Interaction as an emergent property of a Partially Observable Markov Decision Process

Andrew Howes, Xiuli Chen, Aditya Acharya
School of Computer Science,
University of Birmingham

Richard L. Lewis
Department of Psychology
University of Michigan

Contact: HowesA@bham.ac.uk

July 2017

Contents

0.1 Introduction 4
0.2 Partially Observable Markov Decision Processes 6
0.2.1 A technical overview 7
0.2.2 Using POMDPs to predict interaction 9
0.2.3 A caveat: Solving POMDPs is difficult 10
0.3 Menu search as a POMDP 11
0.3.1 POMDP formulation 12
0.3.2 Belief Update 13
0.3.3 Learning 13
0.3.4 Predicting menu search 14
0.3.5 Testing the model against human data 17
0.3.6 Discussion 18
0.4 Interactive decision making as a POMDP 19
0.4.1 POMDP formulation 21
0.4.2 Belief update 23
0.4.3 Learning 23
0.4.4 A typical decision making task 24
0.4.5 Behaviour of the model 25
0.4.6 Implications for decision strategies 26
0.5 Discussion 27

Abstract In this chapter we explore the potential advantages of modeling the *interaction* between a human and a computer as a consequence of a Partially Observable Markov Decision Process (POMDP) that models human cognition. POMDPs can be used to model human perceptual mechanisms, such as human vision, as partial (uncertain) observers of a hidden state are possible. In general, POMDPs permit a rigorous definition of interaction as the outcome of a reward maximizing stochastic sequential decision processes. They have been shown to explain interaction between a human and an environment in a range of scenarios, including visual search, interactive search and sense-making. The chapter uses these scenarios to illustrate the explanatory power of POMDPs in HCI. It also shows that POMDPs embrace the embodied, ecological and adaptive nature of human interaction.

0.1 Introduction

A Partially Observable Markov Decision Process (POMDP) is a mathematical framework for modelling sequential decision problems. We show in this chapter that a range of phenomena in Human-Computer Interaction can be modeled within this framework and we explore its strengths and weaknesses. One important strength of the framework is that it embraces the highly adaptive, embodied and ecological nature of human interaction (?, ?, ?, ?, ?, ?, ?) and it thereby provides a suitable means of rigorously explaining a wide range of interaction phenomena.

Consider as an illustration a task where a user searches for an image in the results returned by a web search engine (Figure 1). ? (?) studied a version of this task in which a person has the goal of finding an image with a particular set of features, for example to find an image of a castle with water and trees. A user with this task must make multiple eye movements and fixations because the relatively high resolution fovea is sufficient to provide the details of only about 2.5 degrees of visual angle. Eventually, after a sequence of these *partial observations* the user might find a relevant image (e.g. the one in the bottom right of the figure). Evidence shows that people do not perform a search such as this using a systematic top-left to bottom-right strategy; they do not start at the top left and then look at each image in turn. But, equally, they do not search randomly. Rather, the search is a rational adaptation to factors that include the ecological distribution of images and the partial observations provided by the visual fixations. We

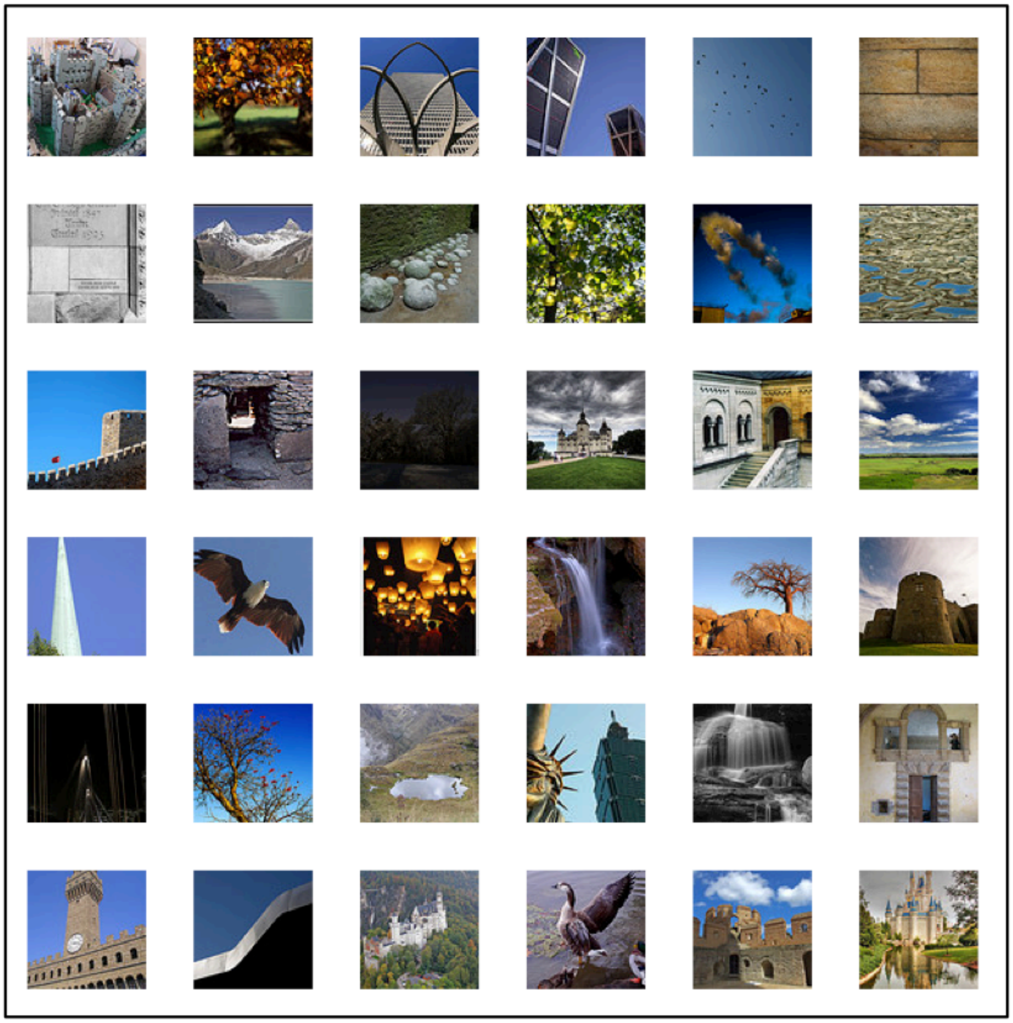


Figure 1: Simulated results from a search engine.

show in this paper that rational strategies like these can be modelled as the emergent solution to a POMDP.

The approach that we take is heavily influenced by a long tradition of computational models of human behaviour (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?). A key insight of this work has been to delineate the contribution to interactive behaviour of information processing capacities (e.g. memory, perception, manual movement times), on the one hand, and strategies (methods, procedures, policies), on the other. A contribution of the POMDP approach is that strategies emerge through learning given a formal specification of the interaction problem faced by the user where the problem includes the bounds imposed by their individual capacities.

To illustrate the contribution of POMDPs, we examine two examples of the application of POMDPs to explaining human computer interaction. The first is a model of how people search menus and the second of how they use visualizations to support decision making. Before describing the examples, we first give an overview of POMDPs. In the discussion, we explore the potential advantages and disadvantages of explaining interaction as an emergent consequence of a POMDP. The advantages include, (1) that the POMDP framing is a well-known and rigorous approach to defining stochastic sequential decision processes and there is a growing range of machine learning algorithms dedicated to solving them, (2) that POMDPs provide a means of defining and integrating theoretical concepts in HCI concerning embodiment, ecology and adaptation, and (3) that POMDPs provide a means to make inferences about the consequences of theoretical assumptions for interaction. The disadvantages concern tractability and the existing scope of application to human tasks.

