

Discovering and Ranking Web Services with BASIL:

A Personalized Approach with Biased Focus

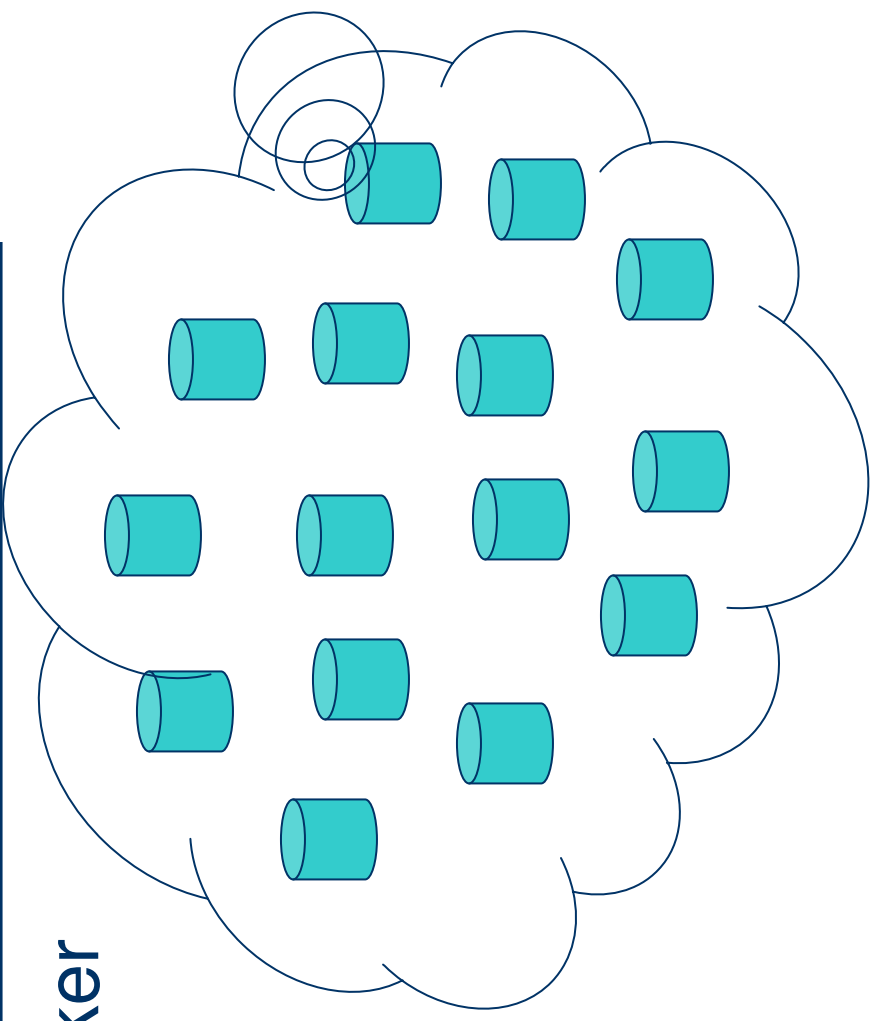
**James Caverlee, Ling Liu,
and Daniel Rocco**

