Preface

The continuous growth of successful applications in machine learning and data mining has led to an apparent view of real progress towards the understanding of the nature and mechanisms of learning machines. From the automated classification of millions of luminous objects stored in astronomical images, to the complex analysis of long sequences of genes in biomedical data sets, machine learning has positioned itself as an indispensable tool for data analysis, pattern recognition, and scientific discovery. This apparent progress in the search for accurate predictive models relies on the design of learning algorithms exhibiting novel functionalities. The history of machine learning shows a research community devoted to the study and improvement of a highly diverse set of learning algorithms such as nearest neighbors, Bayesian classifiers, decision trees, neural networks, and support vector machines (to name just a few). While the design of new learning algorithms is certainly important in advancing our ability to finding accurate data models, so is the understanding of the relation between data set characteristics and the particular mechanisms embedded in the learning algorithm. Rather than testing multiple algorithms to assess which one would perform satisfactorily on a certain data set, the end user needs guidelines pointing to the best learning strategy for the particular problem at hand. Researchers and practitioners in machine learning have a clear need to answer the following question: what works well where? There is a strong need to characterize both data distributions and learning mechanisms to construct a theory of learning behavior. Moreover, we advocate the development of a new generation of learning algorithms that are capable of profound adaptations in their behavior to the input data. This may include changes to the model language itself.

Despite different interpretations of the term metalearning, in this book we pursue the goal of finding principled methods that can make learning algorithms adaptive to the characteristics of the data. This can be achieved in many ways as long as there is some form of feedback relating learning performance with data distributions. Thus one can think of the problem of algorithm selection (or ranking), or algorithm combination, as frameworks that exploit past performance to guide the selection of a final model. The ultimate end is to design learning algorithms that adapt to the problem at hand, rather than invoking the same fixed mechanisms independent of the nature of the data under analysis. We explain in this book that a unifying theme in metalearning is that of exploiting experience or metaknowledge to achieve flexible learning systems.

These ideas have brought us together to write a book that summarizes the current state of the art in the field of metalearning. The motivation for such a book can be traced back to the METAL project [166], in which the first three authors were active participants. Our first (electronic) meetings regarding this book took place in the second half of 2005 and have continued over the following three years until the second half of 2008. The project proved challenging in many ways, most particularly in unifying our view concerning the scope of the term metalearning. After long discussions we finally agreed on the definition provided in Chapter 1. Equally challenging was to decide on a list of topics that stand as clearly representative of the field. We hope the reader will find our selection appropriate and sufficiently broad to offer adequate coverage. Finally, it is our hope that this book will serve not only to place many of the ideas now dispersed all over the field into a coherent volume, but also to encourage researchers in machine learning to consider the importance of this fascinating area of study.

Book Organization

The current state of diverse ideas in the field of metalearning is not yet mature enough for a textbook based on a solid conceptual and theoretical framework of learning performance. Given this, we have decided to cover the main topics where there seems to be a clear consensus regarding their relevance and legitimate membership in the field. In the following, we briefly describe the contents of the book acknowledging the contribution of each of the authors. In Chapter 1, all of us worked on introducing the main ideas and concepts that we believe are essential to understanding the field of metalearning. Chapters 2–4 have a more practical flavor, illustrating the important problem of selecting and ranking learning algorithms, with a description of several currently operational applications. In Chapter 2, Soares and Brazdil describe a simple meta-learning system that, given a dataset, provides the user with guidance concerning which learning algorithm to use. The issues that are involved in developing such a system are discussed in more detail by Soares in Chapter 3, including a survey of existing approaches to address them. In Chapter 4, Giraud-Carrier describes a number of systems that incorporate some form of automatic user guidance in the data mining process.

Chapters 5–7, on the other hand, have a more conceptual flavor, covering the combination of classifiers, learning from data streams, and knowledge transfer. Chapter 5, authored by Giraud-Carrier, describes the main concepts behind model combination, including classical techniques such as bagging and boosting, as well as more advanced techniques such as delegating, arbitrating and meta-decision trees. We invited Gama and Castillo to contribute to Chapter 6; the chapter discusses the dynamics of the learning process and general strategies for reasoning about the evolution of the learning process itself. The main characteristics and new constraints on the design of learning algorithms imposed by large volumes of data that evolve over time are described, including embedding change-detection mechanisms in the learning algorithm and the trade-off between the cost of update and the gain in performance. Chapter 7, authored by Vilalta, covers the important topic of knowledge transfer across tasks; the chapter covers topics such as multitask learning, transfer in kernel methods, transfer in parametric Bayesian methods, theoretical models of learning to learn, and new challenges in transfer learning with examples in robotics. Lastly, Chapter 8, authored by Brazdil, discusses the important role of metalearning in the construction of complex systems through the composition of induced subsystems. It is shown how domain-specific metaknowledge can be used to facilitate this task.

