Draft of June 25, 2009. Comments welcome.
To appear in *Computational Models of Classical Conditioning*, N. Schmajuk (Ed.), Cambridge University Press

Evolution of Attention in Learning

John K. Kruschke and Richard A. Hullinger
Indiana University, Bloomington

A variety of phenomena in associative learning suggest that people and some animals are able to learn how to allocate attention across cues. Models of attentional learning are motivated by the need to account for these phenomena. We start with a different, more general motivation for learners, namely, the need to learn quickly. Using simulated evolution, with adaptive fitness measured as overall accuracy during a lifetime of learning, we show that evolution converges to architectures that incorporate attentional learning. We describe the specific training environments that encourage this evolutionary trajectory, and we describe how we assess attentional learning in the evolved learners.

Birds do it, bees do it, maybe ordinary fleas do it. They all learn from experience. But why is learning so ubiquitous? Why not just be born already knowing how to behave? That would save a lot of time and a lot of error. Presumably, we are born ignorant either because evolution is unfinished or because what we need to know is too complex to be fully coded in the genome. Either way, it seems that evolution has cleverly found a mechanism for dealing with the birth of ignorance, a mechanism that we call learning.

Of course, it may be that learning is merely something that organisms do for fun in their spare time. Perhaps there is not much adaptive value in learning, and little cost, and therefore no selective pressure on the mechanisms of learning. To the contrary, there is good evidence that learning is metabolically costly (Mery & Kawecki, 2003), and therefore it is probably achieving something of reproductive value (Johnston, 1982).

Importantly, what matters is not merely the ability to learn, slowly and eventually. What matters is learning fast. As just one recent example of this fact, Raine and Chittka (2008) showed that different hives of honeybees learned about sources of food at different rates, and those hives that learned faster got significantly more food.

Fast learning favors selective attention

Given that faster learning is better learning, how should learning be speeded up? What sorts of learning mechanisms may have evolved that make learning faster? In this chapter we argue that *selective attention in learning* is a natural consequence of evolutionary pressure to learn quickly in certain environments. We show through simulations that merely by giving a reproductive advantage to organisms that learn faster, an attentional mechanism evolves. Attentional processes yield faster learning in particular environments, and much of our chapter is devoted to describing a range of environments that encourage the evolution of attention in learning.

This perspective on attention in learning, i.e., that selective attention is adaptive and beneficial for learning, contrasts with the intuitive view that selective attention is merely an unfortunate side effect of limited-capacity processing. If attention were merely the consequence of capacity limitations, then selective attention should go away when capacity increases. We show the opposite: Even when there is no metabolic penalty for high learning rates, speed of learning favors selective attention.

The second main purpose of the chapter is to remind readers that some apparent infelicities in learning are, in fact, a natural consequence of having evolved to learn fast. In particular, the highlighting effect, which will be described in detail later in the chapter, seems irrational from a normative statistical perspective, but is a natural consequence of a mechanism for learning quickly, namely attention shifting and learning (for a review see Kruschke, 2010). The highlighting effect should not be construed as an error in an otherwise rational learner. Instead, highlighting should be understood as a signature of a learner who is well adapted to learning fast in particular environments. The benefits of fast learning outweigh the costs of “irrational” generalization which might never actually be tested in the real world.

The highlighting effect has been explained by a theory of attentional shifting and learning (Kruschke, 1996a, 2001, 2003, 2010). The idea is that when a cue-outcome event occurs that contradicts previously learned expectations, attention shifts away from cues that cause error, toward other cues. The re-allocation of attention becomes a learned response to those cues. Various data converge on that explanation, including eye tracking (Kruschke, Kappenman, & Hetrick, 2005). No other theory has yet been able to account for the highlighting effect in as much detail. Therefore, we use the highlighting effect as a strong signature of attentional shifting during learning. One of the main findings of this chapter

For helpful discussion and comments, we thank Jose Burgos, Nestor Schmajuk, and Peter M. Todd. Correspondence can be addressed to John K. Kruschke, Department of Psychological and Brain Sciences, Indiana University, 1101 E. 10th St., Bloomington IN 47405-7007, or via electronic mail to kruschke@indiana.edu. Supplementary information can be found at http://www.indiana.edu/~kruschke/

is that learners who have evolved to learn fast (in certain environments) also exhibit highlighting as a side effect.

The chapter is organized as follows. First, we describe the particular type of training environment in which the simulated learners will evolve. Essentially, the environment implements context-dependent cue relevances, with contexts changing through time. Then we describe a class of learning agents that will be explored. We use variants of backpropagation networks (Rumelhart, Hinton, & Williams, 1986) as a representative class of associative learning models. We then show results from "intelligent design", by which we humorously refer to the process whereby we establish intuitively reasonable architectures (instead of randomly searching for architectures) and use hill-climbing optimization to find optimal learning rates. In all cases, the learning rates that learn the training environments fastest also show robust highlighting. Next, we report results from genetic algorithms that searched the space of architectures and learning rates simultaneously. Again, the best learners show highlighting. We conclude with a discussion of other training environments conducive to learned attention.

Fast learning *of what* favors selective attention *to what*?

We have made the skeletal claim in the introduction that fast learning favors selective attention. To flesh out the claim, we need to define what environmental situations are being learned and what aspects are attended to.

Attention to what? The representation

If the need for speed is paramount in learning, then why not just evolve a high-capacity memorizer? This would be analogous to a high-speed, high-resolution video camera that has yottabytes[1] of memory, recording every moment instantly. It seems that this might be the optimal learner, subject only to costs of hardware. To the contrary, such a system is far from optimal, even if the hardware is free. The problem comes in using the stored information. To use the memory for anticipating outcomes in new situations, either the new situation must retrieve an *exact match* in the vast memory to determine the exact outcome that occurred before, or the new situation must retrieve many *similar* memories and the system must somehow integrate across those memories to anticipate a likely outcome. In the real world there is never an *exact* repetition of a situation. For example, recognizing a person from day to day demands imperfect matching to memory of the person from previous days, because the person's appearance and behavior are never exactly the same from day to day. Therefore, memory retrieval that is based on exact matching would be useless in practice, even if the hardware were available in principle.

Instead of exact matches to memory, retrieval must be based on some form of similarity between stimulus and memories. Similarity can be defined many different ways, and only in the context of specific representational formats. In any case, the point is that the mind has some representational format that is not a mere copy of the sensory surface. The representation is a transformation of the sensory information into a format that has various useful components.

It is the components of the representation that can be selectively attended. By this we mean that the representational components can be selectively enhanced or suppressed. We will not be modeling the entire process of transforming a sensory surface into internal representations of perception, cognition, and action. Instead, our model starts with an input representation that already assumes considerable transformation from sensory input to percept. Specifically, we will assume that the learner's world consists of the presence or absence of various features, such as tones, lights, colors, etc. This sort of input representation is assumed by many venerable models of associative learning.

What makes this sort of feature-based representation so intuitive and effortless is that the features can be easily selectively attended by us. For example, we can talk about the presence/absence of a tone, or the presence/absence of a light, because they can be selectively attended. Aspects of the world that are difficult to selectively attend, such as brightness versus saturation of colors, are used less often in associative learning experiments. The selectively attendable features need not be conceptually simplistic, like pure tones or lights. Instead, the features could be complex entities, such as the presence/absence of the word "radio", or the presence/absence of a picture of a fish. We assume that the learner has already acquired some internal representation of certain features, however simple or complex. Our models, to be described below, allow forms of selective attention to those features.

Learning of what? The environment

What is it that must be learned, that we claim can be speeded by selective attention? We believe that a fundamental challenge faced by an organism is *context-dependent relevances of cues*. The challenges posed by contextual dependencies have been recognized by machine learning researchers for decades (for a review see, e.g., Edmonds & Norling, 2007). Individual learners, such as humans, may be born into environments with cue relevances that change depending on the context. The context-specific relevances must be learned, and learned quickly for reproductive advantage.

Definition of context. The term "context" is used by different authors in different ways. In general, contextual cues may differ from non-contextual cues in their spatial or temporal arrangement, or in their contingent relationship with the outcomes. Contextual cues are sometimes thought of as spatially ambient rather than focal. For example, context may be the color of the background of a visual display (e.g., Dibbets, Maes, Boermans, & Vossen, 2001), or context may be the spatial constellation of items in an array (e.g., Chun, 2000). Contextual cues are sometimes supposed to be relatively static through time compared to focal cues. For example, the context may be the restaurant in which a sequence of different foods (the focal cues) are observed (e.g., Rosas

[1] A yottabyte is 10^{24} bytes, i.e., one trillion terabytes.

Table 1
A training environment that has context-dependent relevancies.

Context		Focal Cues				Outcomes	
I	J	A	B	C	D	X	Y
1	0	1	0	1	0	1	0
1	0	1	0	0	1	1	0
1	0	0	1	1	0	0	1
1	0	0	1	0	1	0	1
0	1	1	0	1	0	1	0
0	1	1	0	0	1	0	1
0	1	0	1	1	0	1	0
0	1	0	1	0	1	0	1

Note: Presence of a cue or outcome is denoted by a 1, and absence is denoted by a 0. Each row denotes a different training trial.

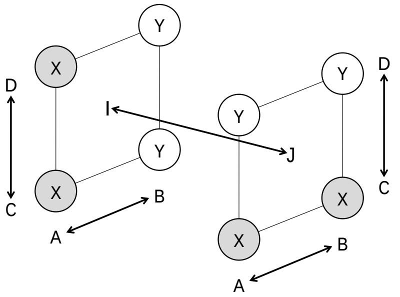


Figure 1. Spatial representation of the training structure in Table 1. Each circle represents a combination of cues A, B, C, D, I, and J. The letter in the circle, either X or Y, represents the correct outcome for that cue combination. The square on the left has a letter I in its center to indicate that cue I is present, while the square on the right has a letter J in its center to indicate that cue J is present. The upper circles have cue D present, while the lower circles have cue C present. The remaining dimension marks whether cue A or cue B is present. Although this diagram represents A–B, C–D, and I–J as if on dimensions, the basic structure does *not* encode or assume any dimensional relationship among the cues.

& Callejas-Aguilera, 2006). Contextual cues are also often intended to be uncorrelated with the outcome, such that contextual information by itself is uninformative regarding what specific outcome to anticipate (e.g., Little & Lewandowsky, 2009; Yang & Lewandowsky, 2003). For our purposes, we define context as a cue that is not correlated with the outcome, and that changes in time less frequently than other cues. In other words, we emphasize the temporal and contingency aspects of context, not its spatial aspect.

An environment with context-dependent relevance. Table 1 shows the standard training environment that we will use to instantiate context-dependent relevance. Cues arbitrarily denoted by labels “I” and ”J” act as context cues. These context cues have zero correlation with the outcomes, but they do indicate which other cues are good predictors of the outcomes. When context cue I is present, focal cues A and B are perfect predictors of the outcomes, but focal cues C and D are uncorrelated with the outcomes. On the other hand, when context cue J is present, the roles of the focal cues are reversed, with cues C and D now being the perfect predictors of the outcomes, and cues A and B being uncorrelated with the outcomes.

During training, we will usually group together several consecutive trials that share the same context cue. Thus, several trials with context cue I will occur, followed by several trials with context cue J, and so forth. In this way, the context cues change less frequently than the other cues. The exact number of trials of in one context or another will be manipulated in different simulations.

