Introducing the BPMN-Chatbot for Efficient LLM-Based Process Modeling

Julius Köpke[1,*], Aya Safan[1]

[1]*University of Klagenfurt, Department of Informatics Systems, Universitätsstraße 65-67, 9020 Klagenfurt am Wörthersee, Austria*
https://www.aau.at/en/isys/ics

Abstract

Generative AI and Large Language Models (LLMs) have recently gained enormous interest in the BPM domain. Various research groups have experimented with extracting process information from text using LLMs. In this paper, we introduce a publicly available web-based tool for the automatic and interactive generation of BPMN process models using text or voice input. In contrast to existing tools, it is heavily optimized for generating high-quality models while keeping the costs (number of tokens) low. In our experiments, the tool achieved higher average correctness while using up to 94% fewer tokens compared to an alternative tool.

Keywords

Large Language Models, LLM, Conversational Process Modeling

1. Introduction

Recently, the BPM community has started experimenting with LLMs to extract process information and to generate process models from natural text with approaches such as [1, 2, 3, 4, 5]. These works indicate promising capabilities of LLMs for this task.

With *ProMoAI* [6, 7], there is also an online tool[1] available in the literature. The tool not only allows the initial generation of a process model from text. It also offers a feedback loop for refining the generated model. *ProMoAI* indicates a high potential of LLMs for this task. However, the applied prompting strategy and intermediate formats impose significantly high usage fees. When we first tried the tool, we generated a 3-step process with two iterations of the feedback loop. This small experiment costed 0.8 USD in OpenAI API fees using GPT-4. While the current GPT-4o model is more cost-effective, we argue that such systems should be optimized to efficiently use LLM resources to reduce costs while maintaining high-quality output. We propose that an optimized system can potentially democratize process modeling or at least make process modeling accessible to a broader audience. We have, therefore, developed our own approach in [8] that focuses on the modeling costs in terms of the required number of tokens.

2. Introducing the BPMN-Chatbot

The *BPMN-Chatbot* implements our efficient approach for LLM-based process modeling in [8]. It is publicly available on the tool's homepage[2], which also includes a video demonstration and further resources.

---

[1]https://promoai.streamlit.app/
[2]https://isys.uni-klu.ac.at/pubserv/BPMN-Chatbot/

2.1. Usage Scenario

The *BPMN-Chatbot* mimics the look and feel of a classical messenger application. However, the output is rendered in the form of BPMN process diagrams. A screenshot of the tool is shown in Fig. 1. First, the user provides a process description using text or voice input (see Fig.1 [A]). The tool then generates a business process and shows it graphically. In an optional feedback loop, the user can provide feedback on the model and obtain an updated version. During the feedback loop, the user can navigate between different model versions using arrow symbols to provide feedback on a specific version (see Fig. 1 [B]). Finally, the generated model can be downloaded as a BPMN-XML File (see Fig.1 [C]). For evaluation purposes, the tool also includes a survey component for eliciting user feedback after the modeling session.

Example   Fig. 1 shows a screenshot of the tool after one iteration of the feedback loop. In particular, the user first asked the BPMN-Chatbot to create a process model for the following scenario: *When an order is received, we first check it, then we collect the items, and finally, we send the package to the customer.* The system then generated a purely sequential model. In the next iteration, the user asked the tool to refine the model with the comment: *If the items are not on stock, we reject the order.* The resulting process shows the correct usage of a gateway to address the comment.
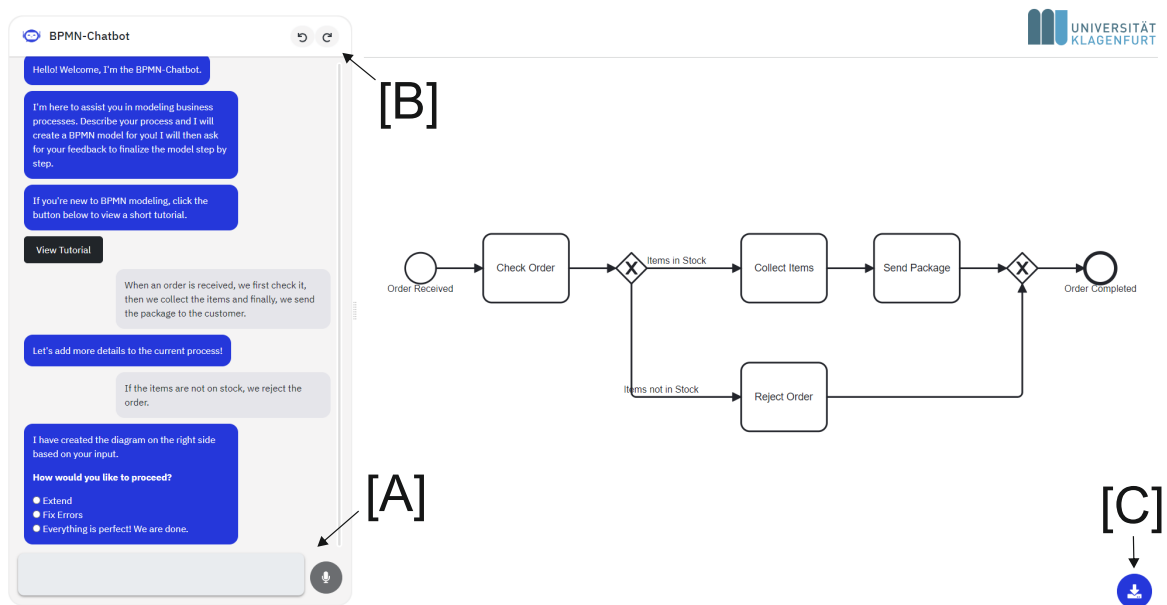

