What Works Well Where in Inductive Learning?

Ricardo Vilalta VILALTA@US.IBM.COM
Irina Rish RISH@US.IBM.COM
IBM T.J. Watson Research Center, 30 Saw Mill River Rd., Hawthorne, N.Y., 10532 USA

Daniel Oblinger OBLINGER@US.IBM.COM
IBM T.J. Watson Research Center, P.O. Box 218, Yorktown Heights, N.Y., 10598 USA

1. Introduction

Despite successful applications of machine learning in industry and science, the reasons explaining why a learning algorithm is more successful than others on particular application domains remain elusive. A fundamental question awaits unanswered: what works well where? Learning algorithms are commonly compared by averaging their performance (e.g., off-training set accuracy) over benchmark domains. But few researchers have looked into a theory explaining why a learning algorithm performs well or not on specific domains. Such step is essential for the advancement of the field.

The design and implementation of learning algorithms will continue with a generate-and-test approach unless a clear set of guidelines is produced. We need a body of knowledge in the form of theories and laws amenable to revision through both experimentation and the proposal of simpler and more accurate explanations of performance. If a theory exist, it would guide the design of learning algorithms on a more principled fashion.

The goal of the workshop What Works Well Where?, as part of the 17th International Conference on Machine Learning, is to bring researchers together to present current work towards a theory of performance, and to propose new avenues of research. We hope to be stepping forward on a promising direction that will produce theories explaining the intricacies of inductive learning.

2. A Theory of Learning Performance

The study of learning performance divides on at least three areas: domain characterization, a functional decomposition analysis of learning algorithms, and a mathematical framework of inductive learning. The order is relevant: each area lays the foundations for the next. We analyze each area in turn.

2.1 Domain Characterization

First, we need a rich characterization of domains. A proper characterization of domains can explain learning performance by identifying the degree of match between the kind of domain (i.e., the kind of input-output distribution) and the inductive bias of the algorithm under analysis. Current domain characteristics are limited to simple measures (no. of features, no. of examples) and statistical/information-theoretic measures. But domain characteristics should be designed explicitly to validate a theory, which may require totally new measures. For example, if the performance of algorithm A is believed to attain certain level when the input-output distribution has a few dense clusters of examples and large inter-cluster distances, then appropriate definitions should be designed to measure the existence of such conditions to validate the theory. Domain characteristics must serve to demonstrate the validity of a theory and not vice versa.

2.2 Functional Decomposition Analysis

Second, we need to scrutinize the interior of learning algorithms, and perform a functional decomposition analysis to understand the effects of learning components. We need to characterize algorithms from a functional point of view (e.g., a recursive partitioning of the instance space vs voting among neighboring examples), looking to the internal components. Most empirical comparisons of learning algorithms show no distinction among the internal components of an algorithm, and hence of the contribution of each component during learning. The particular bias bestowed on the algorithm is assumed to pervade the entire mechanism; the effects of the components embedded in the algorithm tend to be averaged altogether. Controlled experiments can be devised to assess the performance

of a learning algorithm when a component is present or not, or when a component is modified in certain ways. These experiments must take into account the characteristics of the domain, thus the importance of the first area (Section 2.1).

2.3 A Mathematical Framework

Third, we need a mathematical framework in support of the inductive-learning edifice. Once it becomes clear how learning components interact and what their effect is during learning according to the domain under analysis, one can propose a mathematical model capturing the essence of such process. A mathematical model would indicate what to expect when a learning algorithm, configured in a specific way, is applied to a particular domain. At that point, the design of learning algorithms would have a solid foundation in the form of guiding principles, which would ultimately lead to improved learning performance.

Current mathematical models of induction (e.g., the *PAC* or Probably Approximately Correct model) lack the depth and scope necessary for a theory of performance: a worst-case analysis considers all possible distributions, and ignores the behavior of learning algorithms under specific circumstances. In addition, the definition of true error as a loss function over all examples gives equal weight to the training and off-training sets, and fails to recognize the off-training set error as the only measure of true importance.

3. Contributions to the Workshop

Some of the key areas for the development of a theory of performance have already been addressed by several researchers in machine learning. Within this workshop, Thorsten Joachims, in *High Dimensional Feature Spaces in Text Classification: Curse or Blessing?*, relates properties of text classification tasks to the performance of support vector machines and derives an upper bound of generalization error according to such properties. Chun-Nan Hsu in *Why Discretization Works for Naive Bayesian Classifiers* looks to Dirichlet distributions to explain why and when discretization of continuous variables is effective for Bayesian classifiers.

Others have followed a meta-learning approach by automating the generation of rules relating domain characteristic with learning performance. For example, Oscar Ortega Lobo and Masayuki Numao, in *On the Applicability of a Machine Learning Method for Estimating Missing Values*, try to find patterns associating the performance of algorithms that estimate missing values with the characteristics (based on mutual information) of the data. On the same vein, Stephen D. Bay and Michael J. Pazzani in *Characterizing Model Performance in the Feature Space* try to automate the characterization of model errors (or differences) in the different regions or areas of the feature space. Finally Pavel B. Brazdil and Carlos Soares in *Zoomed Ranking: Selection of Classification Algorithms Based on Relevant Performance Information* rank algorithms by first looking to datasets similar to the one under analysis using domain characteristics, and use that as information prior to ranking.

A mathematical framework for learning is advocated by David Wolpert in *Any Two Learning Algorithms Are (Almost) Exactly Identical* where it is made clear that, unless one restricts the set of possible target functions to a small set, any pair of algorithms perform almost identically. The main goal is to find a model-free geometry of learning. Tobias Scheffer in *Predicting the Relation Between Model Class, Domain, and Error Rate* analyzes generalization error assuming the existence of a histogram of error rates for a given model class.

In the realm of ensembles of classifiers, Pedro Domingos in *Why do Model Ensembles Work?* and Phil Rennert in *What Works Well Anywhere* investigate the differences in performance between individual learning algorithms vs combinations of them.

Pat Langley in *Average-Case Analyses of Induction* tries to estimate classification accuracy through average case analysis. Seishi Okamoto does a similar task by looking to domain characteristics in *Average-Case Analysis of the Learning Behavior*.

4. Conclusions

Although significant work is currently directed towards a theory of performance, a general framework is necessary to unify efforts. For example, Section 3 illustrates work using domain characterization and mathematical models, but few has been said on the effect of the several internal functions of learning algorithms (Section 2.2). A unification of efforts is necessary to integrate all work into a single structure that can serve to support a theory of learning performance.

Acknowledgments

We thank Pat Langley and Joseph Hellerstein for their valuable assistance. This work was supported by IBM T.J. Watson Research Center.