Neither the context nor focal cues have any spatial coding in the model. There is no distinction between context and focal cues other than their contingencies with other cues and the outcomes. The columns of Table 1 are labeled separately (as context and focal cues) merely for benefit of the reader; the simulations had no such benefit.

There is redundancy built into the structure of Table 1, with cue B being redundant with cue A, and cue D being redundant with cue C, and outcome Y being redundant with cue X. These redundancies are unnecessary for the basic demonstrations we report below, but the redundancies do provide a symmetry that makes interpretation of the simulations easier.

The structure of Table 1 is also isomorphic to the structure denoted “Type III” in the monograph by Shepard, Hovland, and Jenkins (1961), which reported benchmark results regarding the relative difficulties of six different category structures. Unlike their work, our demonstrations do not assume that cues I–J, A–B, and C–D are alternative values of three distinct dimensions. Despite the fact that the cues in Table 1 are not dimensionalized, it may benefit understanding to display them as if they were, as shown in Figure 1. The items for which context cue I is present are shown on the left side of the figure, and the items for which context cue J is present are shown on the right side of the figure. The correct outcome is denoted by X and Y, along with grey shading for to enhance the visual distinctiveness of outcome X. It can be seen that in context I, cues A and B are relevant to the outcome, but in context J, cues C and D are relevant to the outcome.

Designing fast learners

Our goal is to create fast learners of contextually dependent relevancies. We will use the structure of Table 1 as the test bed. The simulated learners will be trained on several repeated blocks of the structure, and the total accuracy during training will be used as a measure of reproductive fitness. We will explore various model architectures and temporal groupings of context trials. For each design, we will find learning rates that minimize the total error (i.e., maximize the total accuracy) during the lifetime of the learner.

Assessing selective attention: Exhibiting highlighting

Having thereby designed optimal fast learners, we will then assess whether the learner has selective attention. There are many criteria one might establish for declaring that a learner possesses selective attention. The criterion we will use is that the learner exhibits *highlighting*.

In the highlighting procedure (Kruschke, 2010), training begins with the presentation of two cues, denoted I and PE, leading to the outcome E. (Cue I here bears no relation to context I in the other structure; the shared label is accidental coincidence.) We denote such trials as I.PE→E. After this early training, occasional trials introduce a new case: I.PL→L. In later training, those cases predominate, so that the overall number of I.PE→E trials equals the overall number of I.PL→L trials. The outcomes are denoted E and L because they are Early-trained and Late-trained, respectively. The cue PE is so labeled because it is a Perfect predictor of the Early outcome, and the cue PL is so labeled because it is a Perfect predictor of the Late outcome. The cue I is an Imperfect predictor of the two outcomes.

Notice that the two outcomes have symmetric structure. Each outcome has one perfect predictor, and the outcomes share an imperfect predictor. Moreover, there are an equal number of trials of the two cases, overall. If people learn this simple symmetry, then the imperfect predictor should be equally (un-)associated with the two outcomes, and the two perfect predictors should be equally associated with their respective outcomes. This symmetry is easy to assess, as follows. After training, we test people with cue I by itself, asking people to respond with the outcome they think is most likely based on what they have learned. It turns out that people do not give 50/50 responding, but instead clearly prefer the early-learned outcome E (roughly 70/30). This preference is not a mere primacy bias for any ambiguous test, however. When tested with the pair of cues PE.PL, people clearly prefer the later-learned outcome L (roughly 65/35).

The "torsion" in preferences, wherein one ambiguous cue leads to a preference for E but another ambiguous cue leads to a preference for L, is called the highlighting effect. The highlighting effect has been found for many different stimuli, relative frequencies, cover stories, and so on. For a review, with data from a "canonical" experiment that has equal base rates for the various cases, see Kruschke (2010).

The highlighting effect is challenging to explain. Because of the simple symmetry in the structure, many formal models of learning predict symmetric response preferences. The Rescorla-Wagner (1972) model, for example, predicts symmetric associations (with sufficient training).

The most successful account of highlighting so far is an attentional account, suggested informally by Medin and Edelson (1988) and formalized by Kruschke (1996a, 2001). When people are learning the early cases I.PE→E, attention is allocated to both cues, because there is no reason not to. Consequently, moderate strength associations are learned from both cues to outcome E. The associative strengths are only moderate because the two cues mutually support each other in generating the anticipation of the outcome. When subsequently learning cases of I.PL→L, however, attention rapidly shifts away from cue I, because it has already been learned to indicate something other than the correct outcome L. Attention therefore falls on the distinctive cue PL, and a strong association is learned from PL to outcome L. Thus, in learning I.PL→L, people have learned two things: First, they have learned to re-allocate attention away from I to PL. Second, they have learned to associate PL with L.

Because the attentional account of highlighting has been rather successful in quantitatively accounting for many variations of the highlighting design and other cue-outcome mappings (again, for a review, see Kruschke, 2010), we will treat the highlighting effect as a behavioral signature of attentional learning.

Other signatures of attentional learning?. There are other learning phenomena that have been explained in terms of attention, but we do not explore them in the present chapter for two different reasons. First, some of these other phenomena can be explained without appeal to attentional mechanisms. Second, some of these other phenomena require the ability to learn complex, non-linear relationships between cues and outcomes, that the simple models we explore in this chapter cannot learn.

Consider the phenomenon known as *blocking*, wherein an initial training stage involves trials of A→X, and a subsequent training stage involves trials with a redundant relevant cue, A.B→X (Kamin, 1969). In subsequent tests, the association from B to X appears to be weaker than it would have been if the initial phase with A alone had not been experienced. This relative weakness of B has been explained in attentional terms, such that there has been learned suppression of cue B (e.g., Kruschke, 2001; Kruschke & Blair, 2000; Kruschke et al., 2005; Mackintosh, 1975). But the basic blocking effect can also be explained without appeal to attentional learning (e.g., Rescorla & Wagner, 1972; R. R. Miller & Matzel, 1988). Therefore we have chosen not to use blocking, per se, as a signature of attentional learning.

Another phenomenon that has been explained in terms of attention is *latent inhibition* (Lubow, 1989; Schmajuk, 2002). In the basic procedure for latent inhibition, a cue is first presented with no notable outcome, in a set of trials called the pre-exposure phase. Subsequently, the cue is paired with a novel outcome. Latent inhibition occurs when learning of the novel cue-outcome association is retarded because of the pre-exposure phase. One explanation is that the pre-exposure phase produced learned attentional suppression of the cue, which lingered into the subsequent phase in which the cue was paired with an outcome (e.g., Kruschke, 2001; Schmajuk, Lam, & Gray, 1996). The phenomenon can be difficult to obtain in humans, however (but see Nelson & Sanjuan, 2006, for a recent example), and there are a variety of findings suggestive of different underlying mechanisms in latent inhibition. Therefore we have chosen not to use latent inhibition as signature of attentional learning.

Another classic phenomenon that has been attributed to attentional learning is the advantage of *intradimensional* *shifts* relative to *extradimensional shifts* (e.g., Hall & Channell, 1985; Kruschke, 1996b; Slamecka, 1968). In these relevance-shift procedures, a learner is first trained on two-dimensional stimuli for which one dimension is perfectly predictive of the outcome and the other dimension is irrelevant to the outcome. For example, it could be that red circles or squares are mapped to outcome X, while green circles or squares are mapped to outcome Y. In this case, color is relevant while shape is irrelevant. In the shift phase, novel values of the dimensions are used; e.g., blue or yellow stars or triangles. When the same dimension is relevant in the shift phase, the shift is called an intradimensional shift. When the other dimension is relevant in the shift phase, the shift is called an extradimensional shift. Many experiments have demonstrated that for adult humans, intradimensional shift is easier to learn than extradimensional shift. This advantage for intradimensional shifts is naturally explained by attentional learning: In the initial phase, people have learned to attend to the relevant dimension and to ignore the irrelevant dimension. This attentional allocation persists into the shift phase. Whereas this is a strong indicator of attentional learning, a model of it requires representation of dimensions and values within dimensions; e.g., representation of red and green along with dimension of color. In our modeling efforts, we opted to use a simpler representation that avoided assumptions about dimensions, and therefore this phenomenon of intradimensional shift advantage is beyond the scope of our present explorations (but see Kruschke, 1996b, for a related model).

Finally, in theories of category learning, attentional learning is used to explain the differential difficulties of various category structures. In particular, the relative ease of two structures introduced by Shepard et al. (1961) is naturally interpreted in terms of attentional learning. These structures involve three binary-valued dimensions, with the resulting eight instances mapped into two categories. One structure involves a non-linearly separable exclusive-OR on two dimensions, with the third dimension being irrelevant (called Type II by Shepard et al., 1961). The other structure is linearly separable, defined by two diametrically opposed prototypes whereby all three dimensions are relevant to distinguish the categories (called Type IV by Shepard et al., 1961). Despite the fact that the latter structure is linearly separable and the former structure is not, the latter category is harder to learn. Some theories assert that the latter structure is harder because it demands attention to all three dimensions, whereas the former category only demands attention to two dimensions (e.g., Kruschke, 1992; Nosofsky, Gluck, Palmeri, McKinley, & Glauthier, 1994; Shepard et al., 1961). Other theories use more rule-like representations to account for the relative difficulties (e.g., Goodman, Tenenbaum, Feldman, & Griffiths, 2008; Nosofsky, Palmeri, & McKinley, 1994), which might be re-construed in attention-like terms. Even among explicitly attentional approaches, modeling these structures appropriately requires representation of dimensions, and, as mentioned above, we have opted to use simpler representations in the current explorations.

Thus, of the many phenomena that may be considered as indicators of attentional learning, it is the highlighting phenomenon that is both structurally simple and uniquely explained (so far) by attentional learning. Therefore we use highlighting as the behavioral signature of attentional learning.

Design space and functional desiderata

Given a design space consisting of backpropagation networks, we want to explore variations that may implement functional desiderata. One desideratum is that previous learning should be protected, as appropriate, when learning new associations. For example, there should not be catastrophic forgetting of the fact that $2 \times 2 \rightarrow 4$ when subsequently learning the fact that $3 \times 3 \rightarrow 9$ (McCloskey & Cohen, 1989). One way to help protect previous learning, when a new combination of cues is encountered, is by shifting the internal representation of the cues away from the conflicting, previously-associated cues. In other words, if previous learning has associated a particular cue with a particular outcome, and new outcomes also include that previous cue among the presented cues, then the previous association from that cue can be protected by shifting attention away from it when learning the new outcome. Such a shift in internal representation can have an undesirable side effect, however, because the shift might generate arbitrary patterns of activation that correspond to nothing present in the cues. Loosely speaking, if you close your eyes to deflect your attention away from a previously learned cue, then you might imagine anything; an unconstrained shift of representation might cause "hallucinations". Therefore, a second functional desideratum is for the shift of representation to be constrained by the actually present cues.

These functional desiderata can be implemented many ways. We considered the following possibilities. One way to keep the hidden-layer representation faithful to the actually presented cues is to establish hidden nodes that have fixed 1-to-1 connections from corresponding input cues. These 1-to-1 connections cause the corresponding hidden node activations to start the training as approximate copies of the input cue activations. This initial state can be eventually overruled by learned connections from other input cues, but at least there is an initial bias toward faithfulness to present cues. As second way to keep the the hidden layer from hallucinating is to allow learning only for hidden nodes for which the corresponding input cue is activated. This method can be easily implemented by multiplying the hidden-node activation by the corresponding cue-node activation. The multiplicative product is large only if both the cue-node activation and the hidden-node activation are large.

