CENTRALIZED MODEL EVALUATION FOR COLLABORATIVE DATA MINING

Hendrik Blockeel[1], Steve Moyle[2]

[1] K.U.Leuven, Department of Computer Science
Celestijnenlaan 200A, B-3001 Leuven, Belgium
hendrik.blockeel@cs.kuleuven.ac.be
[2] Oxford University Computing Laboratory
Wolfson Building, Parks Road, Oxford, OX1 3QD, UK
steve.moyle@comlab.ox.ac.uk

ABSTRACT

Collaborative data mining refers to a data mining setting where different groups are geographically dispersed but work together on the same problem in a collaborative way. Such a setting requires adequate software support. In this paper we describe an experiment with a simple implementation of such a collaborative data mining environment. The experiment brings to light several problems, one of which is related to model evaluation. We discuss several possible solutions. This discussion can contribute to a better understanding of how collaborative data mining is best organized.

1 INTRODUCTION

Many different approaches to data mining exist. They have arisen from different communities (databases, statistics, machine learning, …). Thus, data mining nowadays is performed by people with highly different backgrounds, each of whom have their preferred techniques. Very few people are experts in all these domains, so to get the most out of a data mining process, ideally multiple experts should work together on the same data mining task. As even experts in a single of these domains may be relatively rare, such a group of experts may not be available in a single location.

These observations provide motivation for the development of a methodology for *collaborative data mining*. Our point of departure is that groups with different expertise who are geographically distributed should be able to collaborate on a certain problem, thus jointly achieving better results than any of them could individually.

Having different experts collaborate on the same task requires some supporting environment. In the context of the European SolEuNet project, ideas have evolved about what functionality such an environment should offer, resulting in a proposal for a collaborative data mining methodology and supporting system called RAMSYS [4] and an implementation using the groupware system Zeno [3].

In this paper we report on a collaborative data mining experiment in which the proposed RAMSYS methodology and its implementation on Zeno were used. Several lessons have been learnt from this experiment regarding the methodology itself as well as its current implementation. An important one relates to model evaluation. We propose an improvement to RAMSYS based on this result.

The paper is structured as follows. In Section 2 we discuss RAMSYS and Zeno. In Section 3 we describe our collaborative data mining experiment and the problems encountered, and in Section 4 we propose and compare possible solution. Section 5 concludes.

2 COLLABORATIVE DATA MINING, RAMSYS AND ZENO

Data mining is about solving problems by analysing data already present in databases [9]. Problem solving, in general, can be codified and a procedure or methodology can be devised. For data mining, one such methodology is the CRoss Industrial Standard Process for Data Mining, CRISP-DM [2]. CRISP-DM reduces the data mining problem into the six inter-related phases of 1) *Business Understanding*; 2) *Data Understanding*; 3) *Data Preparation*; 4) *Modelling*; 5) *Evaluation*; and 6) *Deployment*. These phases, although presented in a linear manner, have many cycles and feedback loops connecting the phases. Often, effort expended in one phase highlights the need for further work in a prior, previously considered complete, phase.

Comment:

The *RAMSYS* methodology is an extension to the CRISP-DM methodology for distributed teams who collaborate in a data mining project. The aim is to combine the great range of expertise available in the data miners to effect more valuable solutions to the data mining problem. The RAMSYS methodology attempts to achieve the combination of a problem solving methodology, knowledge sharing, and ease of communication. It is guided by the following principles [4]: it should enable **light management**, it should allow collaborators to **start and stop any time** and leave them **problem solving freedom**, it should provide efficient **knowledge sharing** and **security**. So far, the RAMSYS efforts have focussed on supporting the *Data Preparation* and *Modelling* phase in a remote-collaborative setting; here we focus on the *Evaluation* phase.

Some of the basic requirements of the RAMSYS methodology is the emphasizing and availability of the *current best understanding* [7] of the data mining problem. This has been implemented using the academic groupware platform Zeno [3], by providing *coordination*, *collaboration*, *communication*, and *awareness*. The provision of these features are achieved by utilizing (new) features in Zeno including *task management*, *resource management*, and *discussion sections*.

The RAMSYS methodology has been trialed (in part or in full) on several data mining projects, one of which is the SPA project described in the next section.

3 AN EXPERIMENT WITH COLLABORATIVE DATA MINING: THE SPA PROBLEM

3.1 The SPA problem

The “SPA problem” was offered to the SolEuNet consortium by a health farm. The health farm has a number of facilities that can be used by its visitors. More specifically, upon their arrival visitors are prescribed certain procedures to follow during their stay at the spa, as well as a schedule for them. The number of people that can simultaneously make use of certain facilities is limited. Thus the spa is faced with a scheduling task: given the procedures that newly arrived visitors need to follow and the limited capacity of certain facilities, create a suitable schedule.

