Enhancing Pipeline-Based Conversational Agents with Large Language Models

Mina Foosherian
BIBA - Bremer Institut für
Produktion und Logistik GmbH
Hochschulring 20
28359, Bremen, Germany
fos@biba.uni-bremen.de

Hendrik Purwins
Accenture GmbH
Center of Excellence for LLM
Balanstrasse 73
81541, Munich, Germany
hendrik.purwins@accenture.com

Purna Rathnayake
University of Bremen
Faculty of Physics and Electrical Engineering
Otto-Hahn-Allee 1
28359, Bremen, Germany
purnasam@uni-bremen.de

Touhidul Alam
Accenture GmbH
Liquid Studio
Balanstrasse 73
81541, Munich, Germany
touhidul.alam@accenture.com

Rui Teimao*
Lloyds Banking Group
Conversational Banking Lab
125 London Wall, Barbican
London EC2Y 5BL, England
rui.teimao@lloydsbanking.com

Klaus-Dieter Thoben
University of Bremen
Faculty of Production Engineering
Badgasteiner Str. 1
28359, Bremen, Germany
tho@biba.uni-bremen.de

Abstract

The latest advancements in AI and deep learning have led to a breakthrough in large language model (LLM)-based agents such as GPT-4. However, many commercial conversational agent development tools are pipeline-based and have limitations in holding a human-like conversation. This paper investigates the capabilities of LLMs to enhance pipeline-based conversational agents during two phases: 1) in the design and development phase and 2) during operations. In 1) LLMs can aid in generating training data, extracting entities and synonyms, localization, and persona design. In 2) LLMs can assist in contextualization, intent classification to prevent conversational breakdown and handle out-of-scope questions, auto-correcting utterances, rephrasing responses, formulating disambiguation questions, summarization, and enabling closed question-answering capabilities. We conducted informal experiments with GPT-4 in the private banking domain to demonstrate the scenarios above with a practical example. Companies may be hesitant to replace their pipeline-based agents with LLMs entirely due to privacy concerns and the need for deep integration within their existing ecosystems. A hybrid approach in which LLMs' are integrated into the pipeline-based agents allows them to save time and costs of building and running agents by capitalizing on the capabilities of LLMs while retaining the integration and privacy safeguards of their existing systems.

1 Introduction

The field of conversational artificial intelligence (CAI) has experienced significant advances in recent years, with the emergence of both commercial and open-source CAI development platforms such as Google Dialogflow, Amazon's Alexa Skills Kit, Cognigy, and Rasa, as well as the more recent large language model (LLM)-based conversational agents (CA) like ChatGPT.

CAs can be text-based agents (Chatbots), Voice-User interfaces (VUI), or embodied-dialog Agents (EDA) (Harms et al., 2019) and generally aim to

[0] Author was at Accenture during the time of the research

Proceedings of the 1st Workshop on Taming Large Language Models: Controllability in the era of Interactive Assistants!, pages 56–67
September 12, 2023

replace or empower humans through natural language interaction.

CAs can be pipeline-based or end-to-end (Chen et al., 2017). In pipeline-based CAs, the natural language understanding (NLU) component processes the user's message sequentially to identify their goal (intent recognition), and extract information pieces called entities. The dialog management component tracks the dialog state and decides on the next action based on the current state. Finally, the natural language generation (NLG) component builds and returns the response. The CAs "intelligence" relies on the agent's training data and internal logic used to create its NLU and dialog management models (Harms et al., 2019).

The end-to-end CAs rely on dialog models trained with large training datasets (Chen et al., 2017). These models learned hidden relations between input and output utterances, effectively avoiding that developers create interim representations (Dinan et al., 2021). A downside is that the necessity of larger datasets makes end-to-end approaches less applicable in domains such as manufacturing, where developers cannot derive training data from existing human-human conversations. End-to-end CAs also bear substantial safety issues, such as generating offensive language and responding inappropriately to offensive content or in safety-critical situations (Dinan et al., 2021). Combinations of pipeline-based and end-to-end approaches are also feasible. Rasa Open Source, for instance, already supports both (Rasa, 2023b,a).

This article uses the term LLM to refer to language models trained with an end-to-end approach and an amount of high-quality training data that only a few organizations in the world can afford e.g., GPT-3 and GPT-4 by OpenAI, BERT, LaMDA, and PaLM by Google, LLaMA by Meta AI. Such models can, for instance, possess emergent abilities, be hard to steer, and humans will likely have difficulties interpreting how they work (Bowman, 2023).

