Big Data Analytics & Applications

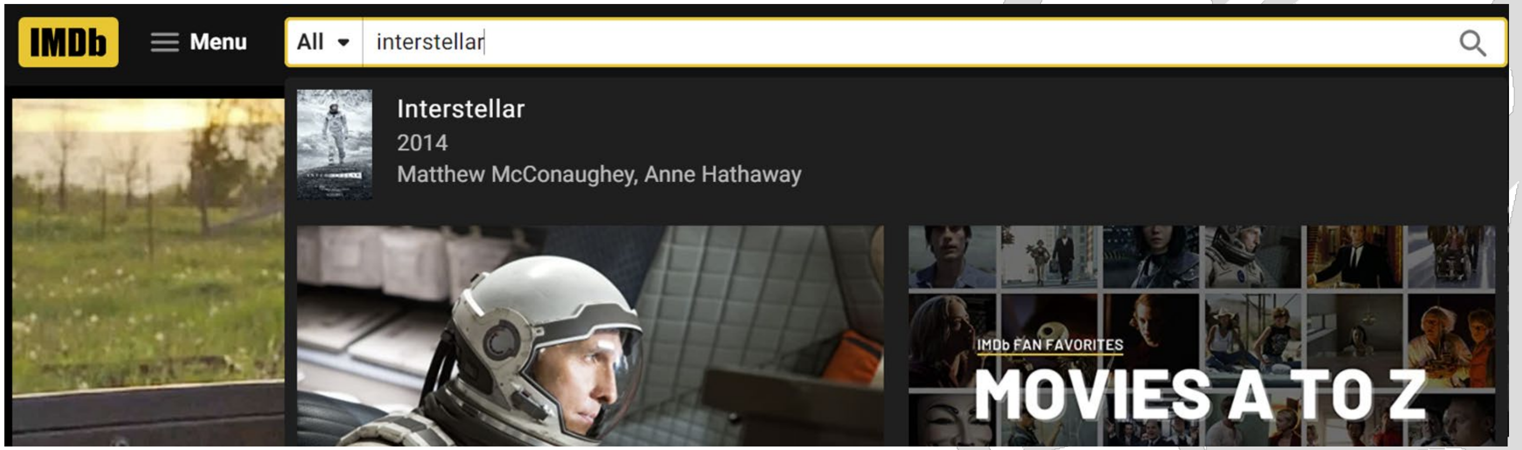
Bin Li

School of Computer Science

Fudan University

Search vs Recommendation

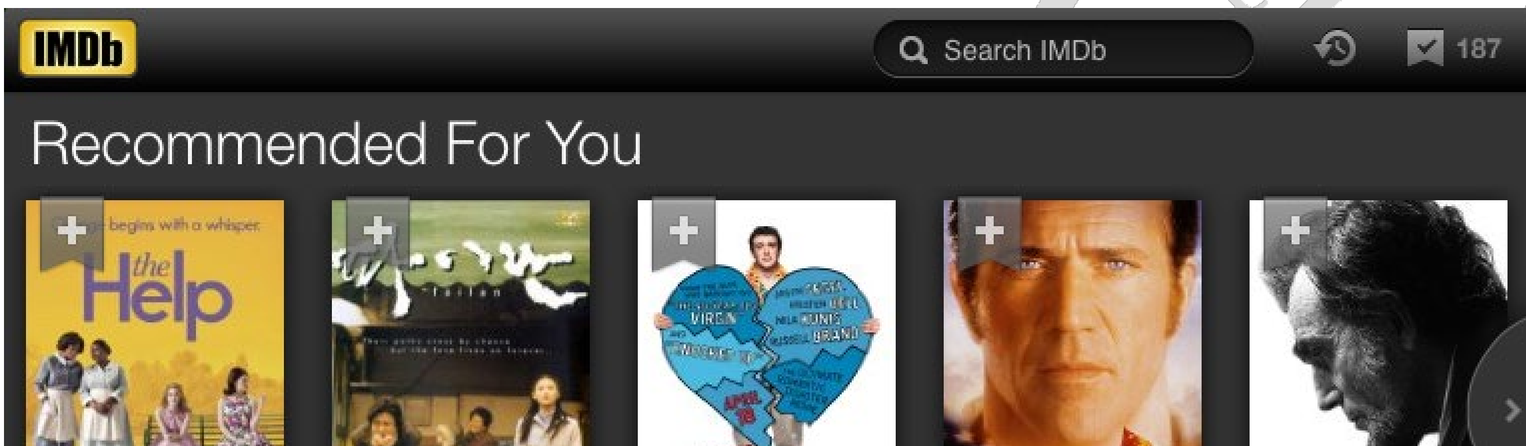
- Search – Information Retrieval



  - Know what you want
  - Query using key words
  - Return expected results
  - You find something!

