

---

## Peer-to-Peer Information Retrieval Using Self-Organizing Semantic Overlay Networks

*by* Chunqiang Tang (Univ. Rochester)  
Zhichen Xu (HP Laboratories)  
Sandhya Dwarkadas (Univ. Rochester)

---

*Presenter:* Michela Becchi

*Discussion Leader:* Christoph Jechlitschek

---

## Motivation

High volume and exponential growth of digital information motivates need for scalable and accurate IR systems

## State of Art Summary

- Centralized indexing (Napster)
  - Single point of failure
  - Performance bottleneck at indexing server
- Flooding based techniques (Gnutella)
  - Network bandwidth
  - Variance in system response
- Heuristic based techniques
  - Failure in retrieving all important documents

## Goal

- Distributed architecture
  - Scalability, fault tolerance, ...
- Efficiency
  - Limitation to number of visited nodes
- Accuracy
  - Retrieval of all significant results
- Content searches in natural language (no simple keyword match)

## Idea

- Semantic Overlay
  - Distance in network prop. semantic dissimilarity
- IR algorithms (VSM and LSI) to Peer-to-Peer Environments
- CAN
  - Mapping overlay network to physical nodes

## Vector Space Model (VSM)

- Documents and queries: term vectors  
( $w_1, w_2, \dots, w_t$ )
- $W_i = TF * IDF$ 
  - TF=term frequency
  - Inverse document frequency
- Ranking of documents according to similarity between document vector and query vector ( $\cos\theta$ )
- Problems: synonyms and noise in documents

## Latent Semantic Indexing (LSI)



$$A = U \Sigma V^T$$

(txd)



$$A_l = U_l \Sigma_l V_l^T$$

- l largest singular values kept
- $V_l \Sigma_l$  rows: semantic vectors

- Elimination of noise and variability in word usage
- Documents and queries: vectors in *semantic space*

Michela Becchi 09/22/2005

7

## Semantic space

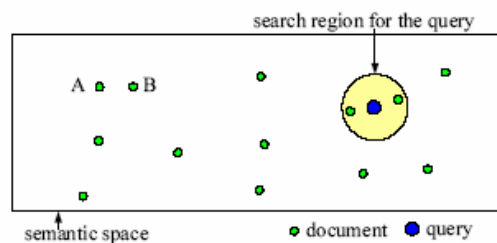


Fig. source [1]

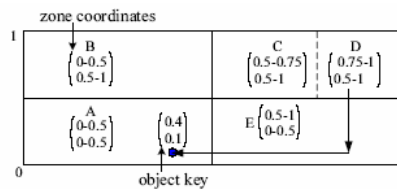
- Map semantic space to physical nodes
- Decentralized nearest-neighbor search

Michela Becchi 09/22/2005

8

## Content Addressable Networks (CAN)

- Distributed hash table (DHT)
- Maps *keys* into *values*
  - Keyword searches
- Partitions *d*-dimensional Cartesian spaces into *zones*, each one assigned to a *node*



- Object location: routing to node according to key
- Node join: route to zone containing a given point and splitting it

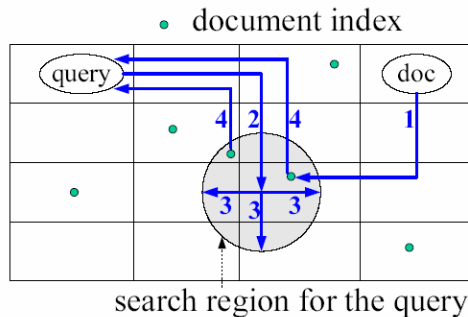
Fig. source [1]

Michela Becchi 09/22/2005

9

## Basic pLSI (distributed LSI)

- Dimensionality of CAN = LSI semantic space
- Key: semantic vector, value: Index (URL, ...)



1. Doc  $\rightarrow V_a \rightarrow \text{zone}_a \rightarrow \text{node}_a$
2. Query  $\rightarrow V_q \rightarrow \text{zone}_q \rightarrow \text{node}_q$
3. Query flooding ( $r$ )
4. Results

Fig. source [1]

Michela Becchi 09/22/2005

10

## PB1: Dimensionality mismatch

- Number of nodes  $\ll l$  (dim. LSI sem space)
  - $l \geq \log_2(n)$  and zones evenly partitioned  $\Rightarrow$ 
    - Only  $\log_2(n)$  dimensions partitioned
    - Search space along unpartitioned dimensions not reduced

