

How to implement HyGene into ACT-R

Cvetomir M. Dimov

Department of Organizational Behaviour, University of Lausanne, Switzerland

Author Note

Cvetomir M. Dimov, Department of Organizational Behaviour, Faculty of Business and Economics, University of Lausanne, Lausanne, Switzerland.

This research is embedded into the Swiss National Science Foundation grant 100014_146702 “Architectural process models of decision making”.

This research includes and extends work first published elsewhere (Dimov, 2016).

Correspondence concerning this article should be addressed to Cvetomir M. Dimov, Faculty of Business and Economics, Bâtiment Internef, Quartier UNIL-Chamberonne, University of Lausanne, CH-1015 Lausanne, Switzerland. Email: Cvetomir.Dimov@unil.ch

Word count: 8238

Number of figures: 7

How to implement HyGene into ACT-R

We investigate if and how the model of hypothesis generation and probability judgment HyGene can be implemented in ACT-R. We ground our endeavour on the formal comparison of the memory theories behind ACT-R and HyGene, whereby we contrast the predictions of the two as a function of prior history and current context. After demonstrating the convergence of the two memory theories, we provide a 3-step guide of how to translate a memory representation from HyGene into ACT-R. We also outline how HyGene's processing steps can be translated into ACT-R. We finish with a discussion of points of divergence between the two theories.

Keywords: ACT-R; HyGene; theory comparison; declarative memory

Introduction

Imagine that you are a reviewer at an established psychological journal and you are faced with the task of evaluating the probability that a manuscript will contribute to theory integration. How will you proceed? We are frequently faced with such problems of judging probability or likelihood in our daily lives and, consequently, a large amount of research has focused on explaining how we make such judgments. This research has uncovered many phenomena to be explained (e.g., *overconfidence*, Einhorn & Hogarth, 1978) and developed corresponding models to address those phenomena. These models include various heuristics from the *heuristics-and-biases research program* (e.g., *availability*, Tversky & Kahneman, 1973) and several ecological approaches (e.g., *Probabilistic Mental Models*, Gigerenzer, Hoffrage, & Kleinbölting, 1991; *Brunswikian Induction Algorithm for Social Cognition*, Fiedler, 1996; see also Fiedler, 2000).

Among these theories, *MINERVA-DM* (Dougherty, Gettys, & Ogden, 1999) emerged as a potential unifier. It brought together the ecological approaches and a theory of memory, *MINERVA2* (Hintzman, 1984), and it expanded that theory of

memory with a conditional likelihood estimation process to address many findings in probability and likelihood judgment. The successor of MINERVA-DM, *HyGene* (Thomas, Dougherty, Sprenger, & Harbison, 2008) – a model of how people acquire information from the environment, generate hypotheses from memory based on this information, test those hypotheses and judge the probability of each – is probably the most comprehensive theory of probability judgment currently.

Extensions of HyGene to model the dynamics of data acquisition and hypothesis generation

In its original formulation, HyGene did not address the dynamics of information acquisition. Mehlhorn, Taatgen, Lebiere, and Krems (2011) set forth to investigate how the dynamics of information acquisition influence the hypothesis generation process. In so doing, Mehlhorn et al. (2011) resorted to the *cognitive architecture ACT-R* (Anderson, 2007), which they used to construct 4 models of hypothesis generation. The 4 models weighted differently each piece of information currently in the focus of attention. ACT-R was necessary to the extent that it allowed for, first, modelling the information acquisition process, second, modelling differential weighting of items in working memory and, third, relating activation of an item in memory to its retrieval time. The activation-retrieval-time relation allowed Mehlhorn et al. (2011) to determine a hypothesis's activation via its retrieval time.

Lange, Thomas and Davelaar (2012) also set forth to investigate the dynamics of information acquisition. In so doing, they extended HyGene to include a dynamic working memory store, based on Davelaar, Goshen-Gottstein, Ashkenazi, Haarmann, and Usher (2005)'s model of cued recall, also in order to model how information acquisition dynamics affect hypothesis generation. The dynamic working memory store extension of HyGene was demonstrated to account for recency effects in the weighting

of evidence in the hypothesis generation process (Lange et al., 2012) and for differential weighting of each piece of evidence as a function of duration of presentation during the hypothesis generation process (Lange, Davelaar, & Thomas, 2013). Moreover, it was shown to be consistent with Mehlhorn et al.'s results. Mehlhorn et al. (2011) and Lange et al. (2012) are two successful examples of one formal theory extending the predictions of HyGene.

ACT-R in judgment and decision making

Mehlhorn et al. (2011) used ACT-R to model information acquisition and hypothesis generation. This is just one of many areas of application that this general theory of cognition has found. These areas are as diverse as analogy making (Salvucci & Anderson, 2001), past tense learning (Taatgen & Anderson, 2002), solving the Tower of Hanoi puzzle (Anderson & Douglass, 2001), associative recognition (Schneider & Anderson, 2012) and novice to expert transition in human-computer interaction (Paik, Kim, Ritter, & Reitter, 2015).

In the field of judgment and decision making specifically, ACT-R has found several other applications. Specifically, the memory system of this cognitive architecture has been used to investigate the properties of the fluency heuristic (Schooler & Hertwig, 2005), to explore the applicability of various decision strategies (Marewski & Schooler, 2011) and to model cue ordering in inference (Dimov & Link, 2017). In addition, the full architecture was used to construct 39 recognition-based decision models (Marewski & Mehlhorn, 2011) and to develop cue-based decision models (Dimov, Marewski, & Schooler, 2013; Fechner et al., 2016; Nellen, 2003). Nevertheless, to our knowledge, ACT-R has never been applied to probability and likelihood judgments. One way to investigate ACT-R's applicability to such phenomena, which will also avoid a large amount of duplication of effort, is to

implement HyGene in ACT-R. Moreover, as Mehlhorn et al. (2011) have exemplified, this has the potential to also benefit HyGene itself by enriching its predictions.