College of Computing

Georgia Institute of Technology

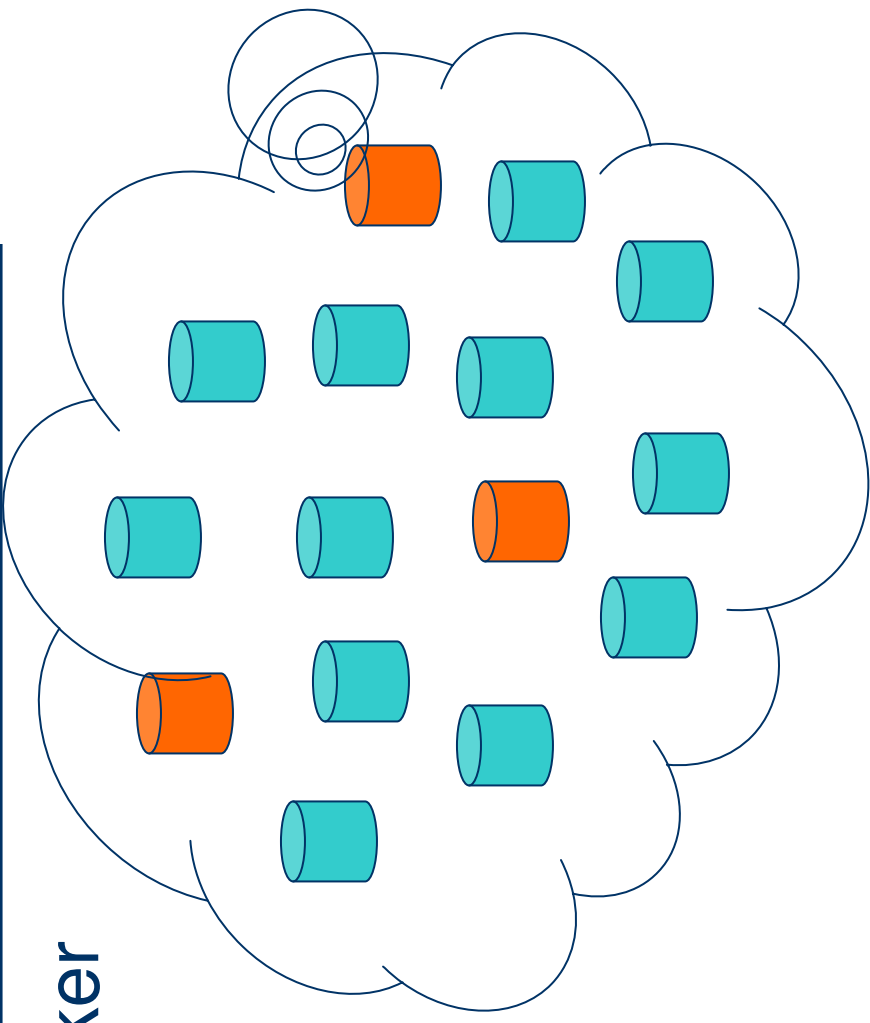
Category-based Service Discovery

- Find all stock ticker services:



Category-based Service Discovery

- Find all stock ticker services:



