How one can learn programming while teaching reasoning to children with autism

School of Computer Science and Information Systems
Birkbeck College, University of London
Malet Street, London WC1E 7HX UK
galitsky@dcs.bbk.ac.uk

Introduction

The study of the phenomena of autism is an interesting example of overlapping between AI and cognitive sciences. Development of the logical AI is necessary to characterize deviations of autistic reasoning. An early development of autism as a cognitive system is a very appealing object of study for logical AI because of its simplicity and compactness. The reasoning of children with mild autism in certain domains is quite naïve and simple whereas the reasoning of control children of the same physical age, by the time they are verbal, is already fairly complex for simulation. Hence exploration of the phenomenology of autistic reasoning stimulates both disciplines; and they need to complement each other to rehabilitate it.

It is well known that most autistic children readily interact with software and prefer such interaction over communication with other people (see e.g. Green 1996, Eigsti & Shapiro 2003). Naturally, they may learn a lot from this software in terms of reasoning, only if it is sufficiently *intelligent.* A brute-force or domain-specific solution, which AI sometimes adopt, attempting to resolve a number of hard problems, is not very helpful in the domain of intelligent software for autistic rehabilitation. Such software must have a robust and adequate model of a domain being taught and implement reasoning in a sufficiently correct and complete manner to lead the learning process. Otherwise, if a model is inadequate or a reasoning implementation is insufficiently expressive, a trainee with autism would likely loose an interest to interact with such software system (Leekam & Prior 1994, Galitsky 2005). Hence the domain of rehabilitation of autistic reasoning stimulates an advancement of the state-of-art of automated reasoning.

Teaching children with autism and programming

The issue of training to overcome various deficiencies of autistic reasoning has been addressed in a number of studies (Green 1996; Baron-Cohen 2000). There is a series of peculiar techniques developed to teach children with autism certain forms of reasoning, mainly reasoning about mental states and actions, reasoning about generic actions, default and defeasible reasoning, deductive, inductive, abductive and analogical reasoning patterns, probabilistic decision-making etc. Skills of reasoning in some of these domains are lacking in every child with autism (Howlin 1998).

Teaching by analogy is the standard technique for both junior students and adults in a majority of subject domains. However, autistic trainees experience significant difficulties learning from examples, they can imitate some forms of behavior and actions of other people but do it without understanding. Also, visual programming tools is an efficient way to introduce abstract and general programming concept, they are quite efficient for both education of programming and efficient software development. In spite of the appeal

to use visual programming tools, autistic children do not learn abstract reasoning patterns from them most of times.

Hence in terms of reasoning patterns, controls learn by induction and analogy, and reinforce learning results by deduction (explicit rules) in most of real-world domain (excluding e.g. math). At the same time, autistic trainees learn by deductive rules most of the time, and other reasoning patterns play auxiliary roles only (Galitsky 2005).

Therefore, teaching autistic trainees in any domain must be preceded by formulating exact and explicit rules. Otherwise, the teaching approach which might be adequate for a control trainee would be unacceptable for an autistic trainee, as our experience shows (Galitsky & Goldberg 2003). Teaching a new entity to a child with autism, one needs to make sure that all entities the current one refers to are fully conceptually understood. On the contrary, a child from a control group is ready to acquire a new entity in the environment where some features are uncertain, assuming she can learn them later.

The idea of this study is to explore the similarity between formulating domain knowledge in a way acceptable by a computer and formulation of this knowledge to be acquired by an autistic trainee. We enumerate the commonalities in cognitive demands of computers and autistic trainees with respect to teaching them knowledge representation and reasoning in real-world domains:

1) All concepts have to be *clearly* and *explicitly* defined. A basis of indefinable concepts may be selected, but a programmer/teacher should be aware that a computer or trainee will not be able to freely operate and provide explanations with these concepts from the basis. For example, when taught the rules for basic mental states of the mental world (knowledge and intention), followed by the rules by derived mental / communicative actions derived from this basis, the autistic trainees are capable of explaining what is *pretending* and *deceiving* (derived) but not what is *knowledge* and *intention* (basic).

