Pre-Generative Conversational AI

Staffan Larsson

Gothenburg University and Talkamatic AB

Gothenburg, Sweden

staffan.larsson@ling.gu.se, staffan@talkamatic.se

Abstract

LLMs such as ChatGPT have raised expectations on Conversational AI (CAI) applications, yet deployment is often hindered by controllability problems. This paper, to be accompanied by a live demo, describes Pre-Generative Conversational AI (PGCAI) and its implementation in Talkamatic Dialog Studio, a tool suite for creating high-quality controllable conversational AI application without the need for coding, prompting or manual dialogue building.

1 Introduction

Generative AI in the form of Large Language Models such as ChatGPT is currently re-shaping the conversational AI landscape, and is generally taken to enable a multitude of practical Conversational AI applications in many different areas, including customer service, education, and more.

However, many companies and organisations are also hesitant when it comes to using an LLM-driven conversational agent to (for example) represent them on their website, or engage in one-to-one educational dialogue with children in schools. One reason for this is a host of well-known problems deriving from the overall problem of controlling the behaviour of LLMs. This may result in generating outputs that do not adhere to the desired agent behaviour (Kann et al., 2022).

For many applications of LLMs, such problems can be handled by manually checking the output of the LLM before using it (e.g. publishing a text or sending an email). However, in conversational AI applications, this is typically not an option, as the LLM interacts directly in real time with users.

This paper proposes a solution to this problem in the form of "Pre-Generative Conversational AI". Instead of letting the user talk directly to a generative AI, with the risks that entails, we instead use generative AI to generate dialogues *before* they are published. At runtime, the dialogue can be handled without using LLMs at all, or using them only for limited tasks such as NLU.

In essence, PGCAI enables using our normal preferred way of working with LLMs (generate-curate-publish) also for conversational AI.

2 Key components

PGCAI has to three key components: a dialogue generator, a dialogue editing tool, and a flexible but controllable dialogue manager.

2.1 Dialogue generator

The dialogue generator uses LLMs to create dialogues based on some content. Of course, there are many types of dialogue one could have about some content. Hence, the dialogue generator relies on distinguishing different dialogue types, or *genres* (Larsson, 2002; Ginzburg and Wong, 2024). Examples of such genreas are educational, instructional, question-answering and negotiative dialogue.

The task of the dialogue generator is to take some content (a text, a database or something else) a specification of a dialogue genre, and produce a dialogue blueprint which can then be used by the dialogue manager to engage in a flexible dialogue. For each type of dialogue, it uses genre-specific prompts to produce dialogues of the type selected by the dialogue designer.

2.2 Dialogue curation tool

Since PGCAI does not require designing or implementing a dialogue (in the form of code or using a GUI), nor requires any prompt writing, we do not use the term "dialogue designer". Instead, the role of the human in building a dialogue application is to *curate* the dialogue, in the sense of taking an existing dialogue blueprint and adapting and perfecting it for the precise use it will be put to. To aid in this process, a dialogue curation tool is needed.

After a dialogue has been generated, it can immediately be tested by interacting with it. If the curator is unhappy with some aspect of the dialogue, they can go in and inspect and edit the dialogue blueprint. The precise structure of this blueprint will depend on the dialogue genre. For question-answering dialogue, the main component is a list of question-answer pairs. For education dialogue, it is a pedagogical interaction consisting mainly of questions of various kinds (right/wrong questions asking about information offered explicitly in the text, or requiring some inference on the part of the user, more open questions asking the user to reflect, and more). Other elements are also present, such as a list of potentially difficult words that the system can explain if needed. For other types of dialogue, other structures are available for curation. Importantly, these structures are quite simple and editing them does not require any deep technical understanding of conversational AI or even of human dialogue. However, genre-specific competence can often be useful, such as pedagogical skills in the case of educational dialogue.

