System Induction Games and Cognitive Modeling as an AGI Methodology

Sean Markan(✉)

Eudelic Systems LLC, Boston, USA
markan@eudelic.com

Abstract. We propose a methodology for using human cognition as a template for artificial generally intelligent agents that learn from experience. In particular, we consider the problem of learning certain Mealy machines from observations of their behavior; this is a general but conceptually simple learning task that can be given to humans as well as machines. We illustrate by example the sorts of observations that can be gleaned from studying human performance on this task.

1 Introduction

A generally intelligent agent must be able to learn the dynamics of its environment through experience. One strategy for developing agents with this capability is to study how humans approach analogous learning problems. In this paper we illustrate how one might proceed with this methodology. We must admit at the outset that we have not pushed this methodology all the way through to the construction of learning systems; we aim only to lay some groundwork for that objective (and to propose a class of learning problems relevant to AGI).

First, we need to define a domain in which learning can take place. We will use a class of learning tasks we call "system induction games" (SIGs). In a SIG, the player is presented with an unknown (black box) Mealy machine [14] and must work out rules that predict its behavior.[1] We will not stipulate the source of the input stream; it might be decided by the player, by a teacher trying to help the player learn, or by chance. We further assume the player produces a guess after each input as to what the output will be. Importantly, we are concerned specifically with Mealy machines governed by some "reasonably simple" set of rules—that is, a set of rules a human could work out within a reasonable amount of time. (A very simple example would be a machine which emits a B symbol if either of the two prior inputs were A.) Our goal is *not* to find methods for inducing Mealy machines in general, which is already a well-studied problem.[2]

[1] Recall that a Mealy machine is a finite-state machine which, at each timestep, receives an input i, produces an output $o = f(s, i)$ (where s is its present state), and changes state to $s' = g(s, i)$. Actually, for our purposes there is no particular reason to assume a finite state space, but the games we will consider do have this property.

[2] In fact, the general problem is NP-complete [6].

B. Steunebrink et al. (Eds.): AGI 2016, LNAI 9782, pp. 223–233, 2016.
DOI: 10.1007/978-3-319-41649-6_22

Playing a SIG amounts to learning about the behavior of an environment through experience: the Mealy machine is the environment being learned. The methodology we explore in this paper is to let humans play SIGs, observe their behavior, and attempt to model it. We believe this combination of SIGs and cognitive modeling is a promising approach for AGI. The domain of SIGs gives rise to learning problems which are nontrivial, natural for humans, and isolated from complications like sensorimotor processing. Furthermore, if one models actual human learning mechanisms, one has a reason to think that the algorithms produced will scale to the difficult problems humans solve.

The plan for this paper is as follows. First we will discuss some related work. We will then illustrate the methodology we propose by examining some data produced by the author while playing SIGs. We will look both at the large-scale behavior involved in figuring out a SIG, as well as a broader but shallower sample of data related to conjecturing rules. In each case we will discuss some of the mechanisms and requirements this data seems to suggest for a generally intelligent learning agent. Finally we will propose directions for future work.

2 Related Work

Probably the closest related work is the "Seek-Whence" project of Hofstadter and colleagues, who examined the problem of extrapolating sequences of integers. Like us, Hofstadter emphasized cognitive plausibility as a guiding principle, seeking to build systems which parsed sequences in the same way a human might [8]. This research has led to at least two full-fledged sequence-extrapolating systems [11,15]. There has also been other work on the psychology of sequence extrapolation. Simon and Kotovsky [10,20] carried out some of the most systematic experiments, though the sequences they considered were very simple compared to those of Hofstadter. SIGs differ from Hofstadter's Seek-Whence domain in that they require the extrapolator to induce a *mapping* from input to output—that is, an explanation for a sequence of outputs conditioned on a sequence of inputs, not just an explanation for a sequence of outputs on its own. Thus SIGs are arguably a closer fit to the problem an agent needs to solve to make sense of its environment. At the same time, SIGs can be much harder to solve (even for people).[3]