In this article, we demonstrate how LLMs can expand the capabilities of pipeline-based CAs without removing the pipeline altogether. The impact of LLMs helps the pipeline-based CAs in generating training data for intent classification, the identification of domain-specific entities and synonyms, requirement characterization for the agent and its personalization and localization, among others. During deployment, LLMs can provide auto-correction to user input, handle context switching and out-of-scope questions, introduce response variability, create conversation summarizations and perform closed Question-Answering (Q&A).

2 The State of Conversational Agents

Broadly, CAs can be categorized into two main categories based on the design methodology employed: Pipeline methods and End-to-end methods (Chen et al., 2017). Agents that are developed using conversational AI platforms (task-oriented CAs), such as Rasa, Google Dialogflow, Cognigy, and IBM Watson, fall into the first category. LLM-based CAs such as ChatGPT can be identified as CAs belonging to the second category. While explicit architectural components can be identified in the pipeline-based CAs, such clear distinctions cannot be identified in end-to-end CAs.

2.1 Pipeline-Based Conversational Agents

2.1.1 Architecture

In the case of task-oriented CAs, the components that can be explicitly identified are NLU, dialog management, and NLG. A typical architecture of such a CA is shown in Figure 1. For NLU and NLG, pipeline-based CAs would traditionally use machine learning-based and template-based approaches, respectively. The dialog management component can be handcrafted, probabilistic, or hybrid. Most of the commercial frameworks and low-code platforms to create task-oriented CAs, such as Google Dialogflow, Cognigy, and IBM Watson, are pipeline-based and use handcrafted rules for dialog management. They are more reliable but less human-like. However, CAs using the probabilistic approach, such as ChatterBot, which are often used for open-domain CAs, create opposite results. Among different platforms, Rasa uses a hybrid approach for the dialog management component (Harms et al., 2019).

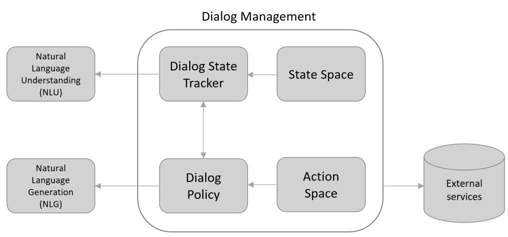

Figure 1: Architecture of a pipeline-based CA. Based on (Harms et al., 2019; Brabra et al., 2022)

2.1.2 Limitations

Conversational breakdown is a common issue during a conversation with a pipeline-based CA, indicating that the agent did not correctly understand the user's utterance or responded inadequately to the user's request (Moore and Arar, 2019; Følstad and Taylor, 2020). Conversational breakdowns can lead to frustration, disappointment, and dissatisfaction (Bentley et al., 2018; Cowan et al., 2017; Luger and Sellen, 2016) if left unaddressed. In pipeline-based CAs, these breakdowns occur for various reasons, such as errors during intent and slot recognition, errors during task fulfillment, errors in generating the response, and users' lack of familiarity with a chatbot's intents (Li et al., 2020). In addition to conversational breakdowns, most of the commercial CAs cannot handle complex queries, lack emotional intelligence, and have limited domain knowledge (Luo et al., 2022).

Pipeline-based CAs are also limited regarding the effort in configuring them and how to operate in real-time conversations. In terms of configuration, intent classes, domain entities, and synonym lists need to be created a priori. It requires a certain amount of depth of domain knowledge to come up with suitable notions. The agent's personality and the power dynamics between the agent and user must be defined and expressed by manually creating individual utterances for the bot. The localization to various language varieties requires a significant amount of rework, in particular when it comes to scarcely-supported dialects.

2.2 Large Language Models

The advancement of Language Models (LM) in NLP has driven significant progress. In general, the LM aims to predict the next word of a sentence given the current context. With the improvement of research, the concept of the LM has evolved in different stages. From a statistical LM (Jelinek, 1998), to predict the next word based on Markov assumption, it further progressed through a distributed word representation learning (Word2Vec), which initiated the usage of language model beyond word sequence (Mikolov et al., 2013). Context-aware pre-trained language models (Peters et al., 2018; Devlin et al., 2018) is one of the early adopters of the modern language model, which sets the paradigm of performing a fine-tuning on any of these pre-trained models on the downstream task and raised the performance achievements on many NLP tasks. One of these models, BERT (Devlin et al., 2018), is based on a parallelizable Transformer based network (Vaswani et al., 2023) with a self-attention mechanism, that begins a new era for future Language models. With the scaling of the model architecture and training data, there has been a rise of many LMs which are named Large Language Model or LLM (e.g., Generative Pretrained Transformer or GPT-series, Pathways Language Model or PaLM-series, etc.) (Brown et al., 2020; Chowdhery et al., 2022; OpenAI, 2023; Anil et al., 2023). One of the prominent differences has been seen in the emergence abilities (Wei et al., 2022) where these models could do a series of complex tasks given some specific prompt in a zero-shot or few-shot learning mechanism. Recent models in NLP are based on LLMs and one of the prominent LLM-based conversational agents has been ChatGPT. It is based on the Instruction tuned GPT models (InstructGPT), fine-tuned with Reinforcement learning from human feedback on dialogue data (Ouyang et al., 2022). An early experiment (Bubeck et al., 2023) from the OpenAI's latest release GPT-4 model, has shown the potential capabilities of the LLMs in different domains that denotes that GPT-4's performance is strikingly close to the human-intelligence level and it is far beyond next-word-level prediction. With the emergence of these LLMs with human-like conversational agents, the evolution of Chatbots has advanced to a different level.

2.2.1 LLM-based Conversational Agents

LLM-based CAs like ChatGPT are trained in a similar way to InstructGPT, specifically optimized for dialogue. Human-generated dialogue data is collected, playing both a Human and an AI role. They used a three-step process: First, they collected a dataset of human-written demonstrations of the desired model behavior and used it to fine-tune GPT-3 using supervised learning. Next, they collected a dataset of rankings of model outputs and used it to fine-tune the supervised model further using reinforcement learning from human feedback. Finally, they evaluate their models by having human labelers rate the quality of model outputs on a test set. This methodology allows the authors to train language models that are better aligned with user intent and more natural interaction with the inclusion of human-generated data. Recent studies (Longpre et al., 2023) show the effectiveness of instruct tuning on different LLM models to improve performance on different prompt settings (zero-shot/few-shot settings). One of the recent LLM, Alpaca (Taori et al., 2023), which is built upon self-instruct (Wang et al., 2022) methods on Llama model (Touvron et al., 2023) shows a 7B parameters LLM can demonstrate high potential to compete with larger GPT-like models.

2.2.2 Limitations and Risks

Despite many benefits of LLMs, they have several limitations (OpenAI, 2023). For instance, their responses are not reliable (they "hallucinate") (Bang et al., 2023; Zhao et al., 2023). Models like ChatGPT still can produce faithful but nonsensical responses when viewed in the light of the common knowledge in a particular area (Alkaissi and Mcfarlane, 2023). LLMs also have long training times and require huge computation resources; thus, they are not easily obtainable with the latest event knowledge. They do not learn from experience as their context window is limited. For certain task-oriented domains, for example, cybersecurity, the models are unable to assess properly due to context limitations. There are also risks regarding the output of LLM-based models, as they could contain harmful advice, buggy code, or inaccurate information. Like other deep learning models, LLM-based models (Brown et al., 2020) are difficult to interpret due to their complex architecture. Also, their ability to make accurate predictions on new inputs cannot be relied upon, as evidenced by their much higher performance variance than humans on standard benchmarks.

2.3 Integrating LLM into Pipeline-based Conversational Agents

As of May 2023, we have found that the involvement of LLMs in pipeline-based CA platforms is mainly limited to NLU and training data generation. For example, Cognigy (Cognigy, 2023), with the help of a third-party Generative AI provider, allows users to generate training data, including intent utterances, lexicons, and flows with pre-configured nodes, Rephrasing bot outputs and completing texts. Even though Cognigy offers a conversation option using generative AI, it is only intended to be used as a preview feature. In another case, (Rasa, 2023a) recently announced the integration of LLMs in their chatbot framework with a new component called IntentlessPolicy. They explain a) how an LLM-based system can take advantage of multiple FAQs without setting up intents for each question, b) how user meaning can be understood in multiple turns of dialogue, and c) how out-of-scope messages can be understandable from the context. They also show that this can be generalized from very little data in a few-shot learning mechanism. They further emphasize that IntentlessPolicy complements intents, rules, stories, and forms. This hybrid approach will better equip with engaging interaction with the user.

3 LLMs to Overcome Limitations of Pipeline-based CAs

Despite the various frameworks for building pipeline-based CAs, it still requires substantial time and expertise to design and develop successful CAs. Related tasks concern the design of high-quality training utterances, the definition of intents and consistent and accurate named entities, the selection of domain-specific synonyms, and the localization of training data and responses. Besides, designers must modulate, for instance, training data, dialog management rules, and pre-defined responses to represent desired assistant traits (e.g., client orientation) or personas. We assume that the strengths of LLMs in processing natural language from different countries and domains can substantially shorten at least the time and potentially also the expertise needed to build pipeline-based CAs. Their capability to generate responses matching the style of a generated persona or mimicking an actual person's style could provide new techniques to create attractive CAs.

