

AI 705: Mini Project

I. DESCRIPTION

The objective of this project is to build a movie recommendation system, where based on the existing reviews of an user, you should suggest **new movies** that the user will like. The dataset chosen for this project is the MovieLens 1M dataset (Refer to the README.txt file for metadata of the dataset) where you are expected to implement K-Means, SVD, Collaborative filtering **using only numpy and pandas**. Additionally, you are encouraged to build upon these points by incorporating ideas you learnt in the class as novelty carries weight as mentioned in the evaluation metrics below.

Note : The dataset attached above has 1M ratings datasets. (It is mandatory that you retain atleast 1 movie of each genre if you choose to remove some columns as part of your pre-processing on the dataset.)

You **SHOULD** implement, compare and contrast both these methods :

- 1) Neighbourhood-based Collaborative Filtering
- 2) (K-means clustering & Reduced SVD) or (Reduced SVD & K-means clustering).

Make sure your project and presentation aligns with the evaluation metrics.

II. EVALUATION METRICS

Following are the evaluation metrics used to grade your midterm presentations:

1) Q&A on Analysis:

- **Analysis of Algorithms:** Why were a certain number of clusters selected in k-means? How many singular values were removed in SVD and why?
- **Analysis of Results:** Through plots and metrics (not just descriptive reasoning), what are the interesting patterns observed in the results? How did SVD help? Reiterating that, your arguments **must** be supported by numbers, metrics, plots.

- 2) **Novelty:** What novel ideas are incorporated in this project and why? What were your motivations for using novel ideas? Following are some areas where you may come up with novel or out-of-the-box ideas and implement them.

- Getting good inferences from Exploratory Data Analysis and USING them in your recommendation algorithm.
- Coming up with different evaluation/loss functions for judging the quality of your recommendations.

Apart from these, you are highly encouraged to be implement your own novel ideas, even if they provide sub-optimal recommendations, for maximizing this component.

3) **Presentation:**

- **Quality of slides:** Slides must not be text-heavy. Use images, plots, metrics, etc. in your presentation.
- Make the presentation coherent, meaningful and one that captures the philosophy of your algorithm well.
- Minimum performance of everyone in the team (adversarial) will be the group's presentation marks. Read IV for more details.

4) **Testing of recommendations:**

- **Input:** We will give you the ratings of some of our favourite movies from the dataset.
- **Output:** Your system should recommend top 5 **new movies**.
- **Explainability:** You are expected to explain why your system recommended us those movies and when it would recommend other movies.

It is important to note that you will be graded as a group and not at an individual level.

III. SANITY CHECKS

The following are the sanity checks every project **must** pass. Otherwise, your submission will be invalidated:

- 1) The codes will go through rigorous Plagiarism checks.
- 2) The plots/numbers/metrics in the presentation slides should match the output when your submitted codes are run. Fudging of results will not be tolerated at any degree.
- 3) Basic checks will be done to validate that **all** team members are involved in the project during the presentation.

IV. FORMAT OF PRESENTATION

The duration and slots of each presentation will be decided based on number of teams and feasibility. It will be communicated very soon once all logistics are finalised. Take notice of the following points:

- 1) **ALL the team members** should be present for the evaluation. No exceptions will be entertained.
- 2) Teams must respect their presentation duration and slot with highest precision. Penalty will be levied if a group exceeds their allotted duration or come late to the presentation (proportional to number of additional seconds consumed).
- 3) **TA will decide** the order and duration of each speaker in a group. For example, say there are four members in the group A, B, C, D . TA can first ask B to present. After a random amount of time, he will ask D to take over, and this process is repeated.
- 4) The teams must upload their implementation codes (with instructions on how to run your code) by 20th Feb EOD. The teams are expected to download their zip file submissions in-front of the TA for the viva on their implementation codes. Stringent plagiarism checks will be run on the submissions.