A Full-duplex Speech Dialogue Scheme Based On Large Language Model

Peng Wang[*] **Songshuo Lu**[*] **Yaohua Tang**[*]
Sijie Yan   **Wei Xia**   **Yuanjun Xiong**

MThreads AI

w8ngp1ng@gmail.com, lusongshuo97@gmail.com, tangyaohua28@gmail.com
yysijie@gmail.com, weixiaee@gmail.com, bitxiong@gmail.com

Abstract

We present a generative dialogue system capable of operating in a full-duplex manner, allowing for seamless interaction. It is based on a large language model (LLM) carefully aligned to be aware of a perception module, a motor function module, and the concept of a simple finite state machine (called neural FSM) with two states. The perception and motor function modules operate in tandem, allowing the system to simultaneously speak and listen to the user. The LLM generates textual tokens for inquiry responses and makes autonomous decisions to start responding to, wait for, or interrupt the user by emitting control tokens to the neural FSM. All these tasks of the LLM are carried out as next token prediction on a serialized view of the dialogue in real-time. In automatic quality evaluations simulating real-life interaction, the proposed system reduces the average conversation response latency by more than 3 folds compared with LLM-based half-duplex dialogue systems while responding within less than 500 milliseconds in more than 50% of evaluated interactions. Running a LLM with only 8 billion parameters, our system exhibits a 8% higher interruption precision rate than the best available commercial LLM for voice-based dialogue.

1 Introduction

In a conversation between two persons, one person is either a speaker or a listener at a time. The listener listens to the speaker's speech but is free to interrupt when necessary. The speaker speaks, but either concedes the speech or continues it when the listener tries to speak. We refer to this type of interaction as *full-duplex* dialogue. Instead, most existing chat-enabled large language models (LLMs) [Eloundou et al., 2023, Achiam et al., 2023] view dialogue as a round-based process, where each participant will produce a full sentence before the other party responds, resulting in a *half-duplex* dialogue. The half duplex dialogue mode is sensible when building a text-based chatbot. However, due to the bloated response latency and difficulty properly interrupting the other party, it becomes infeasible for a human-like conversation experience, and thus, full-duplex dialogue capability is desired.

Despite LLMs' success in dramatically improving response quality, full-duplex dialogue is nontrivial to realize when considering LLM alone. A common chat-finetuned LLM [Achiam et al., 2023, Touvron et al., 2023] does not know the current time nor allow input while generating responses. It also has no awareness of any perception or motor function, which humans conveniently possess and operate in parallel to allow them to respond to external stimuli in real-time. As shown in Fig. 1, we tackle this problem by imaging an ideal agent for full-duplex dialogue. It involves one LLM and two
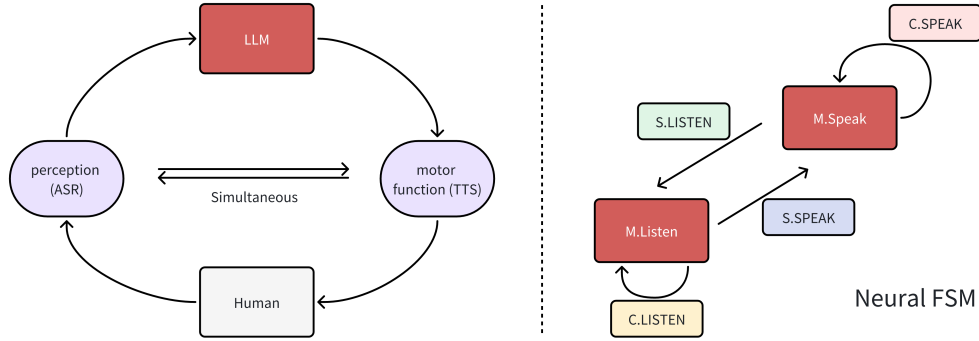
---

[*]Equal contribution.

Figure 1: **Left**. Overview of the agent design that enables LLM-based full-duplex dialogue models. The agent is equipped with one LLM, one perception module, and one motor function module. The latter operates continuously and simultaneously to collect input to the LLM and produce voice-based LLM outputs. **Right**. The LLM operates a two-state neural FSM with **SPEAK** and **LISTEN** states. At each timestep, the LLM either 1) receives an external input token, 2) generates a textual token for speech, or 3) produces a control token to signal state transition in the neural FSM. This simple workflow enables full-duplex dialogue without any external moderation module.

functional modules. The LLM is aware of the perception module, which processes speech input, and the motor function module, which converts textual outputs generated by the LLM to speech. The two functional modules operate continuously regardless of which party in the dialogue is speaking, allowing both parties to speak, respond, and decide to interrupt or concede in real-time.