Benefits of implementing all of HyGene in ACT-R

ACT-R can enrich HyGene's predictions in three ways. First, ACT-R's retrieval equations provide predictions about both retrieval outcome and retrieval latency, whereas HyGene does not currently model latencies. Second, ACT-R includes perceptual and motor components with associated latency parameters, which complement the central cognitive components, thus allowing one to model the full process that an experimental subject undergoes, starting from observing a stimulus on the screen and finishing with a key press on a keyboard. As a result, an ACT-R model can predict absolute response times instead of relative response times and include information acquisition constraints present in the perceptual components and response constraints present in the motor components. Third, ACT-R has established module-to-brain mappings (e.g., Borst & Anderson, 2015a), which have been used to test models of relatively simple (Anderson, Qin, Sohn, Stenger, & Carter, 2003) and complex tasks (Anderson, Fincham, Schneider, & Yang, 2012) with fMRI, and to test models with EEG (Borst & Anderson, 2015b) and with MEG (Borst, Ghuman, & Anderson, 2016). Together, the richer predictions that ACT-R brings can allow researchers to develop models of hypothesis generation and judgment, which can be scrutinized both with outcome data and with, behavioural and neural, process data.

Outline

Our goal is to investigate if and how HyGene can be mapped onto ACT-R. We will start by introducing the two theories and by describing in detail the equations guiding their memory representations. We will then compare the predictions derived from those equations as a function of prior history and current context, and provide a 3-step guide

of how to relate the memory representations of the two theories. We will finish with outlining how HyGene's processing steps can be implemented in ACT-R and discuss limits to our approach.

The cognitive architecture ACT-R

Overview

ACT-R combines theories of perception, motor action and central aspects of cognition into a unified framework, which is implemented as a computer program. In ACT-R, each cognitive function is realized as a separate cognitive *module*. This cognitive architecture interacts with the environment through perceptual modules (an aural and a visual module) and motor modules (a vocal and a manual module). In addition, it has modules modelling central aspects of cognition: An intentional module tracks the current goal, a declarative module models information storage and retrieval, and an imaginal module stores task-relevant information. Finally, a timing module models prospective time perception by tracking the number of ticks that an internal accumulator generates during the interval being timed.

The modules' contents cannot be accessed directly, but only through communication channels with limited capacity, called *buffers*. Information in those buffers can only be accessed and moved around by the procedural module, which is instantiated as a central production system. This production system consists of if-then rules, or production rules, which attempt to match the current state of the buffers and, in case of a match, execute a change in the modules' state. Such a change might be, for example, a retrieval request in the case of the declarative module or a command to move a finger in case of the manual module.

Productions are the representation of choice for *procedural knowledge* (knowledge of how to do something). *Declarative knowledge* (knowledge of facts and experiences) in ACT-R is represented as *chunks*. A chunk consists of a set of slots with associated values (see Figure 1 for an example of a chunk with three slots). In ACT-R, all factual knowledge and experience is modelled as a collection of chunks and is stored in a single declarative memory storage.

[Figure 1 near here]

Declarative memory is populated by buffer clearing: Each time the contents of a buffer is cleared, it is either stored as a new chunk or, if the chunk is already present in declarative memory, it reinforces the already existing chunk, increasing its *activation*. Chunk activation quantifies the strength of a piece of memory and determines retrieval dynamics, such as response time or recall probability: The higher the activation, the more likely a fact is retrieved and, in case of a retrieval, the faster it is retrieved.

ACT-R also has a mechanism for retrieving an aggregate result of a set of chunks, called *blending* (Gonzalez, Lerch, & Lebiere, 2003, Appendix A). Unlike retrieving a specific chunk, blending weights chunks as a function of their activation and produces a weighted average value of a quantity of interest. This process can be used, for example, to produce continuous value estimates.

ACT-R's declarative memory

Here we will describe in detail the subsymbolic equations governing chunk activations, and consequently, retrieval and blending. Chunk activation A_i is modelled as a function of prior exposure (*base-level activation*, B_i), context relevance (*spreading activation*, SA_i), similarity to retrieval request (*partial matching*, PM_i) and noise:

$$A_i = B_i + SA_i + PM_i + \varepsilon = B_i + \sum_j W_j S_{ji} + P \sum_l M_{li} + \varepsilon. \quad (1)$$

Base-level activation B_i reflects how recently and how frequently the event or fact represented by chunk i was encountered in the past. The equation describing learning through base-level activation is:

$$B_i = \ln \sum_{j=1}^{n_i} t_j^{-d}, \quad (2)$$

where n_i is the number of presentations of an item represented by chunk i , t_j is the time since the j^{th} presentation, and d is the decay parameter. The decay parameter determines how quickly memories decay in time. According to the base-level equation, each time an item is presented, the activation of the chunk representing it increases. Then, with time, its activation decays according to a power function. An approximation of the base-level equation, which assumes that the object/event represented by the chunk is encountered periodically, is the optimized learning equation¹:

$$B_i = \ln \left[\frac{T_{L_i}^{-d}}{1-d} n_i \right] = \ln n_i - d \ln T_{L_i} - \ln 1 - d. \quad (3)$$

Aside from being simpler to calculate, optimized learning conveniently separates a chunk's base-level activation into two components: a number of presentations n_i and lifetime T_{L_i} (time since the chunk was first created). This conceptual separation will later come to use when comparing HyGene and ACT-R.

Context effects typically originate from the chunk currently placed in the imaginal buffer. This chunk spreads activation SA_i to chunk i in declarative memory as a function of the number of slots these two chunks have in common. Specifically, in Equation 1, W_j is the total amount of activation from source j (e.g., from the imaginal buffer), while S_{ji} is the strength of association from source j (e.g., a slot currently in the imaginal buffer) to chunk i . S_{ji} is 0 if chunk j is not the value of a slot of chunk i (also, j

and i should not be the same chunk). If the two chunks have common slots, the association strength is set using the following equation:

$$S_{ji} = S + \ln[P(i|j)]. \quad (4)$$

In this equation, S is the maximum associative strength and $P(i|j)$ is conditional probability of encountering chunk i , given that chunk j is present. This probability equals 1 if i and j are the same, meaning that each chunk is maximally associated to itself.

The partial matching mechanism allows for chunks to be retrieved, which do not correspond exactly to the retrieval request. Upon retrieval, all slots of the retrieval request are taken into consideration. Each chunk's activation changes as a function of the similarity between its slot values and the slot values from the retrieval request: In Equation 1, P is the weight given to each slot within the retrieval request, while M_{li} is the degree of similarity between slot value l in the retrieval request and corresponding slot value in chunk i in declarative memory.

