September, 2010

SQL QueRIE Recommendations: a query fragment-based approach

Jayad Akbarnejad, *San José State University*
Magdalini Eirinaki, *San José State University*
Suju Koshy, *San José State University*
Duc On, *San José State University*
Neoklis Polyzotis, *University of California, Santa Cruz*

SQL QueRIE Recommendations: A Query Fragment-based Approach

Javad Akbarnejad
Magdalini Eirinaki
Suju Koshy
Duc On
Neoklis Polyzotis

SAN JOSÉ STATE UNIVERSITY

Motivation

- Scientific disciplines use relational DBMS for storage and retrieval of information
  - Biologists (e.g. UCSC Genome, BMRB)
  - Astronomers (e.g. Skyserver)
  - Chemists (e.g. PubChem)
- DBs are accessible online by users with diverse information needs
- Typical users do interactive exploration

Motivation (cont'd)

- Typical users are not SQL experts
- Scientific datasets increase in size
- Users may miss interesting information
  - They do not write the "right" query
  - They are not aware of all parts of the database

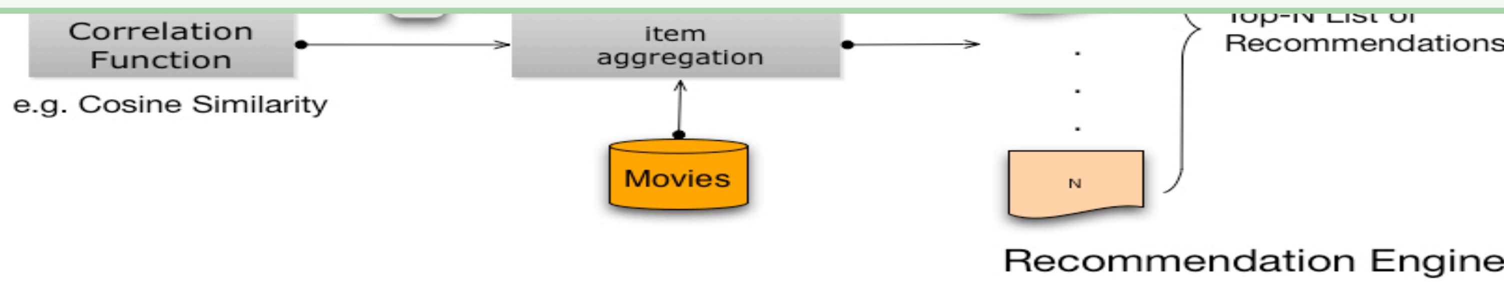
Our goal: Assist users in finding useful information

