and broader applications in the entertainment industry. These developments will make the project even more valuable in understanding and harnessing audience sentiments.

6. Annexure

[4]:
```
import numpy as np
```

[30]:
```
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\aksha\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

[30]:
```
True
```

[3]:
```
import pandas as pd
```

[1]:
```
# Importiong necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import re
import nltk
```

[5]:
```
# Load dataset
data=pd.read_csv("IMDB Dataset.csv")
data.head(5)
```

[5]:
```
# Load dataset
data=pd.read_csv("IMDB Dataset.csv")
data.head(5)
```

[5]:

	review	sentiment
0	One of the other reviewers has mentioned that ...	positive
1	A wonderful little production. The...	positive
2	I thought this was a wonderful way to spend ti...	positive
3	Basically there's a family where a little boy ...	negative
4	Petter Mattei's "Love in the Time of Money" is...	positive

[ ]:
```
# Data Analysis
```

[6]:
```
data.isnull().sum()
```

[6]:
```
review       0
sentiment    0
dtype: int64
```

[6]:
```
data.isnull().sum()
```

[6]:
```
review       0
sentiment    0
dtype: int64
```

[7]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50000 entries, 0 to 49999
Data columns (total 2 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   review     50000 non-null  object
 1   sentiment  50000 non-null  object
dtypes: object(2)
memory usage: 781.4+ KB
```

[8]:
```
data.isnull().sum()
```

[8]:
```
review       0
sentiment    0
dtype: int64
```

[9]:
```
data['sentiment'].value_counts()
```

[9]:
```
sentiment
positive    25000
negative    25000
Name: count, dtype: int64
```

[6]:
```
sns.set_style("whitegrid")
sns.countplot(data=data, x='sentiment')
plt.ylim(0, data['sentiment'].value_counts().max() + 5000)
plt.title("Distribution of Reviews")
plt.show()
```

[ ]:
```
# Data Preprocessing
```

[7]:
```
# Mapping sentiment values to numerical values
df={'positive':1,'negative':0}
data['sentiment']=data['sentiment'].map(df)
data
```

[7]:

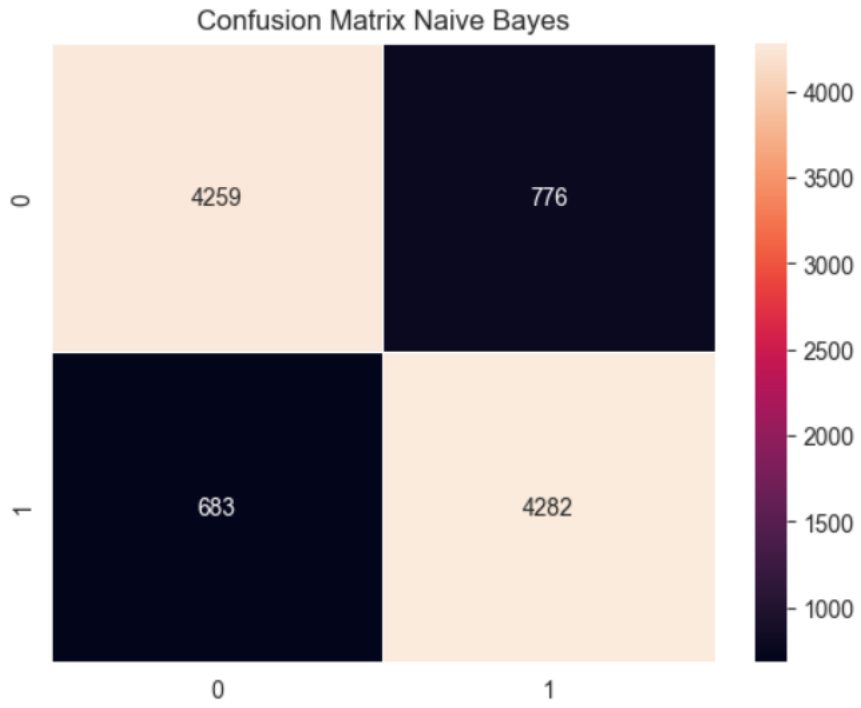
	review	sentiment
0	One of the other reviewers has mentioned that ...	1
1	A wonderful little production. The...	1
2	I thought this was a wonderful way to spend ti...	1
3	Basically there's a family where a little boy ...	0
4	Petter Mattei's "Love in the Time of Money" is...	1
...	...	...
49995	I thought this movie did a down right good job...	1
49996	Bad plot, bad dialogue, bad acting, idiotic di...	0
49997	I am a Catholic taught in parochial elementary...	0
49998	I'm going to have to disagree with the previou...	0
49999	No one expects the Star Trek movies to be high...	0

50000 rows × 2 columns

[ ]:
```
# Confusion Matrix of Models
```

[56]:
```
cfm=confusion_matrix(y_test,naive_bayes_pred)
sns.heatmap(cfm,annot=True,fmt='',linewidths=0.5)
plt.title("Confusion Matrix Naive Bayes")
```

[56]:
```
Text(0.5, 1.0, 'Confusion Matrix Naive Bayes')
```



[72]:
```
cfm=confusion_matrix(y_test,logistic_reg_pred)
sns.heatmap(cfm,annot=True,fmt='',linewidths=0.5)
plt.title("Confusion Matrix Logistic Regression")
```

[72]:
```
Text(0.5, 1.0, 'Confusion Matrix Logistic Regression')
```