Acknowledgements

We wish to express our gratitude to all those who helped in bringing this project to fruition. We are grateful to the University of Porto and Faculty of Economics and also to the Portuguese funding organization FCT for supporting the R&D laboratory LIAAD (Laboratory of Artificial Intelligence and Decision Support) where a significant part of the work associated with this book was carried out. We also acknowledge support from the Portuguese funding organization FCT for the project ALES II – Adaptive LEarning Systems. This work was also partially supported by the US National Science Foundation under grant IIS-0448542.

The motivation for this book came from the involvement of the first three authors in an earlier grant from the European Union (ESPRIT project METAL [166]). We would like to acknowledge financial support from this grant and the contribution of our colleagues: Hilan Bensusan, Helmut Berrer, Sašo Džeroski, Peter Flach, Johannes Fürnkranz, Melanie Hilario, Jörg Keller, Tom Khabaza, Alexandros Kalousis, Petr Kuba, Rui Leite, Guido Lindner, Reza Nakhaeizadeh, Iain Paterson, Yonghong Peng, Rui Pereira, Johann Petrak, Bernhard Pfahringer, Luboš Popelínský, Ljupčo Todorovski, Dietrich Wettschereck, Gerhard Widmer, Adam Woznica and Bernard Zenko.

We are greatly indebted to several other colleagues for their many comments and suggestions that helped improve earlier versions of this book: Bart Bakker, Theodoros Evgeniou, Tom Heskes, Rich Maclin, Andreas Maurer, Tony Martinez, Massimiliano Pontil, Rajat Raina, André Rossi, Peter Stone, Richard Sutton, Juergen Schmidhuber, Matthew Taylor, Lisa Torrey and Roberto Valerio.

Pavel Brazdil would also like to gratefully acknowledge that Robert Kowalski drew his attention to the topic of metareasoning in the 70's and

to express his gratitude to the late Donald Michie for having accepted the proposal to include metalearning in the preparatory stages of the StatLog project [169] in 1989.

Finally, we are most grateful to our editor at Springer, Ronan Nugent, for his patience, gentle prodding and encouragement throughout this project.

Porto, Portugal; Provo, Houston, USA
July 2008

Pavel Brazdil
Christophe Giraud-Carrier
Carlos Soares
Ricardo Vilalta

1

Metalearning: Concepts and Systems

1.1 Introduction

Current data mining (DM) and machine learning (ML) tools are characterized by a plethora of algorithms but a lack of guidelines to select the right method according to the nature of the problem under analysis. Applications such as credit rating, medical diagnosis, mine-rock discrimination, fraud detection, and identification of objects in astronomical images generate thousands of instances for analysis with little or no additional information about the type of analysis technique most appropriate for the task at hand. Since real-world applications are generally time-sensitive, practitioners and researchers tend to use only a few available algorithms for data analysis, hoping that the set of assumptions embedded in these algorithms will match the characteristics of the data. Such practice in data mining and the application of machine learning has spurred the research community to investigate whether learning from data is made of a single operational layer – search for a good model that fits the data – or whether there are in fact several operational layers that can be exploited to produce an increase in performance over time. The latter alternative implies that it should be possible to learn about the learning process itself, and in particular that a system could learn to profit from previous experience to generate additional knowledge that can simplify the automatic selection of efficient models summarizing the data.

This book provides a review and analysis of a research direction in machine learning and data mining known as *metalearning*.[1] From a practical standpoint, the goal of metalearning is twofold. On the one hand, we wish to overcome some of the challenges faced by users with current data analysis tools. The aim here is to aid users in the task of selecting a suitable predictive model (or combination of models) while taking into account the domain of application. Without some kind of assistance, model selection and combination

Meta-learning

[1] We assume here that the reader is familiar with concepts in machine learning. Many books that provide a clear introduction to the field of machine learning are now available (e.g., [82, 26, 3, 174]).

can turn into solid obstacles to end users who wish to access the technology more directly and cost-effectively. End users often lack not only the expertise necessary to select a suitable model, but also the availability of many models to proceed on a trial-and-error basis. A solution to this problem is attainable through the construction of metalearning systems that provide automatic and systematic user guidance by mapping a particular task to a suitable model (or combination of models).