Web Collaborative Filtering

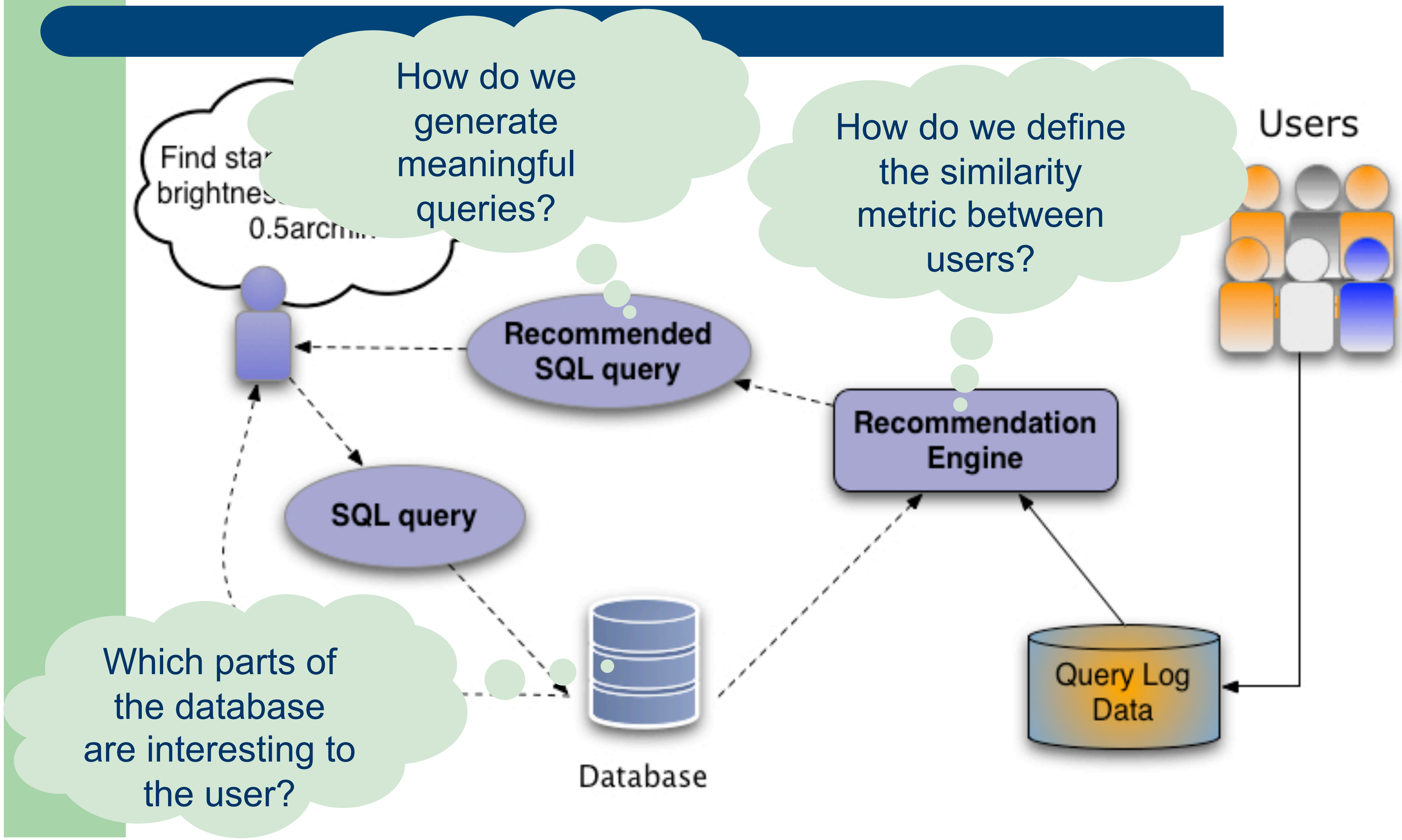
Example: Movie Recommendations

If Alice and Bob **both** like movie X and Alice likes movie Y
then
Bob is likely to be interested in seeing movie Y

If Alice and Bob **both** query data X and Alice queries data Y
then
Bob is likely to be interested in querying data Y



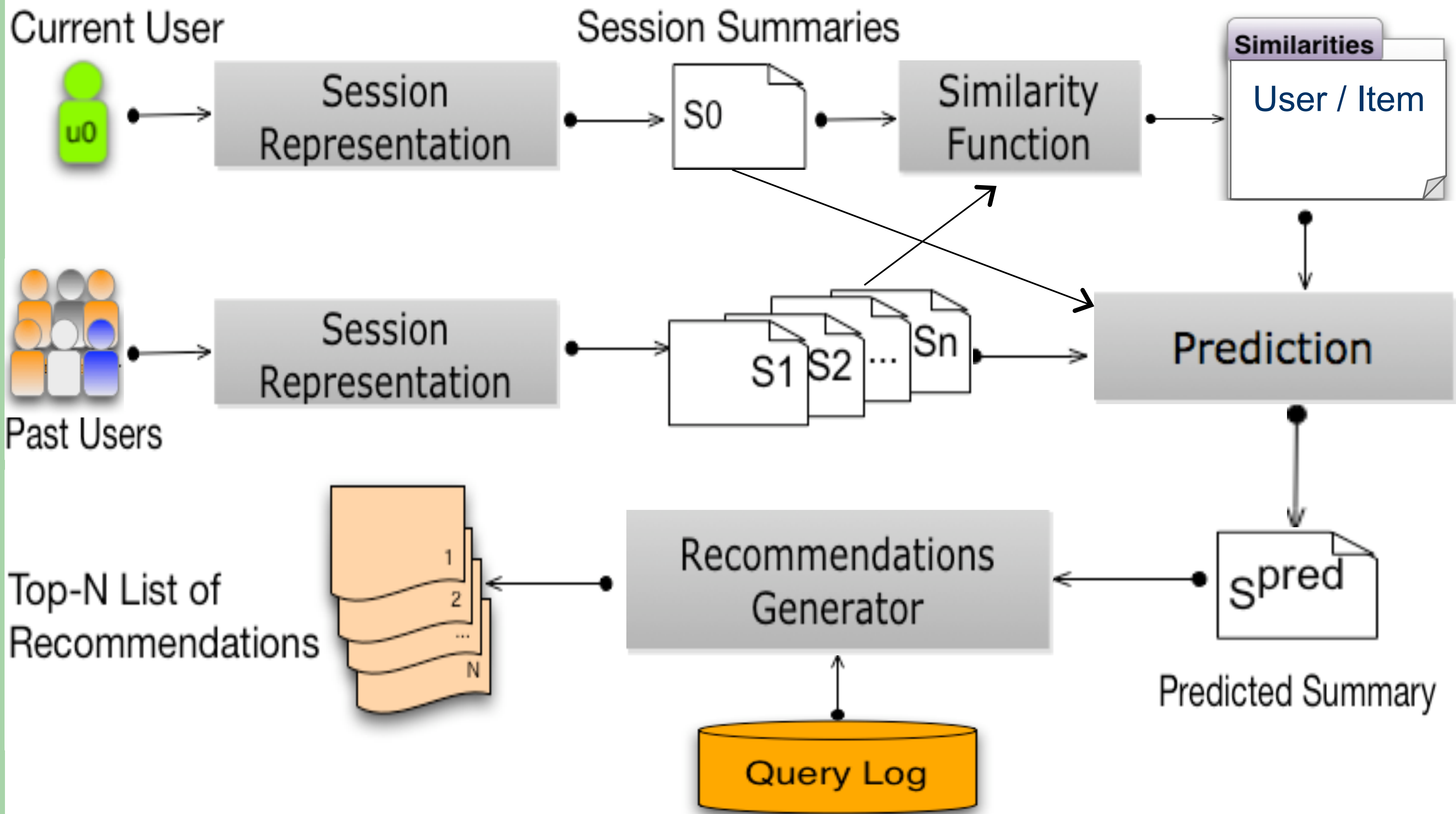
System Architecture



Roadmap

- Introduction
- QueRIE Recommendation Framework
- Experiments
- QueRIE Prototype
- Conclusion

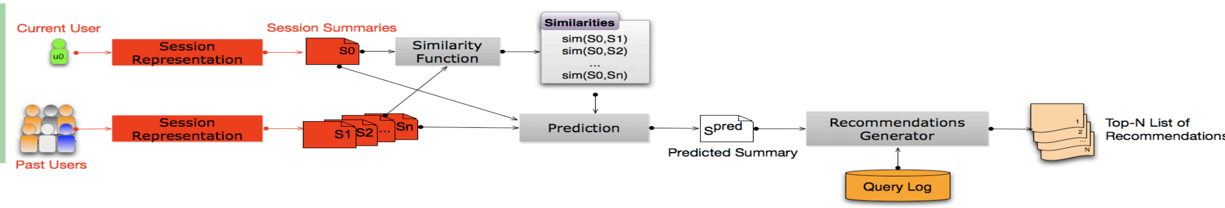
QueRIE Conceptual Framework



QueRIE Recommendation Engines

1. Tuple-based recommendations [SSDBM09, ICDM09]
   - Sessions represented by the tuples “touched” by respective queries
   - User-based similarity: 2 users are similar if they explore the same parts of the DB
   - Predict which parts of DB will interest the user and recommend queries that “touch” them

2. Query fragment-based recommendations



Session Representation

Relations: R(<u>a</u>,b,c)

S(<u>d</u>, e, <u>f</u>)

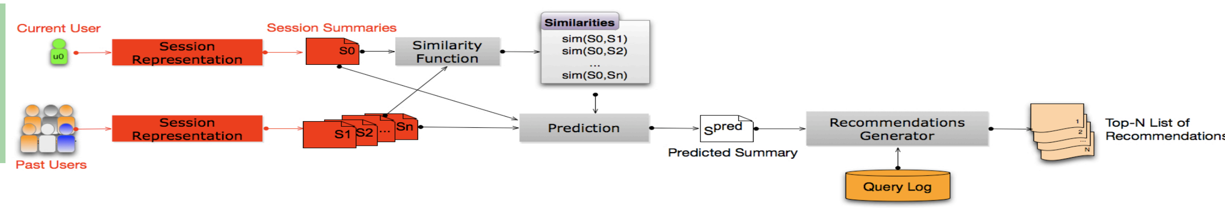
Q_1: SELECT R.a, R.b FROM R WHERE R.b = 2

Q_2: SELECT R.a, R.b, S.e FROM R, S WHERE R.a = S.f AND R.b < 3

Query parsing & relaxation

Q_1: SELECT R.a, R.b FROM R WHERE R.b EQU NUM

Q_2: SELECT R.a, R.b, S.e FROM R, S WHERE R.a EQU S.f AND R.b COMPARE NUM



Session Representation (cont'd)

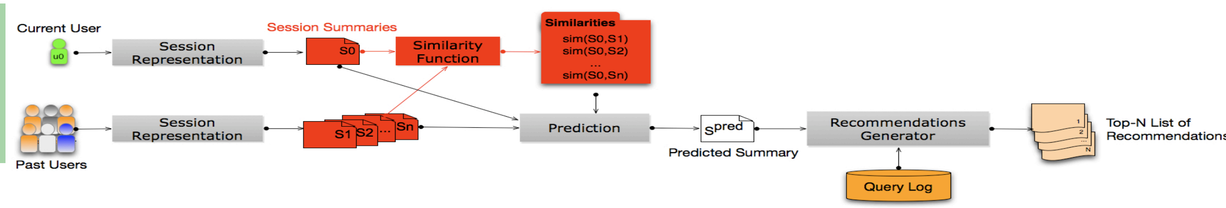
Q_1: SELECT R.a, R.b FROM R WHERE R.b EQU NUM

Q_2: SELECT R.a, R.b, S.e FROM R, S WHERE R.a EQU S.f AND R.b COMPARE NUM

QF = {R, S, …, R.a, R.b, S.e, …, R.b EQU NUM, R.b COMPARE NUM, R.a EQU S.f }

Binary Scheme
Q_1 = <1, 0, …,1,1,0,…,1,0,0>
Q_2 = <1,1,…,1,1,1,…,0,1,1>
S_0 = <1,1,…,1,1,1….,1,1,1>

Weighted Scheme
Q_1 = <1, 0, …,1,1,0,…,1,0,0>
Q_2 = <1,1,…,1,1,1,…,0,1,1>
S_0 = <2,1,…,2,2,1….,1,1,1>

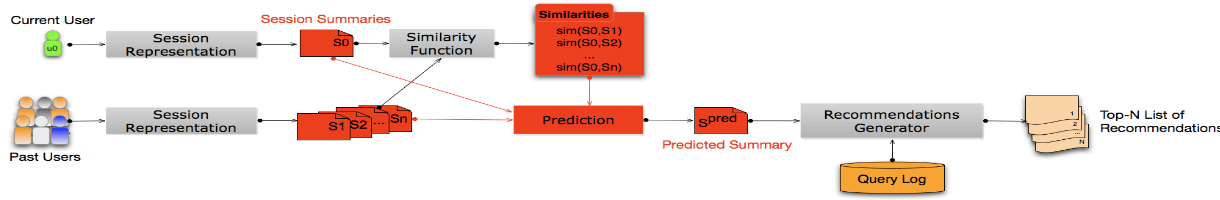


Session Similarity

- Based on the item-based approach
  - Construct *fragment x fragment* similarity matrix offline
  - More efficient than the user-based approach
- Vector-space similarity functions can be used

- High similarity means that the query fragments co-appear frequently in sessions

