In [2]:
```
import pandas as pd
import numpy as np
import re
import nltk
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.feature_extraction.text import TfidfVectorizer
from tqdm import tqdm
```

In [3]:
```
# --- NLTK Setup ---
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
```

```
[nltk_data] Downloading package punkt to C:\Users\samay
[nltk_data]     raj\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to C:\Users\samay
[nltk_data]     raj\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping corpora\stopwords.zip.
[nltk_data] Downloading package wordnet to C:\Users\samay
[nltk_data]     raj\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Out[3]: True

In [4]:
```
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
stop_words = set(stopwords.words('english'))
lemmatizer = WordNetLemmatizer()
```

In [5]:
```
df = pd.read_csv('IMDB Dataset.csv')
```

In [6]:
```
df['sentiment'] = df['sentiment'].map({'positive': 1, 'negative': 0})
```

In [7]:
```
def clean_text(text):
    text = re.sub(r'<.*?>', '', text)  # remove HTML
    text = re.sub(r'[^a-zA-Z]', ' ', text)  # remove numbers/symbols
    tokens = nltk.word_tokenize(text.lower())
    return ' '.join([lemmatizer.lemmatize(word) for word in tokens if word not in stop_words])
```

In [8]:
```
df['clean_review'] = df['review'].apply(clean_text)
```

In [9]:
```
# --- 2. TF-IDF + Logistic Regression ---
tfidf = TfidfVectorizer(max_features=5000)
X_tfidf = tfidf.fit_transform(df['clean_review'])
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, df['sentiment'], test_size=0.2, random_state=42)
```

In [10]:
```
model1 = LogisticRegression()
model1.fit(X_train, y_train)
preds1 = model1.predict(X_test)
```

In [11]:
```
print(" TF-IDF + Logistic Regression")
print("Accuracy:", accuracy_score(y_test, preds1))
print("Confusion Matrix:\n", confusion_matrix(y_test, preds1))
```

```
 TF-IDF + Logistic Regression
Accuracy: 0.8891
Confusion Matrix:
 [[4335  626]
 [ 483 4556]]
```

In [12]:
```
df.head(10)
```

Out[12]:

	review	sentiment	clean_review
0	One of the other reviewers has mentioned that ...	1	one reviewer mentioned watching oz episode hoo...
1	A wonderful little production. The...	1	wonderful little production filming technique ...
2	I thought this was a wonderful way to spend ti...	1	thought wonderful way spend time hot summer we...
3	Basically there's a family where a little boy ...	0	basically family little boy jake think zombie ...
4	Petter Mattei's "Love in the Time of Money" is...	1	petter mattei love time money visually stunnin...
5	Probably my all-time favorite movie, a story o...	1	probably time favorite movie story selflessnes...
6	I sure would like to see a resurrection of a u...	1	sure would like see resurrection dated seahunt...
7	This show was an amazing, fresh & innovative i...	0	show amazing fresh innovative idea first aired...
8	Encouraged by the positive comments about this...	0	encouraged positive comment film looking forwa...
9	If you like original gut wrenching laughter yo...	1	like original gut wrenching laughter like movi...

In [13]:
```
df['review'][0]
```

Out[13]:
```
"One of the other reviewers has mentioned that after watching just 1 Oz episode you'll be hooked. They are righ
t, as this is exactly what happened with me.<br /><br />The first thing that struck me about Oz was its brutali
ty and unflinching scenes of violence, which set in right from the word GO. Trust me, this is not a show for th
e faint hearted or timid. This show pulls no punches with regards to drugs, sex or violence. Its is hardcore, i
n the classic use of the word.<br /><br />It is called OZ as that is the nickname given to the Oswald Maximum S
ecurity State Penitentary. It focuses mainly on Emerald City, an experimental section of the prison where all t
he cells have glass fronts and face inwards, so privacy is not high on the agenda. Em City is home to many..Ary
ans, Muslims, gangstas, Latinos, Christians, Italians, Irish and more....so scuffles, death stares, dodgy deali
ngs and shady agreements are never far away.<br /><br />I would say the main appeal of the show is due to the f
act that it goes where other shows wouldn't dare. Forget pretty pictures painted for mainstream audiences, forg
et charm, forget romance...OZ doesn't mess around. The first episode I ever saw struck me as so nasty it was su
rreal, I couldn't say I was ready for it, but as I watched more, I developed a taste for Oz, and got accustomed
to the high levels of graphic violence. Not just violence, but injustice (crooked guards who'll be sold out for
a nickel, inmates who'll kill on order and get away with it, well mannered, middle class inmates being turned i
nto prison bitches due to their lack of street skills or prison experience) Watching Oz, you may become comfort
able with what is uncomfortable viewing....thats if you can get in touch with your darker side."
```

In [14]:
```
#count vectorizer arranges the words in alphabetical order.
#saving and using the model is called pickling.
import numpy as np
from sklearn.feature_extraction.text import CountVectorizer

count = CountVectorizer()

docs = np.array(['The sun is shining',
                 'The weather is sweet',
                 'the sun is shining, the weather is sweet, and one is two'])
bag = count.fit_transform(docs)
```

In [16]:
```
print(count.vocabulary_)
```

```
{'the': 6, 'sun': 4, 'is': 1, 'shining': 3, 'weather': 8, 'sweet': 5, 'and': 0, 'one': 2, 'two': 7}
```

In [ ]:
```
#21MID0181 Samay Raj Adhikari Hands ON 4
```