Advanced Modeling Methods

MACHINE LEARNING MEETS AGENT-BASED MODELING: WHEN NOT TO GO TO A BAR

W. RAND,* Northwestern University, Evanston, IL

ABSTRACT

One of the promises of ABM is the ability to have adaptive agents make decisions in changing environments. Though great work has been done using adaptive agents in ABM, more research into the theoretical understanding of these systems would be useful. Adaptive agents have already been studied within machine learning (ML) — an area of artificial intelligence specifically concerned with adaptation and building internal models. The first part of this paper presents a framework for understanding ML as a component of ABM, and describes how different ML techniques can be incorporated into some ABMs. At the high level this framework consists of two cycles that involve evaluating input, making decisions and then generating output. Within this generalized framework, the ML algorithm is using the ABM as an environment and a reward generator, while the ABM is using the ML algorithm to refine the internal models of the agents. There are many details that must be answered before any ML technique can be incorporated into an ABM. In this paper I start to explore some guidelines for how to more closely integrate ABM and ML and will discuss complications that arise when combining ABM and ML techniques. To illustrate some of these issues, I will describe an integration of a ML technique within the El Farol Bar Problem. I will conclude with some discussion of this integration and a look toward future research.

Keywords: Machine learning, agent-based modeling, framework El Farol Bar Problem, genetic algorithms

INTRODUCTION

As we pause to reflect on how agent-based modeling (ABM) has changed in the ten years since SugarScape (Epstein and Axtell 1996), one aspect of ABM that could use more analysis is adaptation. Though there are notable exceptions like the El Farol Bar Problem (Arthur 1994) among others, few models make use of an adaptive mechanism within the ABM framework. By an adaptive mechanism, I refer not to the ability of agents to take different actions, but rather the ability for agents to come up with a new strategy of how to take action. This is particularly surprising since the ability to allow agents to adapt to their surrounding is often listed as a reason to use ABM instead of other modeling techniques. When Holland discussed complex adaptive systems (CAS) and their relationship to ABM in *Hidden Order* (Holland 1995), he devoted an entire chapter to adaptive agents, and specifically mentioned internal models as one of the mechanisms that define a CAS. Despite this more effort needs to be placed into understanding adaptive agents. However, as we examine the last ten years of ABM it is important to not only notice its deficiencies but also to see how these areas can be improved. In a fortuitous

* *Corresponding author address*: William Rand, Northwestern Institute on Complex Systems, 600 Foster Street, Evanston, IL 60208-4057; e-mail: wrand@northwestern.edu.

Paper extracted from *Proceedings of the Agent 2006 Conference on Social Agents: Results and Prospects*, ANL/DIS-06-7, ISBN 0-9679168-7-9, D. Sallach, C.M. Macal, and M.J. North (editors), co-sponsored by Argonne National Laboratory and The University of Chicago, September 21–23, 2006.

coincidence, at the same time ABM research has been gathering momentum, so has machine learning — an area of artificial intelligence specifically concerned with adaptation and building internal models. If the ABM community can make use of the knowledge and research developed by the machine learning community, it would greatly facilitate the study of adaptation within ABM.

ABM and ML can be combined in a variety of ways and has been examined in the past (Wolpert, Wheeler et al. 1999). However, in this paper I choose to examine the use of ML to refine the internal models of agents in an ABM. The first part of this paper presents a framework for understanding machine learning as a component of ABM, and describes how different machine learning techniques like genetic algorithms (GAs), neural nets (NNs), and Bayesian Classifiers can easily be incorporated into many agent-based models. At the high level this framework consists of two interlocked cycles that examine input, make decisions and generate output. In this generalized framework, the machine learning algorithm is using the ABM as an environment and a reward generator, while the ABM is using the machine learning algorithm to maintain the internal models of the agents.

There are many details that must be answered before any machine learning technique can be incorporated into even the simplest agent-based model. In this paper, some of these details of how to build this general framework are discussed. Of course even after a general framework has been decided upon there are many more questions that still need to be answered, like what particular technique to use, and how to set the parameters of that technique. This paper will discuss these complications.

The final section of this paper will illustrate some of these issues with an example. This practical example will consist of a genetic algorithm implemented within the context of the El Farol Bar Problem. The design of such an implementation and the consideration of the various issues involved will be discussed.