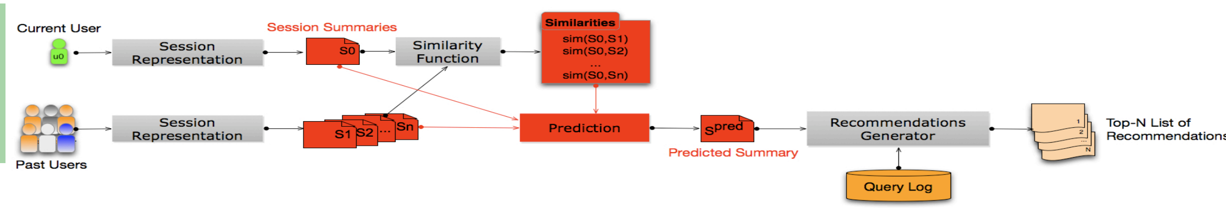
=> the active user might also like to use them



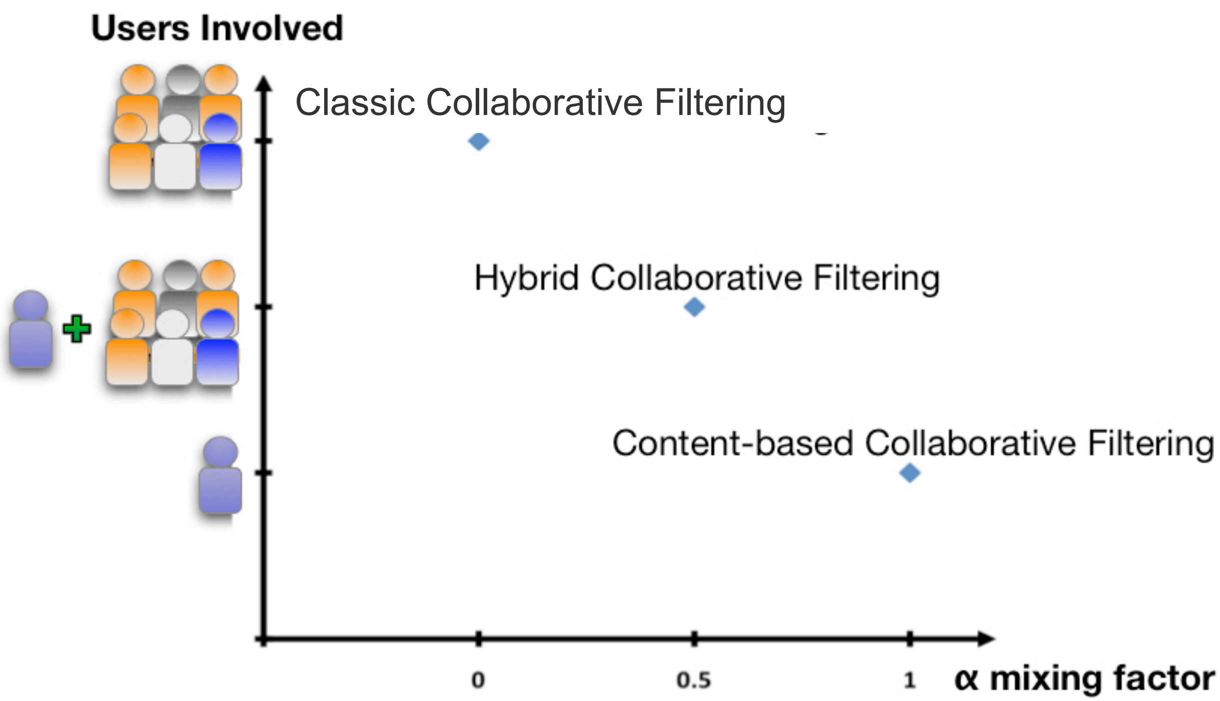
Prediction

- For each fragment ϕ, select top-k similar fragments $\rho \in$ R
- Then compute "predicted summary":

$$S_0^{pred}[\phi] = \frac{\sum_{\rho \in R} S_0[\rho] * sim(\rho, \phi)}{\sum_{\rho \in R} sim(\rho, \phi)}$$