As alluded to earlier, the problem of inducing Mealy machines has been studied within computational learning theory. Angluin's L^* algorithm [1] provides an efficient solution to the related problem of inducing deterministic finite automata (provided the agent can control the input), and that algorithm can be adapted to Mealy machines (see [19] for a discussion). A number of algorithms for learning POMDPs (which might be viewed as a probabilistic generalization of Mealy machines) have also been proposed [3,12,13]. Algorithms in this vein are only partially applicable as models of human learning on SIGs, in part because SIGs

[3] More precisely, a SIG whose description is about the same length as the description for an integer sequence is likely to be harder to figure out than the integer sequence.

involve additional structure (in particular, they should admit reasonably compact verbal descriptions). Humans are able to recognize and exploit a variety of structural regularities which do not exist in generic state machines but which do play an important role in the environments we actually face. (In the Mealy machine context, such regularities might include two letters of the alphabet being functionally equivalent, a factorization of the machine into several independent parts, sparsity of nontrivial transitions, etc.) By examining human learning in environments with these types of structure, we have a possible window into learning mechanisms relevant to general intelligence.

A few researchers, such as Drescher [4] and Bergman [2], have developed systems which learned rich rule-based models of their environments. Drescher's system, which was inspired by Piaget's theory of cognitive development [17], focused especially on inferring hidden states through their indirect effects on observations. Bergman's system discovered causal relationships in an environment with a rich and highly structured (but directly observable) state space. Both projects considered only a single environment, but developed algorithms with substantial cognitive plausibility that may be relevant in the SIG domain.

On a more general level, the topic of human rule induction has been studied extensively in the context of concept learning (see [7] for a recent approach to this problem). The SIG domain, however, adds several elements to traditional concept learning problems: a need for selecting features from a very large feature space, a need to discover hidden state and how it changes, and the possibility of temporally extended actions (rather than one-off category judgments).

3 General Observations on Human SIG-Playing Behavior

We now discuss some observations based on the author's own experience playing SIGs and recording thoughts while playing. (This methodology is similar to the protocol analysis method of Simon and Newell [5,16].[4]) The games considered used input alphabets of $\{A, C, D, -\}$ and output alphabets of $\{B, -\}$.[5] They are listed in Table 1, alongside the number of turns taken to figure out the game with reasonable confidence and the minimal number of states to represent the game as a Mealy machine. The game history was not visible during the game, so information of interest had to be held in memory, and inputs were random[6]

[4] As they are based on a single subject, we should not expect our observations to generalize in all details to other subjects. This is not a problem for our purposes, since the goal is to collect a sample of *some* of the approaches people apply to SIGs, not to survey them completely. One other note is in order: since the author also wrote the games, some steps were taken to avoid recalling how they worked during play. The games were played some weeks after being created, had their inputs shuffled, and were drawn from a larger set of 48 games.

[5] The $-$ input symbol was chosen as a "null" symbol to indicate nothing of note had happened; this allows an asynchronous interaction to be modeled within a synchronous formalism.

[6] There was some weighting, chosen for each game with the intent of making it more learnable.

Table 1. Sample system induction games.

#	Description (output is − if not otherwise specified)	Turns	States
1	If C occurs, do two Bs; add one extra B if an A is received during or right after those Bs	43	4
2	A toggles a hidden state; in the "on" state, each C produces B	77	2
3	If A occurs, do two Bs	102	2
4	C toggles repetition of B; A suppresses Bs for two turns	125	4
5	Do B if last three inputs were a cyclic permutation of ACD	142	7
6	Do B if the last two inputs were AD or DA	241	3
7	Do B every time the inputs switch between D and non-D	250	2
8	Any time C occurs, respond to the next two Ds with B	250	3
9	If − occurs and there have been ≥ 2 As since the last B, do B	419	3
10	The sequence DA activates a hidden state and AD deactivates it; when in the state, blocks of Cs get the response $-B-B\ldots$	463	5

except in the later parts of Games 8–10, where I got stuck and switched to controlling them directly.