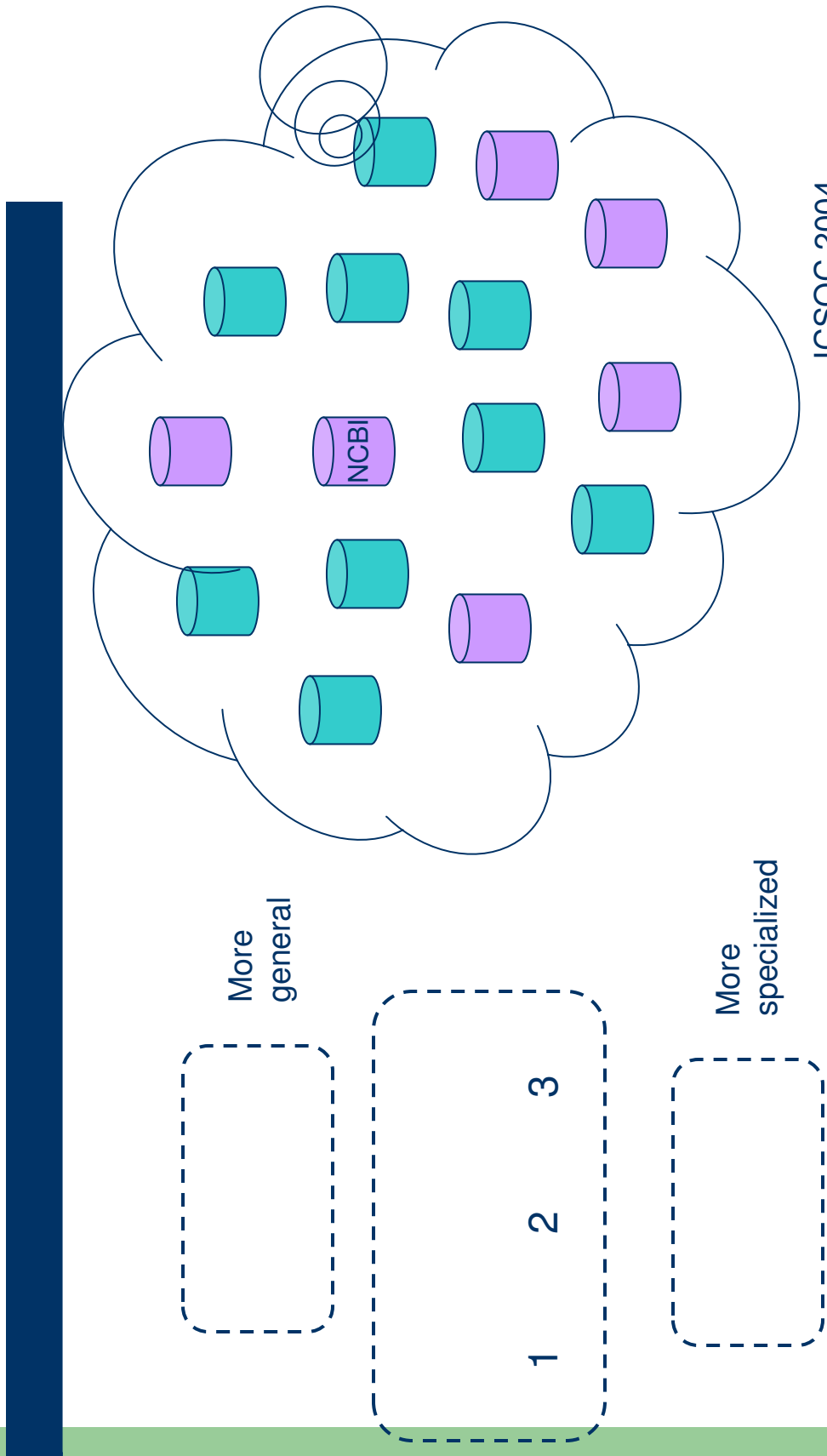
Category-based Service Discovery

- The UDDI approach
- Group services based on common properties
 - All stock ticker services
 - All services offered by New York companies
 - ...
- A user can search on properties or browse the registry to find candidate matches

Personalized Relevance-Based Service Discovery

- Identify services based on their relationships to other services
 - Not supported by today's registries
- Sample discovery tasks:
 - Find the top-ten services that offer more coverage than the BLAST services at NCBI
 - Which medical literature sites are more specialized than PubMed
 - ...

Personalized Relevance-Based Service Discovery



ICSOC 2004

Techniques for Service Discovery and Ranking

- Based on communities
 - Reputation systems
 - PageRank-style (?)
- Schema/Interface matching
 - Find the services with similar inputs, outputs
- Semantic matching
 - Using a markup like OWL
- **Instance/data matching**
 - Use the data that the service provides to better understand the service
 - Use that data to compare across services

Our Solution: BASIL

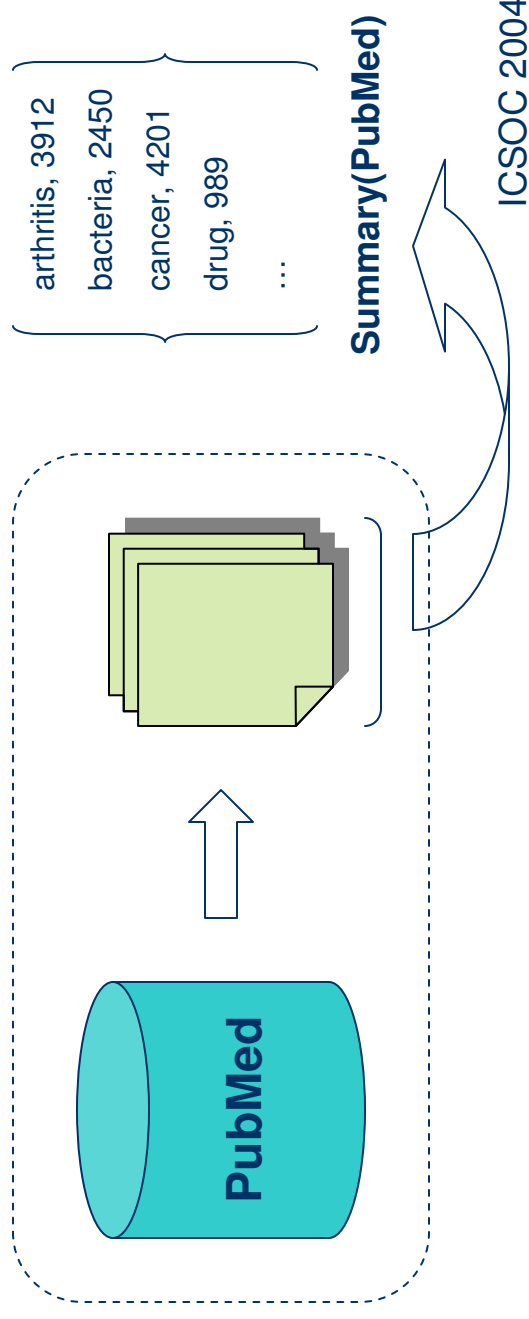
- **BiAsed Service dIscovery aLgorithm**
- Three key components:
 - Source-Biased Probing
 - Evaluation and ranking of services with Biased Focus
 - Identification of interesting relationships based on bi-lateral evaluation of biased focus
- Focuses on the nature and degree of **topical relevance**
- Avoids significant human intervention or hand-tuned categorization schemes