2.3 Flexible dialogue management

LLMs are widely recognised as going considerably beyond the state of the art when it comes to NLU. For this reason, we allow for using LLMs to take care of NLU even when not using them to generate responses to the user. A similar approach is taken in Rasa (Bocklisch et al., 2024). Talkamatic Studio allows the dialogue designer to decide what NLU to use, offering LLMs as options but also non-LLM technologies.

LLMs are also quite adept at handling many different kinds of dialogue in a flexible way, meaning that they often respond appropriately to less expected or less routine user behaviours. The success of PGCAI depends crucially on the ability of the system to achieve dialogue behaviour on par with or surpassing an LLM. Hence, we need to achieve a high level of flexibility in PGCAI, despite the fact that the dialogue blueprints are not generated at runtime. This poses considerable challenges for the dialogue manager. Talkamatic have developed the Talkamatic Dialogue Manager (Larsson and Berman, 2016) which supports a wide (and growing) variety of conversational behaviours across several dialogue genres, including the ones mentioned above.

TDM is based on the Information State Update approach to dialogue management, and more specifically Issue-Based Dialogue Management (Larsson, 2002). As part of a series of research projects and later in Talkamatic, TDM has been gradually extended to cover an increasing range of dialogue behaviours and dialogue genres.

3 Talkamatic Studio

Talkamatic Studio[1] is a comprehensive software service offering all the components needed for PG-CAI. It offers a dialogue generator, a dialogue curation tool, a runtime frontend and backend using TDM for dialogue management, an LLM control panel, and a dialogue analytics tool.

4 Related work

Of course, the control problem for LLMs is not new and a lot of work is being done to address it. The absolute majority of methods for dealing with this problem is of the "guardrails" type. In LLM-based Conversational AI, however, the user is still interaction with an LLM at runtime, and it is difficult or impossible to guarantee that guardrails always work. Ayyamperumal and Ge (2024) discuss various guardrail approaches such as layered protection models, system prompts, Retrieval-Augmented Generation (RAG) architectures and bias mitigation, and observe that "[c]rucial challenges remain in implementing these guardrails." Xu et al. (2024) show that hallucination is not just a temporary glitch, but are in fact inevitable in LLMs.

We believe that in many applications, including using Conversational AI agents for education in schools and to represent companies and organisations, there will be a strong preference for zero risk solutions, i.e. solutions that can *guarantee* there will be no bad output from the system.

5 Conclusion and future work

We have presented Pre-Generative Conversational AI and its implementation in Talkamatic Studio. This approach and implementation addresses a central problem with using LLMs for Conversational AI - the lack of control. To the best of our knowledge, Talkamatic Studio is the only solution that combines dialogues generated by LLMs, control (including complete control with no LLM output generation at runtime), curation (putting a human in the loop), and flexible dialogue across several dialogue genres, going beyond form-filling dialogue.

---

[1] https://talkamatic.se

References

Suriya Ganesh Ayyamperumal and Limin Ge. 2024. Current state of llm risks and ai guardrails. *arXiv preprint arXiv:2406.12934*.

Tom Bocklisch, Thomas Werkmeister, Daksh Varshneya, and Alan Nichol. 2024. Task-oriented dialogue with in-context learning. *arXiv preprint arXiv:2402.12234*.

Jonathan Ginzburg and Kwong-Cheong Wong. 2024. Language games and their types. *Linguistics and Philosophy*, 47(1):149–189.

Katharina Kann, Abteen Ebrahimi, Joewie Koh, Shiran Dudy, and Alessandro Roncone. 2022. Open-domain dialogue generation: What we can do, cannot do, and should do next. In *Proceedings of the 4th Workshop on NLP for Conversational AI*. Association for Computational Linguistics.

Staffan Larsson. 2002. *Issue-Based Dialogue Management*. Ph.D. thesis, Department of Linguistics, Goteborg University.

Staffan Larsson and Alexander Berman. 2016. Domain-specific and general syntax and semantics in the talkamatic dialogue manager. *Empirical Issues in Syntax and Semantics*, 11:91–110.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.