

Learning from Structured Data by Finger Printing

Thashmee Karunaratne

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

Henrik Boström

Department of Computer & Systems Sciences
Stockholm University and
Royal Institute of Technology
Forum 100
SE 164 40, Kista, Sweden
henke@dsv.su.se

Abstract

Current methods for learning from structured data are limited w.r.t. handling large or isolated sub-structures and also impose constraints on search depth and induced structure length. An approach to learning from structured data using a graph based canonical representation method of structures, called finger printing, is introduced that addresses the limitations of current methods. The method is implemented in a system, called DIFFER, which is demonstrated to compare favourable to existing state-of-art methods on some benchmark data sets. It is shown that further improvements can be obtained by combining the features generated by finger printing with features generated by previous methods.

1 Introduction

In many domains, in which a model is to be generated by machine learning, examples are more naturally represented by structured terms than fixed-length feature vectors. For example, in chemo informatics, molecules are naturally represented as two or three dimensional structures of atoms. Another example is when having data on XML format, which could be directly mapped on tree structures.

Several approaches to learning to classify structured data have been introduced in the field of machine learning. The structure classification problem has been addressed as a rule learning problem [1,2,3,4,5], as a graph mining problem [6,7,8,9,10,11] and as a propositionalization problem [12,13,14]. The methods address two main varieties of problems. The first category is discovery methods of features that best discriminate between the classes. Krogel et al [12], for example, introduce an approach to selecting “most interesting” features of structured data that discriminate between the classes. A key requirement of the feature discovery methods is that the discovered features should be comprehensible.

In contrast to discovery methods, classification methods generate global models for classifying all examples, but the models need not necessarily be

comprehensible. Most classification methods assume that all examples can be represented by fixed-length feature vectors, and finding features that suitably contain the relevant information in this format, can be considered a major knowledge engineering bottleneck for these types of method. This is true in particular when the examples are most naturally represented as structured terms (e.g., trees, lists, etc.). Existing methods for structure classification are limited w.r.t. the complexity and volume of structured data, which is described in detail in section 2, and hence more robust methods for learning from structured data are needed. The method presented in this paper, which extracts features from structures by a method called *finger printing*, is motivated exactly by this need.

The rest of the paper is organized as follows. Section 2 discusses the state-of-art structure classification methods and their limitations. Section 3 introduces the novel finger printing method, which implementation is described in section 4. Section 5 presents an empirical evaluation, comparing the novel method to some state-of-the-art methods on some benchmark datasets. Finally, section 6 describes the concluding remarks and possible further extensions to the demonstrated theory.

2. Current Approaches to Learning from Structured Data

Current state-of-art methods for feature discovery and classification use several forms of structure transformation. Inductive logic programming [5] has drawn immense popularity since its inception, mainly due to that background knowledge and data as well as the result of the methods are represented in the same format: logic programs. Propositionalization methods is one class of ILP methods that transform the relational rule learning problem into a standard attribute-value learning problem by identifying suitable features [12,15]. However, these, as well as the standard ILP methods, are often faced with a huge search space, either for which constraints have to be imposed, or the domain has to be restricted in terms of the number of examples considered [15]. The limits on search depth and clause length typically result in that the substructures discovered by ILP methods are quite small and usually are limited to 5-6 structural relations [16].

Graph mining methods are efficient enough to discover considerably large substructures, unlike the propositionalization methods [9,16]. Although the current algorithms already perform quite well, they still have some limitations. One of these is that the discovered graphs are by necessity connected. This prevents inclusion of isolated or far away frequent nodes or sub graphs. Thus two fragments within a graph that are not connected are not being considered in conjunction by current methods, even if the contribution of these fragments when taken together would be a highly potential feature. Another limitation of current graph mining methods is that they only consider exact matches of the sub-graphs and hence do not allow mining “similar sub-graphs” [17], i.e., sub-graphs that are not exactly equal to each other, but differ only by a few nodes. For example, in a chemoinformatics application, different molecules may have carbon chains of different lengths, but other atoms such as nitrogen and oxygen are connected to the carbon chains in a same topology, i.e., substructures differ only by its length of the carbon chain. These substructures are not equal to each other since they differ by the length of the carbon chain, but rather “similar” since the topology of the substructure is same. Current methods consider these substructures as completely different, since the substructures do not exactly match with each other. A further description about similar sub-graphs can be found in [17]. Furthermore, memory and runtime are challenges for most of the graph mining algorithms [17]. It is an open question how to realize the class distribution over sub-graphs without searching different branches of the lattice several times [7]. The need for inexact graph match-

ing might become more and more important in this context. As Washio & Motada [7] reports “Even from a theoretical perspective, many open questions on the graph characteristics and the isomorphism complexity remain. This research field provides many attractive topics in both theory and application, and is expected to be one of the key fields in data mining research”.