We focus on one type of web service

- **Data-intensive web services**
 - Access to huge amounts of data
 - Tools for searching, manipulating, and analyzing data
 - Examples: Amazon, Google, Lifesciences resources like BLAST (genetic sequence search)
- Unlike transactional services (e.g. for purchasing a box of pencils)

Modeling Data-Intensive Web Services

- Service Summary
 - Bag-of-words model
 - XML Tags and Text
- ActualSummary(S_i) = $\{(t_1, w_1), (t_2, w_2), \dots, (t_N, w_N)\}$

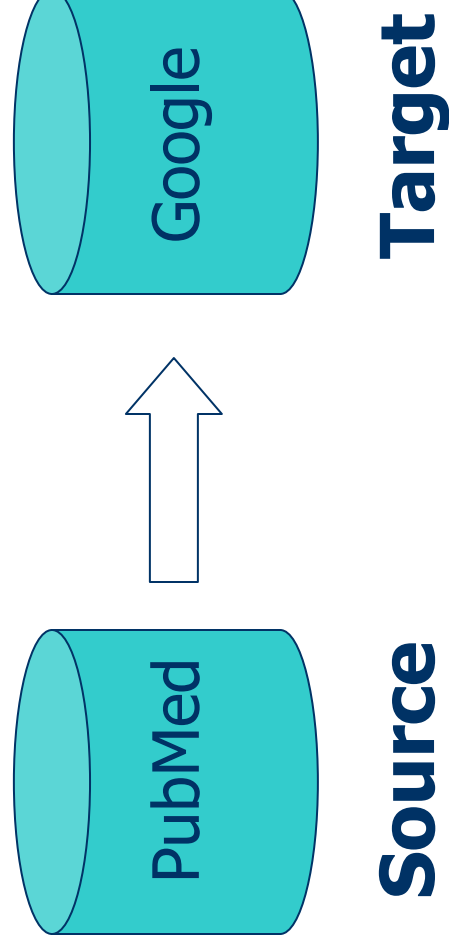


Estimating Service Summaries

- Query-based Sampling [Callan '99]
 - Send a query; retrieve top-m documents; repeat until stopping condition reached
 - EstSummary(PubMed) [only a fraction of all terms in Actual Summary]
 - Over text databases, need ~300 docs for high-quality estimated summaries
- Good at generating *overall summaries*
- But not necessarily good for *comparing summaries* (see paper)
 - Intuition: a service with broad coverage (like Google) will have few terms in common with a service with narrow coverage (like PubMed)

Source-Biased Probing

- Bias the estimate of the target towards the source of bias
 - $\text{EstSummary}_{\text{PubMed}}(\text{Google})$ vs. $\text{EstSummary}(\text{Google})$
- Hone in on what Google has in common with PubMed



Source-Biased Probing

```
SourceBiasedProbing(Source  $\sigma$ , Target  $\tau$ )  
  For target service  $\tau$ , initialize  $\text{ESUMMARY}_\sigma(\tau) = \emptyset$ .  
  repeat  
    Invoke the probe term selection algorithm  
    to select a one-term query probe  $q$  from the  
    source of bias  $\text{ESUMMARY}_\sigma(\sigma)$ .  
    Send the query  $q$  to the target service  $\tau$ .  
    Retrieve the top- $m$  documents from  $\tau$ .  
    Update  $\text{ESUMMARY}_\sigma(\tau)$  with the terms and  
    frequencies from the top- $m$  documents.  
  until Stop probing condition is met.  
  return  $\text{ESUMMARY}_\sigma(\tau)$ 
```