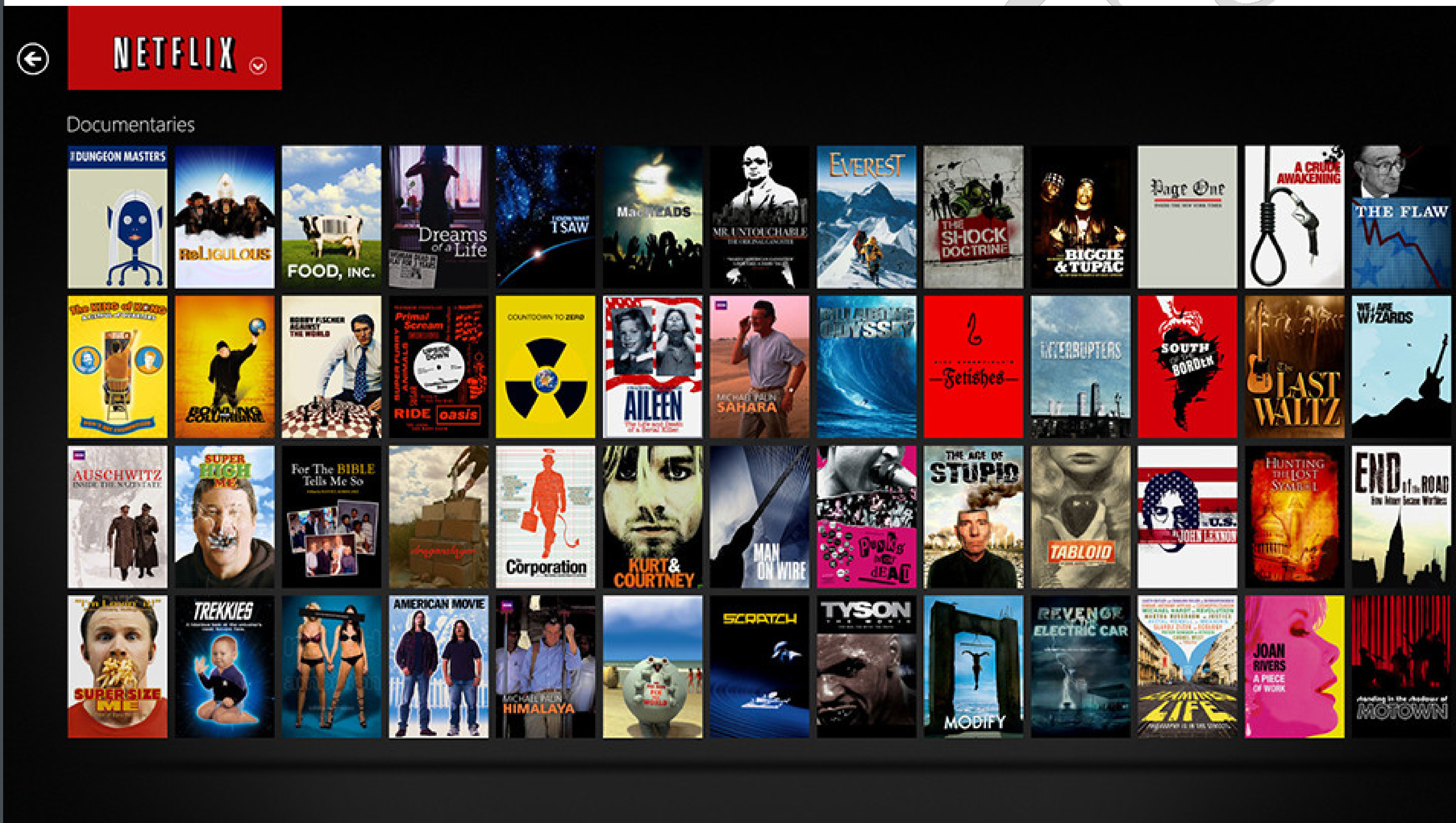
Search vs Recommendation

- Recommendation – Information Discovery



  - Do not know its existence
  - Do not know how to find
  - Return serendipitous results
  - Something finds you!