Finally, the noise component ε consists of permanent noise and instantaneous noise computed at the time of a retrieval request. We will only consider instantaneous noise below. Instantaneous noise is drawn from a logistic distribution with a location $\mu = 0$.

It is important to emphasize that the various observables determined by the declarative system are not linearly related to activation. Instead, probability of successful retrieval, probability of retrieving one chunk among several matching chunks, weight in blending, and retrieval latency are all exponentially related to activation. Specifically, the probability of retrieving a chunk, given a retrieval threshold τ , is:

$$P_{retrieval} = \frac{1}{1 + e^{-\frac{A_i - \tau}{s}}}, \quad (5)$$

where s is the scale parameter of the logistic distribution of ε and τ is the retrieval threshold, which specifies the minimum activation required for a retrieval to succeed. In addition, if activations are sufficiently above threshold, such that retrieval failure is highly unlikely, the probability of selecting one among several chunks is described by the Boltzmann selection rule:

$$P_i = \frac{e^{A_i/s}}{\sum_j e^{A_j/s}} . \quad (6)$$

Finally, retrieval time decreases exponentially as activation increases:

$$RT_i = F e^{-A_i} . \quad (7)$$

Activation determines both retrieval of a specific chunk and the contribution of that chunk to the blending process. Specifically, the analogue of activation for the blended chunk (the so-called *match-score* M) equals:

$$M = \log \sum_{i \in MS} e^{A_i}, \quad (8)$$

where the summation is over the *match set* MS . The match score is in essence the sum of activations of all chunks included in the blending process. Just like for retrieval, observables are related to the exponent of the match score. As a consequence, all observables related to the match score are based on the sum of exponents of the blended chunks' activations:

$$e^M = \sum_{i \in MS} e^{A_i}. \quad (9)$$

HyGene

Overview

HyGene describes how people generate hypotheses and judge the probability of those hypotheses. It extends MINERVA-DM by adding semantic memory and working memory storages to its episodic memory storage. Both semantic memory and episodic

memory store memory traces as feature lists. Episodic memory stores traces based on direct experience, while semantic memory stores information abstracted from episodic memory or obtained outside of direct experience. The episodic storage can contain multiple equivalent traces, one for each encounter with an item or an event, and is thus sensitive to experiential base rates. Semantic memory does not contain such frequency information – it is in essence a list of prototypes. Finally, working memory temporarily stores a limited number of traces.

MINERVA2: HyGene’s episodic memory

The theory describing HyGene’s episodic memory is MINERVA2. It represents stimuli as feature lists, or vectors, which, upon encoding, are stored as episodic traces in memory. Features in MINERVA2 are assumed to be binary with values 1 or -1, if present, or 0 if there is no knowledge of that feature (see Figure 2 for an example of a vector with 12 features).

[Figure 2 near here]

Encoding in MINERVA2 is a noisy process, resulting in imperfect copies of the experienced events in episodic memory. This process is governed by a *learning rate parameter* L , which describes the probability of storing a value: With a probability of L a feature value is correctly stored as either 1 or -1, while with a probability of $1-L$ nothing gets stored (a value of 0). Features with values of 0 can also occur if they are irrelevant to the task.

According to MINERVA2, during retrieval all episodic traces are matched in parallel against a retrieval probe. In so doing, they produce a cumulative output called an *echo*. In the echo, each trace i is weighted by its *activation* $A_{M,i}$. Activation in

MINERVA2 is a function of the *similarity* $S_{M,i}$ of a trace to the retrieval probe.

Similarity increases as the number of matching features increases:

$$S_{M,i} = \sum_{j=1}^{N_i} \frac{P_j T_{i,j}}{N_i}, \quad (10)$$

where P_j is the value of feature j of the probe, $T_{i,j}$ is the value of feature j of trace i and N_i is the number of features relevant for the comparison (i.e., the summation takes place over all non-zero feature values). The resulting quantity lies between -1 and 1, where it equals -1 if the probe and trace are maximally different, and 1 if the probe and trace are the same.

The degree to which a trace is activated is a cubic function of its similarity to the probe:

$$A_{M,i} = S_{M,i}^3 = \left(\frac{2k-1}{N_i} \right)^3, \quad (11)$$

where k is the number of matching features. The activations of all traces in memory are summed to compute the final echo intensity:

$$I = \sum_{i=1}^K A_{M,i}, \quad (12)$$

where K is the number of traces in long-term memory. The consequence of Equation 11 and Equation 12 is that traces more similar to the probe echo more strongly. Because activation is a superlinear (or what Hintzman, 1988, calls “positive accelerated”, p. 529) function of similarity, the more similar traces contribute much more to the echo. In MINERVA2, observables, such as recognition probability or frequency judgments, are linearly related to the echo intensity, and thus, to each trace’s activation.

HyGene’s hypothesis generation and testing process

To generate hypotheses and make probably judgments, HyGene assumes a six-step process. In a first step, data is sampled from the environment, which activates traces in episodic memory as a function of their similarity to the data (see Equation 10). Second,

episodic traces are merged into a single *unspecified probe*, whereby each contributes as a function of its activation (see Equation 11). The unspecified probe represents those hypotheses in episodic memory related to the data. Third, to determine what the unspecified probe represents, it is matched against hypotheses in semantic memory, whereby the similarity between hypotheses and the unspecified probe is computed according to Equation 10. Fourth, hypotheses are generated sequentially. The probability that a hypothesis is sampled is determined by its activation relative to the activation of all hypotheses in semantic memory, following Luce’s choice axiom. If the generated hypothesis is sufficiently active, it is placed in working memory. Working memory stores a limited number of hypotheses. The hypotheses currently entertained in working memory are called the *set of leading contender hypotheses*, or *SOC*. These represents the decision maker’s best explanations of the data. Fifth, the probability of each hypothesis in SOC is estimated by matching this hypothesis to episodic memory and comparing its echo intensity (see Equation 12) to the sum of echo intensities of all hypotheses:

$$P(H_i|D_{obs}) = \frac{I_i}{\sum_{j \in SOC} I_j} \quad (13)$$

Equation 13 leads to the property of *constrained additivity*: The judged probability of the explicitly considered hypotheses is additive. A consequence of constrained additivity is that if the number of number of hypotheses is larger than the explicitly considered hypotheses, judgment will be subadditive. Sixth and last, HyGene models can engage in a hypothesis-guided search in memory. We will not consider this last step in this work.

