More Efficient Indexes for Subgraph Isomorphism Search

Jan Hidders, Jonathan Riggio

WISE Lab, Vrije Universiteit Brussel, Belgium

Nowadays in the database community, the limitations of the Relational Model such as, its limited ability to explicitly capture requirement semantics have led to the emergence of new database technologies. Among them, *Graph Databases* are appealing because, in real world applications, the extraction of relevant information often relies on processing graph structures (e.g. social media networks, logistic and routing...). Graph databases are therefore a more intuitive way of representing graph-like data and queries. The need for information systems that are able to efficiently retrieve the data and express various queries has become even more crucial with the rise of Big Data.

One of the main graph database problems is the resolution of graph queries[5], i.e., identifying certain graphs in a large collection of graphs, which is formalized as follows. Considering a graph database D of n graphs, $D = \{g_1, \ldots, g_n\}$ and a query graph q, resolving a graph query is equivalent to retrieving all the graphs $g_i \in D$ where the query graph q occurs as a subgraph, i.e., which contain subgraphs that are isomorphic to the query graph.

Processing *Graph Queries* efficiently is a critical problem in many graph-related applications. However, the resolution of graph queries is directly related to the *Subgraph Isomorphism Problem* [4] which belongs to the class of NP-complete problems. The consequence is that for this problem, a sequential scan on the graph database that checks iteratively whether q is a subgraph of g_i is inefficient. Hence the need to build graph indexes able to support graph queries processing.

Our main goal is to improve graph queries processing by taking advantage of the fact that the subgraph isomorphism problem can be solved in linear time for planar graphs[3] and that some properties over graphs can be decided in linear time for graphs with bounded treewidth given a tree-decomposition[1, 2]. Therefore we focus on the following research questions:

- Given a graph database, can we reduce the average query response time by restricting the index to certain structures such as planar graphs or bounded treewidth graphs for which we know the subgraph isomorphism problem to be tractable?
- What is the cost of building and maintaining such an index? If it is too expensive, is there any heuristic that would return results close to the optimum?

References

[1] S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12(2):308–340, 1991.

[2] B. Courcelle and M. Mosbah. Monadic second-order evaluations on tree-decomposable graphs. *Theoretical Computer Science*, 109(1):49–82, 1993.

[3] D. Eppstein. Subgraph isomorphism in planar graphs and related problems. In *SODA*, volume 95, pages 632–640, 1995.

[4] C. Nabti and H. Seba. Subgraph isomorphism search in massive graph databases. In *The International Conference on Internet of Things and Big Data–IoTBD 2016*, 2016.

[5] X. Yan, P. S. Yu, and J. Han. Graph indexing: a frequent structure-based approach. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 335–346. ACM, 2004.