Movie Recommendation System Database

Domain: *Entertainment Industry Database Systems*

Problem Statement

Currently the entertainment or media industry generates massive amounts of data through streaming platforms, movie databases and user interactions. However, it is a challenge to effectively organize and utilize this data and provide personalized recommendations. The systems we have now lack the comprehensive database structures which can effectively handle complex relationships between movies, users, ratings, cast, and metadata while providing meaningful insights through advanced queries. Therefore, we need a properly designed database system that can store movie information, track user preferences and generate intelligent recommendations using SQL based algorithms.

Objectives

- Design relational database schema while following proper database design principles with multiple interconnected entities such as movies, ratings, users, reviews, genres, and cast.
- Implement advanced SQL features like stored procedures for recommendation algorithms, triggers for automation of data updates.
- Develop a recommendation engine that can suggest movies based on user rating patterns, genre preferences and other filtering techniques.
- Create data analysis capabilities that can identify trending movies, analyze user behavior patterns and generate statistics reports.
- Demonstrate database performance optimization through indexing, query optimization and efficient data retrieval mechanism for large datasets.

Solution Approach

Phase 1: Design database schema with core entities like (Users, Movies, Genres, Rating, Reviews) and junction tables (MovieGenre, MovieCast, Watchlist) to handle many-to-many relationships and ensure data integrity with the help of constraints and foreign key relationships.
Phase 2: Populate database with sample data and implement core functionality including CRUD operations.
Phase 3: Develop recommendation engine using SQL technique like filtering queries to identify users with similar tastes, content-based filtering based on movie attributes. Procedures like GetRecommendations() and GetSimilarMovies() for personalization and content-based recommendations.
Phase 4: Focus on optimization and documentation.

Expected Outcome

Working database with multiple tables like movies, users, genres, and cast that stores movie information and user data. I will try to make a recommendation system that will suggest movies to the users and the suggestions will be based on their previous ratings and preferences. Procedures and triggers will be handling core database operations like adding ratings, updating movie statistics and generating recommendations. Implementation of SQL queries that will help analyze data to find popular movies, user preferences. Documentation will include database design diagrams and basic instruction on how to use the database.