Semantic Link Based Top-K Join Queries in P2P Networks

Jie Liu, Liang Feng, and Chao He
China Knowledge Grid Research Group, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing, 100080, China
{lj, feng_liang, hc}kg.ict.ac.cn

ABSTRACT
An important issue arising from Peer-to-Peer applications is how to accurately and efficiently retrieve a set of K best matching data objects from different sources while minimizing the number of objects that have to be accessed. This paper resolves this issue by organizing peers in a *Semantic Link Network Overlay*, where semantic links are established to denote the semantic relationship between peers' data schemas. A query request will be routed to appropriate peers according to the semantic link type and a lower bound of rank function. Optimization strategies are proposed to reduce the total amount of data transmitted.

Categories and Subject Descriptors
H.2.4 [**Database Management**]: Systems – *query processing*;
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *retrieval models*, *search process*, *selection process*.

General Terms: Algorithms, Design, Management.

Keywords: Join Query, Peer-to-Peer, Semantic Link, Top-K.

1. INTRODUCTION
An important issue arising from Peer-to-Peer (P2P) applications is how to efficiently process top-K join queries with the minimum transmission cost.

Example 1. Figure 1 is an example of top-K join queries, where data from ISI Web of Knowledge (ISI) and Journal Citation Reports (JCR) are located at different peers. The query to retrieve the top 100 papers most frequently cited and published at the highest impact journals can be represented as:

SELECT *Title*, *Author*, *Journal*, *Times Cited*, *Impact Factor*
FROM *ISI*, *JCR*
WHERE *ISI. Journal=JCR. Journal*
ORDER BY f (*TimesCited*, *ImpactFactor*)
STOP AFTER 100

The ORDER BY statement defines a monotonic increasing rank function, which takes attributes *TimesCited* and *ImpactFactor* as parameters to calculate the weighted rank value of paper citation and journal impact in the join results. The *STOP AFTER* clause sets the number of ordered results as 100.

To answer this query, a straightforward way is to join all the data from *ISI* with *JCR*, and then to select the top 100 papers with the maximum rank value. However, this approach suffers from a great consumption of bandwidth while generating the whole join results.

To reduce the number of data that must be actually involved in the join process, this paper proposes a semantic link network to answer top-K join queries in P2P networks.

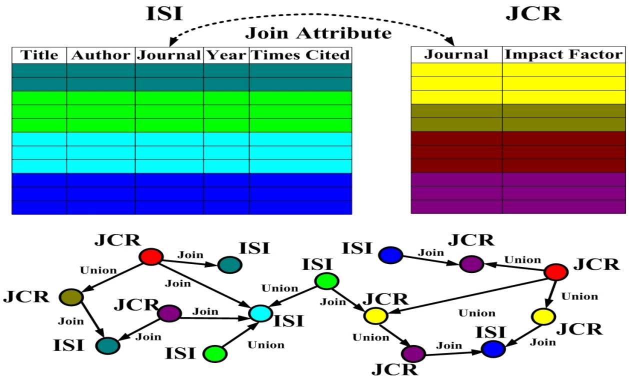

Figure 1. An example of top-K join queries in P2P networks.

2. RELATED WORK
A ranking algebra for ranking computation makes a first step towards a completely decentralized P2P search engine that offers meaningful and efficient rankings [1]. An algorithm for supporting ranked join queries in P2P networks introduced in [3] produces the top-K join results by multicasting the original query together with a lower bound of the rank value. However, the multicast approach in [3] forwards the query to all the neighbors of the current peer, which results in heavy traffic and poor scalability.

To efficiently support top-K join queries in P2P networks, this paper extends our previous work in [4] by establishing semantic links to denote the semantic relationship between peers' data schemas.