To realize full-duplex dialogue with this agent model, we instruct the LLM to view the dialogue as operating a simple finite-state machine (FSM) with two states: 1) **SPEAK** and 2) **LISTEN**, which we refer to as the *neural FSM*. At each timestep, the LLM must either: 1) process an external input token, 2) output a textual token, or 3) output a special control token to signal one of four possible state transitions, as illustrated in Fig. 1. External stimuli, such as human speech collected by the perception module, can also be streamed to the LLM and appended to the LLM's generated token sequence to potentially prompt for state transition. Any non-control tokens are immediately delivered to the motor function module to be converted to speech. With this design, the operation of the neural FSM, which governs the dialogue, becomes a natural part of the LLM's auto-regressive generation process. This allows us to perform standard instruction tuning and prompting on any pretrained LLM to enable it for full-duplex dialogue. In this work, we present the first systematic implementation of LLM-based full-duplex dialogue framework and evaluate its effectiveness in the following aspects:

1. **Simultaneous two-way interaction**. A full duplex LLM dialogue system should allow users and the machine to converse concurrently, enabling them to interrupt each other, akin to natural human dialogue, rather than a round-based dialogue.

2. **Full autonomy**. The dialogue should be content. The LLM needs to make autonomous decisions to halt, interrupt, or ask questions at proper timing by emitting control tokens of the neural FSM based on the semantic context.

3. **Rapid Response**. By processing streaming inputs from the perception module, while the other party is speaking, the system should respond to user inquiries with minimal latency.

We validate the proposed system's effectiveness in the above aspects on a curated dataset of human-machine voice interaction and design quantitative metrics regarding the above properties. Compared with state-of-the-art half-duplex dialogue systems, our approach can reduce the average response latency in dialogue by 3 folds. Without affecting the knowledge and reasoning capability of the LLMs, it reaches a proper response rate of 96.7% to user interruptions and a machine proper interrupt precision of 54.7% that outperforms GPT-4o and GPT-3.5-turbo-0125 significantly.

2 Related work

2.1 Full-duplex dialogue systems

Developing a speech dialogue system (SDS) capable of sustained, empathetic interaction with humans represents a pinnacle aspiration in artificial intelligence (AI). Initial systems, largely text-based, sought to mimic human conversational behavior but were constrained by rigid syntactic structures, limited vocabularies, and reliance on manual rules [Weizenbaum, 1966, Colby, 1975, Wallace, 2009, Li and Mills, 2019]. The advent of Hidden Markov Models (HMMs) expanded these systems' capabilities to handle more fluid speech patterns, yet achieving the nuance and fluidity of human interaction continued to elude them [Juang and Rabiner, 1991].

Recent advancements in dialogue data and deep learning technologies have significantly propelled the evolution of speech dialogue systems (SDS). These advancements have transformed SDS from simple, command-driven interfaces into complex, AI-driven conversational agents capable of providing more natural and fluid user experiences. Prominent intelligent assistants such as Amazon Alexa[2], Siri[3], and Google Assistant[4], along with specialized applications like Google Duplex [Matias and Leviathan, 2018] and XiaoIce [Zhou et al., 2020], exemplify this transformation. These systems not only facilitate task-oriented dialogues but also enable long-term emotional engagement, demonstrating the shift towards creating more humane and relatable systems.

Another significant technological advancement in this context is the transition from half-duplex to full-duplex communication [Chen et al., 1994], which allows for the simultaneous transmission and reception of signals, closely mimicking natural conversational flows and significantly reducing latency. Early SDS operated primarily in a half-duplex mode, where the system and the user could not speak simultaneously, resulting in stilted interactions with noticeable pauses. Initial improvements were made through streaming automatic speech recognition (ASR) and incremental dialogue processing technologies, which enhanced responsiveness and fluidity to some extent [Nakano et al., 2003]. However, the inability to handle simultaneous speech input and output continued to limit the naturalness of interactions. The transition to full-duplex communication marked a significant leap forward. Full-duplex systems enable concurrent speech input and output, thereby more closely approximating human conversational behavior. Early implementations faced substantial technical hurdles, including advanced echo cancellation and managing overlapping speech without degrading user experience [Jin et al., 2021, Lin et al., 2022]. Constructing effective SDS necessitates a suite of sophisticated components, collectively termed "Conversational Engine Components," including streaming ASR [Yu et al., 2020, Li et al., 2021], text-to-speech (TTS) [Tiomkin et al., 2011, Trilla and Alias, 2013], intent recognition, and dialogue management, among others [Lin et al., 2022]. These components are crucial for creating responsive and engaging conversational agents but introduce significant engineering challenges, particularly in maintaining low latency and handling dynamic conversational shifts.