0.2 Partially Observable Markov Decision Processes

Originally conceived in operations research (?, ?), POMDPs have been influential in Artificial Intelligence (?, ?). They provide a mathematical framework for formalizing the interaction between an agent and a stochastic environment, where the state of the environment is not fully known to the agent. Instead, the agent receives observations that are partial and stochastic and that offer evidence as to the state. The problems faced by the agent are therefore to (1) generate a good estimate of the state of the environment given the history of actions and observations, and (2) to generate actions that maximize the expected reward gained through interaction.

POMDPs have also been used in Cognitive Science to explain human behaviour. For example, in one contribution to explaining human vision, a target localization task was framed as a POMDP problem by ? (?). Butko et al. studied a task in which a visual target was present in one position in a grid and participants had to localize the target by moving their eyes to gather information. The information received at each time step, a partial observation, was from the fixated point (with high reliability) and surrounding points (with lower reliability). Butko et al.'s model learned a series of eye movements, a strategy, that maximized the reward gained from performing the task (it performed fast and accurate localization). The authors showed how this learned strategy could generate human-like behaviour without assuming ad-hoc heuristics such as inhibition-of-return to previous locations. They also showed that the learned strategy could sometimes contradict what might seem intuitively good strategies. For example, the model did not always fixate the location with the highest probability of containing the target. Instead, it sometimes it preferred to look just to the side of the target so as to gather more information about multiple potential targets with peripheral vision.

POMDPs have also been used to explain the behaviour of non-human species. A random dot motion discrimination task was framed as a POMDP problem by ? (?). In this task, primates were shown a visual display containing a group of moving dots. A fraction of the dots moved in one of two fixed directions (left or right) and other dots moved in random directions. The primates were rewarded for determining the direction in which the majority of dots moved, where the number of random dots was manipulated. ?'s (?) model showed that primate decision times could be modelled as a reward maximizing solution to a POMDP. The model determined the optimal threshold for switching from information gathering to decision.

0.2.1 A technical overview

A POMDP defines a problem in which an agent takes a sequence of actions under uncertainty to maximize its reward. Formally, a POMDP is specified as a tuple $< \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, \mathcal{Z}, \mathcal{R}, \gamma >$ (Figure 2), where $\mathcal{S}$ is a set of states; $\mathcal{A}$ is a set of actions; and $\mathcal{O}$ is a set of observations. At each time step t the agent is in a state $s_t \in \mathcal{S}$, which is not directly observable to the agent. The agent takes an action $a_t \in \mathcal{A}$, which results in the environment moving from s_t to a new state s_{t+1}. Due to the uncertainty in the outcome of an action, the next state s_{t+1} is modelled as a conditional probability function $T(s_t, a_t, s_{t+1}) = p(s_{t+1}|s_t, a_t)$, which gives the probability that the agent lies in s_{t+1}, after taking action a_t in state s_t (red arrows in Figure 2). The agent then makes an observation to gather information about the state. Due

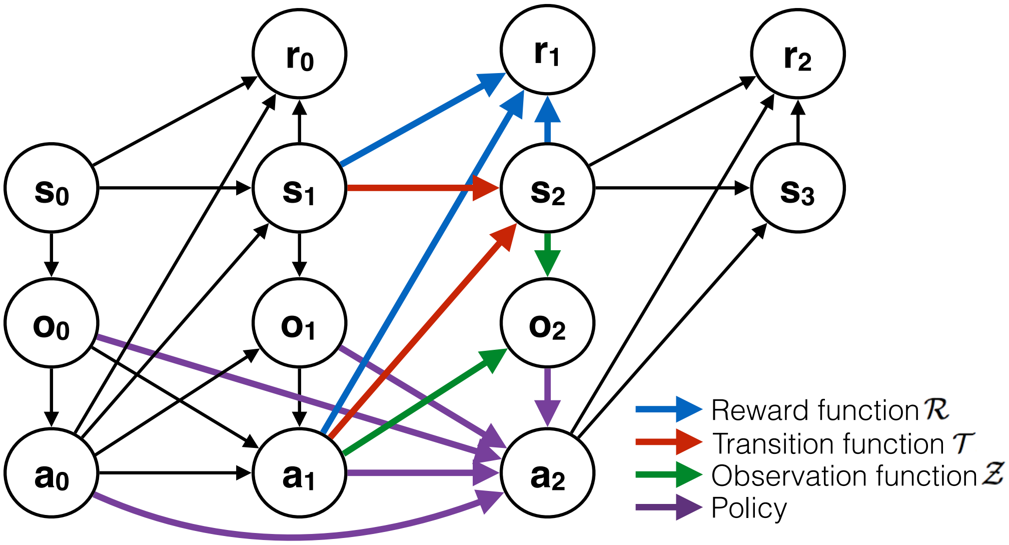

r0
r1
r2
s0
s1
s2
s3
o0
o1
o2
a0
a1
a2
Reward function $\mathcal{R}$
Transition function $\mathcal{T}$
Observation function $\mathcal{Z}$
Policy

Figure 2: General representation of a POMDP

to the uncertainty in observation, the observation result $o_{t+1} \in \mathcal{O}$ is also modelled as a conditional probability function $Z(s_t, a_t, o_{t+1}) = p(o_{t+1}|s_t, a_t)$ (green arrows in Figure 2). Some agents are programmed to keep a history of action-observations pairs $h =< a_0, o_0, a_1, o_1, ... >$ that are used to inform action selections (purple arrows in Figure 2). In other agents, the history is replaced by a Bayesian Belief B distribution over possible states of the world and the observations are used to update this belief.

In each step t, the agent receives a real-valued reward $r_t = R(s_t, a_t, s_{t+1})$ if it takes action a_t in state s_t and results in s_{t+1} (blue arrows in Figure 2). The goal of the agent is to maximize its expected total reward by choosing a suitable sequence of actions. A discount factor $\gamma \in [0, 1)$ is specified so that the total reward is finite and the problem is well defined. In this case, the expected total reward is given by $\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)$, where s_t and a_t denote the agents state and action at time t. The solution to a POMDP is an optimal policy (what to do when) that maximizes the expected total reward.

Applied to the visual search problem studied by ? (?) (Figure 1) the problem is to maximize a reward function R in which finding images with a higher number of task-matching features than images with fewer matching features gains a higher reward. However, eye-movements and fixations incur a negative reward (a cost) that is proportional to the time taken. Therefore, maximizing reward means finding a sequence of action that trades more matching features against time. Each state s in this task might consist of a representation of the 36 images on the display and the location of the current fixation. The images might be represented as a bitmap or in terms of a more abstract symbolic feature vector. The actions A might include eye movements, mouse movements, and button presses. Again these might be abstracted to just include fixation locations and selections. An observation O would encode information from the fixated location (to 2.5 degrees of visual angle) with high reliability and information from the periphery with lower reliability according to the observation function Z. The transition function T models consequences of actions in A such as changing the fixation location. The transition function also models the reliability of action. For example, users might intend to fixate the second image from the left in the third row, but with a small probability fixate an adjacent image.

The usefulness of the assumption that users can be modelled with POMDPs rests on the ability of researchers to find psychologically valid definitions of $< \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, \mathcal{Z}, \mathcal{R}, \gamma >$ and the use of machine learning algorithms to find approximately optimal strategies. Finding optimal strategies is computationally hard and often requires the application of the latest machine learning methods. Lastly, the usefulness of POMDPs also rests on demonstrating some correspondence between the behaviour generated from the learned optimal strategies and human behaviour.

0.2.2 Using POMDPs to predict interaction

What should be clear from the previous section is that defining a user's interaction problem as a POMDP is not in-and-of-itself sufficient to predict human behaviour. What is also required is a solution to the POMDP. The solution is a policy that makes use of the available resources to efficiently move through the state space to a desired goal state. While POMDPs often define very large and sometimes intractable state spaces, machine learning methods can sometimes be used to find efficient policies (?, ?).