A comparison of ACT-R’s declarative memory and MINERVA2

MINERVA2 is the foundation for each of HyGene's steps. If we want to implement HyGene in ACT-R, it is necessary to determine if ACT-R can mimic MINERVA2. To this end, we will compare the predictions of the two memory theories as a function of prior history and context. When considering prior history, we will keep context effects (i.e., similarity in MINERVA2, and spreading activation and partial matching in ACT-R) constant. When considering context, we will keep prior history effects (i.e., number of presentations in both theories, learning rate in MINERVA2 and lifetime in ACT-R) constant. For ACT-R, we will always use the exponent of chunk activation, because all observables are related to it, while for MINERVA2, we will use trace activation directly.

Prior history

Presentation frequency

MINERVA2 encodes each observation as a separate trace. When memory is probed, each trace in memory contributes with a certain activation to the echo intensity and content, as given by Equation 11. Thus, if an item occurred n times and its similarity to the probe is fixed to S_M , we will have a total activation of the traces related to that item of $A_M^{tot} = nA_M = nS_M^3$. That is, the relative total activation of all traces related to two items with the same similarity to the probe is proportional to the relative occurrence frequencies of those items:

$$\frac{A_{M,1}^{tot}}{A_{M,2}^{tot}} = \frac{n_1 S_M^3}{n_2 S_M^3} = \frac{n_1}{n_2}. \quad (14)$$

To make the equivalent comparison in ACT-R, we will assume that the chunks representing the two items have the same lifetime T_L . In this case, the relative exponentiated activation of two chunks is:

$$\frac{e^{A_1}}{e^{A_2}} = e^{A_1 - A_2} = e^{B_1 - B_2} = e^{\ln \left[\frac{T_L^{-d(1-d)n_1}}{T_L^{-d(1-d)n_2}} \right]} = e^{\ln \left[\frac{n_1}{n_2} \right]} = \frac{n_1}{n_2}. \quad (15)$$

Thus, MINERVA2 and ACT-R converge in their encoding of frequency. It is important to emphasize that this result holds only for chunks with the same lifetime T_L , as is the case in many experiments and for some real-world events and objects. When the chunks have different lifetimes, however, the predictions of the two theories will diverge.

Constant difference in presentation frequency

While both theories preserve presentation frequency in activation, what happens when we increase the presentation frequency of two items, while keeping the difference in presentation frequency between the two constant? The typical empirical finding is that the chance of committing an error in such a relative frequency judgment task increases with presentation frequency (see Hintzman, 1988).

MINERVA2 explains relative frequency-judgment effects through the learning rate L . Specifically, imperfect encoding (i.e., $L < 1$) leads to a higher variability of echo intensity as the number of presentations of an item increases. This variability is between-subjects: Each subject encodes a different set of features for the same item, which leads to different values of echo intensity among subjects when that item is later presented as a probe. This higher variability then increases the probability of a less frequently presented item to have a higher activation than a more frequently presented item as presentation frequency increases (see Hintzman, 1988).

ACT-R can account for relative frequency judgments, because chunk activation is a random variable. Specifically, assuming that the difference in presentation frequency Δn is kept constant (i.e., $n_1 = n_2 + \Delta n$) and that context effects are constant, the activation difference becomes:

$$\begin{aligned} A_1 - A_2 &= \ln n_1 - \ln n_2 + \varepsilon_1 - \varepsilon_2 = \ln \frac{n_2 + \Delta n}{n_2} + \varepsilon_1 - \varepsilon_2 \\ &= \ln \frac{n_2 + \Delta n}{n_2} + \hat{\varepsilon}. \end{aligned} \tag{16}$$

In the limit, the difference between A_1 and A_2 goes to 0:

$$\lim_{n_2 \rightarrow \infty} A_1 - A_2 = \lim_{n_2 \rightarrow \infty} \left[\ln \frac{n_2 + \Delta n}{n_2} + \hat{\varepsilon} \right] = \ln 1 + \hat{\varepsilon} = 0 + \hat{\varepsilon}. \quad (17)$$

In other words, the two activation distributions overlap more as presentation frequency increases and the difference in presentation frequency is kept constant, because activation noise is independent of presentation frequency and activation is logarithmic with presentation frequency. This results in noise overtaking the effects on any observable as the number of presentations increases:

$$\lim_{n_2 \rightarrow \infty} \frac{e^{A_1}}{e^{A_2}} = e^{A_1 - A_2} = e^{0 + \hat{\varepsilon}} = e^{\hat{\varepsilon}}. \quad (18)$$

To summarize, to account for an increase in error rate in a relative frequency judgment task, MINERVA2 assumes between subject variability due to encoding, while ACT-R assumes that retrieval is an inherently noisy process.

Memory decay

When considering base-level activation in the previous two sections, we assumed that the lifetimes of the two chunks under consideration are the same and that the learning rate stays constant over vectors. Under these assumptions, memory decay does not affect relative activation of chunks. When there is considerable departure from this assumption, one can plug in the actual chunk lifetimes to estimate their relative activation. For example, let us assume that one chunk is x times older than another. That is, we assume that $T_{L1} = xT_{L2}$. Let us also take the default value³ of the decay parameter $d = 0.5$. Given these assumptions, the relative activation of the two chunks will be:

$$e^{B_1 - B_2} = e^{\ln \left[\frac{T_{L1}^{-d(1-d)n_1}}{T_{L2}^{-d(1-d)n_2}} \right]} = e^{\ln \left[\frac{(xT_{L2})^{-0.5(1-d)n_1}}{T_{L2}^{-0.5(1-d)n_2}} \right]} = e^{\ln \left[\frac{n_1}{\sqrt{x}n_2} \right]} = \frac{n_1}{\sqrt{x}n_2}. \quad (19)$$

This means that the relative activation of the younger chunk will be boosted \sqrt{x} times compared to that of the older chunk. For example, if chunk 1 was presented once 9

hours ago, while chunk 2 was presented once 1 hours ago, the latter will have a 3 times larger activation.