A second area for improvement is the robustness of a CA at run-time, i.e., when it interacts with a user. Often, pipeline-based CAs produce repetitive responses (robust but less attractive) or experience conversational breakdown because users switched contexts (not robust). In addition, pipeline-based CAs' narrow domain knowledge provokes out-of-scope answers due to smaller training data and limited responses. All of the situations above lower the user's satisfaction and could encourage them to give up on the agent. We assume that LLMs' extensive general and domain knowledge, coupled with their capability of generating attractive and diverse natural language texts, has the potential to achieve more robust and attractive CAs.

We conclude that LLMs have the capability to enhance pipeline-based CAs during the design and development phase (*delivery accelerator*) and a dialogue with a user (*real-time booster*). In contrast to relying on LLMs only, this hybrid approach is helpful because the pipeline-based approach grants the CA designer more control and transparency over the agent's behavior. The former is critical to counter, for instance, hallucinations, while the latter helps trace and potentially explain unexpected or unwanted behavior.

4 GPT-4 Experiments

To demonstrate the impact of LLMs on CAs, we conducted a series of experiments with GPT-4. The example scenario is a chat agent serving as a client advisor for private banking. A supporting document contains the exact prompts and replies in the conducted experiments.

Parameters We used the Azure Open AI playground with the default parameters for our experiments: Max Response: 800, Temperature: 0.7, Top P: 0.95 Frequency penalty: 0, Presence penalty: 0, Deployment: GPT-4, Past messages included: 10, Max tokens: 8192. The temperature value of 0.7 means that generated responses are not deterministic, i.e., the exact response may vary during reproduction. To keep this article short, we sometimes shorten the actual prompts and answers by inserting an ellipsis.

4.1 LLM as Delivery Accelerator

LLM, as a delivery accelerator, involves scenarios to assist developers and designers in building and refining the CA. This can include generating training data, creating lists of entities and synonyms, designing personas to guide the agent's responses, and localizing the agent for different languages and cultures. These tasks can be time-consuming and require significant expertise, so automating them with generative models can save developers time and resources. In Table 1, we show examples of how LLM can be used in the cases mentioned above for our scenario. The following sections review each development aspect's limitations in pipeline-based CAs and demonstrate how LLMs could address them.

Creating intents lists One of the initial steps in designing a pipeline-based CA is to define and identify possible user inquiries or intents. To create a comprehensive list of intents, designers require approaches such as analyzing existing data, sessions with domain experts, and user research. However, LLMs can provide valuable assistance to the designers to gain general insight. We test GPT-4's ability to identify customer intents within a specific industry. We provided the following prompt:

`For designing a chatbot, give me a list of 10 most prominent intents in a conversation about banking between a client and an agent.`

The first five results provided by GPT-4 (omitting the explanations):

```
1. Check account balance ...
2. View recent transactions ...
3. Transfer funds between accounts ...
4. Pay a bill or set up recurring payments ...
5. Update personal information ...
```

We observe that all of these are common consumer banking interactions that can trigger contact with banking customer service.

Generating training utterances for intent classification Writing high-quality training data is a time-consuming task. GPT-4 is capable of helping CA designers with this matter. In A.1, we provide ten examples generated by GPT-4. We observe that all generated examples are valid for the intent cancel_account with various phrasings. However, further prompt tuning would be required to increase variety in length and formality. Generating training data using LLMs and incorporating those into the training pipeline would still require human feedback in the loop to avoid incorrect, biased, or inappropriate training data.

Creating lists of named entities We test GPT-4's ability to identify entities relevant to a specific industry. We provided GPT-4 with the following prompt:

`For designing a chatbot in the finance domain, give me a list of relevant named entities that the NLP back-end of the chatbot should be able to extract.`

GPT-4 returned the following entities (with explanations omitted here):

`account numbers, transaction amounts, dates, currency, transaction IDs, percentage rate, financial product names, contact details, company names, bill or invoice numbers, tax-related terms, account types, payment methods, financial goals and financial institutions.`

We observe that these are useful terms, although there is an overlap between companies and finan-

Benefit	Example
Creating intents list	check account balance, transfer funds, pay bill
Generating training utterances for intent classification	Provide a list of different wordings with the intent of canceling an account
Creating named entities	account numbers, transaction amounts, date, currency
Creating synonyms lists	insolvent, bankrupt, penniless, broke
Traits of good agents	financial knowledge, interpersonal skills, client orientation, ethical standards
Persona-based responses	Dr. Jennifer Melfi (The Sopranos), simple English.
Localization	German, Swiss German, Spanish, French

Table 1: Examples of LLMs as a delivery accelerator. A pipeline-based CA for private banking is configured with the help of GPT-4.

cial institutions.