THE FRAMEWORK

At a high level ABM and ML both utilize fairly simple algorithmic structures to control their flow of operation. Roughly these algorithms can be described as: initialize the system, observe what is happening, refine the system, take actions, and repeat until time is up. To give this high level description more context I will first discuss the *ABM cycle*, then the *ML cycle* and finally an *integrated cycle*.

The first cycle is the standard agent-based model cycle and can be broken down into three steps: (1) initialize the world and a population of agents, (2) each agent observes its world, and (3) each agent takes an action based on the current observations, and the model repeats by going back to (2). This cycle becomes an *adaptive* agent-based model if we incorporate a fourth step between (2) and (3) where each agent updates their internal model of the world, and decides what action today based on that internal model. The adaptive ABM cycle is illustrated in Figure 1.

The second distinct cycle, as seen in Figure 2, is the machine learning cycle and can be broken down into four steps as well: (1) create an initial internal model, (2) observe the world

Paper extracted from *Proceedings of the Agent 2006 Conference on Social Agents: Results and Prospects*, ANL/DIS-06-7, ISBN 0-9679168-7-9,
D. Sallach, C.M. Macal, and M.J. North (editors), co-sponsored by Argonne National Laboratory and The University of Chicago, September 21–23, 2006.

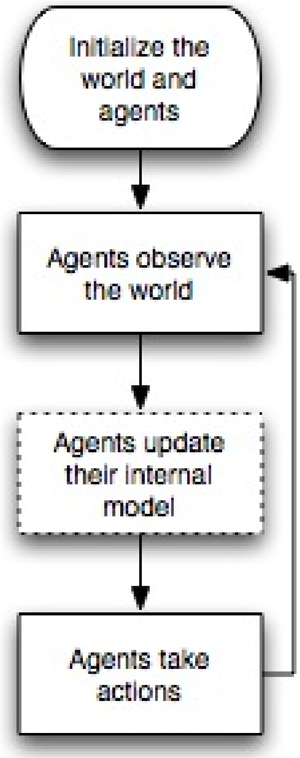

Figure 1 ABM cycle

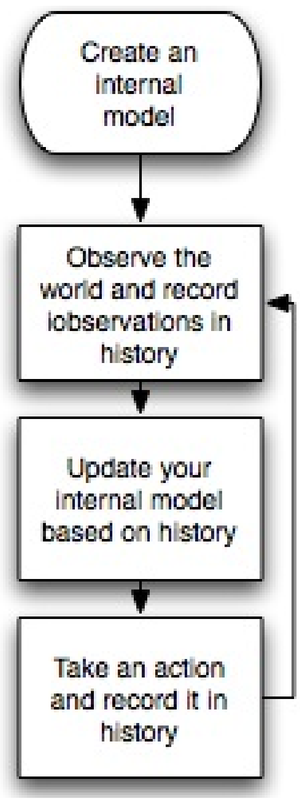

FIGURE 2 ML Cycle

Paper extracted from *Proceedings of the Agent 2006 Conference on Social Agents: Results and Prospects*, ANL/DIS-06-7, ISBN 0-9679168-7-9,
D. Sallach, C.M. Macal, and M.J. North (editors), co-sponsored by Argonne National Laboratory and The University of Chicago, September 21–23, 2006.

and take note of rewards received, (3) update the internal model, (4) take an action based on the internal model and the current observations, go back to (2) and repeat.

As is obvious the two cycles are quite similar to each other and thus integrating the frameworks is not very difficult at all. However how this integration is practically accomplished could be done in many different ways. In this paper I have chosen to explore the use of the ML cycle as a model refinement engine for the ABM. Thus the integrated cycle focuses on the ABM and interrupts its standard flow in step three, by sending data to the ML cycle to handle the model refinement. This is illustrated in Figure 3.

