Evolving Cognitive Models: A Novel Approach to Verbal Learning

Noman Javed[1]
Centre for Philosophy of Natural and Social Science
London School of Economics
London, UK
N.Javed3@lse.ac.uk

Dmitry Bennett[1]
Centre for Philosophy of Natural and Social Science
London School of Economics
London, UK
D.Bennett5@lse.ac.uk

Laura K. Bartlett
Centre for Philosophy of Natural and Social Science
London School of Economics
London, UK
L.Bartlett@lse.ac.uk

Peter C. R. Lane
School of Physics, Engineering and Computer Science
University of Hertfordshire
Hatfield, UK
P.C.Lane@herts.ac.uk

Fernand Gobet
Centre for Philosophy of Natural and Social Science
London School of Economics
London, UK
F.Gobet@lse.ac.uk

***Abstract*—A common goal in cognitive science involves explaining/predicting human performance in experimental settings. This study proposes a single GEMS computational scientific discovery framework that automatically generates multiple models for verbal learning simulations. GEMS achieves this by combining simple and complex cognitive mechanisms with genetic programming. This approach evolves populations of interpretable cognitive agents, with each agent learning by chunking and incorporating long-term memory (LTM) and short-term memory (STM) stores, as well as attention and perceptual mechanisms. The models simulate two different verbal learning tasks: the first investigates the effect of prior knowledge on the learning rate of stimulus-response (S-R) pairs and the second examines how backward recall is affected by the similarity of the stimuli. The models produced by GEMS are compared to both human data and EPAM – a different verbal learning model that utilises hand-crafted task-specific strategies. The models automatically evolved by GEMS produced good fit to the human data in both studies, improving on EPAM's measures of fit by almost a factor of three on some of the pattern recall conditions. These findings offer further support to the mechanisms proposed by chunking theory (Simon, 1974), connect them to the evolutionary approach, and make further inroads towards a Unified Theory of Cognition (Newell, 1990).**

***Keywords*—chunking, evolution, GEMS, LTM, STM, CHREST**

I. Introduction

As computational and Artificially Intelligent tools impact creative fields in increasingly disruptive ways, it is important to consider the opportunities and likely impact of such tools in the creative field of scientific discovery. In this paper, we present a tool, GEMS, which supports the creative problem solving, theory building and desire for understanding of a human scientist with the speed, precision and various strengths of AI. Our approach asks the human scientist to define basic cognitive operators, which create a space of candidate models, and specify the experiments and related criteria by which models will be evaluated: in this paper, we define a space of chunking-based computational models and seek good models of a verbal learning task. The GEMS system then generated a variety of candidate models, all with a high level of fit to the target criteria set by the human scientist: typically, the range and variety of generated models far exceed that which could be developed by hand. The generated models can then be analysed and interpreted by the human scientist to uncover overall patterns and develop innovative theoretical insights.

The development of cognitive models may be divided into two stages. The first stage involves building the basic structures and mechanisms that represent the workings of a real cognitive system – such as long-term memory (LTM), short-term memory (STM), attention, and chunking. The second stage involves tuning the models to simulate a specific behaviour, for example, by specifying attentional and learning strategies.

Both stages require a number of assumptions with regard to the cognitive/psychological mechanisms that are involved in various experimental settings. This often leads to experimenters' bias, with researchers inadvertently overlooking potentially influential mechanisms at play. Another problem is that models are typically designed to fit a specific behavioural dataset, with each subsequent additional task requiring hand-crafted changes to the model.

In this paper, we demonstrate how evolutionary programming can be combined with a complex cognitive architecture to automatically generate candidate computational models in a typical cognitive psychology experiment – these models are represented as computer programs in GEMS. We demonstrate how such models can improve on hand-crafted alternatives, all while addressing the crucial issues of constrained variability and individual differences in psychology. We use Genetic Programming (GP) (Koza, 1992) to search a space of candidate models. The fitness function guiding this search is designed to find models which simulate the behaviour of human participants. Unlike many studies that involve the GP approach, this fitness function evaluates models' performance on a simulated psychology experiment, instead of a typical pure input-output mapping for the program. As human participants do not achieve 100% success in our example tasks, the "best" models are those that replicate this less-than-perfect accuracy. Also, as human responses take a certain amount of time, a simulated

[1] Equally contributing authors.

time for the program to convert each input into an output must be measured and compared with the observed response time. We combine these two performance aspects in our models. With all that said, GEMS is still a machine learning system – albeit one that learns to produce models of cognition using evolution.

Another important contribution of the current study is the integration of a complex cognitive-based psychological architecture and a genetic programming system into a single whole. Typically, models based on GP and genetic algorithms produce long and complex programs/strategy sets, but their inherent cognitive structures are often rudimentary and lack simulations of the LTM and perceptual apparatus (for example, see Bartlett et al., 2023; Gunaratne & Patton, 2022). Here, we combine GP, as instantiated in GEMS, with the CHREST cognitive architecture; therefore, we keep the inherent complexity of the psychological structures and mechanisms characterising verbal learning, while adding the automaticity of task-specific strategy discovery.

In sum, the contribution of the present paper is threefold: first, we propose a single modelling framework that minimises experimenter's bias (inherent in hand-crafting task-specific learning strategies) by automatically generating cognitive models; second, the generated models will account for humans' suboptimal learning routines and response times; third, the generated models will perform tasks that require complex cognitive structures and mechanisms – including STM, LTM, chunking, and attention.

GP often generates a large number of candidate models: we will select a small set of the best fitting, understandable and qualitatively different models – thus also presenting the proof-of-concept solution to the issue of individual differences in the strategies used to carry out the task.

