Building computational models of social cognition in memo

Kartik Chandra, MIT (kach@mit.edu)
Sean Dae Houlihan, Dartmouth College (dae.houlihan@dartmouth.edu)
Max Kleiman-Weiner, UW (maxkw@uw.edu)

Background & Motivation

One of the most influential computational paradigms in modern cognitive science is the Bayesian modeling of social cognition (Cushman, 2024). This paradigm models people's intuitions about other agents in terms of recursive probabilistic reasoning: agents are treated as approximately-rational decision-makers, who make Bayesian inferences about *other* agents' mental states from their observable behavior. Computational models designed in this tradition have been used in seminal work on theory-of-mind (Baker, Jara-Ettinger, Saxe, & Tenenbaum, 2017), language and communication (Goodman & Frank, 2016), emotion understanding (Houlihan, Kleiman-Weiner, Hewitt, Tenenbaum, & Saxe, 2023), and many other key areas of interest in cognitive science.

In practice, however, designing such computational models requires a tremendous amount of technical expertise. A cognitive scientist working in this area needs not only a firm understanding of the domain of cognition they seek to model, but also (1) a strong grasp of the theoretical foundations of decision theory, game theory, probability theory and information theory, (2) working proficiency with common algorithms for approximately-optimal decision-making under uncertainty, and (3) practical experience implementing probabilistic models and inference procedures using probabilistic programming languages (PPLs). In addition, there is (4) a lot of unwritten "folk wisdom" that guides researchers in designing and debugging their models: intuitions about what is easy and difficult to model with current tools, what is computationally tractable with current hardware, and what common classes of bugs one should be aware of. Together, these requirements make this powerful methodological framework inaccessible to many researchers in our field.

We seek to address this problem with a CogSci Tutorial. We believe the time is right for two reasons. First, there is currently a tremendous amount of interest in this area, driven in part by emerging questions about how to endow modern AI systems (e.g. large language models) with human-like social intelligence. Second, recent years have seen the development of a variety of powerful new tools that make it dramatically easier to build such models.

In particular, this tutorial will focus on the use of a new PPL, **memo** (Chandra, Chen, Tenenbaum, & Ragan-Kelley, 2025, or see website https://github.com/kach/memo), which was designed from the ground up specifically for implementing computational models of social cognition. memo provides dedicated abstractions for agents, beliefs, and actions, which can be used to write simple, intuitive implementations of computational models of social cognition (often using 4-$5\times$ fewer lines of code than handwritten implementations). Additionally, memo compiles models to optimized GPU code that typically runs 100-$1000\times$ faster than previous approaches. Together, these features not only lower the barrier to entry for novice modelers, but also greatly expand the scale and complexity of computational models that experienced researchers can study. For this reason, memo is being used by a growing number of computational cognitive science and AI labs around the world—indeed, it was used in several papers that will be presented at CogSci 2025. This tutorial will give a hands-on introduction to memo, empowering participants to build computational models of social cognition using state-of-the-art tools and techniques.

Audience & Objectives

Our tutorial will be designed to meet the needs of a broad audience of students and faculty at many levels of expertise.

For novice participants looking to start building computational models of social cognition, we wish to equip them with the tools they need to get started. At the end of the tutorial, novice participants should feel confident in implementing classic models and motifs (e.g. Bayesian inverse planning, Rational Speech Acts) in memo. Furthermore, participants should understand the conceptual vocabulary of this paradigm well enough to seek out the help they need for more ambitious modeling efforts.

For experienced participants who have built such models before, we wish to provide a structured space for discussing the state of the art in this paradigm. As a community, what are the greatest practical challenges we currently face? How do we currently work around those challenges? Individual labs often develop their own "tricks of the trade" passed on by oral tradition; we will design this tutorial to provide a forum where this wisdom can be exchanged for the benefit of all. Finally, we will discuss how we can address those challenges, and what new research directions that would unlock for us.