A concern in machine learning is with finding approximately optimal policies. These are policies that come close to maximizing the reward signal through task performance. Following ? (?), we assume here that what is known in machine learning as a policy corresponds to what in HCI is often known as a strategy. In HCI it is often observed that users make use of a wide range of strategies and much HCI research has been aimed at uncovering these strategies and explaining why some are used in some circumstances and other elsewhere (?, ?, ?, ?). One finding is that users find strategies that are finely tuned to the particulars of the context and the bounds imposed by their own individual embodiment (?, ?). In fact, users might be so good at finding efficient strategies that, which strategies they find might be predicted by the approximately optimal policies that solve well-defined POMDPs.

In tasks such as the visual search task in Figure 1 adaptation occurs within the bounds imposed by the human visual system (?, ?), which are part of the problem definition. While the performance is bounded by these constraints, it is possible for people to find highly efficient strategies that trade the minimal number of eye movements for the highest quality image. ? (?) found that when people performed this task in a laboratory they appeared to be computationally rational (?, ?, ?). In other words, people were as efficient as they could given the computational limits of their visual system.

There has been much research in HCI and cognitive science demonstrating computationally rational adaptation (?, ?, ?, ?, ?, ?, ?, ?, ?, ?). These analyses take into account the costs and benefits of each action to the user so as to compose actions into efficient behavioural sequences. One prominent example of this approach is Card's cost of knowledge function (?, ?). In addition, there are models of multitasking in which the time spent on each of two or more tasks is determined by their relative benefits and time costs (?, ?). This literature supports the general idea that finding reward maximizing policies that solve well-defined POMDPs is a promising approach to modeling interaction.

0.2.3 A caveat: Solving POMDPs is difficult

The POMDP formalism is very general and is applicable to a diverse range of problems (?, ?, ?). Unfortunately, the generality of POMDPs can result in high computational cost for deriving optimal control policies. As a consequence research on POMDPs is often focused on general methods for finding approximate solutions (?, ?). Recent work on Deep Q-Networks is also a promising avenue for application to POMDPs. For example, ? (?) have shown that Deep Q-Networks are capable of learning human-level control policies on a variety of different Atari 2600 games.

0.3 Menu search as a POMDP

Imagine that a goal for a user who is experienced with menus, but who has never used Apple's OS X Safari browser before, is to select 'Show Next Tab' from the Safari Window menu. A user might solve this goal by first fixating the top menu item, encoding the word 'Minimize'; rejecting it as irrelevant to the target, moving the eyes to the next group of items, that begins 'Show Previous Tab', noticing that this item is not the target but is closely related and also noticing, in peripheral vision, that the next item has a similar word shape and length to the target; then moving the eyes to 'Show Next Tab', confirming that it is the target and selecting it.

In this section we show how interactive behaviours such as these can be predicted by solving a POMDP. Importantly, the aim is not to predict the interactions that people learn with specific menus and the location of specific items, rather the aim is to predict how people will perform interactive menu search for newly experienced menus. The requirement is that the model should learn, from experience, the best way to search for new targets in new, previously unseen, menus.

? (?) hypothesized that a key property of the menu search task is that human search strategies should be influenced by the distribution of relevance across menus. If highly semantically relevant items are rare then the model should learn to select them as soon as they are observed, whereas if they are very common then they are less likely to be correct, and the model should learn to gather more evidence before selection. The goal for the model, therefore, is not just to learn how to use a single menu, but rather it is how to use a new menu that is sampled from an experienced distributions of menus.

To achieve this goal ? (?) built a computational model that can be thought of as a simple POMDP. In the model an external representation of the displayed menu is fixated and an observation is made that encodes information about the relevance of word shapes ('Minimize' and 'Zoom', for example have different lengths) and semantics (word meanings). This observation is used to update a vector representing a summary of the observation history (a belief about the state). This vector has an element for the shape relevance of every item in the menu, an element for the semantic relevance of every item in the menu, and an element for the current fixation location. The vector elements are null until estimates are acquired through observation. An observation is made after each fixation action, e.g. after fixating

'Minimize' in the above example. After having encoded new information through observation, the policy chooses an action on the basis of the current belief. The chosen action might be to fixate on another item or to make a selection, or to exit the menu if the target is probably absent. Belief-action values are updated incrementally (learned) as reward feedback is received from the interaction.

0.3.1 POMDP formulation

A state s is represented as a vector consisting of n elements for the shape, n for the semantic relevance, and 1 for the fixation location. The semantic/alphabetic relevance had 5 levels [Null, 0, 0.3, 0.6, 1]. The shape relevance had 2 levels [0 for non-target length; 1 for target length]. The fixation was an integer representing one of the menu item locations [1..n]. From each state there were $n+2$ actions in A, including n actions for fixating on n menu item locations, an action for selecting the fixated item, and an action to exit the menu without selection (target absent). It was assumed that the consequences of actions were entirely reliable and, therefore, the transition function T had probability 1.

An observation o_t modelled human vision by encoding information from the fovea with high reliability and information from the periphery with lower reliability according to the observation function $\mathcal{Z}$. ? (?) modelled the observations with which people determine the semantic relevance of items by matching them to the goal specification. To implement this assumption, they used relevance ratings gathered from human participants and reported by ? (?). They give the following example: if the model sampled the goal Zoom and foveated the word Minimize then it could look-up the relevance score 0.75 which was the mean relevance ranking given by participants. ?'s (?) model also observed the length of each menu item (0 for non-target length; 1 for target length). Observations of alphabetic relevance were determined using the distance apart in the alphabet of target and fixated first letters. This was then standardized to a four-level scale between 0 and 1, i.e., [0, 0.3, 0.6, 1]. Further details are reported in ? (?).

Visual acuity is known to reduce with eccentricity from the fovea (?, ?). In ?'s (?) model, the acuity function was represented as the probability that a visual feature was recognized. Semantic information was available with probability 0.95 at the fovea and probability 0 elsewhere. The model made use of semantic features and shape features but could easily be enhanced with other features such as colour. These parameter settings resulted in the following availability probabilities: 0.95 for the item fixated, 0.89 for items immediately above or below the fixated item, and 0 for items further away. On each fixation, the availability of the shape information was determined by these probabilities.

? (?) defined rewards for saccades and fixations in terms of their durations as determined by the psychological literature. Saccades were given a duration that was a function of distance. Fixations were given an average duration measured in previous work on menus. A large reward was given for successfully finding the target or correctly responding that it was absent.

0.3.2 Belief Update

The belief-state B was a vector with the same number of elements as the state S, but these elements initially had null values. These values were updated through observation.

0.3.3 Learning

?'s (?) model solved the POMDP using Q-learning. The details of the algorithm are not described here but they can be found in any standard Machine Learning text (e.g. ? (?)). Q-learning uses the reward signal, defined above, to learn state-action values (called Q values). A state-action value can be thought of as a prediction of the future reward (both positive and negative) that will accrue if the action is taken. Before learning, an empty Q-table was assumed in which all state-action values were zero. The model therefore started with no control knowledge and action selection was entirely random. The model was then trained until performance plateaued (requiring 20 million trials). On each trial, the model was trained on a menu constructed by sampling randomly from the ecological distributions of shape and semantic/alphabetic relevance. The model explored the action space using an ϵ-greedy exploration. This means that it exploited the greedy/best action with a probability $1-\epsilon$, and it explored all the actions randomly with probability ϵ. q-values were adjusted according to the reward feedback. The (approximately) optimal policy acquired through this training was then used to generate the predictions described below.

