Collaborative Agent-Based Learning with Limited Data Exchange

(Extended Abstract)

Xavier Rafael Palou Michael Rovatsos
School of Informatics
The University of Edinburgh
Edinburgh EH8 9AB, United Kingdom
{s0680898,mrovatso}@inf.ed.ac.uk

ABSTRACT

We describe a collaborative agent-based learning model suitable for environments with limited data exchange and provide an overview of its empirical evaluation.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent Systems*

General Terms

Algorithms, Design, Performance

Keywords

Distributed data mining, collaborative learning

1. INTRODUCTION

In many application domains, distributed and open environments are increasingly becoming the norm rather than the exception, and distributed data mining is a prolific active research area where new, efficient and effective techniques are being developed and continuously improved.

More specifically, for the distributed classification problem, one of the most widely used approaches is to gather all the data from different local nodes in a central repository and then apply traditional data mining techniques to the entire data. However, in many domains the exchange and sharing of data is not allowed or feasible, because local data may be too costly to communicate, or it may not be possible to reveal all information for security reasons, due to legal restrictions or to avoid loss of competitive advantage. A domain example where these types of restrictions apply is the distributed medical care environment, where patient information is used and sharing this data with other centers is legally restricted or prohibited (e.g. in brain tumour classification [1]).

Learning predictive models in distributed environments has been recently addressed by using multiagent systems (MAS) techniques in combination with data mining (DM) techniques since MAS can contribute a number of crucial capabilities that may be useful for solving DDM problems. While different approaches exist in the literature, a recent Multiagent Learning Framework (MALEF) [4] attempts to ensure adherence to agency aspects like collaboration between heterogeneous classifiers, decentralised learning control, and the autonomy of self-directed learning processes. This abstract framework uses communication and collaboration among the different local classification learning agents. The learning agents perform a series of consecutive learning steps using two functions: classifier training and measurement of the resulting classifier quality. They may additionally perform integration operations using different parts of the knowledge received from other learning agents. Also, a number of very generic categories of integration operations are proposed in the framework.

Cite as: Collaborative Agent-Based Learning with Limited Data Exchange, (Extended Abstract), Xavier Rafael Palou, Michael Rovatsos, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 1191–1192

2. MALEF & LIMITED DATA EXCHANGE

Building on the abstract MALEF learning framework we propose a concrete, workable instantiation for classification domains with limited data sharing. We maintain the iterative notion of learning steps described in the framework, but re-interpret the learning step as a completion of four different stages(fig.1):

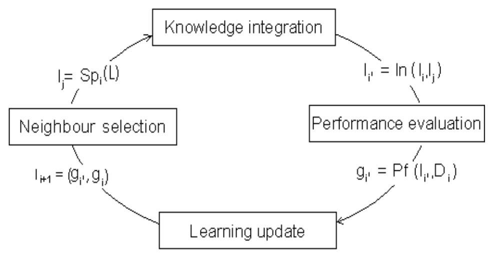

Figure 1: Learning step of collaborative model for the learner agent l_i

Initially a learner (l_i) performs a *neighbour selection* operation, $Sp_i(L)$, in which another learner (l_j) is selected (e.g. randomly or using a greedy in accuracy search strategy) among all learners (L). Then a *knowledge integration* step is performed, $ln_i(l_i, lj)$, using some part of the learning information of the current (l_i) with the selected learner (l_j), thus obtaining a modified learning process ($l_{i'}$). *Performance evaluation*, $g_{i'} = Pf(l_{i'}, D_i)$, of the integrated knowledge is performed using a local data set (D_i). Finally, the learner is updated l_{i+1} if the performance has increased $g_{i'}$ compared to its previous value g_i.

Focusing on knowledge integration between two learners, we proposed a number of operations suitable for heterogeneous scenarios (i.e. societies in which agents use heterogeneous learning algorithms). For these scenarios three different types of learning knowledge to be integrated were considered: training data, classification outputs, and hypothesis.

Regarding the integration of training data, we proposed an operation based on merging small portions of local data sets. As far as output integration is concerned, we analysed *voting* and *the arithmetic merging of posterior estimated distributions*[3]. Finally, for the integration of hypothesis, we proposed a new method for merging classifiers represented as trees. This last type of operation was of particular interest since the predictive models are not viewed as black boxes to be combined depending on the quality of some output combination operation. Instead, hypothesis merging aims to modify the integrator learner's model producing a new, richer model than that available before collaboration.