II. GEMS

Cognitive modelling involves the development of computational models to explain a target human behaviour. These computational models are typically developed within established frameworks, such as the symbolic approach outlined by Simon (1981) or the connectionist framework advocated by Rumelhart & McClelland (1986). GEMS aligns with the symbolic, information-processing approach, in which each individual cognitive model is defined by a control program interpreted within a simplified cognitive architecture. The basic architecture of GEMS, which resembles a program synthesis system, is shown in Figure 1.

At the heart of GEMS lies Genetic Programming (GP) (Koza, 1992). Within our meta-modelling system, GP relies on a set of operators serving as fundamental building blocks for constructing cognitive models. (For example, one operator might place an item into STM, while another could match two items within STM.) These operators, akin to basic functions or operations implemented as programming functions, play a pivotal role in accurately simulating cognitive processes across the components of the cognitive architecture. Domain experts meticulously select these operators, assigning relevant semantics and timings based on pertinent research literature. The choice of operators is also contingent on the specific experiment being simulated; for instance, for a short-term memory task, long-term memory operators may not be necessary. The process of natural selection may further remove unnecessary operators during evolution, contradicting the experimenter's initial assumptions (Bartlett et al., 2023; Frias-Martinez & Gobet, 2007).

GP generates numerous programs by combining these operators. The combination of these operators potentially results in a vast search space for programs, which expands exponentially as the number of operators increases. Harnessing the capabilities of GP as a potent parallel search algorithm, GEMS initiates multiple searches to discover optimal models closely resembling human behaviour. Each cognitive model is interpreted within a simplified cognitive architecture, incorporating task-specific input/output components, a short-term memory, and a long-term memory. A clock monitors each model's time during tasks. While executing the program and interpreting the operators, actions are performed using the simplified cognitive architecture. Together, this architecture and the protocol of the experiment are termed the model evaluation environment.

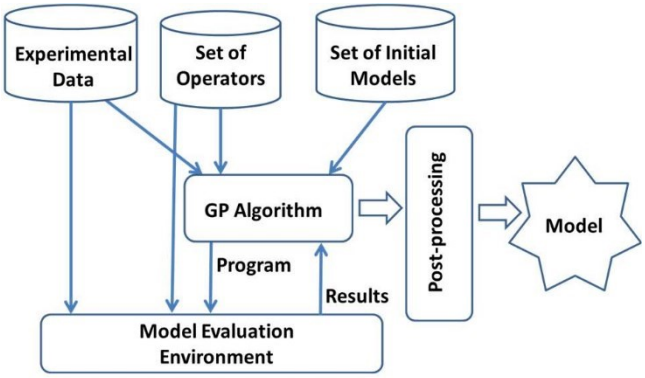

Figure 1. A high-level overview of GEMS. GEMS takes as inputs initial models (programs), cognitive operators and experimental data and then proceeds to evolve populations of models.

GP is an evolutionary search technique, starting with a population of candidate solutions that evolve gradually through multiple iterations until a termination condition, such as a predefined number of iterations, is met. This evolution process is inspired by biological evolution, with candidate solutions selected based on their fitness. New candidates emerge from existing ones through crossover and mutation processes, respectively combining and modifying solutions.

Since GP operates as a fitness-guided search algorithm, GEMS must utilise a fitness evaluation mechanism for the models – this is given by the human cognitive scientist running the simulation. Unlike many other machine learning algorithms, GEMS employs a simulation of a scientific experiment to assess program fitness. While typical machine learning algorithms evaluate program performance based on an input-output mapping, GEMS acknowledges that human subjects do not always achieve perfect success. Therefore, the "best" model for GEMS is one that reflects this imperfect accuracy. Additionally, as human responses take time, the program's simulated time for processing inputs and generating outputs must be measured and compared to the actual response time. These two factors are essential when evaluating a model's performance.

At the end of the GP process, GEMS will have produced a large population of candidate solutions. However, the multitude of evolved candidates and the presence of dead code

(functionally similar segments with different content) and genuine variations obscure the range and interpretability of interesting solutions. To enhance the comprehensibility of candidate programs, post-processing steps are implemented to distil fewer, higher-quality solutions (Lane et al., 2022). These steps involve removing dead code and time-only code, as well as clustering models based on syntactic similarity. This process reduces the number of solutions and categorizes them into distinct groups, thereby improving understanding.

We will now describe the cognitive architecture CHREST, which will provide cognitive operators accessing long-term memory either for learning or for recognising patterns.

III. CHREST

CHREST (Chunking Hierarchy and REtrieval STructures) (Gobet, 1993, 2000; Gobet & Lane, 2012; Gobet & Simon, 2000) is a computational cognitive model that simulates human learning and is based on the chunking theory, a well-established theory in cognitive psychology (Chase & Simon, 1973; Gobet et al., 2001; Simon, 1974).

The key concept of the chunking theory, a "chunk", can be defined as a meaningful unit of information constructed from elements that have strong associations between each other, such as a cluster of digits making up a phone number. Thus, "chunking" involves forming and updating these meaningful units in the LTM (Gobet, Lloyd-Kelly, & Lane, 2016; Simon, 1974). Although the specific content of chunks may vary between individuals, chunking mechanisms are largely consistent across domains, individuals, and cultures (Chase & Simon, 1973; Gobet et al., 2001; Miller, 1956; Simon, 1974).

CHREST operationalized chunks as nodes in a graph data structure that represents LTM, with chunking being the process of adding new data to LTM (Gobet, 1993, 2000; Gobet & Lane, 2012; Gobet & Simon, 2000). Chunking is performed via two psychologically plausible cognitive processes: discrimination and familiarisation. Discrimination is the process of adding a new node to the LTM network. Familiarisation updates existing nodes with new information. Thus, learning is influenced both by the environmental stimuli and the data that have already been stored (Gobet & Lane, 2012). CHREST's other major cognitive structure – the pool of STM stores – allows for additional ways to create links between chunks, such as linking chunks across visual, verbal and action modalities, or linking stimuli and responses within a single modality.