In practice there is insufficient information to solve this scheduling task for the following reason. Visitors stay for several weeks and a schedule for their whole period of stay is made, but during their stay new visitors will arrive. While some information about these new visitors is available in advance (such as time of arrival, age, sex, …) the procedures they need to follow will be known only at the time of their arrival. The best one can do is to estimate the demand for the facilities for the near future, and use these estimates for producing schedules for the current patients. It is here that data mining comes in: by mining a database of previous visitors and trying to link properties of these visitors to the procedures they followed, predictive models could be built that estimate the demand for certain facilities based on known properties of future visitors.

Thus the data mining task can succinctly be described as follows: given a set of visitor descriptions that will arrive during a certain week, estimate how many of these visitors will need to follow each of some 40 available procedures.

3.2 The collaborative data mining process

Four groups (with 2 to 4 people each) worked on this project: CTU (Czech Technical University in Prague), BRI (University of Bristol), LIACC (University of Porto) and KUL (University of Leuven). CTU served as contact with the end user (the health farm).

Following the RAMSYS methodology implies following the CRISP-DM methodology, hence we here briefly describe the efforts according to the different phases. Phase 1 (business understanding) involved becoming familiar with the data mining problem, which was done by all groups separately. During Phase 2 (data understanding) several groups explored the data using visualisation techniques, association rule discovery, etc. and published their results on Zeno. In Phase 3 (data preparation) the main effort consisted of data transformations. As the original database consisted of multiple tables, this involved to some extent computation of aggregate functions. Data transformations were performed mainly using CTU’s SumatraTT tool [1].

In this paper we focus mainly on Phases 4 and 5: modelling and evaluation. Concerning modelling, a wide variety of approaches was taken by the different groups: support vector machines (BRI), neural nets (BRI, LIACC), linear regression (LIACC, KUL), instance based learning (LIACC, CTU), decision trees (LIACC, CTU, KUL), etc. Besides the different algorithms, approaches also differed in the version of the data set that was used (these versions resulting from different data transformations).

There is an intense feedback from 5 to 4: based on model evaluation, data miners wish to change their model building approach and go through Phases 4 and 5 once more. In the collaborative setting, the feedback should not remain within one group but flow to all groups for which it is relevant.

3.3 Evaluation of the collaborative data mining process

Our evaluation of this collaborative data mining experiment is ambiguous. The end-user found the results interesting and useful [6]. The bad news is that the added value of collaboration of different groups on this task was much smaller than hoped. The most notable collaboration was that the results of data transformations performed by one group were used for modelling by another group. This is in line with the kind of collaboration that RAMSYS promotes, but it is only a minimal version of it.

To achieve more intensive collaboration, several processes must be made more efficient. The CRISP-DM process is iterative, consisting of many steps and cycles. If collaboration is to happen at the level of a single step, it needs to happen very efficiently. To make this possible, **information exchange** should be made more efficient and **synchronization** should be improved. The information flow between groups was often hampered because documentation of results was too concise, too extensive, or even both (groups being flooded with information from colleagues without being able to find the most relevant information in there). As groups do not always have the right resources available at the right time, it may take a while before a group reacts to results from other groups.

The solutions to these problems are to be found both at the technical and management level (e.g. defining strict formats for exchanged documents so that relevant information is easier to identify).

Another process that needs to be made more efficient, is **comparative evaluation of models**. In order to compare different models, they must be evaluated according to the same criteria. The original RAMSYS methodology proposed to determine an evaluation criterion in advance so that each group can evaluate their models according to this criterion. The SPA experiment revealed several problems with this proposal. Firstly, it may be difficult to propose a good evaluation criterion in advance, and the preferred evaluation criteria may change over time, because insight in what are good and bad criteria typically develops during the knowledge discovery process. E.g., in the SPA experiment, visual data analysis revealed strong outliers. These turned out (after discussion with the end user) to be related to unavailability of certain procedures due to maintenance and were therefore irrelevant, but they strongly influenced certain error criteria and needed to be left out.

Secondly, one criterion may not be sufficient. Different criteria measure different properties, all of which may be relevant, see e.g. [5]. It is more realistic to talk of a set of criteria, instead of a single one. And finally, subtle differences in the computation of certain criteria, the data set from which they are computed, the partitioning used for cross-validation, … can make the comparison unreliable.

