Project Title: Recommender Systems

## **Table of Contents**

1. Introduction
2. Task 0: Initial Setup
3. Task 1: User-based Collaborative Filtering
4. Task 2: Item-based Filtering
5. Task 3: A Better Recommender System

* 3.1 Develop a New Solution
* 3.2 Evaluate Performance

1. Task 4: Presentation
2. Conclusion
3. References

## **1. Introduction**

The project focuses on the implementation and evaluation of recommender systems, a crucial aspect of data science widely employed in real-world applications. The primary objective is to develop and assess user-based collaborative filtering (Task 1), item-based collaborative filtering (Task 2), and propose an enhanced recommender system (Task 3) using the MovieLens 1M dataset. The MovieLens 1M dataset provides a comprehensive collection of user ratings for movies, serving as the foundation for building and testing various recommendation algorithms.

**Project Goals**

1. Implement user-based collaborative filtering using KNN and study the impact of different K values on the prediction accuracy.

2. Develop and evaluate item-based collaborative filtering employing KNN, comparing the performance of different similarity metrics.

3. Propose a novel solution for movie recommendation, either based on existing publications (Option 1) or through an original algorithm (Option 2), and assess its effectiveness.

4. Compare the performance of the proposed solution with two baseline methods (Movie Average and KNN-based Collaborative Filtering) using evaluation metrics such as Average Precision (AP) and Normalized Discounted Cumulative Gain (NDCG).

**Dataset: MovieLens 1M**

The dataset, MovieLens 1M, serves as the cornerstone of this project. It contains a vast collection of user ratings for movies, enabling the exploration and implementation of various recommender system algorithms. The dataset provides essential information such as user ratings, movie details, and user preferences, essential for training and evaluating the recommendation models.

By leveraging this dataset, the project aims to contribute insights into the effectiveness of different collaborative filtering approaches and propose an innovative solution to enhance the overall quality of movie recommendations.

## **2. Task 0: Initial Setup**

**Dataset Download**

To initiate the project, the first step involves downloading the MovieLens 1M dataset. This dataset, available in the form of “ml-1m.zip”, contains essential information about user ratings for movies. The README.txt file within the dataset provides comprehensive details about its structure and contents.

**The following steps outline the dataset download process:**

1. Download the Dataset: Obtain the “ml-1m.zip” file from the provided source.

2. Extract the Dataset: Unzip the contents of `ml-1m.zip` to access the necessary files and folders for analysis.

**Jupyter Notebook Creation**

Creating a Jupyter Notebook provides a structured environment for developing, implementing, and documenting the code. Here are the steps for setting up the Jupyter Notebook:

1. Create a New Jupyter Notebook: Open your preferred Python environment (preferably Anaconda) and create a new Jupyter Notebook named `assignment3.ipynb`.

2. Organize Code Sections: Utilize the provided template (`assignment3-copy1.ipynb`) to organize the code into distinct sections, ensuring clarity and readability.

3. Add Comments: Throughout the code, include detailed comments to explain the implementation of various tasks. This is crucial for understanding the logic behind each step.

**Code Organization**

Effectively organizing the code within the Jupyter Notebook enhances readability and simplifies the debugging process. Follow these guidelines for code organization:

1. Cell Structure: Divide the code into cells, each dedicated to a specific task or analysis.

2. Commenting: Add comments to elucidate the purpose and functionality of each code segment.

3. Remove Unnecessary Lines: Clean the code by removing any redundant or unnecessary lines to maintain conciseness.

4. Kernel Restart & Run All: Before submission, ensure the code runs seamlessly by executing the "Restart & Run All" option from the Jupyter Notebook's Kernel menu.

## 3. Task 1: User-based Collaborative Filtering

### 3.1 Implementation

Describe the steps taken to implement user-based collaborative filtering using KNN. Include code snippets and comments.

### 3.2 Evaluation

Explain how the impact of different K values was studied and how RMSE was used for evaluation. Summarize results and findings.

---

## 4. Task 2: Item-based Filtering

### 4.1 Implementation

Detail the steps to implement item-based collaborative filtering using KNN. Discuss the choice of parameters and metrics.

### 4.2 Evaluation

Compare the performance of different similarity metrics and summarize the results using RMSE.

---

## 5. Task 3: A Better Recommender System

### 5.1 Develop a New Solution

#### 5.1.1 Option 1: Solution from Publications

Describe the chosen solution, provide citations, and explain the implementation in Python with comments.

#### 5.1.2 Option 2: Proposed Algorithm

Explain the proposed algorithm, justify its originality, and provide a strong list of references.

### 5.2 Evaluate Performance

#### 5.2.1 Recommendation to Users

Explain the process of recommending Top-30 movies to selected users and the evaluation metrics used (AP and NDCG).

#### 5.2.2 Results Comparison

Present a comparison of Movie Average, KNN-based Collaborative Filtering, and the new solution's performance using visualizations.

---

## 6. Task 4: Presentation

Summarize key results and findings in slides with the following sections:

- Cover page/slide & project info

- Task 1 results

- Task 2 results

- Task 3.1 details and citations

- Literature review (if applicable)

- Algorithm details for Option1RecSys or Option2RecSys

- Task 3.2 key results and visualizations

- List of references

---

## 7. Conclusion

Summarize the overall findings and the effectiveness of the proposed recommender system.

---

## 8. References

List all the references used in the project, including publications, articles, and relevant documentation.