Efficient processing of graph similarity search

Ryan Choi · Chin-Wan Chung

Abstract A graph similarity search is to find a set of graphs from a graph database that are similar to a given query graph. Existing works solve this problem by first defining a similarity measure between two graphs, and then presenting a filtering mechanism that reduces the number of candidate graphs. The candidate graphs are then verified by performing expensive graph search operations such as finding maximum common subgraphs. Existing works, however, do not report some similar graphs from a graph database while dissimilar graphs are not discarded during the filtering phase. To overcome this problem, in this paper, we first present a graph distance measure that can identify hidden but similar graphs that could not be discovered by previous graph distance measures. We then devise a series of filtering and validation rules to discard and identify non-matching and definitely-matching graphs, respectively, by calculating lower and upper bounds of the distance between a query and a data graph. To execute these filtering and validation rules efficiently during runtime, an index structure is also proposed. Lastly, a verification algorithm that verifies candidate graphs according to our graph distance measure is presented. Experiments on real datasets show that our approach can efficiently and effectively perform graph similarity search by significantly reducing the number of candidate graphs that must be verified, and by returning similar graphs.

Keywords Graph · Graph similarity search · Graph database · Algorithms

1 Introduction

Graphs are widely used to model complex structured data in many advanced applications such as bioinformatics [12], image processing [17], social networks [28], etc. In these applications, graph queries are performed to discover new domain knowledge, i.e., new graph
patterns are retrieved and analyzed further to discover new relationships between primitive components, or given a set of desired features, graph patterns that contain those features can be retrieved. For example, in an image processing system, an image can be represented by a graph constructed from a set of primitive objects found in the image, and relationships between two objects are represented by edges/paths [17]. Then, the images that have the same features as a given image can be retrieved by comparing the graph structure of each image in the database against the graph structure of the given image. Another example is, in social networks, a community can be modeled by a graph, and the communities that share the same characteristics can be found by checking whether desired characteristics represented by graphs exist in the graph representations of the communities [28].

We have a close look at how graph similarity search is used in biochemistry. Chemical compounds can be well modeled by graphs. In biochemistry and metabolic engineering, graph similarity search is frequently used to find a family of a known/unknown biochemical compound. Figure 1 shows an oxidation pathway of a chemical compound called lignans. By oxidation, a chemical compound X is transformed to X’ by adding/removing organic atoms. In addition, there are certain places in a chemical compound where these addition and removal of atoms can take place. For example, in Figure 1, two edges with O can be added to the middle of compound B to be transformed to compound C. Furthermore, all chemical compounds in Figure 1 are in the same family, because there is a main chemical structure that is commonly shared by all these compounds. In this example, it is the two hexagonal carbon shapes that are found at the end of each compound. Moreover, by a metabolic process, a chemical compound can be transformed to a compound that is in a different family. Figure 2 shows a metabolic pathway from phenylalanine to two chemical compounds, flavanone and lignans, each of which is in a different family. Finding a metabolic pathway is important, because not all chemical compounds are natural products.