Managing Collaborative Feedback Information for Distributed Retrieval

Pascal Felber, Toan Luu, Martin Rajman, Étienne Rivière

Université de Neuchâtel

Ecole Polytechnique Fédérale de Lausanne

Outline

- Motivation
- Collaborative Feedback based Retrieval System
  - General approach
  - System architecture
- Challenges and Ongoing works
- Related works
- Conclusion

Motivation

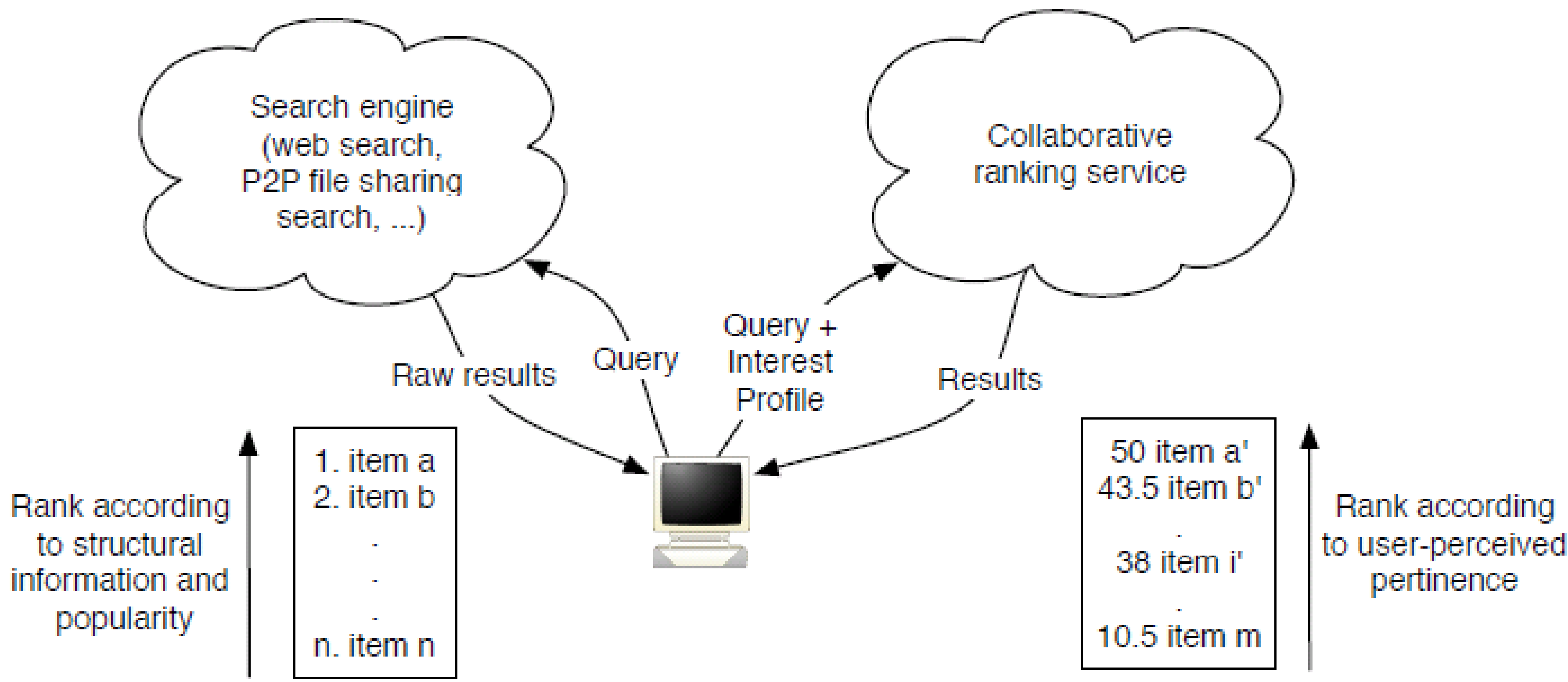
- Many research efforts on P2P Web search
  - ~18,800 results on Google Scholar with keywords “P2P web search” in Oct 2008
- No P2P system has reached the level of quality and efficiency of centralized search engines (bootstrap problem)
  - Faroo, Yacy: released in 2006, ~hundred users in 2008.
- Our argument for P2P Web search:
  - Do not try to replace centralized search engines, but complement them with additional functionalities!