Movie Recommendation

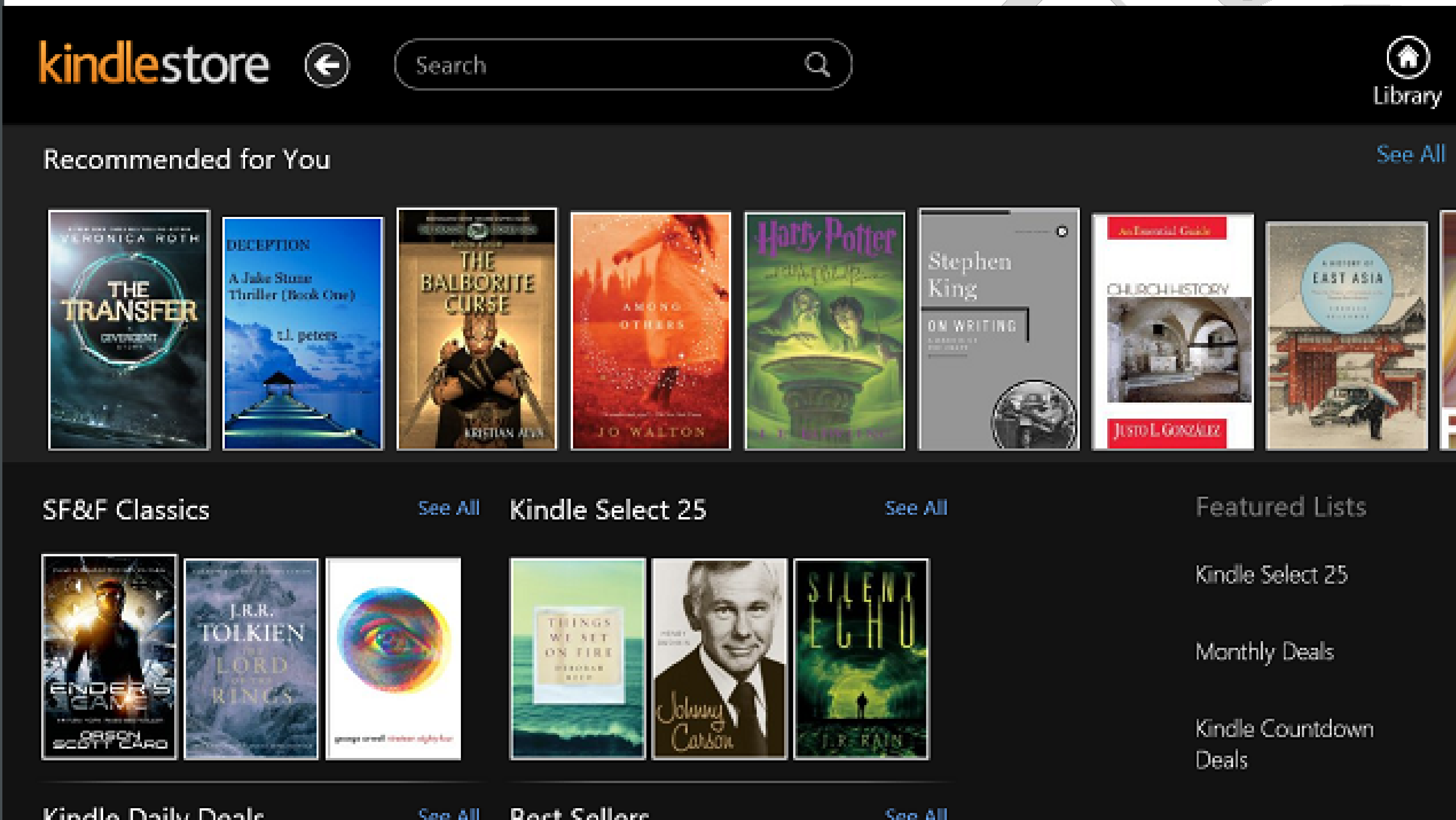


Netflix Prize

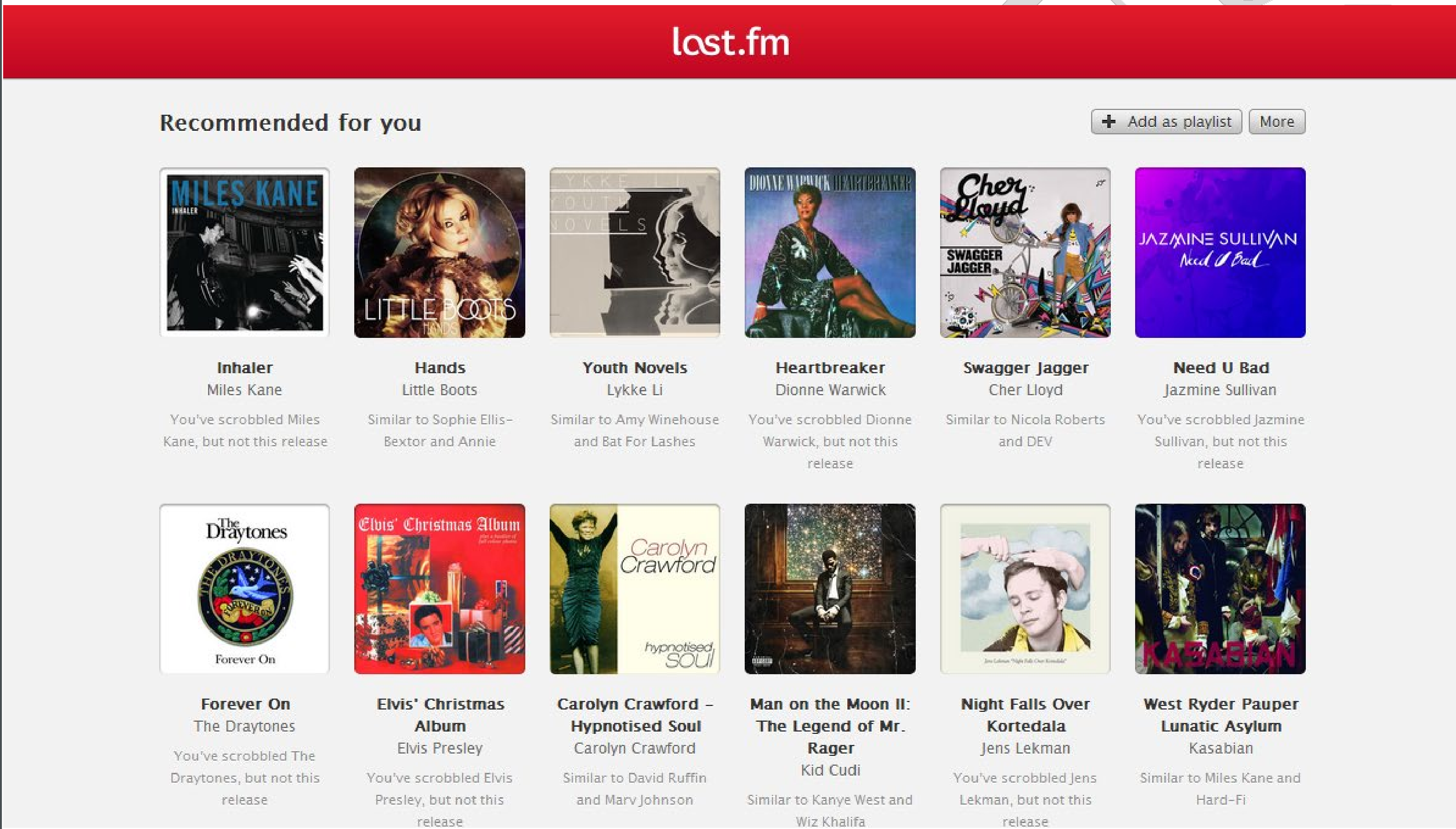
- October 2006, Netflix offered a $1,000,000 Grand Prize
- The grand prize accelerated the research of recommendation
- The winning team uses machine learning techniques

