Using Predictive Clustering and Probabilistic Constraint Solving for Structural Predictions

Kurt Driessens KURT.DRIESSENS@CS.KULEUVEN.BE
Hendrik Blockeel HENDRIK.BLOCKEEL@CS.KULEUVEN.BE
Department of Computer Science, Katholieke Universiteit Leuven, Leuven, Belgium

Abstract

In this extended abstract we introduce a new idea on how to make predictions in a structured output space. We present the basic principles of the approach and propose a discussion in the SRL community on the overall field of structured predictions and the proposed approach.

1. Problem Introduction

Most relational or structural learning approaches, including the ones under the umbrella of SRL, focus on learning in a context where the input data are relational, but not the output data. When predictive models are learned, they typically represent functions $X \rightarrow Y$ where the elements of X are graphs (or something of equal complexity) and the elements of Y are symbols, numbers, or booleans. Through repeated application of such models one can predict structures to some extent, e.g. by predicting for all nodes of a graph whether there should be a link between these nodes, a graph structure can be predicted. But this typically only works if the set of nodes is given in advance and, in the case of link prediction for instance, couples of nodes can be used as inputs of the function.

Take for instance the simple problem of learning a function that maps any graphs onto its "double", that is, a graph consisting of two disjoint components each of which is isomorphic with the input graph. Or take the problem of learning to map any number n onto K_n, a clique of n nodes. There are learners that from examples (K_n, n) can learn a function mapping K_n onto n, but it is unclear how the inverse function could be learned.

The above are toy examples, but there are ample applications that exhibit this kind of input/output structure. We list a few for illustration:

Presented at the ICML Workshop on Open Problems in Statistical Relational Learning, Pittsburgh, PA, 2006.

Molecular Structure Prediction : Determining the exact structure of a molecule of a given compound is a difficult task. Using techniques such as mass-spectrometry, chemists try to predict these structures starting from data that describes the mass (or more precisely the mass-to-charge ratio) of ions formed by the molecules.

Single Ended Line Testing : Re-discovering the structure of buried copper-wire pairs is becoming a bottle neck in the distribution of newer X-DSL technologies. The exact topologies of telephone lines have often been lost through history, but can be a large influence on the bit-rate that can be obtained with X-DSL technology. Single ended line testing is used to determine this topology, without having to send an employee of the phone company to the customers premises. The input data is generated by transmitting a signal onto the line under consideration and listening to the echo's it causes. This signal is then sampled and translated into a large numerical array. The output is of course the topology of the buried phone line, including lengths and thickness of wires used in the different segments and possibly the existence of side lines, a.k.a. bridged taps.

Protein Folding : Also the well-known problem of protein folding falls into this category of problems. The input data in this problem consists of a sequence of amino acids, while the output is a combination of several levels of folding structures.

The problem of predicting structured values is not entirely new: it has been considered in case-based reasoning, evolutionary learning, etc. But to our knowledge it has not received much attention in SRL. Given the strong development of this field the last few years, the question arises whether and how the many recent achievements in SRL could be used to improve the state-of-the-art in predicting structured values, and what contributions it could make to problems such as the ones mentioned above.

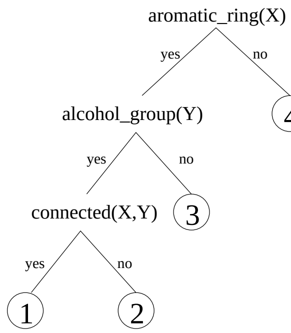

Figure 1. An example of a predictive clustering tree.

In the remainder of this paper we present one possible approach to structured prediction that we are currently considering. This is ongoing work and we welcome any ideas the SRL community may have on this approach or more generally the problem of structured value prediction.

2. Predictive Clustering Trees

Prediction with decision trees occurs in two phases: first an example is assigned to a leaf of the tree according to the tests occurring in the nodes of the tree. Then some prediction is made by that leaf, based on the examples covered by it (e.g. the majority class, or the mean of the target values in the leaf). In principle one could also switch the information used in these two phases: assign the instance to a leaf based on e.g. its overall similarity to the examples in that leaf, and then predict that the example fulfills the tests in the tree that lead to that leaf. The latter, non-conventional, method lies at the basis of our approach to structure prediction.

The partitioning made by a decision tree in the example space can be seen as a clustering of that example space. By changing the splitting criteria used during the construction of the decision tree to the standard criteria as used when building clusters (minimizing intra-cluster-variance and maximizing inter-cluster-variance) one can build a clustering that uses observable properties to distinguish between clusters (Blockeel et al., 1998). Note that this requires a similarity measure to be defined on the learning examples.

Once a predictive clustering tree (PCT) is built, it can be used for multiple target prediction in the case of propositional trees or to derive multiple relational or structural constraints in case of a first-order decision tree. An example of a (relational) predictive clustering tree is shown in Figure 1. By matching a new example — with an unknown structure — with the correct cluster represented by a leaf of the PCT, a number of constraints on the structure of the new example can be derived. For example, cluster 2 in Figure 1 “requires” its examples to have both an aromatic ring and an alcohol group present, but they should not be connected.

3. Structure Prediction Using Predictive Clustering Forests

The constraints dictated by a single clustering tree will not be sufficient to derive a complete structure for the new example. If they did, the tree would not be a generalization of the data. By building a forest of PCT’s, through the use of standard forest-construction algorithms such as sampling of the data or introducing variations of the language used to build the tree, one can derive extra constraints which could limit the number of allowed structures.

It is however possible that these constraints contradict each other and a way needs to be designed to merge these in a well founded way. Based on the used similarity measure, one could, for each cluster of each tree, derive a probability for the new example to belong to that cluster. This will lead to a set of first-order constraints, each of them tagged with a probability estimate that the example’s structure should satisfy the constraint. This set of constraints, some of which will contradict each other, should allow a derivation of at the least a backbone of the structure of the example under consideration. This will involve some sort of search and also a kind of inference similar to that made by Markov Logic Networks (Richardson & Domingos, 2006), but we have not investigated this in detail yet.

Proposal for Discussion

We are soliciting comments from the SRL community on both the specific level of this approach (do people think the overall approach is interesting; any ideas on solving the probabilistic constraint problem) as well as the more general level of structure prediction: to what extent can SRL contribute to this field, are there concrete ideas on how to use other SRL techniques in this domain?

References

Blockeel, H., De Raedt, L., & Ramon, J. (1998). Top-down induction of clustering trees. *Proceedings of the 15th International Conference on Machine Learning* (pp. 55–63).

Richardson, M., & Domingos, P. (2006). Markov logic networks. *Machine Learning*, *61*, 107–136.