Creating synonyms lists We test GPT-4's ability to create synonyms lists for populating entity definitions. We provided the following prompt:
`For designing a chatbot in the domain of private banking, give me a synonym list for the word "insolvent".`
GPT-4 returned these 13 terms:
`Bankrupt, Impoverished, Penniless, Financially ruined, Broke, Indigent, Destitute, Impecunious, In default, In debt, Insufficient funds, Unable to pay debts, Financially distressed.`
While some of these results are unlikely to be used in natural language, all of them are similar in meaning to the term "insolvent". Additionally, the list is not exhaustive, lacking slang terms, e.g., "skint".

Traits of good agents We test GPT-4's ability to identify traits associated with a particular role, an important element of chatbot personality design. We provided the following prompt:
`Describe the traits of a good client advisor in private banking in max. 100 words.`
GPT-4 responded:
`A good client advisor in private banking possesses strong financial knowledge, excellent communication and interpersonal skills, and a deep understanding of client's needs and goals. They maintain high ethical standards...`
We observe that those are indeed characteristics of a good agent (for details see A.2).

Persona-based responses We show examples of how GPT-4 can be used to personalize responses according to a (film) persona. We provided the following prompt:
`Write how a chatbot would respond to each of these scenarios, in the style of Dr. Jennifer Melfi, a fictional therapist from the TV series The Sopranos. 1. Inform customer product no longer available. ...`
GPT-4 responded:
`1. I understand that you were looking forward to obtaining this product, and I regret to inform you that it is no longer available. It must be disappointing, but perhaps we can explore alternative options that may fulfill your needs just as well.`
We observe that GPT-4 is able to answer this question *in character*, including empathy for the client, apologies and polite pointing in alternative directions.
In a second variation of this experiment, we asked GPT-4 to `write chatbot statements in grammatically correct yet simple English`. GPT-4 responded:
`1. I'm sorry, but the product you are looking for is no longer available.`
The example demonstrates how the language can be adapted by context, which in practice could come from the user data for personalization or from the owner's brand for consistency in customer communication.

Localization We test GPT-4's ability to localize agents to different language and dialect varieties of a language (for details see A.3). We provided the following prompt:
`Translate these three statements into German, Swiss German, Spanish, and French. 1. I regret to inform you that the product is no longer available. ...`

GPT-4 responded:
`German: 1. Es tut mir leid, Ihnen mitteilen zu müssen, dass das Produkt nicht mehr verfügbar ist. ...`
`Swiss German: 4. Es tuet mer leid, Ihne mitz'teile, dass s'Produkt nümme verfüegbar isch. ...`
`Spanish: 7. Lamento informarle que el producto ya no está disponible. ...`
`French: 10. Je regrette de vous informer que le produit n'est plus disponible. ...`
We observe that the translations are of high quality in all four languages. Even for Swiss German, a German dialect, the translation is good. This is astonishing because Swiss German does not have a standardized written format for all its varieties.

4.2 LLM as Real-time Booster

LLM, as a real-time booster, involves improving the agent's ability to understand and respond to user input in real-time conversations. This includes contextualizing the conversation to provide more relevant and personalized responses, intent classification and response generation to handle fallback and unhappy paths when the agent is unable to understand or fulfill the user's request, and disambiguating ambiguous requests by asking for more information. Additionally, generative models can assist with integrating closed Q&A and providing quick and accurate answers to common questions, among others. In Table 2, we provide examples of how LLM can improve the above-mentioned aspects in pipeline-based CAs.

Auto-correction We test GPT-4's ability to correct low-literacy utterances. We provided the following examples of phrases which were then autocorrected by GPT-4. The prompt starts with:
`Please rephrase the following 5 utterances into orthographically and grammatically correct American English...`
GPT-4 responded:
a) `wunt to cancal this accunt` → `I want to cancel this account`
b) `i want 2 get rid of my acount` → `I would like to delete my account`
c) `goodbye to my acount` → `I am closing my account`
We observe that orthographic errors are corrected (a), politeness is introduced (b), and utterances are re-phrased using clear English (c).

Context Switching We test GPT-4's ability to handle a user that switches between two intents (address change and money transfer) before providing enough information to complete the first task:
`User: Hi, I would like to transfer money. My bank account is 334402.`
`Chatbot: Hello! ... Please provide the following information:`
`1. The recipient's bank account number.`
`2. The amount you would like to transfer.`
`Once I have this information, I can proceed with the transfer.`
`User: BTW, I also need to change my address: It's Park Avenue 14.`
`Chatbot: Thank you for providing your new address. To complete the address change, please also provide the following information:`
`1. City`
`2. Postal Code ...`
`User: I live in 10012 New York.`
`Chatbot: Great, thank you for providing the city and postal code for your address change. .... Now, let's get back to the money transfer request. Please provide:`
`1. The recipient's bank account number.`
`2. The amount you would like to transfer....`
`User: I want to transfer 400 Dollars to account number 831123`
`Chatbot: Thank you for providing the necessary information.`
After receiving sufficient information about the address change, the bot gets back to get the complete information for the money transfer.

