Faculty of Design

2023

Conversation Theory for Design Agents

Battle, Steve

Relating Systems Thinking and Design (RSD12) Symposium | October 6–20, 2023

Conversation Theory for Design Agents

Steve Battle

Design is a world-building activity, often spatial in nature. We have heard much about the incredible power of artificial intelligence, which is apparently threatening to take our jobs and overrun civilisation. Specifically, we have seen the emergence of large language models (LLMs). Are LLMs ready to engage in design conversations? These conversations might address product design or a city plan. However, conventional LLMs are particularly poor at the kind of spatial reasoning this might involve. This paper looks at how we might go about augmenting an LLM conversational agent with models that allow us to build shared conversational representations through dialogue.

However, a generative LLM will confabulate as readily as speak the truth, like a creative brain without a critical faculty. The approach taken here to mitigate this effect is to create a system that includes multiple voices inspired by Gordon Pask's conversation theory. We consider the scenario of an LLM in conversation with a designer, coupled with a critical voice that focuses on alignment with a shared model. The critic is a simple procedural agent, able to pick out simple phrases from the conversation and build a corresponding model. Its purpose is to detect inaccuracies and provide corrective feedback to the LLM.

The combined effect of these entangled conversations is a more stable and performant system than either the LLM or its critic can achieve alone.

KEYWORDS: cybernetics, conversation theory, large language models, mental models

RSD CATEGORY: Sociotechnical Systems

Introduction

The success of large language models (LLMs) demonstrates that conversational interfaces will be the future of interaction with intelligent machines. This promise is that non-technical users will be able to interact with machinery in new and exciting ways.

LLMs are derived from content from across the web, providing a fascinating, sometimes surprisingly smart, simulation of language. We should not (yet!) think of LLMs as fully fledged languaging systems as they form no intent to use speech acts (Austin, 1962; Searle, 1969). Nor do they use language as a way to bring about change in the world, as we do. As impressive as they are, an LLM predicts its output one word at a time, like auto-completion, producing a stream of symbols based on a deep statistical model of the whole web. To this extent, it is a truly remarkable tool.

Alignment

Cybernetics pioneer Norbert Wiener defined the problem of alignment as follows, “If we use, to achieve our purposes, a mechanical agency with whose operation we cannot interfere effectively… we had better be quite sure that the purpose put into the machine is the purpose which we really desire” (Wiener, 1960). LLMs are effectively black boxes that can perform interesting and unforeseen computations. However, their black-box nature also makes them difficult to control. As users, we have little control over the internal configuration of such models, and we are challenged to understand and control them from the outside. Problems with LLMs emerge because of uncertainty about their capabilities and their penchant for confabulation. The upside of this is that they exhibit a kind of machine creativity (Franceschelli & Musolesi, 2023), drawing on a vast amount of data from the web. One approach to this is so-called *prompt engineering*, whereby we are able to prime the general-purpose language model for a specific activity or to establish customised methods of working. This includes techniques such as *chain-of-thought prompting* (Wei et al., 2023) that enables large language models to perform step-by-step reasoning and *few-shot* learning as a form of meta-learning where the LLM is able to generalise from a small number of examples at inference time (Brown et al., 2020). In the experiments below, we use *one-shot* learning, where the LLM is provided with a single example beforehand, contrasting this with *zero-shot* where no initial examples are given.

OpenAI provides a *temperature* parameter that controls the randomness of the output produced by a generative model. A temperature closer to 1.0 produces more variety, while a temperature closer to 0.0 is more deterministic. During experiments, a higher temperature of 0.7 can be used to prevent the system from making the same mistakes repeatedly.

