An Instance Learning model of Task-Action Mappings

Daniel Freudenthal (T.D.Freudenthal@tue.nl)
IPO Center for User-System Interaction
Eindhoven University of Technology
P.O. Box 513, 5600 MB
Eindhoven, The Netherlands.

Abstract

A model is described which simulates the behaviour of subjects in an experiment where they learned to control an interactive device. The amount of foreknowledge that the subjects had was manipulated, and the resulting differences in behaviour were assessed. The model employs a combination of instance learning and difference reduction strategies. Subjects without foreknowledge start out exploring the device in a random fashion, which is replaced by more informed strategies as the number of succesful instances increases. For subjects with foreknowledge, strategies are more informed from the outset. The model provides a reasonable fit to the number of actions that both groups of subjects require for later rounds, but is less successful in modelling the behaviour of very early attempts.

Introduction

Existing models of interactive behaviour such as GOMS (Card, Moran & Newell, 1983) can provide an estimate of how long it will take a user to perform a certain task. Task Action Grammars (Payne & Green, 1986) give an account of the consistency of a device and can therefore provide an estimate of a device's learnability. One drawback of these models is that they aim to model a proficient user, and, as such, one might argue that they are models of the task, rather than models of the user. Thus, these models give limited insight into the learning process taking place, or to the strategies that users bring to the task of learning to perform interactive tasks. One exception to such models is the Ayn model by Howes (1994). The Ayn model is a model which learns to perform a menu task, though its' initial behaviour is driven by semantic knowledge regarding the relevance of a menu item considering the present goal. This knowledge is given to the model by the analyst. Similarly, Howes & Young's (1996) Task Action Learner (TAL), learns the interaction through an instructor, who provides advice regarding the proper course of action when the model does not know which action to perform. The CE+ model (Polson & Lewis, 1990), aims to describe learning processes, but, since it is a hand simulation, it is not compared to to actual data. Rieman, Young and Howes (1996) developed the IDXL model which integrates TAL and Ayn adding cost-benefit considerations in menu selection as well as a simple analogy model. The LICAI (Kitajima & Polson, 1995, 1997) model finally, uses Kintsch's Construction integration (C-I) model to generate goals from instructions and focuses on the selection of controls based on their semantic match with the goal and their activation value in memory. LICAI models expert (though not errorless) behaviour. Both IDXL and LICAI aim to model display-driven behaviour.

One drawback of these models is the fact that, though they model empirical results, their performance is not compared to actual data from human subjects, nor do they take the actual foreknowledge that subjects possess into consideration. The present model was developed in an attempt to learn more about how strategies in learning operating procedures vary as a function of relevant foreknowledge. The model is compared to data collected from human subjects.

Experimental Task

The model that was developed in ACT-R (Anderson & Lebiere, 1998) interacts with the device shown in fig. 1. The device, modelled after the one used by Kieras & Bovair (1984), is a simulated medical laser which has three different power levels which differ in their operating procedures. As the number of actions that is needed for every power level differs, this also constitutes a measure of complexity. The device employs a power button (left-most control), limiter (top middle), battery (bottom middle), and three 'laser activation buttons'. The three different power levels can be used in the following way:

- Green: switch power to I, press 'green'.
- Yellow: switch power to I, limiter to 'Tr', 2. press 'yellow'.
- Red: switch power to I, 2. limiter to 'Tr', 3. battery to 'Lo, press 'red'.

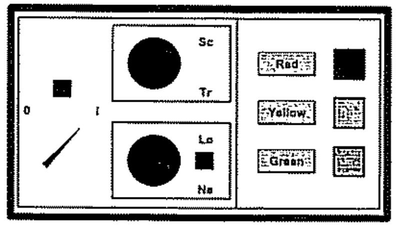


Fig. 1: The experimental device.

The fact that a laser is active is signalled through the flashing of the light next to the laser activation button. The described procedures are the minimum procedures required. Thus, the green laser can be used in any state of the device, provided the power is on. The order in which the actions are performed is also largely irrelevant, though the laser activation button has to be used last. Thus, the yellow laser can also be used by first switching the limiter to 'Tr', then switching then power to I, and then pressing yellow.

