Mellors MSDS610 Week6 Assignment - Notebook 2: Running the Validation

Loading Libraries and Data

In [1]:

```
import pandas as pd
import joblib
from sklearn.metrics import accuracy_score, classification_report
```

In [2]:

```
xgb_model = joblib.load("xgb_model.pkl")
vectorizer = joblib.load("tfidf_vectorizer.pkl")
```

In [3]:

```
X_val = pd.read_csv("X_val.csv")
y_val = pd.read_csv("y_val.csv")
```

In [4]:

```
X_val.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 480 entries, 0 to 479
Data columns (total 1 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   combined_text  480 non-null    object
dtypes: object(1)
memory usage: 3.9+ KB
```

In [5]:

```
X_val.head()
```

Out[5]:

	combined_text
0	action comedy horror monster pub duringcre...
1	comedy drama mystery independent film every ...
2	action comedy crime new york money launderi...
3	comedy hotel infidelity onenight stand frie...
4	comedy alcohol baby party family fraternit...

In [6]:

```
y_val.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 480 entries, 0 to 479
Data columns (total 1 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   worth_funding   480 non-null    int64
dtypes: int64(1)
memory usage: 3.9 KB
```

In [7]:

```
y_val.value_counts()
```

Out[7]:

```
worth_funding
0                241
1                177
2                 62
Name: count, dtype: int64
```

Testing the Validation

In [8]:

```
X_val_tfidf = vectorizer.transform(X_val["combined_text"])
```

In [9]:

```
y_pred = xgb_model.predict(X_val_tfidf)
```

In [10]:

```
accuracy = accuracy_score(y_val, y_pred)
report = classification_report(y_val, y_pred)

print(f"Validation Accuracy: {accuracy:.4f}")
print("Classification Report:\n", report)
```

```
Validation Accuracy: 0.5292
Classification Report:
               precision    recall  f1-score   support

           0       0.54      0.77      0.63       241
           1       0.54      0.35      0.42       177
           2       0.33      0.11      0.17        62

    accuracy                           0.53       480
   macro avg       0.47      0.41      0.41       480
weighted avg       0.51      0.53      0.50       480
```

Summary

For this assignment, I decided to switch up the model I used. Last week I used an RF model, which performed poorly, and since this is a text-based dataset I decided to try XGBOOST (I also tried SVM and LogisticRegression, but they performed as poorly as the RF model). I liked the results I was getting from XGBoost - I have tuned it and ran it and re-ran it many times - my initial prediction was 98% and I rab my validation and got a 52%. So, I assumed that my model was overfitting. As such, I went back and adjusted the parameters for my XGB, by adjusting the parameters to reduce overfitting (improve generalization), which it did, bringing the accuracy down with it (to the now 72%). And, no matter what I did (including adding SMOTE and parameter tuning), my validation always stayed low. I am not sure if I am missing something, because I feel like my model is able to work well with the training data, or if it is something else. The only real conclusions I can come up with are: This is not a good model type for what I am trying to do, text-based features are not good predictors of film success, or I do not have enough data (my dataset is too small, <5,000 entries). With each parameter tuning to improve generalization, my validation did improve (very, very, very, slightly) to the point that as of submitting, my initial training models is 72% accuracy and my validation has 53% accuracy.

In [ ]: