Constraint Acquisition and Machine Learning

Steve Prestwich

Department of Computer Science
University College, Cork, Ireland

Abstract. This is a proposal for a 10–30 minute presentation on new connections between Constraint Acquisition (the task of learning constraint programs from examples) and various types of Machine Learning: classification, statistical learning and data mining. I am interested in possible collaborations or projects.

For more than 20 years Constraint Acquisition (CA) has used forms of Machine Learning (ML), mainly Inductive Logic Programming and Version Space Learning. Are other fruitful connections between ML and CA possible? Our recent work shows that this is an area with rich possibilities:

- **Classification.** In CA we are given a *bias*: a set of candidate constraints to be learnt or discarded. This can be viewed as binary classification, and the ML literature describes a great variety of classifiers. New CA methods might be derivable from some, an idea we demonstrated by deriving the new BAYESACQ method from a naive Bayes classifier. It was shown to be considerably faster than existing methods on a set of benchmarks. Other classifiers have unique properties that might lead to new types of CA method.
- **Statistical learning.** An even faster method called SEQACQ was derived from a sequential analysis algorithm called the sequential probability ratio test. As an example of its speed, it correctly learned 50 SAT clauses from a bias of over a billion candidates in just over a minute.
- **Data mining.** A drawback of most CA methods is that they suffer from a *data collection bottleneck*: they use forms of supervised learning and require human preparation of large datasets of examples labelled *solution* or *non-solution*. Some methods only require solution examples, but these are also time-consuming to collect. However, the MINEACQ method uses techniques inspired by Data Mining to learn constraints from *unlabelled* datasets containing solutions and/or non-solutions. In many applications unlabelled data is plentiful and can be (for example) scraped from the Web.
- **Symbolic ML.** In CA we learn constraint programs from examples, but an alternative is possible: extrapolating large constraint programs from small ones. This has been done before by non-ML methods, and in the Symbolic ML field of Inductive Logic Programming. XACQ is a recent method using a *symbolic classifier* which learns a mathematical formula for classification.

I am interested in exchanging ideas on future developments in this area.