Federated Text Search

Luo Si

Department of Computer Science
Purdue University
Abstract

Outline

- Introduction to federated search
- Main research problems
  - Resource Representation
  - Resource Selection
  - Results Merging
- A unified utility maximization framework for federated search
- Modeling search engine effectiveness
Federated Search

Visible Web vs. Hidden Web

Visible Web: Information can be copied (crawled) and accessed by conventional search engines like Google or Yahoo!

Hidden Web: Information hidden from conventional engines. Provide source-specific search engine but no arbitrary crawling of the data (e.g., USPTO)

- No arbitrary crawl of the data (e.g., ACM library)
- Updated too frequently to be crawled (e.g., buy.com)

Hidden Web contained in (Hidden) information sources that provide text search engines to access the hidden information
Federated Search

Search the United States Code

U.S. Department of Education
My.Education.gov

NTIS
U.S. Department of Commerce
National Technical Information Service

THOMAS
Legislative Information on the Internet

FedBizOpps/Commerce Business Daily

U.S. EPA Publications Search

GPO Access
The U.S. Government's Official Web Portal

ARC
Archival Research Catalog

MMWR
Morbidity and Mortality Weekly Report

United States Patent and Trademark Office

Search the United States Code

National Library of Medicine

National Center for Biotechnology Information

U.S. Securities and Exchange Commission

Protein Interaction Search:

Astronomy Picture of the Day
Searchable Archive

Supernova Cosmology Project

American Memory
Historical Collections for the National Digital Library

USDA

NSF

EU

science for a changing world

Historical Collections for the National Digital Library

National Library of Medicine

01101011 National Center for Biotechnology Information

Protein Interaction Search:
Introduction

Hidden Web is:

- Larger than Visible Web (2-50 times, Sherman 2001)
- Created by professionals

Valuable

Searched by Federated Search

Federated Search Environments:

Small companies: Probably cooperative information sources
Big companies (organizations): Probably uncooperative information sources
Web: Uncooperative information sources
Federated Search

Components of a Federated Search System and Two Important Applications

Information source recommendation: Recommend information sources for users’ text queries (e.g., completeplanet.com): Steps 1 and 2

Federated document retrieval: Also search selected sources and merge individual ranked lists into a single list: Steps 1, 2 and 3
Introduction

Solutions of Federated Search

Browsing model: Organize sources into a hierarchy; Navigate manually

From: CompletePlanet.com
Introduction

Solutions of Federated Search

Information source recommendation: Recommend information sources for users’ text queries

- Useful when users want to browse the selected sources
- Contain resource representation and resource selection components

Federated document retrieval: Search selected sources and merge individual ranked lists

- Most complete solution
- Contain all of resource representation, resource selection and results merging
**Introduction**

**Modeling Federated Search**

**Application in real world**

- FedStats project: Web site to connect dozens of government agencies with uncooperative search engines
  - Previously use centralized solution (ad-hoc retrieval), but suffer a lot from missing new information and broken links
  - Require federated search solution: A prototype of federated search solution for FedStats is on-going in Carnegie Mellon University

- Good candidate for evaluation of federated search algorithms

- But, not enough relevance judgments, not enough control…

Require Thorough Simulation
Introduction

Modeling Federated Search

TREC data

- Large text corpus, thorough queries and relevance judgments

Simulation with TREC news/government data

- Professional well-organized contents
- Often be divided into $O(100)$ information sources
- Simulate environments of large companies or domain specific hidden Web
- Most commonly used, many baselines (Lu et al., 1996)(Callan, 2000)....
- Normal or moderately skewed size testbeds: Trec123 or Trec4_Kmeans
- Skewed: Representative (large source with the same relevant doc density), Relevant (large source with higher relevant doc density), Nonrelevant (large source with lower relevant doc density)
Introduction

Modeling Federated Search

Simulation multiple types of search engines

- **INQUERY**: Bayesian inference network with Okapi term formula, doc score range \([0.4, 1]\)
- **Language Model**: Generation probabilities of query given docs, doc score range \([-60, -30]\) (log of the probabilities)
- **Vector Space Model**: SMART “Inc.ltc” weighting, doc score range \([0.0, 1.0]\)

