Leopold-Franzens-Universität Innsbruck

Institute of Coputer Science

Computational Logic

Model Learning

Introduction to Scientific Working

Robert Wiklicky
Christoph Wirnsperger

Supervisor:
Cezary Kaliszyk

May 29, 2017

Contents

Abstract

This paper reviews the field machine learning and a specific instance of it, which is model learning. First, we will give a brief presentation of some machine learning wellsprings, take Artificial Intelligence and Brain Learning for an example. Then, after further introduction of the concepts of Adaptive Control Theory and the definitions of Blackbox Testing and Mealy machines, we will provide a descriptive demonstration to model learning by means of Angluin's algorithm. Being done analyzing these concepts, we finally present a concise discussion of real world fields of usage for model learning.

1 Introduction

Over the last few years the term "machine learning" and its impacts gained popularity, not only amongst computer scientists. Mostly this is due to the overwhelming increase in computing power we have experienced and are still experiencing. More popularly known under the term "Artificial Intelligence", machine learning instances like Apple's SIRI, various image recognition programs, intelligent advertisement systems and a lot more have caused a lot of "non-specialists" to jump on the machine learning bandwagon. But what actually do we mean, when we speak of machine learning? Learning itself has a broad range of processes that is difficult to define precisely. A dictionary definition includes phrases such as "to gain knowledge, or understanding of, or skill in, by study, instruction, or experience," and "modification of a behavioral tendency by experience." Speaking explicitly of machines, we can say that a machine learns whenever it changes its structure, program, or data (based on its inputs or in response to external information) in such a manner that its expected future performance improves [7]. In our article, we will propose a brief general overview over some wellsprings of machine learning. Then we will focus on the specific instance "Adaptive Control Theory" using deterministic Mealy machines and Angluin's algorithm. 5

2 Machine Learning Domains

2.1 Statistics

Being a little bit philosophical in his book "A Brief History of Time" [5], Stephen Hawking is outlining the fact that at some point in history somehow our universe was created. And that this universe contains an astonishing amount of regulation, meaning the apple fell of a tree a few centuries ago, when Newton discovered his infamous law of gravitation and it probably will still fall off today or tomorrow. Nowadays statistics makes use of this universal regulation by trying to predict the unknown outcomes of a set of input events. Assuming events which turn out to have a high probability of happening will very likely happen again in the future, machine learning algorithms are able to use statistics to learn about the sample environment and therefore make smart interpretations out of input events.

2.2 Brain Models

The idea of finding a way to artificially mimic the human brain probably has been around for a lot of time. Still it was Warren McCulloch and Walter Pitts who first tried an approach in the field of "Neural Networks" [10] in 1943. Because until the late 2000s computing power was not enough to efficiently simulate Neural Networks, the idea fell behind a little bit. The last few years however, Neural Networks gained an increasing amount of momentum again in research. Basic concept of neural networks is the neuron. The neuron takes an arbitrary amount of inputs, sums and weighs these inputs before sending the result through a non-linear function producing the final output of the neuron. If you put a lot of neurons together, where the output of a neuron can be the input of another neuron, you create a neural network. These networks do not impose the actual biological constraints of a human brain, yet are highly capable of solving a various amount of learning problems [1].

2.3 Artificial Intelligence

"The history of AI is a history of fantasies, possibilities, demonstrations, and promise." Although humans probably have dreamed of intelligent assistants ever since, only in the last half century mankind was able to make relevant progress in this area [2]. It can be quite hard to really classify the phrase "Artificial Intelligence", since apparently AI has a specific definition for many people that do not necessarily coincide with the definition of some other people [8]. Roger Schank tried to specify the four most general viewpoints on AI [8], with one of them being that AI means having a machine learn. He states that intelligence entails learning and thus getting better over time. A dog that fails to understand his environment does not exhibit any intelligent behavior, neither does a static machine that is not able to learn from its mistakes over time.

