Algorithm Selection for Maximum Common Subgraph

Paulius Dilkas

27th October 2017

1 The Problem

Maximum common induced subgraph for undirected graphs. We are considering three types of labelling: no labelling, labels on vertices, and labels on both vertices and edges. When labelling is used, every vertex (and possibly edge) gets assigned a random (?) label (are they uniformly distributed?). The number of distinct labels is approximately equal to 33% of the number of vertices, just like in what studies? Some of the graphs also contain loops, but no multiple edges (confirm!).

2 Algorithms

Clique encoding [9] $k\downarrow$ [4] McSplit and McSplit $\downarrow$ [7]

3 Problem Instances

In order to determine which algorithm should be used for which problem instance, we run all algorithms on two databases that contain a large variety of graphs differing in size, various characteristics, and the way they were generated.

The McSplit paper [7] used the same datasets to compare these (and a few constraint programming) algorithms and found McSplit to win with unlabelled graphs, the clique encoding to win with labelled graphs, and McSplit $\downarrow$ to win with the `largerGraphs` dataset. However, in some cases the difference in performance between McSplit and the clique encoding or between McSplit $\downarrow$ and $k\downarrow$ was very small.

(Somewhere) 1000 s limit, 512 GB limit (clique becomes impossible for some instances), insert CPU specs.

3.1 Labelled graphs

All of the labelled graphs are taken from the ARG Database [2, 3], which is a large collection of graphs for benchmarking various graph-matching algorithms.

The graphs are generated using several algorithms:

- randomly generated,
- 2D, 3D, and 4D meshes,
- and bounded valence graphs.

Furthermore, each algorithm is executed with several (3–5) different parameter values. The database includes 81400 pairs of labelled graphs. Their unlabelled versions are used as well.

3.1.1 Characteristics of Graph Labelling

Definition 3.1. A *(vertex) labelled graph*

3.2 Unlabelled graphs

We also include a collection of benchmark instances for the subgraph isomorphism problem[1] (with the biochemical reactions dataset excluded since we are not dealing with directed graphs). It contains only unlabelled graphs and consists of the following sets:

images-CVIU11 Graphs generated from segmented images. 43 pattern graphs and 146 target graphs, giving a total of 6278 instances.

meshes-CVIU11 Graphs generated from meshes modelling 3D objects. 6 pattern graphs and 503 target graphs, giving a total of 3018 instances. Both `images-CVIU11` and `meshes-CVIU11` datasets are described in [1].

images-PR15 Graphs generated from segmented images [11]. 24 pattern graphs and a single target graph, giving 24 instances.

LV Graphs with various properties (connected, biconnected, triconnected, bipartite, planar, etc.). 49 graphs are paired up in all possible ways, giving $49^2 = 2401$ instances.

scalefree Scale-free networks generated using a power law distribution of degrees (100 instances).

si Bounded valence graphs, 4D meshes, and randomly generated graphs (1170 instances). This is the unlabelled part of the ARG database. `LV`, `scalefree`, and `si` datasets are described in [10, 12].

phase Random graphs generated to be close to the satisfiable-unsatisfiable phase transition (200 instances) [8].

[1] `http://liris.cnrs.fr/csolnon/SIP.html`

largerGraphs Large random and real-world graphs. There are 70 graphs, giving $70^2 = 4900$ instances. This set is not actually part of the main collection of benchmark instances, but is used in [4, 6, 7].

Note that this set of instances was taken from the repository of [7] and has some minor differences from the version on Christine Solnon's website.

4 Features

The initial set of features was based on the algorithm selection paper for the subgraph isomorphism problem [6]:

- number of vertices,
- number of edges,
- density,
- number of loops,
- mean degree,
- maximum degree,
- standard deviation of degrees,
- whether the graph is connected,
- mean distance between all pairs of vertices,
- maximum distance between all pairs of vertices,
- proportion of all vertex pairs that have a distance of at least 2, 3, and 4.

We excluded feature extraction running time as a viable feature by itself since it would not provide any insight into what properties of the graph affect Counting the number of (distinct and not) labels was later rethought to be unnecessary and replaced by a boolean feature "labelled" because if labelling is enabled, the number of labels is equal to the number of vertices and the number of distinct labels is equal to 33% of that.

Features that could be computed if we end up using a presolver:

- uniformity of the distribution of edges,
- how many candidate pairs were removed,
- proportion of candidate pairs removed over all pairs,
- min values removed per variable,
- max values removed per variable,
- CPU time taken to compute all this.