Outperforms “Cinematch” by 10%

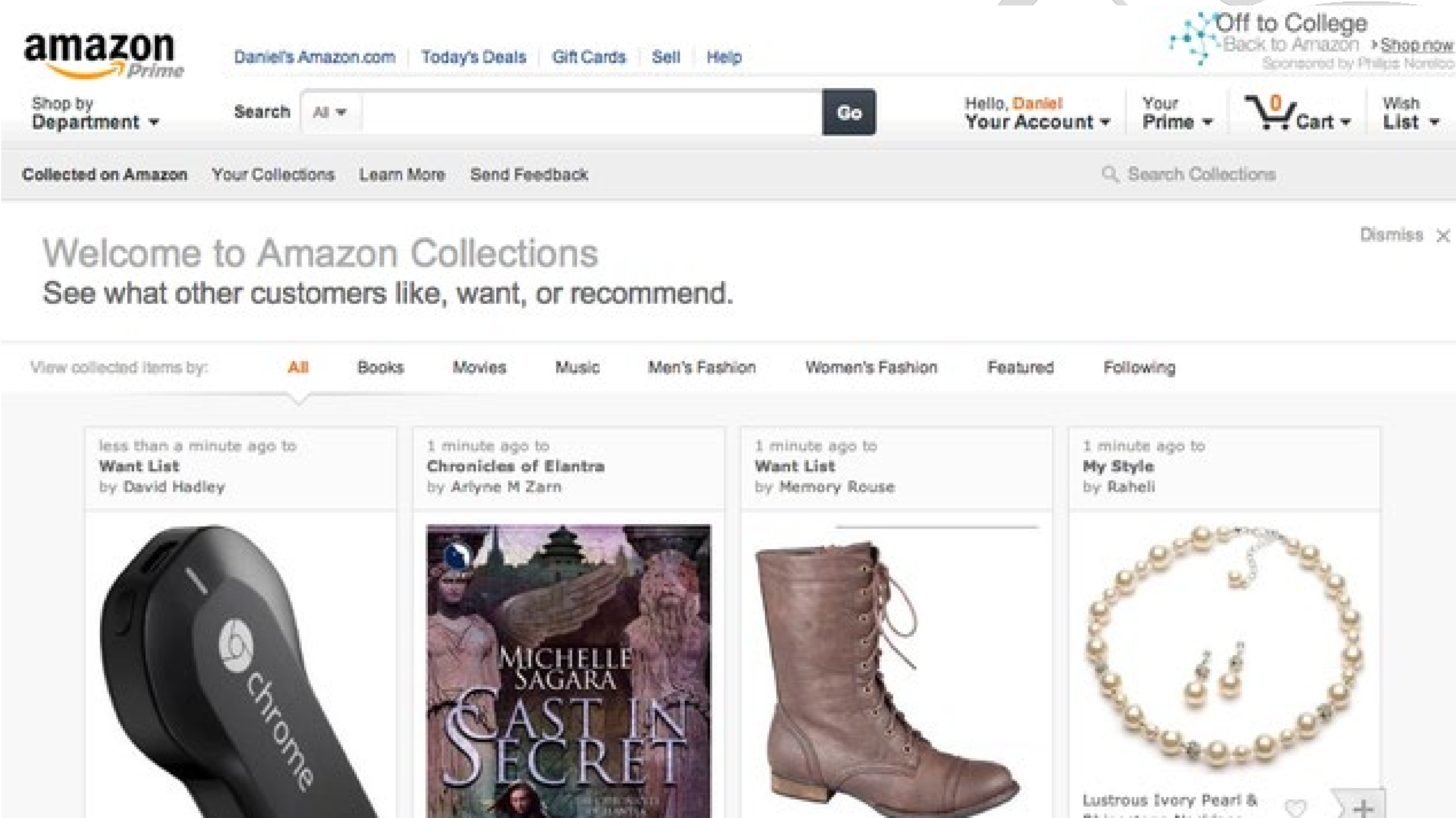
Book Recommendation



Music Recommendation



Product Recommendation



Omnipresent Recommendations

Linkedin Jobs

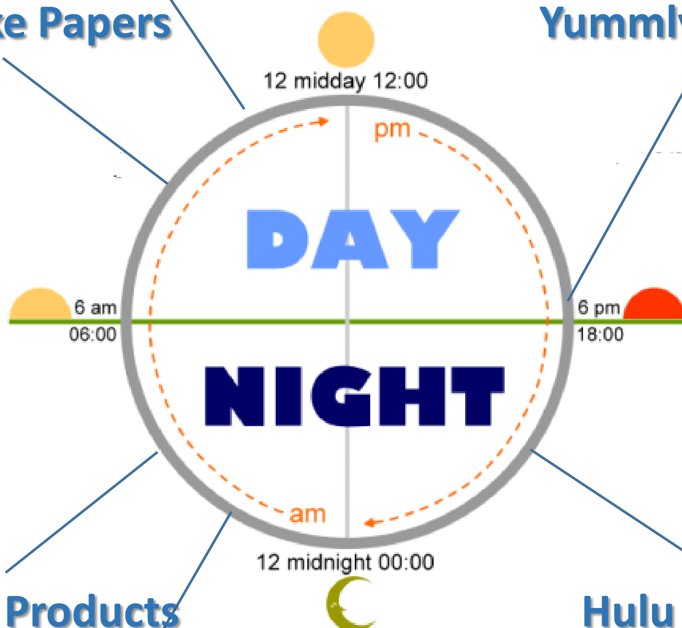
Flipboard News

TripAdvisor Hotels

CiteULike Papers

Yummly Recipes

Qunar Itinerary



Amazon Products

Hulu Videos

Foursquare Venues

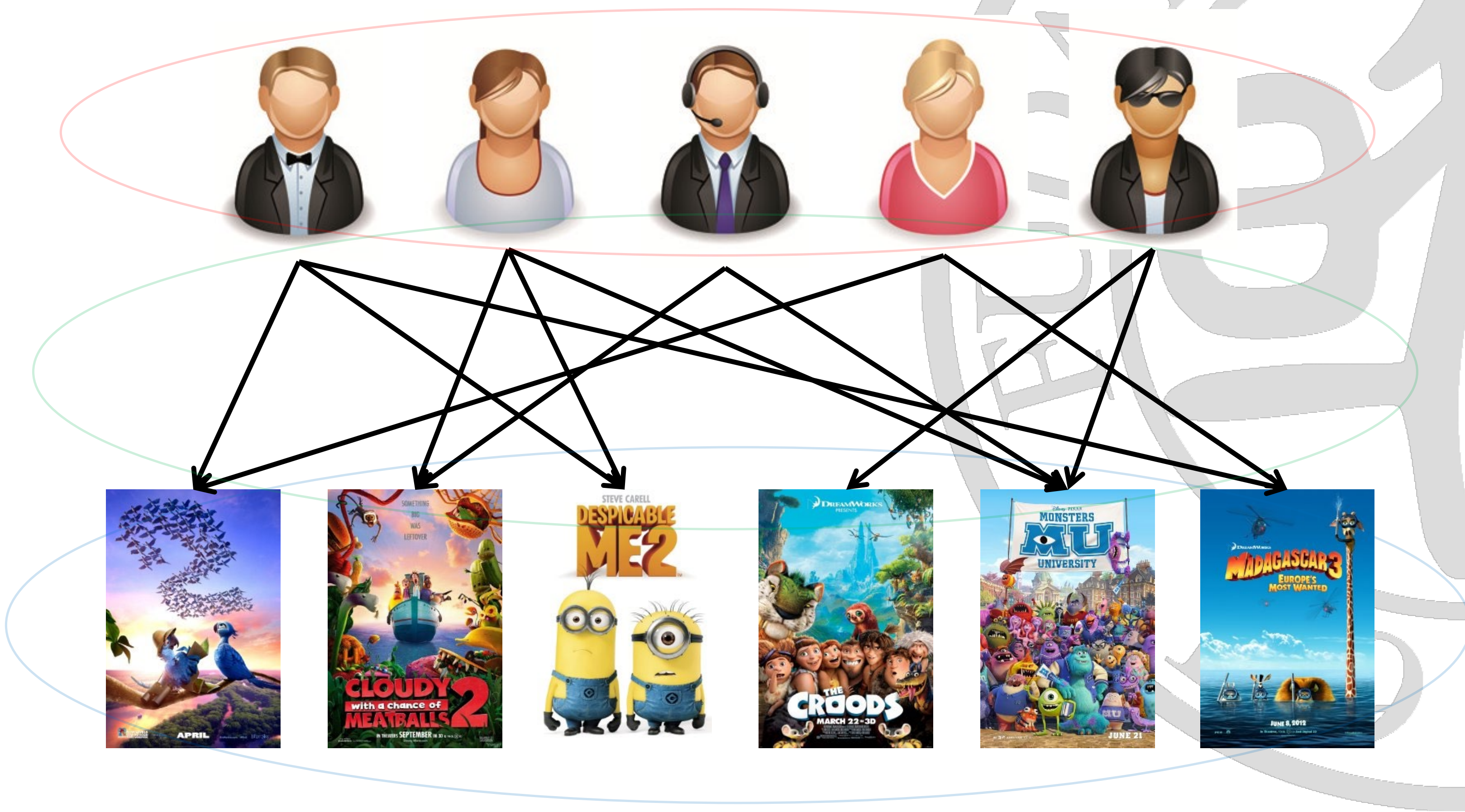
Netflix Movies

Kindle Books

Last.fm Music

Recommendation Problem

- Three key elements



Recommendation Problem

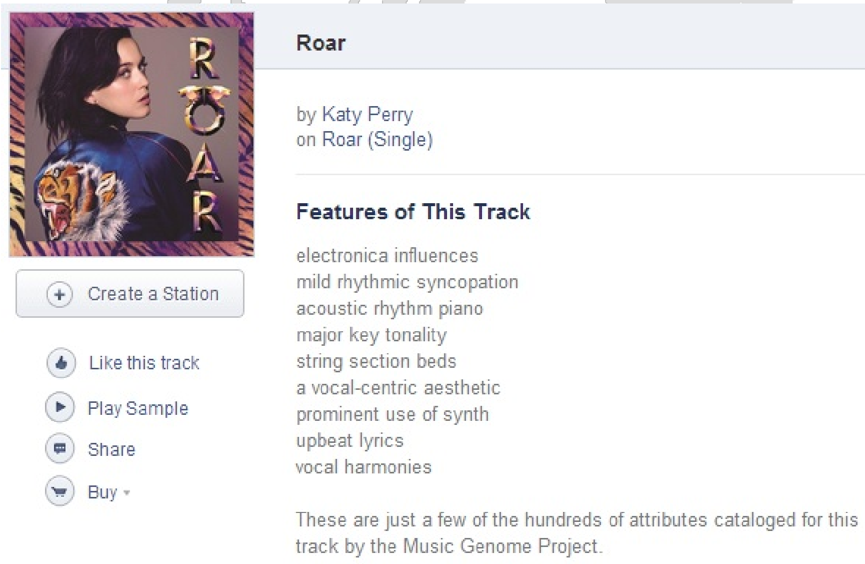
- User profiles
  - Basic: Genders, Ages, Occupations, Regions, etc.
  - Extra: Social relationships, User Tags, etc.

Male; Age 28; IT Engineer; US CA94035
[Tags] Travel, Steve Jobs, Photography, “TBBT”, …

Female; Age 20; Accounting; AU NSW2007
[Tags] Music, Lady Gaga, Katy Perry, “Gossip Girl”, …

Recommendation Problem

- Item attributes
  - Basic: Any form of descriptive data (e.g., movie metadata)
  - Extra: Item taxonomy, knowledge base (e.g., Wikipedia)



Recommendation Problem

- Preference (explicit)
  - Ratings
  - Likes
- Preference (implicit)
  - Click-through
  - Purchased records

4		5			3
	3	4		3	
	3			4	
4					4
			2	5	

Recommendation Problem

- Given user set
  - User profiles – optional
- Given item set
  - Item attributes – optional
- Given preference