In summary, ILP/propositionalization methods can be useful for learning from structured data if discovery of small substructures is sufficient, but they do require that non-trivial constraints on the search space are provided. If the domain of interest requires the discovery of large substructures, graph mining methods are often more suited. However, these cannot be used to discover several isolated substructures and require exact matching of substructures. Hence, there are demands for much robust methods for learning from structured data.

3. Finger Printing

Our approach to structure classification employs a graph transformation method which could address some of the limitations discussed in the previous section. The method has the ability to combine isolated substructures and has the potential to discover “similar substructures”. It also does not require any constraint to be imposed on the search space. Our method follows a data to model (bottom – up) search strategy and digs down any potential substructures irrespective of its length. Since the graphs are transformed into a canonical form called *finger print*, which is a hashed form of a sparse vector of zeros and ones, the computational cost in manipulation of the graphs are considerably low. Our method could be applied to any form of structured data, from trees to undirected graphs, from sequences to tuples etc., and hence all these types of structured data are referred to as graphs during the rest of this paper.

3.1 The finger printing method

Several methods have been suggested to represent structured data for learning algorithms, and canonical forms of graphs are among the most popular due to their computational simplicity. Our method of transforming graphs into a canonical form is called *finger printing*. The method first includes a preprocessing step in which each structured data is represented by a labeled graph in the *forest*, and then transforming each graph into an adjacency matrix using the definitions 1 and 2 given below.

3.3 Feature construction

The feature set used for classification is the most discriminative set of substructures found by the pairwise maximal substructure search algorithm. We use the standard covering statistic for finding the most discriminating set of substructures.

Definition 5 *Covering Statistic: The coverage of a substructure s_i is defined as*

$$C(s_i) = p(s_i) = n(s_i) / N$$

where $n(s_i)$ is the number of examples for which the substructure is present in the example, and N is the total number of examples present in the training set.

Each substructure discovered using the maximal common substructure search algorithm is evaluated by the coverage statistic given in Definition 5. The weighted substructure set is then ranked in descending order. We also use a maximum and a minimum threshold in order to filter the most discriminating set of substructures. This set of substructures obtained through the search and filter procedure are the most discriminating feature set for the domain of examples. This feature set is then used when building the classification model.

4. Implementation

We have developed a feature construction and classifier system called DIFFER (**D**iscovery of **F**eatures using **F**inger **E**R prints), using the methodology described in section 3. DIFFER handles examples containing structured data. Therefore inputs to the system are the structures. DIFFER produces an output containing the derived feature set and the presence/non-presence of those features in examples in a form of a text file that can be used by most standard classification methods (at the moment the output file is of the format of .arff which is the recognizable format for WEKA data mining toolkit). In summary the main features of DIFFER are:

- any type of structured data can be handled such as trees, graphs, tuples, strings etc. etc.
- the canonical form used for graph transformation is the finger print
- isolated sub-graphs/nodes can be identified. i.e., the substructures need not to be necessarily connected.
- similar sub-graphs can be identified as well. i.e., the substructures treated as common are not equivalent to each other but “similar”.

- There are no constraints on length of the searched substructure, and therefore sub-graphs of any size may be identified.

5. Experimental Evaluation

We have used some benchmark datasets to compare the performance of DIFFER with other available methods for learning from structures. One benchmark dataset concerns predicting mutagenicity on *Salmonella typhimurium*, which comprises of 230 molecules. Debnath et. al. [18] grouped 188 examples out of these as “regression friendly”. This subset contains 125 examples with a positive log mutagenesis whereas the remaining 63 examples are inactive with respect to log mutagenesis. We have used this regression friendly subset of the mutagenesis dataset for our experimental evaluation. This is a two class problem of predicting whether a compound is mutagenic or not.

The second benchmark data set concerns the very popular east-west train problem [19], which contains 20 trains where 10 each are headed to east and west respectively. The task is to identify the characteristics of the trains that make them headed east or west. The third dataset, carcinogenesis is also, like the first, from the domain of chemo-informatics. The dataset was originally developed within the US national toxicology program [20]. It consists of 298 compounds that have been shown to be carcinogenic or not in rodents. Although the original dataset contains 3 classes of carcinogenesis, these were treated as one class as done in most previous studies.

