Distributed Learning: An Agent-Based Approach to Data-Mining

Winton Davies
Department of Computing Science
King's College
University of Aberdeen
Scotland, AB9 2UE
wdavies@csd.aberdeen.ac.uk

Peter Edwards
Department of Computing Science
King's College
University of Aberdeen
Scotland, AB9 2UE
pedwards@csd.aberdeen.ac.uk

1 Introduction

This extended abstract summarises our current research which spans the fields of knowledge discovery and software agents. Knowledge discovery (or data-mining) is concerned with extracting knowledge from databases and/or knowledge bases (Piatetsky-Shapiro & Frawley, 1991) using machine learning techniques. Traditionally, data-mining systems are designed to work on a single dataset. However, with the growth of networks, data is increasingly dispersed over many machines in many different geographical locations. Also, whilst most practical data-mining algorithms operate over propositional representations, we are using first order learning algorithms (Muggleton, 1992). This is to enable us to explore the aspects of knowledge integration and theory refinement which do not appear in propositional systems. However, this paper only presents preliminary, propositional results which do not reflect the more complex aspects associated with first order learning.

Software agents (Levy, Sagiv & Srivastava, 1994; Oates, Prasad & Lesser, 1994) are one response to the problem of using the vast amounts of information stored on networked systems. There are many types of software agent (Wooldridge & Jennings, 1995); however, agents are typically thought of as being 'intelligent' programs which have some degree of autonomy. We intend to design an open, flexible data-mining agent. A group of these agents will be able to co-operate to discover knowledge from distributed sources.

The issues involved have been covered to some extent by researchers concerned with multi-agent machine learning. A number of multi-agent learning systems have been built. These include MALE (Sian, 1991), ANIMALS (Edwards & Davies, 1993) and ILS (Silver, 1990). The first of these was a homogeneous, blackboard-based system, while the others used a distributed problem-solving approach. All three used propositional learning methods. A number of researchers have addressed the problem of reconciling different views of similar data; these include Gams (1989), Brazdil & Torgo (1990), and Svatek (1995).

Our high-level model is as follows. One or more agents per network node are responsible for examining and analysing a local data source. In addition, an agent may query a knowledge source for existing knowledge (such as rules or predicate definitions). The agents communicate with each other during the discovery process. This allows the agents to integrate the new knowledge they produce into a globally coherent theory. In addition, a supervisory agent, responsible for co-ordinating the discovery agents may exist. A graphical interface allows the user to assign agents to data sources, and to allocate high level discovery goals. It allows the user to critique new knowledge discovered by the agents, and to direct the agents to new discovery goals, including ones that might make use of the newly discovered knowledge.

As far as possible, our intention is to base our work on the integration of existing technologies. This is in order to concentrate on the core issues of how agents can resolve different views of the world.

We plan to use agents based on Agent Oriented Programming (AOP) (Shoham, 1990), and techniques developed as part of the Knowledge Sharing Effort (Patil et al., 1992). To support this we have developed an agent programming language, Agent-K, which is detailed in Davies & Edwards (1994). This provides Agent-0 with the ability to handle KQML messages. In order to allow interaction between learning agents, we intend to define extensions to the communication performatives of KQML and to define a simple machine learning ontology. These additions to the KSE will allow agents to communicate about the fundamental activities involved in machine learning. This is in order to support distributed learning as well as interactions between different learning strategies.

We have examined a number of recent ILP algorithms many of which allow the inclusion of background knowledge expressed in first order predicate calculus. A knowledge base could thus be used to supply existing domain knowledge to an ILP-based data-mining agent. We have chosen to use the information-gain based FOCL (Pazzani & Kibler, 1992) because it has an efficient inductive algorithm as well as a built-in theory revision mechanism. However, we also wish to provide our discovery system with an unsupervised algorithm in order

to explore interactions between supervised and unsupervised learning agents. To this end we are developing our own relational clustering algorithm which will permit unsupervised first order learning.

Our approach is in some respects similar to that of KBG (Bisson, 1992) which also performs conceptual clustering over a first order logic representation. KBG finds similarities between the entities found in relations, whereas our algorithm attempts to find similarities between the relations that hold between entities. Our algorithm combines DINUS (Lavrac & Dzeroski, 1994) and COBWEB (Fisher, 1987). DINUS converts a first order learning problem into a propositional one. COBWEB then finds clusters in the propositional representation. A final step is to describe the clusters in first order form. This algorithm is still in the preliminary phase of investigation.

The remainder of this paper concentrates on the different ways that agents in a distributed learning system can interact. We briefly describe the different approaches to distributed learning and the related issues of theory revision and knowledge integration. We then conclude with a report on our preliminary experiments in this area. This work focuses on a distributed learning system composed of FOCL agents.

