Evolutionary Higher-Order Concept Learning

Claire J. Kennedy
Department of Computer Science
University of Bristol
Bristol BS8 1UB, U.K.
kennedy@cs.bris.ac.uk

ABSTRACT

Current concept learners are limited in their applicability as they generally rely on comparatively poor knowledge representation facilities (e.g. attribute value pairs, flattened horn clauses). The work carried out in support of my thesis has involved extending concept learning to a higher order setting by developing a novel representation based on closed Escher terms for highly structured data. The added expressiveness offered by the proposed representation results in an explosion of the search space, which is compounded by the increased complexity of its structure. This paper describes an investigation into the use of genetic programming techniques to allow the exploitation of higher-order features during the induction of structured concept descriptions.

1 Introduction

Escher (Lloyd 1995) is a new programming language which combines optimal features of both functional and logic programming languages to provide a facility for learning in a higher-order context. While induction in this environment benefits from exploiting Escher's strongly typed computational model and richer representation language, the greater expressiveness of the concept descriptions results in an explosion of the the search space. In addition, higher-order concept learning still lacks a counterpart to the clean refinement methods and theories of the propositional and first-order settings; subsequently the search for a solution to the problem can become intractable.

It is proposed that the problem of induction in the Escher context may be overcome by constructing the concept definition in an evolutionary manner. Under this framework the preferred representation of the examples is a single closed term for each individual. This provides a compact and self-contained description. During learning it is necessary to extract parts of the individual terms. This is achieved by using various selector functions (Flach *et al.* 1998) which are capable of extracting the components of tuples, sets and lists. Consequently the alphabet used by the genetic programming algorithm to construct the classifying descriptions includes the selector functions, as well as the components themselves and some comparison functions. By treating an example in this manner, it is possible to induce descriptions of concepts of arbitrary complexity. During learning items are required to be combined in a manner that involves checking type compatibility due to the non-closure of the functions inherent in the applications. Fitness is evaluated by predictive accuracy on a set of test data.

Preliminary experiments with the simple playing or not playing tennis problem, described by Mitchell (Mitchell 1997) are currently being conducted as a proof of concept. Further experiments will involve the finite element Mesh design problem, predicting mutagenic compounds and Michalski's train problem (Muggleton and Page 1994).

2 Conclusions

The research carried out addresses concept learning in an original way. Escher overcomes the limitations imposed by the propositional and first-order settings. The explotion of the search space under Escher's higher-order setting is alleviated by the evolutionary approach.

References

Flach, P., C. Giraud-Carrier and J.W. Lloyd (1998). Strongly typed inductive concept learning. In: *Proceedings of the International Conference on Inductive Logic Programming, ILP'98*.

Lloyd, J.W. (1995). Declarative programming in escher. Technical Report CSTR-95-013. Department of Computer Science, University of Bristol. Available at `http://www.cs.bris.ac.uk/`.

Mitchell, T.M. (1997). *Machine Learning*. McGraw-Hill.

Muggleton, S. and C.D. Page (1994). Beyond first-order learning: Inductive learning with higher order logic. Technical Report PRG-TR-13-94. Oxford University Computing Laboratory.