```
# Install NLTK if not already available
import nltk
nltk.download('movie_reviews')
nltk.download('punkt')
nltk.download('stopwords')

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from nltk.corpus import movie_reviews
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize

from sklearn.feature_extraction.text import TfidfVectorizer
```

```
[nltk_data] Downloading package movie_reviews to /root/nltk_data...
[nltk_data]   Unzipping corpora/movie_reviews.zip.
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
# Load file IDs
positive_fileids = movie_reviews.fileids('pos')
negative_fileids = movie_reviews.fileids('neg')

# Read reviews as raw text
positive_reviews = [
    movie_reviews.raw(fileid) for fileid in positive_fileids
]

negative_reviews = [
    movie_reviews.raw(fileid) for fileid in negative_fileids
]

print(f"Positive reviews: {len(positive_reviews)}")
print(f"Negative reviews: {len(negative_reviews)}")
```

```
Positive reviews: 1000
Negative reviews: 1000
```

```
stop_words = set(stopwords.words('english'))

def preprocess_text(text):
    tokens = word_tokenize(text.lower())
    tokens = [
        word for word in tokens
        if word.isalpha() and word not in stop_words
    ]
    return " ".join(tokens)
```

```
import nltk
nltk.download('punkt_tab')

positive_cleaned = [preprocess_text(review) for review in positive_reviews]
negative_cleaned = [preprocess_text(review) for review in negative_reviews]
```

```
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt_tab.zip.
```

```
# TF-IDF Vectorizers
tfidf_pos = TfidfVectorizer(max_df=0.95, min_df=2)
tfidf_neg = TfidfVectorizer(max_df=0.95, min_df=2)

# Fit and transform
tfidf_pos_matrix = tfidf_pos.fit_transform(positive_cleaned)
tfidf_neg_matrix = tfidf_neg.fit_transform(negative_cleaned)
```

```
def get_top_tfidf_terms(tfidf_matrix, feature_names, top_n=15):
    mean_tfidf = np.mean(tfidf_matrix.toarray(), axis=0)
    tfidf_scores = dict(zip(feature_names, mean_tfidf))
    sorted_terms = sorted(tfidf_scores.items(), key=lambda x: x[1], reverse=True)
    return sorted_terms[:top_n]
```

```
# Feature names
pos_features = tfidf_pos.get_feature_names_out()
neg_features = tfidf_neg.get_feature_names_out()

# Top terms
top_pos_terms = get_top_tfidf_terms(tfidf_pos_matrix, pos_features)
top_neg_terms = get_top_tfidf_terms(tfidf_neg_matrix, neg_features)

print("Top 15 Positive TF-IDF Terms:")
for term, score in top_pos_terms:
    print(term, round(score, 4))

print("\nTop 15 Negative TF-IDF Terms:")
for term, score in top_neg_terms:
    print(term, round(score, 4))
```

```
Top 15 Positive TF-IDF Terms:
film 0.0532
movie 0.034
one 0.031
like 0.0216
story 0.0181
good 0.0178
also 0.0169
time 0.0165
life 0.0164
even 0.0161
would 0.016
character 0.0159
characters 0.0159
well 0.0156
much 0.0153

Top 15 Negative TF-IDF Terms:
film 0.0512
movie 0.0418
one 0.0317
like 0.0244
even 0.02
would 0.0185
good 0.0185
bad 0.0181
time 0.0175
get 0.0172
story 0.0168
much 0.0165
characters 0.016
plot 0.0159
character 0.0158
```

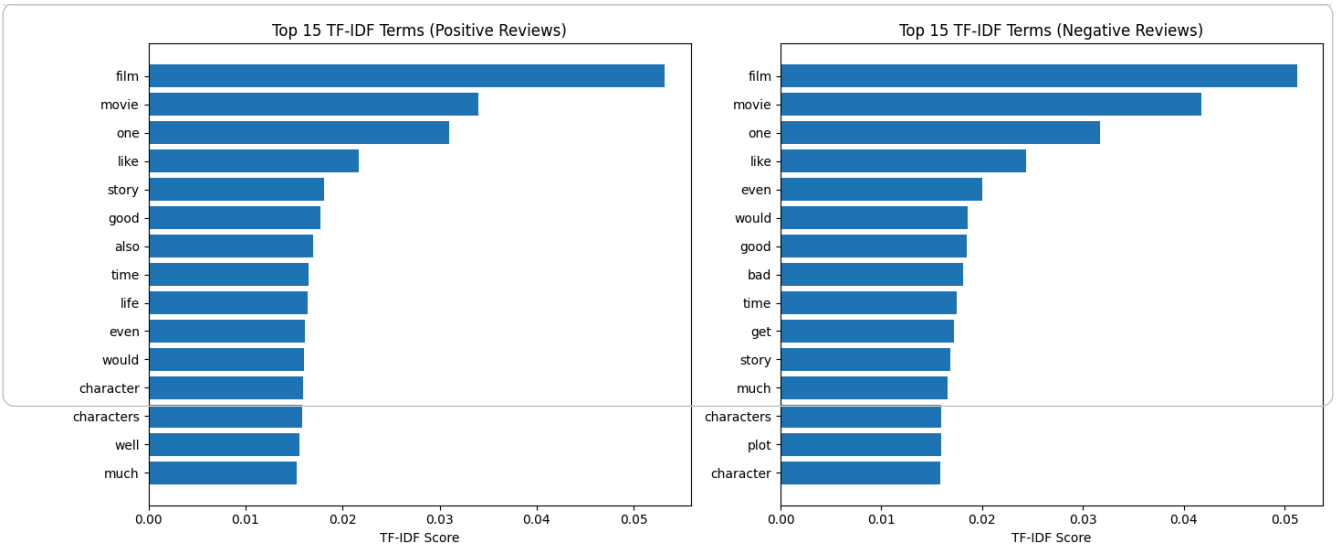
```
df_pos = pd.DataFrame(top_pos_terms, columns=["Term", "TF-IDF Score"])
df_neg = pd.DataFrame(top_neg_terms, columns=["Term", "TF-IDF Score"])
```

```
plt.figure(figsize=(14, 6))

# Positive reviews plot
plt.subplot(1, 2, 1)
plt.barh(df_pos["Term"], df_pos["TF-IDF Score"])
plt.title("Top 15 TF-IDF Terms (Positive Reviews)")
plt.xlabel("TF-IDF Score")
plt.gca().invert_yaxis()

# Negative reviews plot
plt.subplot(1, 2, 2)
plt.barh(df_neg["Term"], df_neg["TF-IDF Score"])
plt.title("Top 15 TF-IDF Terms (Negative Reviews)")
plt.xlabel("TF-IDF Score")
plt.gca().invert_yaxis()

plt.tight_layout()
plt.show()
```


Top 15 TF-IDF Terms (Positive Reviews)
film
movie
one
like
story
good
also
time
life
even
would
character
characters
well
much
0.00
0.01
0.02
0.03
0.04
0.05
TF-IDF Score
Top 15 TF-IDF Terms (Negative Reviews)
film
movie
one
like
even
would
good
bad
time
get
story
much
characters
plot
character
0.00
0.01
0.02
0.03
0.04
0.05
TF-IDF Score