IQN Routing: Integrating Quality and Novelty for P2P Web Search

EDBT 2006, Munich, Germany

Sebastian Michel *, Matthias Bender *, Peter Triantafillou ◊, Gerhard Weikum *

* Max-Planck-Institut für Informatik
66123 Saarbrücken, Germany

◊ RACTI and University of Patras
26500 Rio, Greece

27th March 2006
Outline

1 Introduction

2 Distributed Web Search with Minerva

3 IQN Routing

4 Evaluation

5 Conclusion and Outlook
P2P Systems

Became famous through file-sharing applications, like Gnutella, KaZAA, Napster.

Applications like:

- Internet telephony (e.g. Skype)
- Filesharing
- Pub/Sub

Question:
Is there an interesting and legal P2P application?
Motivation

Why P2P Web Search?

- Benefit from intellectual input from a large user community. (Bookmarks, click-streams, ...)
- Break information monopolies
- Coverage of the web
- Exploit mostly idle resources

Related to distributed IR, but some additional aspects

- high dynamics
- each peer has its own collection
- peers are independently crawling the web
Minerva Design Fundamentals

- Peers with local collections, e.g., built by focused crawler. Tailored to the users’ specific interest profiles.
- Peers share metadata about local indexes
- Form physically distributed term $\rightarrow$ peer directory
- Layered on top of DHT
- Peers use directory to discover promising peers for query
Minerva System Architecture
Inside Peerlists ...

Peer X
ip:port
Time to Live: Jan 30th 6:45pm
Statistical Metadata:
Collection size,
Document frequency (df) ,...
Hash Sketch
0110111111
0110111111

Peer Y
ip:port
Time to Live: Jan 30th 1:31pm
Statistical Metadata:
Collection size,
Document frequency (df) ,...
Hash Sketch
0101010111
0101010111

Peer Z
ip:port
Time to Live: Jan 30th 10:45am
Statistical Metadata:
Collection size,
Document frequency (df) ,...
Hash Sketch
0101011111
0101011111

Peer V
ip:port
Time to Live: Jan 30th 3:24pm
Statistical Metadata:
Collection size,
Document frequency (df) ,...
Hash Sketch
0111011111
0111011111

Peer W
ip:port
Time to Live: Jan 30th 2:40pm
Statistical Metadata:
Collection size,
Document frequency (df) ,...
Hash Sketch
0101110111
0111011111

Peer X
ip:port
Time to Live: Jan 30th 3:01pm
Statistical Metadata:
Collection size,
Document frequency (df) ,...
Hash Sketch
0101110111
0110101111
How to find promising Peers?

State of the art: Find peers with high quality documents.

Existing Strategies:

- based on per-term metadata
- combine metadata for all query terms
- select peers with highest expected result quality
Why Quality-Only is not Enough!

Problem:
Peers crawl the web independently. → **overlapping collections**
(\textit{For instance, cnn.com might have been crawled by a lot of peers})

Goal: \textbf{Integrate Quality and Novelty (IQN).}

→ achieve high recall with fewer peers than traditional approaches

\textbf{We define}: Usefulness := \textit{quality} \ast \textit{novelty}
Example

Naive routing strategy:

\[
\{A\} \quad \{A,B\} \quad \{A,B,C\} \quad \{A,B,C,D\}
\]

Overlap aware routing strategy:

\[
\{A\} \quad \{A,E\}
\]
IQN Algorithm: Integrating Quality and Novelty

Extend per-peer, per-term meta-data with synopses that describe local collections.

Select all peers to query a-priori
Based on statistics (not their actual query results).

- Choose first Peer X based on quality only → use X’s per-query descriptor as initial representation of already seen documents.
- Then choose Peer Y with the highest usefulness w.r.t. the already seen docs.
- Merge representations for the peers selected so far and iterate.
Quality Prediction

Find best suitable peers based only on quality measures.

Most popular selection strategies:

- CORI (Callan et al.)
- GLOSS (Gravano et al.)
- Decision-theoretic framework (Fuhr)

Based on:

- document frequency
- maximum term frequency
- total number of distinct terms
- total number of documents
Novelty Prediction

Add statistics that allow novelty estimation.

We are interested in \(|S_B - (S_A \cap S_B)|\)

Two possible approaches:

- represent whole collection
- use separate representations for (term-specific) index lists

Term-specific representations allow query-specific overlap estimation!

Multi-keyword queries

Combine per-term descriptors of a peer to form per-query descriptor
Data Synopses

Searching for appropriate data synopses ...