Researchers using chunking theory have empirically established time durations for its core cognitive operations, with discrimination taking around ten seconds, familiarisation taking two seconds, and recognition of a pattern requiring approximately one hundred milliseconds. These time costs are based on empirical data (Card, Moran, & Newell, 1983; Feigenbaum & Simon, 1984) and were mostly tested in the simulation of chess experiments (De Groot & Gobet, 1996; Gobet & Simon, 2000).

EPAM, CHREST and related models have been applied to predict and simulate behaviour in verbal learning research (Feigenbaum, 1959; Feigenbaum & Simon, 1984; Richman & Simon, 1989; Richman, Simon, & Feigenbaum, 2002). They have also been used to shed light on perception and the fundamental mechanisms of concept formation (Bennett, Gobet, & Lane, 2020; Lane & Gobet, 2012), problem solving (Lane, Cheng, & Gobet, 2000), acquisition of language and syntactic categories (Freudenthal et al., 2016), emotion processing in problem gambling (Schiller & Gobet, 2014), developmental trends and cognitive decline due to ageing (Mathy et al., 2016; Smith, Gobet, & Lane, 2007), expert behaviour (Gobet & Simon, 2000; Richman et al., 1996; Richman, Staszewski, & Simon, 1995; Simon & Gilmartin, 1973), and various other phenomena (see Figure 2).

For further details of the chunking theory and CHREST, refer to Gobet and Lane (2012). The interaction of GEMS and CHREST is presented in Figure 3.

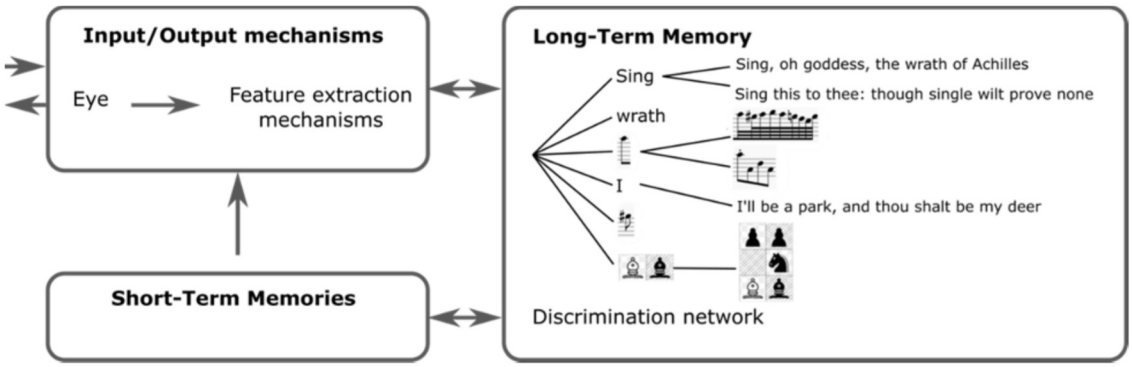

Figure 2. A high-level overview of CHREST. Its LTM is capable of chunking complex stimuli from multiple dissimilar domains.

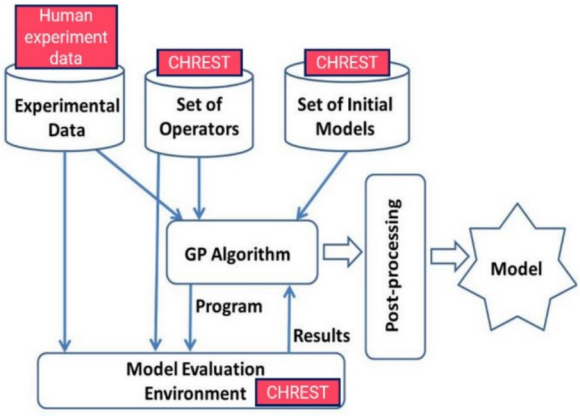

Figure 3. A high-level overview of GEMS, with indication of where it interfaces with CHREST.

IV. Verbal Learning

As mentioned above, chunking plays a crucial role in a wide range of cognitive phenomena. In the current paper, we focus on chunking in verbal learning, as research in this field has been fundamental to the development of cognitive architectures. Also, verbal learning tasks are sufficiently taxing to require complex cognitive modelling, and yet simple enough to be computationally efficient in multiple populations of hundreds of evolving cognitive agents. For example, the current tasks are impossible for the previous models of GEMS and their rudimentary LTM, STM, perceptual and attentional mechanisms.

Verbal learning (VL) tasks typically entail teaching lists of paired stimulus-response (S-R) nonsense syllables to human participants, with the aim to uncover the fundamental laws of learning. Notable examples of verbal learning research include the "magic number 7 (plus or minus two)" study that established the STM capacity to be around seven chunks (Miller, 1956), and the study of the primacy-recency effect (where people were found to make fewest mistakes at the beginning and at the end of a memorised sequence) (McCrary & Hunter, 1953). Another significant yet less known contribution of verbal learning research lies in its role in shaping and refining mechanisms of cognitive models such as EPAM and CHREST. Indeed, Richman et al. (2002) reported that various versions of EPAM had captured at least 20 regularities that were identified by research into human rote learning. For example, EPAM was able to reproduce the primacy-recency serial position curve that describes people's tendency to remember first and last items better than middle-of-the-list ones (Feigenbaum, 1963; McCrary & Hunter, 1953). Similarly, single-shot learning (Rock, 1957) was explained and simulated by EPAM (Gregg & Simon, 1967) – it was dependant on the complexity/simplicity of the stimuli (as determined by a chunks composition and availability in the LTM) and upon the attentional strategy of the participant. Other examples of verbal learning simulations include the similarity interference effects (Hintzman, 1968, 1969) and forgetting. An example of the former includes the finding that humans (and EPAM) produced more errors in S-R learning trials when the stimuli – nonsense consonant trigrams – were similar to each other (e.g. "ZIK" and "ZYJ") than when they were dissimilar. Examples of the latter are the established "know-forget-know" cycle, which occurs during the learning of a single list, and "retroactive inhibition", which happens when learning a second list disrupts memory of a first list (Feigenbaum & Simon, 1962; Thune & Underwood, 1943).