Our goals

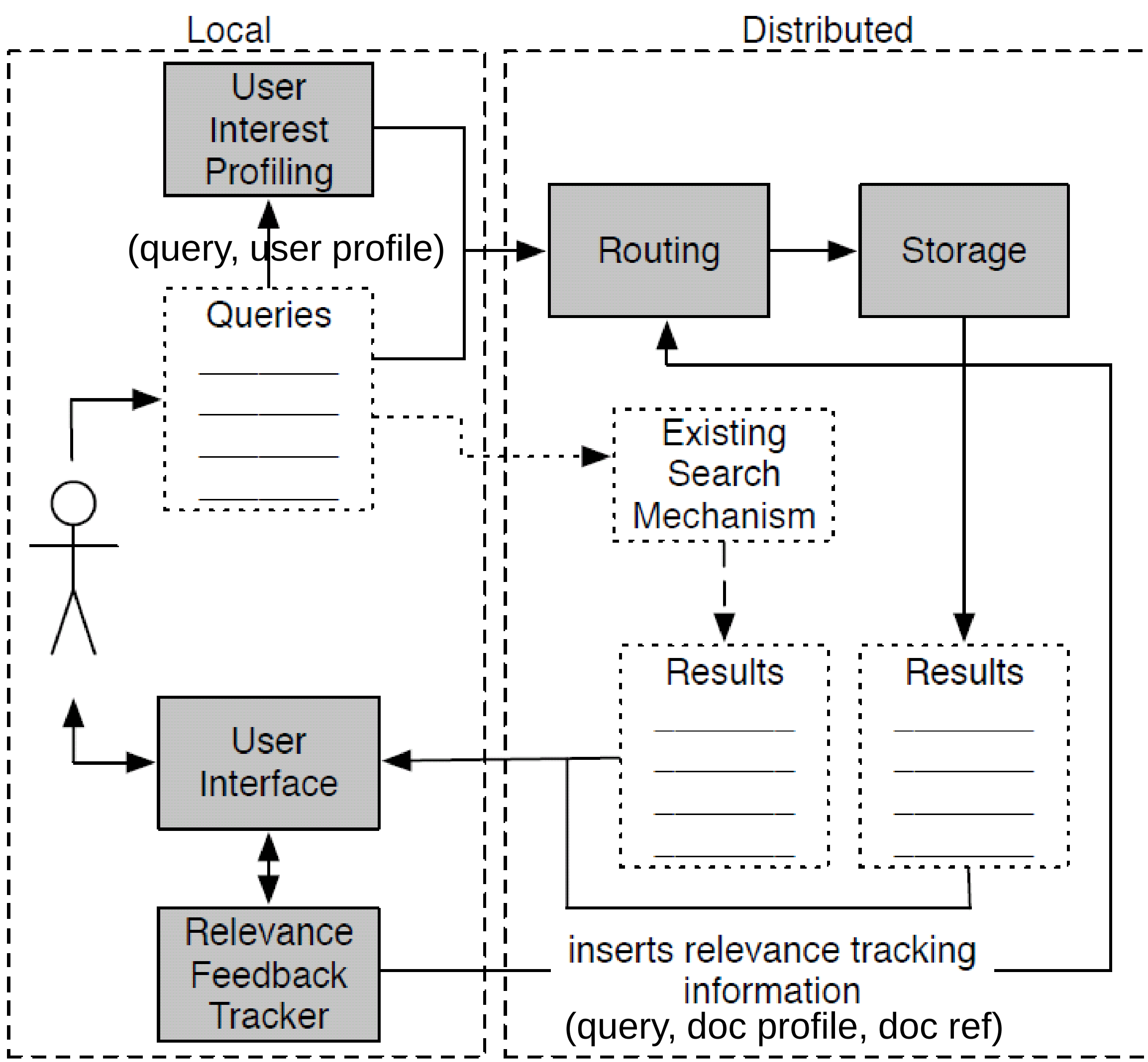
Design a Collaborative (relevance) Feedback-based Retrieval System that:

- Provides a dedicated P2P storage system for aggregating the search results obtained by a community of users from a givent (or possibly several) centralized search engine(s)
  - Collaborative (P2P) storage and aggregation
  - Balanced load
  - Decentralized and dynamic settings
- Uses semantic profiling
  - User profiles and aggregated (relevance) feedback information are used to better target the results (re-ranking)

General approach



System architecture



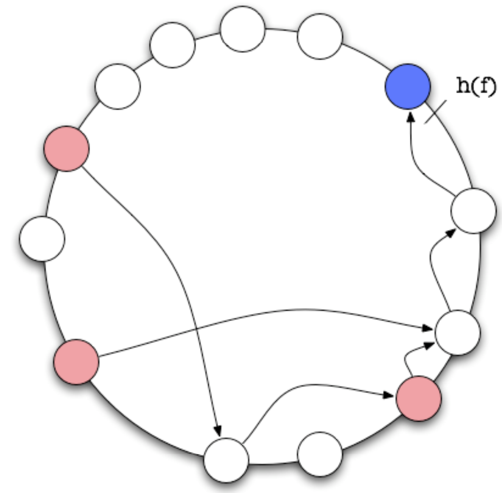
User/Document profiles

- Document profile
  - Set of most representative keywords extracted from the document or from the document summary (snippet)
  - Document profiles are used to preserve the informational diversity of the relevance feedback information stored by the collaborative system.
- User profile (for a query)
  - Set of the most relevant expansion keywords generated for the query by the local query expansion system
  - User profiles are used by the collaborative system to filter the results generated for a given query before retrieving them
- User/document profiles represented by Bloom filters
  - Compact and encoded (privacy)
  - Adequate for computing the Jaccard similarity : $\frac{|S_1 \bigcap S_2|}{|S_1 \bigcup S_2|}$

