Meta-Learning

Pavel Brazdil,
LIAAD-INESC Porto L.A./Faculdade de Economia, University of Porto, Portugal

Ricardo Vilalta
Department of Computer Science, University of Houston, USA

Christophe Giraud-Carrier
Department of Computer Science, Brigham Young University, USA

Carlos Soares
LIAAD-INESC Porto L.A./Faculdade de Economia, University of Porto, Portugal

Synonyms

Ranking learning methods; adaptive learning; self-adaptive systems; learning to learn; dynamic selection of bias.

Definition

Meta-learning allows machine learning systems to benefit from their repetitive application. If a learning system fails to perform efficiently, one would expect the learning mechanism itself to adapt in case the same task is presented again. Metalearning differs from base-learning in the scope of the level of adaptation; whereas learning at the base-level is focused on accumulating experience on a specific task (e.g., credit rating, medical diagnosis, mine-rock discrimination, fraud detection, etc.), learning at the metalevel is concerned with accumulating experience on the performance of multiple applications of a learning system.

Briefly stated, the field of metalearning is focused on the relation between tasks or domains, and learning algorithms. Rather than starting afresh on each new task, metalearning facilitates evaluation and comparison of learning algorithms on many different previous tasks, establishes benefits and disadvantages, and then recommends the learning algorithm, or combination of algorithms, that maximizes some utility function on the new task. This problem can be seen as an algorithm selection task (Rice 1976) [9].

The utility or usefulness of a given learning algorithm is often determined through a mapping between a characterization of the task and the algorithm's estimated performance (Brazdil & Henery, 1994) [2]. In general, metalearning can recommend more than one algorithm. Typically, the number of recommended algorithms is significantly smaller than the number of all possible (available) algorithms (Brazdil et al., 2009) [3].

Motivation and Background

The application of machine learning systems to classification and regression tasks has become a standard, not only in research but also in commerce and industry (e.g., finance, medicine, and engineering). However, most successful applications are custom-designed, the result of skillful use of human expertise. This is due, in part, to the large, ever increasing number of available machine learning systems, their relative complexity, and the lack of systematic methods for discriminating among them. The problem is further compounded by the fact that, in Knowledge Discovery from Databases, each operational phase (e.g., pre-processing, model generation) may involve a choice among various possible alternatives (e.g., progressive vs. random sampling, neural network vs. decision tree learning), as observed by Bernstein et al. (2005) [1].

Current data mining systems are only as powerful as their users. These tools provide multiple algorithms within a single system, but the selection and combination of these algorithms must be performed before the system is invoked, generally by an expert user. For some researchers, the choice of learning and data transformation algorithms should be fully automated if machine learning systems are to be of any use to non-specialists. Others claim that full automation of the data mining process is not within the reach of current technology. An intermediate solution is the design of assistant systems aimed at helping to select the right learning algorithm(s). Whatever the proposed solution, there seems to be an implicit agreement that metaknowledge should be integrated seamlessly into the data mining system. Metalearning focuses on the design and application of learning algorithms to acquire and use metaknowledge to assist machine learning users with the process of model selection. A general framework for this purpose, together with a survey of approaches, is in (Smith-Miles, 2008) [8].

Metalearning is often seen as a way of redefining the space of inductive hypotheses searched by the learning algorithm(s). This issue is related to the idea of search bias, that is, search factors that affect the definition or selection of inductive hypotheses (Mitchell, 1997) [5]. In this sense, metalearning studies how to choose the right bias dynamically, and thus differs from base-level learning, where the bias is fixed or user-parameterized. Metalearning can also be viewed as an important feature of self-adaptive systems, that is, learning systems that increase in efficiency through experience (Vilalta & Drissi, 2002) [7].

Structure of the Metalearning System

A metalearning system is essentially composed of two parts. One part is concerned with the acquisition of metaknowledge for machine learning systems. The other part is concerned with the application of metaknowledge to new problems with the objective of identifying an optimal learning algorithm or technique. The latter part – application of metaknowledge – can be used to help to select or adapt suitable machine learning algorithms. So, for instance, if we are dealing with a classification task, metaknowledge can be used to select a suitable classifier for the new problem. Once this has been done, one can train the classifier and apply it to some unclassified sample for the purpose of class prediction.

In the following sections we begin by describing scenarios corresponding to the case when metaknowledge has already been acquired. We then provide an explanation of how this knowledge is acquired.

Employing Metaknowledge to Select Machine Learning Algorithms

The aim of this section is to show that metaknowledge can be useful in many different settings. We will start by considering the problem of selecting suitable machine learning algorithms from a given set. The problem can be seen as a search problem. The search space includes the individual machine learning algorithms and the aim is to identify the best algorithm. This process can be divided into two separate phases (see Fig. 1). In the first phase the aim is to identify a suitable subset of machine learning algorithms based on an input dataset. The selection method used in this process can exploit metaknowledge. This is in general advantageous, as it often leads to better choices. In some work the result of this phase is represented in the form of a ranked subset of machine learning algorithms. The subset of algorithms represents the reduced bias space. The ranking (i.e. ordering of different algorithms) represents the procedural search bias.