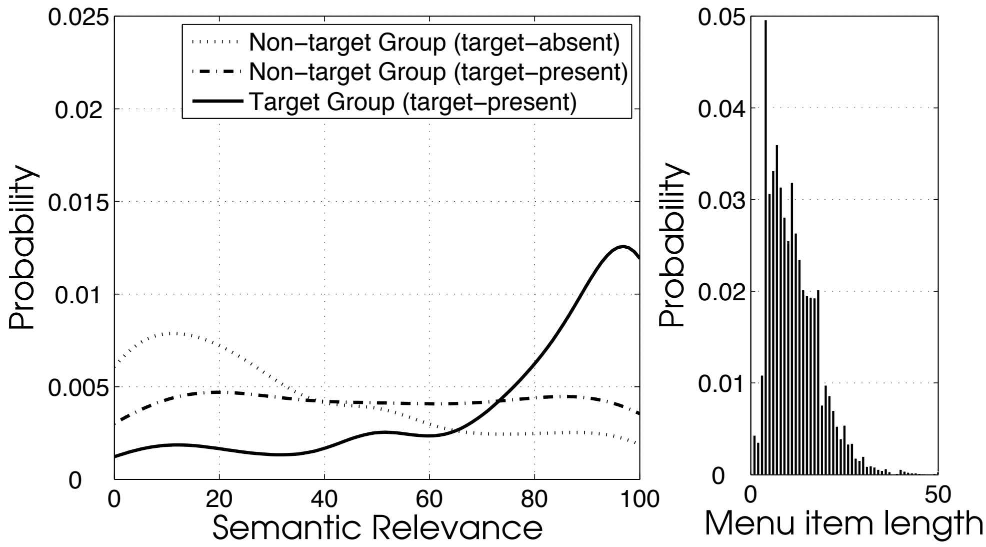


Figure 3: Menu ecology of a real-world menu task (Apple OS X menus). Left panel: The distribution of semantic relevance. Right panel: The distribution of menu length.

0.3.4 Predicting menu search

The purpose of this first study was to examine the model's predictions on commonly used computer application menus. The task was to search for specific target items in a vertically arranged menu. How items were organized in the menu was varied. Items could be unorganized, alphabetically organized, or semantically organized. To determine the statistical properties of a menu search environment we used the results of a previous study (?, ?) in which the menus of 60 applications from Apple OS X were sampled. Together these applications used a total 1049 menus, and 7802 menu items. ? (?) used these to determine the ecological distribution of menu length, item length, semantic group size and first letter frequencies (for alphabetic search). The probability of each length (number of characters) observed by ? (?) is reproduced in Figure 3 right panel. The distribution is skewed, with a long tail of low probability longer menu items.

? (?) then ran a study in which participants rated how likely two menu items were to appear close together on a menu. Each participant rated pairs sampled from the Apple OS X Safari browser menus. The probability of each semantic relevance rating is shown in Figure 3 left panel. These ratings

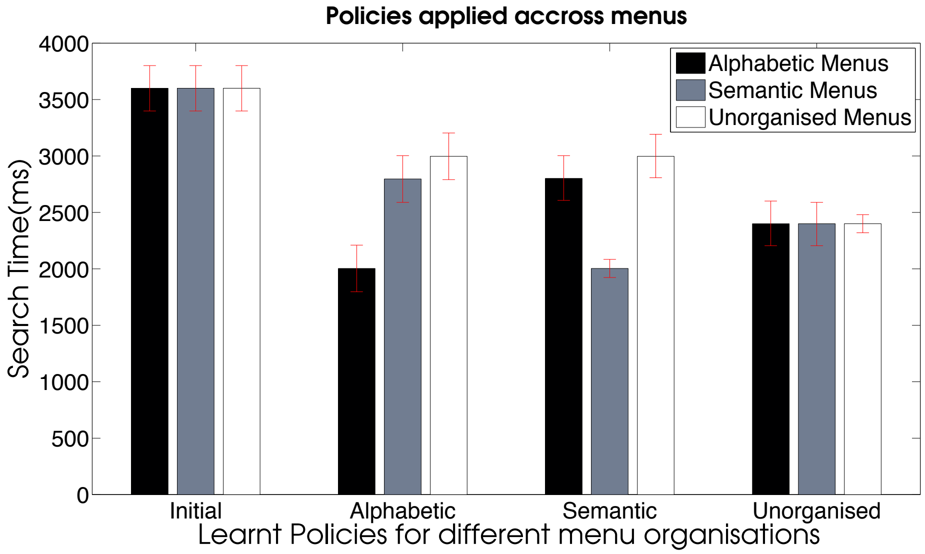


Figure 4: The search duration taken by the optimal strategy for each type of menu (95% confidence intervals (C.I.s))

were used by ? (?) both to construct example menus and to implement the model's semantic relevance function.

The first set of results reported by ? (?) were for the approximately optimal strategy (after learning), unless stated otherwise. The optimal policy achieved 99% selection accuracy. The utility of all models plateaued, suggesting that the learned strategy was a good approximation to the optimal strategy.

Figure 4 is a plot of the duration required for the optimal policy to make a selection given four types of experience crossed with three types of test menu. The purpose of this analysis is to show how radically different patterns of behaviour emerge from the model based on how previously experienced menus were organized. Prior to training (the left most *Initial* set of three bars in the figure), the model offers the slowest performance; it is unable to take advantage of the structure in the alphabetic and semantic menus because it has no control knowledge. After training on a distribution of *Unorganized* menus (far right set in the figure), performance time is better than prior to training. However, there is no difference in performance time between the different menu organizations. After training on a distribution of semantically organized menus (middle right set in the figure), the model

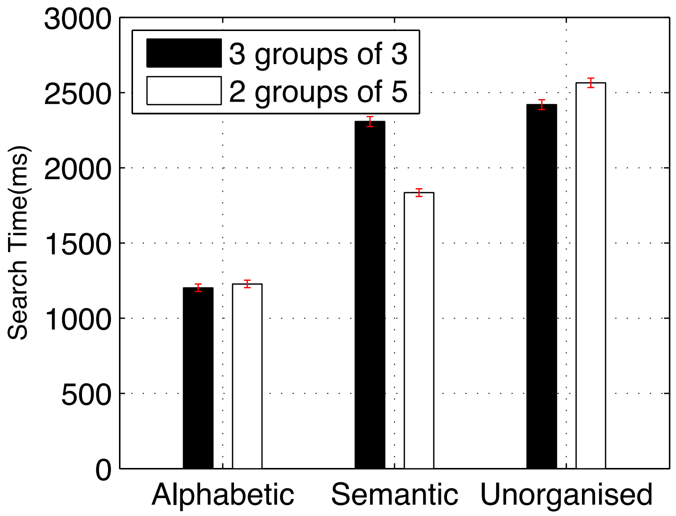


Figure 5: The effect of semantic group size (95% C.I.s).

is able to take advantage of semantic structure, but this training is costly to performance on alphabetic and unorganized menus. After training on a distribution of alphabetically organized menus (middle left set in the figure), the model is able to take advantage of alphabetic ordering, but this training is again costly to the other menu types. The optimal policy must switch the policy depending on the menu type.

Figure 5 shows the effect of different semantic groupings on performance time (reported by ? (?). It contrasts the performance time predictions for menus that are organized into 3 groups of 3 or into 2 groups of 5. The contrast between these kinds of design choices has been studied extensively before (?, ?). What has been observed is an interaction between the effect of longer menus and the effect of the number of items in each semantic group (See ?'s (?) Figure 8). As can be seen in Figure 5 while the effect of longer menus ($3 \times 3 = 9$ versus $2 \times 5 = 10$) is longer performance times in the unorganized and alphabetic menus, the effect of organization (3 groups of 3 versus 2 of 5) gives shorter performance times in the semantic condition. This prediction corresponds closely to a number of studies (See ?'s (?) Figure 8).