The three benchmark datasets were given as input to DIFFER and a summary of the outcome of the method is given in Table 1 below.

Dataset	No. of examples	features ¹ selected
Muta	188	171
Trains	20	30
Carci	298	154

Table 1: Summary of induced features for benchmark datasets

We have used all the data as training examples during feature generation. This does not impose any bias on feature construction since we are not considering class distribution of features during the feature construction. Transformation of structures into graphs is carried out according to the definition 2.

¹ Since the set of features discovered by Maximal common substructure search algorithm is small we did not apply any threshold here.

For the first and third datasets, molecules are represented by graphs in such a way that a node of graph is an atom in the molecule with its label (carbon for example), plus the bonds attached to the atom. For example, a carbon atom with two single bonds and a double bond is converted to a labeled node $c[112]$. The second dataset contains trains as its structures, and each train has cars with are described by a set of properties, e.g., shape, no. of wheels etc. A node in this domain is represented by a tuple $car(<properties>)$. For example a car with a long rectangular shape, a flat roof, sides that are not double and 3 wheels, is represented by $car(rectangle,long,not_double,flat,3)$.

Features generated by DIFFER is used as input to a standard machine learning method. The method used in this experiment is random forest [21] with 50 trees and where 10 random features are evaluated at each node, as implemented in the WEKA data mining toolkit [22]. 10 fold cross validation is used as the evaluation method. The results we obtained with DIFFER were compared with existing state-of-the-art methods, including a propositionalization method, RSD [12] and two graph mining methods, SUBDUE-CL [9] and $Tree^2\chi^2$ [11]. We have used all the data in each of the 3 benchmark datasets as training examples for RSD as well. The WEKA [22] implementation of random forest of 50 trees with 10 random features for evaluation at each node is used for reproduction of accuracies in RSD. 10 fold cross validation is used as the evaluation method. We did not reproduce the results for SUBDUE-CL and $Tree^2\chi^2$, but give the accuracies reported in [9] and [11] respectively. The results are summarized in Table 2.

Dataset	Accuracy				
	DIF-FER	RSD	SUB-DUE-CL	$Tree^2\chi^2$	DIF-FER + RSD
Trains	80%	75%	-	-	85%
Mutagenesis	80.61%	88.86%	-	80.26%	92.76%
Carcinogenesis	65.25%	54.37%	61.54%	-	65.33%

Table 2: Comparison of DIFFER with some state-of-the-art methods

These results show that DIFFER may perform as well as or better than existing methods. We also have studied what happens when merging features of DIFFER with those of other methods. When merging the feature set of RSD with the feature set of DIFFER, an increase in accuracy was observed (final column of Table 2). We analyzed the feature set generated by RSD for the mutagenesis dataset and rather surprisingly, we found that it did not contain any atom-bond features. Nonetheless it con-

tained global molecular structure properties such as whether or not two connected nitro groups are present. In contrast to this, the features generated by DIFFER contains inner structural information of atom-bond connections. The experiment demonstrates that by merging these two complementary sets of features the accuracy of the resulting model can be increased.

6. Concluding Remarks

Learning from structured data is an important challenge for machine learning methods, with many important applications, for example within analyzing data from the web, in chemo- and bioinformatics, in management and business transaction domains. These domains are often complex not only in terms of the presence of structures, but also often in terms of the size of the data sets to be analyzed. Existing techniques for learning from structured data are demonstrated to have a number of limitations w.r.t. to effectively analyzing the data due to inability to discover isolated sub-graphs or capture topology of similar sub-graphs and by requiring that non-trivial constraints on the search space is provided, something which may prevent the discovery of large interesting substructures. In order to overcome these limitations, a novel method, that transforms structured data into a canonical representation, called finger prints, has been presented.

The new method, which has been implemented in a system, called DIFFER, has been shown to be competitive with the existing state-of-the-art methods on some standard benchmark data sets, without imposing constraints on the search space. The reason for its effectiveness can be explained by its ability to mine large as well as isolated discriminative sub-graphs. A very interesting observation is that the classification performance can be improved by merging the features generated by DIFFER with features generated by other methods and thereby integrating the different qualities of several methods. Thus rather than searching for new feature extraction methods that on its own compete with existing methods, it appears to be a promising approach to search for new methods that generate complementary features.

There are several possible directions for future work. At present DIFFER's substructure search is a pair-wise approach, for which the computational cost grows quadratically with the number of examples. A more efficient procedure could be obtained by using some incremental way of searching for the substructures. Sampling of which pairs to consider is also a straightforward way of controlling the computational cost [3]. Alternatives to the use of the

covering statistic in conjunction with maximum and minimum thresholds could also be explored. Candidates for this include model driven approaches such as voting by the convex hull or a coverage measure.

