

BLINKS: Ranked Keyword Searches on Graphs

Hao He, Haixun Wang, Jun Yang, and Philip S. Yu

By Zhenning Shangguan

Mar-24-2009

Overview

- Problem and challenges
- Contributions
- Optimal graph search strategies
- Bi-level indexing and searching in BLINKS
- Evaluation

Problems & Challenges

- Ranked (top- k) keyword search over general node-labeled directed graphs
- Challenges:
 - Heuristic graph exploration strategies → poor performance for certain graphs
 - No full usage of indexing → slow search processing
 - Naive indexing mechanism → high storage requirement

Definitions

- Query Answers

DEFINITION 1. Given a query $q = (w_1, \dots, w_m)$ and a directed graph G , an answer to q is a pair $\langle r, (n_1, \dots, n_m) \rangle$, where r and n_i 's are nodes (not necessarily distinct) in G satisfying the following properties:

(Coverage) For every i , node n_i contains keyword w_i .

(Connectivity) For every i , there exists a directed path in G from r to n_i .

- Top- k Query

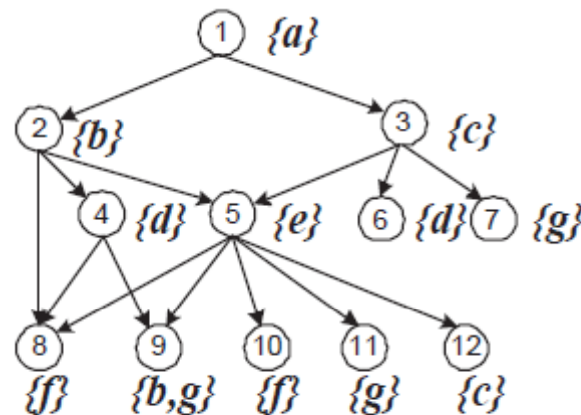
DEFINITION 2. Given a query and a scoring function S , the (best) score of a node r is the maximum $S(T)$ over all answers T rooted at r (or 0 if there are no such answers). An answer rooted at r with the best score is called a best answer rooted at r . A top- k query returns the k nodes in the graph with the highest best scores, and, for each node returned, the best score and a best answer rooted at the node.

Definitions

- Scoring Function

$$\underline{S}(T) = f(\bar{S}_r(r) + \sum_{i=1}^m \bar{S}_n(n_i, w_i) + \sum_{i=1}^m \bar{S}_p(r, n_i))$$

- Example



(A) The graph G

$q = (c, d)$

(B) A query q

$T_1 = \langle 3, (3, 6) \rangle$
 $T_2 = \langle 2, (12, 4) \rangle$

(C) Answer trees

Contributions

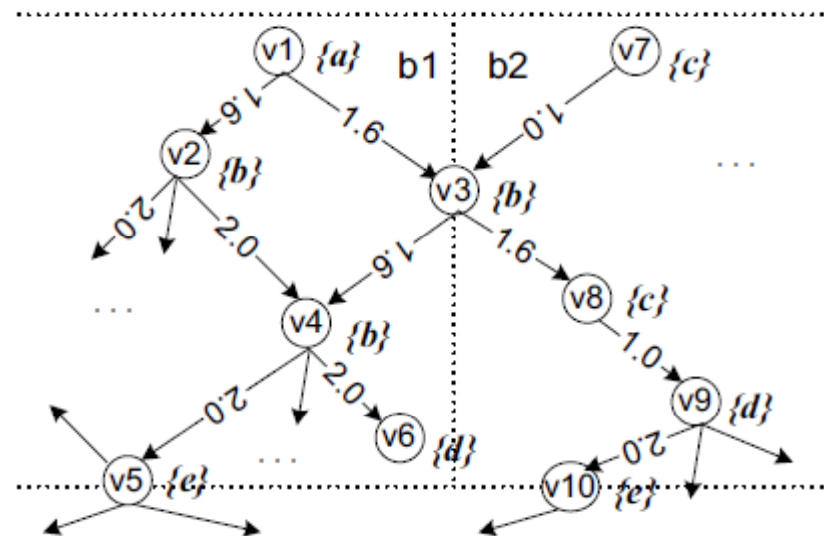
- Optimal search strategy
 - Equi-distance expansion within each cluster
 - Cost-balanced expansion across clusters
- Combining indexing with search
 - Reduces running time of backward search
 - Supports forward search – search becomes bidirectional
- Graph partitioning-based bi-level indexing
 - Block level
 - Intra block level
- Performance
 - Orders-of-magnitude improvement

Optimal Graph Search Strategies

- Equi-distance expansion within each cluster
 - The node u to visit for cluster E_i (by following edge $\langle u, v \rangle$ for some v in E_i) is the node with the shortest distance (among all nodes not in E_i) to the cluster origin O_i .
- Cost-balanced expansion across clusters
 - The cluster E_i to expand next is the one with the smallest cardinality

