Unified Mathematical Model

Objectives

- Develop a unified model and notation for
  - Rating prediction (scoring)
  - Recommendation
- Integrate this with search and context-aware computing
- Answer: how does search relate to recommendation?

Review of Recommender Tasks

We've looked at 2 key tasks:

- *Predict* how much a user will like an item
- *Recommend* items a user might like

Often, recommendation is top-predicted items

Scoring Items

Mathematically, we can think of prediction as a *scoring* function:

$$s(i; u)$$

$s(u, i)$

Computes the *score* of item i for user u.

This is the heart of many recommenders.

Expanding Scoring

What about

- Current context (at the theater, 11:30 AM on the streetcorner)
- Query terms (search)

Question often arises: how does search relate to recommendation?

Full Scoring Function

$$s(i; u, q, x)$$

i: the item to score

u: the active user

q: the user's query

x: the current context

Different systems use different variables

Full Scoring Function

$s(i; u, q, x)$	traditional recommender
$s(i; u, q, x)$	traditional search
$s(i; u, q, x)$	personalized search
$s(i; u, q, x)$	context-aware recommender
$s(i; u, q, x)$	context-aware pers. search (Google, Bing)

Computing s

- Much of what we do is compute s
- Content-based filtering: compute from user taste profile
- Demographic: compute from user demographics + segmented preferences
- Association rules: compute from context of currently-displayed item
- Collaborative: compute from user preferences and community preferences

Scoring to Recommendation

Likewise, we can define an *ordering* function to produce recommendation lists:

$$O(I; u, q, x)$$

Like s, but takes a set of items I and orders them instead of scoring a single item.

Basic Top-*N* Recommendation

$O(I; u, q, x)$ is defined by:

- Score each item $i \in I$ using $s(i; u, q, x)$
- Sort items in decreasing order of score
- Truncate after n items

Tweaking Top-*N* Recommendation

Variations of $O(I; u, q, x)$ may choose other orders:

- Diversity top-N to avoid too much similarity
- Re-prioritize top-N to promote high-value items

Extended Recommendation

An ordering may also depend on the number of items desired:

$$O(n, I; u, q, x)$$

Some recommenders may produce different top-5 and top-10 lists!

Wrap-up

- Recommendation can conceptually integrate with search and context-awareness
- We'll be using this notation throughout
- Rest of specialization will have more ways of computing these functions
  - Individual algorithms
  - Hybrids compose s and O from subsidiary scoring and ordering functions

Unified Mathematical Model