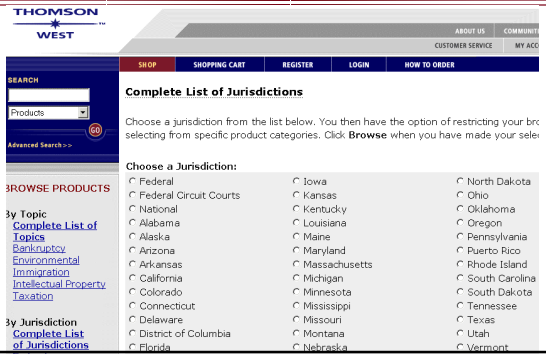
Distributed Information Retrieval (Searching for Needles in a Field of Haystacks)

Jamie Callan
Language Technologies Institute
Carnegie Mellon University
callan+@cs.cmu.edu

Case 1: Large Commercial Information Provider



Case 1: Large Commercial Information Provider

Task: Provide rapid access to several terabytes of information
Difficulty: Long queries, many Web-search optimizations not applicable

- Each information source is stored separately
  - E.g., potentially hundreds or thousands of databases
  - Simplifies maintenance, increases speed
- Searching all databases is computationally expensive
- Missing relevant databases is costly
  - Customers don't find what they need
- One organization, many distinct databases
- How does a customer know which to search?

Case 2: Customer Support

- **A customer-service organization may support many products**
  - Possibly for different client companies
- **Information about different product lines may be stored in different databases**
  - For administrative reasons
    - » E.g., to simplify maintenance
  - For contractual reasons, …
    - » E.g., don't mix information about Microsoft and Apple
  - For intellectual property reasons
    - » E.g., to better control access to some information
- **Which database should a new customer service request go to?**

Case 3: Competitive Analysis

Task: Gather information on a competitor's products

Difficulty: Useful information is scattered across many sources

- Sales literature: *On the competitor's Web site*
- Product comparisons:
  - Media: *Wall Street Journal, Consumer Reports, PC Magazine*
  - Online retailers: *http://www.cnet.com/*
- Common problems:
  - Customer support database: *On the competitor's Web site*
  - Newsgroups: *http://www.dejagnus.com/*
- How does an analyst know where to search?

Case 4: Distributed Wireless Networks

Environment: A student using limited-range wireless device on a campus that has few wireless stations needs to get a bus schedule

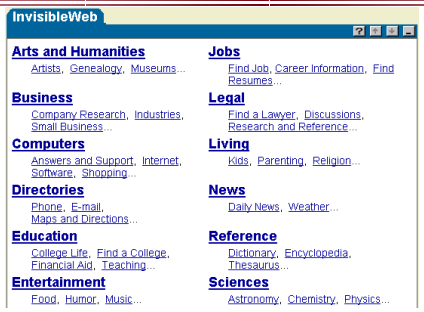
Difficulty:

- The student may not be in range of a wireless station
- The student may not know the location of a database that contains the desired information
- How does the student get the desired information?

Case Summaries

1. **Large Information Provider (West)**
   - Different information stored in many different databases
   - Each database managed by the same organization
2. **Customer Support Bureau**
   - Information stored in a few different databases
   - Each database managed by the same organization
3. **Competitive Analysis**
   - Different information stored in many different places
   - Each database managed independently
4. **Distributed Wireless Networks**
   - User has no access to centralized resources

Multi-Database Solutions: The Browsing Model (#1)



Organize databases into a classification hierarchy

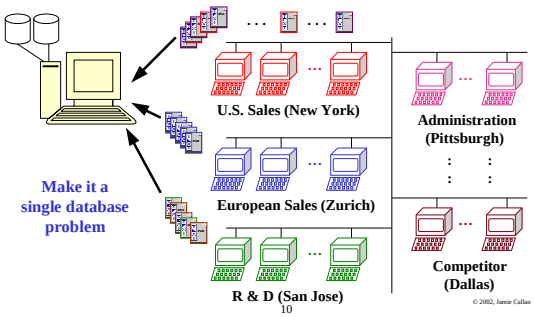
Manual navigation

Multi-Database Solutions: The Browsing Model (#1)