This departure from linear encoding of frequency of chunks with different lifetimes is a prediction that MINERVA2 does not incorporate, because MINERVA2 does not directly model memory decay. Instead, memory decay can be engineered by varying the learning rate L (see Hintzman, 1988). Decreasing the learning rate is equivalent to modelling forgetting of feature values, which decreases the similarity between probe and vector. This leads to the vector being less activated than it would be with a higher learning rate. The drawback of this “engineered” solution, compared to ACT-R’s “built-in” memory decay, is that the rate of decay remains unconstrained.

Effects of current context

Context effects in MINERVA2 are modelled through the similarity between a retrieval probe and a memory trace. Specifically, according to MINERVA2, a vector’s activation changes as the cube of the number of matching feature values between probe and the memory vector (see Equation 10 and Equation 11). Figure 3 depicts this cubic relationship between similarity and activation. In practice, we do not expect activation to cover the full range between -1 and 1. This is because we expect two unrelated vectors to differ on half of their non-zero features and to match on the other half, which results in a similarity of 0 (see Hintzman, 1988). Thus, vectors unrelated to the probe contribute around 0 activation to the echo intensity, while related vectors contribute superlinearly to their similarity with the probe. The resulting expected *operative range* of the similarity-activation curve lies within the unshaded region of Figure 3.

[Figure 3 near here]

We will now consider the two mechanisms in ACT-R, through which such context effects can be modelled: partial matching and spreading activation. Partial matching is directly equivalent to MINERVA2's matching of probe and memory vectors, because it assesses the degree of similarity between a retrieval probe and a chunk in declarative memory. Spreading activation, on the other hand, takes place independent of the retrieval request.

Partial matching

The partial matching mechanism allows for retrieving chunks, whose values do not exactly correspond to the retrieval request, but are instead similar to it. According to Equation 1, the contribution of partial matching to activation is a function of the similarity M_{li} of slot l in the retrieval request to the slots of chunk i in memory. One way to translate vector similarities to chunk similarities is to assume that all chunks have only one slot, as is shown on Figure 4. In this case, the entire trace in MINERVA2 is represented by this single slot. The slot value from the retrieval probe is then matched against the single slot values of chunks in declarative memory.

[Figure 4 near here]

To mimic context effects from MINERVA2 onto ACT-R, we need to equalize the contribution of partial matching in a chunk's (exponentiated) activation to the contribution of similarity in a trace. Assuming equal prior histories, this would amount to the following condition:

$$e^{PM_i} = S_{M,i}^3. \tag{20}$$

Note that in Equation 20, we have dropped the index l , because our chunks have only a single slot. The resulting value of a chunk's similarity to the retrieval request, given $S_{M,i}$ and a constant mismatch penalty P , equals:

$$M_i = \frac{3 \ln S_{M,i}}{P} = \frac{3}{P} \ln \left(\frac{2k-1}{N_i} \right). \quad (21)$$

A second way to translate vector similarities to chunk similarities relies on the assumption that there are as many slots in chunks as non-zero features in vectors, as is depicted in Figure 5. This amounts to the following condition:

$$e^{P \sum_{l=1}^{N_i} M_{li}} = S_{M,i}^3, \quad (22)$$

where the summation is over all non-zero features N_i . Given the assumptions that the similarity between slots with the same value (i.e., 1 & 1 or -1 & -1) is constant and equal to M_i , and that the similarity between slots with differing values is 0, the similarity equals:

$$M_i = \frac{3 \ln S_{M,i}}{Pk} = \frac{3}{Pk} \ln \left(\frac{2k-1}{N_i} \right). \quad (23)$$

[Figure 5 near here]

A third and final way to translate vector similarities to chunk similarities assumes that a chunk has several slots and each slot is mapped to an equal part of a trace. For example, Figure 6 depicts a case, in which a 12-feature vector is mapped onto a chunk with 3 slots, whereby each slot accommodates 4 features. This way of translating vector similarity to chunk similarity is a generalization of the previous two cases. Under this assumption, the relationship between the two quantities is:

$$e^{P \sum_{l=1}^m M_{li}} = S_{M,i}^3, \quad (24)$$

where m is the number of slots in a chunk.

[Figure 6 near here]

To determine the similarities M_i we can again assume that each matching feature value pair (e.g., 1 & 1 or -1 & -1) contributes the same amount M_i to similarity. M_i is again given by Equation 23. Computing M_i amounts to multiplying M_i by the number of matching feature values for slot l . Note that Equation 21 and Equation 23 are only defined when at least half of the feature values match.

Spreading activation

The second mechanism to model context effects in ACT-R is spreading activation. A chunk in the imaginal buffer spreads activation to chunks in memory if the latter have slots in common with the former. Irrespective of their similarity, slots of chunks in declarative memory, whose values are not present in the imaginal buffer, do not receive any spreading activation. Because of this binary nature of spreading activation, we will assume that each feature in MINERVA2 corresponds to a slot in a chunk, as in Figure 5. For simplicity, we will assume that $W_j = 1$ and that $P(i/j) = p$ for all i and j , between which activation is spread. In this case, the spreading activation equation reduces to:

$$SA_i = \sum_j W_j \{S + \ln[P(i/j)]\} = \sum_j \{S + \ln[p]\}. \quad (25)$$

To mimic context effects, we need to set the contribution of spreading activation in a chunk's (exponentiated) activation equal to the contribution of similarity in a trace. The exponent of spreading activation is a power function of the number of matching slots k :

$$e^{SA_i} = (pe^S)^k. \quad (26)$$

Both ACT-R's (Equation 26) and MINERVA2's (Equation 11) expressions are superlinear functions of the number of matching slots k , albeit with different functional forms. To demonstrate the close match between the two, Figure 7 plots ACT-R's

functional form for a base of 1.5 (i.e., $pe^S=1.5$) next to the operative range of MINERVA2's similarity/activation relationship.

[Figure 7 near here]

Equating exponentiated spreading activation to a trace's activation is equivalent to the following condition:

$$e^{SA_i} = (pe^S)^k = S_{M,i}^3 = \left(\frac{2k-1}{N_i}\right)^3. \quad (27)$$

This equation can be solved given a number of matching feature values k by fixing either p or S . Alternatively, one can fit a power function, which mimics a cubic function, to arrive at a solution independent of the value of k (e.g., as in Figure 7).