PRACTICAL DECISIONS

Of course utilizing machine learning techniques within agent-based modeling is not as simple as describing the framework. There are many practical details that must be addressed when deciding how to integrate ABM and ML. One question that must be answered is whether the machine learning technique should be a supervised learning technique (an external teacher determines whether any action taken was correct or incorrect) or an unsupervised learning technique (agents take actions and occasionally gain rewards but there is not necessarily a chain of causation from any action to any reward). Supervised learning requires explicit knowledge of

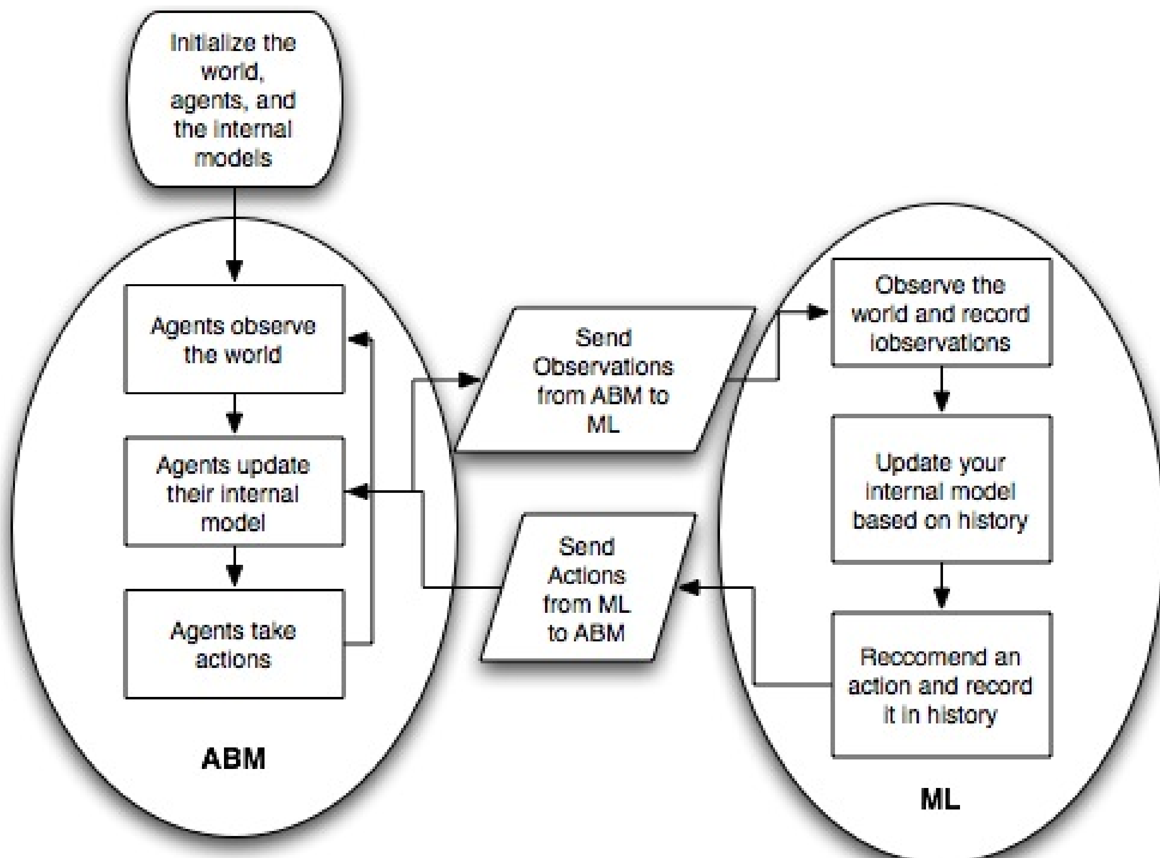

FIGURE 3 Integrated cycle

what actions provoked what rewards, i.e., a mapping of inputs and outputs. Unsupervised learning does not require this but instead simply builds a model of how the world behaves.

Another question that must be addressed is whether or not the agent's own action needs to be taken into account while building the internal model of the world. In many cases within ABM, agents assume that they can make decisions about the world as if they were not a part of the world. This is sometimes called the *Wonderful Life Assumption*[1] from the Frank Capra movie *It's A Wonderful Life* where James Stewart's character George Bailey wonders if anything he has done has made the world a better place. However in many cases like the El Farol Bar Problem, agents' actions do influence the world. In fact in the case of the El Farol Bar Problem, since the only variable of interest is the attendance at the bar, agents' actions basically define the world. On the other hand since agent's actions may have a minimal effect on the world it may be possible to make the Wonderful Life Assumption and safely ignore the agent's own action when building up an internal model. This may actually be the case in the El Farol Bar Problem since each agent only contributes 1/100 to the total attendance of the bar.

