**Project Report: Sentiment Analysis on IMDb Movie Reviews**

**1. Introduction**

The project aims to classify the sentiment of IMDb movie reviews as either positive or negative. Sentiment analysis is a natural language processing (NLP) technique widely used to determine people's opinions. This analysis helps in understanding audience feedback and behavior towards movies based on the collected text reviews.

**2. Problem Statement**

The goal of this project is to develop a model that can accurately classify movie reviews as either positive or negative, providing insight into the audience's reception of films. By applying machine learning models, we aim to evaluate which algorithms perform best on text sentiment classification.

**3. Dataset Overview**

* **Source**: IMDb Dataset
* **Format**: CSV file with two columns:
  + review: The text of the review
  + sentiment: The sentiment label (either *positive* or *negative*)

**Dataset Statistics**

* Basic statistics were extracted using:

df\_review.info()

df\_review.describe()

* The dataset contains **duplicates**, which were identified and removed to ensure data quality:

df\_review = df\_review[~df\_review.duplicated()]

**4. Data Preprocessing**

The preprocessing steps ensure that the text data is clean and suitable for machine learning models. The following operations were performed:

1. **Tokenization**: Splitting text into individual words using nltk.word\_tokenize.
2. **Lemmatization**: Normalizing words to their base forms using WordNetLemmatizer.
3. **Stopword Removal**: Removing common English stopwords from the NLTK library.
4. **Handling Contractions**: Expanding contractions (e.g., *"don't"* to *"do not"*) using the contractions library.
5. **Emoji Handling**: Removing or converting emojis using the emoji library.
6. **Feature Extraction**: Using **TF-IDF** (Term Frequency-Inverse Document Frequency) to convert text into numerical features.

**5. Model Selection and Training**

The following machine learning algorithms were tested to classify the sentiments:

1. **Logistic Regression**
2. **Linear Support Vector Classifier (SVC)**
3. **XGBoost**
4. **Decision Tree**

**Train-Test Split**

* The dataset was split into training and testing sets using:

train,test = train\_test\_split(df\_review,test\_size =0.3,random\_state=42)

**6. Model Evaluation**

The models were evaluated using **accuracy** and **confusion matrix** metrics.

**Key Metrics**

* **Accuracy Score**: Measures how often the model correctly classifies reviews.
* **Confusion Matrix**: Provides detailed information on true positives, true negatives, false positives, and false negatives.

**7. Streamlit App for Model Deployment and EDA**

To make the model easily accessible and to allow interactive visual exploration of the data, we developed a **Streamlit app**. The app offers the following functionalities:

* **Sentiment Prediction**: Users can input movie reviews directly into the app and receive a prediction of whether the sentiment is positive or negative.
* **Exploratory Data Analysis (EDA)**: The app visualizes key insights from the dataset, including:
  + Word clouds for positive and negative reviews.
  + Distribution of review sentiments.
* **Interactive Data Visualization**: Integrated with Plotly and Seaborn for rich, interactive plots.

The app provides a user-friendly interface and demonstrates the deployment of the trained models for real-world use.

**8. Results and Observations**

* **SVC** outperformed other models, achieving the highest accuracy.

**9. Conclusion**

This project successfully applied multiple machine learning models to perform sentiment analysis on IMDb reviews. The **SVC** provided the most reliable results. This analysis offers useful insights into how users respond to different movies, which could benefit film producers and critics by predicting audience reception.

**10. Code and Resources**

* **Libraries**: nltk, scikit-learn, matplotlib, seaborn, TfidfVectorizer, LogisticRegression, Streamlit, Plotly
* **Technologies**: Python, Streamlit
* **Deployment**: The Streamlit app allows users to interact with both the machine learning model and visual EDA tools.