  - Explicit/Implicit preference data – mandatory

- Real-world RSs tend to make full use of available data
- The most basic RS problem only use preference data – focus of the ML research for RS

Recommendation Problem

- Goal
  - Predict ratings
  - Rank items

4					3
	3	4		3	
?	3	?	?	4	?
4		5			4
			2	5	

RS Problem Example: MovieLens

- UserID::Gender::Age::Occupation::Zip (user info file format)
  - Age is chosen from 7 ranges: * 1: "Under 18" * 18: "18-24" * 25: "25-34" * 35: "35-44" * 45: "45-49" * 50: "50-55" * 56: "56+"
  - Occupation is chosen from 20 choices: * 0: "other" or not specified * 1: "academic/educator" * 2: "artist" * 3: "clerical/admin" * 4: "college/grad student" * 5: "customer service" * 6: "doctor/health care" * 7: "executive/managerial" * 8: "farmer" * 9: "homemaker" * 10: "K-12 student" * 11: "lawyer" * 12: "programmer" * 13: "retired" * 14: "sales/marketing" * 15: "scientist" * 16: "self-employed" * 17: "technician/engineer" * 18: "tradesman/craftsman" * 19: "unemployed" * 20: "writer"
- MovieID::Title::Genres (movie info file format)
  - Titles are provided by the IMDB (including year of release)
  - Genres are selected from 18 genres: * Action * Adventure * Animation * Children's * Comedy * Crime * Documentary * Drama * Fantasy * Film-Noir * Horror * Musical * Mystery * Romance * Sci-Fi * Thriller * War * Western

