Computational Scientific Discovery in Cognitive Science

Peter C. R. Lane
*S*chool of Physics, Engineering and Computer Science
University of Hertfordshire
Hatfield AL10 9AB, UK
p.c.lane@herts.ac.uk

Fernand Gobet
*C*entre for Philosophy of Natural and Social Science
London School of Economics and Political Science
London WC2A 2AE, UK
f.gobet@lse.ac.uk

***Abstract*—AI-guided scientific discovery will become an indispensable tool for scientists in the not too distant future. This half-day tutorial begins by introducing the area of computational scientific discovery within the cognitive sciences along with an overview of recent models of cognitive behaviour developed using these tools. The second half of the tutorial presents a detailed walk-through of our methodology and computational system, where participants will learn how to design experiments, evolve and analyse candidate models using the GEMS (Genetically Evolving Models in Science) system. Although specialised towards the cognitive sciences, many of the principles of model definition and discovery can be more broadly applied, and so the tutorial should be of interest to the wider Cognitive Machine Intelligence community.**

***Index Terms*—cognitive models, genetic programming, scientific discovery, visualisation**

I. Significance

This half-day tutorial provides a summary of some recent work on computational scientific discovery in the cognitive sciences, and a practical introduction to our own system of scientific discovery known as GEMS (Genetically Evolving Models in Science). Participants will learn how to apply a discovery system to develop computational theories of their own data, and how scientist and discovery system work together.

A challenge to theory development in the cognitive sciences is the abundance of reported experiments with their results often unreplicated, unanalysed and unmodelled. Cognitive scientists often represent theories of cognitive behaviour in the form of computer programs which simulate empirical data; examples include ACT-R [1] and CHREST [2]. However, developing such programs can be a time-consuming activity. In addition, attempting to find alternative solutions from which to develop theoretical insight can require a considerable amount of additional work: there might be multiple solutions from different theoretical paradigms and it can be important to compare and contrast theories on the same empirical data.

Computational scientific discovery can potentially resolve these issues by providing the computational means for scientists to intelligently analyse psychological data and convert it into explainable models fostering theoretical development [3]. As an example of such a system, we present GEMS: this system is built around a program synthesis technique which generates candidate models with the aid of evolutionary computing. Generated models are processed to simplify their overall structure, and a novel visualisation scheme is used to reveal similarities and contrasts between the models. Cluster analysis can separate candidate models into groups representing potential theoretical differences.

GEMS supports cognitive scientists/psychologists in their job, and does not replace them. The role of the human scientist is critical to the overall success of the system. The methodology which we are proposing involves a scientist first selecting the experimental phenomena that they wish to model, and then setting out the core theoretical assumptions from which a computational model may be constructed. The GEMS system is then employed to search the space of candidate computational models, returning a number of high-quality models. At this stage, the GEMS system already performs a useful service in demonstrating that there are (or are not) candidate models meeting the original theoretical assumptions.

GEMS, like any evolutionary system, tends to generate a large number of candidate solutions, many of which differ very little from each other. To aid the scientist in understanding the generated solutions, visualisations and quantitative metrics can be used to provide information on the diversity of the models produced. For example, in models of human behaviour there are likely to be several possible models each employing different internal strategies of visual attention or decision-making behaviour. The scientist is now able to navigate and consider the full range of generated models, look for repeating or unique patterns within the models, and begin to develop novel theories of human cognition.

II. Potential Audience and Expected Background

The primary audience for this tutorial would be cognitive scientists familiar with developing their own computational models. The tutorial will present such scientists with a new methodology to support and enhance their development of models and theories. A secondary audience would be machine-learning practitioners who are interested in applying their techniques to the area of scientific discovery. These practitioners would see how the application of searching and clustering

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. ERC-ADG-835002).

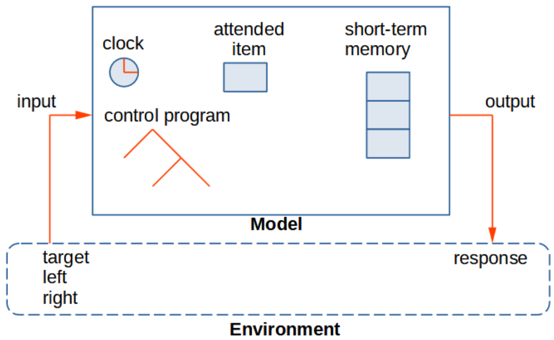

Fig. 1. Cognitive architecture used for DMTS models

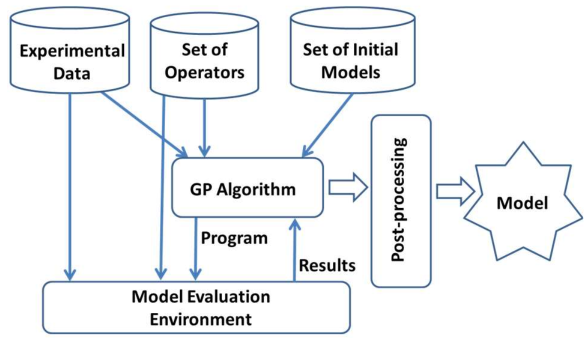

Fig. 2. Overview of GEMS system

algorithms can assist in the development of computational models and, ultimately, theories.

