TASK-1 (MOVIE GENRE CLASSIFICATION)

PROGRAM:

```python
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 accuracy_score, classification_report,
confusion_matrix

# Step 1: Load the dataset from the local path
file_path = r"/content/train_data.txt"
data = pd.read_csv(file_path, sep=':::', engine='python',
header=None)  # Adjusted for your specific file format

# Inspect the data to understand its structure
print(data.head())
print(data.columns)

# Step 2: Data Preprocessing
# Manually specify the columns if they are not automatically parsed
correctly
data.columns = ['Index', 'Title', 'Genre', 'Plot']  # Example
columns based on the data sample you provided

# Now, let's focus on the 'Plot' for text and 'Genre' for labels
X_text = data['Plot']  # Features: movie plot summaries
y = data['Genre']  # Labels: movie genres

# Split the dataset into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X_text, y,
test_size=0.2, random_state=42)

# Step 3: Convert text data to numerical features using TF-IDF
```

```
tfidf = TfidfVectorizer(stop_words='english', max_df=0.7)
X_train_tfidf = tfidf.fit_transform(X_train)
X_test_tfidf = tfidf.transform(X_test)

# Step 4: Train the model (Logistic Regression)
model = LogisticRegression(max_iter=1000)
model.fit(X_train_tfidf, y_train)

# Step 5: Make predictions and evaluate the model
y_pred = model.predict(X_test_tfidf)
```

OUTPUT:

```
Confusion Matrix:
[[  53    0    0    0    0   23    1   38  111    0    0    0    0    9
     0    0    0    0    0    0    4    9    4    0   10    0    1]
 [   0   20    9    0    0   37    0    5   30    0    0    0    0    2
     0    0    0    0    0    0    0    8    0    0    0    0    1]
 [   3    1   14    0    0   17    0   31   48    1    0    0    0    9
     0    0    0    0    1    0    2    8    0    0    2    0    2]
 [   1    0    0    3    0   27    0   19   27    4    0    0    0    2
     1    0    0    0    0    0    6   14    0    0    0    0    0]
 [   0    0    0    0    0    2    0   38   19    0    0    0    0    0
     0    0    0    0    0    0    0    2    0    0    0    0    0]
 [   3    0    0    0    0  846    0   87  447    2    0    0    0   11
     2    0    0    0    3    0    1   36    0    0    5    0    0]
 [   6    0    0    0    0   15    1   14   59    0    0    0    0    3
     0    0    0    0    0    0    0    2    0    0    6    0    1]
 [   3    0    0    0    0   49    0 2300  220    0    0    0    0    7
    11    0    0    0    3    0    1   64    0    0    1    0    0]
 [   7    0    0    0    0  194    0  228 2192    0    0    0    0   11
     0    0    0    0    1    0    0   55    1    0    7    0    1]
 [   0    0    3    0    0   37    0   34   49    8    0    0    0    1
     4    0    0    0    2    0    0   11    0    1    0    0    0]
 [   0    1    4    1    0   12    0   11   30    0    0    0    0    6
     0    0    0    0    0    0    2    7    0    0    0    0    0]
 [   0    0    0    0    0   15    0    9    1    0    0   12    0    0
     0    0    0    0    2    0    0    0    1    0    0    0    0]
 [   0    0    0    0    0    0    0   27   18    0    0    0    0    0
```

```
    0   0   0   0   0   0   0   0   0   0   0   0   0]
[   3   0   0   0   0  27   0  29 100   1   0   0   0 245
    0   0   0   0   1   0   2  15   0   0   8   0   0]
[   0   0   0   0   0  11   0  61   9   0   0   0   0   0
   56   0   0   0   1   0   0   6   0   0   0   0   0]
[   0   0   0   0   0   9   0   7  26   0   0   0   0   0
    3   0   0   0   0   0   0   4   0   0   0   0   1]
[   0   0   0   0   0   7   1   5  29   0   0   0   0   5
    0   0   0   0   0   0   0   1   0   0   8   0   0]
[   0   0   0   0   0   3   0  30   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   1   0   0   0]
[   0   0   0   0   0  51   0  98  15   1   0   0   0   0
    1   0   0   0  22   0   0   4   0   0   0   0   0]
[   0   1   0   0   0  26   0   1 119   0   0   0   0   0
    0   0   0   0   0   0   0   4   0   0   0   0   0]
[   2   0   0   0   0  14   0  37  42   0   0   0   0  10
    0   0   0   0   1   0  24  10   0   0   3   0   0]
[   2   0   0   0   0 117   0 299 300   0   0   0   0   7
    1   0   0   0   1   0   0 317   0   0   1   0   0]
[   2   0   0   0   0   7   0  59   3   0   0   0   0   0
    0   0   0   0   0   0   0   6  15   1   0   0   0]
[   0   0   0   0   0  10   0  48   3   0   0   0   0   0
    2   0   0   0   4   0   0   7   0   7   0   0   0]
[   8   0   0   0   0  28   0  19 171   0   0   1   0  32
    0   0   0   0   0   0   0  15   0   0  35   0   0]
[   1   0   0   0   0   0   0   6  10   0   0   0   0   0
    0   0   0   0   1   0   0   2   0   0   0   0   0]
[   1   0   0   0   0  14   0   4  47   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   1   0 133]]
```