Figure 1: Screenshot of the tool after one process model refinement.

2.2. Architecture

The tool is implemented in the form of a React single-page web application. An architectural sketch is shown in Fig. 2. We focus here on the core components *Prompt Generation* and *Model2Model Translation*. The prompt generation component creates messages that are sent to the LLM API (currently OpenAI). The *Model2Model Translation* component is used to transform an intermediate process format returned by the LLM to standard BPMN that is then visualized via a rendering component that uses the bpmn.js[3] library.
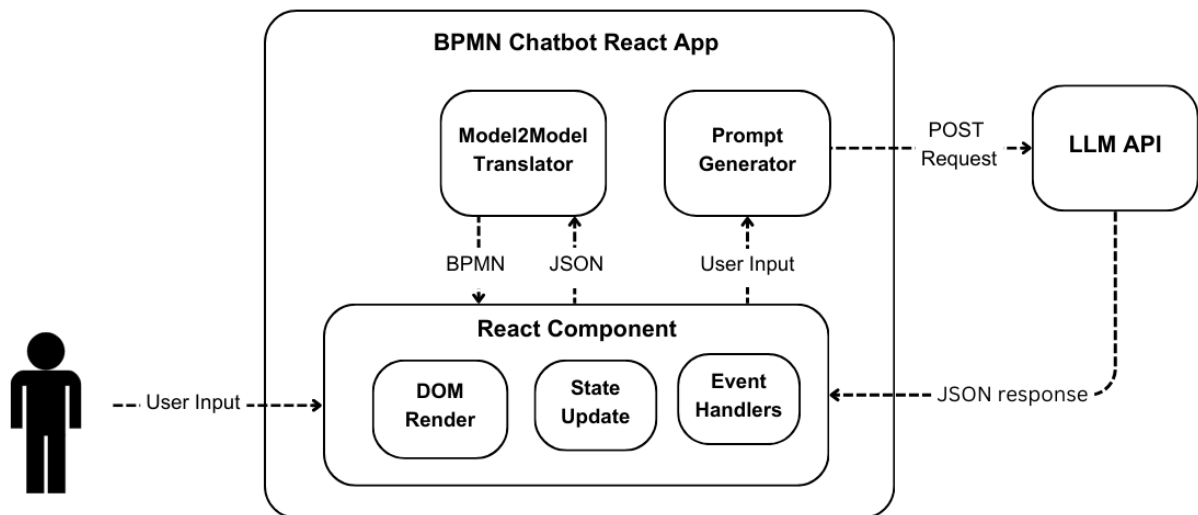

Figure 2: BPMN Chatbot prototype architecture.

2.2.1. Prompt Generation

This is the core component of the tool, which is responsible for the low numbers of required tokens. To keep the number of tokens low, we aim for prompts with minimal overhead on top of the user input. To optimize the feedback loop costs, we only send the feedback comment and the referenced process model to the LLM. This approach is further enhanced by using an efficient intermediate JSON representation [8] for processes returned from (and sent to) the LLM. It covers the core set of BPMN elements (also used by other approaches [6, 7]), providing block-structured process models

We have iteratively optimized our prompt in preliminary experiments on process descriptions disjoint from our evaluation data sets. This process has led to the following prompt shown in Listing 1.

> You are a business process modeling expert. I will provide you with a textual description of a business process. Generate a JSON model for the process.
> Analyze and identify key elements:
> 1. Start and end events.
> 2. Tasks and their sequence.
> 3. Gateways (xor or parallel) and an array of "branches" containing tasks. For xor gateways, there is a condition for the decision point and each branch has a condition label.
> 4. Loops: involve repeating tasks until a specific condition is met.
> Nested structure: The schema uses nested structures within gateways to represent branching paths.
> Order matters: The order of elements in the "process" array defines the execution sequence.
> When analyzing the process description, identify opportunities to model tasks as parallel whenever possible for optimization (if it does not contradict the user intended sequence).
> Use clear names for labels and conditions.
> Aim for granular detail (e.g., instead of "Task 1: Action 1 and Action 2", use "Task 1: Action 1" and "Task 2: Action 2").
> Sometimes you will be given a previous JSON solution with user instructions to edit.

