Machine Learning Theory

Avrim Blum[*]
Department of Computer Science
Carnegie Mellon University

Abstract

Machine Learning Theory, also known as Computational Learning Theory, aims to understand the fundamental principles of learning as a computational process and combines tools from Computer Science and Statistics. This essay is intended to give a very brief introduction to the area, some of its past successes, and some of its current challenges and potential future directions.

What is Machine Learning?

The area of *Machine Learning* deals with the design of programs that can learn rules from data, adapt to changes, and improve performance with experience. In addition to being one of the initial dreams of Computer Science, Machine Learning has become crucial as computers are expected to solve increasingly complex problems and become more integrated into our daily lives.

Writing a computer program is a bit like writing down instructions for an extremely literal child who just happens to be millions of times faster than you. Yet many of the problems we now want computers to solve are no longer tasks we know how to explicitly tell a computer how to do. These include identifying faces in images, autonomous driving in the desert, finding relevant documents in a database (or throwing out irrelevant ones, such as spam email), finding patterns in large volumes of scientific data, and adjusting internal parameters of systems to optimize performance. That is, we may ourselves be good at identifying people in photographs, but we do not know how to directly tell a computer how to do it. Instead, methods that take labeled training data (images labeled by who is in them, or email messages labeled by whether or not they are spam) and then learn appropriate rules from the data, seem to be the best approaches to solving these problems. Furthermore, we need systems that can adapt to changing conditions, that can be user-friendly by adapting to needs of their individual users, and that can improve performance over time.

[*] I would like to thank Dick Karp, Neha Dave, Michael Kearns, and Tom Mitchell for comments and suggestions on earlier drafts of this essay.

What is Machine Learning Theory?

Machine Learning Theory, also known as *Computational Learning Theory*, aims to understand the fundamental principles of learning as a computational process. This field seeks to understand at a precise mathematical level what capabilities and information are fundamentally needed to learn different kinds of tasks successfully, and to understand the basic algorithmic principles involved in getting computers to learn from data and to improve performance with feedback. The goals of this theory are both to aid in the design of better automated learning methods *and* to understand fundamental issues in the learning process itself.

Machine Learning Theory draws elements from both the Theory of Computation and Statistics and involves tasks such as:

- Creating mathematical models that capture key aspects of machine learning, in which one can analyze the inherent ease or difficulty of different types of learning problems.
- Proving guarantees for algorithms (under what conditions will they succeed, how much data and computation time is needed) and developing machine learning algorithms that provably meet desired criteria.
- Mathematically analyzing general issues, such as: "Why is Occam's Razor a good idea?", "When can one be confident about predictions made from limited data?", "How much power does active participation add over passive observation for learning?", and "What kinds of methods can learn even in the presence of large quantities of distracting information?".

A Few Highlights

Consider the general principle of "Occam's razor", that simple explanations should be preferred to complex ones. There are certainly many reasons to prefer simpler explanations — for instance, they are easier to understand — but can one mathematically argue for some form Occam's razor from the perspective of performance? In particular, should computer programs that learn from experience use some notion of the Occam's razor principle, and how should they measure simplicity in the first place?

One of the earliest results in Computational Learning Theory is that there is indeed a reason as a policy to seek out simple explanations when designing prediction rules. In particular, for measures of simplicity including description length in bits, *Vapnik-Chervonenkis dimension* which measures the effective number of parameters, and newer measures being studied in current research, one can convert the level of simplicity into a degree of confidence in future performance. While some of these theoretical results are quite intricate, at a high level the intuition is just the following: there are many more complicated explanations possible than simple ones. Therefore, if a simple explanation happens to fit your data, it is much less likely this is happening just by chance. On the other hand, there are so many complicated explanations possible that even a large amount of data is unlikely to rule all of them out, and even some that have nothing to do with the task at hand are likely to still survive and fool your system. This intuition can then be turned into mathematical guarantees that can guide machine learning algorithms.

