Discrete Applied Mathematics 164 (2014) 161–177

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Using local similarity measures to efficiently address approximate graph matching

Segla Kpodjedo *, Philippe Galinier, Giulio Antoniol

DGIGL, Ecole Polytechnique de Montreal, Canada

ARTICLE INFO

Article history:
Received 16 October 2010
Received in revised form 13 January 2012
Accepted 18 January 2012
Available online 10 February 2012

Keywords:
Approximate graph matching
Local similarity measure
Local search

ABSTRACT

In this paper, we investigate heuristics for Approximate Graph Matching (AGM), in particular when it can be formulated as a Maximum Common Edge Subgraph (MCES) problem. First, we observe empirically that initializing a local search with a tiny subset of a known optimal solution always results in much better solutions than starting with an empty solution. The main challenge could then be to retrieve such small subsets for any problem instance. For this purpose, we propose several local similarity measures and evaluate their ability to predict node matches which could be used to start a local search. The resulting algorithm (**SIM-T**) is a classic tabu algorithm that is initialized by a greedy procedure relying mainly, in its earliest steps, on similarity measures.

We conducted experiments on a large collection of random graphs of various orders (from 50 to 3000 nodes) and densities. Results obtained are mostly excellent, especially on similar pairs of labeled graphs. Comparisons made with two recent state-of-the-art algorithms – *"BP"* and *"PATH"* – indicate a superiority of our approach, in terms of both scores and computation times.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Graph comparison is an important question which can be answered using graph matching techniques. Graphs to be matched can represent images [24], molecules [19,26], software artifacts [1,14], and in most settings, desired matchings are neither necessarily complete[1] nor require perfect correspondences for all matched elements. Consequently, flexible (approximate) formulations have been proposed to define graph matching as the search for matchings which optimize a particular criterion. The Maximum Common Edge Subgraph (MCES) [19] problem is among the most well-known approximate graph matching (AGM) formulations and consists, given two graphs, in *finding a common partial subgraph with a maximal number of edges*. Other well-known AGM formulations used in the literature include the Error-Tolerant Graph Matching (ETGM) problem [5,22] (find the cheapest transformation[2] from one graph to the other), and the Weighted Graph Matching (WGM) problem [25] (find a matching which minimizes the distance between the adjacency matrices of two weighted graphs). Various kinds of techniques [11,19,20,27] have been proposed to address those formulations of graph matching. We refer the reader to the extensive review and classification provided in [6].

In general, the choice of an AGM formulation (or objective function) follows some assumptions about the kind of graphs one is trying to match; for instance, WGM formulations are indicated for complete matchings between graphs with real numbers (weights) on their arcs. In our work, a core assumption is that the graphs to be matched are similar in the sense

* Corresponding author. Tel.: +1 514 733 8449; fax: +1 514 340 4657.
E-mail address: segla.kpodjedo@polymtl.ca (S. Kpodjedo).

[1] Some nodes may remain unmatched.

[2] Edit operations are defined and costs are assigned to them.

0166-218X/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2012.01.019

that a majority of nodes and arcs from one graph can be perfectly matched to nodes and arcs from the other graph. For example, one graph may be the result of transformations (due to noise or evolution) applied to the other graph and affecting a limited number of its nodes and arcs. Such assumption is particularly relevant for graphs with symbolic labels (adequate representations in many areas, from diagrams [1,14] to molecules [19,26]). It is also compatible with many graph matching formulations such as Graph Isomorphism [17], Subgraph Isomorphism [7], Maximum Common Induced Subgraph (MCIS) [8], Maximum Common Edge Subgraph (MCES) [19], etc.

Our approach for solving such AGM problems combines local search techniques (such as the Tabu search [9]) and node similarity measures. Local search algorithms represent a family of meta-heuristics in which a solution is iteratively improved, through limited modifications, with the purpose of gradually moving toward optimal areas. Regarding node similarity measures, they are often used in graph matching techniques [11,20] and can be presented as values assigned to pairs of nodes to express the likelihood of matching them.

In [13], we found that initializing a local search with a few node matches taken from a known (near-) optimal solution is enough to get excellent matchings and we proposed a similarity measure meant to predict the "right" node matches. As a follow-up, we proposed in [12] an algorithm integrating similarity measures to the objective function and obtained very interesting results on a synthetic benchmark.

