```python
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

# Sample movie dataset
data = {
    'title': ['Inception', 'The Matrix', 'Titanic', 'The Notebook', 'Interstellar'],
    'genres': ['Action Sci-Fi', 'Action Sci-Fi', 'Romance Drama', 'Romance Drama', 'Adventure Sci-Fi']
}

# Convert to DataFrame
movies = pd.DataFrame(data)

# TF-IDF Vectorization on genres
vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(movies['genres'])

# Compute cosine similarity matrix
cosine_sim = cosine_similarity(tfidf_matrix, tfidf_matrix)

# Function to get recommendations
def recommend(movie_title, top_n=3):
    idx = movies[movies['title'] == movie_title].index[0]
    sim_scores = list(enumerate(cosine_sim[idx]))
    sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)[1:top_n+1]
    movie_indices = [i[0] for i in sim_scores]
    return movies['title'].iloc[movie_indices]

# Example usage
user_input = 'Inception'
recommendations = recommend(user_input)
print(f"Recommendations for '{user_input}':\n", recommendations.to_string(index=False))
```