[1] Download at http://grouplens.org/datasets/movielens/

RS Problem Example: MovieLens

- UserID::MovieID::Rating::Timestamp
  - Ratings in 5-star scale {1,2,3,4,5}
  - Timestamp is represented in seconds (can be transformed into dd-mm-yyyy)

Training Data

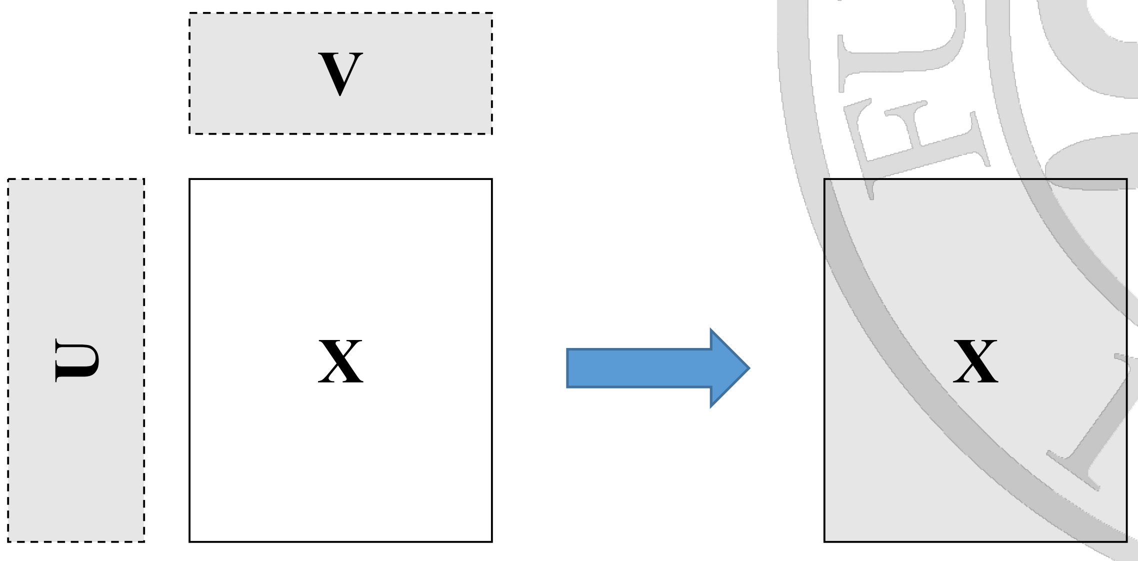
user	movie	date	rate
1	34	11-04-02	3
1	296	09-05-02	4
2	11	18-01-02	5
2	59	23-02-02	4
2	124	03-04-02	2
3	58	05-07-02	3

Test Data

user	movie	date	rate
1	75	21-02-03	?
1	126	09-03-03	?
2	92	18-01-03	?
2	257	29-05-03	?
3	66	22-03-03	?
3	394	02-06-03	?

RS Problem Formalization

- Given a User-Info Matrix (optional): **U**
- Given an Item-Info Matrix (optional): **V**
- Given a User×Item partially observed Preference Matrix: **X**
- Complete the missing entries in **X**



RS Categorization

- Data Perspective
  - Demography-based (rely on user profiles) ☆
  - Content-based (rely on item attributes) ☆
  - **Collaborative Filtering based (rely on preference) ★**
  - Hybrid

- Method Perspective
  - Rule-based (database approach)
  - **Memory-based (information retrieval approach) ★**
  - **Model-based (machine learning approach) ★**
  - Hybrid

