ISSN:0975-9646

Inverse ObjectRank: Dynamic Authority Based Search in Databases

C. VIJAYA RAM
J.B.Institute of Engineering & Technology.

CH.Srinivasulu
Dept of IT
J.B.Institute of Engineering & Technology.

T.Jacob Sanjay Kumar
Dept of IT
J.B.Institute of Engineering & Technology

Abstract-The search algorithms to provide high quality, high recall search in databases, and the Web. Conceptually, these algorithms require a querytime PageRank-style iterative computation over the full graph. This computation is too expensive for large graphs, and not feasible at query time. Alternatively, building an index of precomputed results for some or all keywords involves very expensive preprocessing. We Introducing the quality of the results of ObjectRank dramatically changes according to various calibration parameters. One of the most interesting parameters is the specificity metric, for which the novel method of Inverse ObjectRank is employed, Ranking solely using ObjectRank. Objects with general context, like the *"Access Path Selection"* of ranked higher than more focused (specific) objects, like the *"Fundamental Techniques for Order Optimization"* . Intuitively, one might want to rank the *"Fundamental Techniques for Order Optimization"* higher because this paper is mostly cited by "sorting" , whereas the *"Access Path Selection"* only cited by "sorting". Inverse ObjectRank can achieve subsecond query execution time on the English Wikipedia data set, while producing high-quality search results that closely approximate the results of Inverse ObjectRank on the original graph. The Wikipedia link graph contains about 108 edges, which is at least two orders of magnitude larger than what prior state of the art dynamic authority-based search systems have been able to demonstrate. Our experimental evaluation investigates the trade-off between query execution time, quality of the results, and storage requirements of Inverse ObjectRank.

INTRODUCTION

The Inverse ObjectRank [1] is a system to perform authority-based keyword search on databases, inspired by PageRank [3]. PageRank is an excellent tool to rank the global importance of the pages of the Web. The PageRank as a tool to measure the global importance of the pages, independently of a keyword query. We appropriately extend and modify PageRank to perform keyword search on databases. The original variant of ObjectRank [1], the *"Access Path Selection in a Relational Database Management System"* paper would be ranked highest, because it is cited by four papers containing "sorting" (or "sort"). The *"Fundamental Techniques for Order Optimization"* paper would be ranked second, since it is cited by only three "sorting" Keys.

THE DATA MODEL

We view a database as a labeled graph, which is a model that captures both relational and XML databases, as well as the web. The *data graph* D(V,ED) is a labeled directed graph where every node v has a label _(v) and a set of keywords. For example, the node "SIGMOD" of Figure 2 has label "Conference" and the set of keywords {"SIGMOD"}. Each node represents an *object* of the database. The *authority transfer graph* G(V,E) represents the authority flows between the nodes of the data graph. Given a data graph D(V,ED), G(V,E) is created as follows. For every edge e = (u ! v) 2 ED we create (potentially) two edges ef = (u ! v) and eb = (v ! u). The edges ef and eb are annotated with *authority transfer rates* a(ef) and a(eb), which denote the maximum portion of authority that can flow between u and v. The authority transfer rates are assigned for every type of semantic connection by domain experts.

DATASET FOR DEMONSTRATION

Our demo system performs authority-based keyword search on bibliographic databases. It also provides calibration parameters such as the specificity metric and the quality metric. Users can specify various combinations of calibration values to control the behavior of the system. A user inputs (a) a keyword query, (b) a choice for combining semantics (AND or OR), (c) the importance of global quality of the results (i.e., Global ObjectRank), (d) the importance of containing the actual query keywords (translated to a damping factor value d), and (e) a specificity metric (i.e., Inverse ObjectRank). The output of the system is a ranked list of nodes of the database (to be more formal, of the authority transfer graph) according to the input parameters based on the ranking function in [4].

INVERSE OBJECTRANK

Conceptually, given a query keyword w, the ObjectRank value rw(v) of an object/node v of the data graph is computed as follows: Myriads of random surfers are initially found at the objects containing the keyword "sorting", which we call base set, and then they traverse the database graph. In particular, at any time step a random surfer is found at a node and either (i) makes a move to an adjacent node by traversing an edge, or (ii) moves back to a "sorting" node. Notice how ObjectRank produces keyword-specific rankings, in contrast to the global ranking of PageRank.