Federated search metric

- Information source size estimation: Error rate in source size estimation
- Information source recommendation: **High-Recall**, select information sources with most relevant docs
- Federated doc retrieval: **High-Precision** at top ranked docs
Abstract

Outline

- Introduction to federated search
- Main research problems
  - Resource Representation
  - Resource Selection
  - Results Merging
- A unified utility maximization framework for federated search
- Modeling search engine effectiveness
Previous Research on Resource Representation

Resource descriptions of words and the occurrences

- STARTS protocol (Gravano et al., 1997): Cooperative protocol
- Query-Based Sampling (Callan et al., 1999):
  - Send random queries and analyze returned docs
  - Good for uncooperative environments

Centralized sample database: Collect docs from Query-Based Sampling (QBS)

- For query-expansion (Ogilvie & Callan, 2001), not very successful
- Successful utilization for other problems, throughout this proposal
Research on Resource Representation

Information source size estimation

Important for resource selection and provide users useful information
- Capture-Recapture Model (Liu and Yu, 1999)
  Use two sets of independent queries, analyze overlap of returned doc ids
  But require large number of interactions with information sources

Sample-Resample Model (Si and Callan, 2003)
Assume: Search engine indicates num of docs matching a one-term query
Strategy: Estimate df of a term in sampled docs
  Get total df from by resample query from source
Scale the number of sampled docs to estimate source size
Research Problems (Resource Representation)

Experiment Methodology

Methods are allowed the same number of transactions with a source

Two scenarios to compare Capture-Recapture & Sample-Resample methods

- Combined with other components: methods can utilize data from Query-Based Sample (QBS)
- Component-level study: can not utilize data from Query-Based Sample

Capture-Recapture (Scenario 1)

Capture-Recapture (Scenario 2)

Sample-Resample

Queries

Downloaded documents

Data may be acquired by QBS (80 sample queries acquire 300 docs)
Research Problems (Resource Representation)

Experiments

To conduct component-level study

- Capture-Recapture: about 385 queries (transactions)

- Sample-Resample: 80 queries and 300 docs for sampled docs
  (sample) + 5 queries (resample) = 385 transactions

Measure:

\[
\text{AER} = \frac{|N-N^*|}{N^*}
\]

<table>
<thead>
<tr>
<th></th>
<th>Trec123 (Avg AER, lower is better)</th>
<th>Trec123-10Col (Avg AER, lower is better)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cap-Recapture</td>
<td>0.729</td>
<td>0.943</td>
</tr>
<tr>
<td>Sample-Resample</td>
<td>0.232</td>
<td>0.299</td>
</tr>
</tbody>
</table>

Collapsing every 10th source of Trec123

Estimated Source Size

Actual Source Size
Abstract

Outline

- Introduction to federated search
- Main research problems
  - Resource Representation
  - Resource Selection
  - Results Merging
- A unified utility maximization framework for federated search
- Modeling search engine effectiveness
Research Problems (Resource Selection)

Goal of Resource Selection of Information Source Recommendation

High-Recall: Select the (few) information sources that have the most relevant documents

Research on Resource Selection

Resource selection algorithms that need training data

  DTF causes large human judgment costs

- Lightweight probes (Hawking & Thistlewaite, 1999)
  Acquire training data in an online manner, large communication costs
Research Problems (Resource Selection)

Research on Resource Representation

“Big document” resource selection approach: Treat information sources as big documents, rank them by similarity of user query

- Cue Validity Variance (CVV) (Yuwono & Lee, 1997)
- CORI (Bayesian Inference Network) (Callan, 1995)
- KL-divergence (Xu & Croft, 1999)(Si & Callan, 2002), Calculate KL divergence between distribution of information sources and user query

CORI and KL were the state-of-the-art (French et al., 1999)(Craswell et al., 2000)

But “Big document” approach loses doc boundaries and does not optimize the goal of High-Recall
Research Problems (Resource Selection)