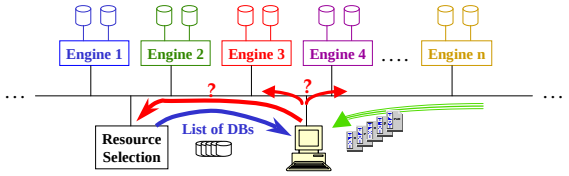
Static ranking

Manual database selection

Multi-Database Solutions: The Web-Search Model (#2)



Multi-Database Solutions: Distributed Information Retrieval (#3)



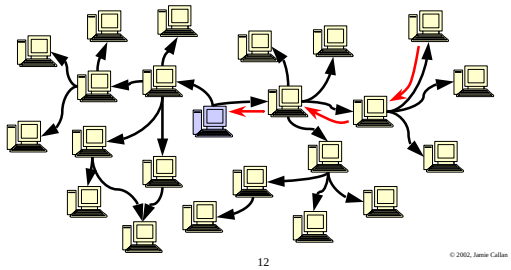
Dynamic DB ranking

Automatic or manual database Selection

- Sites manage content independently
- Uncooperative engines
- Possibly unknown software
- DB contents not visible

Solution #4: The Message-Passing Model

"Tell your neighbors what you need, they tell their neighbors, etc, until a solution is found"



Multi-Database Solutions: A Brief Summary

- **Web-search (single database) model**
  - Doesn't scale
  - Doesn't work well when information isn't free
    - » Because copying of high-value information is restricted
- **Browsing model**
  - Predictable results for predictable information needs
  - Time-consuming if hierarchy is deep
  - Time-consuming if hierarchy doesn't match information need

Our Focus

- **Directory & Message-passing models**
  - Larger scale problems
  - Doesn't require ability to copy each database
  - Less user effort

Directory Model: Major Approaches

Single Site / LAN / Few Sites

- Relevant document distribution (RDD)
- Query Clustering

Many Sites / WAN / Internet

- GlOSS, gGlOSS, hGlOSS, vGlOSS
- Inference nets (CORI)

Single Site / LAN / Few Sites: Relevant Document Distribution (RDD)

Assumption: New queries often look a lot like old queries

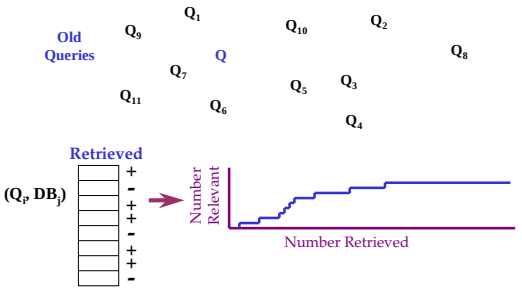
Approach:

- Find a few old queries that are similar to the new query
  - similarity is determined by vector similarity of query term
- Look up the effectiveness of each database for these old queries
- Compute an estimated effectiveness of each database for the new query as the average of its effectiveness for the similar old queries

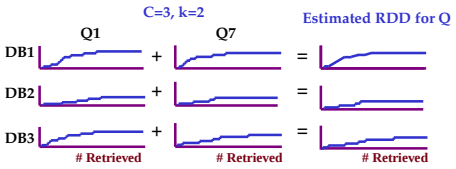
This method is designed for environments where there is little variety in queries and new collections are not added often

(Voorhees, et al., 1995)

Single Site / LAN / Few Sites: Relevant Document Distribution (RDD)



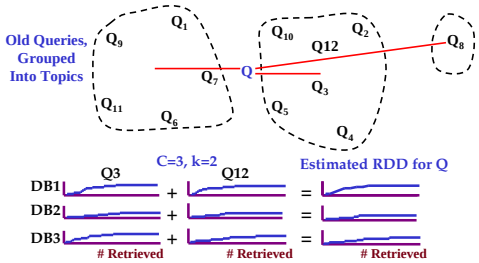
Single Site / LAN / Few Sites: Relevant Document Distribution (RDD)



- New query Q is most similar to old queries Q1 and Q2
- The estimated RDD is the average of the RDDs for the k most similar queries
- **Note:** Effectiveness depends on training queries being similar to expected queries