One surprising result from verbal learning experiments was that prelearning of stimuli was not as helpful to the human learner as prelearning of responses (Chenzoff, 1962; Richman et al., 2002). The EPAM verbal learning model suggests that this is because responses need to be learnt completely, while the stimuli can be learnt partially as long as the learnt representations make it possible to discriminate a given stimulus from other stimuli (e.g., learning only the first letter of the nonsense trigram "CET" is sufficient to discriminate it from "JYT").

While the above research was important for establishing and linking rigorously defined cognitive mechanisms and structures, one of its shortcomings was its dependence on learning strategies that were handcrafted by psychology experts. For example, EPAM's (and CHREST's – which inherited most of EPAM's mechanisms) default S-R learning strategy was pre-defined as "learn stimulus as little as possible, before switching attention to the response; learn the response fully" (e.g., with the pair "XEJ" – "BIJ", only learn letter "X" from the "XEJ" stimulus, but learn the response "BIJ" fully). Despite this hand-crafting, the EPAM models often did not fit the human data very well; for example, the difference between EPAM and the human learning error rates differed by almost a factor of three for recall of the low-similarity stimuli (see Table 3).

V. Current Study

The current study aims to simulate two verbal learning tasks reported by Chenzoff (1962) and Hintzman (1969), and simulated by EPAM (Richman et al., 2002). Crucially, using GEMS to generate CHREST models will allow us to move away from hand-crafting task-specific attentional and learning strategies, account for individual differences, and optimise the fit of the models to the human data.

VI. Method

The human data for this project were taken from previous research: an investigation of the effect of pattern prefamiliarisation (i.e., prior knowledge) on learning rates (Chenzoff, 1962), and a study into the effect of stimuli similarity on memorisation and backward recall (Hintzman, 1969). The EPAM data – which is the current state-of-the-art for these simulations – were taken from Richman et al. (2002).

A. The Effect of Prefamiliarisation of Stimuli and Responses

Chenzoff's (1962) study simulated the effect of prior knowledge/prefamiliarisation on the learning rate. Four experimental conditions were devised, varying the familiarity (F) and unfamiliarity (U) with both stimuli and responses: F-F, U-F, F-U, and U-U. For example, F-F means that both stimuli and responses were familiar to the model; U-F means that the stimuli were unfamiliar, but the responses were familiar.

Ten stimulus – response pairs of nonsense CVC (consonant vowel consonant) trigrams were used in each condition. The stimulus trigrams were: VOD, HAX, CEM, KIR, SIQ, FEP, BAJ, LOZ, TUW, and YUG. The corresponding response trigrams were: XIL, TOQ, WEP, DUF, MIZ, JUK, NAS, HOV, BIR, and GAC. Thus, the complete list of possible S-R pairs was: VOD-XIL, VOD-TOQ, …, YUG-GAC.

Familiarity was manipulated during the first (i.e., the prelearning) part of the experiment. Depending on the experimental condition, models were trained either with the stimuli, the responses, both, or nothing at all. Like with the original human experiment, training stopped when the models had memorised the entire training list.

During the second part of the experiment, models were tasked with learning a S-R association list for all four experimental conditions. The experiment stopped when the models gave the correct response to each of the stimuli. The total number of trials and learning time was recorded for each of the four conditions, as well as converted into the ratio of the F-F condition.

Table 1. Overview of GEMS operators. Each operator type had a time cost (in milliseconds, ms) as follows: input (100 ms), output (140 ms), LTM (2,000 ms for familiarisation, 10,000 ms for discrimination), and syntax (0 ms).

Operator	Function	Type
PROG-X	a sequence of 2, 3 or 4 subprograms	Syntax
REPEAT2	repeats a subprogram 2 times	Syntax
ATTEND-STIMULUS	place the stimulus value into input slot 1	Input
ATTEND-RESPONSE	place the response value into input slot 2	Input
REC-AND-LEARN-ST	calls CHREST's recognise-and-learn-pattern function to learn a stimulus	LTM
REC-AND-LEARN-RES	calls CHREST's recognise-and-learn-pattern function to learn a response	LTM
RECOGNISE-ST	calls CHREST's recognize-pattern function to locate a pattern in long-term memory	LTM
LEARN-AND-LINK	calls CHREST's learn-and-link-two-patterns function to associate stimulus with response	LTM
RESPOND	retrieve the linked pattern using the stimulus and assign it to the model's output slot	Output
WAIT-X	advances model-clock (in ms): 1000 or 2000	Time

B. *The Effect of Stimulus Similarity on Backward Recall*

Hintzman's (1969) study was interested in humans' ability to recall a stimulus from the response. There were two experimental conditions: low similarity and high similarity. The low similarity stimuli were the following nonsense trigrams: SPP, JCL, PDR, HHN, CRB, GGJ, NBH, MJS, FSG, BNF, VFD, DMC, RVM, and LLV. The high similarity stimuli were: BDH, BDJ, BDL, BFH, BFJ, BGL, BGH, CDJ, CDL, CFH, CFJ, CFL, CGH, and CGJ. Each of the stimuli trigrams corresponded to a response – a random number ranging from 2 to 15. The models had to learn the correct response to each of the stimuli. For example, SPP-2 or CGJ-14. As in the human experiments, training stopped when models accurately responded to all stimuli, e.g., responding "2" when presented with "SPP".