On the other hand, we wish to address a problem commonly observed in the practical use of data analysis tools, namely how to profit from the repetitive use of a predictive model over similar tasks. The successful application of models in real-world scenarios requires continuous adaptation to new needs. Rather than starting afresh on new tasks, one would expect the learning mechanism itself to relearn, taking into account previous experience (e.g., [50, 254, 193]). This area of research, also known as *learning to learn*, has seen many new developments in the past few years. Here too, metalearning systems can help control the process of exploiting cumulative expertise by searching for patterns across tasks.

Learning to learn

Our goal in this book is to give an overview of the field of metalearning by attending to both practical and theoretical concepts. We describe the current state of the art in different topics such as techniques for algorithm recommendation, extending metalearning to cover data mining and knowledge discovery, combining classifiers, time-changing data streams, inductive transfer or transfer of metaknowledge across tasks, and composition of systems and applications. Our hope is to stimulate the interest of both practitioners and researchers to invest more effort in this interesting field of research. Despite the promising directions offered by metalearning and important recent advances, much work remains to be done. We also hope to convince others of the important task of expanding the adaptability of current computer learning systems towards understanding their own learning mechanisms.

1.1.1 Base-Learning vs. Metalearning

We begin by clarifying the distinction between the traditional view of learning – also known as base-learning – and the one taken by metalearning. 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 learning task, learning at the meta level is concerned with accumulating experience on the performance of multiple applications of a learning system. In a typical inductive learning scenario, applying a base-learner (e.g., decision tree, neural network, or support vector machine) on some data produces a predictive function (i.e., hypothesis) that depends on the *fixed* assumptions embedded in the learner. Learning takes place at the base level because the quality of the function or hypothesis normally improves with an increasing number of examples. Nevertheless, successive applications of the

Base-learning

learner on the same data always produces the same hypothesis, independently of performance; no knowledge is extracted across domains or tasks.

As an illustration, consider the task of learning to classify medical patients in a hospital according to a list of potential diseases. Given a large dataset of patients, each characterized by multiple parameters (e.g., blood type, temperature, blood pressure, medical history, etc.) together with the diagnosed disease (or alternatively no disease), one can train a learning algorithm to predict the right disease for a new patient. The resulting predictive function normally improves in accuracy as the list of patients increases. This is learning at the base level where additional examples (i.e., patients) provide additional statistical support to unveil the nature of patterns hidden in the data.

Working at the base level exhibits two major limitations. First, data patterns are usually not placed aside for interpretation and analysis, but rather embedded in the predictive function itself. Successive training of the learning algorithm over the same data fails to accumulate any form of experience. Second, data from other hospitals can seldom be exploited unless one merges all inter-hospital patient data into a single file. The experience or knowledge gained when applying a learning algorithm using data from one hospital is thus generally not readily available as we move to other hospitals. A key to solving these problems is gathering knowledge about the learning process, also known as *metaknowledge*. Such knowledge can be used to improve the learning mechanism itself after each training episode. Metaknowledge may take on different forms and applications, and can be defined as any kind of knowledge that is derived in the course of employing a given learning system. Advances in the field of metalearning hinge on the acquisition and effective exploitation of knowledge about learning systems (i.e., metaknowledge) to understand and improve their performance.

Meta-knowledge

1.1.2 Dynamic Bias Selection

The field of metalearning studies how learning systems can become more effective through experience. The expectation is not simply that a good solution be found, but that this be done increasingly more effectively through time. The problem can be cast as that of determining the right bias for each task. The notion of learning bias is at the core of the study of machine learning. Bias refers to any preference for choosing one hypothesis explaining the data over other (equally acceptable) hypotheses, where such preference is based on extra-evidential information independent of the data (see [173, 112] for other similar definitions of bias).

