A New Mixed Integer Linear Program for the Graph Edit Distance Problem

Mostafa Darwiche, Romain Raveaux, Donatello Conte, Vincent t'Kindt

A New Mixed Integer Linear Program for the Graph Edit Distance Problem

Mostafa Darwiche[1,2], Romain Raveaux[1], Donatello Conte[1], and Vincent T'kindt[2]

[1] Laboratoire d'Informatique Fondamentale et Appliquée (LIFAT, EA6300), Université François Rabelais Tours, France
[2] Laboratoire d'Informatique Fondamentale et Appliquée (LIFAT, EA6300), ERL-CNRS 6305, Université François Rabelais Tours, France
{mostafa.darwiche,romain.raveaux,donatello.conte,tkindt}@univ-tours.fr

Keywords: Graph Edit Distance, Graph Matching, Mixed Integer Linear Program.

1 Introduction

When talking about structural representation of objects and patterns, one could consider graph-based representation, which has been proven in the past decades to be efficient and convenient in many fields. A graph consists of two sets of vertices and edges, where vertices depict the main components of the objects and edges draw the relationships between them. Also, a group of numerical or nominal values can be assigned to vertices and edges in order to provide more information and characteristics. Such values are referred to as attributes/labels. Throughout the years, the attention towards using graphs to model objects have grown in many fields such as *Pattern Recognition* and *Chemionformatics* [4]. The problematic then occurs when having two graphs, how to compare and measure the (dis)similarities between them? Such question has intrigued many researchers who have come up with different classes of problems that are all *Graph Matching problems*. The Graph Edit Distance (GED) problem, which is one of them, provides a dissmilarity measure between two graphs and belongs to *Error-tolerant graph matching* class of problems in particular. The GED problem defines a set of edit operations, which are substitution, insertion and deletion of a vertex or edge where each operation has an associated cost. Solving the problem consists in finding the set of edit operations that transform one graph into another while minimizing the total cost. Let $G = (V, E, \mu, \xi)$ and $G' = (V', E', \mu', \xi')$ be two graphs, with μ and ζ the functions to assign attributes for vertices/edges. The optimal solution of the GED problem is the set of operations $\lambda(G, G') = \{o_1, ..., o_k\}$ with o_i an elementary vertex/edge edit operation and k the number of operations with the minimum cost. This problem has been proved to be NP-hard [5] and numerous heuristics can be found in the literature to solve it. However, only two *mixed integer linear programs* (MILP) exist in the literature [1, 2]. The intent of this work is to propose a new MILP formulation.

2 New MILP formulation for the GED problem

The proposed MILP formulation is inspired from the formulation presented in [2], referred to as (F2). It defines two sets of binary variables: variables $x_{i,k}$ represent the substitution of two vertices $u_i \in V$ and $v_k \in V'$, variables $y_{ij,kl}$ represent the substitution of two edges $e_{ij} \in E$ and $f_{kl} \in E'$. The number of variables in total is $(|V| \times |V'|) + (|E| \times |E'| \times 2)$ for undirected graphs, where y variables are doubled, in comparison with (F2), by considering $y_{ij,kl}$ and $y_{ij,lk}$ for every two edges e_{ij} and f_{kl}. Another variation from (F2) is the constraint that preserves the topology of the graphs, where a new constraint is introduced that only depends on the number of vertices in the new formulation. The number of constraints is then $|V'| + |V| + (|V| \times |V'|)$, against $|V| + |V'| + (|V'| \times |E|)$ constraints in (F2). Two assumptions are made here: the new formulation has a number of constraints independent from the number of edges of the graphs, which should logically lead to a better formulation than (F2) especially in the case of dense (highly connected) graphs. The second assumption is that even having more variables and reducing the number of constraints, the new formulation will perform better than (F2). These assumptions can only be validated through experiments. So far, the new formulation is tested against (F2) on CMU-House graph database [3] of medium graph sizes. Over 660 instances, the new formulation was able to solve 333 instances to optimality against 25 instances by (F2). Both formulations were solved by CPLEX 12.6.0 with 900 seconds as time limit. This preliminary result is promising and shows that the two assumptions hold for this graph database. More graph databases will be considered to evaluate both formulations with different graph sizes and structures in order to confirm the assumptions and the effectiveness of the proposed formulation. The obtained results will be presented at the conference.

References

1. Lerouge, J., Abu-Aisheh, Z., Raveaux, R., Héroux, P., Adam, S.: Exact graph edit distance computation using a binary linear program. In: Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR). pp. 485–495. Springer (2016)
2. Lerouge, J., Abu-Aisheh, Z., Raveaux, R., Héroux, P., Adam, S.: New binary linear programming formulation to compute the graph edit distance. Pattern Recognition 72, 254–265 (2017), https://doi.org/10.1016/j.patcog.2017.07.029
3. Moreno-García, C.F., Cortés, X., Serratosa, F.: A graph repository for learning error-tolerant graph matching. In: Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR). pp. 519–529. Springer (2016)
4. Sanfeliu, A., Alquézar, R., Andrade, J., Climent, J., Serratosa, F., Vergés, J.: Graph-based representations and techniques for image processing and image analysis. Pattern recognition 35(3), 639–650 (2002)
5. Zeng, Z., Tung, A.K., Wang, J., Feng, J., Zhou, L.: Comparing stars: On approximating graph edit distance. Proceedings of the VLDB Endowment 2(1), 25–36 (2009)