3. SEMANTIC LINK MODEL
Let *Schema* (P_i) and *Schema* (P_j) be the data schemas of peer P_i and peer P_j. A *Semantic Link* between P_i and P_j is a pointer with a type α directed from the predecessor P_i to its successor P_j, and α can be one of the followings:

(1) *Union Link*, P_i—*Union*→P_j, indicates *Schema* (P_i) =*Schema* (P_j), i.e., data in P_i and P_j satisfy a union condition in SQL.
(2) *Inclusion Link*, P_i—*Inclusion*→P_j, indicates *Schema* (P_i) $\supset$ *Schema* (P_j), i.e., the attribute set of P_j is included in P_i's attribute set.
(3) *Extension Link*, P_i—*Extension*→P_j, indicates *Schema* (P_i) $\subset$ *Schema* (P_j).
(4) *Overlap Link*, P_i—*Overlap*→P_j, indicates *Schema*(P_i) $\cap$ *Schema*(P_j)≠NULL and *Schema*(P_i) $\not\subset$ *Schema*(P_j) and *Schema*(P_j) $\not\subset$ *Schema*(P_i).

WWW 2006, May 23–26, 2006, Edinburgh, Scotland.
ACM 1-59593-323-9/06/0005.

(5) *Join Link*, P_i—*Join* (δ) $\rightarrow P_j$, indicates that *Schema* (P_i) and *Schema* (P_j) satisfy a join condition in SQL, and δ is the join selectivity factor, i.e., the ratio of the number of answers in P_i participating in the join operation to the total number of answers in the Cartesian product of P_i and P_j [2].

(6) *Empty Link*, P_i—$\varnothing\rightarrow P_j$, indicates that there is no semantic relationship between *Schema* (P_i) and *Schema* (P_j).

When a peer P_i joins a P2P network, it will randomly take a peer P_j as one of its neighbors and then send a SOAP message to get *Schema* (P_j). By checking the semantic relationship between *Schema* (P_i) and *Schema* (P_j), the semantic links between P_i and P_j can be established. Semantic links between P_i and other peers can be derived according to the heuristic reasoning rules [4].

4. GENERAL ARCHITECTURE

Let R and S be two relations scattered in P2P networks, a top-K join query can be denoted as Top-K ($R \bowtie_{r(a)\theta s(b)} S$), where $r(a)$, $s(b)$ are the join attributes satisfying a join condition $r(a)\theta\, s(b)$. Let $f(r(p), s(q)) \in [0, 1]$ be a predefined monotonic increasing rank function, where $r(p)$ and $s(q)$ are the rank attributes of R and S. The top-K join query returns the first K answers ordered by the rank value calculated by the rank function f.

The general architecture of semantic-link-based top-K join queries is shown in Figure 2. Upon receiving a Top-K ($R \bowtie_{r(a)\theta s(b)} S$) query, peer P_i performs the following steps to get the answers:

(1) *Query Processing* — P_i will first parse the query and get the parameters of the *Select-List*, the *join condition* $r(a)\ \theta\ s(b)$, and the number of ordered results K.

(2) *Set the Lower Bound T* — P_i will then get the K join results and set the lower bound T of the rank function f by sampling approach as proposed in [3].

(3) *Select Peers Storing Relation R* — P_i will then send a SOAP message *Top-K* (*Select-List*, $r(a)\ \theta\ s(b)$, $f(r(p), s(q))$, T, K) to its neighbors P_j (j=1…m). If $r(p) \in$ *Schema* (P_j), $r(a) \in$ *Schema* (P_j), and *Schema* (P_j) $\cap$ *Select-List*≠NULL, then P_j is selected as one of the join candidates. After that, P_j will sort on the rank attribute $r(p)$ and calculate the maximum $max(P_j.\ r(p))$. To prune irrelevant join data with a rank value below T, P_j then estimates the possible maximum rank value of the join results produced by P_j and its neighbor P_k by $f(max\ (P_j.\ r(p)), 1)$, where the second parameter in rank function f is set with the maximum value 1.