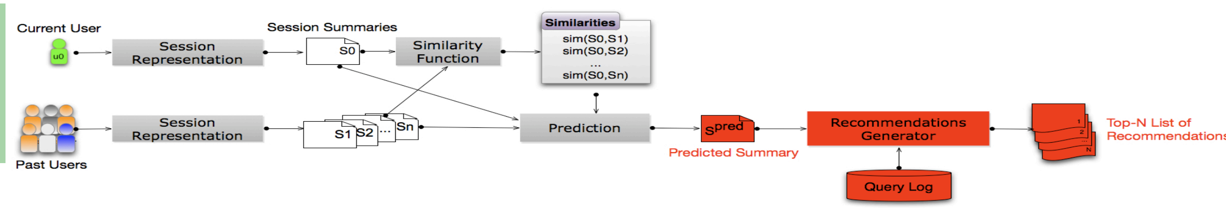


Prediction – the α factor



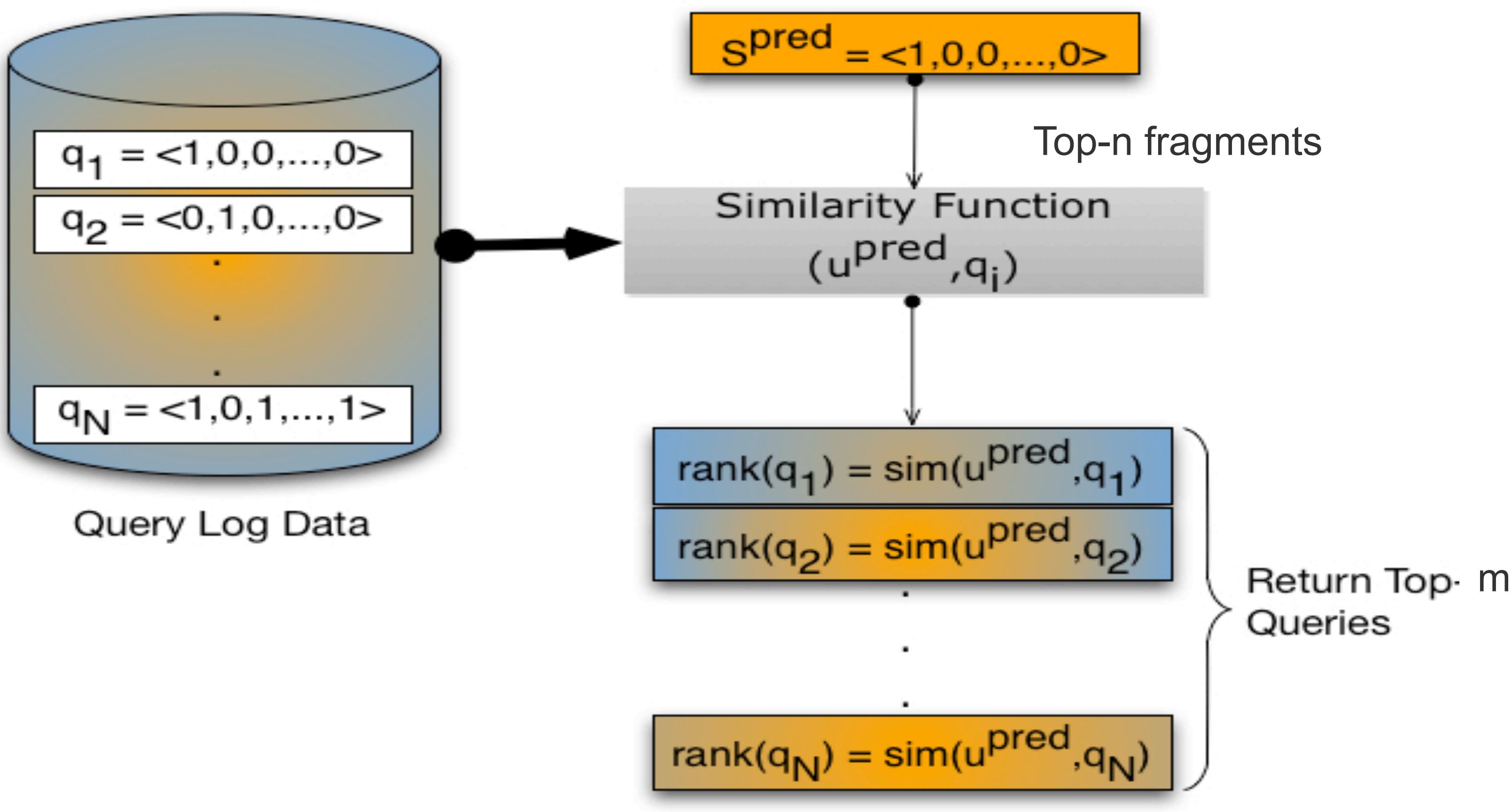
S_0^{pred} contains:

- Only other users' fragments
- Both S_0 and other users' fragments
- Only S_0 fragments



Recommendations Generator

- Use queries of past users



Roadmap

- Introduction
- QueRIE Recommendation Framework
- Experiments
- QueRIE Prototype
- Conclusions