Out-of-scope Questions We added an out-of-scope question to the conversation in the previous paragraph:
`User: Where is Germany?`
`Chatbot: Germany is a country located in Central Europe....`
We observe that the bot gives a sensible answer when appending a random general knowledge question to the conversation in the previous paragraph. Staying in context, the agent will then ask the user for a final confirmation:
`Chatbot: Before we proceed with the1030 money transfer and address1031 change, please confirm if the1032 provided details are correct...`

Benefit	Explanation
Auto-correct	Correct / rephrase an orthographically and grammatically incorrect utterance to make it more easily classifiable by the bot e.g., "wunt to cancal this accunt" -> "I want to cancel this account"
Context switching	follows the user in switching back and forth between different intents like address change and money transfer
Out-of-scope questions	can be answered when regarding general knowledge
Creating stylistic variability	utterances can be rephrased, achieving a better writing style while maintaining the same meaning
Closed Q&A	Exact formulation of answer is picked from a defined set of options
Summarizing conversation	summarization for hand-over to a human agent

Table 2: Examples of LLMs as a real-time booster. A pipeline-based CA can be enhanced during deployment in various ways by GPT-4 overcoming its limitations.

Creating stylistic variability We test GPT-4's potential to introduce variability in English writing style. We provided the following prompt:

`For a chatbot, write 10 variations each one more apologetic than the previous one, of the statement: „I didn't understand what you said, please rephrase." Vary in vocabulary, grammar and tone ...`

GPT-4 responded (examples 1, 4, and 9):

`1. Sorry, I didn't quite get that. Could you rephrase your statement, please? ...`

`4. Apologies for the confusion, I'm unable to grasp what you're saying. Kindly rephrase your statement....`

`9. My most profound apologies for not comprehending your statement. I would be grateful if you could rephrase it for me.`

We observe that the generations are of great stylistic variability and that a controlled degree of servitude is introduced into the utterances (see A.4 for more details).

Closed Q&A We test GPT-4's capability to avoid hallucinations in closed Q&A by only providing exact predefined answers that are not altered (see A.5 for details). When testing the system with informally articulated questions, we got five correct answers from 5 trials.

Summarizing conversation We test GPT-4's capability to summarize a conversation between a chatbot and a user and state what the agent picking up the conversation needs to do (see A.6 for details). Summarizing for a CA is particularly useful when requesting confirmation from the client before concluding a conversation or handing over the conversation to a human operator. We observe that the model could deliver the response in the requested format.

5 Outlook

This paper proposes a hybrid approach that leverages LLMs, in particular GPT-4, to enhance pipeline-based CAs. Using this approach, maintainers of existing CAs can adopt new domains and overcome the limitations in conversations with users while ensuring seamless integration with the existing ecosystem. This approach accelerates the CA delivery process through the assistance of LLMs in generating intents, entities, synonyms, respective training data, and agent personality traits. During deployment, LLMs can boost pipeline-based CAs' performance by utilizing autocorrect, context-switching capabilities, answering out-of-scope questions, creating diverse and stylistically richer responses, and incorporating Closed Q&A and summarization. This paper presented experiments to showcase the scenarios mentioned above.

In future work, we will extend the ad-hoc subjective assessment to a more rigorous evaluation among different LLMs and provide an integrated solution to demonstrate the proposed hybrid approach. Given the existing risks regarding the reliability of LLMs (OpenAI, 2023; Bang et al., 2023; Zhao et al., 2023), our future research will focus on examining the factors that prompt business owners to consider the integration of LLMs within their pipeline-based CAs.

Acknowledgements

This work was supported by the European Union's Horizon 2020 research and innovation program via the project COALA "COgnitive Assisted agile manufacturing for a LAbor force supported by trustworthy Artificial Intelligence" (Grant agreement 957296). In addition, this work was supported by REasoning for Conversation and Information Technology Exchange (RECITE) project which is an OASIS Open Project dedicated to developing a standard for dialog modeling in conversational agents.

References

Hussam Alkaissi and Samy Mcfarlane. 2023. Artificial hallucinations in chatgpt: Implications in scientific writing. *Cureus*, 15.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity.

