Movie Recommendations with PageRank
Team Project

Sehyun Kwon

Mathematical Data Science
Hanyang University ERICA

Project goals and workflow

Use our class's favorite movies to build a recommendation system and explain its results.

1. Submit your favorite movies individually.
2. Receive the cleaned, shared class dataset.
3. Analyze the data as a team.
4. Present your findings and explain their limitations.

Every team investigates **all three required questions** and **at least one optional question**.

Submitting movies through Google Forms

Submit **5–10 movies** that you have watched and enjoyed. Please list **10 movies if possible**. At least 5 are required.

- Enter your full name.
- Use one movie field per title, including its release year.
  Example: *Interstellar (2014)*.
- List individual films rather than an entire series.
- Do not list the same movie twice. No ranking is required.



Scan to submit

All submitted movies receive equal preference weight.

forms.gle/XoHaRkkobQZJk9hr7

The submission deadline will be announced separately.

What the data tells us

Movie titles and duplicate spellings will be cleaned before the class dataset is shared with all teams.

A submitted movie indicates a positive preference. An omitted movie does **not** tell us whether the student has watched or disliked it.

Our recommendation candidates are movies **the student did not submit**. Some may already be familiar to the student.

Use the class data only for this course project. Do not publish students' names or individual responses outside the course.

Exploring preferences in both directions

Create student nodes and movie nodes. Connect a student and each movie they submitted in **both directions**.

- Student to movie: move to a movie this student likes.
- Movie to student: move to a student who likes this movie.

One possible exploration path is

$$\text{me} \to \text{movie I like} \to \text{another student} \to \text{another movie.}$$

The reverse edge is an exploration rule. It does not mean that a movie recommends a student.

PageRank and implementation choices

Global PageRank: teleport uniformly to any student or movie node.

Personalized PageRank: teleport to the target student.

Rank movie nodes only. For personalized recommendations, exclude movies the target student submitted.

Your team chooses the implementation tools, α, initial scores, stopping criterion, and iteration limit. Document your choices, including dangling-node handling.

Explain why your choices are reasonable and how you checked correctness and stability. State any library functions used.

Required question 1: popularity and PageRank

Compare two rankings:

- **Popularity:** number of students who submitted each movie.
- **Global PageRank:** the scores of movie nodes.

How much do the top 10 lists overlap? Which movies change position the most?

Can two equally popular movies have different PageRank scores? Explain the result using their connections.

Similar rankings are also a finding. Explain similarities as well as differences, and state how you handle ties.

Required questions 2 and 3: recommendations

2. Do different students get different recommendations? Choose two students with different submitted lists. Compare their top 5 personalized recommendations and the popularity ranking.

3. Why was this movie recommended? Choose one recommendation. Show a small subgraph with paths connecting the target student to that movie. Explain which connections help account for the result.

The score reflects many paths, not just one selected path.

If fewer than five eligible movies have positive scores, report that. Do not treat zero scores as evidence for a recommendation.

Optional questions: changing the graph or rules

Choose at least one of the five optional questions, or propose your own.

1. Replace one submitted movie. How much do the student's recommendations change? Do other students' recommendations change too?

2. Change α. Compare recommendation rankings and the number of iterations needed to meet your stopping criterion.

3. No shared favorites. What if nobody else submitted any of a student's movies? What changes when one shared favorite is added? You may construct a small example if needed.

Optional questions: hidden favorites and feedback

4. Hide one favorite. Remove both directions of one student–movie connection. Does the movie appear among that student's recommendations?

Remove the hidden connection from popularity counts too. Distinguish movies that retain other connections from movies that become isolated.

5. Ask the recipient. With their consent, ask a student whether they have seen, like, or are interested in the recommended movies.

Keep modified or constructed data separate from the original data. Do not generalize a small feedback sample to the entire class.

What to submit

Report: model and dataset summary, implementation checks, all required questions, your optional investigation, limitations, and each member's contribution.

Code and instructions: data processing, dependencies, execution steps, settings, and reasons for your choices.

Results and presentation slides: ranking comparisons, two students' recommendations, an explanatory subgraph, and the optional experiment.

Report length, presentation time, and deadlines will be announced separately. An interactive website or animation is optional.

Assessment and final reflection

We will consider your understanding of the model, implementation checks, experimental design, interpretation, and reproducibility.

PageRank does not have to outperform popularity ranking. If the results are similar or recommendations fail, investigate why.

What was your most surprising result? Can you explain it through the graph's connections?

What does your system know about a student, and what does it not know?

Other project topics are welcome

You are welcome to choose your own PageRank project topic. **The ideas below are only examples. You may choose a different topic that is not listed here.**

- **Game recommendations:** find games a user may enjoy.
  Steam review datasets (UCSD)
- **Sports rankings:** compare PageRank with league standings.
  Football-Data: match results (CSV)
- **Streamer recommendations:** discover new channels.
  Twitch interaction dataset (UCSD)
- **Communication networks:** compare in-degree and PageRank.
  Email-Eu-core dataset (Stanford SNAP)
- **Restaurants or music:** recommend places or artists using a class survey that your team collects with participants' consent.