The second desideratum, i.e., protection of previous learning by a shift of hidden representation, can be achieved in different ways. One way is to do gradient descent on error with respect to the hidden weights first, before changing the output weights. In this way, the previously learned output weights are protected, if possible. After the hidden weights are shifted, then input is re-propagated to the hidden nodes and the output weights are learned. A second way to im-

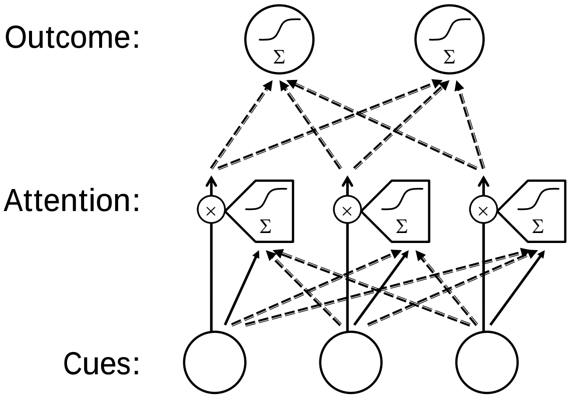


Figure 2. An architecture for exploring learning rates that minimize error.

plement a shift is to have two sets of weights: One set is the regular type, the other set is "first and fast": first-updating but with fast decay to zero before the next trial begins (a related scheme for fast-decaying weights was proposed by Hinton & Plaut, 1987). Again, the first-fast weights protect the output weights from catastrophic forgetting, but in this case the slow hidden weights do not need to change radically to implement the protection.

Figure 2 illustrates all these design possibilities in a single network architecture. Not all of the options need to be implemented simultaneously. The diagram indicates that the hidden nodes and outcome nodes are standard backpropagation nodes that first sum their weighted inputs and then squash the sum with a sigmoid function. Formally, denote the activation of the i^{th} input node as a_i^{in}, the activation of the j^{th} output node as a_j^{out}, and the weight connecting node i to node j by w_{ji}. Then the sigmoidal activation function is given by

$$a_j^{\text{out}} = 1 \bigg/ \left[1 + \exp\left(- \left[\sum_i w_{ji} a_i^{\text{in}} - \theta_j \right] \right) \right] \qquad (1)$$

where θ_j is the *threshold* of the j^{th} node. A graph of the sigmoidal output, as a function of the summed inputs, is a tipped "S" shape as shown schematically inside the nodes of Figure 2. The sigmoid activation asymptotes at 1.0 as the summed input exceeds the threshold by a large positive amount, and the sigmoid activation asymptotes at 0.0 as the summed input is far below the threshold. When the summed input equals the threshold, then the sigmoid activation is 0.5.

The dashed arrows in Figure 2 indicate learnable weights, all of which are initialized at zero. The solid arrows impinging upon the hidden nodes are fixed, non-learnable 1-to-1 connections that implement the idea that each hidden node starts as an approximate copy of the corresponding input cue. For purposes of demonstration in the simulations, the 1-to-1 connection weights were arbitrarily fixed at 10.0, with thresholds in the sigmoid function also set at 10.0. Consequently, in the naive network with all zero weights except the 1-to-1 connections, when an input cue is active, the corresponding hidden node has activation of 0.5, and when the input cue is not active, the corresponding hidden node has activation of nearly zero.

Figure 2 also shows a solid arrow from the input cues passing *beside* the hidden linear-sigmoid node, via a circle marked with a multiplication sign. These arrows indicate optional multiplication of the hidden node activation by the corresponding input cue activation.

Finally, the dashed arrows in Figure 2, which indicate learnable connections, each represent two different learnable connections. Both connections learn via the standard backpropagation algorithm, but they can have different learning rates. Crucially, one connection is the traditional "slow" learner, whereas the other connection is a "first-fast" learner. The first-fast connection adjusts its weights before the slow connection, i.e., it learns *first*, and then activation is repropagated and error is recomputed before the slow connection is adjusted. Moreover, the first-fast weight decays to zero before the next trial starts, i.e., it is *fast* decaying, whereas the slow weight does not decay.

In summary, there are four learning rates in the architecture of Figure 2: The hidden, a.k.a. attention, nodes have incoming weights that have a slow learning rate and a first-fast learning rate. The outcome nodes also have slow and first-fast learning rates, that can be different from the attention-node learning rates. If any of the learning rates is zero, it is tantamount to that sort of learning being unavailable to the network. There is also an optional multiplicative "gating" of the input activation by the corresponding hidden activation.

Results: Optimal learners exhibit highlighting

For each architectural option, we used hill-climbing optimization to discover the learning rates that minimized the total error during training on the context-dependent-relevance structure in Table 1. It might seem that higher learning rates would always produce faster learning and smaller total error, but this is not the case. The reason is that learning rates that are too high cause the weights to overshoot the best values, thereby producing larger error on subsequent training trials. Thus, even though we allow the learning rates to be arbitrarily large as needed, the best learning rates turn out to be moderate in magnitude.

The main question is whether a network that has optimal learning rates also embodies selective attention. Selective attention is assayed behaviorally by exhibition of highlighting. For each architectural option, we found the optimal learning rates, then trained a naive network on the highlighting structure and tested whether the network exhibited highlighting.

In detail, the simulations proceeded as follows. In training the context-dependent-relevance structure, blocks of four consecutive trials used the same contextual cue (I or J). At the beginning of each block of four trials, there was a 50/50 chance of being trained in context I or context J. Each simulated network was trained on 20 random blocks, constituting 80 trials. For each simulated network, the total error across training was recorded. 50 different random training sequences were averaged to compute the error for a given learning rate. Formally, denote the correct, "teacher", value at outcome node k on trial t in sequence s as T_{stk}, with $T_{stk} = 1$ if k is the correct outcome for the present cues, and $T_{stk} = 0$ otherwise. Then the overall error was measured as

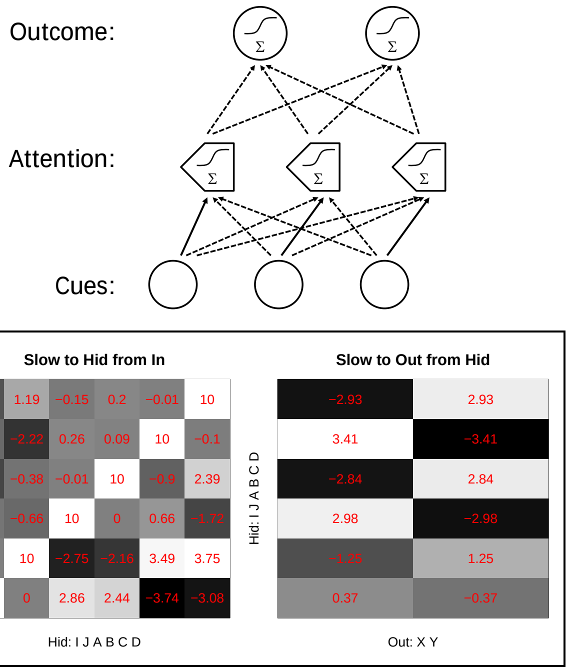
$$\text{RMSD} = \left[\frac{1}{50 \times 80 \times 2} \sum_{\text{seq } s}^{50} \sum_{\text{trial } t}^{80} \sum_{\text{out } k}^{2} \left(T_{stk} - a_{stk}^{\text{out}} \right)^2 \right]^{1/2} \qquad (2)$$

where the summations are over sequences, trials, and outcome nodes, respectively, and where a_{stk}^{out} is computed by the sigmoid activation function in Equation 1. The overall error in Equation 2 is also called the root-mean-squared deviation (RMSD) between the taught and generated values. As a reference for the magnitude of the RMSD, consider its value if the network learned nothing, so that the network's outcome activations were always exactly 0.5 (which is what the sigmoid activation function generates when all the weights are zero). In this case, because T is always zero or one, $T_{stk} - a_{stk}^{\text{out}}$ is always ±0.5. Hence the RMSD is 0.5 when there is no learning at all. The RMSD gets smaller than 0.5 when there is successful learning.

A hill-climbing optimization routine was used to find learning rates that produced the smallest possible error. The optimizer started with reasonable learning rates specified by the programmer, and then incremented or decremented the various learning rates until adjustments no longer yielded any significant reduction in RMSD. (The arbitrary starting point for the learning rates was manually set at several different values for different runs, to gain confidence that a global minimum was achieved.) The hill-climbing optimizer found learning rates that minimized the RMSD.

Having converged to the optimal learning rates, the network was then reset to all-zero weights and tested on the highlighting structure. It began with 8 trials of I.PE→E, then a random mix of 12 trials of I.PE→E with 4 trials of I.PL→L, followed by a random mix of 8 trials of I.PE→E with 24 trials of I.PL→L. Notice that there were an equal number of I.PE→E and I.PL→L trials overall. After the training, the network was tested with I.PE, I.PL, I alone, and PE.PL.

Different architectural options were used, with optimal learning rates determined for each. Figures 3 and 4 show the results when only slow weight learning was permitted, with no multiplicative gating. In other words, the first-fast learning rates were fixed at zero, while the slow learning rates on both layers of nodes were allowed to be whatever values minimized the RMSD. Figure 3 shows the learned weights at the end of training, for one representative network. The weights are displayed in matrix format, with the weight values indicated numerically and by the shading in the cells. The left matrix shows the weights to the hidden (attention) nodes from the cue nodes. Notice that the diagonal cells of the left matrix are all 10, reflecting the fact that the 1-to-1 connections are set permanently to 10 in these simulations. Of special interest is the lowest row of this matrix, which represents the weights from context cue I. The connections from context cue I to the hidden nodes corresponding to cues A and B have become *positive*, but the connections from context cue I to the hidden nodes corresponding to cues C and D have become *negative*. The weights from context cue J, in the next row up, show the opposite pattern. These weights suggest that the network has learned to pay attention to A and B when context I is present, but to pay attention to C and D when context J is present.



−1.29	1.19	−0.15	0.2	−0.01	10
−0.66	−2.22	0.26	0.09	10	−0.1
−1.7	−0.38	−0.01	10	−0.9	2.39
−0.26	−0.66	10	0	0.66	−1.72
0	10	−2.75	−2.16	3.49	3.75
10	0	2.86	2.44	−3.74	−3.08

−2.93	2.93
3.41	−3.41
−2.84	2.84
2.98	−2.98
−1.25	1.25
0.37	−0.37

Figure 3. Simulation results when there is only slow weight learning on hidden and outcome layers, with no first-fast learning and no attentional multiplication, as suggested by the network diagram in the upper part of the figure. The network diagram shows only three cues, whereas the simulations involved six. The lower panel shows the weights at the end of training in a typical run on the context-dependent structure (Table 1). In the left matrix, the rows index the input cue, in the order I, J, A, B, C, and D, as indicated along the left edge of the matrix. The columns index the hidden node, in the same order, as indicated at the bottom edge of the matrix. Notice that the weights from input node I (lowest row) are positive (2.86 and 2.44) to hidden nodes A and B, but negative (−3.74 and −3.08) to hidden nodes C and D. The weights from input node J show the opposite pattern. These weights indicate that the network has learned to pay attention to A and B in context I, but to pay attention to C and D in context J. The RMSD across 80 training trials and 50 simulated subjects was 0.250.