The promising result of combining the features generated by DIFFER and RSD also leads to considering merging the features of DIFFER and other methods, perhaps further improving the predictive performance.

References

- [1]. Zaki, M.J., Aggarwal, C.C. (2003), "*XRules: an effective structural classifier for XML data*" KDD, Washington, DC, USA, ACM 316–325
- [2]. Quinlan, J. R., Cameron-Jones, R. M., (1993), "*FOIL*", Proceedings of the 6th European Conference on Machine Learning, Lecture Notes in Artificial Intelligence, Vol. 667, pp. 3-20. Springer-Verlag (1993)
- [3]. Muggleton, S. and Feng, C., (1992), "*Efficient induction in logic programs*", Inductive Logic Programming, pages 281-298. Academic Press
- [4]. 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*", Technical Report PRG-TR-08-99, Oxford University Computing Laboratory, Oxford, 1999.
- [5]. Muggleton, S., (1995), "*Inverse entailment and Progol*", New Generation Computing, Special issue on Inductive Logic Programming, 13(3-4):245-286
- [6]. Cook, J. and Holder, L., (1994), "*Substructure discovery using minimum description length and background knowledge*", Journal of Artificial Intelligence Research, 1:231-255
- [7]. Washio T. and Motoda, H. (2003), "*State of the Art of Graph-based Data Mining*", SIGKDD Explorations Special Issue on Multi-Relational Data Mining, pp 59-68, Volume 5, Issue 1
- [8]. Dehaspe, L. and Toivonen, H, (1999). "*Discovery of frequent datalog patterns*", Data Mining and Knowledge Discovery, 3(1):7-36
- [9]. 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
- [10]. De Raedt, L. and Kramer, S., (2001), "*The levelwise version space algorithm and its application to molecular fragment finding*" In IJCAI'01: Seventeenth International Joint Conference on Artificial Intelligence, volume 2, pages 853-859
- [11]. Bringmann, B., and Zimmermann, A., (2005), "*Tree - Decision Trees for Tree Structured Data*", Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2005), Notes in Artificial Intelligence, (LNAI) 3721, pp. 46-58, Springer (2005)
- [12]. Krogel, M-A., Rawles, S., Železný, F., Flach, P. A., Lavrač, N., and Wrobel, S., (2003). "*Comparative evaluation of approaches to propositionalization*", Proceedings of the 13th International Conference on Inductive Logic Programming (ILP'2003), number 2835 in Lecture Notes in Computer Science, pages 197--214, Springer Verlag.
- [13]. Lavrac, N. and Flach P., (2000), "*An extended transformation approach to Inductive Logic Programming*", University publication, Department of Computer science, University of Bristol
- [14]. Lavrac N., Zelezny F., Flach P., (2002): "*RSD: Relational Subgroup Discovery through First-order Feature Construction*", Proceedings of the 12th International Conference on Inductive Logic Programming (ILP'02), Springer-Verlag, ISBN 3-540-00567-6
- [15]. Nattee, C., Sinthupinyo, S., Numao, M., Okada, T., (2005), "*Inductive Logic Programming for Structure-Activity Relationship Studies on Large Scale Data*", SAINT Workshops 2005: 332-335
- [16]. Inokuchi, A., Washio, T., and Motoda, H., (2003), "*Complete mining of frequent patterns from graphs*" Mining graph data, Machine Learning, 50:321-354
- [17]. Fischer, I. and Meinel, T., (2004), "*Graph based molecular data mining - an overview*", IEEE SMC 2004 Conference Proceedings, pages 4578--4582
- [18]. 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", Journal Med. Chem. 34:786-797
- [19]. Michie, D., Muggleton, S., Page, D., and Srinivasan, A., (1994), "*To the international computing community: A new East-West challenge*" Oxford University Computing laboratory, Oxford, UK, URL:

<ftp://ftp.comlab.ox.ac.uk/pub/Packages/ILP/trains.tar.Z>

- [20]. US National Toxicology program,
<http://ntp.niehs.nih.gov/index.cfm?objectid=32BA9724-F1F6-975E-7FCE50709CB4C932>
- [21]. Breiman, L., (2001), "*Random Forests*",
Machine Learning 45(1): 5-32 (2001)
- [22]. Ian H. Witten and Eibe Frank (2005) "Data Mining: Practical machine learning tools and techniques", 2nd Edition, Morgan Kaufmann, San Francisco, 2005.