||Jai Sri Gurudev||

**BGSKH Education Trust**

**BGS COLLEGE OF ENGINEERING AND TECHNOLOGY**

**Mahalakshmipuram, Bengaluru - 560086**



Branch : Artificial Intelligence & Machine Learning

Subject : Natural Language Processing

Subject Code : BAI601

Academic Year : 2024 - 25

**“Amazon Product Review Sentiment Analysis”**

**By:**

Name : AKSHATHA M A

: YASHASWINI D

USN : 1MP22AI003

: 1MP22AI063

**Faculty:**

Dr Madhura Gangaiah

Professor, Dept. of AIML

**Abstract**

With the exponential growth of e-commerce platforms like Amazon, user-generated product reviews have become a crucial source of information for both consumers and businesses. Analyzing the sentiment behind these reviews provides valuable insights into customer satisfaction and product performance. This project focuses on sentiment analysis of Amazon product reviews using Natural Language Processing (NLP) techniques. The objective is to classify reviews as positive, negative, or neutral based on their textual content. Various preprocessing methods such as tokenization, stopword removal, and lemmatization are applied, followed by feature extraction techniques like TF-IDF and word embeddings. Machine learning models including Logistic Regression, Support Vector Machine (SVM), and deep learning approaches like LSTM are employed and evaluated. The results demonstrate the effectiveness of NLP in understanding customer opinions and offer potential applications in automated feedback systems, product improvement, and marketing strategies.

**Introduction**

Online reviews play a significant role in shaping consumer behavior and influencing purchasing decisions in the digital marketplace. Among various e-commerce platforms, Amazon stands out due to its vast product range and substantial volume of customer reviews. However, manually analyzing this vast amount of textual data is impractical. Sentiment analysis, a subfield of Natural Language Processing (NLP), offers a solution by automatically determining the sentiment expressed in textual data.

This project explores sentiment analysis of Amazon product reviews with the aim of classifying each review as positive, negative, or neutral. By leveraging NLP techniques, this study transforms raw textual data into structured formats suitable for analysis. The process involves several key steps: text preprocessing, feature extraction, model training, and evaluation. Machine learning and deep learning algorithms are applied to build predictive models capable of understanding the sentiment embedded in user opinions.

**Project Code**

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import classification\_report

# Load the dataset

df = pd.read\_csv("Amazon\_review.csv.csv")

# Drop rows with missing reviews or ratings

df\_clean = df[['reviews.text', 'reviews.rating']].dropna()

# Remove neutral reviews (rating == 3) and create sentiment labels

df\_clean = df\_clean[df\_clean['reviews.rating'] != 3]

df\_clean['sentiment'] = df\_clean['reviews.rating'].apply(lambda x: 1 if x >= 4 else 0)

# Split into train and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(df\_clean['reviews.text'], df\_clean['sentiment'], test\_size=0.2, random\_state=42)

# Convert text to TF-IDF vectors

vectorizer = TfidfVectorizer(stop\_words='english', max\_features=5000)

X\_train\_vec = vectorizer.fit\_transform(X\_train)

X\_test\_vec = vectorizer.transform(X\_test)

# Train a Logistic Regression model

model = LogisticRegression()

model.fit(X\_train\_vec, y\_train)

# Evaluate the model

y\_pred = model.predict(X\_test\_vec)

print("Classification Report:\n")

print(classification\_report(y\_test, y\_pred))

# Show some sample predictions

sample\_reviews = X\_test.sample(5, random\_state=42)

sample\_preds = model.predict(vectorizer.transform(sample\_reviews))

print("\nSample Predictions:\n")

for review, pred in zip(sample\_reviews, sample\_preds):

print(f"Review: {review[:100]}...\nPredicted Sentiment: {'Positive' if pred else 'Negative'}\n")

**Output**