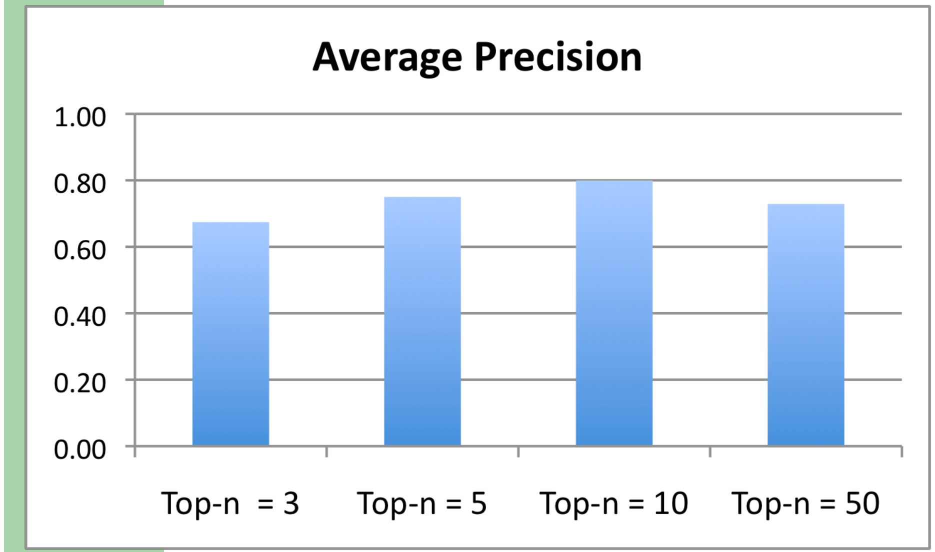
Experimental Setup

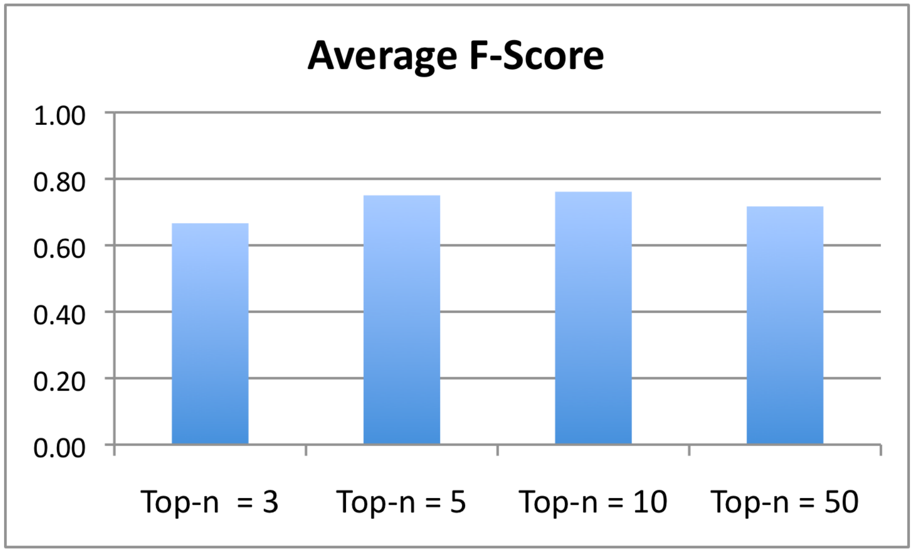
- SkyServer Dataset

#Sessions	180
#Distinct Queries	1400
#Distinct query fragments	755
#Non-zero pair-wise fragment similarities	30436
Avg. number of queries per session	9.3
Min. number of queries per session	3

- Validation method: Holdout Set
- Evaluation Metrics: Precision, Recall, F-Score

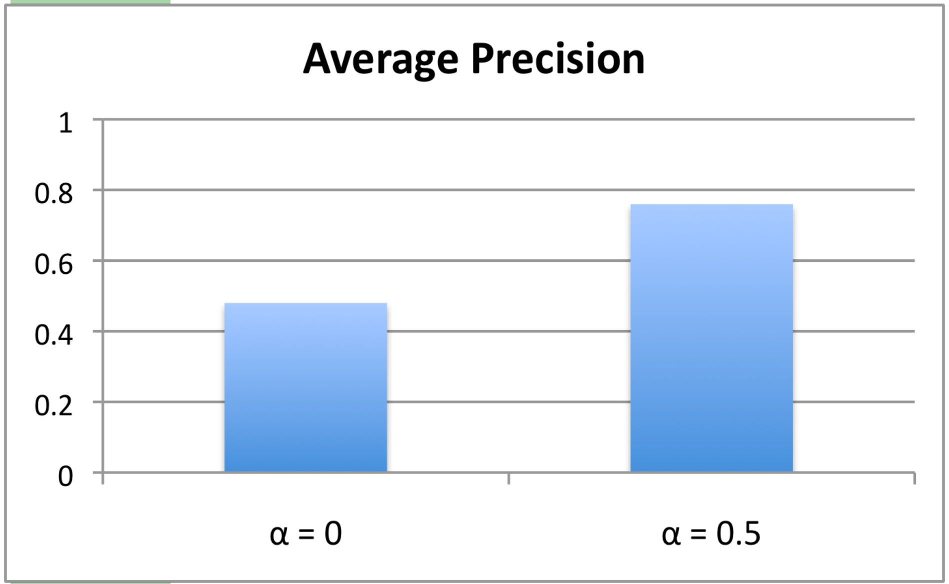
Experimental evaluation – top-n

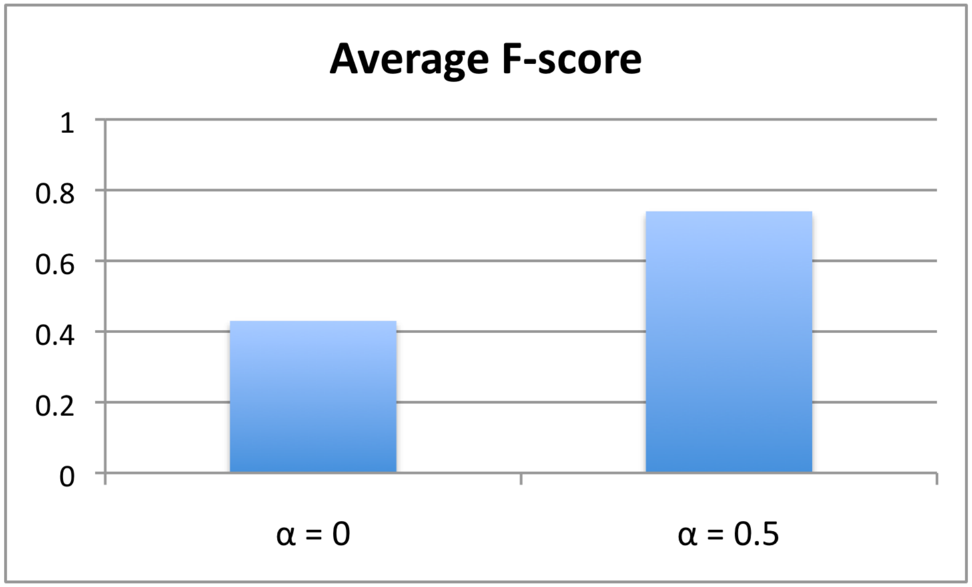




- Precision and recall drop for large *n.*
- More fragments with low similarity included in the mix