Learning bias

Unlike base-learning, where the bias is *fixed* a priori or user-parameterized, metalearning studies how to choose the most adequate bias dynamically. The view presented here is aligned with that formulated originally by Rendell et al. [206]: *Metalearning is to learn from experience when different biases are appropriate for a particular problem.* This definition leaves some important issues unresolved, such as the role of metaknowledge (explained below) and how the process of adaptation takes place. We defer giving our own definition of metalearning (until Section 1.3) after we provide additional concepts through a brief overview on the contents of the book.

Declarative bias

Procedural bias

Metalearning covers both declarative and procedural bias. Declarative bias specifies the representation of the space of hypotheses, and affects the size of the search space (e.g., represent hypotheses using linear functions only, or conjunctions of attribute values). Procedural bias imposes constraints on the ordering of the inductive hypotheses (e.g., prefer smaller hypotheses). Both types of bias affect the effectiveness of a learning system on a particular task. Searching through the (declarative and procedural) bias space causes a metalearning algorithm to engage in a time-consuming process. An important aim in metalearning is to exploit metaknowledge to make the search over the bias space manageable.

In the following introductory sections we discuss how metaknowledge can be employed in different settings. We consider for instance the problem of selecting learning algorithms. We then broaden the analysis to discuss the impact of metalearning on knowledge discovery and data mining. Finally, we extend our analysis to adaptive learning, transfer of knowledge across domains and composition of complex systems, and the role metaknowledge plays in each situation.

1.2 Employing Metaknowledge in Different Settings

We proceed in this section by showing that knowledge gained through experience can be useful in many different settings. Our approach is to provide a brief introduction – a foretaste – of what is contained in the remainder of the book. We begin by considering the general problem of selecting machine learning (ML) algorithms for a particular application.

1.2.1 Selecting and Recommending Machine Learning Algorithms

Consider the problem of selecting or recommending a suitable subset of ML algorithms for a given task. The problem can be cast as a search problem, where the search space includes the individual ML algorithms, and the aim is to identify the set of learning algorithms with best performance. A general framework for selecting learning algorithms is illustrated in Figure 1.1. According to this framework, the process can be divided into two phases. In the first phase the aim is to identify a suitable subset of learning algorithms given a training dataset (Figure 1.1a), using available metaknowledge (Figure 1.1c). The output of this phase is a ranked subset of ML algorithms (Figure 1.1d), which represents the new, reduced bias space. The second phase of the process then consists of searching through the reduced space. Each learning algorithm is evaluated using various performance criteria (e.g., accuracy, precision, recall, etc.) to identify the best alternative (Figure 1.1e).

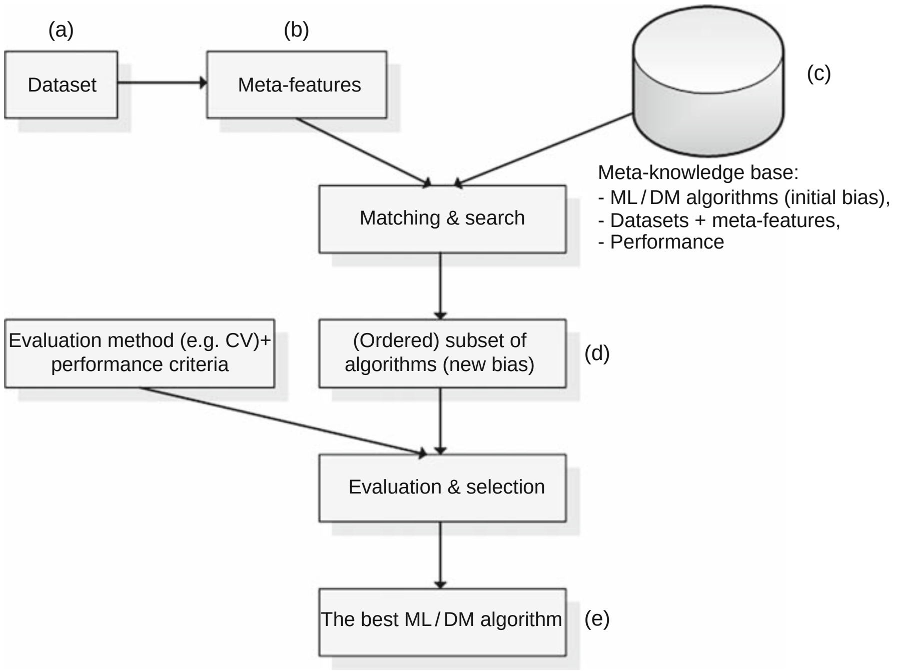
(a)
(b)
Dataset
Meta-features
(c)
Meta-knowledge base:
- ML / DM algorithms (initial bias),
- Datasets + meta-features,
- Performance
Matching & search
Evaluation method (e.g. CV)+ performance criteria
(Ordered) subset of algorithms (new bias)
(d)
Evaluation & selection
The best ML / DM algorithm
(e)