Another highlight of Computational Learning Theory is the development of algorithms that are able to quickly learn even in the presence of large amounts of distracting information. Typically, a machine learning algorithm represents its data in terms of *features*: for example, a document might be represented by the set of words it contains, and an image might be represented by a list of various properties it has. The learning algorithm processes this information to make some prediction (Is this document of interest? Who is the person in this image?). However, it is up to the algorithm *designer* to decide on what these basic features should be, and the designer may not know in advance what features will turn out to be the most useful. Thus, one would like the designer to be able to pour as many features as possible into the learning algorithm and have the algorithm itself quickly focus in on those that are actually needed. An exciting early result in Computational Learning Theory was the development of algorithms that in many cases have provably only a *logarithmic* dependence in their convergence rate on the number of distracting features: this means that every time you double the amount of information available, it at worst can hurt the algorithm by a small additive amount. So, designers do not have to be stingy in providing information to the algorithm. Moreover, recently there has been substantial work on how learning algorithms can automatically change their input representation through what are known as *kernel functions*, which themselves can be learned from data.

Machine Learning Theory also has a number of fundamental connections to other disciplines. In cryptography, one of the key goals is to enable users to communicate so that an eavesdropper cannot acquire any information about what is being said. Machine Learning can be viewed in this setting as developing algorithms for the eavesdropper. In particular, provably good cryptosystems can be converted to problems one cannot hope to learn, and hard learning problems can be converted into proposed cryptosystems. Moreover at the technical level, there are strong connections between important techniques in Machine Learning and techniques developed in Cryptography. For example, *Boosting*, a machine learning method designed to extract as much power as possible out of a given learning algorithm, has close connections to methods for amplifying cryptosystems developed in cryptography.

Machine Learning Theory also has close connections to issues in Economics. Machine learning methods can be used in the design of auctions and other pricing mechanisms with guarantees on their performance. Adaptive machine learning algorithms can be viewed as a model for how individuals can or should adjust to changing environments. Moreover, the development of especially fast-adapting algorithms sheds light on how approximate-equilibrium states might quickly be reached in a system, even when each individual has a large number of different possible choices. In the other direction, economic issues arise in Machine Learning when not only is the computer algorithm adapting to its environment, but it also is affecting its environment and the behavior of other individuals in it as well. Connections between these two areas have become increasingly strong in recent years as both communities aim to develop tools for modeling and facilitating electronic commerce.

Future Directions

Research in Machine Learning Theory is a combination of attacking established fundamental questions, and developing new frameworks for modeling the needs of new machine learning applications. While it is impossible to know where the next breakthroughs will come, a few topics one can expect the future to hold include:

- Better understanding how auxiliary information, such as unlabeled data, hints from a user, or previously-learned tasks, can best be used by a machine learning algorithm to improve its ability to learn new things. Traditionally, Machine Learning Theory has focused on problems of learning a task (say, identifying spam) from labeled examples (email labeled as spam or not). However, often there is additional information available. One might have access to large quantities of unlabeled data (email messages not labeled by their type, or discussion-group transcripts on the web) that could potentially provide useful information. One might have other hints from the user besides just labels, e.g., highlighting relevant portions of the email message. Or, one might have previously learned similar tasks and want to transfer some of that experience to the job at hand. These are all issues for which a solid theory is only beginning to be developed.

- Further developing connections to economic theory. As software agents based on machine learning are used in competitive settings, "strategic" issues become increasingly important. Most algorithms and models to date have focused on the case of a single learning algorithm operating in an environment that, while it may be changing, does not have its own motivations and strategies. However, if learning algorithms are to operate in settings dominated by other adaptive algorithms acting in their own users' interests, such as bidding on items or performing various kinds of negotiations, then we have a true merging of computer science and economic models. In this combination, many of the fundamental issues are still wide open.

- Development of learning algorithms with an eye towards the use of learning as part of a larger system. Most machine learning models view learning as a standalone process, focusing on prediction accuracy as the measure of performance. However, when a learning algorithm is placed in a larger system, other issues may come into play. For example, one would like algorithms that have more powerful models of their own confidence or that can optimize multiple objectives. One would like models that capture the process of deciding *what* to learn, in addition to *how* to learn it. There has been some theoretical work on these issues, but there is certainly is much more to be done.

Conclusions

Machine Learning Theory is both a fundamental theory with many basic and compelling foundational questions, and a topic of practical importance that helps to advance the state of the art in software by providing mathematical frameworks for designing new machine learning algorithms. It is an exciting time for the field, as connections to many other areas are being discovered and explored, and as new machine learning applications bring new questions to be modeled and studied. It is safe to say that the potential of Machine Learning and its theory lie beyond the frontiers of our imagination.