

```
import pandas as pd
import nltk
from nltk.sentiment import SentimentIntensityAnalyzer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report

# Download VADER lexicon
nltk.download("vader_lexicon")

# Load IMDB dataset
df = pd.read_csv("IMDB Dataset.csv", encoding="utf-8", on_bad_lines="skip")
df = df.sample(10000, random_state=42) # Use a subset for faster processing

# Initialize Sentiment Analyzer
sia = SentimentIntensityAnalyzer()

# Apply VADER sentiment scoring
df["vader_score"] = df["review"].apply(lambda x: sia.polarity_scores(x)["compound"])
df["sentiment"] = df["vader_score"].apply(lambda x: "positive" if x > 0.05 else "negative" if x < -0.05 else "neutral")


# TF-IDF Vectorization
vectorizer = TfidfVectorizer(max_features=5000, stop_words="english")
X = vectorizer.fit_transform(df["review"])
y = df["sentiment"]

# Train-Test Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train Logistic Regression Model
model = LogisticRegression()
model.fit(X_train, y_train)

# Predictions
y_pred = model.predict(X_test)

# Evaluate Model
print("Accuracy:", accuracy_score(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

 [nltk\_data] Downloading package vader\_lexicon to /root/nltk\_data...  
[nltk\_data] Package vader\_lexicon is already up-to-date!  
Accuracy: 0.8165

	precision	recall	f1-score	support
negative	0.81	0.61	0.70	685
neutral	0.00	0.00	0.00	7
positive	0.82	0.93	0.87	1308
accuracy			0.82	2000
macro avg	0.54	0.51	0.52	2000
weighted avg	0.81	0.82	0.81	2000

/usr/local/lib/python3.11/dist-packages/sklearn/metrics/\_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero\_division` parameter to control.

/usr/local/lib/python3.11/dist-packages/sklearn/metrics/\_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero\_division` parameter to control.

/usr/local/lib/python3.11/dist-packages/sklearn/metrics/\_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero\_division` parameter to control.