Data concerning the learning curves of subjects interacting with this device were collected in a study described in Freudenthal (1998). In this study, subjects were given written instructions which either explained the operating procedures for the device (the 'model group') or which simply stated that the power levels differed in terms of their operating procedures (the 'no model group'). Subjects who received the instructions stating the operating procedures then filled out a questionnaire concerning the operating procedures. With the experimental device that was used, it is possible to test this knowledge with a limited number of multiple choice questions concerning the necessary position of the controls for the different tasks. Subjects then interacted with the device during 12 trials which were organised into 4 rounds of 3 trials. On each trial subjects were instructed to (find out how to) use a specific power level (make a specific light flash). All three power levels had to be used in every round. The order in which this was to be done was randomized over subjects, but fixed over rounds. Amount, content and timing of all the actions subjects performed where collected in a logfile.

The Model

Analysis of the task a novice user faces makes clear that remembering the sequence of actions that he has performed on a succesful trial is not a very efficient strategy. For one, this number of actions can be very large, and secondly, this does not make an improvement in performance on a consecutive trial very likely. A more efficient option is to remember the states of the device one has visited, the actions that were performed, and the effect this has had. Thus, the most efficient type of learning is *instance learning*. Instance learning (Logan, 1988), has been shown to provide an adequate account of the (implicit) learning that occurs while controlling dynamic systems (Dienes & Fahey, 1995). The model thus remembers all states of the device it has encountered during the interaction. With the device state, it also stores the action and the effect this had. The model uses three basic strategies. It can either select controls randomly (i.e. a trial-and-error strategy), it can retreive stored instances, or it can use a basic variant of the label-following-heuristic (Kitajima & Polson 1997). The label-following-heuristic states that subjects will choose a control whose label has a semantic match with the present goal. In the task modelled here, this constitutes pressing the laser-activation button that corresponds to the present goal. Inspection of the data from the subjects shows that this button is selected more often than other controls, though irrelevant laser-activation buttons are sometimes selected even in later rounds.

When the model has no knowledge regarding the operating procedures (no instances are known), it will start out by selecting controls randomly. The random selection of controls is obtained by adding noise to the base-level activation of the chunks encoding the different controls. When the model has selected a control, it will push a subgoal to perform an action on that control, and it will observe the result from that action. This result can be one of three situations; there is no response, one of the lights is active, but it is not the desired light, the desired light is active, or there is no response. When the action has been performed, the subgoal will be popped and the model will store the state (positions of the controls), the action it performed, and the result it had in the chunk that is created upon popping the subgoal. The model will continue to select controls randomly until it reaches the desired goal, storing all encountered instances. When the model is presented with a goal, it will check memory to see if a chunk exists which encodes a solution to this goal. If the device's present state is identical to the one in which the present goal was solved on a previous occasion, it will retrieve the action it took the last time, and perform this action again. This instance learning strategy is combined with difference reduction strategies. When the model has a chunk which encodes a solution to a task, but the present state of the device is different from the one in which the task was solved, it will select the controls

which are in a different position than encoded in the solution chunk. Thus, the model can recognize how to reach a state in which it is succesful without having been in this state before. This is possible due to the relative simple device that it interacts with, because the position of all controls can be read off the display at all times. (Visual scanning is not modelled at present though, the state of the device is known to the model at all times). The use of the difference reduction strategy may be less easy with a more complex device.

Performance of more knowledgable subjects can be modelled by providing the model with (part of) the necessary solution chunks at the outset. Since subjects with foreknowledge filled out a questionnaire concerning their foreknowledge, the activation levels of the solution chunks can be based on the number of subjects that correctly answered the relevant questions in the questionnaire. A model with the solution chunks can retrieve these from memory in order to select the correct control.

The combination of instance learning and difference reduction has two implications for the larger task of learning several operating procedures. Firstly, it predicts that, when a sub-optimal solution is found, the model will continue to use this sub-optimal solution until it 'stumbles' on a more efficient solution. Secondly, the model will also learn from instances that are succesful with respect to a goal that is different from the present one. For example, if the model has the goal to employ the red laser, but happens to fire the yellow one, it will store this instance and employ it when it is faced with the task of firing the yellow laser. This attribute of instance learning models suggests that order effects are a natural consequence of instance learning, and makes them a likely candidate for models of exploratory learning.