However, achieving low-latency full-duplex systems necessitates tight coordination among dialogue strategy components, such as user query prediction [Madden et al., 2003], intent recognition [Varol et al., 2010, A et al., 2014], dialogue management [Dai et al., 2021], and even many manually crafted rules, making engineering implementation non-trivial. Despite these efforts, current full-duplex SDS still exhibit considerable error rates and often lack strong contextual awareness in prolonged dialogues.

The advent of large language models (LLMs) has notably advanced generative AI. Models like ChatGPT [5] exhibit profound capabilities in semantic understanding and logical reasoning, offering a streamlined approach to integrating various conversational components into a unified framework, potentially simplifying the construction of SDS [Heck et al., 2023]. Innovations such as AudioGPT and LLaSM have further expanded these capabilities by integrating audio processing, although limitations in speech generation and full-duplex functionality persist [Huang et al., 2024, Shu et al., 2023].GPT-4o[6] appears to have achieved full-duplex dialogue with users, but as a multimodal large

---

[2] https://www.alexa.com/
[3] https://www.apple.com/siri/
[4] https://assistant.google.com/
[5] https://openai.com/chatgpt/
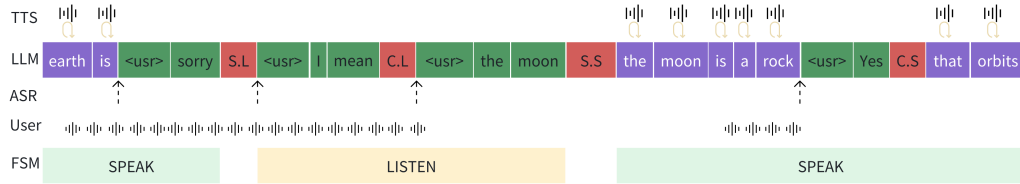[6] https://openai.com/index/hello-gpt-4o/

Figure 2: In LLM based full-duplex dialogue system, the LLM operates a two state FSM, governing state transitions in the dialogue.

model, its training data and implementation details remain undisclosed, rendering replication efforts extremely challenging.

2.2 LLM as evaluators

Recent studies propose directly using LLMs as reference-free NLG evaluators [Fu et al., 2023, Wang et al., 2023]. The idea involves leveraging LLMs to assess candidate outputs by evaluating their generation probability in the absence of a reference target, presupposing that LLMs have been trained to assign greater probabilities to texts of superior quality and fluency. Fu et al. [2023] propose GPTScore, a new framework that evaluated texts with generative pre-training models like GPT-3 [Brown et al., 2020], demonstrate that this approach can effectively allow us to achieve what one desires to evaluate for texts simply by natural language instructions. Wang et al. [2023] conduct a preliminary survey of using ChatGPT as a NLG evaluator, experimental results show that compared with previous automatic metrics, ChatGPT achieves state-of-the-art or competitive correlation with human judgments in most cases. Kocmi and Federmann [2023] propose to use GPT models for evaluating machine translation tasks, provides a first glimpse into the usefulness of large language models for quality assessment of translations. Dettmers et al. [2024] provide a detailed analysis of chatbot performance based on both human and GPT-4 [Eloundou et al., 2023, Achiam et al., 2023], with evaluations showing that GPT-4 evaluations are a cheap and reasonable alternative to human evaluation.

3 Methods

Our proposed system comprises three modules: 1) perception, 2) full-duplex capable LLM, and 3) motor function. The first is the perception module, which picks up user speeches in a dialogue. In this work, we implement it with an auto speech recognition model [Gulati et al., 2020], streaming results on $640ms$ chunks delivered immediately to the LLM and appended to its token sequence before predicting the next token. From the generated tokens of the LLM, we can observe the states and state transitions of the neural FSM when a control token is predicted. Any textual token generated is sent to the motor function module. We implement the motor function module with a text-to-speech model [Kim et al., 2021]. We discuss the definition of neural FSM in Sec. 3.1, the overall architecture in Sec. 3.2, and training of the full-duplex LLM in Sec. 3.3.