A 3-step guide for translating episodic traces to chunks

The results above demonstrate that the predictions made by ACT-R's declarative memory and those made by MINERVA2 strongly correspond and that ACT-R can mimic the predictions by MINERVA2. Based on these results, we will provide a 3-step guide of how to translate MINERVA2's vectors to ACT-R's chunks. In step 3, one can model context effects through either partial matching or spreading activation.

Step 1: Identify unique episodic traces. Create separate chunks for each unique episodic trace.

Step 2: Count the number of instances of each unique episodic trace and set base-level activation of the corresponding chunk to $B=c+\ln(n)$, where c is any constant.

Step 3a: Compute the similarity between slots by counting the number of features upon which probe and traces disagree and then applying Equation 21 or Equation 23, depending on the representation chosen.

Step 3b: Assume that each feature in MINERVA2 is represented by a separate chunk slot and determine the values of p and S in Equation 27 through fitting.

Translating HyGene's processes

In what follows, we will briefly outline how HyGene's processing steps can be realized in ACT-R. The description is enabled by the demonstrated correspondence between the two memory theories. We will trace the processing steps in the same order as presented in HyGene's introduction.

Sampling data from the environment

As already mentioned, ACT-R has modules modelling perception. It is thus able to model data acquisition dynamics in various experimental conditions. Generally, data acquisition will consist of sequentially encoding observations through one of the perceptual modules. Each observation will be stored in the imaginal buffer after it is encoded. While the data acquisition details will vary from task to task, the final stage of this process will result in a chunk in the imaginal buffer that contains the acquired data.

Extraction of an unspecified probe

Once the data is placed in the imaginal buffer, an unspecified probe is constructed via blending. In blending, each chunk contributes as a function of its activation. We have demonstrated how activation in MINERVA2 can be mapped onto activation in ACT-R. If one follows our guideline, the result of blending will be equal to the echo. The unspecified probe, now a blended chunk, will then be placed in the imaginal buffer.

Matching the unspecified probe against hypotheses

Once the unspecified probe chunk is in the imaginal buffer, it will activate hypothesis chunks in declarative memory. We have demonstrated that vector-vector similarity can be modelled with either partial matching (Equation 23) or spreading activation

(Equation 27). Whichever mechanism is chosen, the activation of hypotheses in memory will be equivalent to that by HyGene.

Hypotheses generation

In HyGene, the probability of selecting a hypothesis is determined via Luce's choice axiom, that is, it is directly proportional to that hypothesis's activation. In ACT-R, if hypotheses are sufficiently above threshold, which hypothesis is retrieved is determined by Boltzmann selection rule (Equation 6), which is proportional to the exponent of hypotheses' activations. The Boltzmann selection rule differs from Luce's choice axiom in that the quantities input are exponentiated. Because ACT-R's exponentiated activation plays the same role as MINERVA2's activation, hypothesis retrieval will be equivalent for the two theories.

Once a hypothesis is retrieved, it must be decided if it will be stored in memory or not. In HyGene, whether a hypothesis is stored in working memory or not is determined by its activation relative to the activations of hypotheses currently in working memory. ACT-R does not allow the cognitive agent direct access to chunk activation. However, through its timing module, it can track the subjective time through the number of ticks accumulated during the retrieval of each hypothesis: Hypotheses with higher activations will be retrieved faster than hypotheses with lower activations. The number of ticks can then be used to determine whether a hypothesis is stored in the SOC or not.

The SOC itself is modelled as a chunk in the imaginal buffer, whose slots store the currently entertained hypotheses. Populating the SOC will entail three steps. In step 1, the unspecified probe is retrieved. In step 2, a hypothesis is retrieved, as determined by Equation 6, and its retrieval time tracked with the timing module. This hypothesis is then stored in the imaginal buffer together with its subjective retrieval time. In step 3,

the SOC chunk is retrieved and the hypothesis in the imaginal buffer can be added to the SOC chunk. The model will cycle between these three steps until the SOC is populated.

Hypotheses evaluation

Once generation has ceased, the SOC chunk is in the imaginal buffer. Each hypothesis in SOC is evaluated via blending. The exponents of the match score of the blended chunk for each hypothesis, following Equation 9, equals the sum of exponents of all chunk activations, making it equivalent to the echo intensity. If one were to access the match score related to each hypothesis, one would follow Equation 13 to judge the probability of each hypothesis and the result will be the same as in HyGene. Moreover, the judged probabilities would be also additive for the explicitly considered hypotheses and subadditive when the number of hypothesis exceeds working memory capacity.

Discussion

We have outlined how HyGene can be implemented in ACT-R. We started by investigating the compatibility of the memory theory behind ACT-R and MINERVA2, the episodic memory theory behind HyGene. At first sight, the two memory theories seem different: ACT-R uses the notion of a memory chunk with slots, while MINERVA2 uses vectors with binary feature values; Chunks use a single internal representation of an external event/object, whose memory strength increases upon repeated encounters or recalls, while vectors are stored anew upon each encounter. Moreover, ACT-R explicitly models memory decay, while MINERVA2 introduces imperfections in memory records through a noisy encoding process. Finally, context effects in MINERVA2 are a function of the number of matching features between probe and memory trace, while ACT-R assumes two possible mechanisms. Given these

differences, it is not obvious at which points these two theories will be tangential to each other.

Our results show that, both as a function of prior history and current context, the theories can make similar predictions. We have specified the conditions under which the predictions of these two theories correspond and when they differ. We have also provided a 3-step guide of how to translate MINERVA2's feature lists to ACT-R's chunks so that they make identical predictions.

Finally, we outlined how each of HyGene's steps can be modelled with ACT-R. Given that each step relies on the equations behind MINERVA2, it was essential that we outlined how to map MINERVA2's vectors to ACT-R's chunks. We will now discuss four points of departure of the two theories. First, we will discuss how ACT-R could access subsymbolic values, necessary in the hypothesis evaluation process. Second, we will opine on where a feature vector lies according to ACT-R's terminology. Third, we will discuss the address in functional form behind the equations of the two theories. Finally, we will comment on the necessity of separate semantic and episodic memory storages.