Single Site / LAN / Few Sites: Query Clustering

Assumption: Many queries, but fewer distinct topics



(Voorhees, et al., 1995)

Directory Model: Summary of Techniques for a Few Sites

Single Site / LAN / Few Sites

- Relevant document distribution (RDD)
- Query Clustering

Common Assumptions:

- Few collections
- Relatively stable collections,
- Relatively stable information needs (queries)
- Considerable overlap in query contents
- Good training data

Directory Model : Techniques for Many Sites

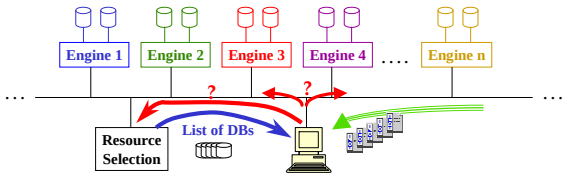
Assumptions:

- Many collections (hundreds, thousands, tens of thousands, …)
  - Perhaps dispersed widely, managed independently, expensive, …
- Collections are homogeneous and heterogeneous
- Queries are heterogeneous and can't be predicted in advance
- Training data is not practical
  - Many collections
  - Collections constantly being added, deleted, changed

Solution:

- Content-based database selection (gGlOSS, CORI, CVV, Napster, …)

Multi-Database Solutions: Directory Model



Dynamic DB ranking

Automatic or manual database Selection

- Find out what each database contains
- Decide where to search for this query
  - Pick DBs similar to query
- Search one or more databases
- Merge results from different searches

Not necessarily easy tasks!

Directory Model: Issues

- **Site description:** Contents, search engine, services, etc
  - Issue: cooperative vs. uncooperative sites
- **Resource ranking**: ranking resources by how likely to contain desired content
- **Resource selection:** selecting the best subset from a ranked list
- **Searching:** Interoperability
- **Result merging**: Merging a set of document rankings
  - different underlying corpus statistics
  - different search engines

Directory Model: Database Representation

1987 WSJ (132 MB)	1991 Patent (254 MB)	1989 AP (267 MB)
stobb (1)	sto (1)	sto (7)
stochast (1)	stochast (21)	sto1 (4)
stock (46704)	stochiometr (1)	sto3 (1)
stockad (5)	stociometr (1)	stoaker (1)
stockard (3)	stock (1910)	stoand (1)
stockbridg (2)	stockbarg (30)	stober (6)
stockbrok (351)	stocker (211)	stocholm (1)
stockbrokag (1)	stockholm (1)	stock (28505)
stockbrokerag (101)	stockigt (4)	stock' (6)
stockdal (8)	stockmast (3)	stockad (35)
stockhold (970)	stockpil (7)	stockard (12)

Word + frequency models are most common (gGlOSS, CORI, language models, CVV, …)

Directory Model: Language Model Acquisition

Cooperative protocols are the state of the art

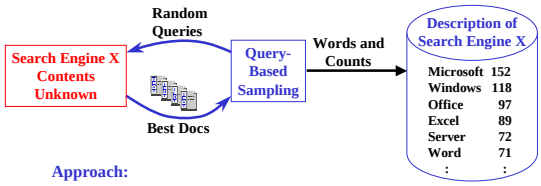
- Each database provides its language model upon request
  - e.g., STARTS (proposed extension to Z39.50), XML

Weaknesses:

- DBs that can't or won't cooperate (e.g., legacy databases, competitors)
- Databases that misrepresent about their contents
- Different approaches to representing text (incompatible models)
  - stemming, stopping, normalization, proper names, …

Cooperative protocols are most appropriate in environments controlled by a single party

Acquiring Resource Descriptions By Query-Based Sampling



Approach:

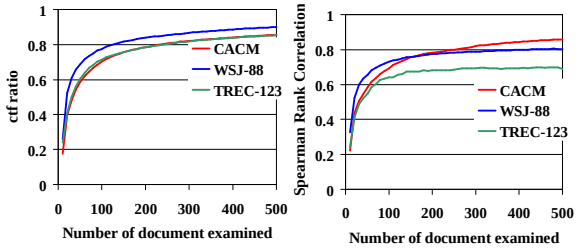
- Obtain sample documents by running queries on a database
- Build resource descriptions by analyzing retrieved documents