Even after the family of techniques has been decided upon, there are still many specific algorithms that are more or less useful and must be carefully considered. Neural networks for instance, are good at classifying large amounts of data fairly quickly, but in the end they do not yield white box results; that is after they have run for awhile it is very difficult to determine how they are making their decisions. Decision trees, on the other had, do create very white box results, but are not very good at classifying continuous data. There are a variety of books that discuss ML algorithms, their implementations and their pros and cons (Mitchell 1997; Hastie, Tibshirani et al. 2001).

Even after the particular technique, there are still a large variety of parameters that need to be set, and tuned in order to work properly within the ABM environment. Much of this is a matter of art to get the results one desires, but some sets of ML algorithms have more literature than others regarding advice on how to tune the parameters.

All in all, there are many matters to consider when combining ABM with ML, but the advantages that one gains from having truly adaptive agents, which can modify not only the actions they are taking but also the strategies that they use to determine those actions is often worthwhile.

A CASE STUDY: THE EL FAROL BAR PROBLEM

It is difficult to discuss many of these issues without a particular example to focus the discussion around. After all in the end a programmer or model builder must actually write some code to integrate ABM and ML. Thus, it is important to think of how these issues affect actual model development. In order to illustrate a few of these issues, I will consider one integration in specific details. I have chose to use the El Farol Bar Problem (Arthur 1994) as an example. This was an early ABM that included adaptive agents. In short, the model consists of a 100 agents trying to decide whether or not to attend a bar on a certain night. If they attend and the bar is

---

1 This is related to Wolpert et al's *wonderful life subworld utility* but here I am concerned with a particular subworld—the one where the agent does not account for their own action—and hence the assumption I am discussing is less general than Wolpert's utility function (Wolpert, Wheeler et al. 1999).

Paper extracted from *Proceedings of the Agent 2006 Conference on Social Agents: Results and Prospects*, ANL/DIS-06-7, ISBN 0-9679168-7-9,
D. Sallach, C.M. Macal, and M.J. North (editors), co-sponsored by Argonne National Laboratory and The University of Chicago, September 21–23, 2006.

crowded they receive no reward; if they stay home they receive no reward. However, if they attend and the bar is not crowded (less than 60 attendees in Arthur's model) they receive a reward. The only information they have to make this decision is the attendance of the bar the last week which is printed in the paper and can be remembered over time. So the question was: how do agents decide which strategy to use to determine whether to attend the bar or not?

Arthur's original model included a simple ML technique in it. In Arthur's model all the agents had a group of strategies. They would take this set of strategies and see which strategy would have done the best of predicting the bar attendance if they had used it in the past. Since at each time step a new data point is generated it is possible that the actual strategy from the group of strategies that each agent will use can change at every time. This is a very simple ML technique and can be described within the framework of Figure 2. Initialize your group of strategies by generating some random strategies, like take last week's attendance double it, or subtract the third to last week's attendance from last weeks, or take a running average of the last three weeks attendances (Step 1). Then at each time step observe how the strategies have done on the current set of training data, i.e. the previous bar attendances (Step 2). After that, refine your internal model by selecting the best one given the new data (Step 3). Finally act on the strategy that reflects your refined model (Step 4) and repeat (at Step 2). This ML technique could be used for other problems than the El Farol Bar Problem; for instance, Arthur's technique could be used to predict stock market price or estimate the rainfall in a certain geographic location. Thus Arthur's technique is not particular to the El Farol Bar Problem and could be replaced by any number of standard ML techniques.

I wanted to make use of a different ML technique than the one Arthur described. So first I had to decided whether to use a supervised or unsupervised learning technique. It might appear at first that it is necessary to use an unsupervised technique since an agent's action is not directly responsible for their reward. However since in Arthur's version of the problem the agents are not necessarily trying to maximize their utility but rather just trying to minimize their error of prediction and then take an action based upon that, we can assume that previous time data series is in fact a supervised training set. Given this assumption we can safely choose a supervised machine learning technique. Since supervised ML techniques tend to be faster than unsupervised techniques, in general when there is enough information available to classify the problem as a supervised problem it is helpful to utilize a supervised ML technique. However, it would also be possible to use an unsupervised learning method if someone simply wanted to build up a model of how attendances influenced each agent's rewards.