2) Definitions for concepts can be either *procedural* or *declarative*. A trainee can be taught a sequence of actions to achieve a goal, or a clause for a sequence of conditions an environment should satisfy to achieve this goal. To be capable of training in a declarative way, respective trainees' skills have to be developed. For example, if a child with autism is requested to be at the top of a rock in the middle of a puddle with a fishing pole, the child needs some skills to determine the order of operations: put on rubber boots, take a fishing pole, cross the puddle and climb the rock. In contrast to a control child who would acquire this skill independently on the basis of trial-and-error, a child with autism needs a substantial guidance to learn how to search for a proper sequence of actions independently.

3) All special cases should be addressed. For example, for an arbitrary predicate like *want* we would expect a smart trainee to operate with *want(Who, What)* with arbitrary *Who* and *What*. It is neither the case for a child with autism who does not understand that other people may want something,

When we refer to an autistic or computer software trainee, we assume a medium-to-high-functioning individuals with autism and a standard software environment without sophisticated machine learning systems like explanation based generalization (Mitchell 1986) or inductive logic programming (Muggleton & Firth 1999).

Programming behavior in the mental world

Initial mental state

| Find an action that can be performed at this time |
| Estimate how wanted/ unwanted is the current state |
| Assume the chosen action has been committed |
| Estimate how wanted/unwanted is the resultant state |
| Compare the number of wanted / unwanted features for the current and resultant states |

| Explicitly wanted actions |

or

| Action is a valid mental formula |

| Continue or change behavior |

Choose the form of behavior:
- commit physical action
- share information
- conceal information
- ask to obtain information
- ask to commit an action
- ask to share information
- ask to conceal information
- cheat to achieve
- suggest to perform
- follow a suggestion
- decline a suggestion, etc

State/Action preference order
- Explicitly preferred
- Action that leads to desired state that is not current
- Action that eliminates unwanted state that is current
- Action that does not lead to unwanted state that is not current

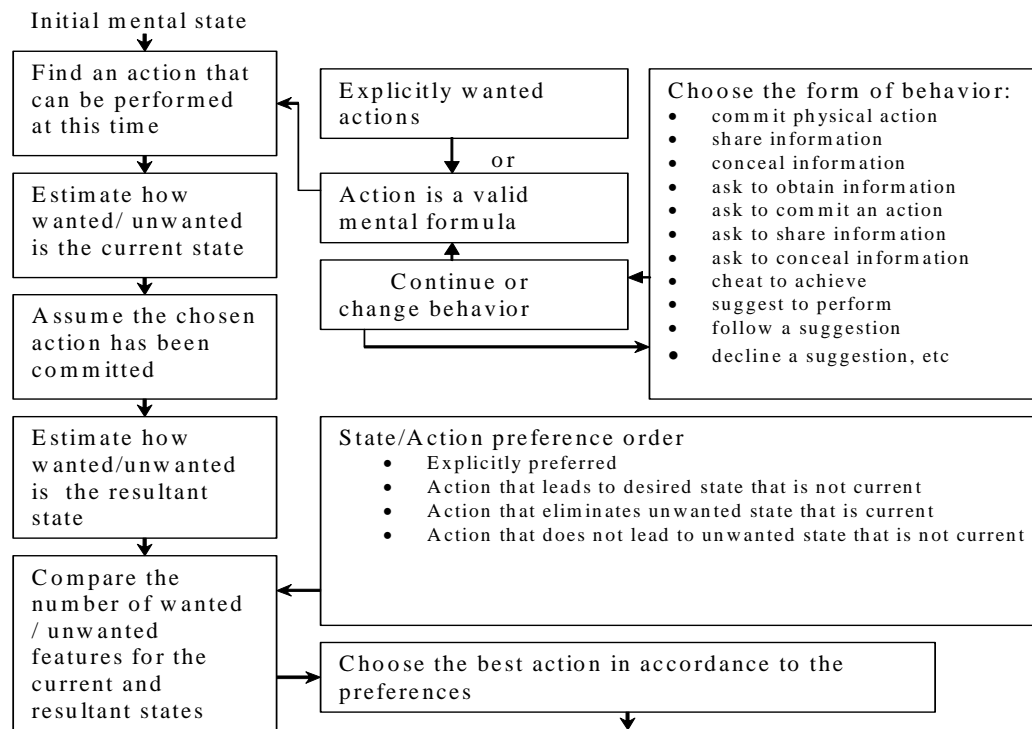| Choose the best action in accordance to the preferences |