Access to subsymbolic values

In its current formulation, ACT-R does not allow symbolic processes to access subsymbolic values. Therefore, even though the total amount of activation of chunks in declarative memory (i.e., the match score from the blending process) is proportional to the amount of echo, there is no direct way for the model to access these values. Access to the match score of the blended chunk is necessary for the probability judgment of each hypothesis. Therefore, for ACT-R to judge the probability of each hypothesis, it should include a mechanism to either directly or indirectly access these subsymbolic values. A direct access to subsymbolic values would require modifying ACT-R to allow

access to memory activations. Note that this path has been taken by the *Source of Activation Confusion* model of memory (e.g., Diana, Reder, Arndt, & Park, 2006), a model strongly related to ACT-R's declarative memory. An indirect access to subsymbolic values might rely on a prestored mapping of ticks, from the timing module, to probability estimates.

Representation

According to ACT-R, chunks and their slots lie at the symbolic level while the equations which determine chunk dynamics belong to the subsymbolic level. On the other hand, in MINERVA2 the feature values are directly used to compute activation and therefore to determine retrieval dynamics. This means that feature values serve a role similar to chunk activation and suggests that, in ACT-R's terminology, MINERVA2's vectors occupy the subsymbolic level. Consequently, the various possible mappings of memory traces to chunks, such as those depicted in Figures 4, 5 and 6, are alternative subsymbolic representational implementing to ACT-R's subsymbolic equations.

Functional forms

As a function of prior history, MINERVA2's mechanism is more flexible than that of ACT-R due to the unconstrained nature of MINERVA2's learning rate parameter. In essence, one can choose any value of L to model forgetting through time, which would result in an arbitrary forgetting functional form and rate. Meanwhile, ACT-R's memory decay rate has been experimentally constrained (Anderson, Fincham, & Douglass, 1999).

One the other hand, when considering context effects, ACT-R's functional form is more flexible for two reasons. First, similarity M_{ij} is generally unconstrained (although, in practice, meaningful values are chosen). Second spreading activation's

functional form, a power function, is more flexible than the cubic function in MINERVA2. However, the cubic functional form, while more constrained, is not strongly justified. Instead, its purpose is to allow “those items in memory that are most similar to the test probe to dominate the overall echo from secondary memory, while preserving the sign” (Dougherty et al., 1999, p. 183). Thus, any function which increases sufficiently steeply, such as a power function, will provide the required effect.

Separate long-term memory storages

HyGene assumes three memory storages: working memory, episodic memory and semantic memory. While ACT-R has a storage for task-relevant information analogical to working memory (i.e., the imaginal module), it only has a single long-term storage, modelled with the declarative module, as opposed to separate episodic and semantic storages. Does ACT-R require separate semantic and episodic storages?

Already since its inception by Tulving, the idea that there are two memory compartments has seen opposition (e.g., Anderson & Ross, 1980). In an extensive critique, McKoon, Ratcliff, and Dell (1986) offered alternative interpretations of many of the pieces of evidence brought forth in support of this dichotomy and, moreover, argued that it is impossible to address questions about the existence of two memory storages, because the difference between the two is not rendered as a specific theory. Moreover, recent memory theories, such as event memory (Rubin & Umanath, 2015), assume or argue in favour of a single memory storage. Anderson (2007) has also argued that episodic memories only differ from semantic memories “because we can retrieve the specific context in which they were learned” and that they “both belong in declarative memory” (p. 93). These indicate that a single declarative memory might suffice to model the various processes required.

Conclusion

“[B]rand names” tend to make difficult the analysis and comparison of . . . mechanisms or the exchange of knowledge between research groups. One can argue that it has caused and causes an enormous amount of duplication of effort.

Simon, 1998, as cited in Anderson, 2007, p.239

Simon (as cited in Anderson, 2007) criticized the branding of models, because it made their mechanism opaque to outsiders, which leads to “an enormous amount of duplication of effort” (p.239). Mischel (2008), too, famously paralleled theories in psychology to toothbrushes, in that “no self-respecting person wants to use anyone else’s”. Critiques of the fragmentedness and inefficiencies of psychological theorizing abound, but we think they only tell one side of the story. On the other side lie truly cumulative efforts, among which are the two theories that we compared. It is their precise formalism that allowed us to compare the predictions of the two theories formally and to find points of agreement between the two. With our work, we aimed to make our modest contribution to extending the areas of applicability of ACT-R and increasing the amount of detail in HyGene’s predictions. More generally, we would like to foster the endeavour of knowledge cumulation and theory integration.

Acknowledgements

The author would like to thank Julian Marewski for the provided encouragement.

Footnotes

1. Dimov (2016) demonstrated that the optimized learning equation is an unbiased approximation of the base-level learning equation even when there is some noise in this periodicity. Specifically, the noise is allowed to increase superlinearly as a function of the number of periods since the event or object encounter.
2. MINERVA2's typically uses S as a notation for similarity. Since this overlaps with ACT-R's associative strength, here we will use S_M for similarity in MINERVA2. Similarly, we will use A_M to indicate activation of a trace and keep A as an activation of a chunk in ACT-R.
3. In ACT-R there are default values for many of the parameters, departure from which is discouraged unless justified.

References

- Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* Oxford University Press.
- Anderson, J. R., & Douglass, S. (2001). Tower of Hanoi: evidence for the cost of goal retrieval. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 27, 1331-1346.
- Anderson, J. R., Fincham, J. M., & Douglass, S. (1999). Practice and retention: A unifying analysis. *Journal of Experimental Psychology: Learning, Memory and Cognition*, 25, 1120-1136.
- Anderson, J. R., Fincham, J. M., Schneider, D. W., & Yang, J. (2012). Using brain imaging to track problem solving in a complex state space. *NeuroImage*, 60, 633-643.
- Anderson, J. R., & Ross, B. H. (1980). Evidence against a semantic-episodic distinction. *Journal of Experimental Psychology: Human Learning and Memory*, 6, 441-466.
- Anderson, J. R., Qin, Y., Sohn, M. H., Stenger, V. A., & Carter, C. S. (2003). An information-processing model of the BOLD response in symbol manipulation tasks. *Psychonomic Bulletin & Review*, 10, 241-261.
- Borst, J. P., & Anderson, J. R. (2015a). Using the ACT-R Cognitive Architecture in combination with fMRI data. In *An introduction to model-based cognitive neuroscience* (pp. 339-352). Springer New York.
- Borst, J. P., & Anderson, J. R. (2015b). The discovery of processing stages: Analyzing EEG data with hidden semi-Markov models. *NeuroImage*, 108, 60-73.

