Refining searches

1

Refine initially: query

- Commonly, query expansion
  - add synonyms
    - Improve recall
    - Hurt precision?
    - Sometimes done automatically
  - Modify based on prior searches
    - Not automatic
    - All prior searches
      vs
    - *your* prior searches
    - Example: Yahoo
      - Google does too

2

Refining after search

- Use user feedback
  or
- Approximate feedback with first results
  - Pseudo-feedback
  - Example: “Yahoo assist”

- change ranking of current results
  or
- search again with modified query

Explicit user feedback

- User must participate

- User marks (some) relevant results
  or
- User changes order of results
  - Pros and cons?

Explicit user feedback

- User must participate

- User marks (some) relevant results
  or
- User changes order of results
  - Can be more nuanced than relevant or not
  - Can be less accurate than relevant or not
    - Example: User moves 10th item to first
      - says 10th better than first 9
      - Does not say which, if any, of first 9 relevant

5

User feedback in classic vector model

- User marks top p documents for relevance

  p = 10 to 20 “typical”
- Construct new weights for terms in query vector
  - Modifies query
  - Could use just on initial results to re-rank

6

Deriving new query for vector model

For collection C of n doc.s

- Let C_r denote set all relevant docs in collection,

Perfect knowledge Goal:

Vector $\mathbf{q}_{opt}$ =

$1/|C_r|$ * (sum of all vectors d_j in C_r) -

$1/(n- |C_r|)$ * (sum of all vectors d_k not in C_r)

centroids

7

Continuing: Deriving new query for vector model

Give query **q** and relevance judgments for a subset of retrieved docs

- Let D_r denote set of docs judged relevant
- Let D_{nr} denote set of docs judged not relevant

Modified query:

Vector $\mathbf{q}_{new} = \alpha\mathbf{q}$ +

$\beta/|D_r|$ * (sum of all vectors d_j in D_r) -

$\gamma/(|D_{nr}|)$ * (sum of all vectors d_k in D_{nr})

For tunable weights α, β, γ

8

Remarks on new query

- α: importance original query
- β: importance effect of terms in relevant docs
- γ: importance effect of terms in docs not relevant

- Usually terms of docs not relevant are least important
  - Reasonable values α=1, β=.75, γ=.15
- Reweighting terms leads to long queries
  - Many more non-zero elements in query vector $\mathbf{q}_{new}$
  - Can reweight only most important (frequent?) terms
- Most useful to improve recall
- Users don't like: work + wait for new results

9

Simple example user feedback in vector model

- **q** = (1,1,0,0)
- Relevant: **d1** = (1,0,1,1)
  d2 = (1,1,1,1)
- Not relevant: **d3**=(0,1,1,0)
- α, β, γ = 1
- $\mathbf{q}_{new}$ = (1,1,0,0) + (1, 1/2, 1, 1) - (0,1,1,0)
  = (2, 1/2, 0, 1)

Term weights change New term

Observe: Can get negative weights

10

Re-ranking

- Examples - status?
  - Google experiment
  - Wikia search
- Algorithms usually based on machine learning
  - Learn ranking function that best matches partial ranking given

11

Comparing rankings

- Kendall Tau measure compares two orderings, rk_1 and rk_2, of the same set of m objects:
  $\tau(rk_1,rk_2)$
  - Let A = # pairs whose orders agree in rk_1 and rk_2
  - Let I = # pairs whose orders disagree in rk_1 and rk_2: inversions

  $A + \mathrm{I}$ = (m choose 2)

$\tau(rk_1, rk_2) = (A- \mathrm{I})/(A+ \mathrm{I})$ Ordinal correlation

$= 1 - (2\, \mathrm{I}$ / (m choose 2))

12

Implicit user feedback

- Click-throughs
  - Use as relevance judgment
  - Use as reranking:
    When click result, moves it ahead of all results didn't click that come before it
  - Problems?
- Better?

13

Single user feedback vs group

- Compare Recommender Systems
  - Items
  - Users
  - Recommend Items to Users
    - Consider documents matching search as items?
- Items have characteristics
- Recommend new items based on similarity to items that:
  - User liked in past: Content-based
  - Liked by other users similar to this user: Collaborative Filtering
    - Similarity- characteristics of shared liked items

14

Recommender System attributes

- Need explicit or implicit ratings by user
  - Purchase is 0/1 rating
    - Movie tickets
    - Books
- Have focused category
- User can give explicit preferences for certain characteristics
- Content-based: much like vector-based IR
- Collaborative filtering: need items already rated by user *and other users*

15

Limitations of Content-based

- Can only recommend items similar to those user rated highly
- New users
  - Insufficient number of rated items
- Only consider features explicitly associated with items
  - Do not include attributes of user

16