Of course it is also necessary to consider whether or not to model the agent's own action. However the way this problem has been framed for the agents, they are automatically not making the Wonderful Life Assumption. This is because agents are predicting the attendance at the bar and making a decision about whether or not they will attend the bar based on that prediction. Thus, it can be assumed that their prediction automatically takes into account their own decision. The agent is asking the ML technique to predict next week's attendance and is not putting any restrictions on their request; therefore the prediction should take into account whatever action the agent will take.

Second, it was necessary to choose a particular machine learning technique. There is no obvious decision here, but partially since it was originally suggested in Arthur's paper, I decided to investigate the use of the genetic algorithm (GA) as originally devised by Holland (Holland 1975). Fogel had previously explored such a technique within the El Farol Bar Problem (Fogel, Chellapilla et al. 1999). The GA makes sense in this context because it has the ability to create a fairly robust time series predictor (by doing simple regression) and it is similar to Arthur's original technique, in that it considers a population of solutions, evaluates them, decides which ones to keep using, changes them slightly and re-evaluates them. In addition the GA is often described as manipulating *schemata* and thus may be similar to the human process of induction (Holland, Holyoak et al. 1986) which is what Arthur's original model was intended to emulate. Clearly, then by examining the benefits of various ML algorithms and choosing the one that seemed to satisfy the task at hand I was able to choose a particular algorithm.

In order to integrate a GA within the El Farol Bar Problem I had to first place the original El Farol Bar Problem within the context of the Integrated cycle described above. Thus I filled out the left hand bubble with the details of El Farol Bar Problem. Then I filled out the right hand bubble with the details of the GA. The result is illustrated in Figure 4.

After I had visualized the integration I had to actually accomplish the task, which involved not only setting the parameters of the El Farol Bar Problem but also that of the GA of each agent. My over-riding goal in this task was to see if I could generate results similar to Arthur's original results. Thus I used a set of parameters similar to what Arthur had described for the El Farol Bar Problem. For the GA, I could have chose to use Fogel's parameters, but I decided that those were farther away from Arthur's original model than I wanted to deviate since the Fogel's parameters seemed to require a larger amount of computational resources than