Figure 4 show the result of subsequent testing of the network with the highlighting procedure. The same learning rates were used, but starting with a naive network. The figure shows the weights at the end of training on a typical run. It can be seen that the weight from PL to hidden-I is strongly negative, but the weight from PE to hidden-I is fairly positive. In other words, the network has learned to suppress attention to I when PL is present, but to attend to I when PE is present. The learned weights result in a strong highlighting effect: When presented with cue I by itself, the network produces a strong outcome preference for E, but when presented

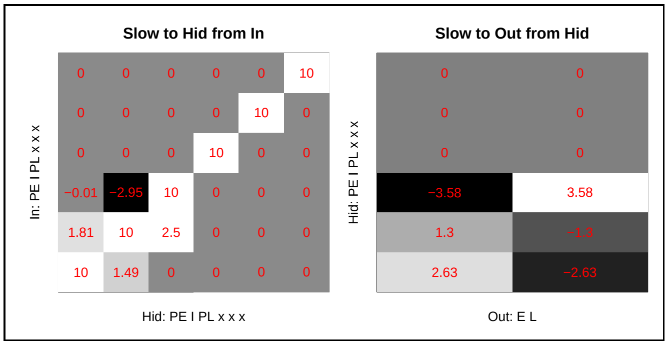


Figure 4. Results from test of highlighting, using the same architecture and parameter values as in Figure 3. These weights result in a strong preference for outcome E when tested with cue I, and a strong preference for outcome L when tested with cues PE.PL. Notice in the left matrix that there is a strong inhibitory weight (−2.95) from cue PL to hidden node I, indicating that the network has learned to suppress cue I when cue PL is present. The right matrix shows that the weights from hidden node I to the outcomes are not symmetric; they excite outcome E (+1.3) but inhibit outcome L (−1.3).

with cues PE.PL, the network produces a clear outcome preference for L. In summary, when the slow-weight learning rates are optimized so that the context-dependent-relevance structure is learned with least error, then the network exhibits robust highlighting.

We also found optimal learning rates when the architecture included first-fast learning on the input-to-hidden connections, and multiplication by the input cues. The pattern of results for the optimal learning rates was the same, but the RMSD decreased to 0.233, and the magnitude of highlighting increased.

These simulations establish examples of what we mean by learned attention: Individual cue activations are amplified or attenuated depending on which other cues are present. The networks have learned to selectively enhance or suppress particular cues, in a context-dependent manner. It is this sort of context-dependent, learned modulation that we are calling “selective attention” when analyzed at the level of hidden network activations. At the behavioral level, attention can only be assayed by overt outcome activation patterns without reference to hidden internal activations. We use the highlighting effect as a behavioral-level signature of selective attention.

In the final discussion we shall return to these explorations in design of network architectures but with other training environments. Before those explorations, we will describe the more thorough search of design space that is possible via genetic algorithms.

Evolution: Genetic algorithms discover fast learners

Human designers cannot manually explore the myriad (indeed infinite) combinations in the design space. It could well be that there are unforeseen combinations of design options that learn even better than those discovered by hill-climbing on learning rates in a pre-set architecture. In this section we report the results of extensive searches of the design space by simulated evolution, i.e., genetic algorithms (e.g., Goldberg, 1989). An advantage of a genetic algorithm (GA) is that it can explore a wide range of architectural combinations and learning rates simultaneously, unlike the hill-climbing searches that were restricted to a particular architecture. To simulate the evolution of attention in learning, we follow the approach presented in G. F. Miller and Todd’s (1990) and Todd and Miller’s (1991) work on evolving networks (agents) that learn. We use a genetic algorithm (GA) to evolve populations of agents in the context-dependent-relevance structure of Table 1, and we look for signs of attention shifting (i.e., highlighting) in the best performing agents.

Overview

Agents. Each agent in the simulation consists of a connection matrix that describes each node in the network, the type of connection between each of the nodes (no connection, fixed connections, slow-learning, or first-fast connection), and the initial strength of each connections. Additionally, each agent contains a structure that specifies the learning rates to be used for backpropagation of error at each layer of the network and other learning-related details such as whether the agent implements multiplication of hidden activations by cue activations.

This genetic structure can be used to specify an infinite space of backpropagation networks with different numbers of input, hidden, and output nodes, different connection architectures, learning rates, and error propagation methods. In our simulations, the “genome” explicitly specifies various weights and learning rates, which might not be very biologically plausible, but nevertheless serves our purpose of thoroughly searching the space of design possibilities. More biologically plausible specifications may be possible, see for example the work of Burgos (2007). In order to keep the evolutionary process tractable and to make comparison with the hill-climbing simulations straightforward, the networks are constrained as follows: Each network has six input nodes (one for each of the binary cue values I, J, A, B, C and D), six hidden nodes, and a single output node. The single output node represents outcome X by an activation of 1, and outcome Y by an activation of 0. Its threshold is fixed at zero, so the outcome node’s baseline activation is 0.5, the neutral value between X and Y. Each input node is constrained to have a fixed 1-to-1 connection weight of 10.0, and the hidden nodes’ thresholds for the sigmoid functions were fixed at 10.0.

Each network is trained in an environment that has a random assignment of abstract cues (i.e., I, J, A, B, C, and D) to physical input nodes. Because no individual agent knows which cues will play which role, the best evolved initial weights should be symmetric across input cues. To simplify the simulations, we enforced this logical symmetry rather than let it noisily evolve. This symmetry is produced

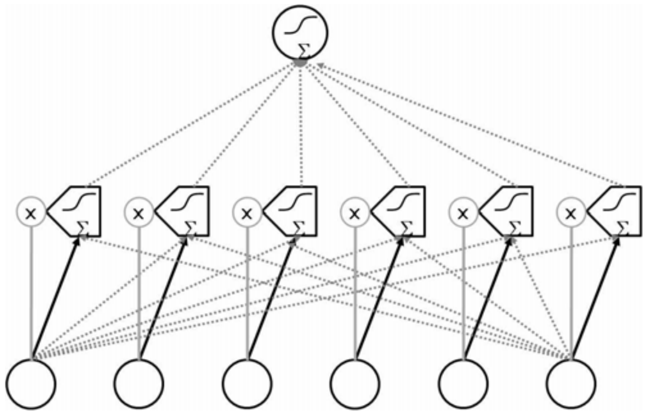


Figure 5. An architecture showing potential connections for an evolved network. The faint dotted lines indicate connections that may be fixed (non-learning), slow-learning or first-fast connections. The faint multiplication nodes indicate that a particular agent may or may not evolve attentional multiplication. Note that for simplicity not all of the possible connections from the input to the hidden layer were included.

in a three step process when birthing a network. First, as noted above, each input node will have a fixed 1-to-1 connection. Second, the connection type and initial weight for the connection between the first input node and its adjacent hidden node is copied identically for each input-to-adjacent-hidden node connection. Third, the connection type and initial weight for the connection between the first hidden node and the output node is replicated to each hidden-to-output connection. Figure 5 shows a simplified diagram of the possible network architectures. Figure 5 is much like Figure 2 except that all six cues are explicitly indicated, and there is only a single outcome node as described above.

Environment and Fitness. The scenario under which our agents are evolving is a very simple context environment. The agents do not have to move, they do not have to actively seek out stimuli, and they do not have to interact with, or face competition from, other agents or any other outside factors. As with Todd and Miller's (1991) simulations, it may be helpful to think of the agents as being born in an aquatic world where they are attached to the sea floor, passively watching potentially edible stimuli float by. Each passing stimulus has a set of distinctive cues and based on those cues the agent must decide if the stimulus is edible or inedible. The agent's fitness is increased when it makes a correct decision, to eat something that is edible or to avoid something that is not, and the agent's fitness is identically decreased when it makes an incorrect decision. After each trial, the agent receives feedback on the correct eat/avoid response for the just-seen stimulus, thereby allowing the agent to learn the regularities of the environment throughout its lifetime.

In this environment of the passive learner, temporal changes in contextual cues are generated by the environment. In the sea-floor scenario, context cues might change with daylight, tides, or seasons. For example, what is good to eat at high tide might be poor eating at low tide. Context could also be the presence or absence of schools of fish, which may occur more randomly and not a fixed intervals. For example, what is poor eating when schools of jellyfish are around might be good to eat when the waters are clear.

Reproduction. Each agent sees a fixed number of stimuli during its lifetime and its total fitness level is the sum of the trial-by-trial fitness that it has accumulated across all learning trials. Agents for the next generation are selected on the basis of the current agents' fitnesses, in one of three ways. First, the next generation can be generated using crossover and mutation. In crossover, two agents are selected from the current population, with the probability of selection directly related to the relative fitness. The genetic specifications of the two agents are spliced together, and a small degree of random mutation is applied, to generate an agent to be used in the next population. A second method of reproduction is by mutation only. Again, agents with higher relative fitness have higher probability of being chosen as progenitors of mutated offspring, but there is no crossover with other agents. Finally, the third method of reproduction ensures that the best current solution is not wiped out by a mutation or ill-advised mating. The genetic specification of the highest-fitness agent in the population is simply copied into the next generation. Over time, this random but fitness-driven selection process should result in populations of networks that are very good at performing in a context environment.

Assessing selective attention by highlighting

As with the hill-climbing simulations, a highlighting task was used to determine whether the evolved agents were exhibiting signs of selective attention. At the end of each agent's lifetime, its learned connection weights were reset to 0 and it was subjected to a highlighting task as described previously for the hill-climbing simulations. The agent's responses during both the learning and testing phases of the highlighting task were recorded for analysis but played no role in the agent's fitness and thus the highlighting task performance had no effect on the evolutionary trajectory. An agent was labeled as "highlighting" if the ordinality its responses to the ambiguous I and PE.PL cues showed the signature torsion in preferences described previously, i.e., if its output activation preferred E to L for input I, and its output activation preferred L to E for input PE.PL.

Simulation Parameters

For the results reported below, the GA was run with the following parameters: Each simulated population ran for 4,000 generations and each generation consisted of 100 agents. The initial population was seeded with the "base" agents shown in Figure 6. Each agent had fixed 1-to-1 connections with a weight of 10 between each input node and its direct hidden node. Initially there were no other connections between the input and hidden layer. Each hidden node was connected to the output node with a slow learning connection with an initial weight of 0 and a slow learning rate selected from a gaussian distribution centered on .05 with a standard deviation of .04. If the selected learning rate was negative, the absolute value of the rate was used, ensuring that all initial learning rates were small positive values. It should be noted that this base agent behaves as a single-layer network and cannot perform perfectly in a non-linear discrimination such as the context-dependent-relevance environment.

At the end of each generation, the parents for the next generation were selected. The first step of this process was to perform copy the most fit agent from the current generation into the new generation without any crossover or mutation (elitist selection with $N = 1$). This step ensured that the best performing agent's genetic structure was not corrupted by a non-adaptive mutation or cross-over.

After the best creature has been copied, the rest of the parents were selected using fitness proportionate selection. In order to provide the maximum differentiation between agents even when the population was converging on similar solutions, the agents' actual fitness values were modified before the selection process. Each agent's fitness was reduced by a value slightly less than the worst-performing agent's fitness. This subtraction had the effect of stretching the fitness values so that worst agents had a "relative fitness" value of near zero while the best agent's relative fitness values was equal to the difference in fitness between the best and worst agents in the population. All agents were then subjected to roulette wheel selection based on their relative fitness values. Each agent that was selected as a parent had a 50% chance of creating offspring via sexual reproduction and a 50% chance of asexual reproduction.