Fig. 1: The algorithm for choice of mental/communicative action in the mental world. An agent takes into account possible actions of opponent agents using a similar architecture (Galitsky 2003).

Experience accumulated while helping autistic children to understand the mental world is valuable for building engineering applications where modeling of human agents' attitudes is crucial. The chart (Figure 1) depicts the way we explain to autistic trainees how they should select a proper mental / communicative action. Firstly, the trainee selects a set of actions he can legally perform at the current step (physically available for him, acceptable in terms of the norms, etc.). Such an action may be explicitly wanted or not; also, this action may belong to a sequence of actions in accordance with a form of behaviour which has been chosen at a previous step or is about to be chosen. In the former case, the trainee may resume the chosen behaviour form or abort it.

Having a set of actions which are legal to be currently performed, the trainee applies a preference relation. This relation is defined on states and actions and sets the following order (1 is preferred over 2-5, 2 is preferred over 3-5, etc.):

- Explicitly preferred (wanted) action.
- The action that leads to a desired state that is not current.
- Action that eliminates an unwanted state that is current.
- Action that does not lead to an unwanted state that is not current.

- Action that does not eliminate a wanted state that is current.

Hence the trainee has an initial intention concerning a Chosen Action or State, assesses whether this condition currently holds, then selects the preferred Chosen Action, assumes that it has been executed, deduces the consequences, and finally analyses whether they are preferential. Naturally, the preference, parameters of trainee's attitudes and multiagent interactions may vary from scenario to scenario. Before an action can be assumed, the trainee needs to check that a potential action is a valid mental formula. A valid mental formula is neither an axiom (such as *an agent knows what it knows*) nor implausible formula (such as *pretending about someone else's state*). A resultant state comprises one or more explicitly wanted or unwanted states; the autistic trainee performs the comparative analysis of preferences on a state-by-state basis.

The same algorithm (Figure 1, Galitsky 2002) for the simulation of decision-making by human agents is used in solving engineering problems in such domains as solving constraint satisfaction problem in the environment of conflicting human and automatic agents, (scheduling for the broadcasting industry), automated synthesis of scenarios (e.g. for Internet advertisement), modeling mental states of investors for market predictions, extraction of the mental behavior patterns from the wireless-based location services data, and simulating relationships between economic agents (Galitsky 2003).

We have conducted evaluation of how the simulation-based algorithm which turns out to be optimal for teaching children with autism outperforms the traditional modal logic-based approaches because of higher expressiveness of representation language operating on the level of behaviors, closer following the natural language set of mental entities and higher efficiency of search (Galitsky 2003).

Programming adjustment of action

The experience teaching children with autism sheds a light on how to make reasoning by intelligent software more efficient. Teaching autistic children to make decisions concerning proper behavior, it is important to distinguish typical and atypical cases. Typical situations are assumed first, and a typical action (or response) is selected. However, it might be necessary to adjust the selected action be to specific (atypical) circumstances, if the assumption that the situation was typical is defeated by these circumstances. It is very important to teach children with autism a proper algorithm of how to adjust a selected action to an environment to avoid an exhaustive search through a totality of possible actions on one hand and nevertheless find an adequate action on the other hand.