The results show that deriving adaptive strategies using a reinforcement learning algorithm, given a definition of the human menu search problem as a POMDP, bounded by the constraints of the human visual systems (em-

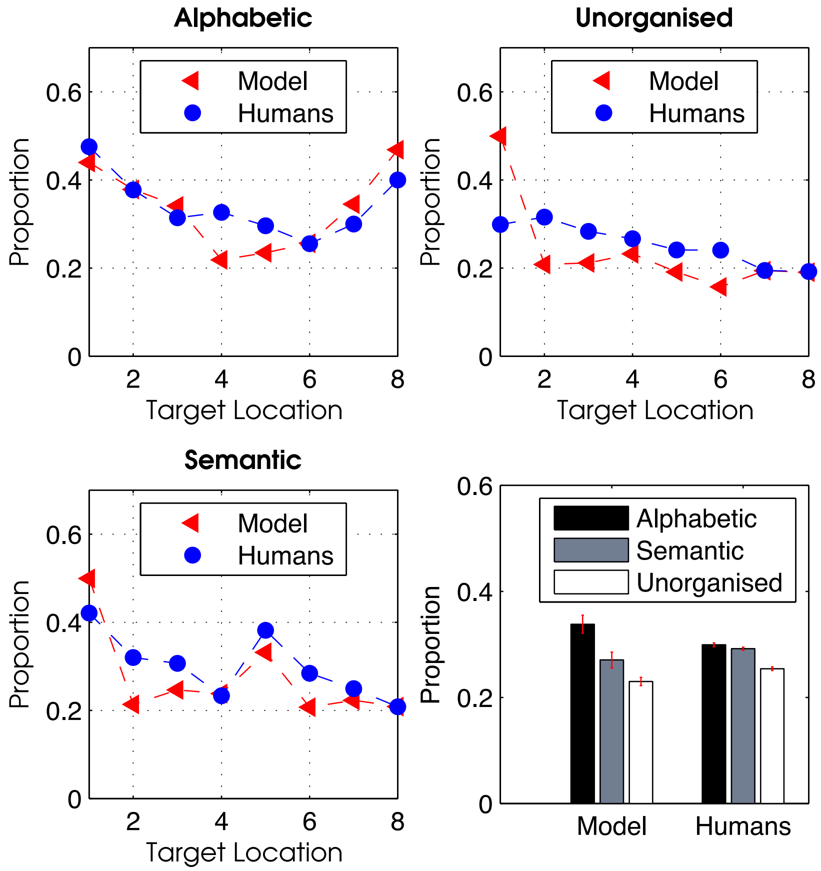


Figure 6: The proportion of gazes on the target location for each of the three types of menu (95% C.I.s).

bodiment) and the ecology of the task environment (ecology), can lead to reasonable predictions about interactive behaviour.

0.3.5 Testing the model against human data

? (?) also tested the model against human data (?, ?, ?). These data were collected using laboratory menus with different distributions to those in the other study. The model was trained on these distributions and the optimal policy was then used to generate the predictions described in the following results section.

Target location effect on gaze distribution

Figure 6 shows the effect of target position on the distribution of item gazes for each menu organization. The model is compared to human data reported in (?, ?). The adjusted R^2 for each of the three organizations (alphabetic, unorganized, semantic) were reported as $0.84, 0.65, 0.80$. In the top left panel, the model's gaze distribution is a consequence of both alphabetic anticipation and shape relevance in peripheral vision. Interestingly, both the model and the participants selectively gazed at targets at either end of the menu more frequently than targets in the middle. This may reflect the ease with which words beginning with early and late alphabetic words can be located. In the top right panel, there is no organizational structure to the menu and the model's gaze distribution is a consequence of shape relevance only in peripheral vision. The model offers a poor prediction of the proportion of gazes on the target when it is in position 1, otherwise, as expected, the distribution is relatively flat in both the model and the participants. In the bottom left panel, the model's gaze distribution is a function of semantic relevance and shape relevance. Here there are spikes in the distribution at position 1 and 5. In the model, this is because the emergent policy uses the relevance of the first item of each semantic group as evidence of the content of that group. In other words, the grouping structure of the menu is evidence in the emergent gaze distributions. The aggregated data is shown in the bottom right panel; the model predicts the significant effect of organization on gaze distribution, although it predicts a larger effect for alphabetic menus than was observed.

0.3.6 Discussion

? (?) reported a model in which search behaviours were an emergent consequence of an interaction problem specified as a POMDP. The observation function and state transition function modelled the cognitive and perceptual limits of the user, and the reward function modelled the user's preferences for speed and accuracy. Predicted strategies were generated using machine learning to infer an approximately optimal policy for the POMDP. The model was tested with two studies. The first study involved applying the model to a real world distribution of menu items and in the second study the model was compared to human data from a previously reported experiment.

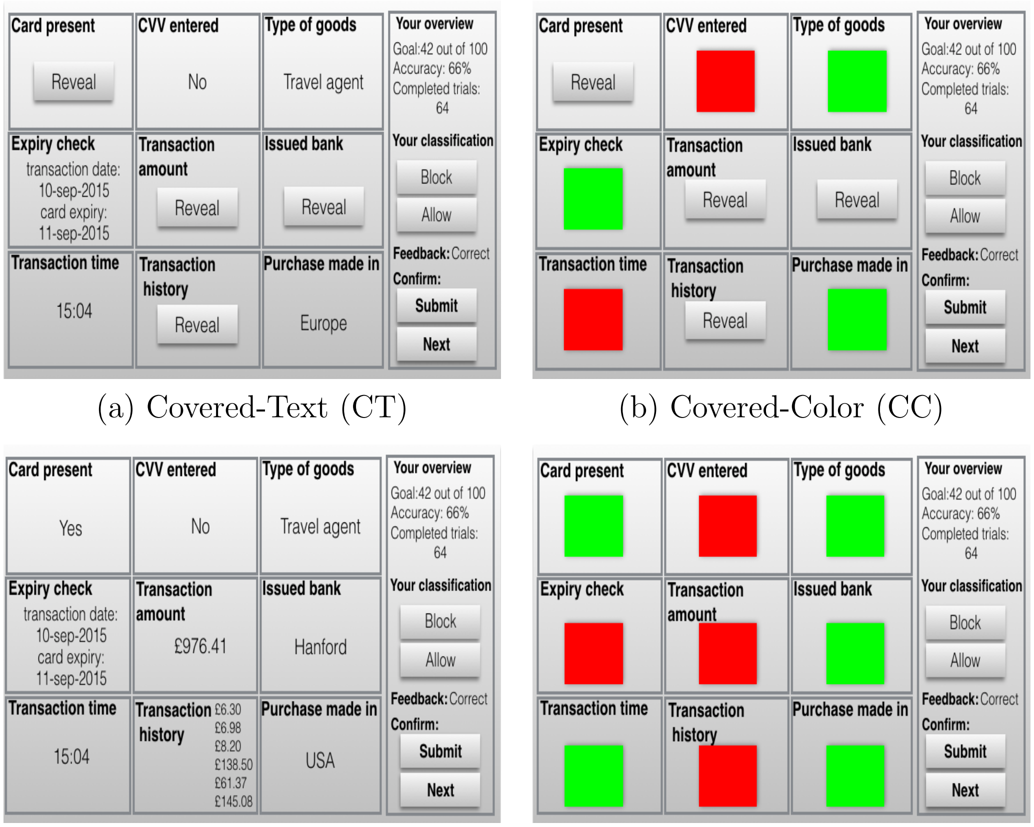
Unlike in previous models, no assumptions were made about the gaze strategies available to or adopted by users, Instead, a key property of the approach is that behavioural predictions are derived by maximizing utility given a quantitative theory of the constraints on behaviour, rather than by maximizing fit to the data. Although this *optimality* assumption is sometimes controversial, the claim is simply that users will do the best they can with the resources that are available to them. Further discussions of this issue can be found in (?, ?, ?, ?).

0.4 Interactive decision making as a POMDP

The previous example of how a POMDP can be used to model interaction focused on a relatively routine low-level task. In this section we look at a model reported by ? (?) of a higher-level decision making task. The model shows how decision strategies are an emergent consequence of both the statistical properties of the environment (the experienced cue validities, different time cost for extracting information) and of the constraints imposed by human perceptual mechanisms. For a decision making task that is supported by visualization, this theory can be written precisely by formulating the decision problem as a POMDP for active vision and solving this problem with machine learning to find an approximately optimal decision strategy (emergent heuristics). The model offers an alternative theory of decision making to the heuristic model of ? (?).

In the resulting model, eye movement strategies and stopping rules are an emergent consequence of the visualization and the limits of human vision (modelled with an observation function) (?, ?, ?, ?). The assumption is that people choose which cues to look at and when to stop looking at cues informed by the reward/cost that they receive for the decisions they make. Better decisions will receive higher rewards, which will reinforce good eye movement strategies.

