Title: Incremental and parallel algorithms for dense subgraph detection

Abstract:

The task of maintaining densely connected subgraphs from a continuously evolving graph is important because it solves many practical problems that require constant monitoring over the continuous stream of linked data often represented as a graph. For example, continuous maintenance of a certain group of closely connected nodes can reveal unusual activity over the transaction network, identification, and evolution of active groups in social network etc. On the other hand, mining these structures from graph data is often expensive because of the complexity of the computation and the volume of the structures (the number of densely connected structures can be of exponential order on the number of vertices in the graph). One way to deal with the expensive computations is to consider parallel computation.

We advance the state of the art by developing provably efficient algorithms for mining maximal cliques and maximal bicliques.

First we consider the design of efficient algorithms for the maintenance of maximal cliques and maximal bicliques in an evolving network. We observe that it is important to locate the region of the graph in the event of the update so that we can maintain the structures by computing the changes exactly where it is located. Following this observation, we design efficient techniques that find appropriate subgraphs for identifying the changes in the structures. We prove that our algorithms can maintain dense structures efficiently. More specifically, we show that our algorithms can quickly compute the changes when it is small irrespective of the size of the graph. We empirically evaluate our algorithms and show that our algorithms significantly outperform the state of the art algorithms.

Next, we consider parallel computation for efficient utilization of the multiple cores in a multi-core computing system so that the expensive mining tasks can be eased off and we can achieve better speedup than their efficient sequential counterparts. We design shared memory parallel algorithms for the mining of maximal cliques and we prove the efficiency of the parallel algorithms through showing that the total work performed by the parallel algorithm is equivalent to the time complexity of the best sequential algorithm for doing the same task. Our experimental study shows that we achieve good speedup over the prior state of the art parallel algorithms and significant speedup over the state of the art sequential algorithms. We also show that our parallel algorithms scale almost linearly with the increase in the processor cores.