Figure 1: Source-Biased Probing Algorithm

Probe Selection

- Uniform random selection
 - Prob(selecting term j) = $1 / N'$
- Weighted random selection
 - Prob(selecting term j) = $w_j / \text{Sum}_i(w_i)$
- Weight-based selection
 - Select terms that occur the most times in all documents
 - Select terms that occur in the most documents
- **Focal term probing**

Probing with Focal Terms

- Instead of treating a source as a single collection of candidate probe terms, let's try to break the source up into rough groups of co-occurring terms
- Cluster terms (not documents)
 - $\text{Term}_j = \{(\text{doc}_1, w_{j1}), \dots, (\text{doc}_M, w_{jM})\}$
- Use off-the-shelf clustering algorithm to find k focal term groups
 - Simple KMeans, in this case

Probing with Focal Terms (2)

- Use round-robin selection to choose a probe from each focal term group

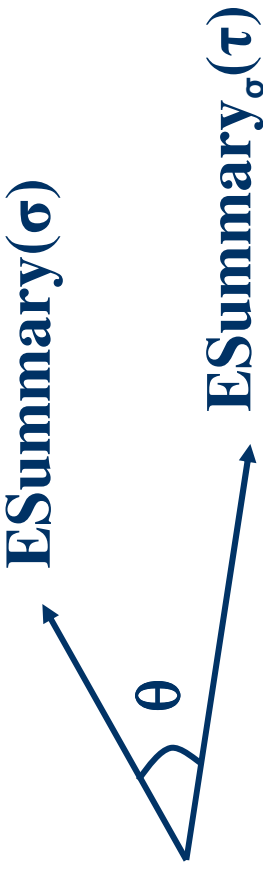
Table 1: Example Focal Terms for PubMed

1	care, education, family, management, ...
2	brain, gene, protein, nucleotide, ...
3	clinical, noteworthy, taxonomy, ...
4	experimental, molecular, therapy, ...
5	aids, evidence, research, winter, ...

Evaluating and Ranking Services

- **Biased Focus**
 - Captures the topical focus of a target on the source
 - $\text{focus}_{\text{source}}(\text{Target})$
- Should range from 0 (no focus) to 1 (complete focus)
- Not a symmetric measure; for example:
 - $\text{focus}_{\text{PubMed}}(\text{Google}) = \text{high}$
 - $\text{focus}_{\text{Google}}(\text{PubMed}) = \text{low}$

Cosine-Based Biased Focus

- **Cosine**
 - normalized inner product
 - Independent of the vector length
- 
- The diagram illustrates the cosine of an angle θ between two vectors. One vector is labeled $\mathbf{ESummary}_\sigma(\sigma)$ and the other is labeled $\mathbf{ESummary}_\tau(\sigma)$. The angle between them is marked with a double-headed arc and the Greek letter θ .

$$\text{Cosine_focus}_\sigma(\tau) = \left(\frac{\sum_{k=1}^N w_{\sigma k} w_{\tau k}^\sigma}{\sqrt{\sum_{k=1}^N (w_{\sigma k})^2} \cdot \sqrt{\sum_{k=1}^N (w_{\tau k}^\sigma)^2}} \right)$$

Other metrics
discussed in the paper

Identifying Interesting Relationships

- Consider two services: A and B
- Evaluate their relationship by understanding the focus of each with respect to the other
 - $\text{focus}_B(A)$ and $\text{focus}_A(B)$
- Relies on a family of lambda-parameters
- Example:
 - Let $\text{lambda_high} = 0.9$
 - if $\text{focus}_B(A) > 0.9$ and $\text{focus}_A(B) > 0.9$, then A and B are lambda-equivalent
- Of course, determining the appropriate lambda is tricky!

Experimental setup

- Two datasets:
 - Newsgroups
 - 780 collections
 - 100-16,000 documents in each
 - 2.5GB total
 - Web collection – ‘in the wild’
 - 50 real web databases
 - 50 docs collected from each