Agents selected for asexual reproduction were subjected to a mutation process where each gene (connection types, learning rates, etc.) had a small chance of being mutated. The exact mutation rate was selected so that there would be approximately one mutation that is expressed in the symmetric agent's phenotype per generation. Once a particular gene was selected for mutation, the new value for that gene was drawn from a gaussian distribution centered on the gene's current value and with a standard deviation proportional to the gene's current value. This process ensured that small values underwent small changes while larger values could change more drastically in a given mutation.

Agents selected for sexual reproduction underwent a crossover process to generate an offspring. Once the two parents were selected, two distinct crossover operations were performed. First, the matrix of connection information was crossed-over, creating an architecture that was a hybrid of the architectures from the two parents. Instead of the classic technique of treating the matrix as a single long vector and selecting a crossover point (or points), the crossover operation acted on the level of entire rows and/or columns of the matrix. The first step of this process was to set the offspring's connection matrix to be an exact copy of the first parent's matrix. Then specific rows/columns from the second parent's connection matrix were copied into the corresponding rows/columns of the offspring. Transplanting an entire row of the connection matrix from a parent had the effect of copying all of the "outgoing" connection information – connection types and connection weights – from a particular node in the architecture. For example, if the third row of a parent's matrix was crossed-over to the offspring, then all of the parent network's connections from node three to other nodes in the network would be copied into the offspring. Similarly, transplanting an entire column of the connection matrix had the effect of copying all of the "incoming" connection information. The particular rows and columns that were crossed over were selected randomly, and the crossover could select a single row and/or column, multiple contiguous rows and/or columns, or multiple non-contiguous rows and/or columns.

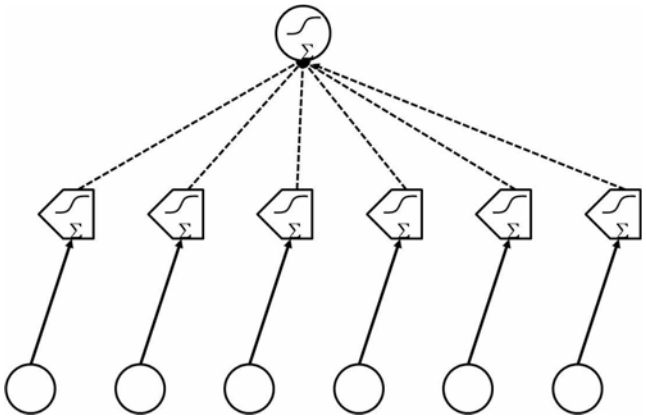


Figure 6. The architecture of the initial population of agents for all evolutionary simulations. The input and hidden layers are connected through non-learning 1-to-1 connections and the hidden and output layers are connected through slow learning connections. Multiplicative attention is not active.

Once the connection matrix crossover operation was complete, the second crossover operation was performed. In this procedure the genetic material specifying the agent's learning rates was crossed over. In the agents the learning rate settings were stored as a vector of floating point values specifying the slow and fast learning rates for each layer of the network. As with the connection matrix, the first step of the process was to make an exact copy of the first parent's learning rate settings in the offspring. Next, a standard version of crossover was implemented. A single crossover point was randomly selected in the offspring's learning rate vector and the learning rate settings from the second parent were spliced into the offspring from that point forward. Following crossover, the resulting agent underwent the same mutation process described for asexual reproduction.

At the beginning of each generation a randomly selected context environment was created and all agents were trained in the same type of environment. Agents were exposed to a randomized block of the four stimuli from one context, then the context was switched and the four trials from the second context were randomly presented. This continued for a total of 80 learning trials. Within each generation, all agents were presented with the same sequence of training trials: The mapping of the context and focal cues, the context switches, and the specific trial order was identical across agents. To monitor the overall accuracy of the evolving agents, at the end of an agent's lifetime it was presented with a randomized block of the 8 stimuli from the environment and its responses to those stimuli were recorded. As with the highlighting test, these trials had no bearing on the agent's fitness. The agent was labeled as "successful" in the context environment if it made ordinally correct decisions across all 8 of these test trials.

Results: Evolved learners exhibit highlighting

Because evolution via a genetic algorithm is an inherently probabilistic and noisy process, data must be collected from large numbers of simulations and then analyzed both aggregately and as independent evolutionary runs before strong conclusions can be drawn. In the first set we ran 50 different populations for 4,000 generations each, tracking for each generation the agent's fitness, accuracy, highlighting-status, successfulness, and details about their architectures and learning rates.

Across the 50 populations, the evolved networks diverged into two distinct architectural solutions, shown in Figure 7. There was a local-maximum in the fitness landscape not far from where the populations began. Populations quickly evolved to have fixed-weight connections between the input and hidden layers and with learning connections from the hidden layer to the output node. This solution cannot achieve 100% accuracy in the context environment, but it can make correct eat/avoid decisions on 7 of the 8 stimuli, which is better accuracy than many other potential architectures. Almost all of the simulated populations converged on this architecture in their early generations, but the vast majority eventually evolved away from this sub-optimal solution. Only 3 of the 50 simulations reached 4,000 generations without moving away from this architecture.

The remaining 47 populations evolved to a higher-accuracy architecture marked by learning connections on both the hidden and output levels. Not surprisingly, these populations performed well in the context environment, with nearly all of the agents in each population achieving perfect ordinal accuracy on the final 8 testing trials. Within this architecture, two distinct solutions were found by the simulations. In 43 of the 47 successful runs, a matched-rate solution evolved. These populations converged on a solution where the learning rate on the hidden layer and the learning rate on the output layer evolved to be similar values (typically around 30). In the other four runs, the populations converged on a solution with a high output learning rate. These agents evolved a learning rate to the output layer that was 40 to 50 times higher than the learning rate on the hidden layer (with rates between 70 and 100 on the output layer and 1 to 2 on the hidden layer).

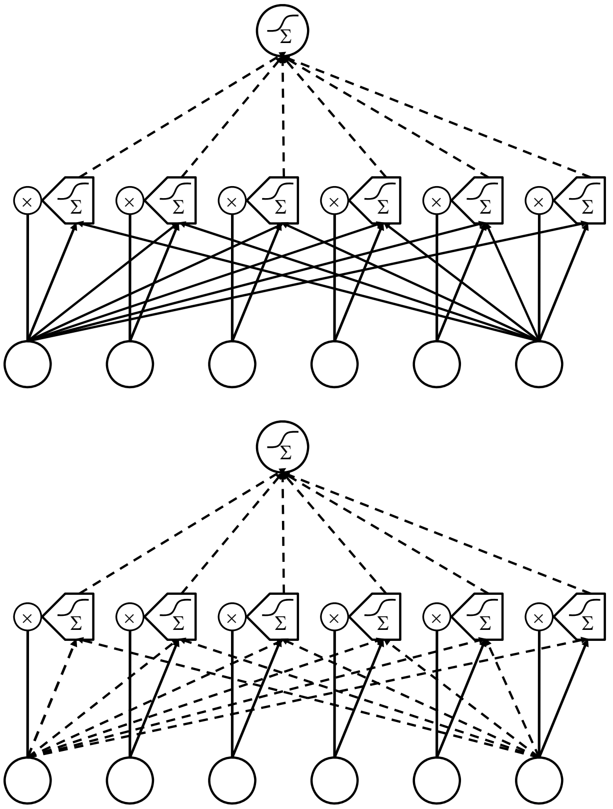


Figure 7. Two classes of evolved architectures. The top architecture has all fixed-weight (non-learning) connections from the input to the hidden layer and learning connections to the output layer. These agents can only perform a linear division of the solution space and as a result can respond correctly on up to 7 of the 8 distinct learning trials. The bottom architecture has fixed 1-to-1 weights on the direct input connections and first-fast learning connections between input nodes as the adjacent hidden nodes. The connections between the hidden nodes and the output were learning connections, slow-learning in some agents and first-fast learning in others. As a result of learning connections on the lower layer, the agents in this class of architectures are not limited to a linear discrimination of the solution space. Therefore, these agents can learn to respond correctly on all 8 of the distinct learning trials.

Both the matched-rate solution and the high output-rate solutions learned the environment quickly, and the evolved agents were not making any *ordinal* mistakes on the learning trials at the end of their lives. However, only the matched-rate agents showed signs of attentional learning as tested with the highlighting task. After the highlighting training, the matched-rate agents show a moderate preference for response E when probed with cue I ($E \approx 0.6$, $L \approx 0.4$) and a similar preference for response L when probed with cue PE.PL ($E \approx 0.4$, $L \approx 0.6$). Overall the torsion of preference is not quite as strong as that seen in the hill-climbing simulations, but it does demonstrate clear highlighting effects. The high output-rate solutions showed no preference for E or L when presented with either of the critical test items. Therefore, it appears as though the matched-rate agents were solving the context environment through the use of learned attentional mechanisms while the high output-rate agents were using a non-attentional solution.

This distinction between the attentional and non-

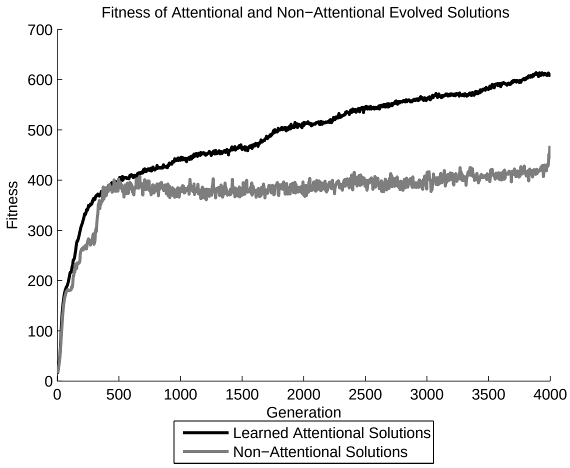


Figure 8. A comparison of the average fitness of the best-performing agents that exhibit learned attention (i.e. showed signs of highlighting) vs. agents that learned the correct responses to all 8 training items in the context environment but did not exhibit signs of attention. Attentional mechanisms clearly confer a fitness benefit in the context environment.

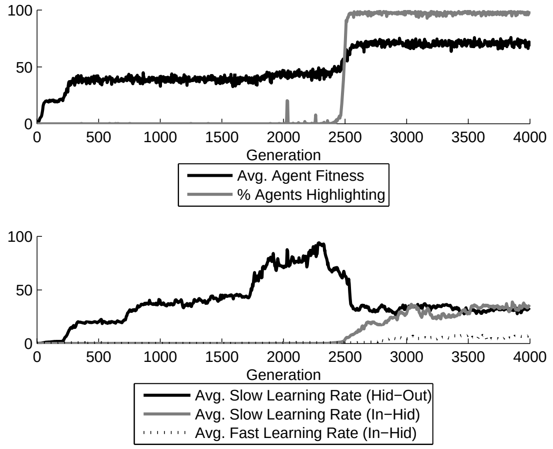


Figure 9. Data from a single population showing the relationship between learning rates, fitness, and highlighting across generations. The top graph shows the average agent fitness and the percentage of the agents in the population that show highlighting effects. Fitness has been scaled so that a fitness level of 100 indicates perfect performance across all learning trials. The lower graph displays the average learning rates for the same population.

attentional solutions allows us to look at whether or not an attentional mechanism is truly adaptive in the context environment. If attention is one of many equivalent ways to learn quickly and perform well in the environment, then we should not see any clear difference between the overall fitness of the attentional creatures when compared to those that do not exhibit signs of learned attention. Figure 8 shows that this is not the case. If we plot the average fitness levels of the best-performing 10% of the agents in both the attentional and non-attentional solutions, we see that the agents with learned attention are clearly outperforming the agents that have not evolved an attention-based solution.

