Evaluating Reachability Queries over Large Social Graphs

Imen BEN DHIA

Advisors:
Talel ABEDESSALEM
Mauro SOZIO
Outline

- Introduction to Reachability and Applications
- Existing Approaches
- Evaluating Access Control Reachability Queries
  - Reachability backbone discovery
  - 2-hop index construction
  - Answering queries
- Ongoing Work
Outline

- Introduction to Reachability and Applications
- Existing Approaches
- Evaluating Access Control Reachability Queries
  - Reachability backbone discovery
  - 2-hop index construction
  - Answering queries
- Ongoing Work
Introduction to reachability

Use cases:

Privacy policies evaluation ⇔ Constrained reachability queries evaluation.

- 2 to 3 different labels
- Distance (up to 4) according to real world scenarios
Applications

- Social networks
- Bioinformatics
Constrained Reachability Problem

- **The problem**: Given two vertices \( u \) and \( v \) in a directed graph \( G \), is \( v \) reachable from \( u \) via a given path?
- A **path** is a sequence of constraints on **label order** and **distance**.

?Query(1, a\a\b, 11)
Yes

?Query(3, a\a\b, 9)
No
Outline

- Introduction to Reachability and Applications

- Existing Approaches

- Evaluating Access Control Reachability Queries
  - Reachability backbone discovery
  - 2-hop index construction
  - Answering queries

- Ongoing Work
Naïve Solutions

- Pre-compute and store the transitive closure (all paths between all possible pairs of nodes)
  - Then, answer any query in constant time: $O(1)$
  - What are Space requirements for an n-node graph? $O(n^2)$

Online Search (BFS/DFS)

- Answer query Single Source Shortest Path Algorithm
- Minimal additional space required: $O(n+m)$
- What is the time complexity to answer query? $O(n+m)$
**Goal:** Finding a compromise between time and space consumption to answer reachability queries.

**Find a compact representation for the transitive closure:**
- whose size is comparable to the data size
- that supports connection tests (almost) as fast as the naïve transitive closure lookup
- that can be built efficiently for large datasets
Related Work

Two main categories of approaches:

- **Using spanning structures (chains and trees)**
  - Path-tree (Jin et al. ’08)
  - Label-constraint reachability queries (Jin et al. ‘10)

- **Using 2-hop strategy**
  - 2-hop labeling (Cohen et al. ‘02)
  - Fast graph pattern matching (Wang et al.‘08)
Shortcomings

- Not distance-aware.
- Constraints on label order are not respected.
- Constraints on node properties are not considered.
- Reach a bottleneck when graphs are large
Introduction to Reachability and Applications

Existing Approaches

Evaluating Access Control Reachability Queries
  • Reachability backbone discovery
  • 2-hop index construction
  • Answering queries

Ongoing Work
Evaluating Access Control Reachability Queries consists in three main steps:

1. Reachability backbone discovery
2. Two-hop index construction
3. Reachability query evaluation over reachability backbone
Introduction to Reachability and Applications

Existing Approaches

Evaluating Access Control Reachability Queries
  • Reachability backbone discovery
  • 2-hop index construction
  • Answering queries

Ongoing Work
Remark:
- Multi-graph (with multiple labels) => a set of single labeled graphs.

Determining a subset of nodes that cover two-hop paths.
- Shortest two-hop paths sampling.
- Determining degree threshold.
Introduction to Reachability and Applications

From Access Control to Reachability

Existing Approaches

Evaluating Access Control Reachability Queries
  - Reachability backbone discovery
  - 2-hop index construction
  - Answering queries

Ongoing Work
Main Idea: 2-Hop Cover & 2-Hop Labeling

- 2-Hop cover is a set of hops \((u, v)\) so that every connected pair is covered by 2 hops.

- For each node \(x\), we maintain two sets of labelings (which are simply lists of nodes): \(L_{in}(x)\) and \(L_{out}(x)\).

- \(u\) can reach \(v\) \(\iff\) \(L_{out}(u) \cap L_{in}(v) \neq \emptyset\)

(Cohen et al., SODA 2002)
2-hop Covers

- **Goal:**
  - Find a cover which minimizes the number of centers $w_i$

- **Problem is NP-hard**
  - => Approximation is required

- **Two main ingredients of the 2-hop cover algorithm:**
  - Set cover algorithm.
  - Densest subgraph algorithm.
Introduction to Reachability and Applications

Existing Approaches

Evaluating Access Control Reachability Queries
  - Reachability backbone discovery
  - 2-hop index construction
  - Answering queries

Ongoing Work
Answering queries

- Reachability computation via reachability backbone
  - Performing two local BFS searches for accessing reachability backbone
  - Reachability join test
Outline

- Introduction to Reachability and Applications
- Existing Approaches
- Evaluating Access Control Reachability Queries
  - Reachability backbone discovery
  - 2-hop index construction
  - Answering queries
- Ongoing Work
Algorithm implementation optimization

Using MapReduce:
- For set cover problem
- To compute densest bipartite graph
Thanks For Your Attention!