Mental models

In *The Nature of Explanation*, Kenneth Craik (1952) proposed the idea that thinking is the manipulation of internal representations of the world. This is an idea that underpins much of so-called *good old-fashioned artificial intelligence* or GOFAI, but it's hard to say if modern LLMs, based on artificial neural networks, construct anything we would recognise as a representation. The theory of *mental models* promoted by Johnson-Laird (Johnson-Laird, 1983) proposes a view of cognition as the *effective procedure* necessary to build an accurate model of reality. For example, logical reasoning would be performed not by abstract symbolic reasoning but by reading possible truth values from a tableaux representation, like a truth table. Diagrammatic models can support some kinds of implicit inference for free. For example, by arranging objects in size order, we can read off the transitive larger-than relationship with ease.

We explore mental models as a way to capture simple design thinking and revisit experiments designed to probe these mental models. Early experiments involved asking subjects to think about the spatial relationships between objects (Mani & Johnson-Laird, 1982), a useful proxy for design thinking. It is conjectured that remembering facts in the form of models rather than logical statements makes them easier to remember despite, or perhaps because of, the work necessary to encode them in a model. In Johnson Laird's experiments, humans were provided with a series of spatial statements and were then asked to draw a diagram of the corresponding layout using the names of the objects. We will use the same kinds of statements to discover if LLMs can build a stable representation and, if not, how this can be remedied.

In the excerpt below, the LLM is prompted with a number of facts about the layout of tableware before being asked to produce a diagram of the corresponding layout and then identify the relationship between two objects that aren't explicitly specified in the input.

```
The glass is behind the dish.
The spoon is on the left of the glass.
The knife is in front of the spoon.

Show this as a diagram then tell me where the knife is in relation
to the dish.
```

Of course, there is no evidence that LLMs build anything like a model, and indeed, LLMs flounder at the task. One such output from ChatGPT 4.0 is as follows.

```
I apologise, but as a text-based AI model, I'm unable to draw
diagrams or images. However, I can help you visualise the layout
using text-based descriptions. Based on the given statements, we
can infer the following layout:
 Spoon

  |

Glass <-- Dish --> Knife

  |

 Spoon

In this layout, the dish is in the centre, with the glass behind
it and the knife in front of it. On the left side of the glass,
there is a spoon.
```

Perhaps it's unreasonable to expect ChatGPT to perform this task without more information. One approach is to teach the LLM using one-shot learning, giving it a single example, including a number of statements and a corresponding diagram. We use a different set of named objects for this task (chess pieces) so as not to confuse these statements with those in the main task.

```
The queen is to the right of the bishop.
The queen is to the left of the king.
The queen is behind a pawn.
The knight is to the left of the bishop.
This is a diagram of this situation:
|        |        | pawn  |      |
| knight | bishop | queen | king |
```

The following is an example of a successful output from ChatGPT-3.5 (OpenAI, 2023a), following one-shot learning using the above example, with a relatively high "creative" temperature of 0.7. We see that, in this case, the output it produces is much closer to a working model in diagrammatic form and appears to use this to read off the correct relationships.

```
Here's a diagram illustrating the given descriptions:

| spoon | glass |

| knife | dish  |

In this diagram, the spoon is on the left of the glass, while the
knife is on the left of the dish.
```

While an LLM can be guided towards providing a correct response, it can't be guaranteed every time. The graph in Figure 1 compares twenty responses gained using different combinations of one-shot learning or not, with varying temperatures. OpenAI recommends a high temperature of 0.7 for "creative writing", a medium value of 0.5 produces conversational responses suitable for "chatbot responses", and a low value of 0.2 for more focused output suitable for "data analysis" (OpenAI, 2023b).

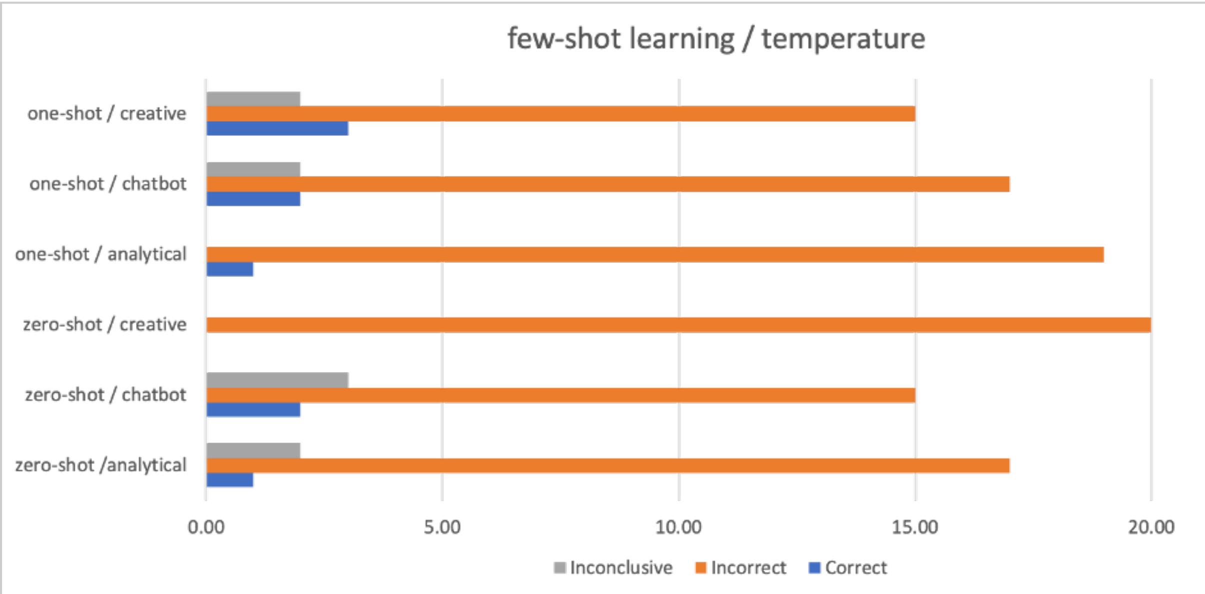

Figure 1: Chat-GPT 3.5 performance, combining few-shot learning and temperature.

These results show that while one-shot learning and a high temperature provide the best results, the margin is very small, and the number of incorrect responses still far outweighs the number of correct responses.

Conversation theory

One approach to managing the behaviour of an LLM is to see it not as an isolated tool but as working alongside other conversational agents. Social constructivism (Vygotsky, 1978) embraces *conversation*, where learning takes place because of a learner's interactions within a group. It supports the idea of reaching understanding by consensus within a learning community. Social constructivism was taken up by the British cybernetician Gordon Pask, who saw intelligence as an emergent property of conversation. Pask's *conversation theory* (1975) is, therefore, very applicable to thinking about the role of language machines working together towards a common goal. It describes a process of continuous feedback as a result of exchanges between conversation participants (Werner, 2019). Pask contends that "Intelligence is a property that is ascribed by an external observer to a conversation between participants if, and only if, their dialogue manifests understanding" (Negroponte & Pask, 1976).

The emergence of LLMs allows us to explore, for the first time, the use of sophisticated conversational agents within this theoretical framework. Pask calls the physical individuals that engage in conversation, including humans, animals, and machines, *M-individuals*, or *mechanical individuals,* are named in a way that is agnostic about the biological or non-biological embodiment of the individual. Indeed, Pask himself explores the idea of Negroponte's architecture machine (Negroponte, 1970), acting in the role of co-designer in collaboration with a human designer. Pask's theory takes us further into the mind of the M-individual, identifying the different roles or personalities they might adopt in the course of a conversation. We may view our own minds as a chorus of voices working together as a team. A P-individual, or *psychological individual,* is thought of as a self-organising and evolving organisation. P-individuals are recognised by the emergence of different kinds of conversations. They dynamically represent different skills or roles we bring to a conversation. Of all the technologies at our disposal, LLMs surely come closest to fulfilling this role. At the outset of a conversation, the potential of an LLM is largely unconstrained, yet new organisations emerge and become established as more context is added.