Fig. 1.1. Selection of ML/DM algorithms: finding a reduced space and selecting the best learning algorithm

The above framework differs from traditional approaches in that it exploits a metaknowledge base. As previously mentioned, one important aim in metalearning is to study how to extract and exploit metaknowledge to benefit from previous experience. Information contained in the metaknowledge base can take different forms. It may include, for instance, a set of learning algorithms that have shown good (a priori) performance on datasets similar to the one under analysis; algorithms to characterize ML algorithms and datasets and metrics available to compute dataset similarity or task relatedness. Hence, metaknowledge encompasses not only information useful to perform dynamic bias selection, but also functions and algorithms that can be invoked to generate new useful information.

We note that metaknowledge does not generally completely eliminate the need for search, but rather provides a more effective way of searching through the space of alternatives. It is clear that the effectiveness of the search process depends on the quality of the available metaknowledge.

1.2.2 Generation of Metafeatures

Following the above example, one may ask how the subset of ML algorithms is identified. One form of metaknowledge used during the first phase refers to dataset characteristics or metafeatures (Figure 1.1b); these provide valuable information to differentiate the performance of a set of given learning algorithms. The idea is to gather descriptors about the data distribution that correlate well with the performance of learned models. This is a particularly relevant contribution of metalearning to the field of machine learning, as most work in machine learning focuses instead on the design of multiple learning architectures with a variety of resulting algorithms. Little work has been devoted to understanding the connection between learning algorithms and the characteristics of the data under analysis.

Simple, statistical and information-theoretic metafeatures

So far, three main classes of metafeatures have been proposed. The first one includes features based on statistical and information-theoretic characterization. These metafeatures, estimated from the dataset, include number of classes, number of features, ratio of examples to features, degree of correlation between features and target concept and average class entropy [1, 88, 106, 120, 169, 238]. This method of characterization has been used in a number of research projects that have produced positive and tangible results (e.g., ESPRIT Statlog and METAL).

Model-based metafeatures

A different form of dataset characterization exploits properties of some induced hypothesis. As an example of this model-based approach, one can build a decision tree from a dataset and collect properties of the tree (e.g., nodes per feature, maximum tree depth, shape, tree imbalance, etc.), to form a set of metafeatures [22, 188].

Landmarkers

Finally, a different idea is to exploit information obtained from the performance of a set of simple and fast learners that exhibit significant differences in their learning mechanism [20, 190]. The accuracy of these so-called *landmarkers* is used to characterize a dataset and identify areas where each type of learner can be regarded as an expert [104, 237].

The measures discussed above can be used to identify a subset of accurate models by invoking a meta-level system that maps dataset characteristics to models. As an example, work has been done with the k-Nearest Neighbor method (k-NN) at the meta level to identify the most similar datasets for a given input dataset [41]. For each of the neighbor datasets, one can generate a ranking of the candidate models based on their particular performance (e.g., accuracy, learning time, etc.). Rankings can subsequently be aggregated to generate a final recommended ranking of models. More details on these issues are discussed in Chapters 2 and 3.

1.2.3 Employing Metalearning in KDD and Data Mining

KDD/DM process

The algorithm selection framework described above can be further generalized to the KDD/DM process. Consider again Figure 1.1, but this time assume that the output of the system is not a learning algorithm but a flexible planning system. The proposed extension can be justified as follows. Typically, the KDD process is represented in the form of a sequence of operations, such as data selection, preprocessing, model building, and post-processing,