Advantages:

- No explicit cooperation assumed beyond normal service
- Resource selection service controls how resources are described

(Callan, et al., 1999; Callan & Connell, 2001)

Query-Based Sampling : Coverage of Learned Vocabulary



- Query terms selected randomly from learned language models
- 4 documents examined per query

The Microsoft Customer Support Database

Term	Avg tf	Term	Avg tf	Term	Avg tf	Term	Avg tf
project	10.924	visual	5.273	articles	4.121	dialog	3.515
excel	8.750	beta	4.986	setup	4.094	command	3.504
office	8.565	service	4.983	mail	4.067	following	3.387
works	7.389	basic	4.903	users	4.042	windows	3.369
server	7.271	file	4.867	set	3.948	new	3.369
word	7.221	nt	4.845	application	3.919	settings	3.317
table	6.639	field	4.729	product	3.890	example	3.152
printer	6.507	access	4.554	menu	3.840	version	3.147
foxpro	6.486	print	4.550	text	3.717	message	3.119
database	6.117	data	4.322	software	3.621	information	3.076
microsoft	5.736	internet	4.268	code	3.617	select	3.072
object	5.637	error	4.217	name	3.611		
user	5.297	box	4.213	system	3.544		

Directory Model: Evaluation

Query 63: Identify a machine translation system being developed or marketed in any country. Identify the developer or vendor, name the system, and identify one or more features of the system.

Answer Key

Rank	Collection	Score	RelDocs
1	1989-90 Ziff Davis	0.571	153
2	1989-90 Ziff Davis	0.569	34
3	1991-92 Ziff Davis	0.563	77
4	1991 Patent	0.407	0
5	1989 AP	0.404	1
6	1988 AP	0.401	4
7	1988 WSJ	0.399	1
8	1987 WSJ	0.399	0
9	1988 Federal Register	0.341	0
10	1989 Federal Register	0.337	0

Techniques for Many Sites: gGlOSS

Assumption: The vector-space retrieval model can apply to collections (represent each collection as a "big document")

Approach:

- Represent each document by a vector (standard vector space model)
- Represent each collection by its centroid vector
  - i.e., a collection is represented by a (mythical) "average" document
- Standard inner product similarity measure of query to each collection
- Rank collections by their similarity to each query

Automatic creation, consistent, dynamic grouping, good for most information needs, works best for homogenous collections

(Gravano, et al., 1995)

Techniques for Many Sites: CORI (Inference Networks)

Assumption: The inference net retrieval model can apply to collections (represent each collection as a "big document")

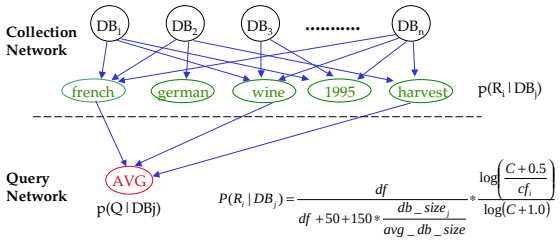
Approach:

- Represent each collection by an inference net (standard inference net model)
- Standard tf.idf similarity measure of query to each collection
- Rank collections by their similarity to each query

Automatic creation, consistent, dynamic grouping, good for most information needs, works well for most collections

(Callan, et al., 1995; Callan, 2000)

Resource Ranking: CORI (Inference Networks)



$$P(R_i \mid DB_j) = \frac{df}{df + 50 + 150 * \frac{db\_size_j}{avg\_db\_size}} * \frac{\log\left(\frac{C + 0.5}{cf_i}\right)}{\log(C + 1.0)}$$