(4) *Select Peers Storing Relation S to be joined with R*—If $f(max\ (P_j.\ r(p)), 1)$ is greater than the lower bound T, P_j will then forward the query *Top-K* (*Select-List*, $r(a)\theta s(b)$, $f(max\ (P_j.\ r(p)), s(q))$, T, K) to its neighbors P_k (k=1…n) through *Join* semantic links. P_k will calculate the rank function by $f\ (max(P_j.\ r(p)), P_k.\ s(q))$ and return data satisfying $f(max\ (P_j.\ r(p)), P_k.\ s(q))$>T to P_j.

(5) *Local Top-K Ranking* — After receiving data from P_k, P_j will do join operation in its local repository, and sort on the rank function f to get the local top-K rank value.

(6) *Global Top-K Ranking* — After a predefined TTL (Time-to-Live), P_j (j=1…m) will send local top-K rank value to P_i, who will then sort on all the returned rank value and send the final lower bound T_{final} to P_j (j=1…m). Finally, join results in P_j (j=1…m) with a rank value greater than T_{final} will be sent to P_i to produce the global top-K answers.

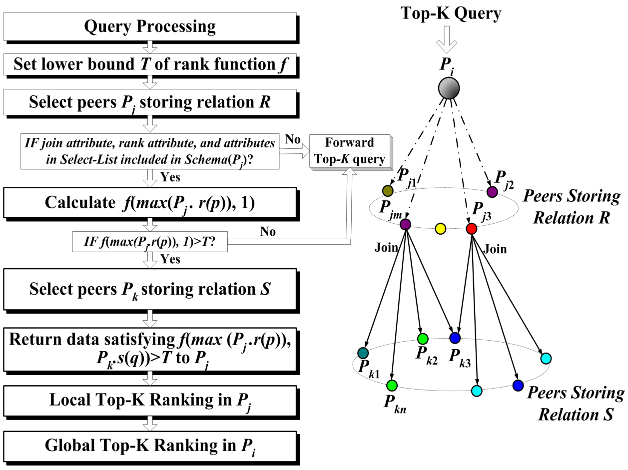

Figure 2. General architecture of semantic-link-based top-K join queries in P2P networks.

5. OPTIMIZATION STRATEGIES

As indicated in [2], the join selectivity factor is an important factor in the cost of performing join operation, and a higher selectivity factor requires a larger number of data comparisons. To reduce the total amount of transmitted data, optimization strategies are implemented in the semantic link network overlay. If the join selectivity factor is high, the nested-loop method is used to produce the join results; otherwise, the sort-merge method should be used. The neighbor with a higher join selectivity factor has the priority to be selected as a join candidate.

6. CONCLUSION

The contribution of this paper is the semantic-link-based infrastructure for efficient processing of top-K join queries. It can be used to construct the semantic overlay of the Knowledge Grid [5, 6] and support advanced applications.

7. ACKNOWLEDGEMENTS

The research work is supported by the National Science Foundation of China (No. 60503047) and the National Basic Research Program of China (No. 2003CB317000).

8. REFERENCES

[1] Aberer, K., and Wu, J. A Framework for Decentralized Ranking in Web Information Retrieval. *In Proceeding of APWeb* 2003, Xi'an, China, 2003.

[2] Mishra, P., and Eich, M. Join Processing in Relational Databases. *ACM Computing Surveys* 24(1) (1992) 63-113.

[3] Zhao, K. et al. Supporting Ranked Join in Peer-to-Peer Networks. *In Proceeding of DEXA Workshops* 2005, Copenhagen, Denmark, 2005.

[4] Zhuge, H. et al. Query Routing in a Peer-to-Peer Semantic Link Network. *Computational Intelligence* 21 (2) (2005) 197-216.

[5] Zhuge, H. The Knowledge Grid. *World Scientific Publishing Co.*, Singapore, 2004.

[6] Zhuge, H. A Knowledge Grid Model and Platform for Global Knowledge Sharing. *Expert Systems with Applications* 22 (4) (2002) 313-320.