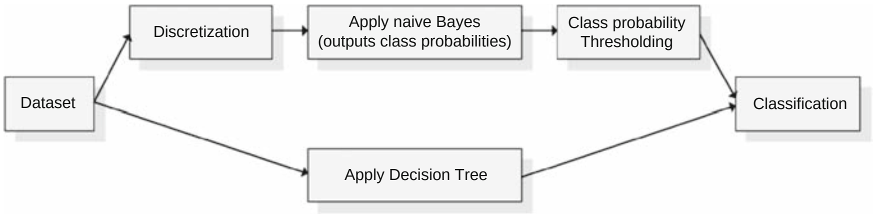
Discretization
Apply naive Bayes (outputs class probabilities)
Class probability Thresholding
Dataset
Classification
Apply Decision Tree

Fig. 1.2. Example of a partial order of operations (plan)

among others. Individual operations can be further decomposed into smaller operations. Operations can be characterized as simple sequences, or, more generally, as partially ordered acyclic graphs. An example of a simple partial order of operations is shown in Figure 1.2 (this example has been borrowed and adapted from [24]). Every partial order of operations can be regarded as an executable plan. When executed, the plan produces certain effects (for instance, classification of input instances).

Partial order of operations

Plan

Under this extended framework, the task of the data miner is to elaborate a suitable plan. In general the problem of generating a plan may be formulated as that of identifying a partial order of operations, so as to satisfy certain criteria and (or) maximize certain evaluation measures. Producing good plans is a non-trivial task. The more operations there are, the more difficult it is to arrive at an optimal (or near-optimal) solution.

A plan can be built in two ways. One is by placing together individual constituents, starting from an empty plan and gradually extending it through the composition of operators (as in [24]). Another possibility is to consider previous plans, identify suitable ones for a given problem, and adapt them to the current situation (e.g., see [176]).

Although any suitable planning system can be adopted to implement these ideas, it is clear that the problem is inherently difficult. One needs to consider many possible operations, some of them with high computational complexity (e.g., training a classifier on large datasets). Metaknowledge can be used to facilitate this task. Existing plans can be seen as embodying certain procedural metaknowledge about the compositions of operations that have proved useful in past scenarios. This can be related to the notion of macro-operators in planning. Knowledge can also be captured about the applicability of existing plans to support reuse. Finally, one can also try to capture knowledge describing how existing plans can be adapted to new circumstances. Many of these issues are discussed in Chapter 4.

1.2.4 Employing Metalearning to Combine Base-Level ML Systems

A variation on the theme of combining DM operations, discussed in the previous section, is found in the work on model combination. By drawing on information about base-level learning, in terms of the characteristics of either

Model combination

Composite learning systems

various subsets of data or various learning algorithms, model combination seeks to build composite learning systems with stronger generalization performance than their individual components. Examples of model combination approaches include boosting, stacked generalization, cascading, arbitrating and meta-decision trees.

Because it uses results at the base level to construct a learner at the meta level, model combination may clearly be regarded as a form of metalearning. Although many approaches focus exclusively on using such metalearning to achieve improved accuracy over base-level learning, some of them offer interpretable insight into the learning process by deriving explicit metaknowledge in the combination process. Model combination is the subject of Chapter 5.

1.2.5 Control of the Learning Process and Bias Management

We have discussed the issue of how metaknowledge can be exploited to facilitate the process of learning (Figure 1.1). We now consider situations where the given dataset is very large or potentially infinite (e.g., processes modeled as continuous data streams).

We can distinguish among several situations. For example, consider the case where the dataset is very large (but not infinite). Assume we have already chosen a particular ML algorithm and the aim is to use an appropriate strategy to mitigate the large dataset problem. Different methods are described in the literature to cope with this problem. Some rely on data reduction techniques, while others provide new functionalities on existing algorithms [99].

Active learning

One well-known strategy relies on *active learning* [281] in which examples are processed in batches: the initial model (e.g., a decision tree) is created from the first batch and, after the initial model has been created, the aim is to select *informative examples* from the next batch while ignoring the rest.

Controlling learning

The idea of controlling the process of learning can be taken one step further. For example, metalearning can be done dynamically, where the characterization of a new dataset is done progressively, testing different algorithms on samples of increasing size. The results in one phase determine what should be done in the next. The aim is to reduce the bias error (by selecting the most appropriate base-algorithm) effectively.

Learning from data streams

Another example involves learning from data streams. Work in this area has produced a control mechanism that enables us to select different kinds of learning system as more data becomes available. For instance, the system can initially opt for a simple naïve bayes classifier, but, later on, as more data becomes available, switch to a more complex model (e.g., bayesian network[2]).