Parameter fitting

A number of parameters have been fit in the model. For one, the amount of foreknowledge that the model starts out with was manipulated. In order to model the group without foreknowledge, the model was given no solution chunks. For the group with foreknowledge, solution chunks were added to declarative memory before running the model. The strength of these chunks was based on the number of subjects that correctly answered the relevant question in the questionnaire filled out after reading the instruction. Since all subjects in the model group answered all questions correctly, all relevant solution chunks were simply entered in the model (though their base level activation had to be set relatively high in order to guarantee retrieval). In order to introduce randomness, the activation values of the chunks and the strenth of the productions were subject to noise. Furthermore, adjustments were made to the strength of the productions that encoded the strategies of retrieving instances from memory, difference reduction, choosing a random control, or choosing a laser activation button. In order to obtain the fits portrayed in figures 1 and 2, the productions encoding retrieval from memory and difference reduction obtained the highest priority for the model group. This corresponds to a strategy where the subject is actively trying to obtain the state in which he or she knows he can solve the task. For the no-model group, the production which selected the laser activation button corresponding to the present goal received the highest priority. This corresponds to a strategy which selects a (random) control followed by a laser activation button. The fact that retrieving solutions from memory received a higher priority for the model group might be construed as them being more confident about the knowledge they had than the no-model group. Though the no-model group did possess all relevant chunks after having solved all tasks in round 1, they only retrieved this knowledge when the activation of the retrieve (or difference reduction) productions exceeded (dus to noise) that of the production that chooses the laser activation button.

Comparison to human data

The performance of the model was compared to the data from human subjects. The run of the model employed the same randomisations as were used in the experiment. At present, the comparison focuses on number of actions only. For both subject groups the average number of actions that was needed for the three different power levels was computed (averaged over the four rounds). Similarly, the average number of actions for every round (averaged over power levels) was computed. Initial comparison with data from the logfiles showed that the random selection of controls was too simple an approximation, as the model needed far more actions than the subjects. A test to see whether the selected control was different from the one selected on the last occasion was therefore added. This prevents the model from selecting the same control on every occasion. Completely removing the possibility of repetitions is too strict though, since subjects do repeat their actions occasionally). Finally, it became clear from inspection of the logfiles that subjects without foreknowledge very quickly caught on to the fact that the leftmost control is a power switch. To incorporate this into the model, chunks encoding the fact that the power has to be on were added to the model for subjects 'without foreknowledge'. These refinements of the model resulted in the fits depicted in figures 2 and 3.

Fig. 2: Model fit (number of actions) as a function of complexity level (averaged over rounds).

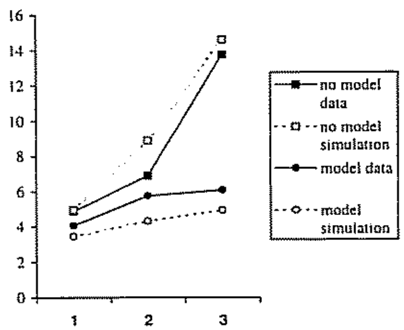
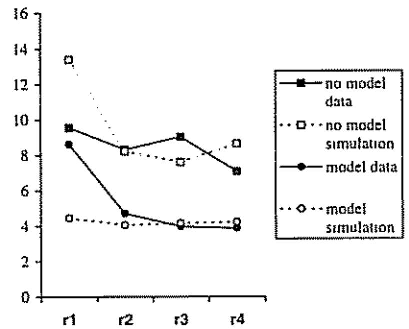


Fig. 3: Model fit (number of actions) as a function of round (averaged over omplexity levels)



As can be seen in figure 2, the model does a reasonable job in fitting the data from the three different complexity levels (though it should be noted that the three levels differs in terms of minimum required number of actions, i.e. 2, 3, and 4 actions are the minimum required for the green, yellow and red laser). The model consistently needs less actions than the subjects for the model group, while it needs more actions for the no model group. Inspection of figure 3, which depicts the data for the different rounds (averaged over complexity levels) shows why this is so. Though the fit is reasonable for rounds 2, 3 and 4 (especially in the model group), the model's predictions for round one deviate considerably from the data. For the model group, an explanation can be found for this deviation; while the model can utilize the foreknowledge it received straight away, this was not necessarily true for the subjects in the actual experiment. As they received written instructions on the operating procedures, they still had to match the controls mentioned in these instructions onto the physical controls of the device. This process is not incorporated in the model. (Since the model interacts with a LISP simulation of the device, it can simply call the simulation using the conceptual name of the control).

Regarding the fit with the data for the no-model group, the model needs more actions than the subjects did. The no-model group did not have to match the conceptual names of the controls onto the physical controls since they did not know these conceptual names. Apparently the subjects managed to use strategies which are more sophisticated than the ones the model employs in this round. At present, no adequate reformulation of the model's stategies for this round has been found.