Due to the rate at which criteria may change, the number of criteria, and the care that must be taken when implementing them, it is unrealistic to expect that the different groups will continuously use the right criteria. An evaluation scheme is needed in which criteria can flexibly be changed or added and it is guaranteed that every group uses exactly the same version of a criterion, without too much overhead.

We propose *centralized model evaluation*. Instead of having all different groups evaluate their own models, one should have a kind of model evaluation server to which groups send the models they have produced, or the predictions produced by their models. When a group decides they are interested in some specific criterion, they should be able to add the criterion to the central evaluation server and immediately see the scores of all earlier submitted models on these criteria. In the next section we explore this direction further.

4 CENTRALIZED MODEL EVALUATION

In our proposal, data mining groups (“clients”) should send predictions or even the models themselves to a “model evaluation server”, which is responsible for the evaluation of the predictive model and automatically publishes the results.

Several levels of communication are possible. An inductive system typically has a number of parameters; for a given set of parameters values the system implements a function I: $2^{X\times C} \rightarrow (X \rightarrow C)$ that maps a dataset (a subset of the universe of labelled instances $X\times C$ with X the instance universe and C the set of target values) onto a function M (a predictive model) that in turn maps single instances onto some target value. One has the option to submit the inductive function I; the model M learnt from a given data set T; or a set of predictions for some data set S, $P = \{(e,M(e))|e \in S\}$. In all cases the server should be able to derive from the submission a score on one or more evaluation criteria, which we assume to be a function $c(M,P)$. The original RAMSYS procedure corresponds to a fourth option, communicating $s = c(M,P)$.

A schematic overview of these options (in reverse order compared to above) is given in Figure 1. It is assumed that I consists of a combination of a machine learning tool and parameter settings, so I is the result of tuning the tool with the parameters. Using I a model M is built from a training set, this M is used to predicted labels for a test set S, from these predictions a score s is computed using the evaluation criterion c. In the case of a cross-validation the process is more complicated but the same basic scheme is valid: different models M_i are then built from different training sets to produce one set of predictions P.

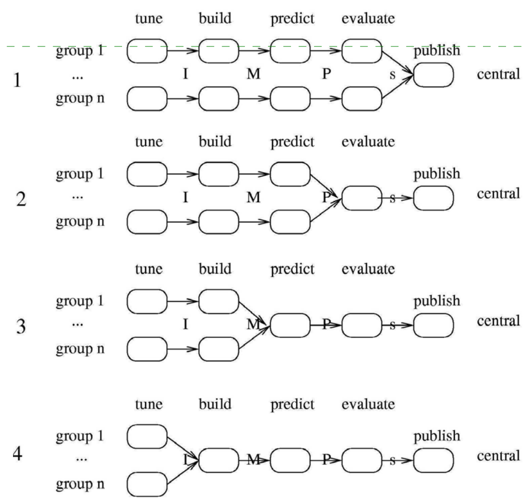

Figure 1: *Overview of different options for centralizing model evaluation in collaborative data mining*

	1	2	3	4
Language complexity	L	L	M	H
Communication cost	L	H	M	M
Result availability	L	H	H	H
Comparability	M	H	H	H
User overhead	H	M	M	L
Flexibility of evaluation	H	M	H	H

Table 1: *Characteristics of different options*.

Table 1 summarizes some characteristics of the four options. In the table H, M and L refer to High, Medium and Low respectively. *Language complexity* refers to the language that is needed for communication. Options 3 and

Comment:

4 impose the challenge of developing relatively complex languages and interpreters for them (e.g., when submitting a model *M* the server needs to be able to compute the predictions *M* makes on some test set). *Communication cost* is low when communicating just a score, high when communicating a (possibly large) set of predictions, and medium when communicating functions. *Result availability* refers to how fast the scores of different models for a new criterion are made available to everyone. It is low for Option 1 (groups need to implement the new criterion themselves); for other options new scores are automatically computed as soon as a single implementation of the new criterion is available. *Comparability* reflects the trust in the comparability of the results, which is higher when a single implementation is used. *User overhead* refers to the overhead for the data mining groups when some option is adopted. In Option 1 it is highest, in Option 4 lowest because the user need only submit *I* (induction system + parameters) and all testing is then done automatically. In Options 2 and 3 the user needs to implement e.g. cross-validation according to given folds. Finally *flexibility of evaluation* is lowest for Option 2 because here the criterion cannot involve the model itself (complexity, interpretability, …) but only its predictions.