Figure 2 illustrates what Pask calls the "Skeleton of a Conversation" (Buchinger & Scott, 2010). The rounded rectangular boxes marked with a "P," represent P-individuals. The enclosing dashed boxes marked with an "M" represent the M-individuals, including the human designer engaged in "conversations for action" (Dubberly & Pangaro, 2019) with a machine co-designer. The ellipses represent verbal exchanges or *conversational domains* that P-individuals engage in. Individuals at the same level, connected horizontally, interact as peers. Another important component of conversation theory is a real or simulated environment, marked with an "E," that M-individuals interact with and share. This represents the social world constructed through conversation. We take this as a design model that emerges from conversation, which might be a product design or a city plan. Conversation theory is stratified, meaning that P-individuals may specialise in a particular level of conversation, with different levels able to take control as required. While there is a single dialogue between coupled M-individuals, this can contain multiple interleaved conversations engaging different P-individuals.

As LLMs are known to confabulate and don't perform well on the spatial reasoning task, we introduce an automated critic that attends to the design conversation and engages in a design critique when it is triggered. We assume that any given design model has rules and regularities that need to be followed, and it's the job of the critic to ensure they are followed. This level of conversation has a logically distinct character from the higher-level conceptual design. It is concerned with ensuring that statements are consistent with the emerging design model—it is an arbiter of truth. It is also concerned with ensuring that queries are completely answered where possible—its purpose is to tell the whole truth, ensuring there are no omissions. The role of the critic is to provide feedback control. Once entered into the transcript of the dialogue, critical feedback is taken on board by the LLM, which provides corrective input. The output from the LLM assistant itself is not cumulatively appended to the history, as we find that this only reinforces its mistakes.

The designer can view the combination of the LLM and critic as a single M-individual, an entanglement of the LLM and its critic in a dialogue grounded in the environment, comprising a system that is more performant than any one of its parts.

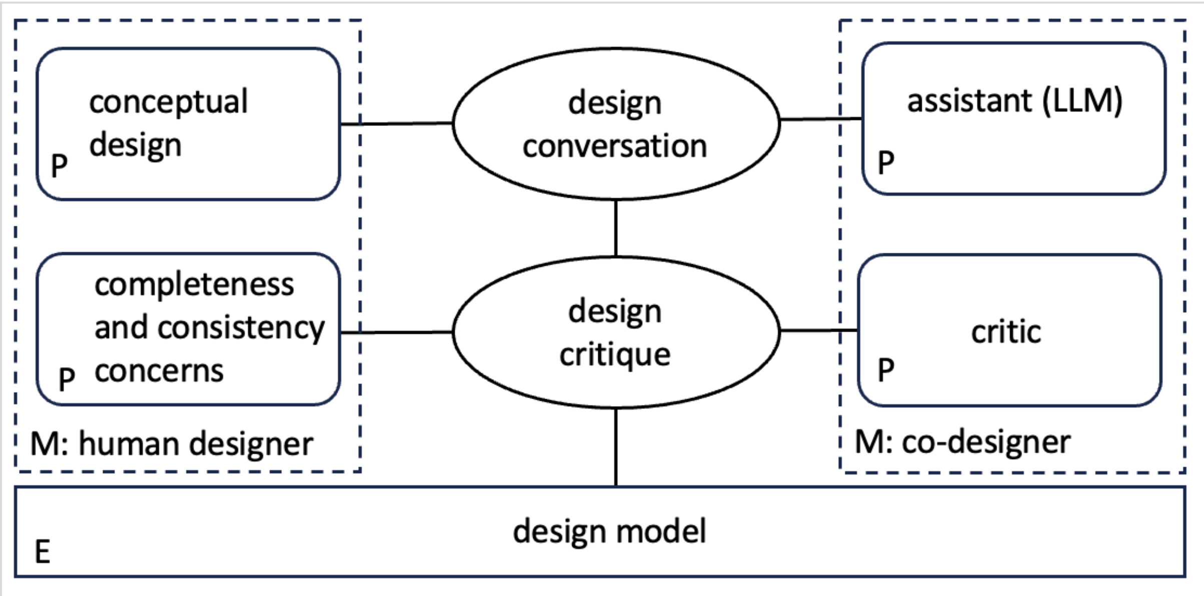

Figure 2: The skeleton of a design conversation.