Research on Resource Representation

But “Big document” approach loses doc boundaries and does not optimize the goal of High-Recall

Relevant document distribution estimation (ReDDE) (Si & Callan, 2003)

Estimate the percentage of relevant docs among sources and rank sources with no need for relevance data, much more efficient
Research Problems (Resource Selection)

Relevant Doc Distribution Estimation (ReDDE) Algorithm

\[
\text{Rel}_Q(i) = \sum_{d \in \text{db}_i} P(\text{rel}|d) \times P(d|\text{db}_i) \times N_{\text{db}_i}
\]

\[
\approx \sum_{d \in \text{db}_i \_\text{samp}} P(\text{rel}|d) \times \text{SF}_{\text{db}_i}
\]

Source Scale Factor
\[
\text{SF}_{\text{db}_i} = \frac{\hat{N}_{\text{db}_i}}{N_{\text{db}_i \_\text{samp}}}
\]

Estimated Source Size

Number of Sampled Docs

“Everything at the top is (equally) relevant”

\[
P(\text{rel}|d) = \begin{cases} 
C_Q & \text{if } \text{Rank}_{\text{CCDB}}(Q, d) < \text{ratio} \times \sum_i N_{\text{db}_i} \\
0 & \text{otherwise}
\end{cases}
\]

Problem: To estimate doc ranking on Centralized Complete DB
ReDDE Algorithm (Cont)

In resource representation:
• Build representations by QBS, collapse sampled docs into centralized sample DB

In resource selection:
• Construct ranking on CCDB with ranking on CSDB

Centralized Sample DB

Resource Representation

Engine 1

Engine 2

Engine N

CSDB Ranking

CCDB Ranking

Threshold
Research Problems (Resource Selection)

Experiments

On testbeds with uniform or moderately skewed source sizes

\[ R_k = \frac{\sum_{i=1}^{k} E_i}{\sum_{i=1}^{k} B_i} \]

Evaluated Ranking

Desired Ranking
Research Problems (Resource Selection)

Experiments

On testbeds with skewed source sizes
Abstract

Outline

- Introduction to federated search
- Main research problems
  - Resource Representation
  - Resource Selection
  - Results Merging
- A unified utility maximization framework for federated search
- Modeling search engine effectiveness
Goal of Results Merging

Make different result lists comparable and merge them into a single list

Difficulties:
- Information sources may use different retrieval algorithms
- Information sources have different corpus statistics

Previous Research on Results Merging

Most accurate methods directly calculate comparable scores
- Use same retrieval algorithm and same corpus statistics
  (Viles & French, 1997)(Xu and Callan, 1998), need source cooperation
- Download retrieved docs and recalculate scores (Kirsch, 1997),
  large communication and computation costs
Research Problems (Results Merging)

Research on Results Merging

Methods approximate comparable scores

- **Round Robin** (Voorhees et al., 1997), only use source rank information and doc rank information, fast but less effective

- **CORI merging formula** (Callan et al., 1995), linear combination of doc scores and source scores
  - Use linear transformation, a hint for other method
  - Work in uncooperative environment, effective but need improvement
Research Problems (Results Merging)

Thought

Previous algorithms either try to calculate or to mimic the effect of the centralized scores

Can we estimate the centralized scores effectively and efficiently?

Semi-Supervised Learning (SSL) Merging (Si & Callan, 2002, 2003)

- Some docs exist in both centralized sample DB and retrieved docs
  From Centralized sampled DB and individual ranked lists when long ranked lists are available
  Download minimum number of docs with only short ranked lists
- Linear transformation maps source specific doc scores to source independent scores on centralized sample DB
Research Problems (Results Merging)

SSL Results Merging (cont)

In resource representation:
- Build representations by QBS, collapse sampled docs into centralized sample DB

In resource selection:
- Rank sources, calculate centralized scores for docs in centralized sample DB

In results merging:
- Find overlap docs, build linear models, estimate centralized scores for all docs
Research Problems (Results Merging)