The overarching activity that appears in the transcripts of these games is the conjecturing, testing, and refining of rules. Some rules are easy to deduce: if a certain input symbol always leads to a certain output, we quickly pick up on that. We also readily pick up on block-related rules, such as "in a block of −s, all turns after the first have output −." Other easy features to pick up on include alternation, situations where an event triggers two outputs in a row (or where two equal inputs in a row trigger an effect), and situations where one of two outcomes can occur, for example "in a block of Cs the outputs are either all − or they alternate, starting with −."

The more challenging component of the task tends to be figuring out the conditions controlling an unreliable event. For example, in Game 9, I quickly saw that in a block of − inputs, the first output could be either − or B, but it was not easy to figure out the controlling condition. Let us call the event that prompts the unpredictable outcome the "probe"; in our example the first − of a block was the probe. To figure out non-obvious conditions, I seemed to consider several types of theories. The type considered first was that the relevant considerations for determining the outcome of a probe took place since the last probe; this assumption led me to seek features of the inter-probe history that were successful predictors (such as, in Game 9, the presence of two As). This search process was biased to first consider recent inputs as explanatory factors, which is reminiscent of prediction suffix trees [18]. A second type of theory was that the probe exposed some underlying (hidden) state which could be toggled on or off by certain events. When working under this assumption, I would look at inter-probe periods where the outcome had switched, and try to extract features that would predict the switches. Interestingly, this sort of theory

resembles Drescher's idea of "synthetic items" [4]. A third type of theory, which I came to in Games 8 and 9 only after the other ideas failed, was that performing the probe itself alters the hidden state. In both cases, this conjecture quickly led to the discovery of the actual rule, but from the limited data it is unclear what general method might apply here.

While this very preliminary study does not allow us to draw definitive conclusions, the above patterns suggest that several existing ideas (such as prediction suffix trees, synthetic items, decision trees, and rule-based concept learning [7]) may be part of a learning "toolbox" employed by humans and perhaps appropriate for AGI too. We can also see that there is a need to do a certain amount of "perceptual" processing of the input, even in the rather abstract domain of SIGs. For example, in Game 10, a pattern of alternating *C*s can be viewed as a stable state, and a multiple-input sequence can be viewed as a single event. In Game 4, a two-turn occurrence of "suppression" is viewed as a single event.

It is also interesting to consider what the types of theories we contemplate tell us about the environments we are biased to learn. The theory types mentioned earlier are most effective when hidden state is controlled by recent or distinctive events, or is relatively stable between probes. These features need not be present in general; one can imagine machines (think hash functions) in which every input "scrambles" the state. On a more fundamental level, the *existence* of effective probes, and indeed of a distinction between "really" hidden and "not-so-hidden" state,[7] is a form of structure that humans seem to productively utilize. It seems likely that we will need to design agents with similar considerations in mind if we want them to display general intelligence.

4 A Model of Early Decision-Making on SIGs

The previous section was effectively a "depth-first" investigation of human SIG behavior. We can also take a breadth-first approach and look at a smaller amount of behavior on a wider sample of games. In an effort to do this, I examined a sample of 73 situations a SIG player could face on the fifth turn and annotated each with a judgment of which guesses would be appropriate, which is a proxy for what hypotheses humans are inclined to consider.[8] We will present a rule-based model which replicates this data and discuss what it suggests about how humans go about rule induction in SIGs.

[7] In the SIG formalism, *all* state is hidden; what we mean by "really" hidden state is information about the state which cannot be inferred from the recent history.

[8] The sample of situations considered was generated by taking all distinct histories (treating relabelings as equivalent) satisfying two conditions: (i) that the history contained mixed evidence and (ii) that the last two inputs were the same. By "mixed evidence" we mean that there was at least one input that was followed by both outputs; this restriction was chosen because cases with no mixed evidence were uniformly felt to be easy decisions (just do what worked last time). The second restriction was an arbitrary choice to reduce the sample to a manageable size.