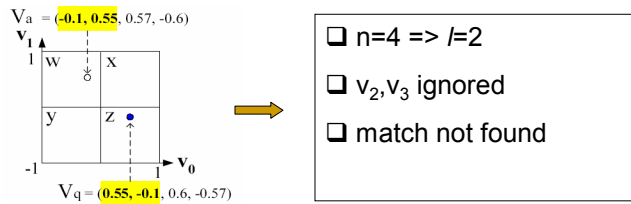
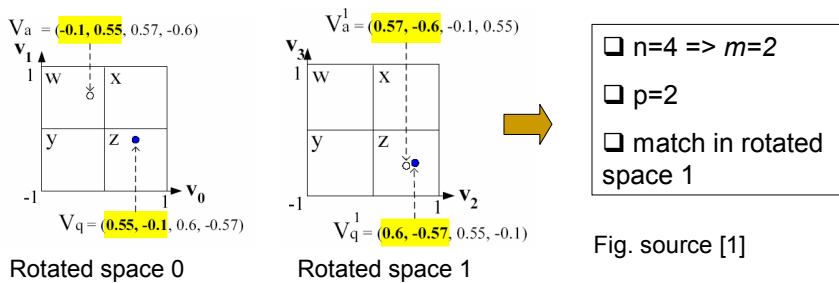


Fig. source [1]

## SOL1: Rolling-Index

- $V = (v_1, v_2, v_3, \dots, v_l)$   
 $V^i = (v_{im}, \dots, v_{i, m-1})$  *Rotated Semantic Vectors*  
 $V^i = (v_{im}, \dots, v_{i, m-1}), i=0, \dots, p-1$  *Vectors*
- $p$  rotated spaces
  - pLSI algorithm executed  $p$  times



Rotated space 0

Rotated space 1

Fig. source [1]

## PB/SOL2: Balancing Index Distribution

- PB: Uneven distribution of indexes
- SOL: *Content-aware node bootstrapping*
  - Routing according to randomly rotated semantic vector of a document to publish
  - Even splitting of the zone along lowest unpartitioned dimension
- Effects:
  - Balanced index distribution
  - Index locality (if node publishes similar documents)
  - Query locality (if locality between documents and queries)

Michela Becchi 09/22/2005

13

## PB3: Content-Directed Search

- PB: Search space grows with data dimensionality
  - Growing nearest-neighbor distance
  - High-dimensional data spaces sparsely populated
- Content-Directed Search

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

□N={8, 12, 14, 18}

□N={8, 12, 14, 18} => a

□N={8, 12, 18, **9, 15, 19**}

□N={8, 12, 18, 9, 15, 19} => b

□N={8, 12, 18, 15, 19, **4, 10**}

□N={8, 12, 18, 15, 19, 4, 10} => c

Fig. source [1]

Michela Becchi 09/22/2005

14

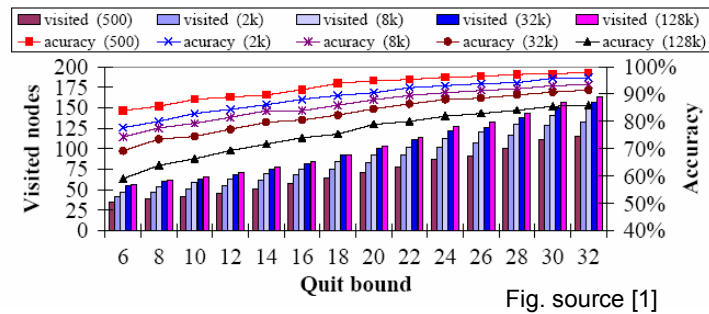
## Content-Directed Search (cont'd)

- Summary vector  $U^i[Z]$ : local indexes and recent queries processed by node  $Z$
- Samples of semantic vectors from neighbors (similar to  $U^i[Z]$  and random)
- Flooding guided by neighbor information
- Dynamic quit threshold  $T$ 
  - Decreasing with space ID and during search
- Possible indexes replication
- Possible parallel processing

## Experimental setup

- TREC-7 and TREC-8 corpus
  - 528,543 documents, 2GB size
- SMART to index
- Metrics:
  - Number of visited nodes
  - Accuracy (pLSI/centralized LSI)