Experiments

**Trec123**
- 3 Sources Selected
- 10 Sources Selected
- 50 docs retrieved from each source

**Trec4-kmeans**
- SSL downloads minimum docs for training

```plaintext
SSL
CORI, k=0.4
```
Abstract

Outline

- Introduction to federated search
- Main research problems
  - Resource Representation
  - Resource Selection
  - Results Merging
- A unified utility maximization framework for federated search
- Modeling search engine effectiveness
Unified Utility Framework

Goal of the Unified Utility Maximization Framework

Integrate and adjust individual components of federated search to get global desired results for different applications ✓

Simply combine individual effective components together ✗

High-Recall vs. High-Precision

**High-Recall**: Select sources that contain as many relevant docs as possible for information source recommendation

**High-Precision**: Select sources that return many relevant docs at top part of ranked lists for federated document retrieval

They are correlated but NOT identical, previous research does NOT distinguish them
Unified Utility Framework

UUM Framework

Estimate probabilities of relevance of docs

In resource representation:
- Build representations and CSDB
- Build logistic model on CSDB

In resource representation:
- Centralized doc scores → Probs of Relevance
- Prob of Rel

Centralized scores

Centralized Sample DB

Prob of Rel

Centralized scores

Doc Rank

CSDB Ranking

Resource Representation

Resource Selection

Engine 1

Engine 2

Engine N

Estimate probabilities of relevance of docs

\[ \theta^* = \{\hat{R}(d_{1j}), \hat{R}(d_{2j}), \ldots\} \]

\[ \hat{R}(d_{ij}) \] the prob of relevance for j\textsuperscript{th} doc from i\textsuperscript{th} source

• Use piecewise interpolation to get all centralized doc scores
• Calculate probs of relevance for all sampled docs
Research Problems (Unified Utility Framework)

Unified Utility Maximization Framework (UUM)

Basic Framework

Let \( \bar{\alpha} = \{\alpha_1, \alpha_2, \ldots\} \) indicate number of docs to retrieve from each source

\[
\theta = \{\hat{R}(d_{1j}), \hat{R}(d_{2j}), \ldots\}
\]

Estimated probs of relevance for all docs

\[
P(\theta | \vec{\alpha_s}, \vec{\alpha_c})
\]

prob of \( \theta \) given all available resource descriptions \( \vec{\alpha_s} \)

and centralized retrieval scores \( \vec{\alpha_c} \)

\[
U(\theta, \vec{u})
\]

utility gained by making selection \( \vec{u} \) when \( \theta \) is correct

\[
\vec{u} = \arg \max_{\vec{u}} U(\vec{u}, \theta)
\]
Research Problems (Unified Utility Framework)

Unified Utility Maximization Framework (UUM)

Resource selection for information source recommendation

High-Recall Goal:
Select sources that contain as many relevant docs as possible

Number of rel docs in selected sources

\[
\hat{u} = \arg\max \sum_{i} I(d_i) \sum_{j=1}^{N_{db_i}} R(d_{ij})
\]

Subject to: \( \sum_{i} I(d_i) = N_{sdb} \)

Solution: Rank sources by number of relevant docs they contain

\[
\text{Rel}_Q(i) = \sum_{j=1}^{N_{db_i}} R(d_{ij})
\]

Called Unified Utility Maximization Framework for High-Recall UUM/HR
Research Problems (Unified Utility Framework)

Unified Utility Maximization Framework (UUM)

Resource selection for federated document retrieval

High-Precision Goal:

Select sources that return many relevant docs as the top part

\[
\hat{u} = \arg \max \sum_{i} I(d_i) \sum_{j=1}^{d_i} R(d_{ij})
\]

Number of rel docs in top part of source

Retrieve fixed number of docs

Subject to: \( \sum_{i} I(d_i) = N_{sdb} \)

Number of sources to select

\( d_i = N_{rdoc} , \text{ if } I(d_i) \neq 0 \)

Solution: Rank sources by number of relevant docs in top part

Rel_Q(i) = \( \sum_{j=1}^{N_{rdoc}} R(d_{ij}) \)

Called Unified Utility Maximization Framework for High-Precision with Fixed Length

UUM/HP-FL
Unified Utility Maximization Framework (UUM)

