6-4: Item-Item Hybrids and Extensions

Introduction

- We've now introduced item-item CF

- Item-item CF is very flexible

- This time: hybrids and extensions

Hybrid Recommenders

- Combine 2 or more algorithms

- General technique, not specific to any one algorithm

- Similar to stacking or boosting in machine learning

Hybrid Techniques

- Weighting – combine algorithm scores
  - Can be extended with feature-weighted coefficients
- Switching – switch algorithms
- Mixed – mix output from diff. algorithms
- Use one algorithm as input to another
- See Burke reading for more discussion

Feature-Weighted Linear Stacking

- Winning NetFlix Prize algorithm
- Linear combination of 100+ algorithms
- Coefficients depend on user/item features
  - Relative weight of algorithms shifts for different recommendation contexts

Extending Item-Item

- Item-item is good for extending directly
- Simple parts with well-defined interfaces provide a lot of flexibility
- It's easy to understand what extensions do

Example: User Trust

- Goal: incorporate user trustworthiness into item relatedness computation
  - User's global reputation, not per-user trust
- Solution: weight users by trust before computing item similarities
- High-trust users have more impact
- Massa and Avesani. 2004. 'Trust-Aware Collaborative Filtering for Recommender Systems'

Extension: Papers and PageRank

- Recommending research papers: useful to consider items as users who purchase the paper's citations
  - Same idea can apply to web pages
- Goal: incorporate paper 'importance' into recommender
- Solution: weight paper user vectors by the paper's PageRank (or HITS hub score)
- Ekstrand et al., 2010. Automatically Building Research Reading Lists.

Restructuring: Item-Item CBF

- Basic item-item algorithm structure doesn't care how similarity is computed
- So why not use content-based similarity?
- Resulting algorithm really isn't a collaborative filter
- But it can work pretty well!
- Example: using Lucene to compare documents as neighborhood & similarity function

Restructuring: Deriving Weights

- Item-item compares individual item pairs
- Alternative approach: infer coefficients from data
  - Find coefficients $w_{i,j}$ that minimize RMSE
  - Learn coefficients with standard machine learning / optimization algorithm (gradient descent)
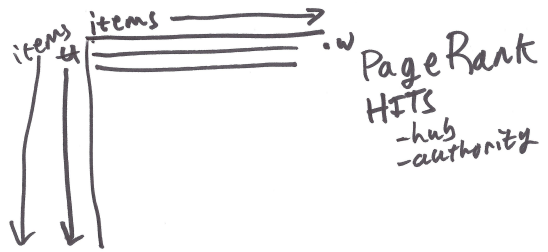
Conclusion

- Single algorithms aren't only recommender solution
  - Many deployed algorithms are hybrids
- Item-item CF is flexible and versatile
- Many interesting recommenders can be built by reconfiguring it

6-4: Item-Item Hybrids and Extensions

items items →

items
↑
↓ ↓ |———————— .w

Page Rank

HITS
- hub
- authority

$$p(u,i) = a_1 P_1(u,i) + a_2 P_2(u,i) \ldots + b$$

$$p(u,i) = f_1(u,i) P_1(u,i) + f_2^{(u,i)} P_2(u,i) \ldots + b$$