2 Distributed Learning

There are three ways learning can occur when data is distributed. These relate to when agents communicate with respect to the learning process:

- The first approach gathers the data into one place. The use of distributed database management systems to provide a single set of data to an algorithm is an example of this (Simoudis, 1994). The problem with such an approach is that it does not make efficient use of the resources usually associated with distributed computer networks.
- The second approach is for agents to exchange information whilst learning on local data. This is the approach taken by Sian (1991). No revision or integration is needed, as the agents are effectively working as a single, tightly coupled, algorithm over the entire data. This restricts the agents to using learning algorithms that have been specially modified to work in this way. Thus the main disadvantage with this approach is that it does not allow the use of 'off-the-shelf' learning algorithms.
- The third approach is for the agents to learn locally, and then to share their results, which are then refined and integrated by other agents in light of their own data and knowledge. This model permits the use of standard algorithms, and also allows inter-operation between different algorithms. Brazdil & Torgo (1990), Svatek (1995) and Provost & Hennessy (1994) have all taken this approach. The main problem here is how to integrate the local results.

We are adopting the latter approach, as it provides distributed processing together with flexibility in deploying 'off-the-shelf' algorithms. The following section describes the relationship between theory revision, knowledge integration and incremental learning. We will then describe an empirical comparison of three different approaches to distributed learning based on theory revision, knowledge integration and a combination of theory revision and knowledge integration.

2.1 Theory Revision, Knowledge Integration & Incremental Learning

Theory refinement and knowledge integration are related techniques. Theory refinement involves revising a theory with respect to new training examples. Knowledge integration involves combining two theories into a single unified theory. Although related, the two processes differ in certain respects. Theory revision generally involves the modification of individual rules; knowledge integration generally involves removing redundancy between different sets of agents' rules, then ordering the remaining rules.

Distributed learning can also be cast as an incremental learning problem. Mooney (1991), demonstrated that any theory revision system should be capable of incremental revision. This is achieved by feeding back a partial theory, and then revising that theory with respect to a single example. In a distributed learning system, the problem is similar; an agent learns a local theory, then has it revised by another agent with respect to that agent's examples. This favours incremental learning algorithms and those algorithms with a theory revision component for use in distributed learning.

2.2 An Empirical Evaluation of Distributed Learning Techniques

In order to compare the efficiency of techniques for distributed learning, we have conducted a series of experiments based on FOCL. The 1984 Congressional Voting Records dataset (435 examples) was used. We used both the inductive and theory revision components of FOCL. The simple knowledge integration component makes use of a simple accuracy measure. We count the number of correct positive and negative examples covered by each theory. Every theory is measured against all the training data. The theory with the highest score is selected. We have evaluated four different distributed learning methods:

1 *No Distribution.* A conventional, non-distributed approach. All the available training examples are provided to FOCL.

2 *Incremental Theory Revision.* An agent learns a local theory from its share of available training examples, and then passes this theory to the next agent, and so on. Five agents were used. In each case the available training examples are split equally between the agents.

3 *Simple Knowledge Integration.* Each of five agents learns a local theory. The five resulting theories are tested against all the training examples, the best theory is selected and is then compared with the test set.

4 *Theory Revision and Simple Knowledge Integration.* As in the previous example, each of the five local agents learns a local theory. Every agent then receives all the other agents' theories. Each agent revises these four theories to fit its local data. As in the previous method, the best theory is found by testing against the entire training set.

For each of these methods, 10 test runs were conducted, and the results averaged. The training set was a random selection of 300 examples, leaving 135 for testing. For each run the current training set was split equally between the five agents (except for Method 1).

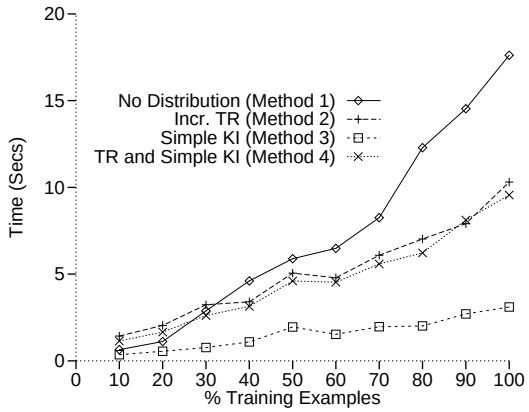

Figure 1: Time to Learn for Different Approaches to Distributed Learning - Congressional Voting Records

The accuracy results showed no statistically significant differences between the four approaches (and thus are not shown here). However, these are preliminary investigations, and as we have only used one dataset, we cannot make general statements about the comparative learning accuracy of these approaches.

The timing results (see Figure 1) do appear to be statistically significant. There is no significant difference between either theory revision approach. However, they do appear to be slightly faster than the non-distributed approach, but only when dealing with a larger proportion of examples. Again it must be noted that this is a test using one dataset only.