Resource selection for federated document retrieval

A variant to select variable number of docs from selected sources

\[
\bar{d} = \arg \max_{d} \sum_{i} I(d_i) \sum_{j=1}^{d_i} R(d_{ij})
\]

Subject to:

\[
\sum_{i} I(d_i) = N_{sdb}
\]

\[
\sum_{i} d_i = N_{Total\_rdoc}
\]

\[
d_i = 10 \times k, \quad k \in [0, 1, 2, .., 10]
\]

Solution: No simple solution, by dynamic programming

Called Unified Utility Maximization Framework

for High-Precision with Variable Length

UUM/HP-VL
Research Problems (Unified Utility Framework)

Experiments

Resource selection for information source recommendation

![Graphs showing resource selection performance](image)
Research Problems (Unified Utility Framework)

Experiments

Resource selection for information source recommendation
Unified Utility Framework

Experiments: Resource selection for federated document retrieval

Trec123

3 Sources Selected

10 Sources Selected

Representative

SSL Merge

Graphs showing precision vs. document rank for different methods with three and ten sources selected.
Abstract

Outline

- Introduction to federated search
- Main research problems
  - Resource Representation
  - Resource Selection
  - Results Merging
- A unified utility maximization framework for federated search
- Modeling search engine effectiveness
Motivation of *Modeling Search Engine Effectiveness* for Federated Search

Ineffective search engines are common in real world (e.g., engines connected by FedStats that return unranked docs), may not *RETURN* relevant docs

In Resource Representation:
- Model search engine effectiveness by investigating rank consistency between individual engines and the effective centralized retrieval algorithm
- Learn mappings that transform rank from individual engine to rank of centralized retrieval algorithm

In Resource Selection:
- Select sources that can *RETURN* the largest amount of highly ranked relevant documents to accomplish the High-Precision goal

“*Modeling Search Engine Retrieval Effective*” (Si & Callan, SIGIR ’05)
Returned Utility Maximization
In Resource Representation:

Learn logistic model to estimate probability of relevance from training data

Effective centralized retrieval algorithm

For ith source, jth training query learn a rank transform mapping

The Mapping: $\phi_{ij} : d_{db\_i}(r_1) \rightarrow d_c(r_2)$
Returned Utility Maximization

In Resource Selection:

Estimate the utility can be returned from each source

Prob of rel of docs from ith engine
(ranked by centralized algorithm)

\[ \phi_{ij} : d_{db_i}(r_1) \rightarrow d_c(r_2) \{ R_i(d_c(1)), R_i(d_c(2)), \ldots, R_i(d_c(n)) \} \]

Prob of rel of docs from ith engine
(ranked by individual engine)

\[ \{ R_i(d_{db_i}(1)), R_i(d_{db_i}(2)), \ldots, R_i(d_{db_i}(m)) \} \]
Returned Utility Maximization Framework (RUM)

High-Precision Goal: Select sources that return most relevant docs

Retrieve fixed number of docs from selected sources

\[
\hat{d}_i = \arg\max_{d_i} \sum_i I(d_i) \sum_{j=1}^J \sum_{r=1}^{d_i} R_i(\phi_{ij}(d_{db,i}(r)))
\]

Subject to: \(\sum_i I(d_i) = N_{sdb}\)

\(d_i = N_{rdoc}, \text{ if } I(d_i) \neq 0\)

Solution: Rank sources by number of relevant docs they can return

\[
RU_i = \sum_{i=1}^J \frac{1}{J} \sum_{r=1}^{d_i} R_i(\phi_{ij}(d_{db,i}(r)))
\]
Experiments: Compare retrieval results with state-of-the-art algorithms CORI and Unified Utility Maximization that do not consider engine effectiveness.

Results on Trec123 with three types of effective engines and three ineffective ones.

![Graphs showing precision vs. doc rank for different scenarios]
Abstract

Outline

- Introduction to federated search
- Main research problems
  - Resource Representation
  - Resource Selection
  - Results Merging
- A unified utility maximization framework for federated search
- Modeling search engine effectiveness