Landmark indexing for scalable evaluation of label-constrained reachability queries

Lucien Valstar, George Fletcher, and Yuichi Yoshida

Dutch Belgian Database Day 2016
Mons, Belgium
October 28, 2016
Introduction & problem statement
Introduction

- Web and many other contemporary applications are generating huge amounts of graph data. Many of these are edge-labelled.
- Examples:
  - RDF, semantic web
  - knowledge graphs
  - social networks,
  - road networks
  - biological networks
Example: social network

- LCR-query: can $v_1$ reach $v_3$ using only edges of the label \{friendOf\}?
  - No, hence query $(v_1,v_3,\{friendOf\})$ is false.

- Can $v_1$ reach $v_3$ using only edges of the labels \{friendOf, likes\}?
  - Yes, hence the query $(v_1,v_3,\{friendOf, likes\})$ is true.

Figure 1: An example of a directed graph with $|V| = 5$ vertices, $|E| = 7$ edges, and edge labels $\mathcal{L} = \{\text{likes, follows, friendOf}\}$. 
Solutions
Breadth-first search

- Given a query \((v,w,L)\) we wish to find out whether the query is a true- or a false-query.
- BFS explores the graph looking for \(w\) using only edges with a label \(l \in L\).
- It has the ‘maximum’ query answering time, but the ‘minimum’ index construction time and index size.
Landmarked-index (LI): our basic idea

- Building a full index, i.e. for all vertices, takes too much time and memory, but can answer all queries immediately.
- Hence we build an index for a subset of the vertices \( k \leq n \) (called landmarks) of vertices: \( v_1, \ldots, v_k \), where \( n \) is the number of nodes.
- Build an index for each \( v_1, \ldots, v_k \).
- Use BFS as baseline and use \( v_1, \ldots, v_k \) to speed up the query answering.
Landmarked-index (LI+): extensions

For large graphs we get that the ratio $k/n$ gets lower. Because we use BFS as a baseline, we may experience two issues.

1) Reaching the landmarks may take a long time, hence we store some (say $b$) label sets connecting non-landmarks with landmarks.

2) False queries are still slow with LI-approach. For each landmark $v$ and a label set $L^*$, we store a subset of the vertices $V^* \subseteq V$ s.t. for all $v^*$ in $V^*$ we have that $(v,v^*,L^*)$ is a true-query. This is used for pruning.
Experimental results
# A few real datasets

| Dataset              | $|V|$  | $|E|$  | $|L|$ | $k$  | $b$  |
|----------------------|------|-------|------|------|------|
| soc-sign-epinions    | 131k | 840k  | 8    | 1318 | 15   |
| webGoogle            | 875k | 5.1M  | 8    | 1751 | 15   |
| zhishihudong         | 2.4M | 18.8M | 8    | 4905 | 15   |
| wikiLinks (fr)       | 3M   | 102.3M| 8    | 1738 | 20   |

- Used server with 258GB of memory and a 32-core 2.9Ghz processor
- Set a 6-hour time limit and a 128GB memory limit
- Method under study: LI+
- Single-threaded
- 3,000 true-queries
- 3,000 false-queries
Results on these graphs
- Index size (MB) and construction time
- Speed-up over BFS

| Dataset          | IS (MB) | IT (s) | True, $|L|/4$ | False, $|L|/4$ | True, $|L|-2$ | False, $|L|-2$ |
|------------------|---------|--------|----------|---------------|--------------|--------------|
| soc-sign-opinions| 1,159   | 114    | 1,733    | 1,894         | 4,213        | 2,958        |
| webGoogle        | 27,117  | 4,691  | 4,181    | 5,908         | 4,385        | 20           |
| zhishihudong     | 16,199  | 6,419  | 803      | 911           | 954          | 20           |
| wikiLinks        | 98,125  | 24,873 | 10,200   | 9321          | 13,082       | 8036         |
Additional results

- Similar results have been obtained on 23 real datasets
- And on dozens of synthetic datasets where we varied:
  - graph size (5k up until 3.125M vertices)
  - label set distribution (exponential, normal, uniform)
  - label set size (from 8 to 16)
  - growth model (Erdos-Renyi, Preferential Attachment)
- Other query related types (e.g. distance queries) were studied
Conclusion
Conclusion

- Landmarked-Index is scalable w.r.t. the graph size.
- Landmarked-Index leads to multiple orders of magnitude speed-ups, although there is some asymmetry still between true- and false-queries.
- Future work:
  - Landmarked-Index could be a groundwork for other types of queries (distance queries, finding a witness, defining a budget per label, RPQ).
  - Maintainability of the index.
Questions?
**Related work**

- Zou et al. “Efficient processing of label-constraint reachability queries in large graphs.” is about LCR.
- Bonchi et al. “Distance oracles in edge-labeled graphs.” is about LCR+distance.

- For more on the LI-algorithm: https://www.youtube.com/watch?v=QKLtpoLdXfk