This asymmetric approach of handling typical (immediate) and atypical (afterwards if necessary) cases is known as default reasoning and would be quite useful in software applications. Nowadays, default reasoning (and nonmonotonic reasoning in general) does not find a lot of applications in software. The flavor of handling typical and atypical (exception) behavior of a program can be followed in the implementation of *try-catch* approach. Having obtained the experience while teaching children with autism to handle exceptional situations, we come to the belief that the object-oriented programming would benefit from division of methods into typical which are in use under normal operations, and atypical which are invoked under incorrect user operation. Firstly, IF conditions for typical method should precede those for atypical method. Secondly, it may be efficient

not to invoke atypical methods directly at all but only do that when typical ones through exceptions.

Serving dinner

Your friends are visiting you. You are serving a dinner. Now your guests are almost done eating the main course. You are being asked to pick up plates...

Pick up plates ▼

The guest keeps eating ▼ ▯▯▯ ⇒ Wait till guests are done eating ▼

The guest is done eating and askes for more food. There are food remains in the plate. ▼ ▯▯▯ ⇒ Pick up the plate first and then offer more food ▼

On my way to school

You are on your way to school. It rains today, so there is paddle in the area which is usually dry. Besides, there are other complications on your way...

I am on my usual way to school ▼

There is a paddle on the way ▼ ⟹ Go around the paddle ▼

Not enough space to go around ▼ ⟹ Go straight ▼

My shoose are very expensive ▼ ⟹ Turn back ▼

Nothing special on my way back ▼ ⟹ Turn back ▼

Fig. 2 The screen-shot of the form for teaching proper adjustment of action (Galitsky & Peterson 2005)

Our attempts to teaching children with autism how to properly select actions and adjust them to context lead to the following environment (Figure 2). To demonstrate how actions are adjusted to environments, we use an interactive form where a sequence of default rules is represented as a series of drop-downs showing current circumstances (on the left) and respective drop downs (on the right) where actions are chosen by trainees. Selecting the items on the left, trainees imitate respective sequence of (changing) circumstances/ contexts, and the appropriate action adjustment (correct action) should be selected on the right. The links between the selections on the left and those on the right is implemented via default rules.

The forms serve as a main means to evaluate trainees' performance choosing proper actions in artificial and real-world environments. The exercises are built providing there is a single best solution (most adequate choice of actions) for each context. The focus of this exercise is to develop the capability of changing plans online. The user interface represents a decision-making procedure in changing environment. Autistic children

enthusiastically interact with the form, extending existing environments by new circumstances and actions, and creating new environments (Galitsky & Peterson 2005).

. One of the success stories with this approach to autistic rehabilitation is that it mediated communication between two non-verbal teenagers. Both of them did not speak, and could only communicate via handwriting or typing on a keyword. Using the exercise depicted at Figure 2 they got interested in the topic and were eager to share the results of their interaction with each other. Being unable to communicate directly, they used the software environment as a communication media, because it was an adequate way to express their reasoning patterns they apply in the real world.

Programming operations with hypotheses

Our accumulated experience of teaching autistic children how to behave properly has contributed to the design of a rule-based machine learning system which automatically generates hypotheses to explain observations, verifies these hypotheses by finding the subset of data satisfying them, falsifies some of the hypotheses by revealing inconsistencies and finally derives the explanations for the observations by means of cause-effect links if possible. How can performance of such systems as inductive logic programming [20] and explanation-based learning be improved by taking into account observations concerning operations with hypotheses by children with autism? We will outline the experimental settings and observations.
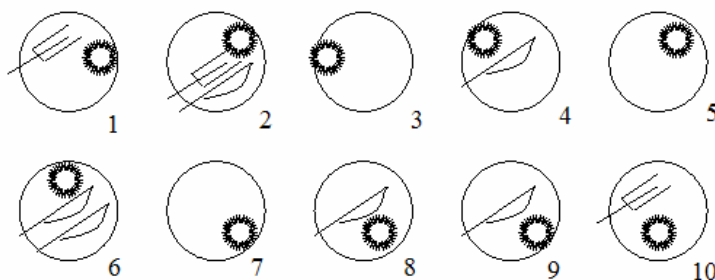

Fig. 3. A hungry subject is suggested to eat cookies from the ten plates.