The test phase involved presenting models with the responses and asking them to recall the corresponding stimuli. The resulting recall error rate was then recorded for both high similarity and low similarity stimuli.

VII. Procedure

In order to automatically generate verbal learning strategies and find high-quality variants, CHREST models were evolved by GEMS. Every simulation consisted of a population of initially random CHREST-based models being presented with a verbal learning task. The operators included `prog-x`, `attend-stimulus`, `recognise-and-learn-stimulus`, `attend-response`, `recognise-stimulus`, `recognise-and-learn-response`, `learn-and-link`, `repeat-2`, and `wait` (for one or two seconds) (see Table 1). The models were trained on a "per condition" basis (see above), with a population size of 200 and 100 generations; the mutation rate was set to 0.05; there were independent runs for all conditions. The fitness function was calculated as a function of correct recall and the absolute difference between the total learning time/recall ratios for humans and for GEMS. In the beginning of the cycle, each agent picked the operators randomly. The agents with the highest fitness (i.e., whose behaviour matched human data most closely) passed on their strategies to the next generation directly (possibly with some mutation) or with crossover with another agent. The cycle continued until the agents reached the 100th generation.

VIII. Results

The results of the best GEMS verbal learning models are presented in Table 2 and Table 3. GEMS was able to achieve a good fit to the human data in the simulation of both Chenzoff's effect of prior knowledge and Hintzman's backward recall studies. The tables also include the results of EPAM-VI (Richman et al., 2002).

*Table 2. The ratios of error rates in S-R pairs with respect to the F-F condition for humans (Chenzoff, 1962), EPAM VI, and GEMS. (Key: **F** stands for familiar pattern, **U** stands for unfamiliar pattern.*

Condition	People	EPAM VI	GEMS
F-F	1	1	1
U-F	1.2	1.9	1.2
F-U	1.6	2.8	1.3
U-U	1.8	3.7	1.6

Table 3. Percentage of correctly recalled stimuli in a backward recall task – for humans (Hintzman, 1969), EPAM VI, and GEMS.

Condition	People	EPAM VI	GEMS
High-Similarity	72	76	72
Low-Similarity	34	13	36

A sample strategy for the learning of the S-R list in the simulation of the prefamiliarisation experiment is presented in Figure 4; a sample strategy for the learning of the S-R list in the backward recall experiment is shown in Figure 5. Concretely, the program shown in Figure 4 is applied for each presentation of a S-R pair. Initially, just the stimulus is shown, and we can see from the second statement (`attend-stimulus`) that this is attended to, and the model then proceeds to devote a lot of cognitive operations and time to learn the stimulus. The response is only available to the model towards the end of the time period, and we can see the model attending to the response at the end. The previous stimulus and response will then be present in the model's STM before the next cycle, which is why the first statement is to LEARN-AND-LINK the previous stimulus-response pair. Of course, the program in the U-F condition has familiar responses (these

are prelearnt), so the model does not have to work so hard to learn them, and this is reflected in the program.

```
(PROG4 (LEARN-AND-LINK)
  (PROG4 (ATTEND-STIMULUS)
    (PROG4 (ATTEND-STIMULUS) (ATTEND-STIMULUS)
      (PROG3 (ATTEND-STIMULUS) (LEARN-AND-LINK) (LEARN-AND-LINK))
  (LEARN-AND-LINK))
  (RECOGNISE-ST) (LEARN-AND-LINK))
  (PROG2 (ATTEND-STIMULUS)
    (PROG4 (LEARN-AND-LINK) (LEARN-AND-LINK)
    (REC-AND-LEARN-RES) (RESPOND)))
  (PROG2 (WAIT-2000) (ATTEND-RESPONSE)))
```

Figure 4. One of the strategies in the final population of models that learnt S-R pairs in the U-F pattern familiarity condition (Chenzoff's experiment). See the Results section for an explanation of the program tree.

The backward recall program (in Figure 5) is much more varied in its attentional and learning strategies. This is because low similarity stimuli allowed it to pass training without fully learning them (this aspect of our simulation turned out to be similar to EPAM's). Thus, during the test phase, the model had to split its learning between mostly focusing on the stimulus initially, and then devoting most of its cognitive operations to learning and linking the responses (albeit in a less rigid fashion when compared to EPAM).

```
(PROG4
    (PROG4
        (PROG4 (ATTEND-STIMULUS)
        (PROG3 (REC-AND-LEARN-RES) (RECOGNISE-ST) (RESPOND))
        (WAIT-1000) (REPEAT2 (RECOGNISE-ST) (ATTEND-RESPONSE)))
    (WAIT-2000) (REC-AND-LEARN-RES) (ATTEND-RESPONSE))
    (REC-AND-LEARN-RES)
    (PROG4 (REPEAT2 (LEARN-AND-LINK) (REC-AND-LEARN-RES))
    (ATTEND-RESPONSE) (LEARN-AND-LINK)
    (REPEAT2 (WAIT-2000) (WAIT-2000)))
    (PROG4 (LEARN-AND-LINK) (RESPOND) (RESPOND)
        (PROG4
            (PROG4 (LEARN-AND-LINK) (ATTEND-RESPONSE) (LEARN-AND-LINK)
            (LEARN-AND-LINK))
        (REPEAT2 (ATTEND-RESPONSE) (REC-AND-LEARN-RES))
    (ATTEND-RESPONSE)
    (PROG3 (REC-AND-LEARN-RES) (RECOGNISE-ST) (RESPOND)))))
```

Figure 5. One of the strategies in the final population of models that learnt S-R pairs in the low-similarity condition (Hintzman's backward recall experiment). See the Results section for an explanation of the program tree.

