Using Background Knowledge for Graph Based Learning: a Case Study in Chemoinformatics

Thashmee Karunaratne and Henrik Boström

Department of Computer & Systems Sciences (DSV), Stockholm University and
Royal Institute of Technology, Forum 100, SE 164 40, Kista, Sweden
{si-thk, henke}@dsv.su.se

Abstract. The benefit of incorporating background knowledge in the learning process has been successfully demonstrated in numerous applications of ILP methods. Nevertheless the effect of incorporating background knowledge in graph learning has not yet been systematically explored. A first step in this direction is taken in this work, where a case study in chemoinformatics is presented, in which various types of background knowledge are encoded in graphs that are given as input to a graph learner. It is shown that the type of background knowledge encoded indeed has an effect on the predictive performance, and it is concluded that encoding appropriate background knowledge may even be more important than selecting which graph learning algorithm to use.

1. Introduction

Inductive logic programming methods is one class of methods for which the benefit of incorporating relevant background knowledge has been demonstrated in numerous applications (e.g. [1]). Graph learning methods are another class of methods that is flexible in terms of data that can be encoded [2]. However, almost all research on graph learning methods concerns improving existing search algorithms or heuristic measures, and the effect of different types of background knowledge on the predictive performance has not been studied in any systematic way [3]. Srinivasan et al put forward the question: “how does domain specific background information affect the performance of an ILP system?” [1]. It seems worthwhile to consider this question also for graph learning methods, and a first step towards an answer is taken in this work by a case study in the domain of chemoinformatics.

The rest of the paper is organized as follows. In the next section, the graph learning method that is used in this study is briefly presented. In section 3, we present an empirical evaluation of different types of background knowledge. Finally, in section 4, we give concluding remarks and outline future work.

2. Graph Learning with DIFFER

DIFFER is a graph learning method that employs a finger printing approach for graph transformation, that successfully avoids the graph isomorphism problem and supports combining isolated substructures [6]. The graph of each example is represented by a set of triples *(L_i, L_j, E_k)*, such that there is an edge labeled E_k between nodes N_i and N_j, which are labeled L_i and L_j respectively. Such a set is referred to as a *finger print*. The finger prints are used for substructure search in such a way that for all pairs of examples, the intersection of their finger prints, which is referred to as *the maximal common substructure*, is formed, and ranked according to their frequency in the entire set of examples (i.e., the number of finger prints for which the maximal common substructure is a subset). An upper and lower threshold is applied to select the most informative substructures (features) for classification. The selected elements of the finger prints are used as (binary) features, allowing predictive models to be built by any standard attribute-value learner.

3. Empirical Evaluation

Two datasets from the chemoinformatics domain are considered in this study: mutagenesis [4], and carcinogenesis, [5], and for each of these, two levels of background knowledge are considered: (1) atom-bond descriptions of each molecule, and (2) two-dimensional substructures such as benzene rings and nitro groups etc. Five node and edge definitions are considered with respect to the two levels. Definition D1 has nodes labeled with atom name, type and the bond types connected, i.e., *node(atom_name atom_type,[bond_type/s])* but no edge labels. In D2, bond types are detached from node labels of D1 and used as edge labels: *node(atom_name,atom_type)* and *edge(bond_type)*. D3 additionally includes the number of similar edges between two atoms in the edge label, encoded as *edge(bond_type,count)*. D1, D2 and D3 all represent background knowledge on level 1, while definitions D4 and D5 represent background knowledge on level 2. D4 has the same edge labels as D2, but the node labels also include lists of structures that the atoms are part of, i.e., *node(atom_name, atom_type, [list of structures])*, and D5 extends D4 by using edge labels similar to D3. Features generated by DIFFER are used as input to a number of standard machine learning methods as implemented in WEKA [9]. Results are summarized in Table 1, where the best learning method for each feature set is shown within parentheses[1]. According to McNemar's test, there is a significant difference between the accuracies of the highest and lowest levels of background knowledge.

Data set	Accuracy				
	D1	D2	D3	D4	D5
Mutagenesis	80.61% (RF)	80.61% (RF)	84.04% (SVM)	87.77% (SVM)	88.3% (SVM)
Carcinogenesis	61.25% (RF)	62.1% (RF)	68.73% (SVM)	71.03% (SVM)	75.0% (SVM)

Table 1. Performance of DIFFER with 5 different graph encodings

[1] RF = Random forest, SVM = Support Vector Machine

4. Concluding Remarks

Our study shows that the predictive performance of a graph learner is highly dependent on the way in which nodes and edges are formed, and it is shown that by incorporating background knowledge concerning two-dimensional substructures, the accuracy can be substantially improved for both the mutagenesis and carcinogenesis domains. Since the accuracies reported for existing graph learning methods on these data sets (e.g., Tree$^2\chi^2$ [7] achieves an accuracy of 80.26% on the mutagenesis data and SUBDUE-CL [8] achieves 61.54% on the carcinogenesis data) are far below the best results in our study, one may conclude that even a quite simple graph learning approach, such as DIFFER, may outperform more elaborated approaches, such as frequent subgraph methods or kernel methods, if appropriate background knowledge is encoded in the graphs. One direction for future research is to investigate if these encodings also have a similar positive effect on the more complex algorithms. Another direction is to study the effect of including more complex types of background knowledge, such as 3D molecular descriptions.

References

[1]. Srinivasan A., King, R.D, and Muggleton S, (1999), "*The role of background knowledge: using a problem from chemistry to examine the performance of an ILP program*", TR PRG-TR-08-99, Oxford

[2]. Cook, J. and Holder, L., (2003), "*Graph-Based Relational Learning: Current and Future Directions*", ACM SIGKDD, V:5, I:1, pp 90-93

[3]. Jiang, W., Vaidya, J., Balaporia, Z., Clifton, C., and Banich, B., (2005), "*Knowledge Discovery from Transportation Graph Data*", IEEE International Conference on Data Engineering, Tokyo, Japan.

[4]. Debnath, A.K. Lopez de Compadre, R.L., Debnath, G., Shusterman, A.J., and Hansch, C. (1991), "Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds: Correlation with molecular orbital energies and hydrophobicity", JMC. 34:786-797

[5]. US National Toxicology program, http://ntp.niehs.nih.gov/index.cfm?objectid=32BA9724-F1F6-975E-7FCE50709CB4C932

[6]. Karunaratne, T. and Boström, H., (2006), "*Learning from structured data by finger printing*", To appear in Proceedings of 9th Scandinavian Conference of Artificial Intelligence, Helsinki, Finland.

[7]. Bringmann, B., and Zimmermann, A., (2005), "*Tree - Decision Trees for Tree Structured Data*", Proceedings of the 9th PKDD

[8]. Gonzalez, J., Holder, L. B. and Cook, D. J. (2001), "*Application of Graph-Based Concept Learning to the Predictive Toxicology Domain",* Proceedings of the Predictive Toxicology Challenge Workshop

[9]. Ian H. Witten and Eibe Frank (2005) "*Data Mining: Practical machine learning tools and techniques*", 2nd Edition, Morgan Kaufmann