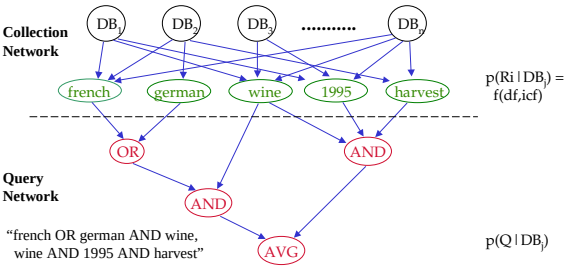
"french wine harvest"

df : Number of documents in DB_j containing R_i

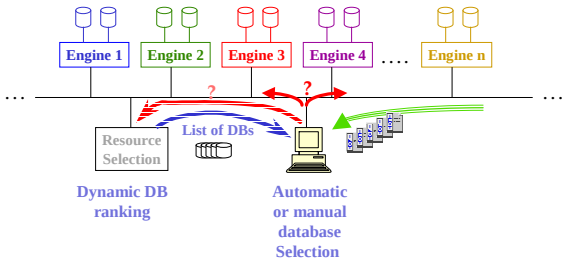
cf : Number of databases containing R_i

C : Number of databases

Resource Ranking: CORI (Inference Networks)



Multi-Database Solutions: Directory Model



Directory Model: Summary of Techniques for Many Sites

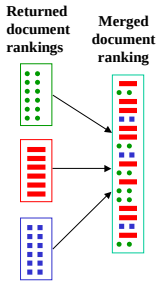
Many Sites / WAN / Internet

- gGlOSS and hGlOSS
- CORI (inference nets)
- Many other algorithms, too (CVV, language models, Napster, …)

Common Assumptions:

- Many collections
- Relatively stable collections
- No assumptions about range of queries
- No training data required
- Minimize network costs
- Not necessarily good at "needle in haystack" queries

Directory Model: Merging Results



- **Global idf + simple score merge**
  - Effective, but limited to a single organization
- **Download & rerank documents at client**
  - Costly in communication, computation
- **Heuristic reranking of scores at client**
  - effective (for Inquery) but ad-hoc
- **Query-specific, semi-supervised learning**
  - Resource selection service assists in merging
  - Fast, very effective, no cooperation among engines
  - Few assumptions
  - Current state-of-the-art

Does All This Stuff Work?

Consistent results by three research groups on several testbeds

- **Types of data:** TREC (newswire, magazine, gov), Web, U.S. Patents
- **Numbers of databases:** O(10) to O(1,000) databases
- **Amount of data:** 2 gigabytes to 55 gigabytes
- **Individual database sizes:** Constant, skewed

Results:

- Searching a few automatically-chosen databases produces results similar to searching one massive database
- Using sampled data produces results similar to using complete data

CORI Resource ranking used by West Publishing (Thomson)

Experimental Results (A Sample)

SYM-236 testbed

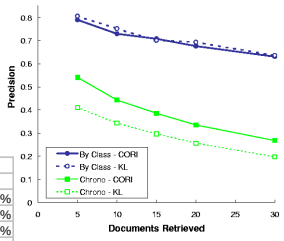
Precision	Central	CORI 5 DBs	CORI 10 DBs
5 docs	64.0	56.8	55.4
10 docs	60.0	53.7	55.3
15 docs	59.3	50.6	53.6
20 docs	57.4	48.3	52.0
30 docs	53.4	37.9	43.5
100 docs	48.4	29.7	35.6

(Powell, et. al., 2000)

100 DB testbed
10 DBs searched

Precision	CORI Full	CORI Sampled	CORI Diff
5 docs	0.4440	0.4440	+0.0%
10 docs	0.4100	0.3920	-4.4%
15 docs	0.3987	0.3627	-9.0%
20 docs	0.3740	0.3470	-7.2%
30 docs	0.3560	0.3267	-8.2%
100 docs	0.2720	0.2576	-5.3%

(Callan, et. al., 2001)



400 DB testbed (U.S. Patents) (Larkey, et. al., 2000)

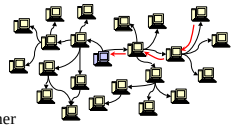
Directory Model: Major Approaches

Single Site / LAN / Few Sites
- Relevant document distribution (RDD)
- Query Clustering

Many Sites / WAN / Internet
- GlOSS, gGlOSS, hGlOSS, vGlOSS
- Inference nets (CORI)

Solution #4: The Message-Passing Model

Assumptions:
- No centralized directory services
- Nodes can communicate directly
  - "peer to peer"
- Nodes can decide for themselves whether they satisfy an information need



Solutions:
- Content-free routing (Gnutella)
- Content-based routing (FreeNet, FastTrack?)