OBJECTRANK WITH PARAMETERS

SpecificityMetric - *Inverse ObjectRank* By analyzing the examples. we can observe how the specificity factor affects the top-10 paper list obtained by ObjectRank for the query "Concurrency Control". The difference in the two results is that for Result (a) no specificity metric was used, while for Result (b) we used Inverse ObjectRank. To measure the quality of these results we use the bibliography section of each chapter in a database texbook [5]. We compare the recall of the top 10 papers in Results (a) and (b) with respect to the set PCC of papers in the bibliography sections of the chapters on "Concurrency Control", which are viewed as the ground truth.

ANALYSIS

It measures to enable users to exploit the domain knowledge related to a given query, we integrate a domain ontology to the ObjectRank system. We first build the *ontology graph* GO(VO,EO), a labeled directed graph that captures a domain knowledge for terms. A *term* consists of one or more keywords and generally it represents a subject in a specific domain such as 'Concurrency Control' in database literature. We create a node v for every term identified. An edge e = (v ! u) is added if there is a semantic relationship between terms v and u. The edge is annotated with the type of the relationship and a weight w (0 < w _ 1) which denotes the strength of the relationship. So far, we only consider the relationship type 'is-a'. To provide the ontology graph of subjects in computer science area, we use a subset of the ACM Computing Classification System4. we compute related terms by running the ObjectRank algorithm on the ontology graph in the same way that we used the ObjectRank algorithm on the publications data graph to compute relevance values between a query and publications. Then, we calculate a new rank value of a publication p on a term t by combining the ObjectRank values of p on terms related to t. For example, when we run the ObjectRank algorithm on the ontology graph with *"Transaction Management"* node as a base set, terms such as *"Concurrency Control"* and *"Crash Recovery"* would get very high authority values. Using the new ranking function, which combines rank values of terms relevant to *"Transaction Management"*, publications relevant to *"Concurrency Control"* or *"Crash Recovery"* are favored even though their ObjectRank values on the given query are not high. In this way, the system can enhance search results automatically under the guidance of the ontology graph.

CONCLUSION

the Inverse ObjectRank system that performs authoritybased keyword search on bibliographic databases. We used Inverse ObjectRank as a keyword-specific specificity metric and other calibration parameters such as Global ObjectRank. Finally, we proposed a methodology that enables us to enhance the query results using an ontology graph.

REFERENCES

[1] S. Brin and L. Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *WWW Conference*, 1998.

[2] V. Hristidis, H. Hwang, and Y. Papakonstantinou. Authority-Based Keyword Search in Databases. *under preparation for journal submission*, 2006.

[3] R. Ramakrishnan and J. Gehrke. *Database Management Systems. Third Edition*. McGraw-Hill Book Co, 2003.

[4] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. In WWW, 1998.

[5] T. Condie, S. D. Kamvar, and H. Garcia-Molina. Adaptive peer-to-peer topologies. In P2P Computing, 2004.

[6] J. Davitz, J. Yu, S. Basu, D. Gutelius, and A. Harris. iLink: Search and Routing in Social Networks. In KDD, 2007.

[7] A. R. Dennis and S. T. Kinney. Testing media richness theory in the new media: The effects of cues, feedback, and task equivocality. Information Systems Research, 1998.

[8] J. Donath. Identity and deception in the virtual community. Communities in Cyberspace, 1998.

[9] B. M. Evans and E. H. Chi. Towards a Model of Understanding Social Search. In CSCW, 2008.

[10] D. Faye, G. Nachouki, and P. Valduriez. Semantic Query Routing in SenPeer, a P2P Data Management System. In NBiS, 2007.

[11] Z. Gyongyi, G. Koutrika, J. Pedersen, and H. Garcia-Molina. Questioning Yahoo! Answers. In WWW Workshop on Question Answering on the Web, 2008.

[12] T. Hofmann. Probabilistic latent semantic indexing. In SIGIR, 1999.