2.4 Adaptive Control Theory

Last but not least we want to introduce the domain of Adaptive Control Theory, which will be the domain of the machine learning concept we will introduce later on. The basic idea of ACT is to create a closed loop controller with parameters that can be updated to change the response of the system. The output of the response is then compared to a desired response from a reference model. Based on this comparison error the control values are updated. Goal is to minimize the error between system response and desired response [2]. One important thing to note is that adaptive control is different from robust control in that it does not need "a priori" information about the bounds of these uncertain or time-varying parameters.

[1] `http://ufldl.stanford.edu/tutorial/supervised/MultiLayerNeuralNetworks/`
[2] `http://www.pages.drexel.edu/~kws23/tutorials/MRAC/MRAC.html`

3 Blackbox Testing

Before we finally introduce the concept of model learning, there are two more things we need to discuss. One of them is "Blackbox Testing". The general idea behind Blackbox Testing, or sometimes also called "Behavioral Testing", is that the tester does not know anything about the internal structure of the item being tested. Thus the tester can only provide inputs and observe the corresponding outputs of the tested system. This makes it easier to concentrate on what the software does, not how it does it. Additionally there may be a functional testing approach, which is concerned with what the system does and a non-functional testing approach, which is concerned with how well the system does [3].

4 Mealy Machines

Still missing now is the definition of the Mealy machine. A deterministic Mealy machine is a finite state machine, whose output values are determined both by its current state and its current inputs. A mealy machine is also a finite-state transducer, which means for each state and input, at most one transition is possible. A more formal way of describing Mealy machines is considering them as a 6-tuple $M = (I, O, Q, q_0, \gamma, \lambda)$. I is a finite set of inputs, O is a finite set of outputs, Q is a finite set of states, $q0 \in Q$ is the initial state. When a state receives an input, it will transform to another state. This transformation is given by the transition function gamma: $Q * I- > Q$. Also a state will produce an output after receiving an input. This output is specified by the output function lambda: $Q * I- > O$ [6]. To get a clearer understanding, we provide a graphic representation of a simple Mealy machine with inputs a, b, outputs A, B, C, states q_0, q_1, q_2 and initial state q_0 [9].

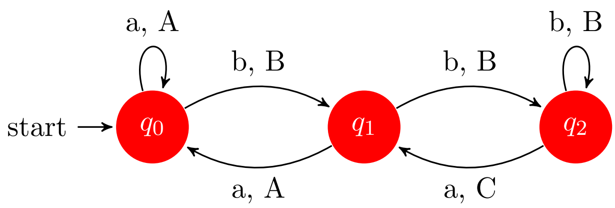

Fig. 1: Simple Mealy machine with three different states

Model learning aims to construct Blackbox state diagram models of software and hardware systems by providing inputs and observing outputs. Recently,

[3] http://istqbexamcertification.com/what-is-black-box-specification-based-also-known-as-behavioral-testing-techniques/

much progress has been made in the design of new algorithms, both in a setting of finite state diagrams (Mealy machines) and in richer settings with data (register automata). Through the use of abstraction techniques, these algorithms can be applied to complex systems [9]. We will now look into further detail of one of those algorithms.

5 Angluin's Algorithm

The number of most learning algorithms grows linearly with the number of inputs and quadratically with the number of states. This means formulating new hypothesis is rather easy, whereas checking a hypothesis' correctness quickly becomes a very computational expensive task for inputs with large numbers. The worst case scenario for testing contains all possible sequences and sub-sequences of inputs, which is the exponential of inputs correspondingly. Therefore we are looking for methods to decrease the number possibilities that should be checked.
Angluin's Algorithm is a very famous algorithm for automate assume-guarantee reasoning. It is also called L* Algorithm. This algorithm iteratively builds a DFA (deterministic finite automaton) which can be represented in a table with rows and columns. Two properties can occur in this algorithm, namely closedness and consistency. For every iteration one of the following steps will be taken:

- **Closed but not consistent:** The top rows have two states, where the rows are identical but not the corresponding states.
- **Consistent but not closed:** If the bottom row is different to the top row it promotes this row to the top row.
- **Closed and consistent:** The algorithm immediately constructs a DFA. If it receives the answer "yes" then the algorithm successfully terminates. If it receives the answer "no" it adds the received counterexample and all its prefixes.