## Varying system size



- Slow increase of # visited nodes
- Effect of quit bound on accuracy

## Varying number of returned documents

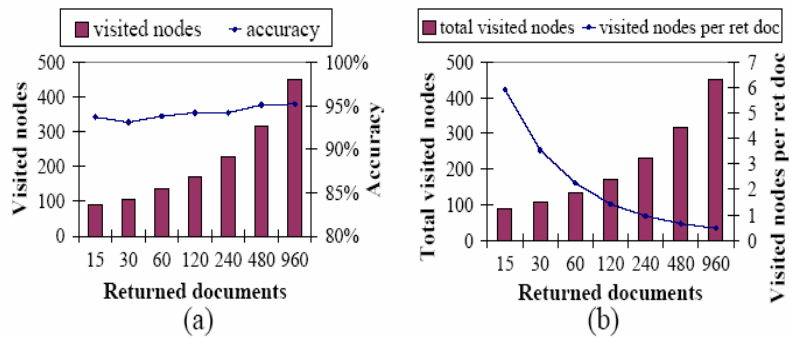
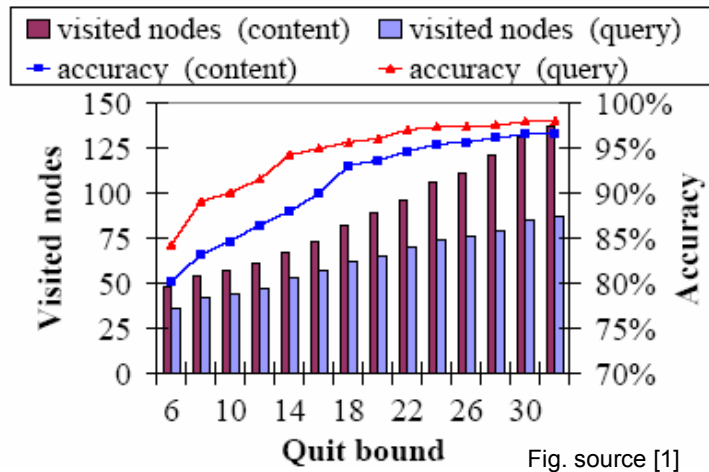


Fig. source [1]

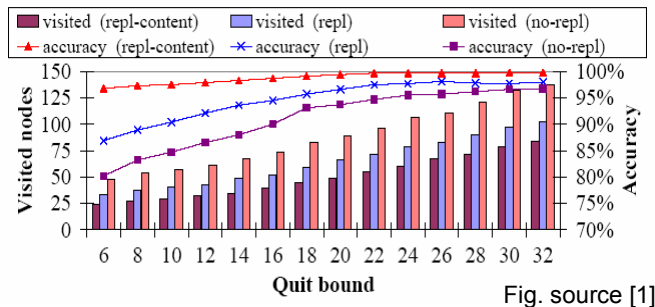
## Content-directed searches



Michela Becchi 09/22/2005

19

## Replication



- no-repl: do not replicate
- repl: replicates neighbors' content
- repl-content: replicates neighbors' content and its neighbors' samples

Michela Becchi 09/22/2005

20

## Rotated spaces and hop counts

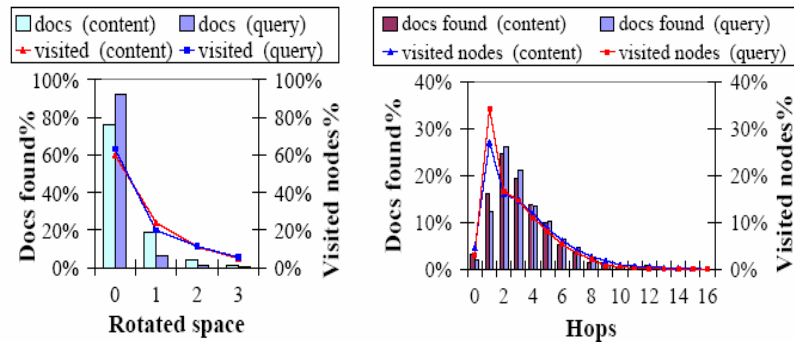


Fig. source [1]

## Conclusions

- pSearch
  - Distributed IR system using VSM and LSI
  - Overlay semantic network mapped on CAN
- Specialties
  - Rolling-indexes for dimensionality mismatch
  - Content-aware bootstrapping for load balancing
  - Context-directed search
- Benefit
  - Accuracy close to centralized LSI
  - Limited flooding
  - Limited bandwidth utilization

---

## References

- [1] C. Tang, Z. Xu, S. Dwarkadas – Peer-to-Peer Information Retrieval Using Self-Organizing Semantic Overlay Networks, Proceedings of SIGCOMM, 2003
- [2] C. Tang, Z. Xu, M. Mahalingam - PeerSearch: Efficient Information Retrieval in Peer-to-Peer Networks – HP technical report HPL-2002-198