Before doing this, however, it may be worth clarifying why we care about obtaining such a model. The logic is as follows. In order to learn how an environment behaves, humans (or artificial agents) must notice regularities in it and entertain hypotheses about the causes of those regularities. This raises the question of *which* regularities and hypotheses humans naturally consider.[9] Human SIG judgments are surely based on the regularities we perceive and the hypotheses we generate, so by modeling the judgments we can begin to inventory the mechanisms behind those elements of the learning process.

Note that even though the preliminary "inventory" we construct takes the form of a collection of rules, these rules should not be confused with the rules that characterize the environments we want to learn—the inventory is instead more akin to a set of "meta-rules" for conjecturing environment-governing rules. Given the preliminary nature of this study, our meta-rules are not particularly sophisticated, but they serve to illustrate the methodology we have in mind.

4.1 Data

The judgments and the number of times each was used are shown in Table 2. Sample histories, together with corresponding judgments, are shown in Table 3. One interesting finding about these judgments is that in general they are not very hard to make; we do have an intuition for what choices are reasonable, just as we have intuition for what would be a reasonable continuation of the sequence $1, 4, 9, \ldots$. As one might expect, however, there is inevitably a bit of fuzziness, and at times a history seems to border two categories, or we change our judgment from one day to the next. In a few cases (< 10), as I worked on the model, I discovered similar histories that I had annotated differently, but which (when considered side by side) I did not feel deserved differing judgments. In these cases I simply adjusted one of the judgments to obtain uniformity.

Table 2. Judgments on the fifth turn of our sample of SIGs.

N	Judgment
16	$-$ is the only good answer
18	$-$ is preferred but B is not too unreasonable
27	B and $-$ are about equally good
3	B is preferred but $-$ is not too unreasonable
9	B is the only good answer

[9] These questions speak to the problem of what inductive bias is appropriate for AGI. There are of course theoretical proposals like Solomonoff induction [21], but we are motivated by a desire to address the issue empirically.

Table 3. Sample judgments on the fifth turn of SIGs.

In	Out	In	Out	In	Out	In	Out	In	Out
A	B	A	B	A	B	A	B	A	B
C	–	C	B	A	–	C	B	C	B
A	–	C	B	A	B	A	–	A	–
C	–	A	–	C	–	D	B	C	B
C		A		C		D		C	
only –		prefer –		both		prefer B		only B	

4.2 Model

The following model captures all 73 of the judgments. It is based on a set of rules, each of which examines the history and, if it meets certain conditions, proposes an output.[10] If a rule finds the condition it was looking for, we say it "fires." The rules are divided into four groups, which we have labeled 1A, 1B, 2, and 3. These rules function in a preference system. When determining the preferred output, rules in group 1 (A and B) are first consulted. If one or more rules in group 1 fire, then the action(s) those rules propose become the preferred one(s). If no group 1 rule fires, then group 2 is consulted in the same way. (It turns out a rule in group 2 always fires, so group 3 is not needed to determine the preferred action; its role is to determine whether the other option is disallowed or just weak.) In most cases, if a lower-ranked rule proposes an action but the preferred action was already set in a higher group, then the former action is felt to be a weak option. However, the rules in group 1A are so compelling that they outweigh this tendency—proposals from lower-ranked rules are not felt to be options at all, not even weak ones.[11]

The rules are listed in Table 4. Those that can't be described in one line are given names and will be described presently. Throughout, let I be the present input and T be the present turn. $|\cdot|$ will mean the length of a collection of turns.

AnalogousBlocks. A "block" is a contiguous group g of turns with the same input such that (a) $|g| \geq 2$ (b) g is not a subset of a larger block. If T is part of a block b, and there is a prior block b' with $|b'| \geq |b|$, and the outputs of b' have agreed with those of b so far, we say that b and b' are analogous. For example, in $(AB)(A-)(C-)(DB)(D?)$ the last two turns constitute a block analogous to the first two. In this situation the rule says to pick the output that occurred in b' in the position corresponding to T.

AlternationWithinTailBlock. If T is part of a block of length at least 3, and the outputs in that block alternate, continue the alternation.

[10] One rule actually proposes both outputs as acceptable.