Frank Bentley, Chris Luvogt, Max Silverman, Rushani Wirasinghe, Brooke White, and Danielle Lottridge. 2018. Understanding the long-term use of smart speaker assistants. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(3).

Samuel R. Bowman. 2023. Eight things to know about large language models. *CoRR*, abs/2304.00612.

Hayet Brabra, Marcos Baez, Boualem Benatallah, Walid Gaaloul, Sara Bouguelia, and Shayan Zamanirad. 2022. Dialogue management in conversational systems: a review of approaches, challenges, and opportunities. *IEEE Transactions on Cognitive and Developmental Systems*, pages 1–15.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems. *ACM SIGKDD Explorations Newsletter*, 19(2):25–35.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Cognigy. 2023. Enterprise-ready generative ai for contact centers | cognigy.

Benjamin R. Cowan, Nadia Pantidi, David Coyle, Kellie Morrissey, Peter Clarke, Sara Al-Shehri, David Earley, and Natasha Bandeira. 2017. "what can i help you with?": Infrequent users' experiences of intelligent personal assistants. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '17, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Emily Dinan, Gavin Abercrombie, A. Stevie Bergman, Shannon Spruit, Dirk Hovy, Y-Lan Boureau, and Verena Rieser. 2021. Anticipating safety issues in e2e conversational ai: Framework and tooling.

Asbjørn Følstad and Cameron Taylor. 2020. Conversational repair in chatbots for customer service: The effect of expressing uncertainty and suggesting alternatives. In *Chatbot Research and Design*, pages 201–214, Cham. Springer International Publishing.

J. Harms, P. Kucherbaev, A. Bozzon, and G. Houben. 2019. Approaches for dialog management in conversational agents. *IEEE Internet Computing*, 23(2):13–22.

Frederick Jelinek. 1998. *Statistical Methods for Speech Recognition*. MIT Press, Cambridge, MA, USA.

Toby Jia-Jun Li, Jingya Chen, Haijun Xia, Tom M. Mitchell, and Brad A. Myers. 2020. *Multi-Modal Repairs of Conversational Breakdowns in Task-Oriented Dialogs*, page 1094–1107. Association for Computing Machinery, New York, NY, USA.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*.

Ewa Luger and Abigail Sellen. 2016. *"Like Having a Really Bad PA": The Gulf between User Expectation and Experience of Conversational Agents*, page 5286–5297. Association for Computing Machinery, New York, NY, USA.

Bei Luo, Raymond Lau, Chunping Li, and Yain Whar Si. 2022. A critical review of state–of–the–art chatbot designs and applications. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 12.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality.

Robert J. Moore and Raphael Arar. 2019. *Conversational UX Design: A Practitioner's Guide to the Natural Conversation Framework*. Association for Computing Machinery, New York, NY, USA.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

ME Peters, M Neumann, M Iyyer, M Gardner, C Clark, K Lee, and L Zettlemoyer. 2018. Deep contextualized word representations. arxiv 2018. *arXiv preprint arXiv:1802.05365*, 12.

Rasa. 2023a. Breaking free from intents: A new dialogue model.

Rasa. 2023b. Training data format - end-to-end training.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. *GitHub repository*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models.

A Appendix

This appendix contains selected transcripts of the prompts and the generated responses from GPT4.

A.1 Generating training utterances for intent classification

Prompt:

`Write 10 varied utterances to train a chatbot intent called cancel_account, which is for customers who want to close their account with ABC Bank. Sometimes mention the bank name, sometimes don't.`

Output:

```
1. I would like to close my account with ABC Bank, please help me with the process.
2. Can you please guide me on how to cancel my account at ABC Bank?
3. I want to terminate my banking relationship with ABC Bank, how can I do that?
4. I'm thinking of closing my account, what is the procedure?
5. I've decided to cancel my ABC Bank account, can you assist me with this?
6. Please help me shut down my account with your bank.
7. I no longer need my account at ABC Bank, how can I close it?
8. What's the process to deactivate my account with ABC Bank?
9. I would like to cancel my account; can you guide me through the steps?
10. I need to close my bank account, what information do you need from me?
```

A.2 Traits of a good agent

Prompt:

`Describe the traits of a good client advisor in private banking in max. 100 words.`

Output:
A good client advisor in private banking possesses strong financial knowledge, excellent communication and interpersonal skills, and a deep understanding of clients' needs and goals. They maintain high ethical standards, practice discretion and confidentiality, and build long-lasting relationships based on trust. Additionally, they demonstrate adaptability, staying updated on market trends and regulations, and proactively identify opportunities to grow clients' wealth. They are also highly organized, adept at problem-solving, and excel at collaborating with various stakeholders to deliver tailored financial solutions.

