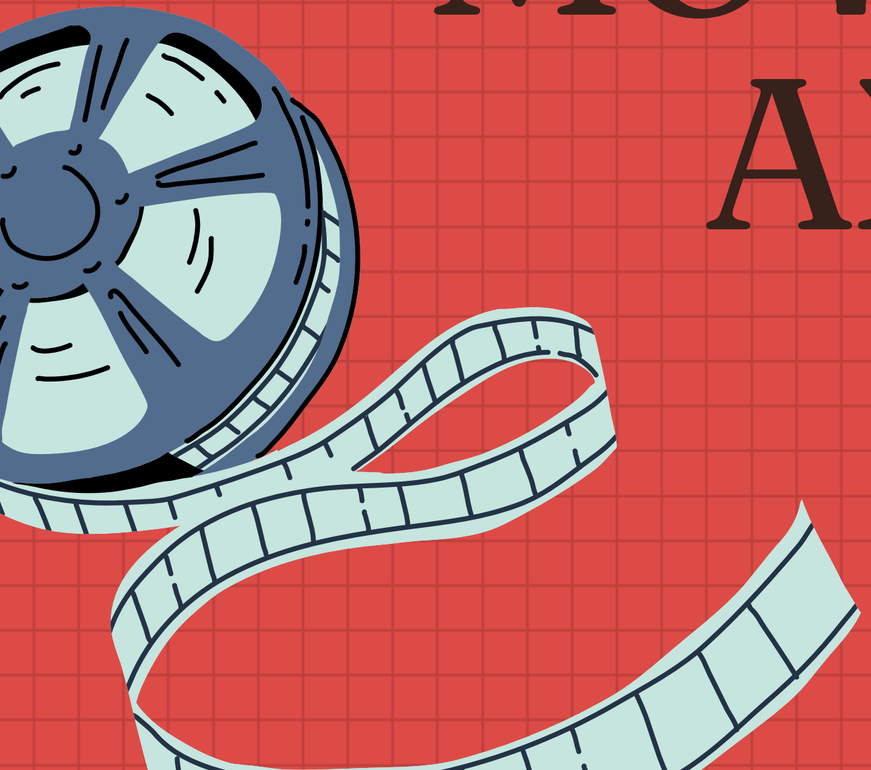
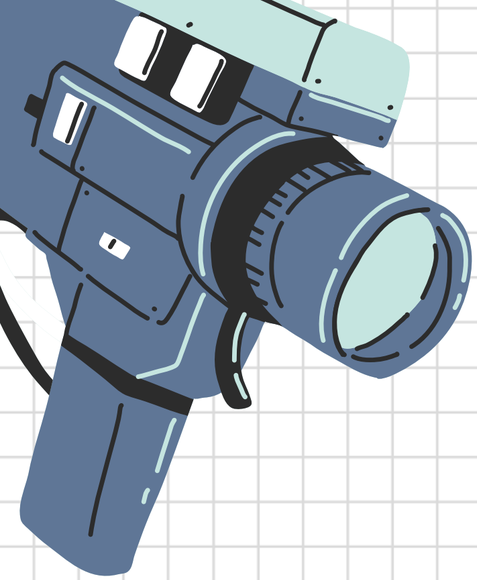
FUNDAMENTALS OF MACHINE LEARNING

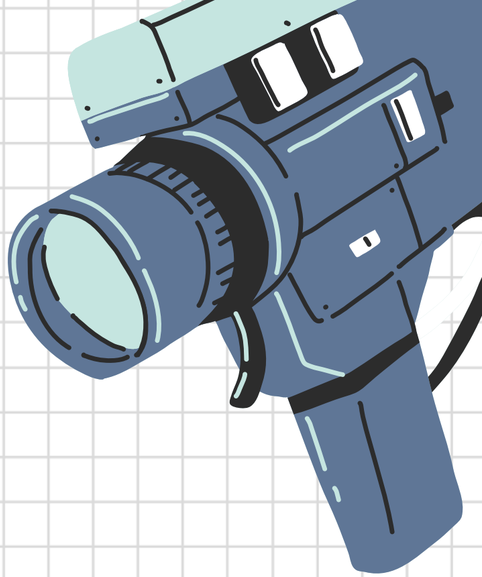
SUB CODE : AI23331

MOVIE REVIEW ANALYSIS



DINESH S 231801034
HARISH TUTU YT 231801050





Problem Statement

The objective of this project is to perform sentiment analysis on movie reviews. Sentiment analysis helps classify user opinions into categories such as positive or negative, which can be beneficial for understanding audience perception. In this case, we aim to develop and compare the performance of multiple machine learning models to predict whether a given movie review expresses a positive or negative sentiment.

INTRODUCTION

The increasing popularity of social media has also presented challenges for those who wish to improve products or offer new services. In this case, film producers need to know how well their customers respond to films. Inherent to movie descriptions are many personal views that are useful when it comes to analysing sentiments. This project will develop a machine learning based approach to building a classification model that discards positive movie reviews and summarizes only the negative ones.

Different algorithms such as Naive Bayes, Logistic Regression, Support Vector Machines (SVM) and K-Nearest Neighbours (KNN) incorporate the same principles. The first step is usually to clean the data then vectorize it using the CountVectorizer after which the data is supplied into the models. The intent also is to evaluate and discuss the findings especially the classification accuracy of the classifiers used in the experiments.