Before introducing ?'s (?) POMDP theory of decision making through interaction, we briefly described the task, the credit card fraud detection task, that they used to illustrate the theory. The task is motivated by a real-world scenario that is known to be extremely difficult for people. Credit card fraud detection analysts attempt to identify fraudulent patterns in transaction data-sets, often characterized by a large number of samples, many dimensions and online updates (?, ?). Despite the use of automated detection algorithms, there continue to be key roles for people to play in the analy-



Card present Reveal	CVV entered No	Type of goods Travel agent	Your overview Goal:42 out of 100 Accuracy: 66% Completed trials: 64
Expiry check transaction date: 10-sep-2015 card expiry: 11-sep-2015	Transaction amount Reveal	Issued bank Reveal	Your classification Block Allow
Transaction time 15:04	Transaction history Reveal	Purchase made in Europe	Feedback: Correct Confirm: Submit Next

(a) Covered-Text (CT)

Card present Reveal	CVV entered 🟥	Type of goods 🟩	Your overview Goal:42 out of 100 Accuracy: 66% Completed trials: 64
Expiry check 🟩	Transaction amount Reveal	Issued bank Reveal	Your classification Block Allow
Transaction time 🟥	Transaction history Reveal	Purchase made in 🟩	Feedback: Correct Confirm: Submit Next

(b) Covered-Color (CC)

Card present Yes	CVV entered No	Type of goods Travel agent	Your overview Goal:42 out of 100 Accuracy: 66% Completed trials: 64
Expiry check transaction date: 10-sep-2015 card expiry: 11-sep-2015	Transaction amount £976.41	Issued bank Hanford	Your classification Block Allow
Transaction time 15:04	Transaction history £6.30 £6.98 £8.20 £138.50 £61.37 £145.08	Purchase made in USA	Feedback: Correct Confirm: Submit Next

(c) Visible-Text (VT)

Card present 🟩	CVV entered 🟥	Type of goods 🟩	Your overview Goal:42 out of 100 Accuracy: 66% Completed trials: 64
Expiry check 🟥	Transaction amount 🟥	Issued bank 🟩	Your classification Block Allow
Transaction time 🟩	Transaction history 🟥	Purchase made in 🟩	Feedback: Correct Confirm: Submit Next

(d) Visible-Color (VC)

Figure 7: Four interface variants for credit card fraud detection. The information cues are represented with text (left panels) or color (right panels) and the information is either immediately available (bottom panels) or revealed by clicking the 'Reveal' buttons (top panels).

sis process. These roles range from defining and tuning the algorithms that automatic systems deploy, to triaging and screening recommendations from such systems, to contacting customers (either to query a transaction or to explain a decision). In terms of triaging and screening, we assume that an automated detection process is running and that this process has flagged a given transaction (or set of transactions) as suspicious and a user will engage in some form of investigation to decide how to respond to the flag. Based on pilot interviews and discussions with credit card fraud analysts and organizations, we believe that there are several ways in which the investigation could be performed. In some instances, the investigation could involve direct contact with the card-holder, in which the caller follows a predefined script and protocols that do not involve investigative capabilities. In some cases, these are passed to the analyst who needs to make a decision as to whether or not the credit card is blocked (this is the approach assumed in this paper). In this instance, the analyst would take a more forensic approach to the behaviour of the card holder and the use of the card, relative to some concept of normal activity. In some cases, investigation could be at the level of transactions, in which the analyst seeks to identify patterns of criminal activity involving several cards. In this instance, the analysis would be looking for evidence of stolen details or unusual patterns of use of several cards, say multiple transactions in different locations within a short time-frame. Other functions that people can perform in the fraud detection process include: risk prioritization, fast closure of low risk cases, documentation of false positives (?, ?), and identification of risk profiles and fraud patterns (?, ?, ?).

? (?) used a simplified version of fraud detection in which the task was to decide whether a transaction should be blocked (prevented from being authorized) or allowed. Participants were provided with 9 sources of information (cues) and these were presented using one of 4 display designs (visualizations). The cues differed in the reliability with which they determine whether or not a transaction is a fraud and the participants must discover these validities with experience and decide which cues are worth using to make a decision.

0.4.1 POMDP formulation

At any time step t, the environment is in a state $s_t \in S$. A state represents a true information pattern presented on the user interface. As shown in Figure 7, nine cues associated with credit card transactions are presented on the interface. The value of each cue was discretized into two levels, representing 'fraudulent (F)' and 'normal (N)' respectively. Therefore, for example, one of the states is a 9-element vector [F,N,N,F,F,F,N,F,N], each item of which represents the value for one of the cues ('F' for fraudulent and 'N' for normal). Hence, the size of the state space is $2^9 = 512$.

An action is taken at each time step, $a_t \in A$. The action space, A, consists of both the information gathering actions (i.e., which cue to fixate) and decision making actions (i.e., Block/Allow transaction). Therefore, the size of the action space is 11 (9 cues plus 2 decision actions).

At any moment, the environment (in one of the states s) generates a reward (cost if the value is negative), $r(s, a)$, in response to the action taken a. For the information gathering actions, the reward is the time cost (the unit is seconds). The time cost includes both the dwell time on the cues and the saccadic time cost of travelling between cues. The dwell durations used in the model were determined from experimental data. In the experiment to be modelled (described below), the participants were asked to complete 100 *correct* trials as quickly as possible, so that errors were operationalized as time cost. In the model, the cost for incorrect decisions is based on participants' average time cost (Seconds) for a trial (CT:17 ± 15; CC:24 ± 7;VT:20 ± 8; VC:13 ± 3). That is, the penalty of an incorrect trial is the time cost for doing another trial.

In addition to the reward, another consequence of the action is that the environment moves to a new state according to the transition function. In the current task the states (i.e. displayed information patterns) stay unchanged across time steps within one trial. Therefore, $T(S_{t+1}|S_t, A_t)$ equals to 1 only when $S_{t+1} = S_t$. $T(S_{t+1}|S_t, A_t)$ equals 0 otherwise. That is, the state transition matrix is the identity matrix.

After transitioning to a new state, a new observation is received. The observation, $o_t \in O$, is defined as the information gathered at the time step t. An observation is a 9-element vector, each element of which represents the information gathered for one of the cues. Each element of the observation has three levels, F (fraudulent), N (normal) and U (unknown). For example, one observation might be represented as [F,N,U,F,U,U,U,N,N]. Therefore the upper bound on the observation space is $3^9 = 19683$.

For the observation function $p(O_t|S_t, A_t)$ The availability of information about a cue is dependent on the distance between this cue and the fixation location (eccentricity). In addition, it is known that an object's colour is more visible in the periphery than the object's text label (?, ?). In our model, the observation model is based on the acuity functions reported in (?, ?), where the visibility of an object is dependent on, for example, the object size, the object feature (colour or text), and the eccentricity.

The observation obtained is constrained by a theory of the limits on the human visual system. The model assumed that the text information was obtained only when it was fixated. The colour information was obtained based on the colour acuity function reported in (?, ?). This function was used to determine the availability of the colour information for each cue given the distance between the cues and the fixated location (called *eccentricity*), and the size of the item. Specifically, on each fixation, the availability of the colour information was determined by the probabilities defined in Equation (1).

$$P(available) = P(size + X > threshold) \tag{1}$$

where *size* is the object size in terms of visual angle in degrees; $X \sim \mathcal{N}(size, v \times size)$; $threshold = a \times e^2 + b \times e + c$; e is eccentricity in terms of visual angle in degrees. In the model, the function were set with parameter values of v=0.7, b=0.1, c=0.1, a=0.035 as in (?, ?).