In Section 1.2.1, we saw how data characteristics can be used to preselect a subset of suitable models, thus reducing the space of models under consideration. In learning from data streams, the control mechanism is activated in

[2] The description of naïve bayes and bayesian networks can be found in many books on machine learning. See, e.g., [174].

a somewhat different way. The quantity of data and data characteristics are used to determine whether the system should continue with the same model or take corrective action. If a change of model appears necessary, the system can extend the current model or even relearn from scratch (e.g., when there is a concept shift). Additionally, the system can decide that a switch should be made from one model type to another. More details on these issues can be found in Chapter 6.

1.2.6 Transfer of (Meta)Knowledge Across Domains

Transfer of knowledge

Another interesting problem in metalearning consists of finding efficient mechanisms to transfer knowledge across domains or tasks. Under this view, learning can no longer be simply seen as an isolated task that starts accumulating knowledge afresh on every new problem. As more tasks are observed, the learning mechanism is expected to benefit from previous experience. Research in *inductive transfer* has produced multiple techniques and methodologies to manipulate knowledge across tasks [192, 258]. For example, one could use a *representational transfer* approach where knowledge is first generated in one task, and subsequently exploited to help in another task. Alternatively one can use a *functional transfer* approach where various tasks are learned simultaneously; the latter case is exemplified in what is known as *multitask learning*, where the output nodes in a multilayer network represent more than one task and internal nodes are shared by different tasks dynamically during learning [50, 51].

Metalearner

In addition, the theory of metalearning has been enriched with new information quantifying the benefits gained by exploiting previous experience [16]. Classical work in learning theory bounding the true risk as a function of the empirical risk (employing metrics such as the Vapnik-Chervonenkis dimension) has been extended to deal with scenarios made of multiple tasks. In this case the goal of the metalearner is to output a hypothesis space with a learning bias that generates accurate models for a new task. More details concerning this topic are given in Chapter 7.

1.2.7 Composition of Complex Systems and Applications

Composition of complex systems

An attractive research avenue for future knowledge engineering is to employ ML techniques in the construction of new systems. The task of inducing a complex system can then be seen as a problem of inducing the constituting elements and integrating them. For instance, a text extraction system may be composed of various subsystems, one oriented towards tagging, another towards morphosyntactic analysis and yet another towards word sense disambiguation, and so on. This idea is somewhat related to the notion of layered learning [243, 270].

If we use the terminology introduced earlier, we can see this as a problem of planning to resolve multiple (interacting) tasks. Each task is resolved using a certain ordering of operations (Section 1.2.3). Metalearning here can help in retrieving previous solutions conceived in the past and reusing them in new settings. More details concerning this topic are given in Chapter 8.

1.3 Definition, Scope, and Organization

We have introduced the main ideas related to the field of metalearning covered by this book. Our approach has been motivated by both practical and theoretical aspects of the field. Our aim was to present the reader with diverse topics related to the term metalearning. We note that different researchers hold different views of what the term metalearning exactly means. To clarify our own view and to limit the scope of what is covered in this book, we propose the following definition:

> Metalearning is the study of principled methods that exploit metaknowledge to obtain efficient models and solutions by adapting machine learning and data mining processes.

Definition of metalearning

Our definition emphasizes the notion of metaknowledge. We claim a unifying point in metalearning lies in how to exploit such knowledge acquired on past learning tasks to improve the performance of learning algorithms. The answer to this question is key to the advancement of the field and continues being the subject of intensive research.

The definition also mentions machine learning processes; each process can be understood as a set of operations that form a learning mechanism. In this sense, a process can be a preprocessing step to learning (e.g., feature selection, dimensionality reduction, etc.), an entire learning algorithm, or a component of it (e.g., parameter adjustment, data splitting, etc.). The process of adaptation takes place when we replace, add, select, remove or change an existing operation (e.g., selecting a learning algorithm, combining learning algorithms, changing the value for a capacity control parameter, adding a data preprocessing step, etc.). The definition is then broad enough to capture a large set of possible ways to adapt existing approaches to machine learning.

The last goal is to produce efficient models under the assumption that bias selection is improved when guided by experience gained from past performance. A model will often be predictive in that it will be used to predict the class of new data instances, but other types of models (e.g., descriptive ones) will also be considered.