Profile maintenance/usage

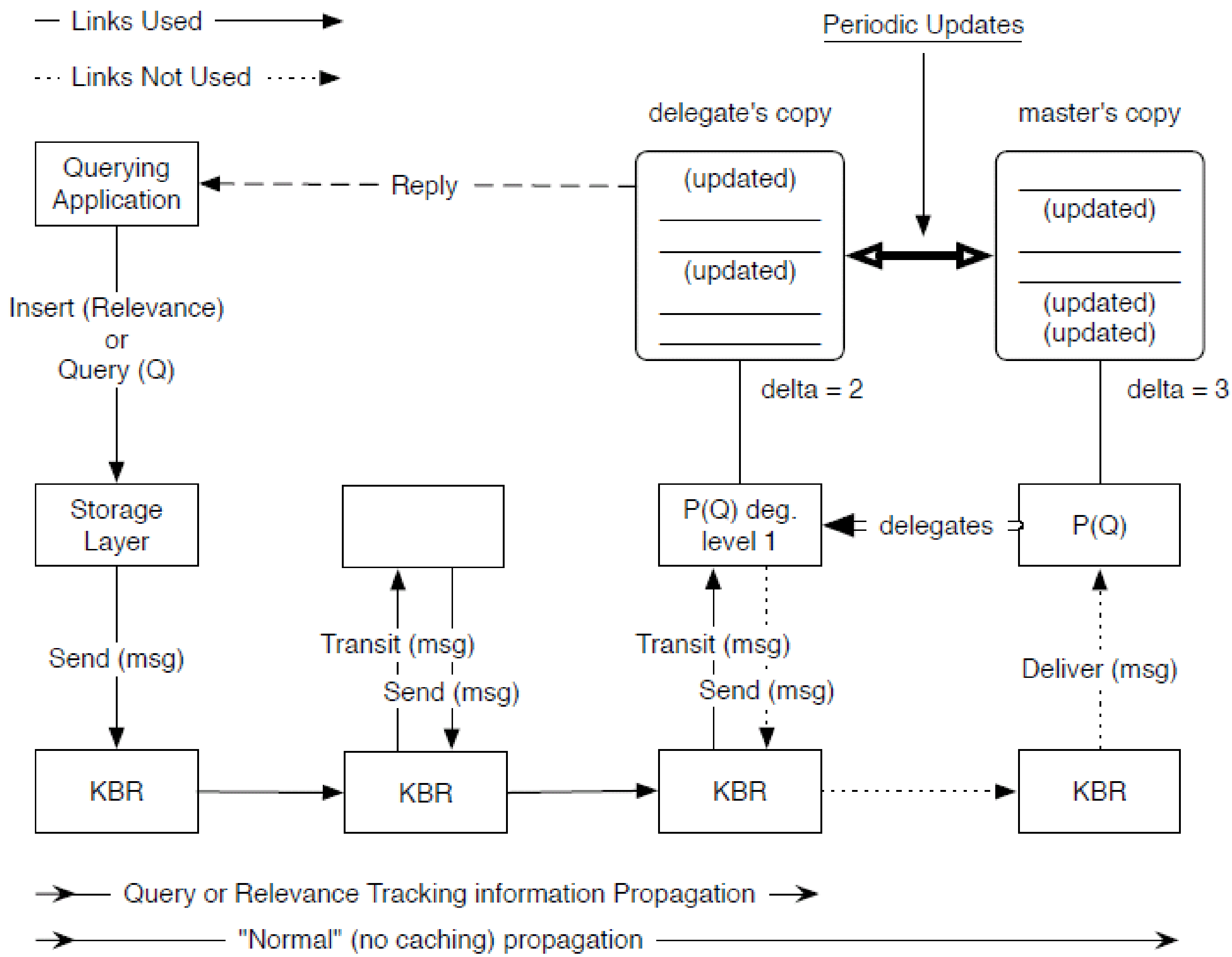
- **Each time a user selects a document in the result list obtained from the collaborative system for a given query**
  - the most representative keywords are extracted from the document
  - the query and the selected keywords are provided to the local query expansion system
  - the selected keywords are stored in a Bloom filter (document profile) that is added to the (query, document reference) relevance feedback transmitted to the collaborative system
- **Each time a user submits a query to the collaborative system**
  - the most relevant query expansion keywords are retrieved from the local query expansion system
  - the selected keywords (user profile) are stored in a Bloom filter that is associated to the query when submitted to the collaborative system.

Routing layer



- Structured P2P overlay
  - O(log N) hops to reach the destination
  - Resilient to dynamic settings
  - Each peer holds a balanced number of query terms
- 2 calls for the application layer:
  - Request: (query, user profile)
  - Feedback: (query, docRef, document profile)
- Skewness of accesses leads to load unbalance
  - Specific, adaptive load balancing mechanisms