Analysis of the architectures shows that the evolved agents match our interpretation of learned attention. Recall that the evolved agents are constrained to have fixed 1-to-1 connections between the input nodes and their direct hidden nodes, making the hidden layer a internalized representation of the outside world. In these networks attention can then be thought of as any mechanism that operates on those internal representations to either enhance or suppress the strength of the internal representations. Under this definition of attention, the learned connections from input nodes to adjacent hidden notes are the implementation of a learned attention mechanism. This conceptualization of attention fits well with the evolved solutions. In the matched rate solutions, the learning rate on the connections between the input nodes and their adjacent hidden nodes were reasonably high, allowing for the presence of particular cues in the environment to cause the internal representation of other cues to be enhanced or suppressed as dictated by the structure of the environment. These are the agents that showed the hallmarks of attentional learning as measured by their responses on the highlighting task.

In the high output-rate solutions, the learning rate on the hidden-to-output layer was very high. When trained using backpropagation, this arrangement means that most of the error-driven weight changes happen at the output level, and little error signal is propagated to the lower level. The small error that is propagated has even less influence because of very low learning rates between the input and hidden nodes. Therefore the high output-rate solutions do not learn to effectively enhance or suppress the critical internal representations and do not show the corresponding signs of learned attention.

Further evidence that learned attention (as represented by fast learning of activation or suppression of internal representations based on input cues) does confer an adaptive benefit can be seen in the evolution of the learning rates and fitness for single populations. Figure 9 shows plots from a population that eventually evolved attentional agents. While this plot is the most clear example of the connection between the learning rates and fitness, nearly all of the 43 runs that evolved attentional mechanisms show the same basic relationship between the learning rates and fitness.

Early in the simulated evolution, the agents evolve fixed-weight connections between the input and hidden layers and learning connections from the hidden layer to the output node, allowing them to learn correct responses to 7 of the 8 stimuli. In Figure 9, the result of this architectural improvement can be seen by the plateau in average fitness that begins around generation 300, where the fixed-weight architecture takes over the population. The fixed-weight architecture dominates the population until generation 1800, when the population's average fitness moves to a slightly higher plateau. This new architecture is the high-output rate architecture described above. The learning rate for the hidden-to-output layer is at its highest levels during this period of evolution. Simultaneously at the beginning of this period, the connections on the input-to-hidden layer make the switch to learning connections, but with very low learning rates. This solution offers a slight improvement, but it does not yet maximize performance.

Around generation 2400, the population shows a dramatic improvement in fitness. It is only when the learning rates (both slow and first-fast) at the input-to-hidden level increase, and the learning rates on the hidden-to-output level decrease, that the fitness is maximized. The increase in the input-to-hidden layer learning rate allows an agent to shift attention towards or away from focal cues based on the context cues. The decrease in the hidden-to-output learning rate allows more of the error signal to be propagated to the lower layer, creating more efficient error-driven attentional shifts. These changes, which can be seen occurring between generations 2400 and 2700 in Figure 9, coincide with the population's transition from non-highlighting agents to highlighting agents, as shown in the upper panel.

Quantifying the improvement in fitness. It is useful to quantify the improvement in average fitness when a population makes the evolutionary step from non-highlighting to highlighting. For these calculations, a population was considered to be a *non*-attentional population until 10 consecutive generations all had at most 10% of the population showing attention shifting, as measured by the ordinal highlighting torsion described above. A population was considered to be an attentional population when over 90% of the agents in the population showed the signature highlighting torsion for 10 consecutive generations. Populations that had between 10% and 90% attentional agents were considered to be in transition. For example, in Figure 9, the population is non-attention shifting before generation 2471, at which point 10 straight generations all had at least 10% of the population showing signs of attention shifting as measured by highlighting. The transition period lasted until generation 2530, when 10 consecutive generations each had greater than 90% of the population producing responses consistent with highlighting.

As a measure of how much the fitness improves from non-attentional populations to attentional populations, we calculated the difference of mean fitnesses across phases relative to the standard deviation of fitnesses within phases. This measure is analogous to "effect size" in statistics and d' in signal detection theory. To calculate the effect size, the average fitness of the final 100 generations of the non-attentional population (generations 2371–2470 in the simulation shown in Figure 9) was subtracted from the average fitness of the first 100 generations of the attentional population (generations 2530–2629 in Figure 9), and the difference of the two means was divided by the average standard deviation within the two windows. For Figure 9, this yields an effect size of 5.5. In other words, when the population changes from non-attentional to attentional, the fitness improves by more than 5 standard deviations of ordinary generation-to-generation variation.

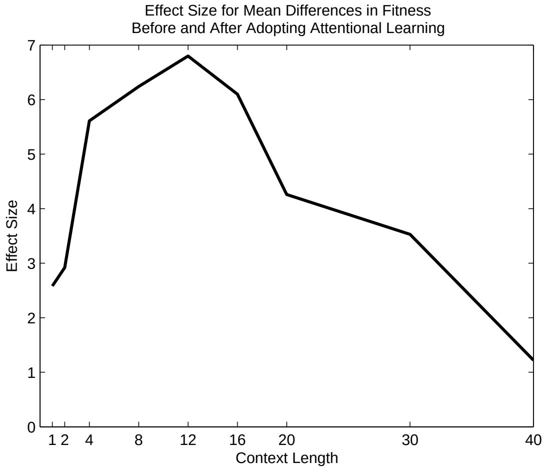


Figure 10. Effect size for the improvement in fitness from non-attentional to attentional populations, as a function of number of trials between context shifts.

The dynamics of context duration

We have shown that attentional mechanisms provide learning benefits in a cue-outcome structure where there are context-dependent relevances of cues, when the context switches at a particular rate. In this section we explore the robustness of the attentional advantage as a function of the rate of context switches. We find that the attentional advantage is robust across different rates of context switching, but is strongest at an intermediate rate. We explain the reasons for this "sweet spot" in the rate of context switches, and gain some additional insight into why attention can help learning.

We ran simulations that were identical to the simulations described above except that the number of trials between context shifts was changed. In the first simulation, the agents saw a single trial in the first context before switching to the next context for one trial, then back to the first context. In the second simulation, the agents saw 2 trials from a particular context before a switch, and so on. In total, we tested context durations of 1, 2, 4, 8, 12, 16, 20, 30 and 40 trials. In these runs, the agent's lifetime was always set to 80 trials; consequently, in the fastest shifting environment the agent would experience 79 context shifts and in the slowest shifting environment, the agent would only experience a single context shift.

Figure 10 shows the benefit of adopting an attentional architecture as a function of the number of trials between context shifts. Each data point on the graph represents the average effect size across 50 simulations for each context length. The graph of effect sizes shows an inverted U-shaped relationship between the relative benefit provided by the evolution of attentional mechanisms and the context length. While the populations all benefited from the evolution of attentional mechanisms, the shortest and longest context lengths exhibited smaller gains in fitness than the populations evolved in moderate-length contexts.

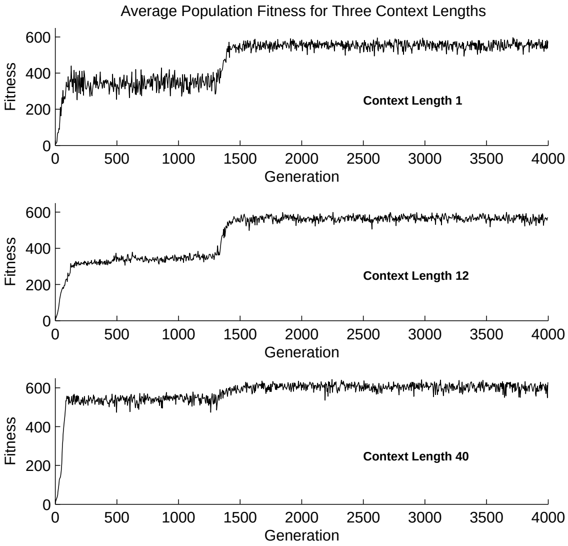


Figure 11. Average population fitness for three simulations run with differing context lengths. From top to bottom the three graphs show the fitness across generations for populations that experienced context shifts after every trial, after every 12 trials and after 40 trials, respectively. In all three simulations, the populations made the transition to attention shifting architectures around generation 1400.

Figure 11 helps to interpret this result. When the context duration is brief, and context is changing very frequently, the context cues vary as frequently as the outcomes and the focal cues. Consequently, the associations from context cues to outcomes can track spurious short-term covariation, leading to difficulty discovering a stable solution. Without sustained time in a given context to learn which cues are focal and which are context, building the right associations is a challenging task. Therefore, in the generations before the population makes the shift to architectures with learned attention, there is considerable variation in individual agent's success, and large variance in the average population fitness. This variability can be seen in the upper panel of Figure 11 as the large variation in fitness for generations 200–1300, as if a seismograph were recording a large and sustained earth quake. Nevertheless, attentional learning was still highly beneficial. When the populations made the transition to attentional mechanisms, at around generation 1400, the average fitness increased drastically. In fact, the mean increase in fitness from pre-attentional to attentional populations is approximately the same for the short-duration contexts as it is for the moderate-duration contexts, as can be seen by comparing the top and middle panels of Figure 11. But the effect size is smaller for the short-duration context because of the larger variability from one generation to the next.

For moderate context durations, the context was changing far less frequently than the focal cues, making the environmental regularities easier to learn. However, across the agent's lifetime, there were still a large number of context shifts, so in order to perform well the agents must not only have reacted quickly at the context boundaries, but also have preserved the associations from the previous context, so that those associations do not have to be learned again when the context recurs. The attentional mechanism promotes fast learning of new associations by shunting attention away from previously learned associations that are causing errors in the new context. This allows for the rapid acquisition of new associations on the most diagnostic cues, and it allows for preservation of previous associations on the cues that are no longer relevant in the new context. As a result, when an agent evolves an attention shifting architecture, it rapidly dominates the population and the population's average fitness drastically increases.

As the context duration extends, the few errors produced in the first few trials of a new context become less costly to the agents, so fast learning becomes less of a priority. At the extreme, where agents experience only a single context shift, we see only a small improvement in fitness for the evolution of attentional mechanisms. The bottom panel of Figure 11 shows that when there is only a single context shift in an agent's lifetime (i.e., context length 40), the pre-attentional agents were already performing very well. With only one context shift to contend with, the networks rapidly evolved architectures like those of the non-successful, non-highlighting, relatively-poor solutions from the original 4-trial context simulations. In the 40-trial context environment, the best pre-attentional agents had fixed-weights across all connections between the input and hidden layer, and learning connections with moderately high learning rates connecting the hidden and output layers. Analysis of these networks shows that the high learning rates allow them to quickly and accurately learn the regularities of the context into which they are born. They perform well for 40 learning trials, and then switch contexts. In the first few trials of the new context they make a few errors, but the high learning rates of the output connections quickly learn associations for the mappings of the new context. Since the agents will never return to the original context, it is of little consequence that the originally learned associations are overwritten at the context boundary. In this environment attention does not provide a strong adaptive benefit because fast learning rates alone are enough to perform nearly optimally.