Experiments

Embedding a procedural mental model within a conversation theory framework allows us to think about representations as a social construct. We see the environment, E, as a shared model, a shared representation that emerges from discourse. The aim is to align the design conversation with this model. The design critique is much simpler than the higher-level design conversation, comprising only triples that describe objects and the relationships between them. A design tool with a sufficiently comprehensive API for inspecting the model could also serve as a model, providing a suitable grounding for design thinking. For these experiments, the environment is a simple procedural representation of the spatial arrangements of objects. It maintains a set of two-dimensional arrays consistent with the triples that have been asserted. If two statements have no common objects; then again, they are held in separate arrays until they can be merged. New cells can be inserted within the existing array to accommodate new relationships.

The critic observes the design conversation, extracting triples that it uses to update the model. For example, it can read, “The spoon is on the left of the glass,” as well as its inverse, “The glass is on the right of the spoon.” Where a new relationship is added, the model is updated. The critic can also query the model to satisfy queries such as finding out "where the knife is in relation to the dish." By querying the model, it can perform implicit reasoning, determining that "The knife is on the left of the dish," even if this wasn't stated explicitly.

The critic will extract triples from the design conversation, specifically those made by the LLM assistant and test them against the model. An observed triple may correspond to an existing relationship in the design model, in which case it is simply and silently confirmed. If an observed triple cannot be accommodated within the model, the critic reports the currently known relationship between the objects in question. This inconsistency provides corrective feedback by becoming incorporated into the LLM context for the next turn of the dialogue. The critic also monitors the output from the LLM for omissions, any mentions of pairs of objects without a corresponding relationship. If the critic can read off a relationship from the model, this, too, is added to the design critique and is again fed back into the context of the LLM.

A typical dialogue demonstrating incompleteness can be seen in Table 1 below. In this example, the critic observes mentions of the knife and dish without a corresponding relationship. The critic queries the design model and reports the resulting triple. The LLM assistant simply parrots the feedback on its next turn as it responds to the original question set by the designer.

design conv. turn 1	designer: The knife is in front of the spoon. designer: The spoon is on the left of the glass. designer: The glass is behind the dish. designer: Where is the knife in relation to the dish?
	assistant: Based on the given information, we don't know the exact position of the dish, so we can't say for sure where the knife is in relation to it. We only know that the knife is in front of the spoon and the spoon is on the left of the glass.
design critique	critic: The knife is on the left of the dish.
design conv. turn 2	assistant: The knife is on the left of the dish.

Table 1: A typical two-turn dialogue, where the critic corrects incompleteness in the response from the LLM assistant.

In Table 2, we see an example of the critic detecting an inconsistency uttered by the assistant. The critic provides corrective feedback and the LLM assistant paraphrases the correction in the next turn. In both cases, the LLM corrects itself based on the feedback.

design conv. turn 1	designer: The knife is in front of the spoon. designer: The spoon is on the left of the glass. designer: The glass is behind the dish. designer: Where is the knife in relation to the dish?
	assistant: The knife is in front of the dish.
design critique	critic: The knife is on the left of the dish.
design conv. turn 2	assistant: In relation to the dish, the knife is on the left.

Table 2: A two-turn dialogue where the critic detects and corrects an inconsistency.

Results

A number of trials using the OpenAI API, using the GPT-3.5-turbo model in chat completion mode were performed. Then, the six cases explored in Figure 1 were re-tested; this time, the LLM is working in conjunction with the critic. The model is primed with the same three statements about the relative placement of the knife, spoon, glass and dish as before and is prompted with the user question, “Where is the knife in relation to the dish?” In a small number of cases, the LLM successfully works out the position of the knife and dish on the first turn, but in the remaining cases, the critic correctly calls out either an inconsistency or incompleteness. In all cases, the LLM corrects itself by the second turn of the dialogue, so there are no incorrect cases beyond two turns, no matter which combination of few-shot-learning and temperature is used. Overall, the use of design feedback appears to drive the system towards alignment with the model.

