Recommendation Engines
- Netflix wants its users to enjoy movies ... but the TV screen can only display a small number of movies ...

- How can Netflix ensure that users enjoy the movie they watch

- If you leave it for the users to pick ... either he will have to scroll and search a lot (poor experience) or she might quickly choose a bad movie (poor experience).

- Netflix wants to optimize user experience by predicting movies users will like ... and recommending them to users.
WHAT COMPANIES CARE ABOUT THE PROBLEM?

- Netflix (movies)
- Amazon (shopping)
- Search engines (ranking news items)
- Spotify (recommending music)
- Google news (customizing news recommendations)
- Yelp (recommending restaurants and services)
- Goodreads (book recommendations)

... many many more
Isn't this problem easy to solve? How would you do it?

Our idea #1:

Alice

- Action: 20%
- Horror: 5%
- Drama: 25%
- Romance: 15%
- Comedy: 30%

Our idea #2:

- MI
- F&F
- Top G.
- Titanic
- Conjurii

80% rated ≥ 4/5
32%
11%
3%
8%
2 action movies from your pool
SOME HURDLES IN DESIGNING RECOMMENDATION ENGINES

- Say Alice watched W, X, Y … Bob watched X, Y, Z … and now Steve is a new user who has watched X and Y …
- What would you recommend to Steve?
- Would you take some average of W and Z? What does that mean?

- If Steve watched Terminator, Matrix, and Borne Identity … are you only going to recommend action movies?
- Are you sure Steve may not like comedy? Or Sci-Fi?

- When you are starting out as a company, you don’t have much user data … what do you do?

- How do you know your recommendation worked well or not?
Quick Foundations: Vectors, Vector spaces, Basis vectors, Feature Vectors

A movie is a point in N-dimensional feature space. Dimensions or Axis define features or attributes of the movie.

GPS = \begin{bmatrix} 33.67 \\ 42.11 \end{bmatrix}

Red \begin{bmatrix} 255 \\ 0 \\ 0 \end{bmatrix}

Greenish Blue \begin{bmatrix} 33.67 \\ 42.11 \end{bmatrix}

\text{Long} \begin{bmatrix} 42.11 \end{bmatrix}

\text{Lat} \begin{bmatrix} 33.67 \end{bmatrix}

R = \begin{bmatrix} 12 \\ 65 \\ 200 \end{bmatrix}
3 TYPES OF RECOMMENDATION ENGINE TECHNIQUES

- Content based filtering
- Collaborative filtering
- Hybrid techniques

Hybrid
- What is a feature space?

- Every movie is a point in this feature space

- Can even treat a user as a (bunch of points on this feature space).

- Recommend movies that are similar to the user.

- BTW, what does “similar” mean?
(2) COLLABORATIVE FILTERING

- Design M representative users — called EIGENUSERS
- Express any new user as a weighted combination of eigenusers.
- Derive the recommendation from these weights.

**Singular Value Decomposition (SVD)** extracts the main type of users present in the data.
Steve = \( w_1 u_1 + w_2 u_2 + \ldots + w_m u_m \)

\[ \downarrow \quad \downarrow \quad \downarrow \]

20% 5% 11%
(3) HYBRID TECHNIQUES

Recommendation

Combiner

CF Based Recommender

Input

Content Based Recommender

Input
Companies need data for content-based or collaborative filtering. Where are they getting the data?
- Cookies in your browser
- Your visited websites
- Your shopping patterns
- Your search queries in the Internet

This data is feeding recommendation engines … but also leaking a lot of information about you to the Internet.

What if tomorrow, a Government says … you have been eating junk food, so we are revoking your medical insurance.

Companies using data for shortlisting candidates for a job …
- Suppose the intelligent algorithm uses data from the past candidates who were, or were not, recruited.
- Trains the eigenusers from this data

What’s the problem?

What kind of other biases can you think of … when data is used to create the “representative” samples … the EIGENITEMS? Are there other biases or fairness?