Bi-level Indexing and Searching

- Portal nodes
 - In-portal nodes: have at least one incoming edge from another block and at least one outgoing edge in this block
 - Out-portal nodes: have at least one outgoing edge to another block and at least one incoming edge from this block



Intra-block Index

- Intra-block keyword-node lists
 - List of nodes that can reach keyword w
- Intra-block node-keyword map
 - Hashmap storing the shortest distance from node u to keyword w
- Intra-block portal-node lists
 - List of nodes that can reach out-portal p
- Intra-block node-portal distance map
 - Hashmap storing the shortest distance from node u to the closest out-portal

Block Index

- Keyword-block lists
 - List of blocks containing keyword w
- Portal-block lists
 - Lists of blocks containing node p as an out-portal

Searching with Bi-level Index

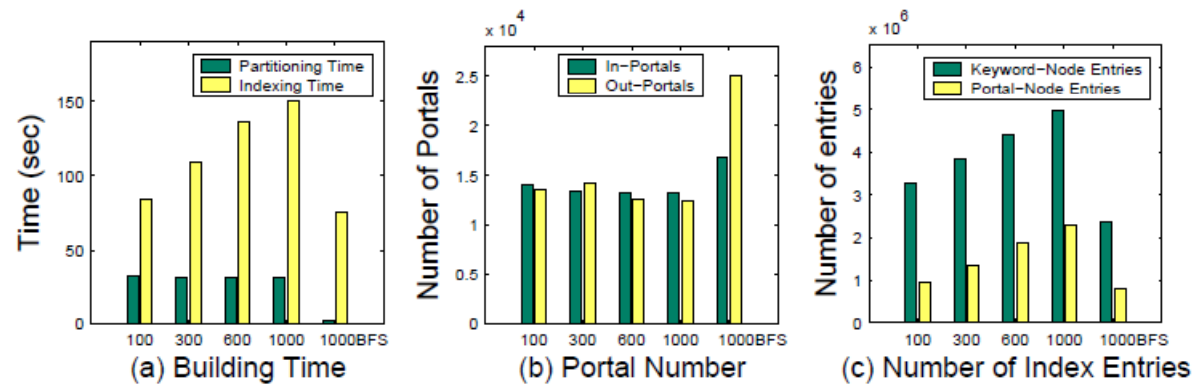
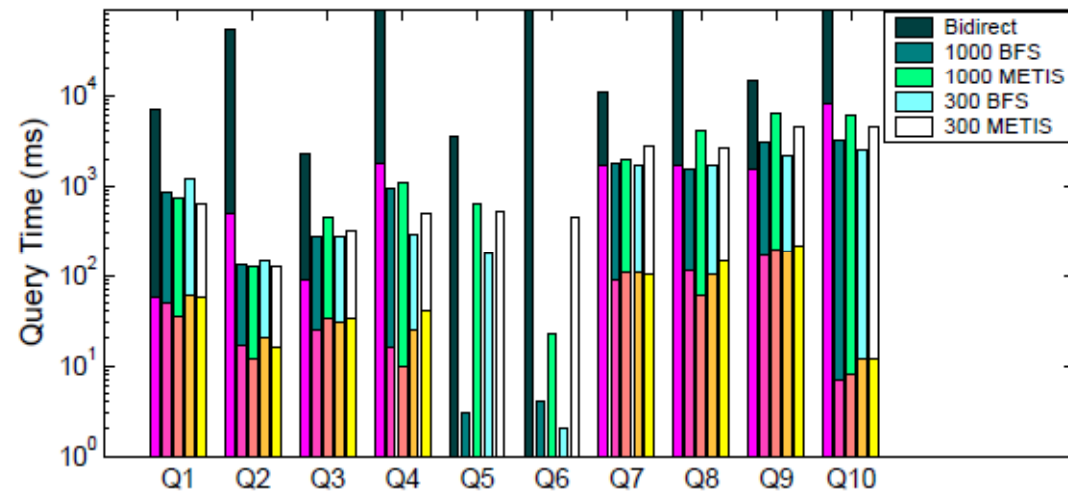
- Backward expansion
 - Keyword-block list → intra-block keyword-nodes lists
→ portal-block lists → portal-node list
- Forward expansion
 - Intra-block node-portal distance map + intra-block node-keyword map
- Pruning
 - Lower bound from search: intra-block keyword-nodes lists
 - Lower bound from index: Intra-block node-keyword map + Intra-block node-portal distance map

Graph Partitioning

- General guidelines
 - Keep the total number of portals low
 - Keep blocks balanced in size
- BFS-based partitioning
 - start from an unassigned node and perform BFS
 - add to this block any nodes that we visit but have not been previously assigned to any block, until the given block size is reached
- METIS-based partitioning
 - Use the METIS algorithm to avoid the poor starting nodes of BFS-based partitioning