The second phase is used to search through the reduced space. Each option is evaluated using a given performance criteria (e.g., accuracy). Typically, cross-validation will be used to identify the best alternative.

We note that metaknowledge does not completely eliminate the need for the search process, but rather provides a more effective search. The search effectiveness depends on the quality of metaknowledge.

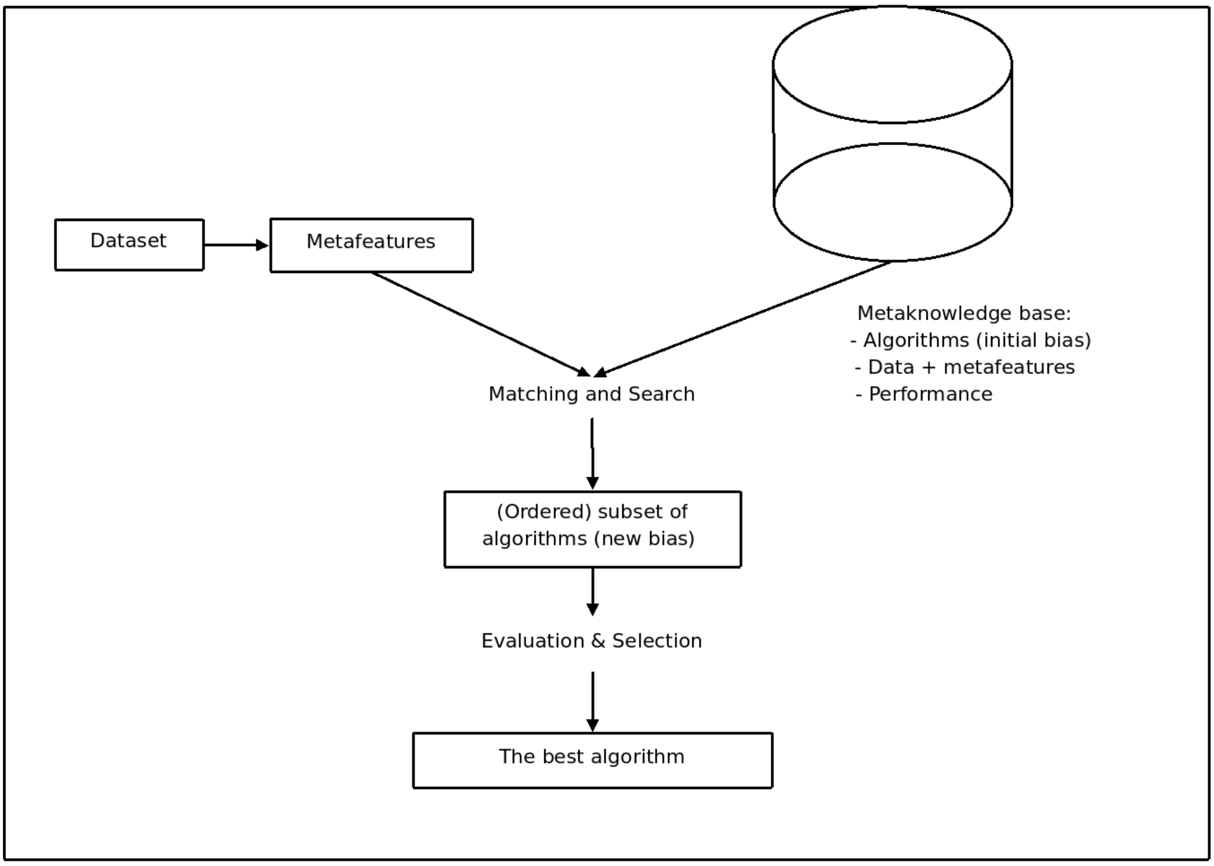

Fig. 1. Selection of machine learning algorithms: Determining the reduced space and selecting the best alternative.

How the Subset of Algorithms is Identified

Let us return to the algorithm selection problem. A metalearning approach to solving this problem relies on dataset characteristics or metafeatures to provide some information that would differentiate the performance of a set of given learning algorithms. These include various types of measures discussed in detail below.

Much previous work in dataset characterization has concentrated on extracting statistical and information-theoretic parameters estimated from the training set. Measures include number of classes, number of features, ratio of examples to features, degree of correlation between features and target concept, average class entropy, etc. (Engels et al., 1998) [4]. The disadvantage of this approach is that there is a limit to how much information these features can capture, given that all these measures are uni- or bi-lateral measures only (i.e., they capture relationships between two attributes only or one attribute and the class).