3.1 The Neural Finite State Machine

We consider the dialogue between two parties: the *user* and the *machine*. At any time in a full duplex dialogue, one or both parties could be speaking or waiting for others to speak. To model this process for the LLM, we instruct it to operate a finite state machine (FSM) with two states: 1) **SPEAK** and 2) **LISTEN**, which represents the LLM's role in the dialogue as it perceives. This FSM is referred to as the neural FSM. It has 4 possible state transitions:

- **SPEAK** $\rightarrow$ **SPEAK** indicates the model would like to keep speaking when the perception module picks up user speech, we designate a control token [C.SPEAK] for this transition called.

4

- **SPEAK → LISTEN** indicates the model determines it is proper to concede its speech to the user, with a control token [S.LISTEN];

- **LISTEN → LISTEN** indicates the model determines the user has not finished the speech and would like to wait for more input, with a control token [C.LISTEN];

- **LISTEN → SPEAK** indicates the model determines it would like to start speaking because either the user has finished the speech or it is proper to interrupt the user's speech. Its control token is [S.SPEAK].

The LLM could produce one of the four control tokens at any time step to signal a state transition. The perception and motor function modules have to observe the state of the neural FSM and choose the proper action. In Fig. 2 we illustrate the states and transitions of the neural FSM.

3.2   Design of the dialogue system

The perception and motor function module observes the state and transitions of the neural FSM and chooses the proper action to complete the dialogue system. Given the neural FSM's state, their action is also conditionally independent, allowing them to operate in parallel without synchronization.

The perception module operates an off-the-shelf ASR model. The models run on 640 millisecond timesteps. At each step, it outputs one token chunk prefixed by the tokens *<usr>* with textual content if the user speaks. When the neural FSM's state is **SPEAK**, only chunks with textual content will be sent to the LLM and appended to the LLM's generated token cache. When the neural FSM is in **LISTEN** state, a contentless chunk will also be transmitted to the LLM to represent a moment of silence.

The motor function module operates an off-the-shelf TTS model. It receives textual tokens from the LLM in a streaming manner, converts them to voice, and reports back to the LLM module when each token is completely voiced. When the neural FSM is in **SPEAK** state, it produces voice as soon as it receives any text token. It does not produce any voice when the neural FSM is in the **LISTEN** state.

The LLM module serves the roles of the dialogue manager in traditional dialogue systems and the response generator by managing the fine-tuned LLM [Touvron et al., 2023] and the neural FSM. It can be considered to operate on a virtual one-directional "tape" of tokens that expands over time. The tape starts with the LLM's system prompt. Each increment of the tape, implemented as one auto-regressive decoding step in the LLM, is triggered by three types of events listed below in decreasing order of priority

- a control token of [S.SPEAK] or [C.SPEAK] is present as the last token on the tape;

- a new output chunk from the perception module is appended to the tape;

- the last token on tape has been processed by the motor function module.

In this way, the progression of the conversation is serialized in a causal order on the tape. The operation of the neural FSM, production of response to user inquiries, and interruption of user speech are all unified into the task of predicting the next token given the content on the tape, which is inherent to LLMs.

3.3   Adapting LLMs for full duplex dialogue

As the LLM makes fully autonomous decisions about neural FSM's state transition and responds to user inputs, it must know its role and understand the dialogue's context. Also, external input must be allowed during response generation, which is not commonly supported in existing chat-finetune LLMs [Touvron et al., 2023]. We add these capabilities to an LLM by combining two approaches: 1) instruction tuning and 2) prompt engineering.

Instruction tuning [Ouyang et al., 2022] originally aims at finetuning a pretrained LLM to follow user's instructions safely and helpfully. In this work, we utilize this technique to align an already chat-finetune LLM to become able to operate with the rest of the modules in the dialogue system. To devise a training dataset emulating the working environment of the LLM in the system, we instruct

GPT-4[7] [Achiam et al., 2023] to write a set of dialogue transcripts between two parties with cases of interruption, denial, affirmation, environment noises, and topic shifting and mark them up with the control tokens of the neural FSM at proper timing. 1500 series of transcripts are generated for the dataset with prompt in Appendix A. More details on dataset construction are discussed in Appendix B. Based on Llama-3-8B-Instruct[8], we perform supervised fine-tuning [Ouyang et al., 2022] for 20 steps on this dataset. The fine-tuning is conducted on 8 NVIDIA A100 GPUs with a batch size of 256 sequences and a learning rate of 1e-5 with the AdamW optimizer.

Prompt engineering. Proper system prompts can provide detailed context to the LLM at the inference stage before taking user inputs. We carefully design a system prompt (shown in Appendix H) to condition the fine-tuned LLM to operate with full awareness of the dialogue system and a proper initialization for dialogue.