Although a technical background will be assumed of the audience, all examples and techniques will be carefully explained. This includes demonstrating what a typical psychological experiment is, how its data are gathered, and how these can be modelled using a cognitive architecture. Although the current implementation uses the ANSI Common Lisp language, the presented tutorial code will be explained. All the techniques presented may be readily implemented in an alternate environment.

III. Tutorial Description

The tutorial is structured so that participants will:

1) Acquire an understanding of the area of computational scientific discovery, especially as applied to the cognitive sciences and psychology;
2) Explore some recent models of cognitive behaviour developed using scientific discovery techniques; and
3) Learn how to design experiments and evolve models using the GEMS system using an example such as the Delayed Match to Sample task [4].

All materials for the tutorial, along with more information about GEMS, its publications and software, will be made available on our project website: https://gems.codeberg.page

At the heart of GEMS is the idea of developing a cognitive model: as an example, we use a standard, symbolic cognitive architecture, see Figure 1. This architecture has input/output, short-term memory and attention mechanisms; its operation is guided by a *control program*, unique to each specific model. Developing a cognitive model for some experimental data means creating a control program which enables the model to simulate some aspect of cognitive behaviour and produce results comparable to those collected from human participants.

GEMS is a system for automatically creating cognitive models by generating their control programs and, as such, is an example of *program synthesis*. Such systems can be conveniently divided into three parts [5]: the task definition (*user intent*), to express what makes a good program; a *search space* of candidate programs; and a *search technique*, to explore the given search space for good programs. Here, the developed programs form the control structure for the cognitive models.

The three stages can be seen in Figure 2, which provides an overview of the GEMS system: the *user intent* is setup by the scientist, who creates the definitions of the experimental data, set of operators and model evaluation, which combined produce a *search space*; the *search technique* is genetic programming, leading to a post-processing step from which one or more models are output for further analysis by the scientist. The tutorial will begin by explaining how to implement a simple cognitive model, using a virtual-machine structure to hold the cognitive architecture and its control program, an evaluation function and an experiment-simulation function. The simple model is then expanded to take account of time, and an example of the delayed match-to-sample (DMTS) task developed. Genetic programming is then introduced, and the post-processing steps explained and illustrated.

The tutorial will introduce a number of novel features of our approach, which extend beyond a direct application of genetic programming. The first such feature concerns the fitness function used to guide the evolutionary process. Like humans, the models are permitted to make mistakes, and their responses take time to make, so the models are assessed on two performance measures, and not directly on the accuracy of their input-output mapping. These two measures mean the fitness function is an example of multi-objective optimisation, and the tutorial will explain how to manage this search process using *phased evolution* [6].

The search process usually generates many models, most of which represent the same model semantically: we introduce a range of post-processing techniques which can dramatically reduce the number of models to a manageable number; in a typical example, the number of models was reduced by a factor of 100, from several thousand models to the order of 10 [6]. Finally, using visualisation and clustering techniques, it is possible to visualise and navigate the space of candidate models through a 2-dimensional representation; an example is shown in Figure 3.

Although such a visualisation provides a useful *qualitative* representation of the differences between models, a more *quantitative* comparison between different groups of models can be performed using cluster analysis techniques, such

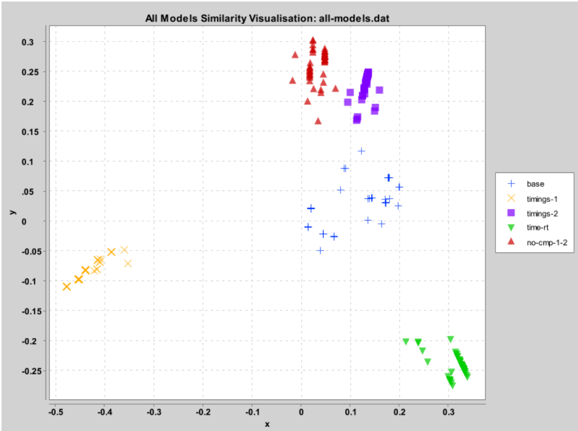

Fig. 3. Visualisation of final models as clustered points

as by computing the Silhouette Index [7]. Quantifying the relative diversity of different clusters provides information about which clusters are more or less similar to each other, which then guides the scientist to the more important models for further analysis. In the DMTS example we use here, the different kinds of models highlight different decision-making and visual-attention strategies.

