Advanced Data Management

Medha Atre

Office: KD-219
atrrem@cse.iitk.ac.in

Oct 6, 2016
Recap

- General regular path query problem (first type mentioned before) is NP-hard [Mendelzon, Wood 1995].
- Some polynomial time algorithms suggested for a restricted set of regular expressions.
- Even polynomial time algorithms for large graphs are expensive as often their complexity is of the order of $n^x$, where $n$ is the total number of nodes in the graph and $x \geq 2$.
- Early solutions consisted of creating regular expressions representing all the paths between every pair nodes in the graph $L(R_{xy})$ [Tarjan 1981].
- Considering entire graph $G$ as an NDFA with $x$ as the start state and $y$ as the final state, and create an intersection graph with the NDFA $M$ of $L(R)$ [Mendelzon, Wood 1995].
Recap

- Creating **equivalence classes of paths** in the entire graph [Abiteboul, Vianu 1997]
- Creating equivalence classes of **nodes** based on their incoming paths – *B-bisimilarity* – 1-index. Similar create 2-index, and from them create *T-index* (template index) [Milo, Suciu 1999].
- XML (XPath) solution space:
  1. **P-indexes**: Path indexes, A(k), D(k), M(k), M*(k), APEX, Bitmapped Path Index (BPI).
  2. **D-indexes**: Node index for determining ancestor-descendant relationship, with method similar to interval labeling.
  3. **T-index**: Used mainly for *twig* queries on XML.
Challenges

- Not nice structure like trees (may have cycles).
- Large sizes and hence possible exponential paths (impossible to index).
- Edge labels as an additional dimension.
- Even with the restricted set of regular language which may have polynomial time solutions, problem remains computationally challenging due to the sheer size of the graphs, e.g., several million nodes.
- Restricted set of regular language is included as a part of the SPARQL 1.1 standard.
Label Constrained Reachability

- Problem definition: Given a graph $G$ with edge label set $S$, a pair of nodes $(x, y)$, and a subset of edge labels $Q \subseteq S$, does there exist a path from $x$ to $y$ such that the path label set $L(p)$ is a subset of $Q$, $L(p) \subseteq Q$.

- $L(p)$ is the set of all the unique edge labels that appear on a given path.

- This is a restricted regular path query problem where the regular language is $R := R^+ | t$, $t := Q$, i.e., $t$ is a terminal that can take any value from the set $Q$. 
Label Constrained Reachability

- Trivial (expensive) solution: For any pair of nodes \((x, y)\), maintain all the unique sets of path-labels for all the paths between them.

- Instead maintain a set \(S_{min}\) of *minimal sufficient* path labels between \((x, y)\), such that:

\[
S_{min} = \{ L(p) | L(p) \in S_0 \land \forall L(p') \in S_0, s.t., L(p') \subset L(p) \}
\]

- Computing \(S_{min}\) requires a modified single source shortest path kind algorithm (e.g., Floyd-Warshall) \(O(|V|^2(|\Sigma|/2))\), \(\Sigma\) is the total number of edge labels.
Label Constrained Reachability

- Hence we go for an alternate solution:
  - Sampling subset of vertices repeatedly.
  - Compute single source generalized transitive closure $M(u, v)$ of minimal path labels just for those vertices, where $u$ is a sampled vertex.
  - Use the above to determine approximate edge weight and error bound (based on Hoeffding-Bernstein bound) for all the edges in the graph.
  - From these two values compute two maximal spanning trees for given $G$.
  - Sample vertices repeatedly (with replacement) to get alternate spanning trees, and stop once condition in the Hoeffding-Bernstein-Tree algorithm is achieved.

- Total computational complexity
  $$O(n|V||E|(|\Sigma|/2) + n/n_0(|E| + |V|\log|V|)).$$
Let us consider key points.

On a spanning tree ( maximal or not ), authors define $P_n$ as the set of paths where both starting and ending edges are not in the spanning tree.

For $P_n$ minimal path labels $NT(u, v)$.

With $L(P_T(u, v))$ as the set of path labels for a spanning tree path between $(u, v)$, we have:

$$M'(u, v) = \{ L(P_T(u, u')) \odot NT(u', v') \odot \{ L(P_T(v', v)) \} | u' \in succ(u), v' \in pred(v) \}$$

Using the above formula, and approximate maximal spanning tree along with the reachability index created on the spanning tree, answer the reachability queries.
Path Pattern Queries

- Regular language considered:

\[ F ::= c | c^{\leq k} | c^+ | FF \]

- 3-D reachability index, where the third dimension is the edge-labels (colors as the authors say), which notes the length of the \textit{shortest} path between the given nodes with just that given edge-label (color).

- Queries evaluated using join-based algorithm, by breaking down the given regular expression into multiple components.

- Authors also discuss regular language containment and equivalence to reduce a given expression to its minimal form in order to achieve better query evaluation, by avoiding unnecessary computations.
Other topics in regular path queries

- Optimizing regular path queries using graph schemas [Fernandez, Suciu 1998].
- Algebraic rewriting of the regular path queries for optimization [Grahne, Thomo 2003].
- Answering regular path queries using views [Calvanes et al 2000].
Next Class

We will review methods of doing “keyword searches” on graph data.

Have a happy mid semester recess and do not forget Assignment-3! :-)}