Beyond the tutorial, we hope the resources we create can be useful to researchers and students in the future: for example, in the form of reference manuals and course materials.

In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society*

Schedule

We are proposing a half-day (3-hour) tutorial session, with the following schedule:

- **Before the tutorial:** In the weeks leading up to the tutorial, we will post a variety of optional materials for participants to consider reading in preparation. These materials will include a curated set of key papers, review articles, and book chapters that cover important background on Bayesian models of social cognition, as well as newly-created resources such as step-by-step guides to installing the requisite software.

- **Welcome and review of theory (30min):** After introducing the workshop and organizers, we will briefly review the mathematical theory behind Bayesian models of social cognition, focusing on key concepts like expected utility maximization, softmax-rationality, and inverse planning.

- **Introduction to memo (45min):** Turning this theory to practice, we will guide attendees in implementing a classic model using the memo probabilistic programming language. We will do this interactively by "live-coding" the model from scratch on a projector, working closely with the attendees every step of the way. We will show how to set up a model, how to visualize its predictions, how to debug common errors, how to fit the model to behavioral data, how to perform statistical analyses of fitted parameters, and how to reason about the model's computational tractability and scalability.

- **Hands-on exploration (45min):** Next, attendees will have some structured time to play with the new tools they have acquired, and to practice building models of their own. All of the organizers, as well as a handful of additional teaching assistants, will give 1:1 hands-on guidance and help with debugging. We will have a variety of "starter packs" available for different kinds of projects, and we will encourage attendees to work in small groups united by common interests (e.g. language, emotion, planning, morality). We will communicate with attendees ahead of time to help them get relevant software installed before the tutorial, so that they can hit the ground running.

- **Frontiers talks (30min):** In the last portion of the tutorial, we will give attendees a taste of the current research landscape by presenting three invited "flash talks" on frontiers of computational models of social cognition (all using memo). We will ask speakers to focus on the practical implementation issues they faced and overcame in their work.

  The following speakers have agreed to present:

  1. Marlene Berke, PhD student at Yale University
  2. Alicia Chen, PhD student at MIT
  3. Katherine Collins, PhD student at Cambridge, UK

- **Horizons panel (30min):** We will end the tutorial with a guided discussion on the current challenges and open questions in this area. What are the limitations of our existing tools, and how do we currently get around those limitations? What tools are currently missing from our toolbox? What would it take to develop new tools, and what new research directions would such tools unlock? Our goal is for all attendees to understand the limits of what is currently possible with memo, and for some attendees to be inspired to figure out how to overcome those limitations.

Organizers

Kartik Chandra is a PhD student at MIT, and is the lead developer of the memo programming language. **Sean Dae Houlihan** is a Postdoctoral Fellow at Dartmouth College and is currently teaching a full term course on computational models of social cognition using memo (website: https://comosoco.daeh.info). **Max Kleiman-Weiner** is an Assistant Professor at the University of Washington, Seattle.

References

Baker, C. L., Jara-Ettinger, J., Saxe, R., & Tenenbaum, J. B. (2017). Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, *1*(4), 0064.

Chandra, K., Chen, T., Tenenbaum, J. B., & Ragan-Kelley, J. (2025). A domain-specific probabilistic programming language for reasoning about reasoning (or: a memo on memo). *psyarxiv preprint*. Retrieved from https://doi.org/10.31234/osf.io/pt863

Cushman, F. (2024). Computational social psychology. *Annual Review of Psychology*, *75*(1), 625–652.

Goodman, N. D., & Frank, M. C. (2016). Pragmatic language interpretation as probabilistic inference. *Trends in cognitive sciences*, *20*(11), 818–829.

Houlihan, S. D., Kleiman-Weiner, M., Hewitt, L. B., Tenenbaum, J. B., & Saxe, R. (2023). Emotion prediction as computation over a generative theory of mind. *Philosophical Transactions of the Royal Society A*, *381*(2251), 20220047.