Proposed System

The proposed system involves the following components:

Collection and Processing of Data: A dataset consisting of movie reviews with associated sentiments (positive or negative) is employed. Noise, stop words, etc. are removed and the CountVectorizer is used to turn words into features to process the text data.

Training of Models: A part of the dataset is used to train four machine learning models (Naive Bayes, KNN, Support Vector Machine, and Logistic Regression) which are later used in sentiment classification of the reviews.

Testing of Models: Then, the accuracy of each trained model is calculated based on the results obtained with a previously untouched test set. The performance of the models is reported in terms of confusion matrices and accuracy scores.

Analysis and Custom Testing: Taking the model with the best quality, we also look into the importance of individual tokens (positive, negative and other sentiment carrying words), and the word 'awesome', and conduct sentiment analysis on any text review(s) provided.

Algorithm Steps

Data Loading: Load the movie reviews dataset in TSV format, which includes columns for sentiment labels (positive or negative) and review text.

data←pd.read_table(path)

Preprocessing: Convert text reviews into numerical feature vectors using CountVectorizer. Remove stop words, consider unigrams (single words), and filter out terms that appear too often or too rarely.

Xtrain,Xtest,ytrain,ytest=train_test_split(X,y,test_size=0.2)

Model Training:

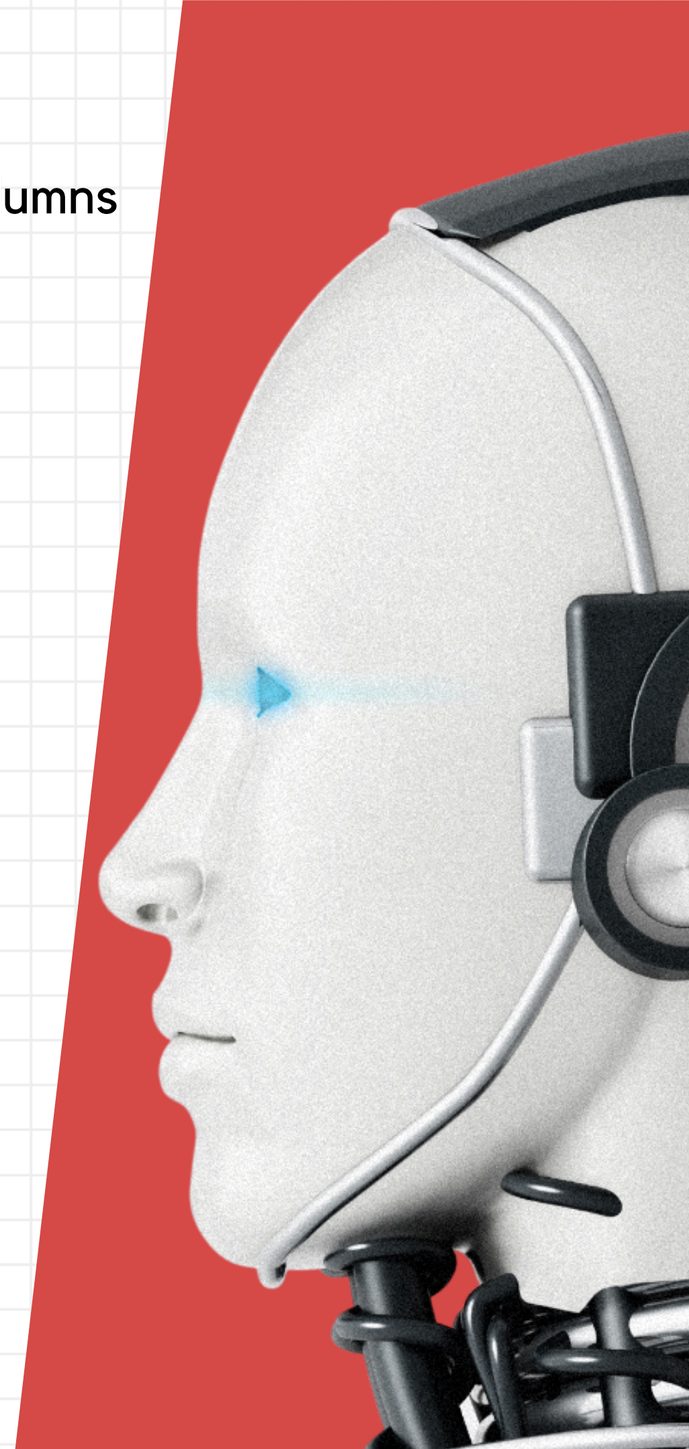
Train the following models on the feature matrix:

Naive Bayes: NB.fit(Xtrain,ytrain)

Logistic Regression: LR.fit(Xtrain,ytrain)

Support Vector Machine (SVM): SVM.fit(Xtrain,ytrain)

K-Nearest Neighbors (KNN): KNN.fit(Xtrain,ytrain)



Evaluation:

For each model, predict the sentiments for the test dataset and calculate accuracy using:

$$\text{accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Samples}}$$

Generate confusion matrices to identify the number of true positives, false positives, true negatives, and false negatives.

Analysis:

Extract token-level insights from the Naive Bayes model, determining how many tokens are categorized as positive or negative.

Token counts (Positive/Negative)

THANK YOU *FOR* WATCHING