For the prefamiliarisation experiment, the r^2 is 0.89 and the RMSE is 0.2. (These were not calculated for the backward recall experiment as there are only two data points.)

We should note that we are reporting only the best models, while many models achieved similarly high fit.

Please see https://github.com/Voskod/GEMS for the code and the best models for all of the experimental conditions.

IX. Discussion

A. Summarising the Results

The current study makes three important contributions. First, the models generated by GEMS moved away from hand-crafted learning strategies that were used in previous verbal learning research. Indeed, while EPAM prescribed one approach to S-R tasks (Richman et al., 2002), our evolved models developed a wide range of strategies. For example, EPAM had rigid rules for its attention function (e.g., try learning just the first letter of a stimulus before proceeding to learn the response in full). On the other hand, the learning strategies used by the GEMS-generated models were much more varied, with attention oscillating between a stimulus and a response multiple times. This provides a concrete example of our second contribution – how GEMS' automatic theory discovery helps to reduce experimenter's biases. We should also note that, despite our use of the genetic programming approach, the models generated by GEMS still conform to the central tenet of cognitive architectures – the same architecture was used for both experiments.

Third, the models generated by GEMS did a good job of simulating human data with regard to the error percentage in stimuli recall, and accounted for the data better than EPAM-IV. For example, humans' error rate with low-similarity stimuli was 34 percent, while for the EPAM-IV model this was approximately 13 percent (despite its hand-crafted verbal learning task-specific strategies). This is in contrast to the models generated by GEMS, which produced a group of strategies with an error rate of 36 percent.

B. Interpretability of Cognitive Models

The models (programs) evolved by GEMS share one of EPAM's strengths – they are not "black boxes" and work directly from the definitions of the operators; this is unlike the connectionist and Bayesian models that are often used in cognitive science. Thus, GEMS models are readily interpretable – both in terms of their underlying structures and the produced sets of cognitive strategies. The produced programs allow for both high and low-level analysis. For instance, glancing through the program in Figure 4 could result in a summarising statement like "the model was mostly focused on the stimuli and not the responses". At the same time, the details of timings and the order of cognitive operations could also be gleaned by a more careful analysis of the output. As an interesting sidenote, the attention shifts displayed by GEMS-generated models may also be in line with research into saccadic eye movement and the underlying attention function (Cajar et al., 2016).

C. Verbal Learning, CHREST, and Cognitive Architectures

Our choice of Verbal Learning was dictated by its computational simplicity, yet sufficient psychological complexity. Indeed, VL simulations share most cognitive mechanisms with much more complex CHREST simulations (e.g., concept learning, acquisition of syntactic categories by children, cognitive decline due to ageing, etc.) that were referenced in the CHREST section above. Thus, the problem of generalizing from an overly simplistic VL experimental setting to real-life complexity is arguably mitigated. (But, where each model takes several hours of real time training for learning of complex natural concepts, simulation of VL tasks is orders of magnitude faster – an important factor when evolving hundreds of models.) In other words, while it is true that intuitions about theory and model performance for low-dimensional stimuli do not necessarily transfer to higher-dimensional ones (Battleday, Peterson, & Griffiths, 2020), our simulations allowed us to sidestep this issue by utilising the same chunking operators that are used in the state of the art cognitive models that work with real-life complexity (e.g., Bennett et al., 2020; Freudenthal et al., 2016).

The largely shared mechanisms (i.e., familiarisation, discrimination, LTM node structure, and STM structure) between EPAM and the newer chunking theory-based models mean that their hand-crafted approach to (and simulated performance on) VL tasks would also be largely identical. Thus, our findings are likely to directly apply not just to

EPAM, but also to its direct descendants – cognitive architectures such as CIPAL (Jessop et al., 2024), MOSAIC (Freudenthal et al., 2016) and CHREST (Gobet & Lane, 2012; Bennett et al., 2020). The broader aspects of our findings are also likely to apply to non-chunking theory-based cognitive architectures that are hand-tuned for specific tasks (e.g., ACT-R, Soar, and LIDA).

D. Future Research and Conclusions

Our study may be subject to three important extensions in future research. Firstly, individual differences form an important aspect of psychology that is often absent from psychological models fit to "the average participant" data. This may be misleading as such data may obscure patterns at the level of the individual participant (Gobet, 2017; Newell, 1973; Vanpaemel & Storms, 2008). Our GEMS framework demonstrated that there may be multiple solutions (models) that satisfy a particular set of constraints. This is in line with research on individual differences in psychology – there is not a single cognitive system in nature, there is inherent variability. Of course, this variability is constrained. For example, individual bees vary in their social behaviour, as do humans, but the intraspecies variability is bounded by species-specific physiological and cognitive structures (Crespi, 2014, 2017; Rubenstein & Hofmann, 2015). In our case, the evolved agents shared the basic cognitive mechanisms and structures (as operationalised by CHREST), but differed in their approaches to S-R learning. For example, one model in the final population had a S-R learning strategy that contained 43 cognitive operations, while another model contained 24 operations. Our study is a rigorous demonstration of how the informational environment may shape both the cognitive strategies and the population of cognitive agents. Future research may extend our in-principle demonstration to link GEMS-generated models to the behaviour of individual participants.

Secondly, the computationally fast, yet psychologically complex verbal learning experimental paradigm may be once again utilised to design and tune cognitive architectures and human-like machine intelligence models. Our current focus was on the learning functions, but memory decay and memory interference experiments would be highly suited to VL, as would GEMS' automatic search and optimisation of cognitive strategy space. A concrete example of such a study would be the semi-automatic "Speed of Forgetting" (SoF) curve parameters' estimation, given ACT-R cognitive operators and human VL experiment data on forgetting. While the structure and the shape of forgetting curves' functions continues to generate debate (e.g., Capik et al., 2024; Sense et al., 2016), GEMS could help by semi-automatically searching the SoF problem space. Of course, the VL paradigm is not a requirement for GEMS – it is just one instance of the kind of data for which it can generate models – and future studies may utilise completely different data, for example in natural concept learning/forgetting tasks.