Message-Passing Model: Content-Free Routing (Gnutella)

- **When a request for information is received**
  - If the node has the desired information, return it (or a pointer to it)
  - If the message time-to-live (TTL) is zero, discard it
  - Otherwise, decrement TTL and forward the request to several nodes
- **Advantages:**
  - Very robust, no central site vulnerable to attack
  - Handles diverse requests
- **Disadvantages:**
  - One request can generate a massive amount of network traffic
  - Two identical requests may get different answers

Message-Passing Model: Content-Based Routing (FreeNet)

- **Nodes observe the traffic they route**
  - Learning which types of requests other nodes can satisfy
- **Nodes (eventually) route messages (more) directly to nodes that can satisfy them**
- **Advantages:**
  - Very robust, no central site vulnerable to attack
  - Less network traffic, better scaling
  - More consistent answers to identical requests
- **Disadvantages:**
  - Each node must have a basic understanding of the content in each message it routes

Message-Passing Model: Content-Based Routing (KaZaA)

- **Some nodes with spare bandwidth are designated "supernodes"**
  - Supernodes learn contents of other nodes in a region of the network
- **Messages are routed first to nearest supernodes**
  - They route messages to nodes containing desired content
- **Advantages:**
  - Very robust, no central site vulnerable to attack
  - Less network traffic, better scaling
  - More consistent answers to identical requests
- **Disadvantages:**
  - Each node must have a basic understanding of the content in each message it routes

Message Passing Models: Summary of Techniques

Approaches:

- Content-free routing
- Content-based routing

Common Assumptions:

- No centralized directories
  - Possibly dynamic "regional"directories
- Direct communication among nodes
- No assumption about database contents
- No assumptions about range of queries

Handling Multiple Databases: Summary of Approaches

- **Web-search model**
  - Minimal user effort
  - Doesn't handle some common situations
  - Handles large-scale problems
  - Many commercial options (e.g., Google, AltaVista)
- **Browsing model**
  - Considerable manual effort for provider and user
  - Handles a wide range of information types
  - Scale is an issue
  - Many commercial options (but most of the work is manual)
- **Directory models**
  - Minimal user effort
  - Handle a wide range of text information
  - Some are high precision
  - Some handle large-scale problems
  - Some commercial options (e.g., Verity), but still developing
- **Message-passing model**
  - Minimal user effort
  - Handles a wide range of information types
  - Scale is an issue
  - Free options (Gnutella, FreeNet)
  - Some commercial options (FastTrack)

Distributed Retrieval: A Return to Cases

1. Very Large Database (West)
- Many different databases, each DB managed by same organization

2. Customer Service Bureau
- Few different databases, each DB managed independently

3. Competitive Analysis
- Many different databases, each DB managed independently

4. Distributed wireless networks
- Nodes in range of other nodes, but not central site, DBs managed independently

Which techniques are appropriate for each case?

For More Information

Available at http://www.cs.cmu.edu/~callan/

- J. Callan. "Distributed information retrieval." In W. B. Croft, editor, *Advances in Information Retrieval* (pp. 127-150). Kluwer Academic Publishers. 2000.
- J. Callan and M. Connell. "Query-based sampling of text databases." *ACM Transactions on Information Systems.* 19(2), pp. 97-130. ACM. 2001.
- J. Callan, M. Connell, and A. Du. "Automatic discovery of language models for text databases." In *Proceedings of the ACM-SIGMOD International Conference on Management of Data*, (pp 479-490), Philadelphia: ACM. 1999.
- J. P. Callan, Z. Lu, and W. B. Croft, W.B. "Searching distributed collections with inference networks." In *Proceedings of the Eighteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, (pp. 21-28), Seattle, WA, 1995. .
- L. Larkey, M. Connell, and J. Callan. "Collection selection and results merging with topically organized U.S. patents and TREC data". In *Proceedings of the 9th International Conference on Information and Knowledge Management (CIKM)* (pp. 282-289). ACM. 2000.
- A.L. Powell, J.C. French, J. Callan, M. Connell, and C.L. Viles. "The impact of database selection on distributed searching." In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 232-239). 2000.
- L. Si and J.Callan. "Using sampled data and regression to merge search engine results." In *Proceedings of the Twenty Fifth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. To appear.