Requirements:

- Compact
- Highly accurate
- (.....)

- Bloom Filter
- Hash-Sketches
- Min-wise independent permutations
Bloom Filter

Bit-array of length m. Insert documents by setting bits using $k$ hash functions.

**Membership-Queries:** A document is contained in a Bloom filter if the corresponding bits are set. Problem: false positives

$$pfp \approx (1 - e^{-kn/m})^k$$
Hash Sketches

- Pseudo-uniform hash function $h$
- Apply $h$ to all documents and record the position of the least significant (leftmost) 1-bit in the binary representation in a bitmap vector $B[0 \ldots L - 1]$.
- Idea: $B[0]$ will be set approximately $\frac{n}{2}$ times, $B[1]$ approximately $\frac{n}{4}$ times, ..... 
- More formally: The leftmost 1 bit at position $k$ provides an estimation of $\log(n)$. 

Min-wise independent Permutations (MIPS)

$N$ independent permutations

Given $|S_A|$, $|S_B|$, and the resemblance $R = \frac{|S_A \cap S_B|}{|S_A \cup S_B|}$, we estimate the overlap between $S_A$ and $S_B$ as $|S_A \cap S_B| = \frac{R\times(|S_A|+|S_B|)}{(R+1)}$. 

$$
\begin{align*}
\text{docID set} & \\
17 & 21 & 3 & 12 & 24 & 8 \\
\hline
h_1 & = (7x + 3) \mod 51 & 8 & 9 & 24 & 36 & 18 & 8 \\
\hline
h_2 & = (5x + 6) \mod 51 & 40 & 9 & 21 & 15 & 24 & 46 \\
\hline
h_N & = (3x + 9) \mod 51 & 9 & 21 & 18 & 45 & 30 & 33 \\
\hline
\end{align*}
$$

Create MIP Vector from Minima of Permutations

Apply Permutations to all docID's

Estimated Resemblance

$$
\begin{align*}
\text{Estimated Resemblance} & = \frac{2}{6} \\
& = 0.33 \\
\end{align*}
$$

$$
\begin{align*}
\text{Estimated Resemblance} & = \frac{2}{6} \\
& = 0.33 \\
\end{align*}
$$

$$
\begin{align*}
\text{Estimated Resemblance} & = \frac{2}{6} \\
& = 0.33 \\
\end{align*}
$$

$$
\begin{align*}
\text{Estimated Resemblance} & = \frac{2}{6} \\
& = 0.33 \\
\end{align*}
$$

$$
\begin{align*}
\text{Estimated Resemblance} & = \frac{2}{6} \\
& = 0.33 \\
\end{align*}
$$
Aggregate Synopses

**Remember:** After having selected the best peer in an iteration of the IQN method, we need to update the reference synopsis.

**MIPs**
Given $MIPs_A[]$ and $MIPs_B[]$, one can form $MIPs_{A∪B}[]$ as follows:

$MIPs_{A∪B}[i] = \min\{MIPs_A[i], MIPs_B[i]\} \; \forall \; i : 1 \leq i \leq n.$

**Hash Sketches**
straight forward: use the bit-wise OR operation

**Bloom Filter**
Again, bit-wise OR: $BF_{A∪B}[i] = BF_A[i] \; OR \; BF_B[i] \; \forall \; i : 1 \leq i \leq n.$
Accuracy Evaluation

Experiments on synthetic data-sets:

Accuracy of Resemblance Estimation
Fixed Collection Size 5000

Accuracy of Resemblance Estimation
Expected 33% Mutual Overlap
Experimental Setup (2)

- **Routing Strategies**
  - pure CORI
  - overlap aware CORI using MIP based overlap prediction
  - overlap aware CORI using Bloom Filter based overlap prediction

- **Datasets:**
  - Subset of the official TREC .GOV collection split into disjoint fragments. Building peers using
    - sliding window over these fragments
    - mirrored collections
    - ...
  - Queries: 50 TREC-2003 Web queries, e.g. juvenile delinquency
  - Measure the recall w.r.t. the query results of the whole document set (relative recall)
Experimental Results: \( \binom{6}{3} \) Benchmark

![Graph showing experimental results for different routing methods.](image_url)
Experimental Results: Sliding Window Benchmark
Conclusion and Outlook

Conclusion

- Comprehensive performance evaluation of MIPs, Bloom Filter, and Hash-Sketches
- Experiments on real-web data showing the impact of overlap aware query routing

Future Work

- Evaluation of the usage of histogram enhanced synopses
- Adaptive synopses lengths
- System behavior under churn