- Borst, J. P., Ghuman, A. S., & Anderson, J. R. (2016). Tracking cognitive processing stages with MEG: A spatio-temporal model of associative recognition in the brain. *NeuroImage*, *141*, 416-430.
- Davelaar, E. J., Goshen-Gottstein, Y., Ashkenazi, A., Haarmann, H. J., & Usher, M. (2005). The demise of short-term memory revisited: empirical and computational investigations of recency effects. *Psychological Review*, *112*, 3-42.
- Diana, R.A., Reder, L. M., Arndt, J., & Park, H. (2006). Models of recognition: A review of arguments in favor of dual-process account. *Psychonomic Bulletin & Review*, *13*, 1-21.
- Dimov, C. M. (2016). Connections between ACT-R's declarative memory system and Minerva2. In A. Papafragou, D. Grodner, D. Mirman, & J. C. Trueswell (Eds.) *Proceedings of the 38th annual conference of the Cognitive Science Society* (pp. 492–495). Austin, TX: Cognitive Science Society.
- Dimov, C. M., & Link, D. (2017). Do People Order Cues by Retrieval Fluency when Making Probabilistic Inferences?. *Journal of Behavioral Decision Making*.
- Dimov, C. M., Marewski, J. N., & Schooler, L. J. (2013). Constraining ACT-R models of decision strategies: An experimental paradigm. In M. Knauff, M. Pauen, N. Sebanz, & I. Wachsmuth (Eds.), *Proceedings of the 35th annual conference of the Cognitive Science Society* (pp. 2201–2206). Austin, TX: Cognitive Science Society.
- Dougherty, M. R., Gettys, C. F., & Ogden, E. E. (1999). MINERVA-DM: A memory processes model for judgments of likelihood. *Psychological Review*, *106*, 180-209.

- Einhorn, H. J., & Hogarth, R. M. (1978). Confidence in judgment: Persistence of the illusion of validity. *Psychological Review*, *85*, 395-416.
- Fechner, H. B., Pachur, T., Schooler, L. J., Mehlhorn, K., Battal, C., Volz, K. G., & Borst, J. P. (2016). Strategies for memory-based decision making: Modeling behavioral and neural signatures within a cognitive architecture. *Cognition*, *157*, 77-99.
- Fiedler, K. (1996). Explaining and simulating judgment biases as an aggregation phenomenon in probabilistic, multiple-cue environments. *Psychological Review*, *103*, 193-214.
- Fiedler, K. (2000). Beware of samples! A cognitive-ecological sampling approach to judgment biases. *Psychological Review*, *107*, 659-676.
- Gigerenzer, G., Hoffrage, U., & Kleinbölting, H. (1991). Probabilistic mental models: A Brunswikian theory of confidence. *Psychological Review*, *98*, 506-528.
- Gonzalez, C., Lerch, J. F., & Lebiere, C. (2003). Instance-based learning in dynamic decision making. *Cognitive Science*, *27*, 591-635.
- Hintzman, D. L. (1984). Minerva 2: A simulation model of human memory. *Behavior Research Methods, Instruments, & Computers*, *16*, 96-101.
- Hintzman, D. L. (1988). Judgments of frequency and recognition memory in a multiple-trace memory model. *Psychological Review*, *95*, 528-551.
- Lange, N. D., Thomas, R. P., & Davelaar, E. J. (2012). Temporal dynamics of hypothesis generation: the influences of data serial order, data consistency, and elicitation timing. *Frontiers in Psychology*, *3*, 1-16.
- Lange, N. D., Davelaar, E. J., & Thomas, R. P. (2013). Data acquisition dynamics and hypothesis generation. *Cognitive Systems Research*, *24*, 9-17.

- Marewski, J. N., & Mehlhorn, K. (2011). Using the ACT-R architecture to specify 39 quantitative process models of decision making. *Judgment and Decision Making, 6*, 439–519.
- Marewski, J. N., & Schooler, L. J. (2011). Cognitive niches: an ecological model of strategy selection. *Psychological Review, 118*, 393-437.
- McKoon, G., Ratcliff, R. & Dell, G. S. (1986). A critical evaluation of the semantic-episodic distinction. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 12*, 295-306.
- Mehlhorn, K., Taatgen, N. A., Lebiere, C., & Krems, J. F. (2011). Memory activation and the availability of explanations in sequential diagnostic reasoning. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 37*, 1391-1411.
- Mischel, W. (2008). The Toothbrush Problem. Observer. Retrieved March 06, 2017, from <http://www.psychologicalscience.org/observer/the-toothbrush-problem>.
- Nellen, S. (2003). The use of the “take-the-best” heuristic under different conditions, modeled with ACT-R. In F. Detje, D. Dörner, & H. Schaub (Eds.), *Proceedings of the Fifth International Conference on Cognitive Modeling* (pp. 171–176). Germany: Universitätsverlag Bamberg.
- Paik, J., Kim, J. W., Ritter, F. E., & Reitter, D. (2015). Predicting User Performance and Learning in Human--Computer Interaction with the Herbal Compiler. *ACM Transactions on Computer-Human Interaction (TOCHI), 22*, 25.
- Rubin, D. C., & Umanath, S. (2015). Event memory: A theory of memory for laboratory, autobiographical, and fictional events. *Psychological Review, 122*, 1-23.
- Salvucci, D. D., & Anderson, J. R. (2001). Integrating analogical mapping and general problem solving: The path-mapping theory. *Cognitive Science, 25*, 67–110.

- Schooler, L. J., & Hertwig, R. (2005). How forgetting aids heuristic inference. *Psychological Review*, *112*, 610-628.
- Schneider, D. W., & Anderson, J. R. (2012). Modeling fan effects on the time course of associative recognition. *Cognitive Psychology*, *64*, 127-160.
- Taatgen, N. A., & Anderson, J. R. (2002). Why do children learn to say broke? a model of learning the past tense without feedback. *Cognition*, *86*, 123–155.
- Thomas, R. P., Dougherty, M. R., Sprenger, A. M., & Harbison, J. (2008). Diagnostic hypothesis generation and human judgment. *Psychological Review*, *115*, 155-185.
- Tversky, A., & Kahneman, D. (1973). Availability: A heuristic for judging frequency and probability. *Cognitive Psychology*, *5*, 207-232.