Aggregating over subgraph patterns in powerlaw graphs

Jan Ramon and Joachim Niehren

1 Context

Recently, many databases are network-structured, e.g., in social networks, economic networks, traffic networks, citation networks or biological networks. A key task in data mining and machine learning, aiming to extract interesting patterns from such databases, is to compute statistics (or aggregates) over the database. It is well-known that real-world networks satisfy statistical regularities, one example is that the degree distribution of such graphs often satisfies a power law. An interesting question is how this statistical structure of the database can influence our choice of aggregation algorithms and the performance of these algorithms.

2 Problem

In database theory, often the worst-case computational complexity of querying and aggregation algorithms is well studied. However, when the database has a "random" structure (where local contexts follow some particular probability distribution), it makes sense to study the average case behavior, which may be very different from the worst case. We are interested in aggregation in the context of machine learning, an operation studied less thoroughly than the classical decision problems. In a first step, this project will focus on the specific class of powerlaw graphs.

3 Activities in the project

The topic subscribes is a cooperation project between the teams Links and Magnet at INRIA and the Cristal lab in Lille. The topic also subscribes to the ANR project AGGREG on aggregate queries. Our intension is to continue this proejct to a master 2 internship in order to prepare a PhD project. This PhD project could then funding partially by AGGREG.

For a start up, the student will (1) perform a literature study, starting from a number of provided pointers and ideas and (2) apply these ideas to the specific problem under consideration, where possible coming up with new approaches.

Important subgoals are (i) a refined analysis of a Yanakakis style aggregation algorithm and (ii) improvements to the naive algorithm as a function of the network parameters.