# Movie Recommendation System Using Neural Collaborative Filtering

## Introduction

In the digital age, movie recommendation systems play a crucial role in personalizing user experiences by suggesting relevant content. This project aims to develop a movie recommendation system using Neural Collaborative Filtering (NCF), a technique that leverages deep learning models to predict user preferences based on their historical interactions with movies.

## Data Description

### Dataset

The recommendation system uses the MovieLens dataset, which contains user ratings and movie information. This dataset is widely used for building recommendation systems and provides a robust foundation for various machine learning approaches.

#### *Files*

1. **rating.csv**: Contains user ratings for movies.
    - Columns: `userId`, `movieId`, `rating`, `timestamp`
2. **movie.csv**: Contains details about movies.
    - Columns: `movieId`, `title`, `genres`

### Sample Data

- **Rating Data:**
    - `userId`: Identifier for the user.
    - `movieId`: Identifier for the movie.
    - `rating`: User's rating for the movie.
    - `timestamp`: Time of the rating (not used in this analysis).
- **Movie Data:**
    - `movieId`: Identifier for the movie.
    - `title`: Title of the movie.
    - `genres`: Genres associated with the movie.

## Steps Involved

1. **Data Loading and Preprocessing:**
    - Load the rating and movie data from CSV files.
    - Extract relevant data and truncate for efficiency.
    - Encode user and movie IDs to create a mapping for the neural network.
2. **Feature Engineering:**
    - Convert user IDs and movie IDs to numerical indices.

    - Normalize the ratings to a range between 0 and 1.
3. **Train-Test Split:**
    - Split the data into training and testing sets (90% training, 10% testing).
4. **Model Definition:**
    - Define a Neural Collaborative Filtering model using TensorFlow and Keras.
    - Use embedding layers for users and movies.
    - Build a neural network to predict user ratings for movies.
5. **Model Training:**
    - Train the model on the training data.
    - Evaluate the model on the test data.
6. **Prediction and Recommendation:**
    - For a randomly selected user, predict ratings for movies they haven't watched.
    - Recommend the top 10 movies with the highest predicted ratings.

# Methods and Methodology

### Neural Collaborative Filtering (NCF)

NCF combines collaborative filtering with neural network-based techniques to model user-item interactions. The process involves:

1. **Embedding Layers:**
    - **User Embeddings:** Converts user IDs into dense vectors.
    - **Movie Embeddings:** Converts movie IDs into dense vectors.
2. **Dot Product:**
    - Calculate the interaction between user and movie embeddings.
3. **Dense Layers:**
    - Apply dense layers to capture complex relationships between users and movies.
4. **Prediction:**
    - Output the predicted rating for each user-movie pair.

### Model Architecture

- **User Embedding Layer:** Maps user IDs to dense vectors.
- **Movie Embedding Layer:** Maps movie IDs to dense vectors.
- **Dot Product Layer:** Computes the interaction between user and movie vectors.
- **Dense Layers:** Three dense layers to refine the prediction.

### Evaluation Metrics

- **Mean Squared Error (MSE):** Used to measure the performance of the model during training.

# Future Work

1. **Model Improvement:**
    - Experiment with different architectures, such as adding dropout layers or using different activation functions.
    - Incorporate additional features such as user demographics or movie attributes.
2. **Scalability:**

    - Optimize the model for larger datasets to improve efficiency and scalability.
3. **Hybrid Models:**
    - Combine collaborative filtering with content-based filtering for more accurate recommendations.
4. **User Feedback Integration:**
    - Implement mechanisms to continuously update recommendations based on user feedback and new ratings.

# Conclusion

The Neural Collaborative Filtering approach effectively models user preferences and provides personalized movie recommendations. The implemented system demonstrated the ability to suggest relevant movies based on user history, showcasing the potential of neural networks in recommendation systems.

# Results

For a randomly selected user, the system was able to recommend movies that they had not previously watched but were predicted to be of high interest. This included:

1. **Top Movies Watched by User:**
    - Provided a list of movies the user had highly rated in the past.
2. **Top 10 Movie Recommendations:**
    - Generated a list of 10 movies with the highest predicted ratings for the user.

# Summary

The project successfully built a movie recommendation system using Neural Collaborative Filtering. By leveraging user and movie embeddings, the model was able to provide accurate and personalized recommendations. Future improvements could focus on expanding the model's capabilities and integrating additional data sources for enhanced performance.