An Agent-Based Operational Model for Hybrid Connectionist-Symbolic Learning*

José C. González, Juan R. Velasco, and Carlos A. Iglesias

Dep. Ingeniería de Sistemas Telemáticos
Universidad Politécnica de Madrid, SPAIN
{cif,jcg,juanra}@gsi.dit.upm.es

Abstract. Hybridization of connectionist and symbolic systems is being proposed for machine learning purposes in many applications for different fields. However, a unified framework to analyse and compare learning methods has not appeared yet. In this paper, a multiagent-based approach is presented as an adequate model for hybrid learning. This approach is built upon the concept of bias.

1 Introduction

In her work *"Bias and Knowledge in Symbolic and Connectionist Induction"* [3], M. Hilario addresses a key issue in the Machine Learning field: the need of a unified framework for the analysis, evaluation and comparison of different (symbolic, connectionist, hybrid, ...) learning methods. This need is justified upon the fact that there are no universally superior methods for induction. She builds this ***unified*** framework upon the concept of ***bias***.

This paper follows the same line, but from a different perspective. The main point here is that the conceptual-level framework put forward by Hilario can be complemented with its counterpart at the operational level. The purpose of this work is, then, three-fold:

- Firstly, showing that the agent-based paradigm can provide a neutral, unbiased, operational model for such a unified framework.

* This research is funded in part by the Commission of the European Communities under the ESPRIT Basic Research Project *MIX: Modular Integration of Connectionist and Symbolic Processing in Knowledge Based Systems*, ESPRIT-9119, and by CYCIT, the Spanish Council for Research and Development, under the project *M2D2: Metaaprendizaje en Minería de Datos Distribuida*, TIC97-1343. The MIX consortium is formed by the following institutions and companies: Institute National de Recherche en Informatique et en Automatique (INRIA–Lorraine/CRIN–CNRS, France), Centre Universitaire d'Informatique (Université de Genève, Switzerland), Institute d'Informatique et de Mathématiques Appliquées de Grenoble (France), Kratzer Automatisierung (Germany), Fakultät für Informatik (Technische Universität München, Germany) and Dep. Ingeniería de Sistemas Telemáticos (Universidad Politécnica de Madrid, Spain).

- Secondly, showing that this model includes most the known forms of meta-learning proposed by the machine learning community.
- Thirdly, showing that this kind of model may help to overcome some of the traditionally weak points of the work around meta-learning.

2 A Distributed Tool for Experimentation

In the MIX project, several models of hybrid systems integrating connectionist and symbolic paradigms for supervised induction have been studied and applied to well-defined real world problems in different domains. These models have been implemented through the integration of software components (including connectionist and symbolic ones) on a common platform. This platform was developed partially under and for the MIX project. Software components are encapsulated as agents in a distributed environment. Agents in the MIX platform offer their services to other agents, carrying them out through cooperation protocols.

In the past, the platform has been mainly used for building object-level hybrids, i.e. hybrid systems developed to improve performance (in comparison with symbolic or connectionist systems alone) when carrying out particular tasks (prediction or classification) on specific real-world problems.

This application-oriented work has led to good results (in terms of increase of performance, measured as a reduction of task error rates). Some amount of qualitative knowledge about hybridization was derived from these experiences. However, this knowledge is not enough for guiding the selection of an adequate problem-solving strategy in face of a particular problem. Summing up, what we should look for are general and well-founded bias management techniques, calling bias *"any basis for choosing one generalization over another, other than strict consistency with accepted domain knowledge and the observed training instances"* [3].

Our proposal is that the same platform used until now for object-level hybrids, be used to explore different bias management policies. A general architecture to do so can be seen in Fig. 1. This architecture will be particularised for several interesting cases. But, before that, a brief overview of the concept of bias, classified along four levels, will be presented.

3 Classes of Bias

Hilario distinguishes between two kinds of bias, representational and search bias, that can be studied at different grain levels. We classify these granularity levels as follows:

- Hypothetical level.
  On the representational side, it has to do with the selection of formalisms or languages used for the description of hypothesis and instances in the problem space.