[11] As it turns out, this mostly applies to group 3 rules. There is only one case in which a 1A rule suppresses a group 2 rule, namely the $AAAAA/B - B-$ situation.

Table 4. Rules modeling a set of fifth-turn SIG decisions.

1A.	$AAAAA$ and $B-B-$ (predict B)
	$AAAAA$ and $BB--$ (predict $-$)
	$B---$ (predict $-$)
1B.	$B---$, $BB--$, or $BBB-$ (predict $-$)
	AnalogousBlocks where both blocks have the same input
	(this implies the input is $AACAA$)
2.	DoLast (copy the most recent output)
	$B-B-$ (predict B)
	$B--B$ (predict $-$)
	AlternationWithinTailBlock
	AnalogousBlocks
3.	AnalogousExperience
	PredominantExperience
	TiedExperience (predict both B and $-$ as acceptable)
	LastExperience
	PredominantAll

AnalogousExperience. Let $e(i)$ be the player's "experience" with input i: the sequence of outputs that have occurred when i was the input. If $|e(I)| \geq 1$, and there is some i with $|e(i)| > |e(I)|$, and $e(i)$ has agreed with $e(I)$ so far, predict the current output from the output at the corresponding position in $e(i)$.

PredominantExperience. If one output has occurred more times than the other in $e(I)$, predict it.

TiedExperience. If both outputs have occurred equally often (and at least once) in $e(I)$, predict that both are acceptable.

LastExperience. If $|e(I)| \geq 1$, predict the last element of $e(I)$.

PredominantAll. If one output has occurred more times than the other in the full history, predict it.

4.3 Discussion

All the listed rules are necessary, although some handle only a couple cases (AnalogousExperience only handles one), and some could be replaced with a different but equally effective rule (for example, we could use an AlternatingExperience rule instead of the TiedExperience rule). Many of the rules are not too surprising; they indicate that we pay attention to alternation, to switches from one output to another, to recent outputs, and to the outputs that previously occurred with the current input. Probably the most interesting feature of the model is the concept of a "block" (a contiguous group of turns with the same

input), which is used in multiple ways: a block boundary can be seen as a "cut point" which allows a pattern that exists within the block but not beyond it to be seen as legitimate (this occurs in the AlternationWithinTailBlock rule), and blocks can be used to create analogies between one sequence of turns and a past sequence of turns, thereby allowing more complex predictions than would otherwise be possible.

There are a few rules that are to some extent artifacts of the limited set of cases we considered. For example, DoLast would probably be a much less robust rule if we hadn't restricted attention to cases where the fourth and fifth inputs were the same. Another question we might ask is, is it really the case that 1A proposals are so *good* that they knock out everything else, or is it rather that lower-ranked rules actually need some conditions which are not met in 1A situations to be viable? It is hard to say without more data. And of course some of our rules (like the 1A rules) are overly specific; they were left this way because we did not have enough data to make a well-supported generalization. Nonetheless, in these rules we can start to see the kernel of a more general system. It seems likely that many of the rules would generalize to longer histories or other types of games. Overall, the rules of the model suggest that we have a diverse set of primitives from which we can construct conjectures about the behavior of environments, and that some of these primitives involve perceiving higher-level entities (such as blocks) within the stream of events.

5 Conclusion

In this paper we used data from human performance on system induction games to generate some preliminary ideas about the underlying mechanisms humans use to approach these problems. By extension, this sort of analysis can suggest mechanisms that might be appropriate for artificial generally intelligent agents that learn from experience. The particular mechanisms proposed here are not as important as the overall methodology, which may be summarized as follows: pick a simple but nontrivial class of games, let humans play them, capture the guesses made (and thought processes used) by humans, and infer learning mechanisms.

Regarding possible future directions for this work, obviously there are many more games to explore in much greater depth, and it would be desirable to construct complete SIG-playing algorithms. A likely stepping stone towards that goal would be to develop a "system grammar" which would formally define a space of SIGs humans can easily understand and learn. Such an effort would be analogous to the goal in linguistics of constructing a grammar that defines the space of acceptable sentences, and indeed at least one linguist has explored analogies along these lines [9] (see Chap. 4).