Thirdly, the current findings offer new possibilities in computational scientific discovery relative to most other machine learning tools that are typically used in cognitive science. For example, existing cognitive architectures (e.g., CHREST, ACT-R, Soar, etc) may be used to create the initial set of models/theories and cognitive operators in GEMS. When combined with human experimental data and a selection fitness function, GEMS would then generate new models – a process that can be more efficient than the experimenter's manual search through the potential model space. Furthermore, GEMS allows the combining of operators from multiple cognitive architectures into a single evolutionary pool, thus allowing for otherwise rare "cross-breeding" of models.

One potential criticism is that our models produced suboptimal learning strategies in order to fit the longer durations of human learning. There are two ways of addressing this criticism. Firstly, it raises a question: suboptimal with respect to what? Of course, a simple algorithm could rote learn an entire S-R list in one trial. But, adding slow learning rates, limited capacity of the STM and interaction of old and new knowledge in the LTM makes it necessary for the critic to show what *is* the optimal learning rate under these conditions – which is not trivial. Moreover, satisficing learning routines have long been known in psychology – e.g., as a form of "bounded rationality" (Simon, 1991).

Another potential criticism of the current modelling approach is "overfitting" – overly complex models achieve near perfect scores on training sets but have poor generalisability beyond the currently simulated data. (Tetko, Livingstone, & Luik, 1995). This study followed the advice of Simon (1992) and attempted to address the issue by doubling the ratio of data explained/free parameters – the same free parameters were used in all conditions of both the experiment on the role of prior knowledge and the backward recall task. Our models did not exhibit overfitting symptoms (the resulting programs were relatively short and psychologically meaningful); however, future research with bigger data sizes may be necessary to address the generalisability issue more fully. For example, one future extension to the current study would be to replicate EPAM's simulation of other verbal learning experiments without resorting to hand-crafting task-specific strategies.

To conclude, our study further integrates genetic/evolutionary aspects with a complex cognitive model and demonstrates a way to automate the discovery of task-specific cognitive processes. More broadly, our findings offer further support for the mechanisms proposed by chunking theory, connecting them to the evolutionary approach, and making further inroads towards a Unified Theory of Cognition (Newell, 1990).

REFERENCES

[1] Bartlett, L., Pirrone, A., Javed, N., Lane, P., & Gobet, F. (2023). Genetic programming for developing simple cognitive models. In F. K. A. M. Goldwater, B. K. Hayes, & D. C. Ong (Ed.), Proceedings of the 45th annual conference of the cognitive science society. Sydney, Australia.

[2] Battleday, R. M., Peterson, J. C., & Griffiths, T. L. (2020). Capturing human categorization of natural images by combining deep networks and cognitive models. Nature Communications, 11.

[3] Bennett, D., Gobet, F., & Lane, P. (2020). Forming concepts of Mozart and Homer using short-term and long-term memory: A computational model based on chunking. In S. Denison, M. Mack, Y. Xu, & B. C. Armstrong (Eds.), Proceedings of the 42nd annual conference of the cognitive science society (pp. 178-184). Toronto.

[4] Capik, A., Hake, H. S., Sener, S., Starr, A., & Stocco, A. (2024). Model-Based Characterization of Forgetting in Children and Across The Lifespan. https://doi.org/10.31234/osf.io/zgrh8

[5] Card, S., Moran, T., & Newell, A. (1983). The psychology of human computer interaction: Erlbaum.

[6] Chase, W. G., & Simon, H. (1973). Perception in chess. Cognitive Psychology, 4, 55-81.

[7] Chenzoff, A. P. (1962). The interaction of meaningfulness with s and r familiarizattion in paired associate learning. Carnegie Institute of Technology, Pittsburgh, PA.

[8] De Groot, A., & Gobet, F. (1996). Perception and memory in chess: Heuristics of the professional eye. London: Van Gorcum.

[9] Feigenbaum, E. A. (1959). An information processing theory of verbal learning. (P-1817). Santa Monica, CA

[10] Feigenbaum, E. A. (1963). The simulation of verbal learning behavour. Proceedings of the Western joint computer conference, 19, 121-132.

[11] Feigenbaum, E. A., & Simon, H. (1962). A theory of the serial position effect. British Journal of Psychology, 53(3), 307.

[12] Feigenbaum, E. A., & Simon, H. (1984). EPAM-like models of recognition and learning. Cognitive Science, 8, 305-336.

[13] Freudenthal, D., Pine, J., Jones, G., & Gobet, F. (2016). Developmentally plausible learning of word categories from distributional statistics. In D. Papafragou, D. Grodner, D. Mirman, & J. Trueswell (Eds.), 38th annual conference of the cognitive science society. Austin, TX.

[14] Frias-Martinez, E., & Gobet, F. (2007). Automatic generation of cognitive theories using genetic programming. Minds and Machines, 17, 287-309. doi:10.1007/s11023-007-9070-6

[15] Gobet, F. (1993). A computer model of chess memory. In W. Kintsch (Ed.), Fifteenth annual meeting of the cognitive science society (pp. 463-468): Erlbaum.

[16] Gobet, F. (2000). Discrimination nets, production systems and semantic networks: Elements of a unified framework. Evanston: The Association for the Advancement of Computing in Education.