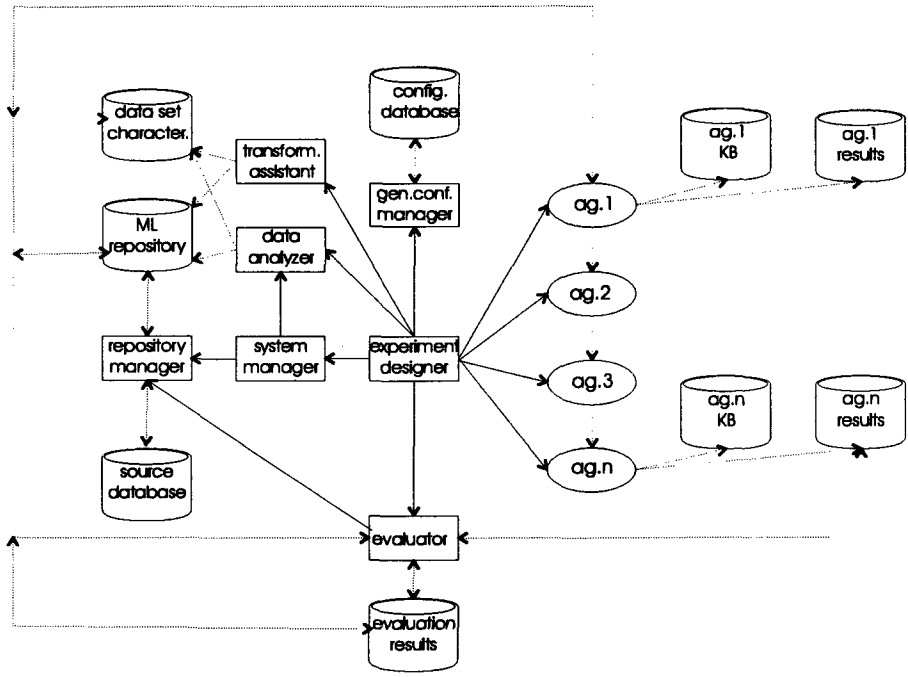

Fig. 1. A multi-agent architecture for bias management

Regarding search, this level deals with the kind of task we are trying to accomplish through automatic means: classification, prediction, optimization, etc.

- Strategic level.
  A particular representation model (production rules, decision trees, perceptrons, etc.) has to be selected, compatible with the formalism preferred at the previous level. This model is built by a particular learning algorithm by searching the hypothesis space.
- Tactical level.
  Once a pair model/algorithm has been selected, some tactical decisions may remain to be taken about the representation model (e.g., model topology in neural nets) or the search model (number of generations in genetic algorithms, stopping criteria when inducing decision trees, etc.)
- Semantic level.
  This level concerns the interpretation of the primitive objects, relations and operators. Concerning representation, this level includes the selection, combination, normalization (scaling, in general), discretization, etc. of attributes in the problem domain. Semantic level search bias includes the selection of weight updating operator in neural nets and fitness updating operator in genetic algorithms, the information-content measure used for the selection

of the most informative attribute in algorithms for the induction of decision trees, etc.

4 Case 1. Semantic Level Bias: Attribute Selection

The determination of the relevant attributes for a particular task is a fundamental issue in any machine learning application. Statistical techniques should play a fundamental role for this purpose. However, commercial tools integrating statistical analysis along with symbolic or connectionist machine learning algorithms have appeared only recently. For instance, the researcher needs to have a clear idea about the correlation between variables for guiding the experiments: dropping variables, crating new ones by combination of others, etc. The evaluator may compare the results obtained by a particular learning algorithm applied to different subsets or supersets of the source data-set looking for statistically significant differences.

The data analyser in Fig. 1 takes a data set in the machine learning repository as input and produces a description of this data set in terms of problem type (classification, prediction or optimization), size (amount of variables and samples), statistical measures (variable distribution and correlation), consistency measures (amount of contradictory samples), information measures (absolute and conditional entropy of variables, presence of missing values), etc.

A transformation agent (not shown in the figure) can be coupled in this architecture. The goal of this agent is proposing experiments from data sets generated from the source one. Transformed data sets may be obtained by several methods:

- Sampling: it is almost compulsory for data sets too big for machine learning processes. Moreover, random or stratified sampling techniques can be necessary for experimental purposes.
- Dropping of variables: the less informative variables can be considered as noise. Noise makes learning more difficult.
- Replacing or adding variables: the new ones can be formed by combination of others (to be deleted or not).
- Clustering of samples: the activity of a system may fall in different macro-states where the behaviour of the system may be qualitatively different. These differences can be associated with completely different deep models, in such a way that learning algorithms might perform better when trained from cases in one individual macro-state.
- Discretization of variables: the precision used to represent a continuous variable can hide the fact that precision does not imply relevance. Some machine learning algorithms handle only discrete variables, but discretization can attain performance improvements with algorithms capable of managing continuous and discrete attributes. Discretization can be achieved by crisp methods (splitting the range of a variable in homogeneous sub-ranges in terms of size or number of cases falling in the range), or non-crisp ones (by connectionist or fuzzy clustering techniques).