Listing 1: Instructions prompt for process generation. [8]

2.2.2. Model Transformation: Intermediate Model to BPMN

This component is responsible for transforming the intermediate JSON process models generated by the LLM to standard BPMN XML models. On the one hand, this is a straightforward mapping of model elements of our nested intermediate process format to the graph-structured BPMN format. However, since the intermediate format does not contain any graphical representation information, the coordinates and size of all BPMN shapes and edges are calculated and inserted by this module. This step is highly relevant for providing a good user experience in the feedback loop: A small change in the process model should also only lead to a small change in the graphical representation. Thanks to the block-structured input processes, we can compute a planar graph that deterministically includes the coordinates of the elements.

3. Maturity of the Tool

The tool has been evaluated in [8]. The evaluation data is available online[4]. Two experiments were conducted:

1. A comparison against ProMoAI [7] and a compatible prompting pattern of [4] assessing the correctness and the required numbers of tokens.
2. An evaluation with visitors of a national science fair in Austria "Lange Nacht der Forschung" (Klagenfurt, May 24th., 2024).

Experiment 1 was based on a subset of 7 processes of the PET Dataset [9]. In the experiment, the tool provided a substantial reduction of tokens of up to 94% compared to ProMoAI and showed correctness values of on average, 95% compared to 86% of the best competitor.

In the second experiment at the science fair, visitors used the tool to model processes of their own choice; 76 process models were created, and 40 visitors participated in a preliminary technology acceptance test. The experiments showed strong indications of the tool's overall usefulness and the high quality of the generated models in the feedback loop.

4. Conclusion and Future Works

With the BPMN-Chatbot, we have presented a highly efficient tool for LLM-based conversational process modeling. It allows users to interactively design processes using text or voice input. It provides substantial cost reductions (up to 94%) while achieving even better levels of correctness compared to an alternative tool and a prompting strategy from the literature. We argue that such tools may drastically change the way how processes are created in the near future.

For future works, we intend to publish the tool as an open-source software that allows users to easily integrate their own prompting strategies and intermediate formats via plugins for evaluating their own approaches with users. We argue that user tests are strongly needed to evaluate the feedback loop's capabilities. We would like to take the opportunity to gather feedback from experts at the demo session. Furthermore, we plan to extend the tool to support more elements of the BPMN Meta-Model, such as pools, lanes, and data objects.

References

[1] H. Fill, P. Fettke, J. Köpke, Conceptual modeling and large language models: Impressions from first experiments with chatgpt, Enterp. Model. Inf. Syst. Archit. Int. J. Concept. Model. 18 (2023) 3. doi:`10.18417/EMISA.18.3`.

---

[4]https://github.com/BPMN-Chatbot/bpmn-chatbot-archive

[2] M. Grohs, L. Abb, N. Elsayed, J.-R. Rehse, Large language models can accomplish business process management tasks, in: J. De Weerdt, L. Pufahl (Eds.), Business Process Management Workshops, Springer Nature Switzerland, Cham, 2024, pp. 453–465. doi:10.1007/978-3-031-50974-2_34.

[3] N. Klievtsova, J.-V. Benzin, T. Kampik, J. Mangler, S. Rinderle-Ma, Conversational process modelling: state of the art, applications, and implications in practice, in: International Conference on Business Process Management, Springer, 2023, pp. 319–336. doi:10.1007/978-3-031-41623-1_19.

[4] N. Klievtsova, J.-V. Benzin, T. Kampik, J. Mangler, S. Rinderle-Ma, Conversational process modeling: Can generative ai empower domain experts in creating and redesigning process models?, 2024. URL: https://arxiv.org/abs/2304.11065v2. arXiv:2304.11065.

[5] M. Forell, S. Schüler, Modeling meets large language models, in: Modellierung 2024 Satellite Events, 2024. doi:10.18420/modellierung2024-ws-003.

[6] H. Kourani, A. Berti, D. Schuster, W. M. P. van der Aalst, Process modeling with large language models, 2024. URL: https://arxiv.org/pdf/2403.07541. arXiv:2403.07541.

[7] H. Kourani, A. Berti, D. Schuster, W. M. P. van der Aalst, Promoai: Process modeling with generative ai, 2024. arXiv:2403.04327.

[8] J. Köpke, A. Safan, Efficient llm-based conversational process modeling, in: NLP4BPM Workshop at BPM 2024 (to appear in Business Process Management Workshops 2024), Krakow, Poland, 2024. Preprint available at https://sites.google.com/view/nlp4bpm2024.

[9] P. Bellan, H. van der Aa, M. Dragoni, C. Ghidini, S. P. Ponzetto, Pet: An annotated dataset for process extraction from natural language text tasks, in: C. Cabanillas, N. F. Garmann-Johnsen, A. Koschmider (Eds.), Business Process Management Workshops, Springer International Publishing, Cham, 2023, pp. 315–321. doi:10.1007/978-3-031-25383-6_23.