Web Query Disambiguation from Short Sessions

Lilyana Mihalkova* and Raymond Mooney

University of Texas at Austin

*Now at University of Maryland College Park
Web Query Disambiguation

scrubs
Existing Approaches

• Well-studied problem:
  – [e.g., Sugiyama et al. 04, Sun et al. 05, Dou et al. 07]

• Build a user profile from a long history of that user’s interactions with the search engine
Concerns

• Privacy concerns
  – AOL Data Release
  – “Googling Considered Harmful” [Conti 06]

• Pragmatic concerns
  – Storing and protecting data
  – Identifying users across searches
Proposed Setting

• Base personalization only on short glimpses of user search activity captured in brief short-term sessions

• Do not assume across-session user identifiers of any sort
How Short is Short-Term?

Number of sessions with that many queries vs. Number of queries before ambiguous query
Is This Enough Info?

98.7 fm
www.star987.com

kroq
www.kroq.com

scrubs
scrubs-tv.com

huntsville hospital
www.huntsvillehospital.com

ebay.com
www.ebay.com

scrubs
scrubs.com
More Closely Related Work

• [Almeida & Almeida 04]: Similar assumption of short sessions, but better suited for a specialized search engine (e.g. on computer science literature)
Main Challenge

• How to harness this small amount of potentially noisy information available for each user?
  – Exploit the relations among sessions, URLs, and queries
Relationship are established by shared queries or clicks that are themselves related.

The choices users make over the set of possible results are also interdependent.

Query strings and URLs are related by sharing identical keywords, or by being identical.
Exploiting Relational Information

• To overcome sparsity in individual sessions, need techniques that can effectively combine effects of noisy relations of varying types among entities
  – Use statistical relational learning (SRL) [Getoor & Taskar 07]
  – We used one particular SRL model: Markov logic networks (MLNs) [Richardson & Domingos 06]
Rest of This Talk

• Background on MLNs
• Details of our model
  – Ways of relating users
  – Clauses defined in the model
• Experimental results
• Future work
Markov Logic Networks (MLNs)

- Set of weighted first-order clauses
- The larger the weight of a clause, the greater the penalty for not satisfying a grounding of this clause
  - Clauses can be viewed as relational features

[Richardson & Domingos 06]
MLNs Continued

• Q: set of unknown query atoms
• E: set of evidence atoms
• What is $P(Q|E)$, according to an MLN:

$$P(Q = q|E = e) = \frac{\exp \left( \sum_{f_i \in \mathcal{F}} w_i n_i(q, e) \right)}{\sum_{q'} \exp \left( \sum_{f_i \in \mathcal{F}} w_i n_i(q', e) \right)}$$

Normalizing constant

Number of satisfied groundings of formula $i$
MLN Learning and Inference

• A wide range of algorithms available in the Alchemy software package [Kok et al. 05]

• Weight learning: we used contrastive divergence [Lowd & Domingos 07]
  – Adapted it for streaming data

• Inference: we used MC-SAT [Poon & Domingos 06]
Re-Ranking of Search Results

• Hand-code a set of formulas to capture regularities in the domain
  - Challenge: define an effective set of relations
• Learn weights over these formulas from the data
  - Challenge: noisy data

MLN
Specific Relationships

- huntsville hospital
  - huntsvillehospital.org
- ebay
  - ebay.com
- scrubs
  - ???
  - ebay.com
- huntsville school
  - ...
  - hospitallink.com
- scrubs
  - scrubs.com
  - ...
  - ebay.com
  - scrubs-tv.com
  - ...
  - ebay.com
Collaborative Clauses

- The user will click on result chosen by sessions related by:
  - Shared click
  - Shared keyword click-to-click, click-to-search, search-to-click, or search-to-search
Popularity Clause

- User will choose result chosen by any previous session, regardless of whether it is related
  - Effect is that the most popular result becomes the most likely pick
Local Clauses

• User will choose result that shares a keyword with a previous search or click in the current session

Not effective because of data sparsity
Balance Clause

• If the user chooses one of the results, she will not choose another
  – Sets up a competition among possible results
  – Allows the same set of weights to work well for different-size problems
Empirical Evaluation: Data

• Provided by MSR
• Collected in May 2006 on MSN Search
• First 25 days is training, last 6 days is test

✘ Does not specify which queries are ambiguous
  • Used DMOZ hierarchy of pages to establish ambiguity
✘ Only lists actually clicked results, not all that were seen
  • Assumed user saw all results that were clicked at least once for that exact query string
Empirical Evaluation: Models Tested

• Random re-ranker
• Popularity
• Standard collaborative filtering baselines based on preferences of a set of the most closely similar sessions
  – Collaborative-Pearson
  – Collaborative-Cosine
• MLNs
  – Collaborative + Popularity + Balance
  – Collaborative + Balance
  – Collaborative + Popularity + Local + Balance
Empirical Evaluation: Measure

- Area under the ROC curve (mean average true negative rate)

\[
AUC-ROC(T) = \frac{1}{|T|} \sum_{t \in T} \frac{1}{|R_t|} \sum_{r \in R_t} TN@r
\]

\[
TN@r = \frac{\text{Num irrelevant docs in pos} > r}{\text{Total num irrelevant docs}}
\]

In Paper: Area under precision-recall curve, aka mean average precision
AUC-ROC Intuitive Interpretation

• Assuming that user scans results from the top until a relevant one is found,
• AUC-ROC captures what proportion of the irrelevant results are not considered by the user
Results

AUC-RO

Collaborative + Popularity + Balance

Random
Collaborative-Pearson
Collaborative-Cosine
Popularity
MLN
Difficulty Levels

0,45
0,47
0,49
0,51
0,53
0,55
0,57
0,59
0,61
0,63
0,65

Increasing Easiness

Proportion Clicked

Possible Results for a given query

Average Case

Popularity

KL

Possible Results for a given query

Increasing Easiness

Average

Possible Results for a given query
Difficult Levels

AUC-ROC

Increasing Easiness

MLN

Popularity

Worst Case
Future Directions

• Incorporate more info into the models
  – How to retrieve relevant information quickly

• Learn more nuanced models
  – Cluster shared clicks/keywords and learn separate weight for clusters

• More advanced measures of user relatedness
  – Incorporate similarity measures
  – Use time spent on a page as an indicator of interestedness
Thank you

• Questions?
First-Order Logic

- Relations and attributes are represented as predicates

- **WorkedFor**\((A, B)\)
  - **Predicate**
  - **Variable**
  - **Literal**

- **Actor**\((A)\)
  - **Predicate**
  - **Literal**

- **WorkedFor**(brando, coppola)
  - Ground Literal, “Grounding” for short
**Clauses**

- Dependencies and regularities among the predicates are represented as **clauses**:

\[ \text{Movie}(T, A) \land \neg\text{Director}(A) \implies \text{Actor}(A) \]

- To obtain a **grounding** of a clause, replace all variables with entity names:

\[ \text{Movie}(\text{godfather}, \text{pacino}) \land \neg\text{Director}(\text{pacino}) \implies \text{Actor}(\text{pacino}) \]