Another idea is based on what are called *landmarkers;* these are simple and fast learners (Pfahringer et al., 2000) [6]. The accuracy of these simplified algorithms is used to characterize a dataset and to identify areas where each type of learner can be regarded as an expert. An important class of measures related to landmarkers uses information obtained on simplified versions of the data (e.g., samples). Accuracy results on these samples serve to characterize individual datasets and are referred to as *sub-sampling landmarks*.

One different class of techniques does not acquire the information in one step, but rather uses a kind of active learning approach. This approach has been used to characterize algorithms by exploiting performance results on samples. The process of obtaining a characterization is divided into several steps. The result of one step affects what is done in the next step. In each step, a decision is first made as to whether the characterization process should be continued. If the answer is positive, the system determines which characteristics should be obtained in the next step (Brazdil et al., 2009) [3].

All the measures discussed above are used to identify a subset of learning algorithms to reduce the search space (Fig. 1). The second phase in the algorithm selection problem can be done using a meta-level system that maps data characteristics to learning algorithms. One particular approach uses the *k*-NN method at the meta-level. The *k*-NN method is used to identify the most similar datasets. For each of these datasets, a ranking of the candidate algorithms is generated based on user-defined performance criteria, such as accuracy and learning time (Nakhaeizadeh and Scnabl, 2007) [10]. The rankings obtained are aggregated to generate a final recommended ranking of algorithms.

Acquisition of Metaknowledge

We now address how metaknowledge can be acquired. One possibility is to rely on expert knowledge. Another possibility is to use an automatic procedure. We explore both alternatives briefly below.

One way of representing metaknowledge is in the form of rules that match domain (dataset) characteristics with machine learning algorithms. Such rules can be hand-crafted, taking into account theoretical results, human expertise, and empirical evidence. For example, in decision tree learning, a heuristic rule can be used to switch from univariate tests to linear tests if there is a need to construct non-orthogonal partitions over the input space. This method has serious disadvantages however. First, the resulting rule set is likely to be incomplete. Second, timely and accurate maintenance of the rule set as new machine learning algorithms become available is problematic. As a result, most research has focused on automatic methods, discussed next.

One other way of acquiring metaknowledge relies on automatic experimentation. For this we need a pool of problems (datasets) and a set of machine learning algorithms that we wish to consider. Then we need to define also the experimental method which determines which alternatives we should experiment with and in which order (see Fig. 2 for details).

