MOVIE RECOMMENDATION SYSTEM

# 1. INTRODUCTION

In the digital era of streaming services and on-demand content, users are often overwhelmed by the vast number of available choices. A movie recommendation system helps in solving this issue by providing users with a personalized list of films based on their preferences or previous interactions. This project is a content-based movie recommendation system developed using Python, Pandas, Scikit-learn, and deployed using Streamlit. It delivers real-time recommendations based on user input with support for typo correction, auto-suggestions, and responsive UI.

# 2. OBJECTIVES

The objectives of this project are to:  
- Build a user-friendly and visually appealing movie recommendation interface.  
- Use a similarity matrix based on TF-IDF vectorization to recommend similar movies.  
- Provide real-time suggestions with auto-complete.  
- Handle user input errors by offering the closest matching movie title.  
- Ensure the web app is optimized for performance across different devices.

# 3. TECHNOLOGIES USED

- Python 3.13.3  
- Pandas  
- Scikit-learn  
- Streamlit  
- FuzzyWuzzy (fuzzy string matching)  
- Scipy

# 4. SYSTEM DESIGN AND MODULES

The system consists of multiple modules that work together to provide personalized movie recommendations. The key components are as follows:

**a. Data Preprocessing Module:**  
 - Cleans and filters the movie metadata.  
 - Applies TF-IDF vectorization for text-based features (genres, descriptions).  
 - Computes cosine similarity matrix.

**b. Recommendation Engine:**  
 - Takes user input.  
 - Matches it using fuzzy logic to the closest valid movie.  
 - Returns top 5 similar movies using the pre-computed similarity matrix.

**c. User Interface (Streamlit App):** - Provides a responsive and material-inspired layout.  
 - Offers a text input field with auto-completion suggestions.  
 - Displays recommended movie titles in styled vertical boxes.  
 - Handles invalid input gracefully and provides real-time suggestions.

# 5. USER INTERACTION FLOW

1. The user opens the web application in their browser.  
2. A randomly suggested movie is displayed on load with its recommendations.  
3. The user starts typing a movie name, and auto-suggestions appear.  
4. Upon pressing Enter or clicking a small 'Enter' button, recommendations are displayed.  
5. If the input movie is misspelled, the system corrects it using fuzzy matching and continues.  
6. The recommendations are shown in a horizontal layout of 5 styled vertical boxes.

# 6. FEATURES AND IMPROVEMENTS (V2.0.0)

- Auto-suggestion in the input box.  
- Fuzzy logic to correct typos in movie names.  
- Fully optimized and responsive layout.  
- Removal of deprecated Streamlit elements like `use\_column\_width`.  
- Improved UX by showing suggestions after the first typed word.  
- Fast UI rendering and optimized logic separation.

# 7. RESULT AND PERFORMANCE

The system performs exceptionally well for its intended purpose. The movie recommendation engine responds quickly to user inputs and provides highly relevant suggestions. The real-time suggestion and fuzzy correction algorithms improve usability. The system has been optimized for both performance and appearance on various screen sizes, making it ideal for mobile and desktop use.

# 8. FUTURE ENHANCEMENTS

- Integration with movie APIs (TMDB/IMDB) for dynamic content.  
- Adding user login and preference history.  
- Collaborative filtering or hybrid approach for better personalization.  
- Improved accessibility support and dark/light themes.

# 9. CONCLUSION

This project demonstrates how machine learning, combined with user-centric design and efficient deployment, can be used to develop a simple yet powerful movie recommendation tool. It successfully solves the problem of movie discovery and enhances user engagement through real-time responsiveness and intuitive interaction.