The work reported in the current paper gives insight on how local similarity measures for node matches can be devised, and investigates additional mechanisms for filtering out ambiguous and misleading node matches. As a result, new and more effective local similarity measures are proposed and experimentally evaluated. Two algorithms using similarity, including the similarity-aware tabu algorithm *SIM-T*, are thus tested and compared with two recent algorithms: *BP*[3] [20] and *PATH* [27]. Furthermore, the random graph generator previously used in [12,13], has been extended and can now model undirected graphs and graphs labeled on nodes; hence, our experiments now include a larger spectrum of graphs. Experimental results on those synthetic graphs for three specific AGM problems (including MCES and WGM) suggest that our algorithm *SIM-T* outperforms *BP* and *PATH*, in terms of both quality (of the returned matching) and computational speed.

The remaining of the paper is organized as follows. We first present, in Section 2, our definition of the AGM problem, then in Section 3, the benchmarks on which our approach is evaluated. Section 4 introduces our tabu procedure and discusses preliminary experiments. Section 5 presents our different propositions for the computation of local node similarity and their evaluation. Section 6 describes our proposed algorithms (notably *SIM-T*) and is followed by Section 7 which presents our experimental plan for the comparisons with *BP* and *PATH*. Results of all algorithms are presented in Section 8. We finally conclude in Section 9 with a general discussion including the limitations of our approach and some perspectives.

2. Problem definition

In the following, we introduce preliminary definitions and notations about graphs and matchings. Then, we analyze what a "good" matching may mean and finally propose the formal definition of the AGM problem we use in this paper.

2.1. Preliminary definitions

The graphs we consider are directed,[4] and they have labels[5] on their vertices and/or arcs.

Let Σ_V and Σ_A represent two finite sets of symbols. A graph, labeled on alphabets Σ_V for nodes and Σ_A for arcs, is defined as a quadruple (V, A, l_V, l_A) where: V is the finite set of vertices (or nodes); $A \subseteq V \times V$ is the set of arcs; $l_V : V \to \Sigma_V$ is the node labeling function; and $l_A : A \to \Sigma_A$ is the arc labeling function.

For practical reasons, we also define another function $L_A : V \times V \to \Sigma_{A+} = \Sigma_A \cup \{\#\}$ as follows: $L_A(x, y) = l_A(x, y)$ if $(x, y) \in A$, otherwise $L_A(x, y) = \#$. In other words, L_A coincides with l_A on A, and the symbol # is used in order to represent the absence of an arc between two vertices.

Let us consider two graphs $G_1 = (V_1, A_1, l_{V1}, L_{A1})$ and $G_2 = (V_2, A_2, l_{V2}, L_{A2})$ labeled on the alphabets Σ_V and Σ_A. A matching between the two graphs is any relation $\mu \subseteq V_1 \times V_2$ such that each vertex is matched to at most one vertex in the other graph (the one-to-one constraint): $\forall x, y \in V_1, \forall z, t \in V_2, (x, z), (x, t) \in \mu \Rightarrow z = t$ and $(x, z), (y, z) \in \mu \Rightarrow x = y$. In the following, an element of μ (a couple) will be referred to as a *node match*.

2.2. Assessing the quality of a matching

Let us consider two graphs G_1 and G_2 and a matching μ between them. A node match (x_1, x_2) in μ matches a vertex $x_1 \in V_1$ to a vertex $x_2 \in V_2$. We either observe a *perfect vertex label match* ($l_V(x_1) = l_V(x_2)$) or a *vertex label error* ($l_V(x_1) \neq l_V(x_2)$). Similarly, a couple of node matches $((x_1, x_2), (y_1, y_2)) \in \mu \times \mu$ matches a pair (x_1, y_1) of vertices of V_1 to a pair (x_2, y_2) of vertices of V_2. We identify four possible cases: (i) a *perfect arc label match*: the arcs share the same

[3] BP stands for Bipartite.

[4] Undirected graphs can be treated as symmetric directed graphs.

[5] When there is only one label, the graphs can be considered unlabeled.

Download English Version:

https://daneshyari.com/en/article/419094

Download Persian Version:

https://daneshyari.com/article/419094

Daneshyari.com