Dissociation-based Optimization in Probabilistic Databases

Maarten Van den Heuvel[1], Floris Geerts[1], Martin Theobald[2]
[1]Universiteit Antwerpen, Belgium
[2]Ulm University, Germany

Probabilistic inference remains a key challenge both for query processing and for query optimization in the context of probabilistic databases. Unfortunately, exact inference is known to be #P-hard in the size of the database. Numerous approaches have been suggested to tackle this combinatorial complexity, among them algorithms for top-k query evaluation and for approximate inference. In our work, we combine ideas from these settings to further accelerate inference under a tuple-independent probabilistic database model.

Deterministic top-k ranking algorithms were originally designed to retrieve the k highest-ranking database objects based on an incomplete, incremental scan of the data and by computing upper and lower bounds on the objects' scores according to some ranking function. Some approaches have been suggested in a probabilistic setting as well by computing upper and lower bounds on the marginal probability of a query answer instead of using a simple ranking function.

Most state-of-the-art inference algorithms, on the other hand, are based on sampling and can take a long time to converge. A relatively new approach uses dissociation of queries as a means to provide safe approximate query plans for unsafe queries. Safe plans can do probabilistic inference in polynomial time, but only for a restricted class of queries. Dissociating a query provides a plan which gives upper and lower bounds on the probabilities of the original query's answers, while running in polynomial time. A problem with this approach, however, is that finding the dissociation that gives the best bounds requires the actual execution of a large number of the possible plans, where the amount of possible dissociations is exponential in the size of the query.

We introduce an approach based on the principle of dissociation combined with the use of metadata in the form of histograms. We will use the metadata we have on our database both to choose a good dissociated query plan that gives tight bounds to the original probabilities and to approximate the dissociation bounds themselves by only scanning the data partially in a top-k fashion and pruning candidate answers where possible. The bounds are approximated by estimating the amount of tuples that can be used in the lineage of an answer from the histograms. How good a query plan (i.e. how tight the bounds are) is also dependent on the amount of times a tuple gets reused in the lineage of an answer, which we can again estimate from the metadata. This way, we can pick a good query plan very efficiently even for large complicated queries, and quickly return the top-k answers for large databases based on lower and upper bounds.