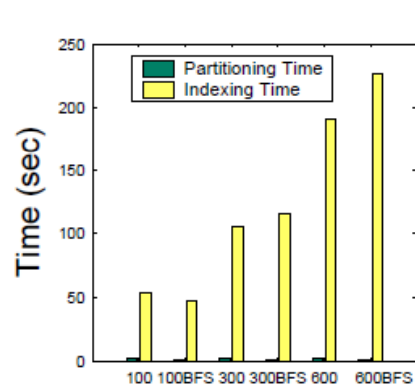
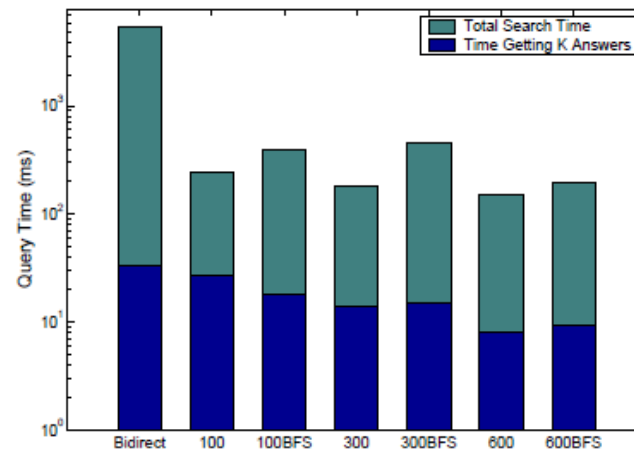
Evaluation

- DBLP Dataset

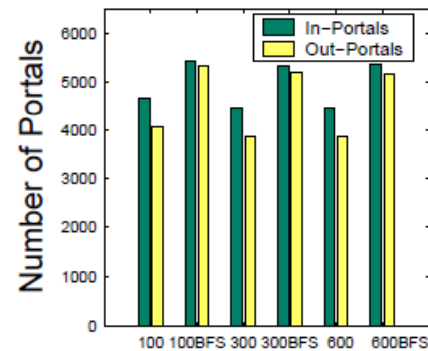


Evaluation

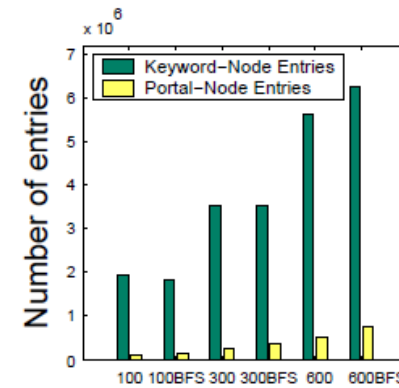
- IMDB Dataset



(a) Building Time



(b) Portal Number



(c) Number of Index Entries

Review

- Relevance: 6 (for SW), 8 (for general AI)
- Significance: 8
- Technical Soundness: 8
- Novelty: 8
- Quality of Evaluation: 7
- Clarity: 9
- Overall: 8
- Review Confidence: 5

Relevance – 7 (avg)

- Focuses on ranked keyword search on general (node-labeled directed) graphs
 - Important topic in DB and Searching
 - Indexing and searching mechanisms might be applied to the searching of semantic web data (RDF & OWL graph model can be considered as a node-labeled and edge-labeled graph)

Significance - 8

- Order-of-magnitude improvement on search performance
- Reduced space requirement
- Useful on large graphs, and thus the huge amount of open linked raw data on the (Semantic) Web

Technical Soundness - 8

- Solutions presented are technically sound
- Theorems and lemmas given as conclusions, while detailed and complete proof can be found in the accompanied technical report
- Space complexity of the constructed indices are formally always analyzed
- Time complexity of the bi-level index based algorithm is not explicitly given, backed up only by evaluations

Novelty - 8

- Major contributions are the bi-level indexing scheme and corresponding search algorithm
- State-of-the-art work in exploiting indices extensively to accelerate keyword searches on general graphs

Quality of Evaluation - 7

- Two specialized datasets (DBLP & IMDB) might not be representative enough, need graphs in other fields (if any)
- Characteristics of the synthesized queries (Q_1 – Q_{10}) are not so clear
- Orders-of-magnitude performance improvements may be debatable with so few query samples tested

Clarity - 9

- Very neat and understandable paper
- Due to space constraints, some technical proofs and formal analysis are omitted on purpose, which can be found in the TR
- Use single-level indexing as an basis
- Occasional obvious typos and minor errors
 - E.g., intra-block portal-node list in Fig. 6 is missing one list element

Overall – 8

- Qualitatively, this is a technically novel and strong research paper that solves a common problem in the searching field. With extensions, it may be applicable in various fields
- Quantitatively,

$$\frac{\sum ratings}{\# criteria} \approx 8$$

Confidence - 5

- No expertise in ranked keyword search field
- Lack of knowledge about the state-of-the-art work in this area
- Due to time constraints, only able to follow some of the formal proof and analysis presented in the TR

Q & A

- Thanks!