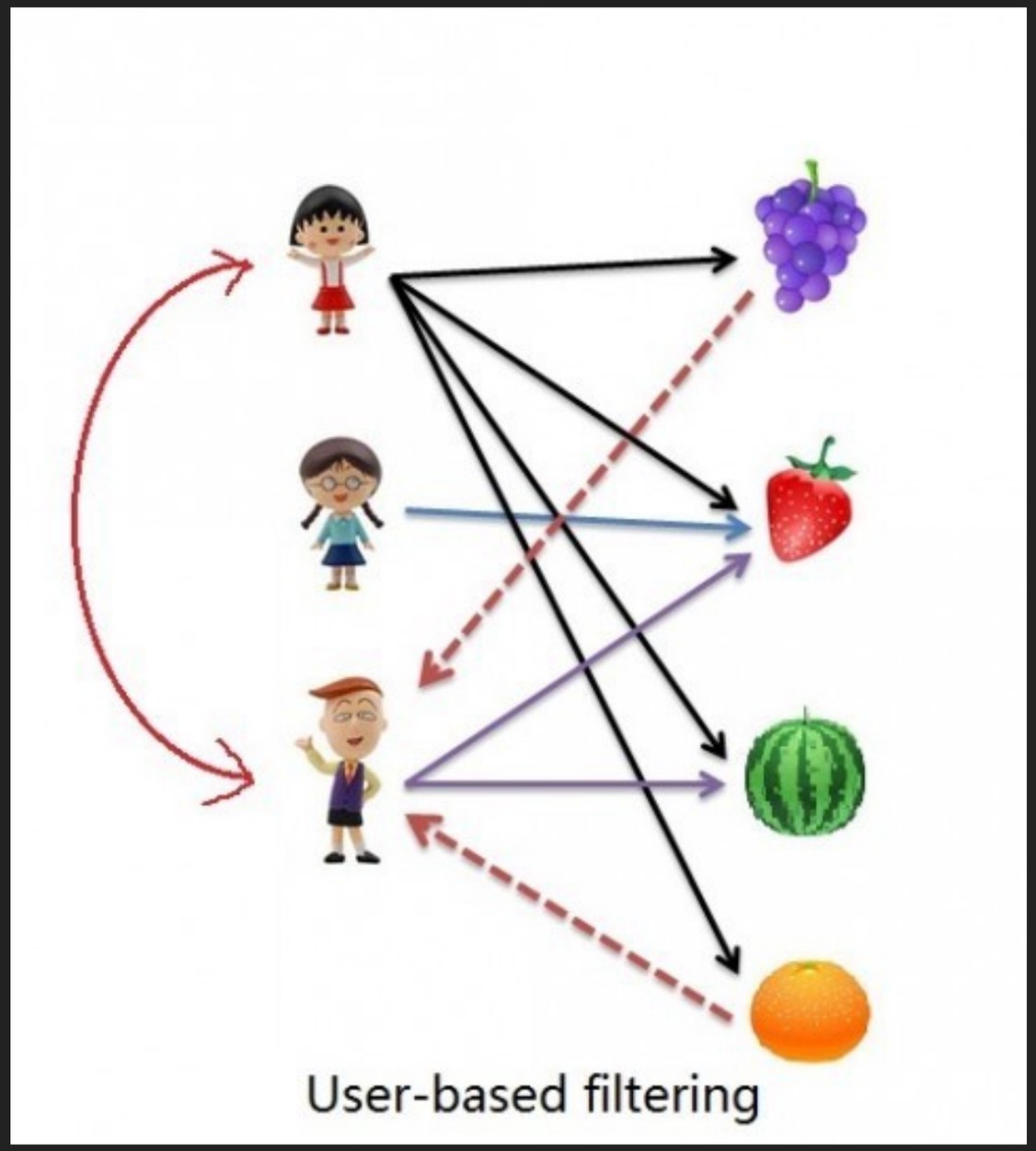
CS 425 ALGORITHMS FOR WEB SCALE DATA

RECOMMENDATION SYSTEM WITH COLLABORATIVE FILTERING

ABOUT DATASET

- This dataset was created by [MovieLens](http://movielens.org). It contains 20,000,263 ratings across 27,278 movies. These data were created by 138,493 users between January 09, 1995 and March 31, 2015.



USER-USER COLLABORATIVE FILTERING

RATING NORMALIZATION

- Find average rating given by a user
- Subtract average value from ratings

- Example of some ratings of user 6: (avg. of ratings = 3.75)
- {Movie id: rating}
- [[1: 5.0, 3: 3.0, 7: 5.0, 17: 5.0, 52: 5.0, 62: 5.0,...]]
- [[1: 1.25, 3: -0.75, 7: 1.25, 17: 1.25, 52: 1.25, 62: 1.25, ...]]