In summary, attentional mechanisms can be especially beneficial when stable and useful associations should be retained for future re-use, despite a temporary change to a new context in which the associations are not useful. When the changes of context are very frequent, there is a benefit from attentional mechanisms, but the frequent changes of context cause learning to be noisy within a context, and therefore cause the relative benefit of attention to be diluted. When the changes in context are rare, then only rarely are there costs incurred from the context change, and therefore only little advantage is gained by attentional mechanisms.

Discussion

Our simulations have demonstrated that selection of fast learners at the behavioral level, as measured by high accuracy over the course of learning, also favors attentional learning at the behavioral level, as measured by exhibition of highlighting. We have shown that the fastest learning at the behavioral level is instantiated at the mechanistic level by particular backprop architectures that include learnable contextual modulation of cue activations. These demonstrations explored a delimited space of possible mechanistic instantiations. Future work will explore a wider range of possible learning architectures and mechanisms. Our demonstrations also explored a limited range of learning environments. The next section reports results from additional variations in environments, to bolster our suggestion that attentional learning may facilitate overall accuracy in a wide array of situations.

Environments that encourage attentional learning

We have emphasized a particular environmental structure for which learned selective attention is adaptive, namely, the structure in Table 1 that expresses context-dependent cue relevance. Presumably, there are variations of this environment that would also engender attentional learning. We believe that a key motivator for the evolution of attentional learning is the combination of an environmental contingency structure in which cue relevance varies according to context, with a reproductive advantage given to fast learners. The structure in Table 1 (illustrated in Figure 1) was our attempt to distill the essence of such an environment.

We speculate that environments with more contextual dependencies would produce even stronger benefits for attentional learning. For such environments, most cues would be irrelevant in most contexts. Environments in which there is massive irrelevance can be very costly to learning agents, because learning will track the irrelevant variation and cause error on subsequent occasions, or at least be costly metabolically. These costs can be mitigated by learning to suppress attention to irrelevant cues, according to context. Therefore, one goal for future research is to simulate environments that expand the basic structure shown in Table 1 across many more cues.

In the remainder of this section we describe two other environments that also yield an advantage for attentional learners. In both environments, the cue-outcome mapping is linearly separable, unlike the structure of Table 1. Because the structures are linearly separable, perfect accuracy can be achieved with only a single layer of connections, and there is no structural necessity to evolve an attentional layer in the network. Nevertheless, when fitness is based on speed of learning, not just eventual accuracy, attentional architectures do evolve.

Table 2
A linearly separable training environment that has context-dependent relevancies and four outcomes. See corresponding illustration in Figure 12.

Context		Focal Cues				Outcomes			
I	J	A	B	C	D	X	Y	V	W
1	0	1	0	1	0	1	0	0	0
1	0	1	0	0	1	1	0	0	0
1	0	0	1	1	0	0	1	0	0
1	0	0	1	0	1	0	1	0	0
0	1	1	0	1	0	0	0	1	0
0	1	1	0	0	1	0	0	0	1
0	1	0	1	1	0	0	0	1	0
0	1	0	1	0	1	0	0	0	1

Note: Presence of a cue or outcome is denoted by a 1, and absence is denoted by a 0.

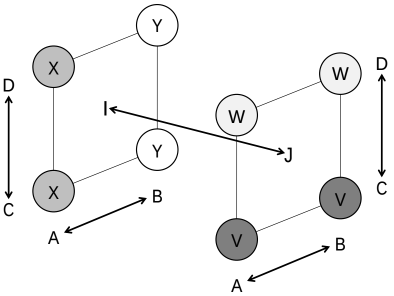


Figure 12. Spatial representation of the training structure in Table 2. Each circle represents a combination of cues A, B, C, D, I, and J. The letter in the circle, V, W, X, or Y, represents the correct outcome for that cue combination. The square on the left has a letter I in its center to indicate that cue I is present, while the square on the right has a letter J in its center to indicate that cue J is present. The upper circles have cue D present, while the lower circles have cue C present. The remaining dimension marks whether cue A or cue B is present. Although this diagram represents A–B, C–D, and I–J as if on dimensions, the basic structure does *not* encode or assume any dimensional relationship among the cues.

Linearly separable, four outcomes, with contextual dependency. Table 2 and Figure 12 show a training environment in which both the relevant cues and the outcomes depend on the context. Notice that in this structure, there are four outcomes instead of only two. Like the training environment previously studied in Table 1, when context cue I is present, focal cues A and B are relevant, but when context cue J is present, focal cues C and D are relevant. This new structure has different outcomes in the two contexts. Specifically, outcomes X and Y occur in context I, but outcomes V and W occur in context J.

The reason this structure in Table 2 is interesting is that it is linearly separable, unlike the previous structure. This linear separability means that the mapping can be solved merely by learning the connections fanning into the outcome nodes, and there is no need to learn any "lateral" connections to the hidden nodes. In other words, the cue-outcome contingencies by themselves do not demand any learned attention.

Despite not needing attentional learning to correctly solve the mapping, the solution can be learned more quickly when

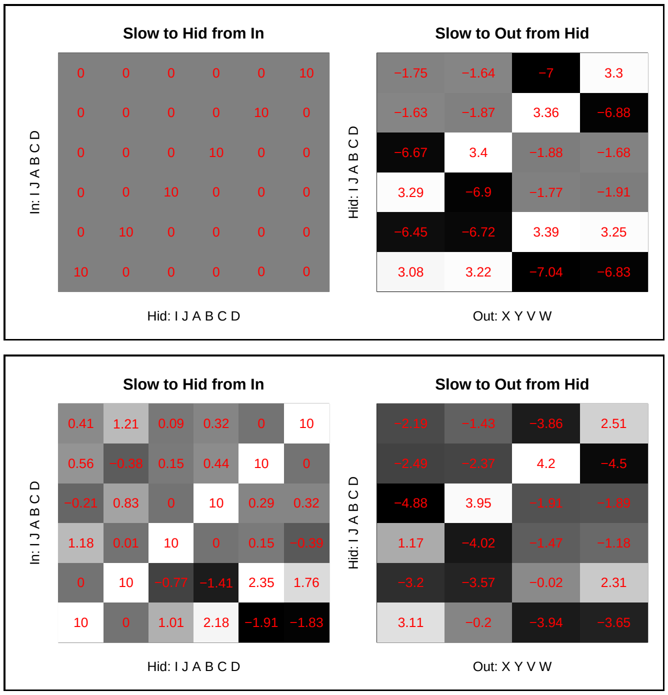


Figure 13. Examples of learned weights when trained on the linearly-separable, four-outcome structure shown in Table 2 and Figure 12. The upper panel shows a typical run when using the best learning rate on the output nodes and *when there is zero learning of connections fanning into the hidden nodes*, resulting in RMSD=0.211. Notice that the weights to the hidden nodes from the input nodes, shown in the upper left matrix, remain fixed at their starting values of zero (except for the diagonal weights, which are fixed at 10). The lower panel shows a typical run when using the best learning rates when there *is* learning allowed for connections fanning into the hidden nodes, resulting in RMSD=0.174. Notice that the weights to hidden nodes from input nodes, in the lower left matrix, have mostly learned non-zero values. The model has learned that when cue I is present, attend to cues A and B and suppress cues C and D (and the opposite for when cue J is present). The learning rates for the lower panel *do* produce highlighting, but the learning rates for the upper panel do *not* produce highlighting.

attentional learning is available. This claim is confirmed through hill-climbing optimizations. Consider first a restricted architecture in which there is no multiplicative gating and in which the learning rates on the connections fanning into the hidden nodes are set to *zero*. This is like the “base-agent” architecture shown in Figure 6. We use hill-climbing optimization to find the optimal learning rate for the outcome nodes. In this case, the accuracy of prediction gets fairly good, with RMSD=0.211, because the outcome layer alone can solve the mapping. A representative run for the best outcome-learning rate is shown in the upper panel of Figure 13. In the subsequent highlighting test, however, *no* highlighting is exhibited. No highlighting occurs because there is no attentional shifting at the hidden nodes.

When the architecture includes multiplicative gating and learning of hidden-node connections (as in Figure 2), then the problem is solved with far less total error (RMSD=0.174). A representative solution is shown in the lower panel of Figure 13. Importantly, the subsequent highlighting test shows very robust highlighting effects. This effect occurs because highlighting is mediated in these networks by the learning of attentionally-modulating connections from cues to their hidden-layer representations. The point here is that even though the attentionally-modulating connections are not necessary to accurately solve the cue-outcome mapping, those learnable connections do improve the speed of learning. Those learnable connections also, as a side effect, engender highlighting. Again, this result supports our general claim that faster learning can be accomplished by attentional learning, in this case even when attentional learning is not necessary to solve the task.

Table 3
A training environment that is linearly separable, with no structurally distinct context cue. The “context” and “focal” cues are structured identically, so the labeling is arbitrary. See corresponding illustration in Figure 14.

Context		Focal Cues				Outcomes	
I	J	A	B	C	D	X	Y
1	0	1	0	1	0	1	0
1	0	1	0	0	1	1	0
1	0	0	1	1	0	1	0
1	0	0	1	0	1	0	1
0	1	1	0	1	0	1	0
0	1	1	0	0	1	0	1
0	1	0	1	1	0	0	1
0	1	0	1	0	1	0	1

Note: Presence of a cue or outcome is denoted by a 1, and absence is denoted by a 0.

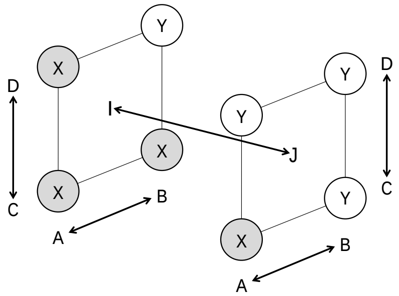


Figure 14. Spatial representation of the training structure in Table 3. Although this diagram represents A–B, C–D, and I–J as if on dimensions, the basic structure does *not* encode or assume any dimensional relationship among the cues.

Linearly separable with no contextual dependency. Table 3 and Figure 14 show a training environment in which the cue–outcome mapping is linearly separable, and the cues are structurally equivalent to each other. In other words, the labeling of one cue as “context” is completely arbitrary. (This structure corresponds to what Shepard et al. (1961) called Type IV, if the cues are represented on dimensions as shown in Figure 14.)

Consider first what happens when the 8 training items are randomly intermixed, such that each 8-trial block of training contains an independently permuted order of the 8 training items. For this training regime, there is no structural or temporal distinction whatsoever between context and focal cues. When all the learning rates are freely optimized, the weights fanning into the hidden nodes have only small learning rates. Consequently, when tested on highlighting, the optimal-learning network does not show highlighting. This makes intuitive sense: When the training structure is completely symmetric and provides little benefit from attentional learning, then little if any highlighting will be exhibited.

Consider what happens, however, if we make one of the cues relatively stable across trials during training, such that the cue behaves as a temporal context cue. For this simulation, we blocked together the four trials with cue I, and alternated them with blocks of four trials with cue J. Otherwise the training was the same as before. For this blocked-context training procedure, the best learning rates on the hidden nodes were reasonably high. Examination of the learned weights revealed why the attentional learning was beneficial. The model learned that when (context) cue I was present, cues A and C should be attended while cues B and D should be suppressed, but when (context) cue J was present, cues B and D should be attended but cues A and C should be suppressed. Moreover, these attentional weights were symmetric in sign, such that, for example, when cues A or C were present, they gave attention to cue I but suppressed cue J. Importantly, when subsequently tested in the highlighting procedure, the network showed robust highlighting effects.

