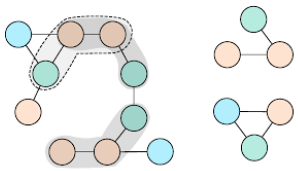
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Prof. R. Wattenhofer

Topics in Graph Pattern Mining



Frequent substructure pattern mining (hereafter "pattern mining") is now an established data mining problem with many scientific and commercial applications. As a general data structure, labeled graph can be used to model much complicated substructure patterns among data.

Given a graph dataset, $\mathcal{D} = \{G_0, G_1, \ldots, G_n\}$, $suppart(H)$ denotes the number of graphs (in $\mathcal{D}$) in which H is a subgraph. The problem of frequent subgraph mining is to find any subgraph H s.t. $support(H) \geq minSup$ (a minimum support threshold). To reduce the complexity of the problem (meanwhile considering the connectivity property of hidden structures in most situations), only frequent connected subgraphs are often studied.

The problems of this field – ever more relevant in the age of easy connectivity and social network presence – are easy to understand and far from hard to find. Here are but a few of those that we are interested in:

1. **Pattern Mining in Code.** Given the parse trees of code from GitHub, can you find patterns in people's code that convey useful information about programmers' practices?

2. **Contrastive Lax Pattern Discovery.** What if we do not insist that our pattern is always matched in full? What if we allow some nodes or links to be missing? Can we still properly cluster these "similar instances" of some more abstract pattern?

3. **Pattern Mining Through Path Stars.** What if we forgot about the fact that we are dealing with graphs? Can some of the previous research on pattern mining in strings be reused for our graph-y purposes?

4. **Neighbourhood Graph Similarity Measures.** We have previously developed a graph similarity measure for coloured graphs that takes into account how similar the colours of neighbours of individual nodes are. We feel that our first, naive approach leaves a lot to be desired.

Requirements. Familiarity with graphs and networks, and solid coding skills will be of great help. *Bachelor's* students will work on a project of a limited scope, mostly on the applied side. *Master's* students will work on more comprehensive project on the interface of theory and practice.

Interested? Please contact us for more details!

Contact

- Peter Belcak: belcak@ethz.ch, ETZ G63