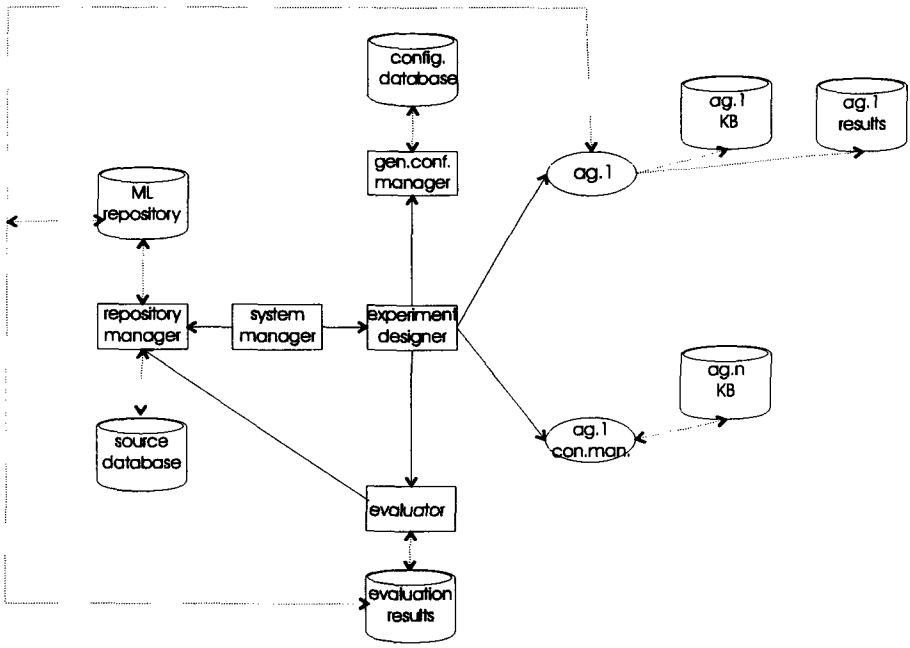

Fig. 2. Architecture for tactical bias selection

5 Case 2. Tactical Level Bias: Parameter Selection

A good amount of work can be found in the literature about systems intended for the selection of adequate representational or search bias at the tactical level. For instance, the C45TOX system, developed for a toxicology application in the MIX project, uses genetic algorithms for optimising the parameters used by the C4.5 learning algorithm. A work with the same goal had been previously developed by Kohavi and John [4]. They used a wrapper algorithm for parameter setting.

In the C45TOX system, the genetic algorithm acts as a specialised configuration manager. It provides the experiment designer with candidate sets of parameters that are used for training a decision tree. This tree is tested using cross-validation. The evaluator agent estimates the performance of the decision tree and transmits the error rate to the genetic agent to update the fitness of the corresponding individual of the population. The knowledge base of the genetic system evolves through the application of genetic operators. When a new generation is obtained, new experiments are launched until no significant improvement is achieved.

The architecture of this system is shown in Fig. 2.

6 Case 3. Hypothetical/Strategic Level Bias: Algorithm Selection

Advances in software technology, and specially in the field of distributed processing permit the easy integration of several algorithms co-operating to carry our a particular task: classification, prediction, etc. Differences in performance estimated at training time can be used to configure strategies for bias management through arbiters or combiners. Both, arbiters and combiners, can be developed according to fixed policies (e.g., a majority voting scheme in the case of arbiters) or variable policies.

One interesting research avenue in the field of meta-learning concerns the selection of the most adequate algorithm for a task according to variable inductive policies. One of the biggest efforts done following this line has taken place in the framework of the STATLOG project [5, 2]. 24 different algorithms were applied to 22 database classical in the machine learning literature. Finding mappings between tasks and biases was proposed first as a classification problem (to select the best candidate algorithm for an unseen task). For this purpose, C4.5 was used. Afterwards, meta-learning was implemented as a prediction problem intended to estimate the performance of a particular algorithm in comparison with others in face of an unseen database.

Some difficulties are evident with this approach. First, 22 data-sets are too few for meta-learning. Second, standard (default) parameters were used to configure each algorithm. Nobody knows, then, if the low performance of an individual system comes from itself or from a bad selection of parameters. All the meta-learning systems described in the literature [7, 1, 8] suffer from similar drawbacks.

In Fig. 3 we show the instantiation of the proposed distributed architecture for strategic bias selection. Systems are characterised according to their performance (basically, error rate, error cost, computing time and learning time) on a particular data-set.