3. TREE MERGING TECHNIQUE

We have proposed a method for unifying heterogeneous hypotheses based on tree representations. Although this not cover all possible classifier representations, it is generic enough to allow some degree of heterogeneity among learners. We chose the tree classifiers because they are easy to use computationally and offer good understandability and readability for humans.

Our method attempts to increase the knowledge of h_i by integrating those branches (rules) from h_j which h_i does not consider or where h_i fails. Bearing this idea in mind, and assuming that each classifier belongs to different learners (l_i/l_j), the following algorithm outlines the tree merging technique:

1. Send h_i from l_i to l_j
2. In l_j classify using D_j (training set of l_j) and comparing the results obtained by h_i and h_j
3. For each instance c of D_j, if $h_j(c)$ predicts the output correctly and $h_i(c)$ fails then select branches from h_j which predict correctly
4. Add these branches to h_i
5. Send h_i to l_i

This method raises a number of issues. Firstly, the possibility of activating two or more rules (tree branches) for the classification of an instance (contradictions/redundancies in verdicts). This was solved by means of the selection of that class whose rules were more frequently predicted, settled via voting (since this is most commonly used in distributed/multi-strategy machine learning for resolving conflicts). Finally, we dealt with the redundancy of branches inside the merged classifier, by filtering repeatedly occurring branches and conditions absorbed by another condition inside a branch, retaining the more general rule. For example, a $a < 1$ condition would be absorbed by $a < 2$ condition.

4. EXPERIMENTAL EVALUATION

Our collaborative learning system was evaluated with different learning experiments using the methods proposed for each of the stages described above. These experiments were conducted using datasets from the UCI repository[2].

A summary of the evaluation is shown in figure 2, where the best collaborative learning configuration (tree merging method with greedy accuracy-based neighbour selection strategy) is compared to two alternative learning setups: *centralised* and *distributed isolated* learning. These simulate the best and worst possible classification accuracy, i.e. collaborating agents should not be able to do any better than a centralised agent (in possession of all training data) or any worse than isolated agents which have no means of improving their initial, locally obtained result. The impressive results of collaborative learning (e.g. 20% better than isolated learning for the Letters dataset) is caused by the tree merging operation which enables learners always to identify the useful parts of the selected peer learner's model and to modify their own current models accordingly.

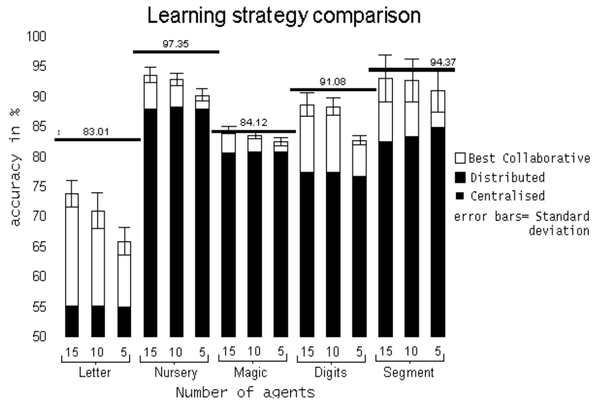

Figure 2: Learning strategies comparison in heterogeneous scenario for different agent configurations

5. CONCLUSIONS

Although more sophisticated search and integration techniques are envisaged for our collaborative model, the empirical investigation indicates the significant potential of our method. For most of the configurations that were investigated, the individual learners improve their initial classification accuracy and performance approaches that of a (hypothetical) centralised solution. This is more effective with a suggested tree-based model merging operation which allows for merging hypothesis across heterogeneous learners, which, to our knowledge, has not been attempted before.

6. REFERENCES

[1] Healthagents. http://www.healthagents.net.
[2] A. Asuncion and D. Newman. UCI machine learning repository, 2007.
[3] L. I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.
[4] J. Tozicka, M. Rovatsos, and M. Pechoucek. A framework for agent-based distributed machine learning and data mining. In *Proc. 6th Intl. joint conf. on Autonomous agents and multiagent sys.*, pages 1–8, New York, NY, USA, 2007. ACM.