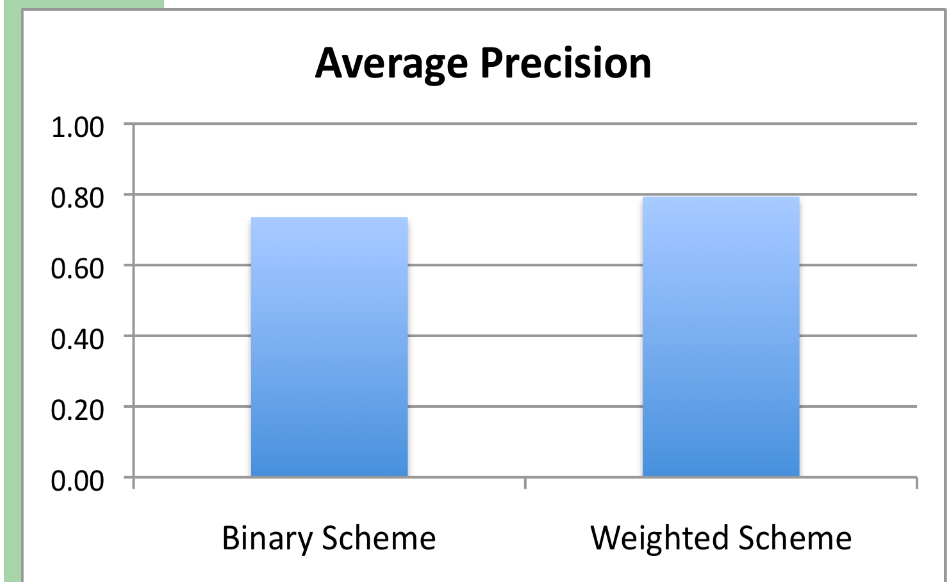
Experimental Evaluation - α

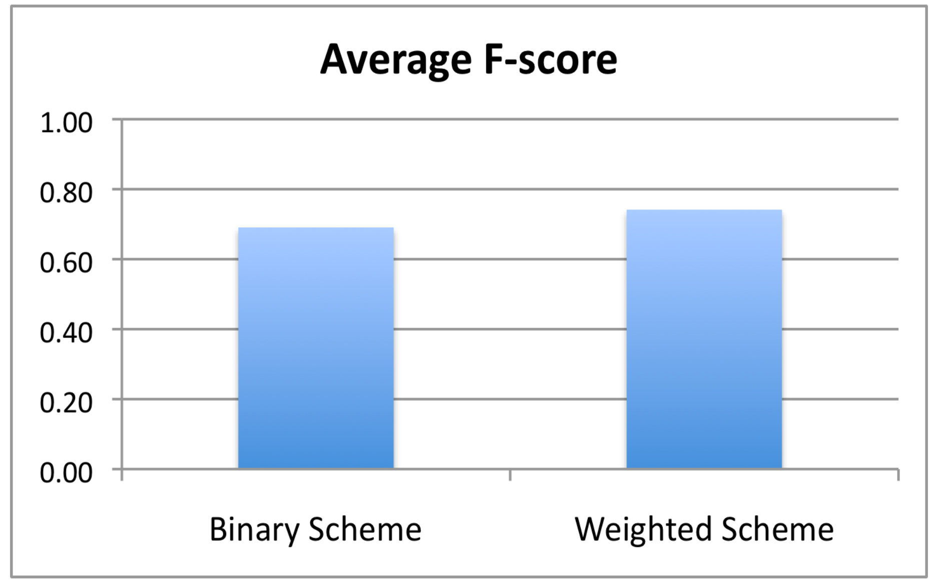




- Including user's current session fragments is beneficial
- Expansion/Restructuring of posted queries