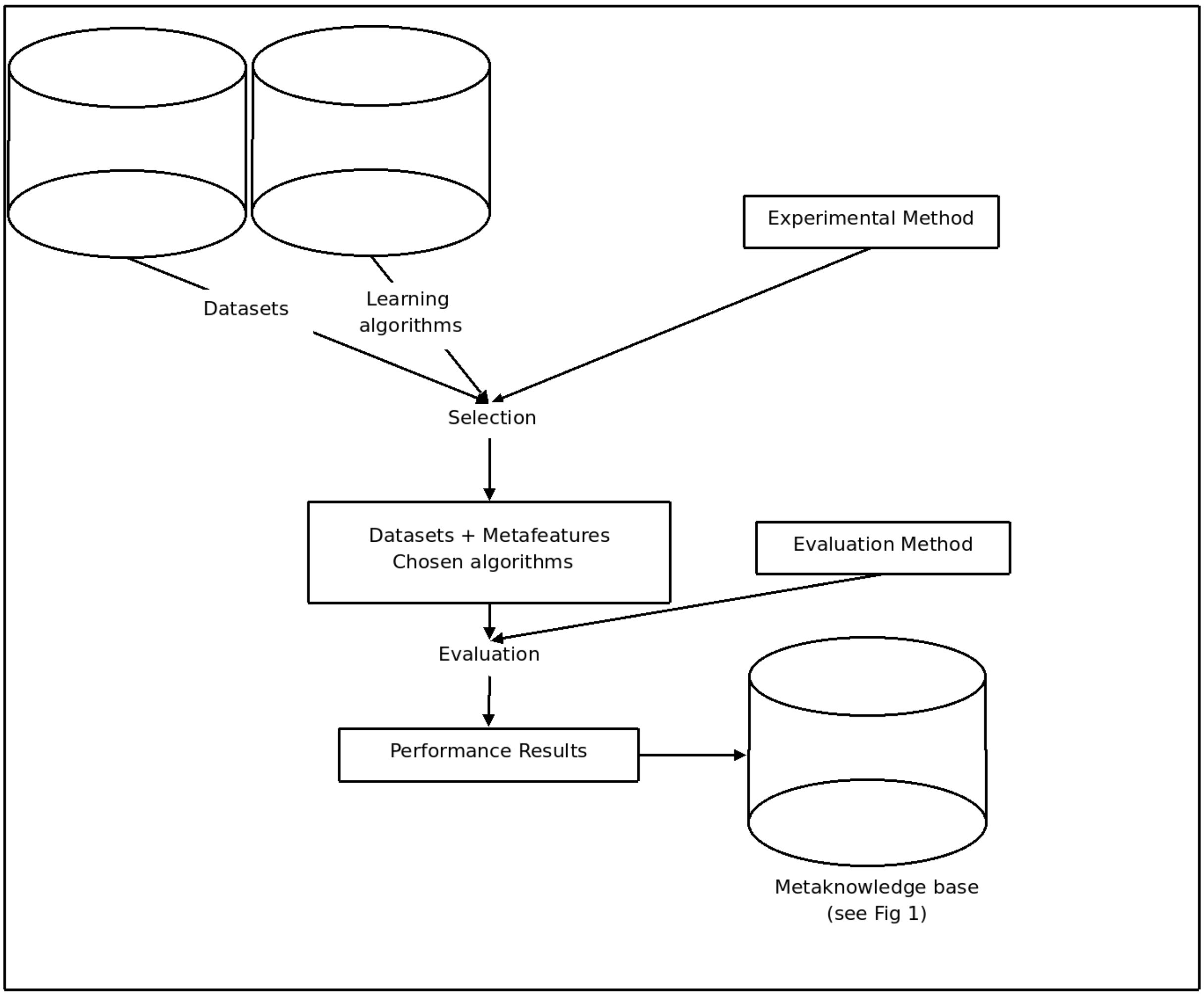

Fig. 2 Acquisition of Metadata for the Metaknowledge base

Suppose we have a dataset (characterized using certain metafeatures), in combination with certain machine learning algorithms. The combination is assessed using an evaluation method (e.g., cross-validation) to produce performance results. The results, together with the characterization, represent a piece of metadata that is stored in the metaknowledge base. The process is then repeated for other combinations of datasets and algorithms.

In this context it is useful to distinguish between two different types of methods potentially available at the meta-level. One group involves lazy learning methods. These delay the generalization of metadata to the application phase. The other group involves learning algorithms whose aim is to generate a generalization model (e.g., a decision tree or decision rules). This generalization model, applied to the metadatabase, represents in effect acquired metaknowledge.

Inductive Transfer

As we mentioned before, learning should not be viewed as an isolated task that starts from scratch on every new problem. As experience accumulates, the learning mechanism is expected to perform increasingly better. One approach to simulate the accumulation of experience is by transferring metaknowledge across domains or tasks. This process is known as *inductive transfer*. In many cases the goal is not simply to generate explicit metaknowledge, but rather to incorporate it in the given base-level system(s). The resulting base-level system becomes thus a generic solution applicable across domains. More details about this can be found in a separate entry on this topic.

See also:

Inductive Transfer

Recommended Readings

[1] Bernstein, A., Provost, F. and Hill, S. (2005). Toward Intelligent Assistance for a Data Mining Process: An Ontology-based Approach for Cost-sensitive Classification. IEEE Transactions on Knowledge and Data Engineering, **17**(4):503-518.

[2] Brazdil P. and Henery, R. (1994). Analysis of Results, in Michie, D., Spiegelhalter, D. J. and Taylor, C.C. (Eds.) *Machine Learning, Neural and Statistical Classification.* England: Ellis Horwood.

[3] Brazdil P., Giraud-Carrier, C., Soares, C. and Vilalta, R. (2009). *Metalearning – Applications to Data Minin*g, Springer.

[4] Engels, R. and Theusinger, C. (1998). Using a Data Metric for Offering Preprocessing Advice in Data-mining Applications. In *Proceedings of the 13th European Conference on Artificial Intelligence, 430-434.*

[5] Mitchell, T. (1997). *Machine Learning*, McGraw Hill.

[6] Pfahringer, B., Bensusan, H. and Giraud-Carrier, C. (2000). Meta-learning by Landmarking Various Learning Algorithms. In *Proceedings of the 17th International Conference on Machine Learning*, 743-750.

[7] Vilalta, R. and Drissi, Y. (2002). A Perspective View and Survey of Metalearning. *Artificial Intelligence Review*, **18**(2):77-95.

[8] Smith-Miles, K.A. (2008). Cross-Disciplinary Perpectives on Meta-Learning for Algorithm Selection. *ACM Computing Surveys*, 41(1), Article No. 6, 2008.

[9] Rice, J.R. (1976). The Algorithm Selection Problem. *Advances in Computers*, *15*, pp. 65-118.

[10] Nakhaeizadeh, G. and Schnabl, A. (1997). Development of Multi-criteria Metrics for Evaluation of Data Mining Algorithms. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, 37-42.