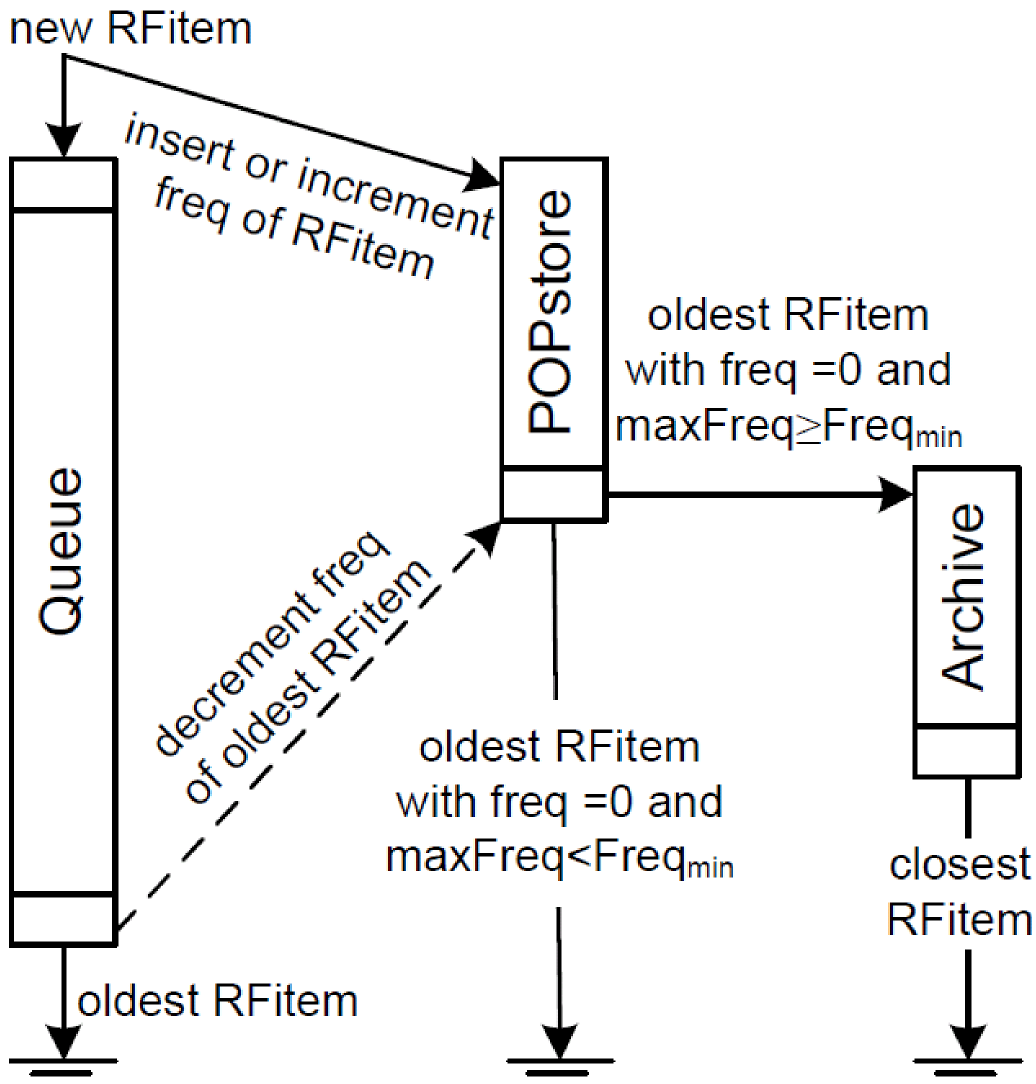
Routing layer: Delegation mechanism



Storage layer

- **Manages of relevance feedback provided by users**
  - Queue: temporal storage for the identification of popular RFitems
  - POPstore: stores the popular-”not yet” popular RFitems.
  - Archive: store past popular Rfitems.
- **Generates the result list for the submitted queries**
  - Send back *k*-most similar RFitem w.r.t. users’ profile

Storage layer: Algorithm



Challenges

- User profiling
  - Deciding on the set of representative keywords.
  - Bloom filter dimensioning
- Routing layer
  - Latency and Throughput
  - Scalability and Load Balancing
  - Fault Tolerance
- Storage layer
  - Replace queue by using a probabilistic modeling for arrival time

Ongoing work

- Query log analysis (AOL)
  - 36 M lines of data
  - 10 M unique (normalized) queries
  - 19 M user click-through events
  - 0.6 M unique user ID‘s
- Prototype implementation
  - Not a simulator !
  - Running on a cluster and on PlanetLab
- Evaluation
  - Retrieval quality
  - Load balance of P2P layer

Related works

- P2P Web search:
  - Mainly concentrates on comparing with centralized systems (scalability, bandwidth consumption, retrieval quality…)
- Meta search engine:
  - Do not take into account relevance feedback and user profiles.
- Search techniques based on user interest profiles:
  - Do not benefit from collaboration in user communities.
- Collaborative (social) annotations:
  - Requires annotation efforts
  - Centralized management

Conclusion

- Using a collaborative approach to complement centralized search
- Customized search result based on:
  - Users' interest profiling
  - Popularity of users' feed back (click through)
  - Diversity of search results
- Specially designed P2P system:
  - Leverages properties of key-based routing
  - Adaptive load balancing mechanisms