The reason the network exhibited highlighting is the same as explained for previous simulations: When there is significant learnability of connections between cues and their hidden-layer representations, then, in the highlighting procedure, the network learns to ignore the shared cue I in the presence of cue PL, as shown for example in Figure 4. This learned modulation of cue activations protects the initially learned association from hidden node I to outcome E, and yields a strong association from hidden node PL to outcome L, again as shown in Figure 4.

In summary, the structure of Table 3 and Figure 14 is symmetric and does not demand large learning rates on the hidden nodes when the items are trained in random order. But when one pair of cues is alternated more slowly than the other cues, whereby the relatively tonic cues serve as context for the others, the model does benefit from relatively larger learning rates on the hidden layer, and consequently shows clear attentional learning as assayed by highlighting.

Summary: Environments that encourage attentional learning. The training structure in Table 1 and Figure 1 was designed to embody two key qualities that might encourage attentional learning. The structure incorporated cues that were individually uncorrelated with the outcomes, but which indicated what other cues were predictive of the correct outcomes. The indirectly relevant “context” cues were also held relatively constant across training trials. We showed that the structure did indeed encourage learning architectures and learning rates that exhibited attentional effects, when the goal was to maximize total accuracy across the lifetime of training.

The structures of the present section (Table 2 and Table 3) begin to expand and delimit the range of environments in which attentional learning is improves overall accuracy. We showed that even when hidden-layer learning is not necessary for eventual perfect accuracy, attentional learning can still be beneficial for acquiring the mapping quickly. This benefit of attentional learning for overall accuracy is especially strong when there are contextual cues to relevance, as exemplified for structural context cues by Tables 1 and 2, and for temporal context cues by Table 3. In the latter case, even when the mapping is linearly separable and perfectly symmetric, if one cue merely alternates more slowly than the others, and thereby may serve as a context cue, then attentional learning is adaptive. Importantly, we also showed that attentional learning is not merely trivially *always* adaptive. Specifically, when the structure is symmetric and training is ordered randomly, then the best hidden-layer learning rates are weak and little if any highlighting occurs. Another case in which attentional learning had little benefit was when training with the original structure of Table 1 but alternating the context only once; recall that in this case the evolved architecture settled on no learning of hidden-layer weights because the cost of the single context switch could be absorbed by fast outcome-weight learning alone.

Costs and benefits of selective attention

When considering selective attention, many people think of it as a necessary evil caused by limited-capacity processing in the brain and body. In a review of the causes and consequences of limited attention, Dukas (2004, p. 107) defined limited attention as a “restricted rate of information processing by the brain” and he focused on the costs of limited attention, such as hindering foragers probability of detecting cryptic food items, and failing to notice an approaching predator while engaged in an attention-demanding task. Clark and Dukas (2003) developed a detailed model of foraging while avoiding predators. They analyzed the optimal width of the focus of attention, and optimal processing capacity. They also assumed an accelerating metabolic cost of increased processing capacity. From the model, they concluded that the use of selective attention was an optimal solution to the trade-offs between foraging for cryptic food, avoiding predators, and sustaining a high-demand processing system.

In our approach, however, we have not had to assume an increased metabolic cost for higher learning rates. We have not had to assume any additional cost for inclusion of additional learnable connections. Instead, selective attention emerged as an optimal solution to an informational problem, not as a compromise conceded to physical shackles. In environments with context-dependent cue relevance, learnable allocation of attention can improve speed of learning.

Moore (2004) provided an extensive review of types of learning, including habituation, sensitization, discrimination learning, imprinting, navigation, mimicry, instrumental learning, language acquisition, etc. He suggested a large cladogram indicating possible evolutionary relationships among the various forms of learning. But nowhere in the catalog did there appear the notion of learning to attend. We believe, on the contrary, that learning what to attend to is a critical aspect of learning well.

References

Burgos, J. E. (2007). Evolving artificial neural networks in Pavlovian environments. In J. W. Donahoe & V. Packard-Dorsel (Eds.), *Neural-network models of cognition: Biobehavioral foundations* (pp. 58–80). North-Holland.

Chun, M. M. (2000). Contextual cueing of visual attention. *Trends in Cognitive Sciences*, *4*(5), 170–178.

Clark, C. W., & Dukas, R. (2003). The behavioral ecology of a cognitive constraint: limited attention. *Behavioral Ecology*, *14*(2), 151–156.

Dibbets, P., Maes, J. H. R., Boermans, K., & Vossen, J. M. H. (2001). Contextual dependencies in predictive learning. *Memory*, *9*(1), 29–38.

Dukas, R. (2004). Causes and consequences of limited attention. *Brain, Behavior and Evolution*, *63*, 197–210.

Edmonds, B., & Norling, E. (2007). Integrating learning and inference in multi-agent systems using cognitive context. *Lecture Notes In Computer Science*, *4442*, 142.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley.

Goodman, N. D., Tenenbaum, J. B., Feldman, J., & Griffiths, T. L. (2008). A rational analysis of rule-based concept learning. *Cognitive Science*, *32*(1), 108–154.

Hall, G., & Channell, S. (1985). A comparison of intradimensional and extradimensional shift learning in pigeons. *Behavioural Processes*, *10*, 285–295.

Hinton, G. E., & Plaut, D. C. (1987). Using fast weights to deblur old memories. In *Proceedings of the 9th annual conference of the cognitive science society* (pp. 177–186). Erlbaum.

Johnston, T. D. (1982). Selective costs and benefits in the evolution of learning. *Advances in the Study of Behavior*, *12*, 65–106.

Kamin, L. J. (1969). Predictability, surprise, attention, and conditioning. In B. A. Campbell & R. M. Church (Eds.), *Punishment* (pp. 279–296). New York: Appleton-Century-Crofts.

Kruschke, J. K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review*, *99*, 22–44.

Kruschke, J. K. (1996a). Base rates in category learning. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *22*, 3–26.

Kruschke, J. K. (1996b). Dimensional relevance shifts in category learning. *Connection Science*, *8*, 201–223.

Kruschke, J. K. (2001). Toward a unified model of attention in associative learning. *Journal of Mathematical Psychology*, *45*, 812-863.

Kruschke, J. K. (2003). Attentional theory is a viable explanation of the inverse base rate effect: A reply to Winman, Wennerholm, and Juslin (2003). *Journal of Experimental Psychology: Learning, Memory and Cognition*, *29*, 1396-1400.

Kruschke, J. K. (2010). Attentional highlighting in learning: A canonical experiment. In B. Ross (Ed.), *The psychology of learning and motivation* (Vol. **, pp. **–**). **: Academic Press. (Pre-print available at author's website, http://www.indiana.edu/∼kruschke)

Kruschke, J. K., & Blair, N. J. (2000). Blocking and backward blocking involve learned inattention. *Psychonomic Bulletin & Review*, *7*, 636-645.

Kruschke, J. K., Kappenman, E. S., & Hetrick, W. P. (2005). Eye gaze and individual differences consistent with learned attention in associative blocking and highlighting. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *31*, 830–845.

Little, D. R., & Lewandowsky, S. (2009). Beyond non-utilization: Irrelevant cues can gate learning in probabilistic categorization. *Journal of Experimental Psychology: Human Perception and Performance*, *35*(2), 530–550.

Lubow, R. E. (1989). *Latent inhibition and conditioned attention theory*. Cambridge UK: Cambridge University Press.

Mackintosh, N. J. (1975). A theory of attention: Variations in the associability of stimuli with reinforcement. *Psychological Review*, *82*, 276–298.

McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: the sequential learning problem. In G. Bower (Ed.), *The psychology of learning and motivation, vol. 24* (pp. 109–165). New York: Academic Press.

Medin, D. L., & Edelson, S. M. (1988). Problem structure and the use of base-rate information from experience. *Journal of Experimental Psychology: General*(117), 68–85.

Mery, F., & Kawecki, T. J. (2003). A fitness cost of learning ability in drosophila melanogaster. *Proceedings of the Royal Society B: Biological Sciences*, *270*(1532), 2465–2469.

Miller, G. F., & Todd, P. M. (1990). Exploring adaptive agency I: Theory and methods for simulating evolution of learning. In D. S. Touretzky, J. L. Elman, T. J. Sejnowski, & G. E. Hinton (Eds.), *Proceedings of the 1990 connectionist models summer school* (pp. 65–80). San Mateo, CA: Morgan Kaufmann.

Miller, R. R., & Matzel, L. D. (1988). The comparator hypothesis: A response rule for the expression of associations. In G. H. Bower (Ed.), *The psychology of learning and motivation: Advances in research and theory* (Vol. 22, pp. 51–92). San Diego, CA: Academic Press.

Moore, B. R. (2004). The evolution of learning. *Biological Review*, *79*, 301–335.

Nelson, J. B., & Sanjuan, M. (2006). A context-specific latent inhibition effect in a human conditioned suppression task. *The Quarterly Journal of Experimental Psychology*, *59*(6), 1003–1020.

Nosofsky, R. M., Gluck, M. A., Palmeri, T. J., McKinley, S. C., & Glauthier, P. (1994). Comparing models of rule-based classification learning: A replication of Shepard, Hovland, and Jenkins (1961). *Memory & Cognition*, *22*, 352–369.

Nosofsky, R. M., Palmeri, T. J., & McKinley, S. C. (1994). Rule-plus-exception model of classification learning. *Psychological Review*, *101*, 53–79.

Raine, N. E., & Chittka, L. (2008). The correlation of learning speed and natural foraging success in bumble-bees. *Proceedings of the Royal Society B: Biological Sciences*, *275*(1636), 803–808.

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and non-reinforcement. In A. H. Black & W. F. Prokasy (Eds.), *Classical conditioning II: Current research and theory* (pp. 64–99). New York: Appleton-Century-Crofts.

Rosas, J. M., & Callejas-Aguilera, J. E. (2006). Context switch effects on acquisition and extinction in human predictive learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *32*(3), 461–474.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by back-propagating errors. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel distributed processing* (Vol. 1). Cambridge, MA: MIT Press.

Schmajuk, N. A. (2002). *Latent inhibition and its neural substrates: From animal experiments to schizophrenia*. Norwell, MA: Kluwer Academic.

Schmajuk, N. A., Lam, Y. W., & Gray, J. A. (1996). Latent inhibition: A neural network approach. *Journal of Experimental Psychology: Animal Behavior Processes*, *22*, 321–349.

Shepard, R. N., Hovland, C. L., & Jenkins, H. M. (1961). Learning and memorization of classifications. *Psychological Monographs*, *75*(13). (Whole No. 517)

Slamecka, N. J. (1968). A methodological analysis of shift paradigms in human discrimination learning. *Psychological Bulletin*, *69*, 423–438.

Todd, P. M., & Miller, G. F. (1991). Exploring adaptive agency II: Simulating the evolution of adaptive learning. In J.-A. Meyer & S. W. Wilson (Eds.), *From animals to animats: Proceedings of the first international conference on simulation of adaptive behavior* (pp. 306–315). Cambridge, MA: MIT Press.

Yang, L.-X., & Lewandowsky, S. (2003). Context-gated knowledge partitioning in categorization. *Learning, Memory*, *29*(4), 663–679.