In the hypotheses formation setting (Figure 3), the subject is notified that some cookies are added an unpleasant taste in accordance to some rule that is not disclosed. The subject is required to eat all cookies with good (expected) taste and state that the rest of cookies are altered. For the purpose of verification, a subject is encouraged to formulate the revealed rule when done with cookies.

When a trainee tries all cookies one-by-one, she discovers that cookies from plates 1,3,5,6,7,10 are normal and those from plates 2,4,8,9 are added an unpleasant taste. The objective of this experimental environment is to come up with an algorithm of forming, confirming and defeating hypotheses such that the least number of cookies with unpleasant taste is eaten.

A good way to do it some children invented is to find the common property of all good cookies and all bad cookies. Applying inductive procedure to positive and negative

examples turns out to be a good advancement of both inductive logic programming and explanation-based learning, which generalize positive examples only.

Conclusions

The objective of this short paper is to demonstrate that experimental cognitive science is relevant to a number of important AI problems in reasoning and machine learning. We focused on the domain of autistic reasoning which is a curious mixture of topics in AI and cognitive sciences. Having commented on the commonalities of teaching autistic children and teaching computers (programming) to solve real-world problems, we provided simplified illustration on how the experience of the former can be applied to the latter. Our claims is that it is significantly easier to teach control children to solve these problems than children with autism, and, obviously, it is even more so for programming.

We illustrated that lessons learned in teaching reasoning about mental world, adjusting one's action to an environment and can be naturally applied to improve the performance of machine reasoning in the respective domains. Hence we conclude that theoretical and experimental cognitive science of autistic reasoning might contribute to such traditionally "technical" areas as machine learning and reasoning.

References

1. Baron-Cohen, S. (2000). Theory of mind and autism: a fifteen year review. In S. Baron-Cohen, H. Tagar-Flusberg, and D. J. Cohen (Eds.), Understanding Other Minds, Volume A, pp. 3–20. Oxford, U.K.: Oxford University Press.
2. Baron-Cohen, S., Golan, O., Wheelwright, S., & Hill, J. J. (2004). Mind Reading: the interactive guide to emotions. London: Jessica Kingsley Limited.
3. Galitsky, B. (2002) Extending the BDI model to accelerate the mental development of autistic patients. *Second Intl. Conf. on Development & Learning*. Cambridge, MA.
4. Galitsky, B. (2003) Natural language question answering system: technique of semantic headers. *Advanced Knowledge Intl.* Adelaide Australia.
5. Galitsky, B. & Goldberg, S. (2003) On the non-classical reasoning of autistic patients. *International Conference on Neural and Cognitive Systems* Boston University, MA.
6. Galitsky, B. and Peterson, D. (2005) On the Peculiarities of Default Reasoning of Children with Autism FLAIRS-05.
7. Galitsky, B. (2005) Distance Learning Rehabilitation of Autistic Reasoning. Encyclopaedia of Distance Learning. Idea Group Publishing.
8. Green, G. (1996). Early behavioural interventions for autism: What does the research tell us? In C. Maurice (Ed.), Behavioural Intervention for Young Children with Autism. Austin,TX: Pro-Ed.
9. Howlin, P. (1998). Children with Autism and Asperger Syndrome: A Guide for Practitioners and Carers. New York: John Wiley and Sons.
10. Leekam SR, Prior M. (1994) Can autistic children distinguish lies from jokes? A second look at second-order belief attribution. *J Child Psychol Psychiatry.* Jul;35(5):901-15.
11. Mitchell, T.M., Keller, R.M. and Kedar-Cabelli, S.T. (1986). Explanation-based generalization: a unifying view. Machine Learning. 1: 47-80.
12. Muggleton, S. and Firth. J. (1999) CProgol4.4: Theory and use. *In* Dzeroski, S. and Lavrac,N. (Eds), *Inductive Logic Programming and Knowledge Discovery in Databases*.
13. Eigsti, I. M. and Shapiro, T. (2003). A systems neuroscience approach to autism: biological, cognitive, and clinical perspectives. Mental Retardation and Developmental Disabilities Research Reviews 9, 205–215.