USER-GENRE MATRIX

- For each genre find all ratings given by user to that genre
- Get average ratings for each genre

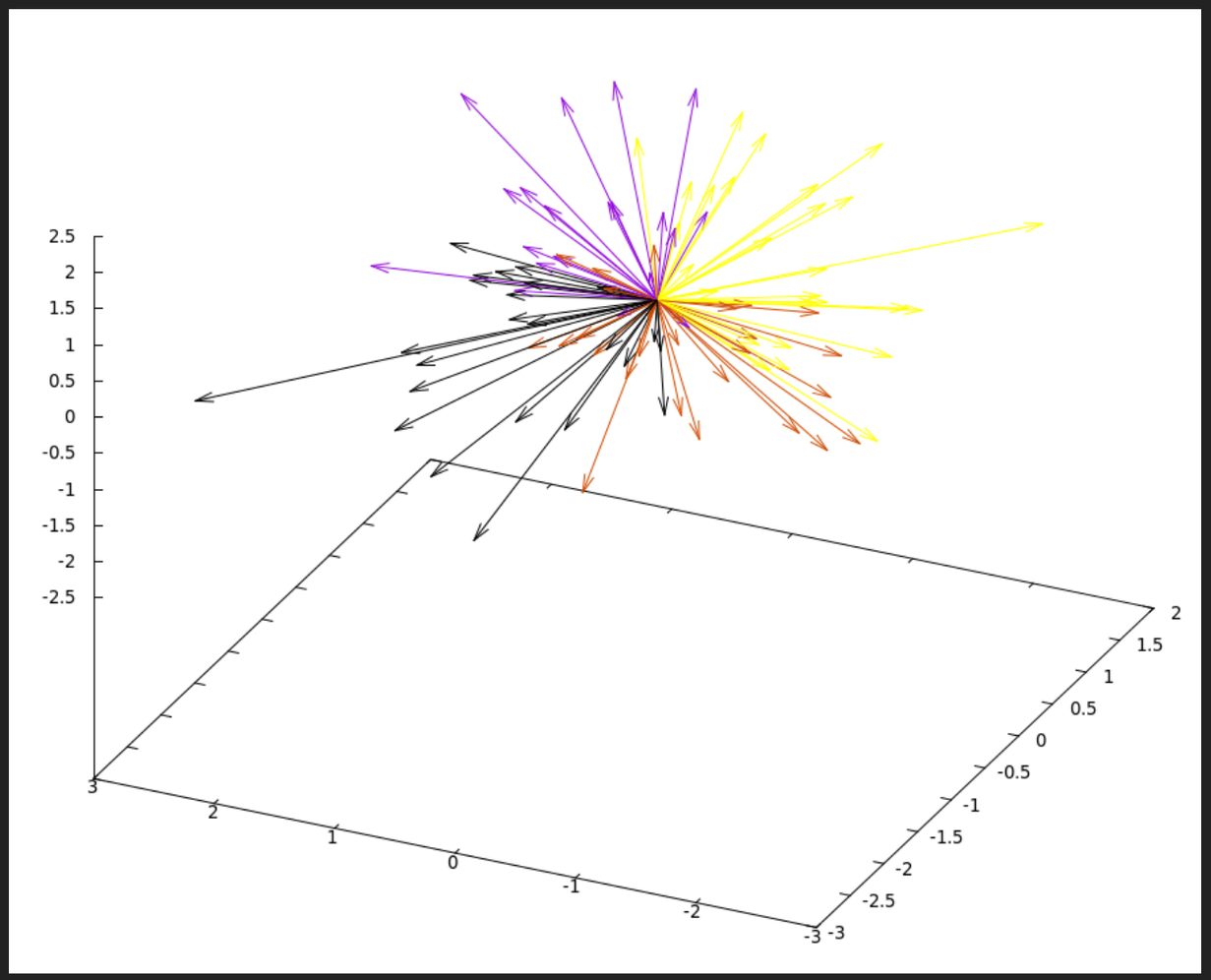
- Example of some ratings to each genre of user 6:
- [{Film-Noir=0.0, Action=-0.083333336, Adventure=0.10714286, Horror=0.0, Romance=0.10714286, War=0.0, Western=0.0, Documentary=0.0, Sci-Fi=-0.25, Drama=0.5833333, Thriller=-0.15, (no genres listed)=0.0, Children's=0.0, Crime=-0.25, Fantasy=-0.41666666, Animation=1.25, Comedy=-0.30555555, Mystery=0.75, Musical=-2.75}]
-

LSH – USER GENRE RATINGS

- Compute the signature of user's genre ratings vectors.
- These signatures are used to quickly estimate the similarity between vectors.
- This implementation is based on Locality Sensitive Hashing (LSH), as described in Leskovec, Rajaraman & Ullman (2014), "Mining of Massive Datasets", Cambridge University Press.

LSH – SUPERBIT

- Super-Bit is an improvement of Random Projection LSH.
- It computes an estimation of cosine similarity.
- Super-Bit Locality-Sensitive Hashing, Jianqiu Ji, Jianmin Li, Shuicheng Yan, Bo Zhang, Qi Tian http://papers.nips.cc/paper/4847-super-bit-locality-sensitive-hashing.pdf Published in Advances in Neural Information Processing Systems 25, 2012



LSH – STEPS

- Compute the SUPERBIT signature of user’s genre ratings vectors.
- Hash a signature. The signature is divided in 31 bands. Each stage is hashed to one of the 744 (2 * √numofusers) buckets. Then, put the id of the user to the bucket.

FIND SIMILAR USERS FROM BUCKETS

- use the LSH like the previous slide and get hashed bucket of the current user’s genre vector
- get the users list from the bucket
- find movies which have ratings greater than 0.75 and check they have been watched by user or not. If not, then add the movie to the users recommendation movies list.
-

SHOW RECOMMENDED MOVIES RANDOMLY

- Example of some recommendations to user 6:
- Number of recommendable movies = 1608
- Some recommendable movies are following for user 6.
  - 858, “Godfather
  - 1120, "People vs. Larry Flynt
  - 1210, Star Wars: Episode VI - Return of the Jedi (1983)
  - 110, Braveheart (1995)
  - 145, Bad Boys (1995)
  - 163, Desperado (1995)