Experimental Evaluation – Weighting Scheme



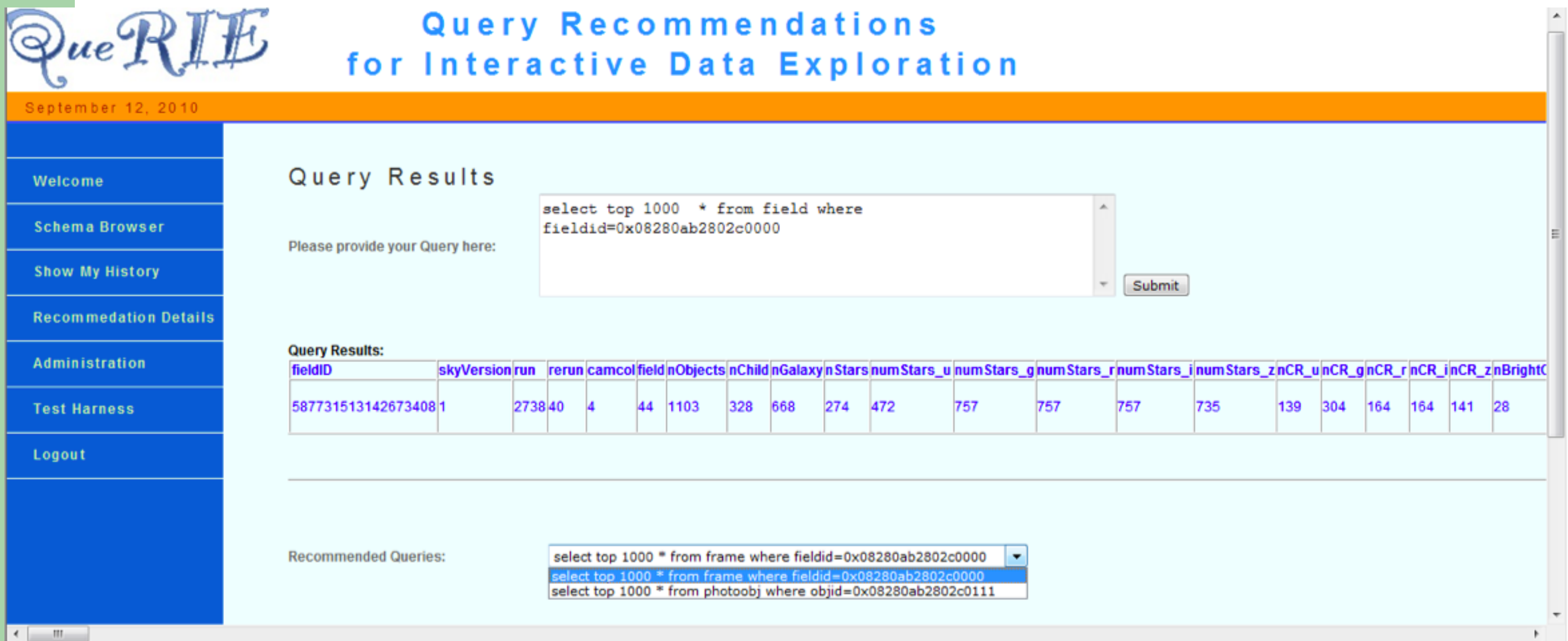


Weighted scheme slightly outperforms the binary

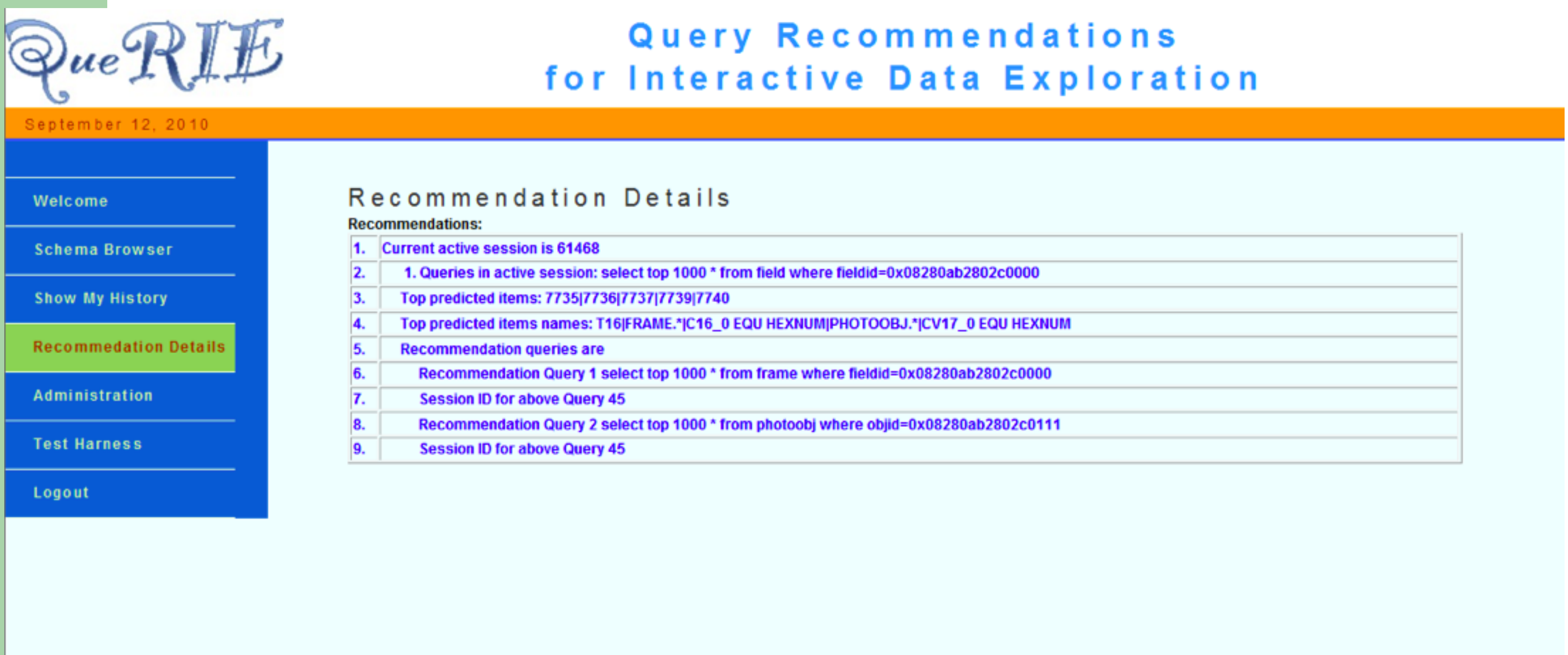
Roadmap

- Introduction
- QueRIE Recommendation Framework
- Experiments
- QueRIE Prototype
- Conclusions

QueRIE Prototype



QueRIE Prototype (cont'd)



QueRIE Prototype

- Demo @ VLDB
  - Session: Data Extraction, Integration and Mining
  - Tue & Wed, 2 – 3:30 PM
  - Lyrebird room

Conclusions

- Non-expert users need help in exploring databases
- Query recommendations can be an effective tool in guiding exploration
- Collaborative filtering provides a natural method to generate recommendations
- Experiments show promising results on real-world datasets
- Ongoing & Future Work:
  - Comparison of two recommendation engines
  - Extend for form-based queries

Thank you !

Questions