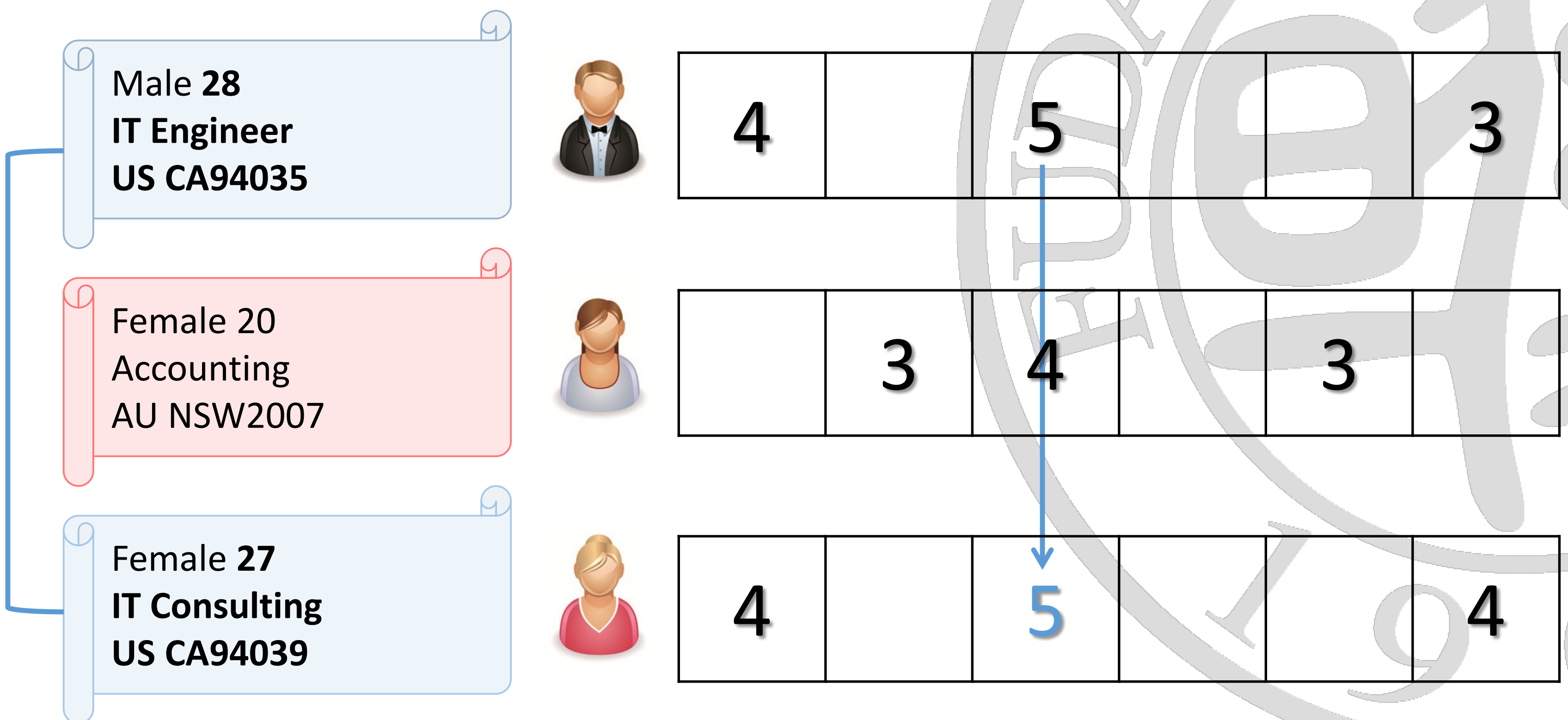
Real-world RSs

- Real-world RSs are usually Hybrid
  - Combine multiple recommendation strategies in different scenarios
  - Mainly based on CF techniques with rule-based and content-based as complementary strategies

- Amazon combines demography-based, Content-based, and CF-based strategies
  - User demographic info
  - User purchased records, click-through histories, etc.
  - Item attributes, item taxonomy
  - Item popularities

Demography-based (Brief Intro)

- User correlation by comparing demographic info
- Recommend items from highly correlated users

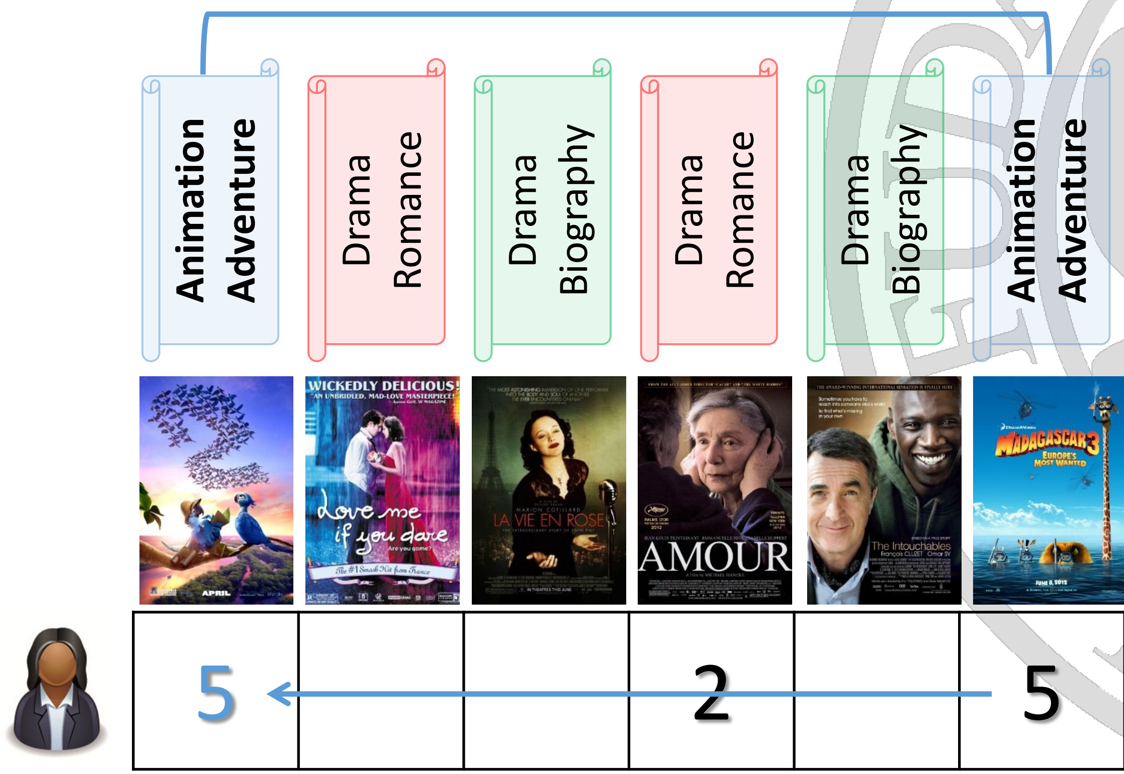


Demography-based (Brief Intro)

- Require User-Info Matrix and Preference Matrix
- Advantages
  - Domain-independent (cross item-domain recommendations)
  - No cold-start problem (not rely on historical preference data)
- Disadvantages
  - Coarse and inaccurate to model preference
  - Demographic data may be incomplete

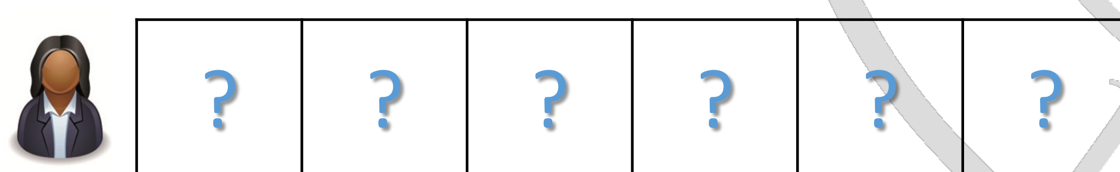
Content-based (Brief Intro)

- Item correlation by comparing item content
- Recommend items highly correlated to historical preference



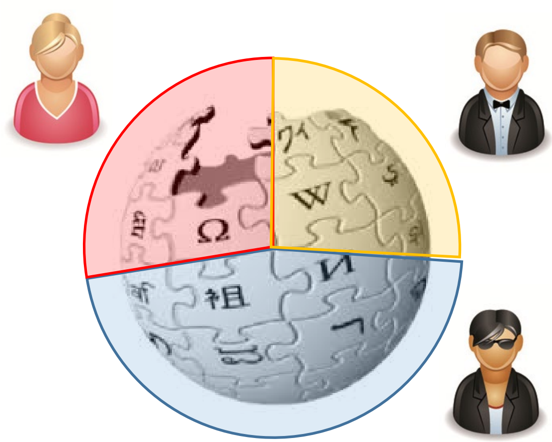
Content-based (Brief Intro)