Conclusions

A simple state learning model combined with difference reduction provides a reasonable fit of the behaviour of subjects in later rounds. For the group with foreknowledge, the model needs fewer actions than subjects in round 1. One likely source of this difference is that the model does not need to map knowledge onto specific controls.

For the group that lacked foreknowledge, the model needed more actions than was true for the subjects. Even specifying the knowledge about the power switch did not improve the first round performance of the model to an extent that it was comparable with that of the subjects. Apparently, in the first round, subjects use a more efficient strategy than the one simulated by a combination of random performance and choosing the control whose name corresponds to the goal. In order to simulate the behaviour of the subjects with foreknowledge in later rounds, the strongest weight had to be assigned to those productions that retrieve previously succesful state-action pairs from memory. For the group without foreknowledge the strongest weight was assigned to the production that selects the control whose name corresponds to the present goal. This might be construed as the group with foreknowledge being more confident about the knowledge they had, and them actively trying to reduce the difference between the present state and a state which was known to lead to succes. The group without foreknowledge tends to rely more on a label-following heuristic; that is, they are more likely to select a control randomly, followed by the laser-activation button which corresponds to the present goal.

Comparison to existing models

Compared to existing models, the present model is considerably simpler, both in terms of the strategies incorporated in it as well as in the range of tasks it models. The basic learning mechanism (instance learning) is comparable to that employed in other models. The aim of the present model however, was the comparison with data from human subjects which was lacking from other models, and the comparison of the behaviour of subjects with differing amounts of foreknowledge. Though the amount and type of foreknowledge can be manipulated in Ayn and IDXL, no comparison with actual human data has been carried out. The present model and task are aimed at studying the role of foreknowledge at a finer grainsize than was done with these other models. This suggested the use of a different mix of strategies for the two groups.

Though this may ultimately be a feature of the task, there are some differences in the type of knowledge that Ayn and the present model use. Like Ayn, the present model uses recognition knowledge (succesful instances encountered), and some semantic-match knowledge (the relevant laser activation button has a higher likelihood of being used). Unlike the present model however, Ayn uses negative control knowledge. That is, Ayn will avoid actions that were previously unsuccesful. In the present model, this has not (yet) been necessary. Though a chunk is created for succesful as well as unsuccesful actions, the present model only uses succesful chunks to guide behaviour.

References

Anderson, J.R. & Lebiere, C. (1998): *The Atomic Components of Thought*. Mahwah, NJ, Lawrence Erlbaum Associates.

Card, S.K., Moran, T.P., & Newell, A. (1983): *The Psychology of Human-computer Interaction*. Hillsdale, N.J.: Lawrence Erlbaum Associates.

Dienes, Z. & Fahey, R. (1995): The Role of Specific instances in Controlling a Dynamic System. *Journal of Experimental Psychology: Learning, Memory & Cognition*, 21, 848-862.

Freudenthal, D. (1998): Learning to Use Interactive Devices; Age Differences in the Reasoning Process. Doctoral Dissertation, Eindhoven University of Technology.

Howes, A. (1994): A model of the Acquisition of Menu Knowledge by Exploration. *Proceedings of CHI 1994 (Boston, MA USA, April 24-28, 1994)* ACM, New York, 1994, pp.445-451.

Howes, A. & Young, R.M. (1996): Learning Consistent, Interactive, and Meaningful Task-Action Mappings: A computational Model. *Cognitive Science*, 20, 301-356.

Kitajima, M. & Polson, P.G. (1995): A comprehension-based model of correct performance and errors ik skilled, display-based, human-computer interaction. *International Journal of Human-computer Studies*, 43, 65-99.

Kitajima, M. & Polson, P.G. (1997): A comprehension-based model of exploration. *Human-Computer Interaction*, 12, 345-389.

Kieras, D.E. & Bovair, S. (1984): The Role of a Mental Model in Learning to Operate a Device. *Cognitive Science*, 8, 255-273.

Logan, G.D. (1988) Toward an Instance theory of automatization. *Psychological Review*, 95, N4, 492-527.

Payne, S.J. (1986): Task-Action grammars: A model of the mental representation of task languages. *Human computer Interaction*, 2, 93-133.

Polson, P.G & Lewis, C.H. (1990): Theory-Based Design for Easily Learned Interfaces. *Human-Computer Interaction*, 5, 191-220.

Rieman, J, Young, R.M. & Howes, A. (1996): A dual-space model of iteratively deepening exploratory learning. *International Journal of Human-computer Studies*, 44, 747-775.