Consider the following example: (also see the original introduction example from Angluin[1])

- $S = E = \{\epsilon\}$
- Table t with:

ϵ	0
a	1
b	1

Fig. 2: Tab t

ϵ	0
a	1
b	1
aa	0
aa	1

Fig. 3: extended Tab t

Initialization:
T is not closed, because $t(a) \neq t(\epsilon)$ 2. Therefore we have to add a and extend our Table t, which is shown in 3.
First attempt:
It rejects an give an counterexample 4, which is ba obviously. Add the counterexample to the given table with its prefixes b. Now the table is no longer consistent and we have to add b as a new row.

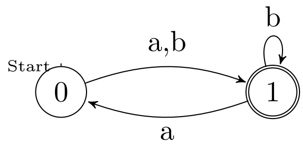

Fig. 4: proofs counterexample

ϵ	0
a	1
b	1
ba	1
aa	0
bb	0
ab	1
baa	0
bab	1

Fig. 5: add b to t

Second attempt:
As you can see Table t is now consistent and closed 5 and the algorithm terminates.

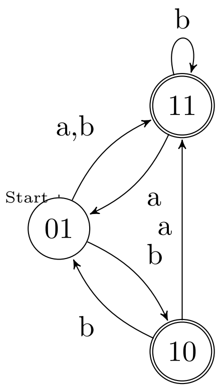

Fig. 6: final state machine

T	ϵ	b
ϵ	0	1
a	1	1
b	1	0
ba	1	1
aa	0	1
bb	0	1
ab	1	1
baa	0	1
bab	1	1

Fig. 7: consistent and closed table t

Analysis:
This algorithm can reduce the worst time execution time tremendously. We add at most several keywords to the table per iteration, which is worst case complexity of $O(n * log(n))$. This is a huge improvement to simply analyse all possible inputs.

Improvements:
Without doubt, the L* Algorithm has much potential of improvements, like the observation table often provides redundant data. As a consequence Kearns and Vazirani replaced the table by a decision tree.

6 Field of usage

Model learning is a growing branch with a lot of space of improvements. In the past few years there were some very important milestones, like AlphaGo. A computer program, which is able to beat the worlds best "Go-players"[4]. The architecture of AlphaGo is implemented with neural networks and searching trees [3].
Another very important segment of model learning is business intelligence, where different kinds of computer programs help the company to decide. In general you feed the program with a huge amount of input from the company. The program will analyse this data and return a decision due to mathematical probability. These decisions already have enormous importance, because most decisions are too complex for a human being [4]. Model learning is one of the most important field of model intelligence and will have tremendous improvements in the next few decades, because there is still a lot of room for optimization.

[4] Go is a board game similar to chess

References

[1] Dana Angluin. Learning regular sets from queries and counterexamples. *Information and computation*, 75(2):97–101, 1987.

[2] Bruce G Buchanan. A (very) brief history of artificial intelligence. *Ai Magazine*, 26(4):53, 2005.

[3] Patricia S Churchland and Terrence J Sejnowski. *The computational brain*. MIT press, 2016.

[4] Peter Gluchowski, Roland Gabriel, and Peter Chamoni. *Management Support Systeme: Computergestützte Informationssysteme für Führungskräfte und Entscheidungsträger*. Springer-Verlag, 2013.

[5] Stephen Hawking. *A brief history of time*. Bantam Books, 1998.

[6] George H. Mealy. *A Method for Synthesizing Sequential Circuits*. Bell Systems, 1955.

[7] Nils J Nilsson. *Artificial intelligence: a new synthesis*. Elsevier, 1998.

[8] Roger C Schank. Where's the ai? *AI magazine*, 12(4):38, 1991.

[9] Frits Vaandrager. Model learning. *Communications of the ACM*, 60(2):86–95, 2017.

[10] Walter Pitts Warren McCulloch. *A Logical Calculus of the Ideas Immanent in Nervous Activity*. Bulletin of Mathematical Biophysics, 1943.