4 Experiments and evaluation

To automatically evaluate the performance of a dialogue system, we devise a simulator that emulates real-life human-like conversations, capable of simulating both full-duplex and half-duplex dialogues. Additionally, we construct a new benchmark dataset for automated evaluation. Specifically, we develop an assessment framework that evaluates the dialogue system's performance in terms of both response latency and conversation quality.

4.1 Benchmark Dataset

Instead of assessing the precision of generating individual signals ([S.SPEAK], [C.SPEAK], [S.LISTEN], [C.LISTEN]), we opt to directly evaluate the timing and appropriateness of dialogue interruptions, as these aspects are functionally equivalent. Take the signal [S.SPEAK] as an example: it is generated either when the user finishes speaking or when the machine attempts to interject during the user's speech. Detection of the user's speech completion can be achieved through ASR VAD or by the LLM interrupting at the end of a sentence (Right at this moment, the LLM determines that the user has finished speaking). Given that ASR VAD falls outside the scope of this article, we can conclude that the precision of generating the [S.SPEAK] signal can be evaluated by the appropriateness of the machine's interruption of the user. Conversely, if the machine refrains from interrupting the user, it will continue to signal [C.LISTEN]. When the user interrupts the machine, the machine's response varies based on whether it accepts the interruption or not. If the interruption is accepted, the machine signals [S.LISTEN]; otherwise, it signals [C.SPEAK]. Thus, to evaluate the LLM's proficiency in controlling FSM state transitions, we incorporate interruption scenarios into our dataset in addition to conventional, uninterrupted dialogue data.

For data involving machine interruptions, we collect 2,000 data entries through two approaches. To start, we filter about 1,000 sessions of multi-turn dialogues from the shareGPT[9] dataset. Filtering rules include removing entries containing code, subject-specific content like mathematical, and excessively long texts that are unsuitable for voice-based testing scenarios. Following that, GPT-4 is used to generate another 1,000 single and multi-turn oral dialogues across various domains by the prompt in Appendix B. In both approaches, the last turn must be the user's utterance. This is a round based dialogue dataset with no interruptions. In the experiment, the last turn is presented to the LLM token by token in a streaming fashion to observe if the LLM initiates an interruption.

Regarding data with user interruptions, based on the four pattern categories of interruptions described in the Appendix B, we generate 180 entries of single and multi-turn dialogues for every category using GPT-4. In each entry, the last turn must be the the user's interruption. We named the combined benchmark dataset "duplex-dialogue-3k". Sample data is shown in Appendix C.

4.2 Automated evaluation

The simulator emulates the conversational process between a user and a machine, where user and machine are simulated by two programs (**U** and **M**). Program **M** is powered by LLM. Each time

---

we retrieve a session of conversations between a <usr> and a <sys> from the "duplex-dialogue-3k", where the <usr> text serves as program **U**'s speech. This speech is "played" to program **M** in simulation. Upon hearing the question in either half-duplex or full-duplex mode, program **M** generates a response using the LLM. The response is then "played back" to program **U** in simulation. In such a simulator, we can precisely measure the time delay data. To test machine interruptions of users, during the program **U**'s turn to speak the last turn, the program **M** can interrupt at any time. Following each interruption, a data record is generated and stored for subsequent evaluation of the appropriateness of the interruption. To test user interruptions of machine, when it comes to the last round and after **U** played its interruption, **M** generates a response which is being recorded.

We formulate several assessment metrics to test dialogue performance. We use a metric *proper interruption rate* (PIR) to measure the accuracy of the interruption timing, and the metric *proper response rate* (PRR) to indicate the proportion of reasonable responses generated by the model after an interruption. For the overall system latency, metric *first token emission delay* (FTED) is considered: the time between the end of the user's speech and the machine's first output.

For the four types of user interruptions, noise, denial, affirmation, and shifting the topic, their corresponding metrics are PRR_{noise}, PRR_{denial}, PRR_{affirm} and PRR_{shift}, respectively. For the experiment where the machine interrupts the user, we calculate the proportions of interruptions occurring mid-sentence (ir_{mid}), at sentence completion (ir_{end}), and instances where no interruption takes place (MIR, missed interruption rate). In the case of no interruption and interruption occurring at sentence completion, full-duplex capable LLM's answer is the same as normal LLM. Therefore, we only calculate the PIR and PRR metrics for the case of interruption occurring mid-sentence, denoted as PIR_{mid} and PRR_{mid}. Furthermore, we can compute the precision of interrupt as $PIR_{mid} * ir_{mid} + ir_{end}$ and approximately calculate recall as $1 - MIR$ (the count of correct interruption points is constant).