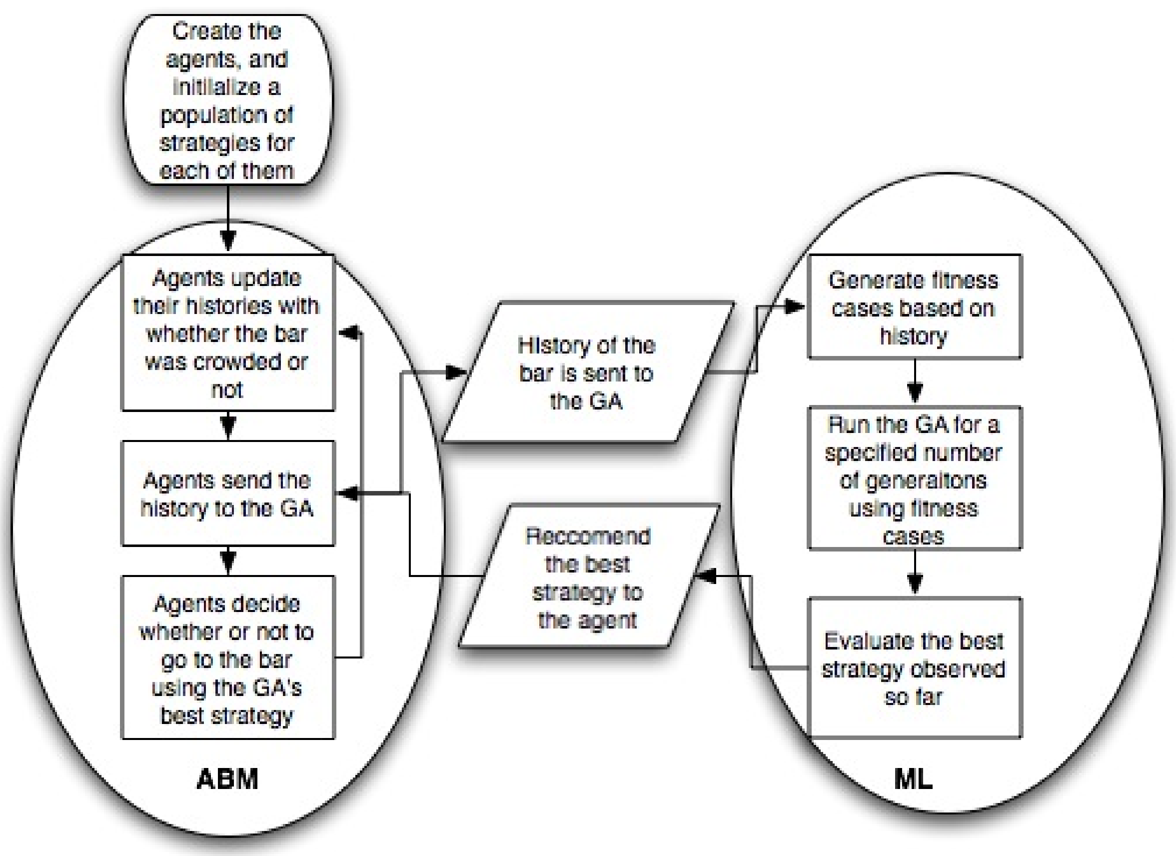

FIGURE 4 Example using El Farol

Arthur's original technique. Thus instead I decided to choose parameters for the GA that would more closely imitate Arthur's original model. In general, it is good to consider two factors when setting ML parameters: (1) review relevant literature on good parameter settings, and (2) keep your modeling goal in mind when setting parameters.

CONCLUSION

Over the last ten years, there have been a lot of exciting and interesting developments in ABM, and the use of truly adaptive agents within ABM is one stimulating promise that is still being explored. The integration of ML techniques within ABM will hopefully allow for the development of novel and original models. This paper has discussed a general framework for these combined algorithms, and has begun to discuss some of the issues that must be addressed. Finally one integration has been described and used as a case study to further discuss the issues.

In the future, this line of research will continue. The integration of the El Farol Bar Problem with a variety of ML techniques and a thorough analysis of the results is warranted. As is consideration of other ABMs and how to integrate them with ML algorithms. As more and more of these integrations are attempted it is hoped that a suite of best practices will develop that can serve as advice and eventually form the basis for a more concrete set of guidelines for future model developers. The scope and use of ABM will be greatly expanded by the increased use of strategically adaptive agents.

REFERENCES

Arthur, W. B. (1994). "Inductive Reasoning and Bounded Rationality." *The American Economic Review* **84**(2): 406-411.

Epstein, J. and R. Axtell (1996). *Growing Artificial Societies: Social Science from the Bottom Up*. Cambridge, MA, MIT Press.

Fogel, D. B., K. Chellapilla, et al. (1999). "Inductive Reasoning and Bounded Rationality Reconsidered." *IEEE Transactions on Evolutionary Computation* **3**(2): 142-146.

Hastie, T., R. Tibshirani, et al. (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, Springer.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*, MIT Press.

Holland, J. H. (1995). *Hidden Order: How Adaptation Builds Complexity*. Reading, Massachusetts, Perseus Books.

Holland, J. H., K. J. Holyoak, et al. (1986). *Induction*, MIT Press.

Mitchell, T. M. (1997). *Machine Learning*, McGraw-Hill.

Wolpert, D. H., K. R. Wheeler, et al. (1999). General Principles of Learning-Based Multi-Agent Systems. *Autonomous Agents*. Seattle, WA, ACM.

Paper extracted from *Proceedings of the Agent 2006 Conference on Social Agents: Results and Prospects*, ANL/DIS-06-7, ISBN 0-9679168-7-9,
D. Sallach, C.M. Macal, and M.J. North (editors), co-sponsored by Argonne National Laboratory and The University of Chicago, September 21–23, 2006.