0.4.2 Belief update

At each time step, the environment is in a state s_i, which is not directly observed. The model maintains a belief b about the state given the sequence of observations. Every time the agent takes an action a and observes o, b is updated by Bayes' rule. At each time t, a belief b_t vector consists of a probability for each of the states, $b_t(s_i)$, where $i \in 1, 2, 3, ...N_s$ and N_s is the total number of states.

0.4.3 Learning

Control knowledge was represented in ?'s (?) model as a mapping between beliefs and actions, which was learned with Q-learning (?, ?). Further details of the algorithm can be found in any standard Machine Learning text (e.g.(?, ?, ?)).

Before learning, a Q-table was assumed in which the values (Q-values) of all belief-action pairs were zero. The model therefore started with no control knowledge and action selection was entirely random. The model was then trained through simulated experience until performance plateaued. The model explored the action space using an ϵ-greedy exploration.

Q-values of the encountered belief-action pairs were adjusted according to the reward feedback. The idea is that, Q-values are learned (or estimated) by simulated experience of the interaction tasks. The true Q-values are estimated by the sampled points encountered during the simulations. The optimal policy acquired through this training was then used to generate the predictions described below (last 1000 trials of the simulation).

While ?'s (?) used Q-learning, any reinforcement learning algorithm that converges on the optimal policy is sufficient to derive the rational adaptation (?, ?). The Q-learning process is not a theoretical commitment. Its purpose is merely to find the optimal control policy. It is not to model the process of learning and is therefore used to achieve methodological optimality and determine the computationally rational strategy (?, ?). Alternative learning algorithms include QMDP (?, ?) or DQN (?, ?).

0.4.4 A typical decision making task

Participants in ?'s (?) study were asked to take on the role of a credit card fraud analyst at a bank. The task was to decide whether a given transaction should be blocked (prevented from being authorized) or allowed. As shown in each panel of Figure 7, nine information sources were laid out in a 3 × 3 grid. An operation panel was presented on the right side of the interface, including Block/Allow decision buttons and a feedback window. The nine cues were selected as relevant to the detection of credit card fraud based on the literature and discussions with domain experts. For example, one cue was called "Transaction Amount". For a particular transaction, this cue signalled either "Fraud" or "Not-fraud" with a *validity* of 0.60. The validity was the probability that the cue indicated fraud given that the ground truth of the transaction is fraudulent. Validities were arbitrarily assigned to the nine cues and reflected the observation that high quality cues are relatively rare in many tasks. The cues had validities [0.85, 0.70, 0.65, 0.60, 0.60, 0.60, 0.55, 0.55 and 0.55], The location of each cue on the interface was assigned randomly for each participant and stayed constant across all trials. Participants were asked to complete 100 correct trials as quickly as possible. As trials in which an error was made (e.g. blocking a non-fraudulent transaction) did not reduce the total number of correct trials required, errors resulted in time costs.

The experiment manipulated three independent factors: *validity*, *format* and *availability*. *Validity* had 9 levels (grouped into high, medium, and low levels of validity). *Format* had two levels: text vs. color. *Availability* had two levels: visible vs. covered. Format and availability give four user interfaces 7.

- Covered-Text (CT) condition (Figure 7a): The cue information was presented in covered text. In order to check each cue, the participants had to click on the associated button on each cue and wait for 1.5 seconds while a blank screen was shown.

- Covered-Color (CC) condition (Figure 7b): The cue information was presented by color (green for possibly normal, red for possibly fraudulent). As with CT, the information was covered until clicked.
- Visible-Text (VT) condition (Figure 7c): The cue information was presented in text. The information was visible immediately (no mouse-click was required).
- Visible-Color (VC) condition (Figure 7d): The cue information was presented in color and no mouse-click was required to reveal it.

0.4.5 Behaviour of the model

?'s (?) were interested in explaining each individuals' interactive decision making behaviour as an approximate solution to a POMDP. The first step in the analysis therefore involved calibrating action dwell times to the empirical data and on average the calibrated dwell time was 0.66 ± 0.10 seconds across the cues.

Having calibrated the model, ?'s (?) then used Q-learning to find a predicted (approximately) optimal strategy for each participant. This strategy was used to predict how many cues should (ideally) be fixated in each condition. This prediction and the human data are reproduced in Figure 8. The model correctly predicts that participants should fixate on more cues in the Visible-Text condition (VT: 6.21 ± 1.32) than in the other three conditions. It also correctly predicts that participants should fixate fewer cues in the Visible-Color condition (Visible-Color: 3.08 ± 1.32) and it learns to be relatively economical in the 'covered' conditions (Covered-Text: 3.95 ± 1.97; Covered-Color: 3.84 ± 1.21).

Intuitively, these findings can be explained in terms of adaptation to information cost and limits of peripheral vision. In the Visible-Text condition, information is cheap and foveated vision is required to read text, therefore more cues are used. In contrast, in the covered conditions (Covered-Text and Covered-Colour) information access is expensive reducing the number of cues used. Lastly, in the Visible-Colour condition, peripheral vision, rather than only foveated vision, can be used to access information and it appears that as a consequence fewer cues are used, at least by being directly fixated.

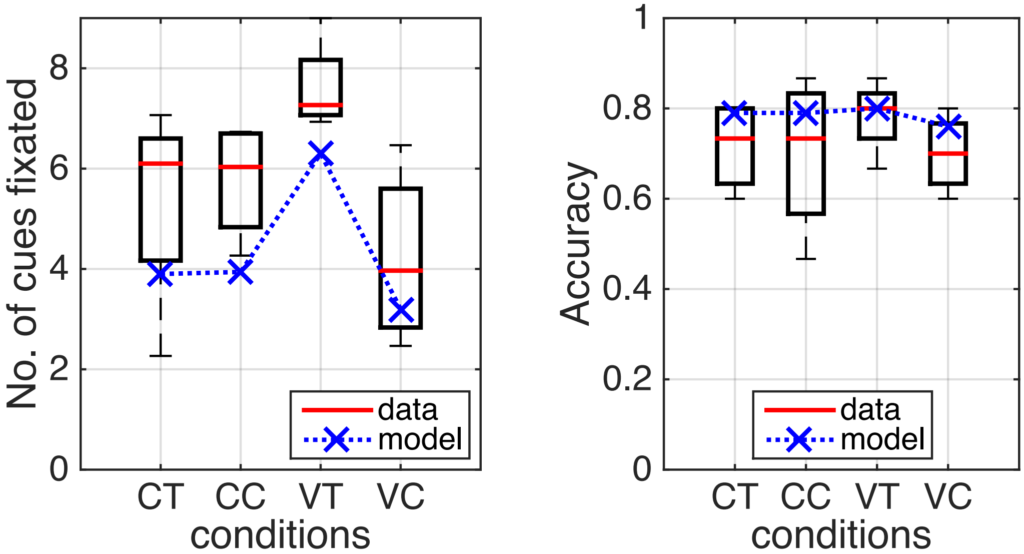


Figure 8: The number of cues (left) and accuracy (right) predicted by the model across 4 experimental conditions (x-axis). Model predictions (blue crosses) are plotted with the participant's data (boxplots). The figure shows that the model predicts that an elevated number of cues will be fixated by participants in the Visible-Text condition and a reduced number in the Visible-Colour.

0.4.6 Implications for decision strategies

? (?) also provides evidence about the use of decision heuristics such as Take-the Best (TTB) and Weighted Additive (WADD). TTB would be indicated by the participants selecting just the very best cue (which in our interface always discriminates) and then making a Block/Allow decision. However, participants did not only use the highest validity cue. Further, WADD would be indicated by the participants using all of the available cues. However, it is clear that this is not happening, with participants preferring to use only a subset of the best cues. While there is plenty of evidence in other tasks that people use TTB, they do not seem to do so in ?'s (?) task, and may not do so exclusively.

TTB and WADD have been extensively studied in the human decision making literature, but more recent work has suggested that people exhibit a more flexible range of strategies. Instead of assuming TTB or WADD, ?'s (?) model derived the optimal strategy given a POMDP problem formulation; this optimal strategy involved using a weighted integration of the best cues.