Option 1 is the current mode of operation within SolEuNet. Option 2 provides significant advantages over Option 1 and is still easy to implement. Option 3 imposes the challenge that a good model description language and an interpreter for it need to be available. A reasonable choice for such a language would be PMML [8], which is already being proposed as a common language for representing models; it handles a reasonable variety of types of models and there exist visualisers for them. If PMML is going to be used anyway in a collaborative data mining system, an interpreter for PMML models would be sufficient to cater for a wide range of different model evaluation criteria.

Option 4 is the most powerful one but seems least feasible. There are different suboptions: (4a) all model building systems are translated into a single common language; (4b) the central model evaluation server has the necessary interpreters for the different languages in which inductive systems, data preprocessing systems, etc. are programmed; (4c) the server has its own versions of the inductive systems, and all that is actually submitted is an identifier of the system to be used and a list of parameters. Option 4c is quite feasible but has the disadvantage that only the systems and versions available at the server can be used. It is somewhat similar in spirit to the option taken in the European MetaL project (http://www.metal-kdd.org) on meta-learning.

In the short term, we believe the most realistic improvement to RAMSYS corresponds to Option 2, which is easy to implement and presents a significant improvement over the current mode of operation. In the longer run, assuming that PMML is general enough to describe any kind of model that could be submitted and that interpreters are available, it seems desirable to shift to Option 3.

Summarizing, centralized model evaluation *reduces workload*; *increases confidence* in comparisons between systems; *guarantees availability* of all criteria for all models; *reduces the time* needed to obtain scores on new criteria; and adds *flexibility* w.r.t. defining new criteria. All of these contribute to the added value that collaborative data mining can have over the non-collaborative approach.

5 CONCLUSIONS

Collaborative data mining, as promoted by and used within the SolEuNet project, is not a trivial enterprise. In order for it to work well, a highly tuned supporting environment is needed. This was recognized early on in the project, which led to the RAMSYS proposal.

An experiment with collaborative data mining, following the RAMSYS methodology as much as possible, indicated the need for more efficient and flexible model evaluation. Our answer to this is centralized model evaluation, of which we have presented and compared several versions. The conclusion is that significant improvements over the approach used for SPA can easily be obtained, while implementing an ideal system will need some more work.

Acknowledgements

This work was supported by the European IST project SolEuNet. H.B is a post-doctoral fellow of the Fund for Scientific Research of Flanders (FWO-Vlaanderen). The authors thank all collaborators on the SPA problem, Olga Stepankova and the anonymous reviewers for useful comments, and Lauryn v.o.s. for providing the SPA data set.

References

[1] Aubrecht, P. & Kouba, Z. (2001). Metadata driven data transformation. In *Proc. of the 5th World Multi-Conference on Systemics, Cybernetics and Informatics*.

[2] Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. (2000). CRISP-DM 1.0: step-by-step data mining guide.

[3] Gordon, T., Voß, A., Richter, G., Märker, O. (2001). Zeno: Groupware for discourses on the internet. *Künstliche Intelligenz* 15: 43-45.

[4] Jorge, A., Moyle, S., Voß, A., and Richter, G. (2002). Remote collaborative data mining through online knowledge sharing. *PRO-VE '02, 3rd IFIP Working Conference on Infrastructures for Virtual Enterprises*, Portugal.

[5] Köpf, C., Taylor, C., & Keller, J. (2001). Multi-criteria meta-learning in regression. In *Proc. of the ECML/PKDD-01 Workshop on Integration of Data Mining, Decision Support and Meta-Learning*.

[6] Stepankova, O., Lauryn, S., Aubrecht, P., Klema, J., Miksovsky, P., Novakova, L., & Palous, J. (2002). Data mining for resource allocation: a case study. *Intelligent Methods for Quality Improvement in Industrial Practice*, 1.

[7] Voß, A., Gärtner, T. & Moyle, S. (2001). Zeno for rapid collaboration in data mining projects. In *Proc. of the ECML/PKDD-01 Workshop on Integration of Data Mining, Decision Support and Meta-Learning*, pp. 43-54.

[8] Wettschereck, D., & Müller, S. (2001). Exchanging data mining models with the predictive modelling markup language. In *Proc. of the ECML/PKDD-01 Workshop on Integration of Data Mining, Decision Support and Meta-Learning*, pp. 55-66.

[9] Witten, I., Frank, E. (1999). Data mining: practical machine learning tools and techniques with Java implementations. Morgan Kaufman, San Francisco.