

```
import pandas as pd

from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.metrics.pairwise import linear_kernel

# Sample movie dataset

movies = pd.DataFrame({

    'movie_id': [1, 2, 3, 4],

    'Title': ['Inception', 'The Matrix', 'Interstellar', 'The Prestige'],

    'description': [

        'A thief steals corporate secrets through dream-sharing technology.',

        'A hacker discovers reality is a simulation.',

        'Explorers travel through a wormhole in space.',

        'Two magicians engage in a rivalry and obsession.'

    ]

})

# User profile (likes sci-fi and thrillers)

user_profile_keywords = 'sci-fi space hacker dream reality'

# Vectorize movie descriptions

tfidf = TfidfVectorizer(stop_words='english')

tfidf_matrix = tfidf.fit_transform(movies['description'])

# Vectorize user profile

user_vec = tfidf.transform([user_profile_keywords])

# Compute cosine similarity
```

```
cosine_similarities = linear_kernel(user_vec, tfidf_matrix).flatten()

# Add similarity scores to movies DataFrame

movies['score'] = cosine_similarities

# Recommend top movies

top_recommendations = movies.sort_values(by='score', ascending=False)

print("Top Recommendations:\n")

print(top_recommendations[['title', 'score']])
```