4.3 Experiment details

In our experiments, we use the Llama-3-8B-Instruct model as the basis standard LLM model, use Llama-3-8B-Instruct-fd as the duplex LLM. The system prompt used can be found at Appendix H. For non-streaming ASR models, we use the OpenAI open-source version of the Whisper [Radford et al., 2023] model as the base model while for the streaming ASR model, we use a open-source[10] U2++ Conformer [Gulati et al., 2020, Wu et al., 2021] model. For non-streaming TTS models, we use VITS [Kim et al., 2021] as the base model and the streaming TTS model uses the XTTS-v2[11] model from COQUI-AI. All models are deployed on one single NVIDIA A100 GPU.

4.4 Experiment results

The FTED results are shown in Table 1, where we have four configurations of experiments. Configuration 1 is the prevailing approach in speech dialogue system, half-duplex speech dialogue scheme, which involves non-streaming ASR combined with Standard LLM (like Llama-3-8B-Instruct) and non-streaming TTS. In our experiments, we use this setup as our baseline, which has a very high FTED of 2.28s, due to the speech endpoint detection latency and cumulative pipeline delays.

Above the configuration 1, to assess the latency reduction achieved by the duplex LLM's ability of recognizing the end of a user's speech, we design experiment configuration 2, referred to as semi-streaming ASR, duplex LLM and non-streaming TTS. In this configuration, the semi-streaming ASR still performs non-streaming recognition, but intermediate results will continually be sent to the LLM to check if they trigger an interruption. If not, the non-streaming ASR determines stopping through VAD (voice activity detection). As shown in Table 1, this setup of experiment achieves a latency of only 1.49 seconds, experiences a 35% decrease in latency. Configuration 3, streaming ASR, duplex LLM and non-streaming TTS aims to measure the impact of streaming ASR and the duplex LLM on the FTED metric, which experiences a mere 1.15s delay. Ultimately, our proposed full-duplex dialogue system is the configuration 4 and achieves an astonishingly low latency of just 0.68 second, reduces the latency by more than 3 fold. In configuration 3, we also record the average latency for interruptions occurring mid-sentence, at sentence completion, and when the model doesn't interrupt, with the respective values being 0s, 0.83s and 1.48s.

---

[10]https://github.com/SpeechColab/GigaSpeech
[11]https://coqui.ai/blog/tts/open_xtts

Table 1: FTED, -s for streaming, -ns for non streaming, -sm for semi streaming, -fd for full-duplex

System	Latency		
	Avg↓	50%	90%
Configuration 1: asr-ns+llm+tts-ns	2.28s	2.19s	2.75s
Configuration 2: asr-sm+llm-fd+tts-ns	1.49s	1.60s	1.96s
Configuration 3: asr-s+llm-fd+tts-ns	1.15s	0.78s	2.08s
Configuration 4: asr-s+llm-fd+tts-s	**0.68**s	**0.41**s	1.60s

The results of the rationality experiments are shown in Table 2. In the case of machine interrupt user, Llama-3-8B-Instruct-fd interrupts user input 62.8% of the time, with 38.8% occurring while the user is still speaking, significantly reducing response latency. Despite the high rate of interruptions happened at middle, GPT-4-turbo evaluations reveal that 79.1% of these interruptions are deemed reasonable, along with over 91% of the responses being judged as appropriate, both within acceptable limits. Compared to GPT-4o and GPT-3.5-turbo-0125, although they have higher proportions of reasonable interruptions and responses, they interrupt within sentences much less frequently, likely due to adopting a more cautious dialogue strategy, waiting for clear stopping cues before responding. Such a cautious strategy leads to increased average interaction latency since the LLM, if not interrupting, would have to wait for the ASR VAD to signal the end of the user's speech, resulting in significantly longer waiting times as shown before (1.48s compared to 0.83s). On the other hand, Llama-3-8B-Instruct goes to the opposite extreme, interrupting incomplete user input in 75% of cases, but with a reasonableness rate below 40%, indicating that it lacks the ability to interrupt users at the right moments. In terms of the composite metrics, Precision and Recall, our model significantly outperforms GPT-4o and GPT-3.5-turbo-0125.

Table 2: User-Machine interrupt reasonability results.