The architecture has several appealing features:

- Full integration. The meta-learning agents are exactly the same used for object-level learning. In the same way, several learning agents can be launched simultaneously for meta-learning, and their results can be compared or integrated in an arbiter or combiner structure.
- On-line learning. Meta-learning can be achieved simultaneously with object-level learning.
- Use of transformed and artificial data-sets. The lack of source data-bases is a difficulty that can be overcome through the generation of new data-sets obtained from the transformation of the original ones. New attributes can be derived or noise can be added in order to test noise-immunity. Even fully artificial data-bases can be generated from rules or any other mechanism, controlling at the same time the level of noise to be added.

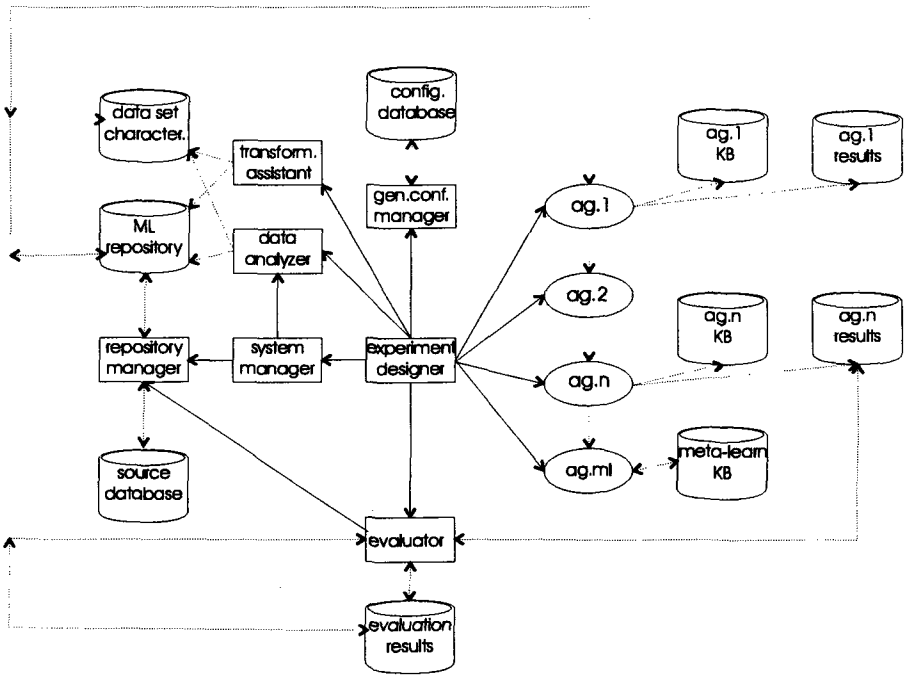

Fig. 3. Architecture for strategic bias selection

7 Current Work

The ideas and the architecture proposed in this paper are being implemented at this moment in the project M2D2 (*"Meta-Learning in Distributed Data Mining"*), funded by CYCIT, the Spanish Council for Research and Development. This approach has been successfully used, for instance, for the development of the C45TOX system.

References

1. P. Chan and S. Stolfo. A comparative evaluation of voting and meta-learning on partitioned data. In Prieditis and Russell [6], pages 90–98.
2. J. Gama and P. Brazdil. Characterization of classification algorithms. In E. Pinto-Ferreira and N. Mamede, editors, *Progress in Artificial Intelligence. Proceedings of the 7th Portuguese Conference on Artificial Intelligence (EPIA-95)*, pages 189–200. Springer-Verlag, 1995.
3. Melanie Hilario. Bias and knowledge in symbolic and connectionist induction. Technical report, Centre Universitaire d'Informatique, Université de Genève, Genève, Switzerland, 1997.
4. R. Kohavi and G. John. Automatic parameter selection by minimizing estimated error. In Prieditis and Russell [6], pages 304–312.

5. Donald Michie, David J. Spiegelhalter, and CharlesC. Taylor, editors. *Machine Learning, Neural and Statistical Classification.* Ellis Horwood, 1994.
6. A. Prieditis and S. Russell, editors. *Proceedings of the 11th International Conference on Machine Learning*, Tahoe City, CA, 1995. Morgan Kaufmann.
7. L. Rendell, R. Seshu, and D. Tcheng. Layered-concept learning and dinamically variable bias management. In *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, pages 308–314, Milan, Italy, 1987. Morgan Kaufmann.
8. G. Widmer. Recognition and exploitation of contextual cues via incremental meta-learning. Technical Report OFAI–TR–96–01, Austria Research Institute for Artificial Intelligence, Vienna, Austria, 1996.