The tutorial will present a number of successful applications of computational scientific discovery. These include recent publications using the GEMS system for the delayed match-to-sample task (DMTS) [6], [8]; the Posner Task [9], [10]; CHREST [2] models of Verbal Learning; and decision making [11]. The applications exemplify the operation of the GEMS system, both at a design level and in their implementation. In addition, the applications will demonstrate how analysis of the models generated by GEMS can provide insights into cognitive behaviour. In particular, they showcase the strength of such scientific discovery tools in generating a diversity of models and potential theories from the fundamental assumptions provided by the scientist.

IV. Biographical Sketch of Presenters

Peter Lane is a Senior Lecturer in Computer Science at the University of Hertfordshire, where he teaches programming and data science. He received his PhD in Computer Science at the University of Exeter, UK. His research interests include the methodology of developing computational models.

Fernand Gobet is a Professorial Research Fellow at the London School of Economics. He received his PhD in psychology from the University of Fribourg (Switzerland). He is a Fellow of the Psychonomic Society. His current research interests include computational scientific discovery, computational modelling, and the psychology of expertise and talent.

The two presenters have a long experience of presenting tutorials together at international conferences, for example, on the EPAM/CHREST cognitive architecture [12]. More recently, in December 2023, they successfully delivered a workshop similar to the current proposal at AI-2023: http://bcs-sgai.org/ai2023/?section=workshops

References

[1] J. R. Anderson and C. Lebière, Eds., *The Atomic Components of Thought*. Mahwah, NJ: Lawrence Erlbaum, 1998.

[2] F. Gobet and H. A. Simon, "Five seconds or sixty? presentation time in expert memory," *Cognitive Science*, vol. 24, no. 4, pp. 651–682, 2000.

[3] L. K. Bartlett, A. Pirrone, N. Javed, and F. Gobet, "Computational scientific discovery in psychology," *Perspectives on Psychological Science*, vol. 18, no. 1, pp. 178–189, 2023.

[4] L. L. Chao, J. V. Haxby, and A. Martin, "Attribute-based neural substrates in temporal cortex for perceiving and knowing about objects," *Nature Neuroscience*, vol. 2, pp. 913–20, 1999.

[5] S. Gulwani, "Dimensions in program synthesis," in *Proceedings of the 12th International ACM SIGPLAN Symposium on Principles and Practice of Declarative Programming*, 2010, pp. 13–24.

[6] P. C. R. Lane, L. K. Bartlett, N. Javed, A. Pirrone, and F. Gobet, "Evolving understandable cognitive models," in *Proceedings of the 20th International Conference on Cognitive Modelling*, T. Stewart, Ed. University Park, PA: Applied Cognitive Science Lab, Penn State, 2022, pp. 176–182.

[7] P. C. R. Lane, N. Javed, L. K. Bartlett, D. Bennett, and F. Gobet, "Exploring the diversity of evolved cognitive models with cluster analysis," in *Proceedings of the 12th International Conference on Frontiers of Intelligent Computing: Theory and Applications*, 2024.

[8] L. K. Bartlett, A. Pirrone, N. Javed, P. C. R. Lane, and F. Gobet, "Genetic programming for developing simple cognitive models," in *Proceedings of the 45th Annual Meeting of the Cognitive Science Society*, M. Goldwater, F. K. Anggoro, B. K. Hayes, and D. C. Ong, Eds., 2023, pp. 2833–2839.

[9] ——, "Generating candidate cognitive models for the Posner cueing task," in *45th European Conference on Visual Perception*, 2023.

[10] L. K. Bartlett, N. Javed, D. Bennett, P. C. R. Lane, and F. Gobet, "Modeling attentional cueing and inhibition of return with genetic programming," in *Proceedings of the Eleventh Annual Conference on Advances in Cognitive Systems*, 2024.

[11] A. Pirrone, P. C. R. Lane, L. Bartlett, N. Javed, and F. Gobet, "Heuristic search of heuristics," in *Artificial Intelligence XL*, M. Bramer and F. Stahl, Eds., vol. LNCS 14381. Cham: Springer Nature Switzerland, 2023, pp. 407–420.

[12] P. C. R. Lane and F. Gobet, "EPAM/CHREST tutorial: Fifty years of simulating human learning," in *Proceedings of the 31st Annual Meeting of the Cognitive Science Society*, N. Taatgen and H. van Rijn, Eds., 2009, pp. 13–14.