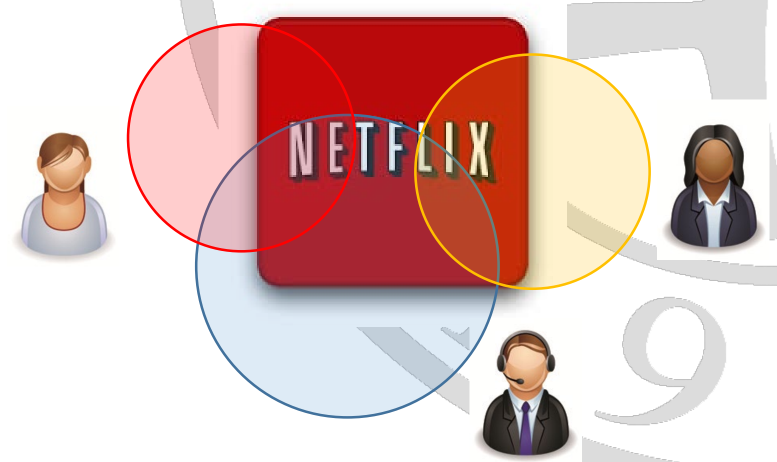
- Require Item-Info Matrix and Preference Matrix
- Advantages
  - Fine and accurate to model preference
  - Tags are effective if provided
- Disadvantages
  - Rely on item attributes (complete and comprehensive)
  - Cold-start problem (new users have no historical data)



Collaborative Filtering (Overview)

- Web 2.0 emphasizes user participation and contributions
  - Tags (Flickr), Articles (Wikipedia), Reviews (Amazon), etc.
- Collective Intelligence (CI)
  - Making use of the union of individual contributions
- Collaborative Filtering (CF) is CI
  - But focus on discovering intersected individual contributions

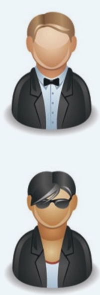




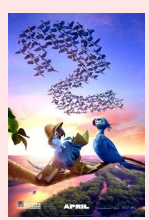
Collaborative Filtering (Overview)

- Main idea of CF
  - Find neighbors based on historical preference – **How to decide?**
  - Recommend items highly rated by neighbors – **How to rank?**

Neighbors?



4		5			3
			2	5	

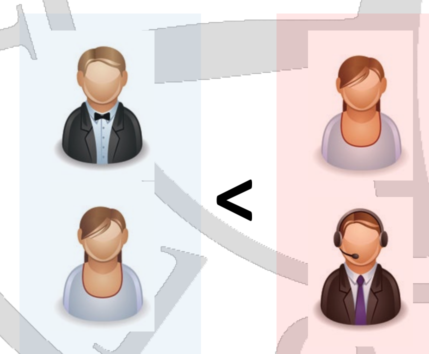


> ? <

Collaborative Filtering (Overview)

- User-based Collaborative Filtering (similar users)
- User-based vs. Demography-based
  - Demography-based uses user-info to compute similarity
  - User-based uses historical preference data to compute similarity

4		5			3
	3	4		3	
	3			4	



Collaborative Filtering (Overview)

- Item-based Collaborative Filtering (similar items)
- Item-based vs. Content-based
  - Content-based uses item-info to compute similarity
  - Item-based uses associated preference data to compute similarity

		5
3	3	4
3	4	
	5	

Collaborative Filtering (Overview)

- Both user-based and item-based are “memory-based”
  - User-based has long history
  - Item-based was invented by Amazon as an improvement of user-based
- User-based vs Item-based – How to choose?
  - It depends …

User-based CF	Item-based CF
item # < user #	item # > user #
Items change rapidly	Items stay stable
News RS	Product RS (e.g., Amazon)

Collaborative Filtering (Overview)

- Model-based (compared to memory-based)
  - Using ML models for preference-matrix completion
  - Recommend items based on the estimated ratings
- Matrix Factorization Approach
  - Singular Value Decomposition (SVD)
  - SVD variants
  - Bayesian Probabilistic Matrix Factorization
- Mixture Model Approach
  - Flexible Mixture Models
  - Bi-LDA (variant of Latent Dirichlet Allocation)

Collaborative Filtering (Overview)

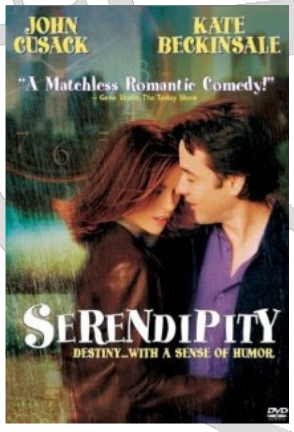
- CF is the most widely used recommendation mechanism
- Advantages
  - Only based on historical preference data
  - Domain independent (model not specific to certain item domains)
  - Well defined ML problem (numerous ML methods can be applied)
- Disadvantages – Challenges
  - Cold-start problem (new user has no preference data)
  - Sparsity problem (preference matrix is very sparse)
  - Noise problem (rely on the quality of preference data)

Hybrid Strategies

- Weighted Hybridization
  - Combine weighted results of multiple recommenders to generate a final recommendation
- Switching Hybridization
  - Switch between different recommenders depending on situations
- Mixed Hybridization
  - Show results of different recommenders at different locations on a webpage
- Cascade Hybridization
  - Refine the result of another recommender from coarse to fine

Recommendation Criteria

- Personalization
  - Relevance to user's tastes
- Diversity
  - Coverage of user's multi-aspect tastes
- Serendipity
  - Exploration of user's new tastes



Recommendation Performance

- Rating Prediction (regression problem)
  - Measure the difference between predictions and ground-truths
- Evaluation Metrics
  - Mean Absolute Error (MAE) , Root Mean Squared Error (RMSE)

$$\mathrm{MAE} = \frac{1}{N} \sum_{n=1}^{N} | r_n - \hat{r}_n |$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N} \sum_{n=1}^{N} (r_n - \hat{r}_n)^2}$$

Thanks

Email: libin@fudan.edu.cn