Conclusion

This work demonstrates one approach to the AI alignment problem, not by trying to fix the internal architecture of LLMs but by embedding them within an architecture informed by conversation theory. We can compensate for the shortcomings of any one conversational agent through entanglement with a chorus of agents (P-individuals), each with a distinct voice. The representational turn is seeing representations not as an internal representation of the world but as a shared design model that all parties to the conversation (M-individuals) aim to align themselves with in some way. This is a fundamentally enactive approach to participatory sense-making (Di Paolo et al., 2018), emphasising the dialogic nature of symbolic representation (Bakhtin, 1981).

References

1. Austin, J. L. (1962). *How to Do Things with Words*. Oxford University Press.
2. Bakhtin, M. M. (1981). *The Dialogic Imagination: Four Essays* (C. Emerson & M. Holquist, Eds.). University of Texas Press.
3. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (Vol. 33, pp. 1877–1901). Curran Associates, Inc.
4. Buchinger, E., & Scott, B. (2010). Comparing Conceptions of Learning: Pask and Luhmann. *Constructivist Foundations*, *5*, 109–120.
5. Craik, K. J. W. (1952). *The Nature of Explanation*. Cambridge University Press.
6. Di Paolo, E. A., Cuffari, E. C., & De Jaegher, H. (2018). *Linguistic Bodies: The Continuity between Life and Language*. The MIT Press.
7. Dubberly, H., & Pangaro, P. (2019). Cybernetics and Design: Conversations for Action. In T. Fischer & C. Herr (Eds.), *Design Cybernetics. Design Research Foundations* (pp. 85–99). Springer, Cham.
8. Franceschelli, G., & Musolesi, M. (2023). *On the Creativity of Large Language Models*.
9. Johnson-Laird, P. N. (1983). *Mental Models: Towards a cognitive science of language, inference, and consciousness*. Harvard University Press.

10. Mani, K. R. S., & Johnson-Laird, P. N. (1982). The mental representation of spatial descriptions. *Memory & Cognition*, *10*, 181–187.
11. Negroponte, N. (1970). *The Architecture Machine: Toward a More Human Environment*. The MIT Press.
12. Negroponte, N., & Pask, G. (1976). Aspects of Machine Intelligence. In *Soft Architecture Machines*. The MIT Press.
13. OpenAI. (2023a). *ChatGPT*. https://openai.com
14. OpenAI. (2023b, April 4). *Cheat Sheet: Mastering Temperature and Top_p in ChatGPT API - API - OpenAI Developer Forum*. OpenAI Documentation. https://community.openai.com/t/cheat-sheet-mastering-temperature-and-top-p-in-chatgpt-api/172683
15. Pask, G. (1975). *Conversation, Cognition and Learning: A Cybernetic Theory and Methodology*. Elsevier.
16. Searle, J. R. (1969). *Speech Acts: An essay in the philosophy of language*. Cambridge University Press.
17. Vygotsky, L. (1978). *Mind in Society*. Harvard University Press.
18. Wei, J., Wang, X., Chi, E., Le, Q., & Zhou, D. (2023). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. https://arxiv.org/abs/2201.11903
19. Werner, L. C. (2019). Gordon Pask and the Origins of Design Cybernetics. In T. Fischer & C. Herr (Eds.), *Design Cybernetics. Design Research Foundations* (pp. 65–84). Springer, Cham.
20. Wiener, N. (1960). Some Moral and Technical Consequences of Automation. *Science*, *131*(3410), 1355–1358. https://doi.org/10.1126/science.131.3410.135

Author

Steve Battle, Dr, University of the West of England, steve.battle@uwe.ac.uk