Probing Efficiency

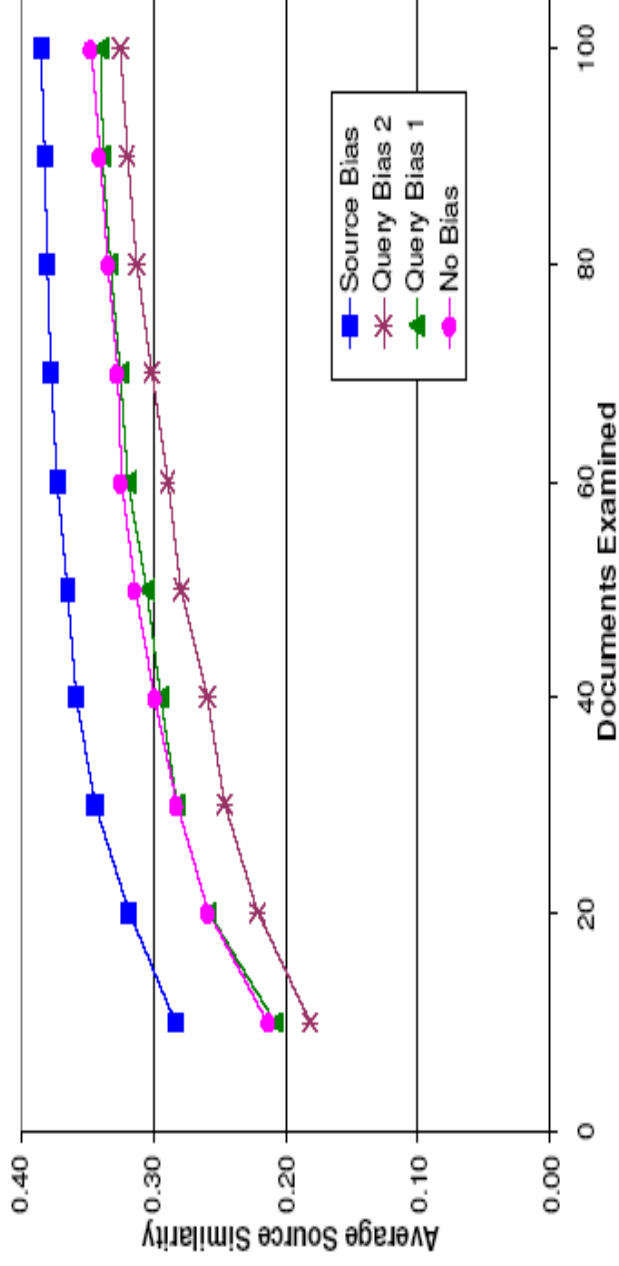


Figure 3: Probing Efficiency for 100 Pairs

SBP Identifies High Quality Documents

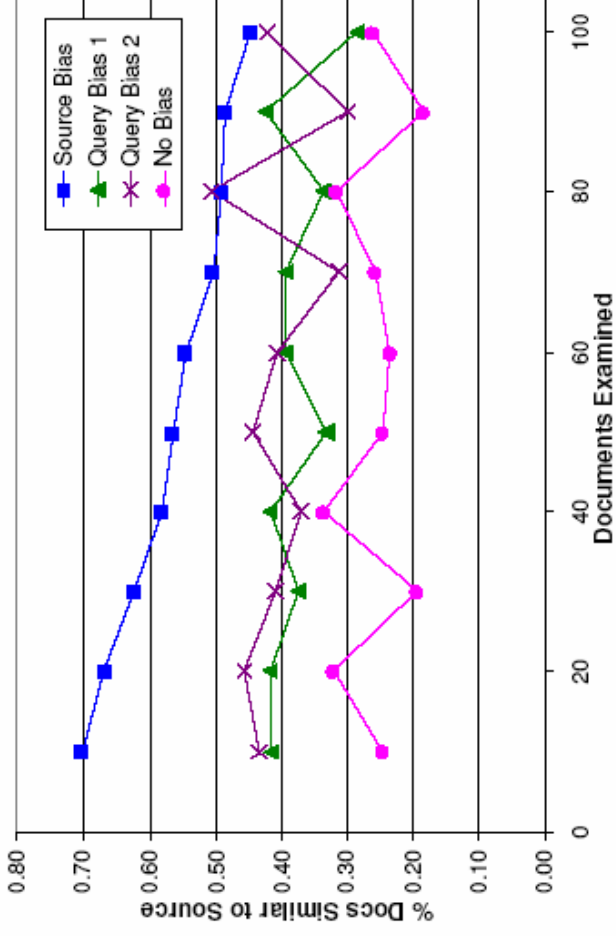
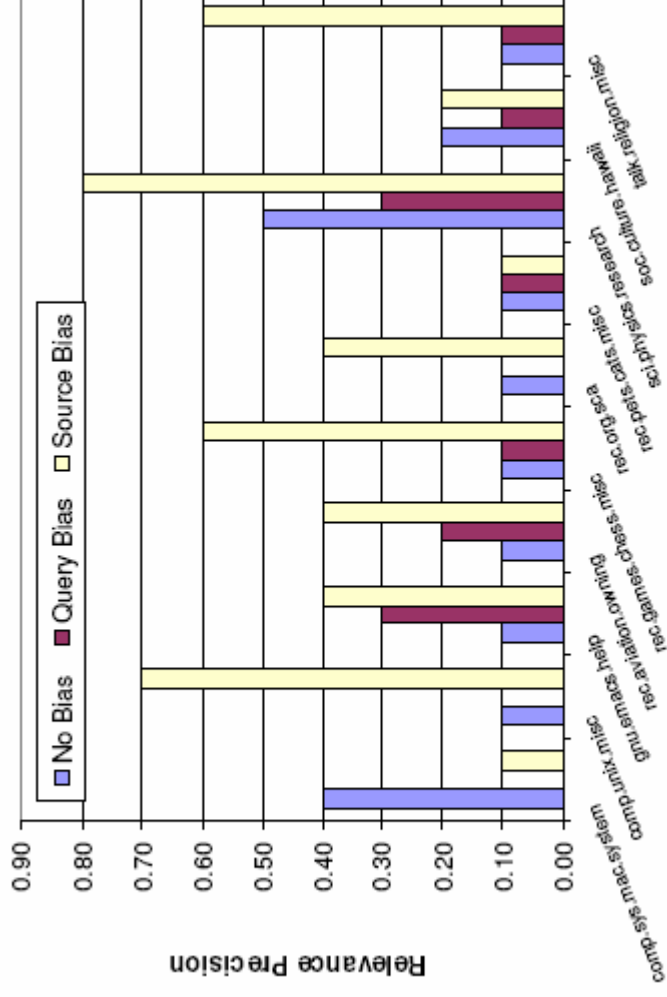


Figure 5: Average Document Quality for 100 Pairs

Precision For 10 Source Newsgroups



Ranking Web Sources

Table 2: Identifying Web Sources Relevant to PubMed

Query Bias	Source Bias
1. AMA	1. Open Directory (13)
2. WebMD	2. Google (27)
3. Linux Journal	3. About (11)
4. HealthAtoZ	4. WebMD (2)
5. DevGuru	5. AMA (1)
6. FamilyTree Magazine	6. HealthAtoZ (4)
7. Mayo Clinic	7. Monster (22)
8. Novell Support	8. Mayo Clinic (7)
9. Random House	9. Random House (9)
10. January Magazine	10. BBC News (12)

Relationships Relative to PubMed

Service (S)	URL	Description	$focus_{PM}(\hat{S})$	$focus_{PM}(PM)$	Relationship
WebMD	www.webmd.com	Health/Medical	0.23	0.18	λ -equivalent
AMA	www.ama-assn.org	Health/Medical	0.19	0.16	λ -equivalent
HealthAtoZ	www.healthatoz.com	Health/Medical	0.18	0.16	λ -equivalent
Open Directory	dmoz.org	Web Directory	0.44	0.08	λ -superset
Google	www.google.com	Web Search Engine	0.37	0.10	λ -superset
About	www.about.com	Web Channels	0.25	0.08	λ -superset
Monster	www.monster.com	Jobs	0.14	0.08	λ -overlap
Mayo Clinic	www.mayoclinic.com	Health/Medical	0.12	0.11	λ -overlap
Silicon Investor	www.siliconinvestor.com	Finance	0.03	0.04	λ -complement
Usenet Recipes	recipes2.alastra.com	Recipes	0.02	0.03	λ -complement

More in the paper!

Conclusions

- Introduced techniques to support Personalized Relevance-Based Service Discovery
 - Source-biased probing
 - Focal term probing
 - Source-biased ranking (with biased focus)
 - Identification of relationships

Open issues

- Exploiting structure
 - E.g. for schema matching, use of ontologies, etc.
- More advanced probing techniques
- Fine-grained inter-service analysis
- Better understanding of complex service computations (e.g. correlating input to output)
- Could extend this “personalization” approach to consider other factors as well

Thank You!