A.3 Localization

Prompt:
Translate these three statements into German, Swiss German, Spanish and French.
1. I regret to inform you that the product is no longer available.
2. I implore you to reconsider cancelling your account.
3. I shall now direct you to an agent who can further assist you.
Output:
German: 1. Es tut mir leid, Ihnen mitteilen zu müssen, dass das Produkt nicht mehr verfügbar ist.
2. Ich bitte Sie inständig, die Kündigung Ihres Kontos zu überdenken.
3. Ich werde Sie jetzt an einen Agenten weiterleiten, der Ihnen weiterhelfen kann.
Swiss German: 4. Es tuet mer leid, Ihne mitz'teile, dass s'Produkt nümme verfüegbar isch.
5. Ich bitte Sie inständig, d'Chündigung vo Ihrem Konto z'überdenke.
6. Ich wird Sie jetzt zuenem Agent wyterleite, wo Ihnen cha wyterhelfe.
Spanish: 7. Lamento informarle que el producto ya no está disponible.
8. Le imploro que reconsidere cancelar su cuenta.
9. Ahora le dirigiré a un agente que pueda asistirle más.
French: 10. Je regrette de vous informer que le produit n'est plus disponible.
11. Je vous implore de reconsidérer l'annulation de votre compte.
12. Je vais maintenant vous diriger vers un agent qui pourra vous aider davantage.

A.4 Creating stylistic response variability

Prompt:
For a chatbot, write 10 variations each one more apologetic than the previous one, of the statement: „I didn't understand what you said, please rephrase.“ Vary in vocabulary, grammar and tone, including terse, sincere and apologetic variations.
Output:
1. Sorry, I didn't quite get that. Could you rephrase your statement, please?
2. My apologies, I'm having trouble understanding. Would you mind rephrasing your question?
3. I'm sorry, I didn't comprehend your message. Please rephrase it for me.
4. Apologies for the confusion, I'm unable to grasp what you're saying. Kindly rephrase your statement.
5. I deeply regret that I didn't understand your message. Please accept my apologies and rephrase your question.
6. My sincerest apologies, I'm struggling to comprehend your message. Could you kindly restate it for me?
7. I'm terribly sorry for not understanding your words. Please forgive me and rephrase your statement.
8. I feel so apologetic for being unable to understand what you said. Please give me another chance and rephrase your message.
9. My most profound apologies for not comprehending your statement. I would be grateful if you could rephrase it for me.
10. I am extremely sorry for my inability to understand your message. It would mean a lot if you could kindly rephrase it for me.

A.5 Closed Q&A

Prompt:
For each question literally answer one of the below answers in exactly that wording, if those answers are suitable. If none of the below answers are a suitable answer

to the question answer: “Please call 001 23 45 89 01”.
1. To change your address you need to sent a mail to info.company.com including your new and old complete address.
2. If you want to open a bank account, provide a copy of your password and a list of current bank accounts.
3. If you want to close an account call 001 23 45 89 28
4. To change your password sent a mail to info.company.com with that request.
Output:
User: I have a new address
Chatbot: To change your address you need to sent a mail to info.company.com including your new and old complete address.
User: How can I get an account with your company?
Chatbot: If you want to open a bank account, provide a copy of your password and a list of current bank accounts.
User: I want to quit
Chatbot: If you want to close an account call 001 23 45 89 28
User: I forgot my pwd
Chatbot: To change your password sent a mail to info.company.com with that request.
User: What are the interest rates I need to pay for a mortgage?
Chatbot: Please call 001 23 45 89 01.

A.6 Summarizing conversation

Prompt:
Summarise the following conversation between a chatbot and a person, and state what the agent picking up the conversation needs to do.
–
Chatbot: Hi, how can I help?
User: I need to get a new debit card
Chatbot: I can help you order a new debit card. Is this a new card or a replacement?
User: Replacement
Chatbot: Is your current card lost, damaged or stolen?
User: Damaged
Chatbot: Please go to www.cardreplace.com to request your new card. Did I help you today?
User: The trouble is the address you have for me is out of date, so before you post it you need to update my address
Chatbot: I’m worry I didn’t understand that. Did I help you today?
User: I need to update my address
Chatbot: My colleague can help you this query, I’m connecting you now. Feel free to add any information that might be help now.
User: My new address is 1 Main Street, Capital City, Countryland, AA1 XZY.
–
Use this format:
Agent Action Required:
Summary:
Output:
Agent Action Required: Update the user’s address and assist with the debit card replacement process.
Summary: The user needs a replacement debit card due to a damaged one. However, their address on file is outdated. The user provided their new address as 1 Main Street, Capital City, Countryland, AA1 XZY.