[17] Gobet, F. (2017). Allen newell's program of research: The video - game test. Topics in Cognitive Science, 9(2), 522-532. doi:10.1111/tops.12265

[18] Gobet, F., & Lane, P. (2012). Chunking mechanisms and learning. In M. M. Seel (Ed.), Encyclopedia of the sciences of learning (pp. 541-544). New York: NY: Springer.

[19] Gobet, F., Lane, P. C., Croker, S., Cheng, P. C., Jones, G., Oliver, I., & Pine, J. M. (2001). Chunking mechanisms in human learning. Trends in Cognitive Sciences, 5(6), 236.

[20] Gobet, F., Lloyd-Kelly, M., & Lane, P. (2016). What's in a name? The multiple meanings of "chunk" and "chunking". . Frontiers in Psychology, 7.

[21] Gobet, F., & Simon, H. (2000). Five seconds or sixty? Presentation time in expert memory. Cognitive Science, 24, 651-682.

[22] Gregg, L. W., & Simon, H. (1967). An information processing explanation of one-trial and incremental learning. Journal of verbal learning and verbal behavior, 6, 780-787.

[23] Gunaratne, C., & Patton, R. (2022). Genetic programming for understanding cognitive biases that generate polarization in social networks. Paper presented at the Proceedings of the Genetic and Evolutionary Computation Conference Companion. https://doi.org/10.1145/3520304.3529069

[24] Hintzman, D. L. (1968). Explorations with a discrimination net model for paired associate learning. Journal of Mathematical Psychology, 5, 123-126.

[25] Hintzman, D. L. (1969). Backward recall as a function of stimulus similarity. Journal of verbal learning and verbal behavior, 8, 384-387.

[26] Jessop, A., Pine, J., & Gobet, F. (2024). Chunk-based Incremental Processing and Learning: An integrated theory of word discovery, implicit statistical learning, and speed of lexical processing. https://doi.org/10.31234/osf.io/dukpt

[27] Koza, J. R. (1992). Genetic programming: On the program ming of computers by means of natural selection. Cambridge, MA: MIT Press.

[28] Lane, P., Bartlett, L., Javed, N., Pirrone, A., & Gobet, F. (2022). Evolving understandable cognitive models. Paper presented at the Proceedings of the 20th international conference on cognitive modelling, Toronto.

[29] Lane, P., Cheng, P. C.-H., & Gobet, F. (2000). CHREST + : Investigating how humans learn to solve problems using diagrams. AISB Quarterly, 103, 24–30.

[30] Lane, P., & Gobet, F. (2012). Using chunks to categorise chess positions. In M. Bramer & M. Petrides (Eds.), Specialist group on artificial intelligence international conference 2012 (pp. 93-106). London: Springer-Verlag.

[31] Mathy, F., Fartoukh, M., Gauvrit, N., & Guida, A. (2016). Developmental abilities to form chunks in immediate memory and its non-relationship to span development. Frontiers in Psychology, 7, 201.

[32] Mccrary, J. W., & Hunter, W. S. (1953). Serial position curves in verbal learning. Science, 117(3032), 131.

[33] Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological Review, 63, 81-97.

[34] Newell, A. (1973). You can't play 20 questions with nature and win: Projective comments on the papers of this symposium. In W. G. Chase (Ed.), Visual information processing (pp. 283-308). New York: Academic Press.

[35] Richman, H. B., Gobet, F., Staszewski, J., & Simon, H. (1996). Perceptual and memory processes in the acquisition of expert performance: The EPAM model. In K. A. Ericsson (Ed.), The road to excellence: The acquisition of expert performance in the arts and sciences, sports, and games (pp. 167–187). Mahwah, MA: Erlbaum.

[36] Richman, H. B., & Simon, H. (1989). Context effects in letter perception: Comparison of two theories. Psychological Review, 96(3), 417-432. doi:10.1037/0033-295X.96.3.417

[37] Richman, H. B., Simon, H., & Feigenbaum, E. A. (2002). Simulations of paired associate learning using EPAM VI. Unpublished.

[38] Richman, H. B., Staszewski, J. J., & Simon, H. (1995). Simulation of expert memory using EPAM IV. Psychological Review, 102(2), 305-330.

[39] Rock, I. (1957). The role of repetition in associative learning. Amercan journal of psychology, 70, 186-193.

[40] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323(6088), 533.

[41] Schiller, M., & Gobet, F. (2014). Cognitive models of gambling and problem gambling. In F. Gobet & M. R. G. Schiller (Eds.), Problem gambling: Cognition, prevention and treatment (pp. 74-103). London: Palgrave Macmillan.

[42] Sense, F., Behrens, F., Meijer, R. R., & van Rijn, H. (2016). An Individual's Rate of Forgetting is Stable Over Time but Differs Across Materials. Topics in cognitive science, 8(1), 305–321. https://doi.org/10.1111/tops.12183

[43] Simon, H. (1974). How big is a chunk? Science, 183(4124), 482-488.

[44] Simon, H. (1981). Information-processing models of cognition. Journal of the American Society for Information Science, 32, 364-377.

[45] Simon, H., & Gilmartin, K. (1973). A simulation of memory for chess positions. . Cognitive Psychology, 5, 29-46.

[46] Smith, R., Gobet, F., & Lane, P. (2007). An investigation into the effect of ageing on expert memory with CHREST. Paper presented at the Proceedings of The Seventh UK Workshop on Computational Intelligence, Aberdeen.

[47] Thune, L. E., & Underwood, B. J. (1943). Retroactive inhibition as a function of degree of interpolated learning. Journal of Experimental Psychology, 32, 185-200.

[48] Vanpaemel, W., & Storms, G. (2008). In search of abstraction: The varying abstraction model of categorization. Psychonomic bulletin & review, 15(4), 732-749. doi:10.3758/PBR.15.4.732