4.1 Distributions of Features

In this section we plot and discuss how the selected features are distributed...

5 Selection Model

We're using LLAMA [5]. Describe k-folding.

References

[1] Guillaume Damiand et al. "Polynomial algorithms for subisomorphism of nD open combinatorial maps". In: *Computer Vision and Image Understanding* 115.7 (2011), pp. 996–1010. DOI: 10.1016/j.cviu.2010.12.013. URL: https://doi.org/10.1016/j.cviu.2010.12.013.

[2] M. De Santo et al. "A large database of graphs and its use for benchmarking graph isomorphism algorithms". In: *Pattern Recogn. Lett.* 24.8 (May 2003), 10671079. ISSN: 0167-8655. DOI: 10.1016/S0167-8655(02)00253-2. URL: http://dx.doi.org/10.1016/S0167-8655(02)00253-2.

[3] P. Foggia, C. Sansone and M. Vento. "A Database of Graphs for Isomorphism and Sub-Graph Isomorphism Benchmarking". In: -. 1st Jan. 2001, 176187.

[4] Ruth Hoffmann, Ciaran McCreesh and Craig Reilly. "Between Subgraph Isomorphism and Maximum Common Subgraph". In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.* Ed. by Satinder P. Singh and Shaul Markovitch. AAAI Press, 2017, pp. 3907–3914. URL: http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14948.

[5] Lars Kotthoff. *LLAMA: Leveraging Learning to Automatically Manage Algorithms.* Tech. rep. arXiv:1306.1031. arXiv, June 2013. URL: http://arxiv.org/abs/1306.1031.

[6] Lars Kotthoff, Ciaran McCreesh and Christine Solnon. "Portfolios of Subgraph Isomorphism Algorithms". In: *Learning and Intelligent Optimization - 10th International Conference, LION 10, Ischia, Italy, May 29 - June 1, 2016, Revised Selected Papers.* Ed. by Paola Festa, Meinolf Sellmann and Joaquin Vanschoren. Vol. 10079. Lecture Notes in Computer Science. Springer, 2016, pp. 107–122. ISBN: 978-3-319-50348-6. DOI: 10.1007/978-3-319-50349-3_8. URL: https://doi.org/10.1007/978-3-319-50349-3_8.

[7] Ciaran McCreesh, Patrick Prosser and James Trimble. "A Partitioning Algorithm for Maximum Common Subgraph Problems". In: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*. Ed. by Carles Sierra. ijcai.org, 2017, pp. 712–719. ISBN: 978-0-9992411-0-3. DOI: 10.24963/ijcai.2017/99. URL: https://doi.org/10.24963/ijcai.2017/99.

[8] Ciaran McCreesh, Patrick Prosser and James Trimble. "Heuristics and Really Hard Instances for Subgraph Isomorphism Problems". In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. Ed. by Subbarao Kambhampati. IJCAI/AAAI Press, 2016, pp. 631–638. ISBN: 978-1-57735-770-4. URL: http://www.ijcai.org/Abstract/16/096.

[9] Ciaran McCreesh et al. "Clique and Constraint Models for Maximum Common (Connected) Subgraph Problems". In: *Principles and Practice of Constraint Programming - 22nd International Conference, CP 2016, Toulouse, France, September 5-9, 2016, Proceedings*. Ed. by Michel Rueher. Vol. 9892. Lecture Notes in Computer Science. Springer, 2016, pp. 350–368. ISBN: 978-3-319-44952-4. DOI: 10.1007/978-3-319-44953-1_23. URL: https://doi.org/10.1007/978-3-319-44953-1_23.

[10] Christine Solnon. "AllDifferent-based filtering for subgraph isomorphism". In: *Artif. Intell.* 174.12-13 (2010), pp. 850–864. DOI: 10.1016/j.artint.2010.05.002. URL: https://doi.org/10.1016/j.artint.2010.05.002.

[11] Christine Solnon et al. "On the complexity of submap isomorphism and maximum common submap problems". In: *Pattern Recognition* 48.2 (2015), pp. 302–316. DOI: 10.1016/j.patcog.2014.05.019. URL: https://doi.org/10.1016/j.patcog.2014.05.019.

[12] Stéphane Zampelli, Yves Deville and Christine Solnon. "Solving subgraph isomorphism problems with constraint programming". In: *Constraints* 15.3 (2010), pp. 327–353. DOI: 10.1007/s10601-009-9074-3. URL: https://doi.org/10.1007/s10601-009-9074-3.