These cues provide information that optimizes the trade-off between time and accuracy imposed in the experiment. This result is consistent with work that emphasises the adaptation of strategies to a cognitive architecture in interaction with a local task (?, ?, ?, ?). Further work is required to determine whether TTB emerges as a consequence of different task scenarios.

0.5 Discussion

In this chapter we have reviewed two models, previously presented by ? (?, ?), and shown that POMDPs permit a rigorous definition of interaction as an emergent consequence of constrained sequential stochastic decision processes. Theories of human perception and action were used to guide the construction of partial observation functions and transition functions for the POMDP. A reinforcement learning algorithm was then used to find approximately optimal strategies. The emergent interactive behaviours were compared to human data and the models were shown to offer a computational explanation of interaction. In the following paragraphs we summarize the ways in which embodiment, ecology and adaptation constrain the emergence of interaction.

Embodied interaction. In the models interaction was both made possible by and constrained by the way in which cognition is embodied. A key element of embodiment concerns the constraints imposed by the biological mechanisms for encoding information from the environment. In both the menu and the decision model, the limitations of the observation function were key factors that shaped the cognitive strategies determining interaction. The observation function modeled human foveated vision. In menu search foveated vision was the primary determinant of beliefs about the semantic relevance of items and in the decision task it was a primary determinant of beliefs about whether or not a transaction was fraudulent. However, peripheral vision also played a role. In the menu search model, peripheral vision provided an extra source of eye-movement guidance through the detection of items with similar or dissimilar shapes to the target. In the decision model, peripheral vision encoded cue validities without direct fixation, but only when cues were displayed with colour, and thereby played a substantive role in distinguishing the properties of the different visualisations.

Ecological interaction. The ecological nature of interaction was most evident in the menu search model, where menus were sampled from distributions that determined the proportions of high, medium, and low relevance distractor items. These ecological distributions were critical to the structure of the emergent cognitive strategies. While we did not report the analysis, it is obvious that the more discriminable these distributions (the greater the d' in signal detection terms) then the more it should be possible for the agent to adopt strategies that localise fixation to areas where the target is more likely to be found. Ecological constraints on the cognitive strategies were also evident in the decision model where the adaptive strategies were dependent on the distribution of cue validities. Here, high validity cues were rare and low validity cues relatively common resulting in behaviours that emphasised fixating high validitiy cues.

Adaptive interaction. In both the menu model and the decision model, the strategies for information gathering and choice, and consequentially the behaviour, were an adaptive consequence of embodied interaction with an ecologically determined task environment. The strategies emerge from the constraints imposed by ecology and embodiment through experience. They are adaptive to the extent that they approximate the optimal strategies for the user; that is those strategies that maximize utility.

? (?) report that the key property of the model, that it predicts interactive strategies given constraints, is evident in the fact that it generates a broad range of strategies that are also exhibited by humans. For example, in addition to those described above, it also predicts a well-known strategy called *center-of-gravity* (also called *averaging saccades* or *the global effect*) (?, ?, ?, ?, ?), which refers to the fact that people frequently land saccades on a region of low-interest that is surrounded by multiple regions of high-interest. Figure 9 shows that this 'center-of-gravity' effect is an emergent effect of our model. The model also predicts inhibition-of-return.

The fact that the model is able to predict the *strategies* that people use is a departure from models that are programmed with strategies so as to fit performance time. This is important because it suggests that the theory might be easily developed in the future so as to rapidly evaluate the usability of a broader range of interactions. For example, in the near future it should be possible to consider multidimensional visualizations that not only make use of color, but also size, shape, grouping etc. It should be possible to increment the observation functions, for example with a shape detection capacity, and then use the learning algorithm to find new strategies for the new visualizations.

Before concluding, we briefly summarise the advantages and disadvantages of the POMDP approach to explaining interaction that have been dis-

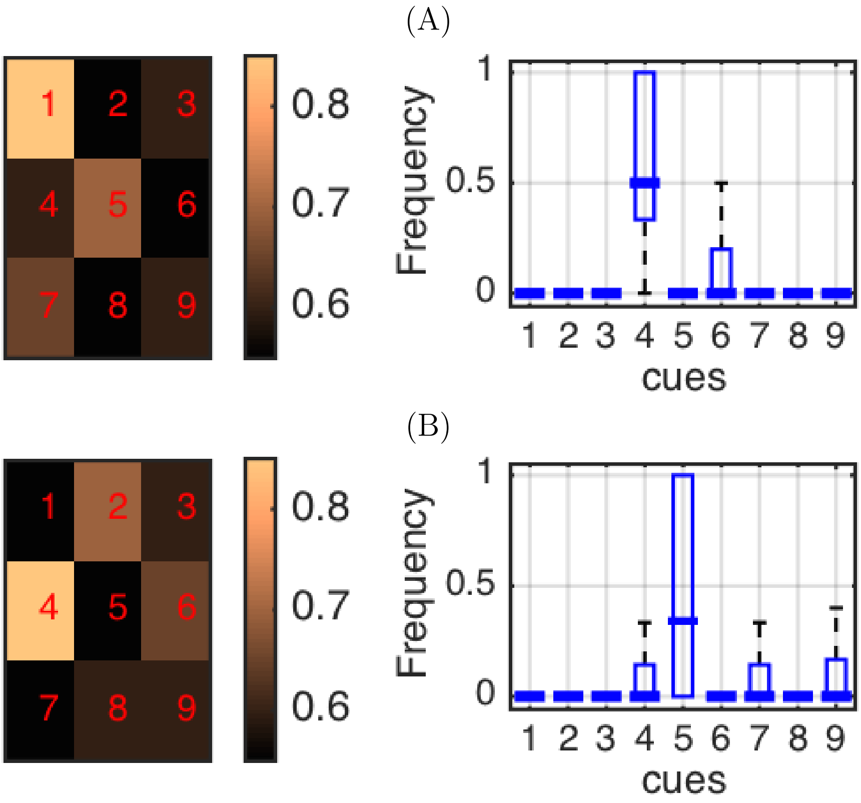


Figure 9: In each row of the figure, the frequency of the model's cue fixations (right panel) is shown for a different spatial arrangement of cue validities (left panel). The validity of ROIs in the left panels is represented as a heat map (high validity is a lighter color). The frequency of model fixations is represented as a box plot. The column numbers in the box plot correspond to the numbers of the ROIs (1..9). In the top row, ROI number 4 has a low validity but is surrounded by relatively high validity ROIs (1,5 and 7). In contrast, in the bottom row, ROI number 5 has a low validity and surrounding ROIs 2, 4 and 6 have high validity. In both rows, the model fixates frequently on the ROI that is surrounded by high validity ROIs. This is known as a centre-of-gravity effect.

cussed in this Chapter. The advantages include:

- The POMDP framing is a well-known and rigorous approach to defining stochastic sequential decision processes and there is a growing range of machine learning algorithms dedicated to solving them.

- POMDPs provide a means of defining and integrating theoretical concepts in HCI concerning embodiment, ecology and adaptation.

- POMDPs provide a means to make inferences about the consequences of theoretical assumptions for behaviour.

The disadvantages include:

- POMDPs are often computationally intractable and careful design is required to define tractable but useful problems.
- Modern machine learning algorithms do not yet provide a good model of the human learning process. Typically, it requires people many fewer trials to acquire rational strategies than are required to solve a POMDP for the same problem.
- Despite the fact that POMDPs are a universal formalism for representing sequential decision processes, the scope of which human behaviours have been modelled, to date, is quite limited. Mostly, the existing models are variants of visual information gathering tasks. The continued expansion of the scope of POMDP models of humans rests on the ability of researchers to find psychologically valid definitions of $< \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, \mathcal{Z}, \mathcal{R}, \gamma >$.

In conclusion, evidence supports the claim that POMDPs provide a rigorous framework for defining interaction as constrained sequential stochastic decision processes. POMDPs can offer a computational explanation of interactive behaviour as a consequence of embodiment, ecology and adaptation.