On a larger scale, more sophisticated types of environments could be considered, for example environments with continuous state spaces or sensorimotor components. Another logical extension would be to examine other aspects of human SIG performance, such as how difficult humans find particular decisions

and what features of the history we remember (certain features are more memorable than others, such as repetition, blockiness, or symmetric patterns like $ACCA$). These would be suitable modeling targets, just as output decisions are.

A final direction for investigation would be to ask where (meta-)rules like those described in Sect. 4.2 come from (assuming they are psychologically real). Were they themselves learned through experience? Or perhaps they can be explained as combinations of simpler primitives. Studying more sophisticated games may help develop answers to these questions.

Acknowledgments. The author wishes to thank the anonymous reviewers and An-Dinh Nguyen for helpful comments.

References

1. Angluin, D.: Learning regular sets from queries and counterexamples. Inform. Comput. **75**(2), 87–106 (1987)
2. Bergman, R.: Learning World Models in Environments with Manifest Causal Structure. Ph.D. thesis, Massachusetts Insitute of Technology (1995)
3. Chrisman, L.: Reinforcement learning with perceptual aliasing: the perceptual distinctions approach. In: Proceedings of the Tenth National Conference on Artificial Intelligence, pp. 183–188. AAAI Press (1992)
4. Drescher, G.L.: Made-Up Minds: A Constructivist Approach to Artificial Intelligence. MIT Press, Cambridge (1991)
5. Ericsson, K.A., Simon, H.A.: Protocol Analysis: Verbal Reports as Data. MIT Press, Cambridge (1993)
6. Gold, E.M.: Complexity of automaton identification from given data. Inform. Control **37**(3), 302–320 (1978)
7. Goodman, N.D., Tenenbaum, J.B., Feldman, J., Griffiths, T.L.: A rational analysis of rule-based concept learning. Cogn. Sci. **32**(1), 108–154 (2008)
8. Hofstadter, D.: Fluid Concepts & Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought. Basic Books, New York (1995)
9. Jackendoff, R.: Language, Consciousness, Culture: Essays on Mental Structure. MIT Press, Cambridge (2007)
10. Kotovsky, K., Simon, H.A.: Empirical tests of a theory of human acquisition of concepts for sequential patterns. Cogn. Psychol. **4**(3), 399–424 (1973)
11. Mahabal, A.A.: Seqsee: A Concept-centered Architecture for Sequence Perception. Ph.D. thesis, Indiana University Bloomington (2009)
12. McCallum, A.K.: Reinforcement learning with selective perception and hidden state. Ph.D. thesis, University of Rochester (1996)
13. McCallum, R.A.: Instance-based utile distinctions for reinforcement learning with hidden state. In: ICML, pp. 387–395 (1995)
14. Mealy, G.H.: A method for synthesizing sequential circuits. Bell Syst. Tech. J. **34**(5), 1045–1079 (1955)
15. Meredith, M.J.E.: Seek-Whence: A model of pattern perception. Ph.D. thesis, Indiana University (1986)
16. Newell, A., Simon, H.A.: Human Problem Solving. Prentice-Hall, Englewood Cliffs (1972)

17. Piaget, J.: The Origins of Intelligence in Children. International Universities Press, New York (1952)
18. Ron, D., Singer, Y., Tishby, N.: The power of Amnesia: learning probabilistic automata with variable memory length. Mach. Learn. **25**(2–3), 117–149 (1996)
19. Shahbaz, M., Groz, R.: Inferring mealy machines. In: Cavalcanti, A., Dams, D.R. (eds.) FM 2009. LNCS, vol. 5850, pp. 207–222. Springer, Heidelberg (2009)
20. Simon, H.A., Kotovsky, K.: Human acquisition of concepts for sequential patterns. Psychol. Rev. **70**(6), 534 (1963)
21. Solomonoff, R.J.: A formal theory of inductive inference. Part I. Inform. Control **7**(1), 1–22 (1964)