Figure 1 does clearly demonstrate that for this dataset simple knowledge integration is the best method for implementing distributed learning. This confirms the findings of Provost and Hennessy (1994) and Brazdil and Torgo (1990).

Our preliminary results do appear show that it is possible to distribute learning efficiently. These experiments are of a restricted nature. For example, we have only used one dataset, and that is effectively propositional in form. In particular we have not been able to test our belief that theory revision is necessary if agents have data that is specific to a location. This will cause distinct local theories to be learnt. A knowledge integration approach would not then be able to pick a correct global theory from the local ones. Our next task is to address these limitations in our experiments.

3 Summary

This paper described some of our work to date on an agent-based approach to distributed knowledge discovery. Our long term goal is that agent-based knowledge discovery will allow us to maximise the usage of distributed computing resources, and minimise the network traffic, as well as facilitate the easy integration and use of multiple learning algorithms.

Acknowledgements

Support for this work is provided through a UK Engineering & Physical Sciences Research Council (EPSRC) studentship. We would also like to thank Mike Pazzani for making FOCL available, and for providing invaluable technical guidance.

References

P. Brazdil & L. Torgo, Knowledge Acquisition via Knowledge Integration, In Current Trends in Artificial Intelligence, B. Wielinga et al. (Eds.), IOS Press, Amsterdam, 1990.

G. Bisson, Conceptual Clustering in a First Order Logic Representation, in Proceedings of Tenth European Conference on Artificial Intelligence (ECAI92), B. Neumann (Ed.), Wiley, 1992, 459-462.

W. Davies & P. Edwards, Agent-K: An Integration of AOP and KQML, in Proceedings of the CIKM'94 Intelligent Information Agents Workshop, Y. Labrou & T. Finin (Eds.), 1994.

P. Edwards & W. Davies, A Heterogeneous Multi-Agent Learning System, in Proceedings of the Special Interest Group on Co-operating Knowledge Based Systems, S. M. Deen (Ed.), University of Keele, 1993, 163-184.

T. Finin et al., DRAFT Specification of the KQML Agent Communication Language, 1993, unpublished draft.

D. Fisher, Knowledge Acquisition via Incremental Conceptual Clustering, Machine Learning, 2, 1987, 139-172.

M. Gams, New Measurements Highlight the Importance of Redundant Knowledge, in Proceedings of Fourth European Working Session on Machine Learning (EWSL89), K. Morik (Ed.), 1989, 71-79.

N. Lavrac & S. Dzeroski, Inductive Logic Programming: Techniques and Applications, Ellis Horwood, 1994, 81-123.

A. Y. Levy, Y. Sagiv & D. Srivastava, Towards Efficient Information Gathering Agents, in Papers from the AAAI Spring Symposium on Software Agents (Technical Report SS-94-03), AAAI Press, 1994, 64-70.

R. J. Mooney, Batch versus Incremental Theory Refinement, AAAI Spring Symposium, 1992.

S. Muggleton, Inductive Logic Programming, Academic Press, 1992.

T. Oates, M. V. N. Prasad & V. R. Lesser, Co-operative Information Gathering: A Distributed Problem Solving Approach, Technical Report Computer Science 94-66, University of Massachusetts, 1994.

R.S. Patil et al., The DARPA Knowledge Sharing Effort: Progress Report, in Proceedings of KA92 - The Annual International Conference on Knowledge Acquisition, Cambridge, MA, 1992.

M. Pazzani & D. Kibler, The Utility of Knowledge in Inductive Learning, Machine Learning, 9, 1992, 57-94.

G. Piatetsky-Shapiro & W. J. Frawley, Knowledge Discovery in Databases, MIT Press, 1991.

F. J. Provost, & D. Hennessy, Distributed Machine Learning: Scaling up with Coarse-Grained Parallelism, in Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology, Menlo Park, AAAI Press, 1994.

Y. Shoham, Agent-Oriented Programming, Technical Report STAN-CS-90-1335, Stanford University, 1990.

E. Simoudis, Personal Communication, 1994.

S. Sian, Extending Learning to Multiple Agents: Issues and a Model for Multi-Agent Machine Learning (MA-ML), in Proceedings of the European Working Session on Learning (EWSL91), Y. Kodratroff (Ed.), Springer-Verlag, 1991, 458-472.

B. Silver et al. ILS: A Framework for Multi-Paradigmatic Learning, in Proceedings of the Seventh International Conference on Machine Learning (ML90), B. Porter & R. Mooney (Eds.), Morgan Kaufmann, 1990.

V. Svatek, Integration of Rules from Expert and Rules Discovered in Data, Prague University, Unpublished Draft, 1995.

M. Wooldridge & N. R. Jennings, Intelligent Agents: Theory and Practice, Knowledge Engineering Review, 10(2), 1995.