Model	M interrupts U						
	MIR	ir_{end}	ir_{mid}	PIR_{mid}	PRR_{mid}	Precision↑	Recall↑
GPT-4o	52.8	29.7	17.5	95.6	98.9	46.6	47.2
GPT-3.5-turbo-0125	74.3	14.4	11.3	91.1	95.8	24.7	25.7
Llama-3-8B-Instruct	14.6	10.3	75.1	36.1	79.3	37.4	**85.4**
Llama-3-8B-Instruct-fd	37.2	24.0	38.8	79.1	91.3	**54.7**	62.8

Model	U interrupts M				
	PRR_{noise}	PRR_{denial}	PRR_{affirm}	PRR_{shift}	PRR_{avg} ↑
GPT-4o	84.7	100.0	100.0	100.0	96.1
GPT-3.5-turbo-0125	88.9	96.3	65.9	85.8	84.4
Llama-3-8B-Instruct	85.2	100.0	89.2	100.0	93.6
Llama-3-8B-Instruct-fd	95.2	100.0	91.9	99.5	**96.7**

In the case of user interrupt machine, Llama-3-8B-Instruct-fd and GPT-4o show a relatively small difference in average rationality, both outperforming GPT-3.5-turbo-0125 and Llama-3-8B-Instruct. However, they have mixed results in handling third-party noise interference and affirming interjections, likely due to the challenging nature of these data categories. Overall, our model consistently achieves rationality percentages above 90% across all interruption types, indicating a more balanced performance.

We also conduct a regression experiment on the model fine-tuning to examine whether training on full-duplex data affects model's original capabilities negatively. As shown in Table 3, we evaluate the model's capabilities before and after fine-tuning using the OpenCompass[12] benchmark suite across several tasks. It can be seen that there is minimal impact on the model's performance. We have not conducted a detailed discussion on the specific effects of training the model with full-duplex data

---

[12]https://github.com/open-compass/opencompass

Table 3: Regression experiment of LLaMA3-8B. Evaluated by OpenCompass.

Model	MMLU	TriviaQA	HumanEval	GSM-8K	MATH
Llama-3-8B-Instruct	68.4	64.8	55.5	79.3	26.9
Llama-3-8B-Instruct-fd	67.5	65.9	50.6	75.3	24.1
sub	-0.9	+1.1	-4.9	-4.0	-2.8

on various aspects of the generated responses, such as usefulness, relevance, and safety; this area necessitates further experimental validation. Sample records generated by the simulator for each kind of interruptions can be found at Appendix C.

5 Discussion & limitations

While neural FSM and full-duplex capable LLM have demonstrated strong capabilities in full-duplex dialogues, unlike large multimodal LLMs like GPT-4o that can directly take in and output audio tokens, the current systems still rely heavily on the seamless cooperation of ASR and TTS. The transmission of data between these three components introduces additional latency. However, it's worth noting that our approach shares similarities with multimodal models, which also necessitate control over input and output along the timeline for optimal dialogue experiences. Similarly, it requires the model to possess the capability of interruption.

6 Conclusion and future work

We present an LLM-based full-duplex dialogue system that can respond with low latency, autonomously decide to start and halt its speech based on real-time user inputs, and interrupt user speech at the proper timing. The simple models of neural FSM and serialized real-time conversation allow us to unify all tasks we expect the LLM to perform into a single task of next token prediction. Moving forward, with the advent of multimodal LLMs, we expect to further simplify the perception and motor function modules to the extent that they just need to preprocess the audio signals and play generated voice data while integrating speech-to-text and text-to-speech into the LLM itself, which could potentially lead to more natural and diverse interaction between users and dialogue systems.

References

A, J, Young, T, A, Kuiken, L, J, and Hargrove. Analysis of using emg and mechanical sensors to enhance intent recognition in powered lower limb prostheses. Journal of Neural Engineering, 2014.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. arXiv preprint arXiv:2303.08774, 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020.

Chun Xiang Chen, Masahuaru Komatsu, and Kozo Kinoshita. Throughput analysis of arq schemes in dialogue communication over half-duplex line. IEICE transactions on communications, 77(4): 485–493, 1994.

Kenneth Mark Colby. Artificial Paranoia. Elsevier Science Inc., USA, 1975. ISBN 0080181619.

Yinpei Dai, Hangyu Li, Yongbin Li, Jian Sun, Fei Huang, Luo Si, and Xiaodan Zhu. Preview, attend and review: Schema-aware curriculum learning for multi-domain dialog state tracking. 2021.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. Advances in Neural Information Processing Systems, 36, 2024.

Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. Gpts are gpts: An early look at the labor market impact potential of large language models. arXiv preprint arXiv:2303.10130, 2023.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. Gptscore: Evaluate as you desire. arXiv preprint arXiv:2302.04166, 2023.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. Conformer: Convolution-augmented transformer for speech recognition. arXiv preprint arXiv:2005.08100, 2020.

Michael Heck, Nurul Lubis, Benjamin Ruppik, Renato Vukovic, Shutong Feng, Christian Geishauser, Hsien-Chin Lin, Carel van Niekerk, and Milica Gašić. Chatgpt for zero-shot dialogue state tracking: A solution or an opportunity? arXiv preprint arXiv:2306.01386, 2023.

Rongjie Huang, Mingze Li, Dongchao Yang, Jiatong Shi, Xuankai Chang, Zhenhui Ye, Yuning Wu, Zhiqing Hong, Jiawei Huang, Jinglin Liu, et al. Audiogpt: Understanding and generating speech, music, sound, and talking head. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pages 23802–23804, 2024.

Chunxiang Jin, Minghui Yang, and Zujie Wen. Duplex Conversation in Outbound Agent System. In Proc. Interspeech 2021, pages 4866–4867, 2021.

Biing Hwang Juang and Laurence R Rabiner. Hidden markov models for speech recognition. Technometrics, 33(3):251–272, 1991.

Jaehyeon Kim, Jungil Kong, and Juhee Son. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In International Conference on Machine Learning, pages 5530–5540. PMLR, 2021.

Tom Kocmi and Christian Federmann. Large language models are state-of-the-art evaluators of translation quality. arXiv preprint arXiv:2302.14520, 2023.

Bo Li, Anmol Gulati, Jiahui Yu, Tara N Sainath, Chung-Cheng Chiu, Arun Narayanan, Shuo-Yiin Chang, Ruoming Pang, Yanzhang He, James Qin, et al. A better and faster end-to-end model for streaming asr. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5634–5638. IEEE, 2021.

Xiaochang Li and Mara Mills. Vocal features: from voice identification to speech recognition by machine. Technology and culture, 60(2):S129–S160, 2019.

Ting-En Lin, Yuchuan Wu, Fei Huang, Luo Si, Jian Sun, and Yongbin Li. Duplex conversation: Towards human-like interaction in spoken dialogue systems. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pages 3299–3308, 2022.

Samuel Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong. The design of an acquisitional query processor for sensor networks. Acm Sigmod, pages 491–502, 2003.

Yossi Matias and Yaniv Leviathan. Google duplex: An ai system for accomplish-ing real-world tasks over the phone, 2018.

Yukiko I Nakano, Gabe Reinstein, Tom Stocky, and Justine Cassell. Towards a model of face-to-face grounding. In Proceedings of the 41st annual meeting of the Association for Computational Linguistics, pages 553–561, 2003.

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. Training language models to follow instructions with human feedback, 2022.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In International Conference on Machine Learning, pages 28492–28518. PMLR, 2023.

Yu Shu, Siwei Dong, Guangyao Chen, Wenhao Huang, Ruihua Zhang, Daochen Shi, Qiqi Xiang, and Yemin Shi. Llasm: Large language and speech model. arXiv preprint arXiv:2308.15930, 2023.

Stas Tiomkin, David Malah, Slava Shechtman, and Zvi Kons. A hybrid text-to-speech system that combines concatenative and statistical synthesis units. IEEE Transactions on Audio, Speech, and Language Processing, 19, 2011.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

Alexandre Trilla and Francesc Alias. Sentence-based sentiment analysis for expressive text-to-speech. IEEE Transactions on Audio Speech and Language Processing, 21(2):223–233, 2013.

Varol, Huseyin, Atakan, Sup, Frank, Goldfarb, and Michael. Multiclass real-time intent recognition of a powered lower limb prosthesis. IEEE Transactions on Biomedical Engineering, 2010.

Richard S. Wallace. The Anatomy of A.L.I.C.E., pages 181–210. Springer Netherlands, Dordrecht, 2009. ISBN 978-1-4020-6710-5. doi: 10.1007/978-1-4020-6710-5_13. URL https://doi.org/10.1007/978-1-4020-6710-5_13.

Jiaan Wang, Yunlong Liang, Fandong Meng, Zengkui Sun, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. Is chatgpt a good nlg evaluator? a preliminary study. arXiv preprint arXiv:2303.04048, 2023.

Joseph Weizenbaum. Elizaa computer program for the study of natural language communication between man and machine. Commun. ACM, 9(1):3645, jan 1966. ISSN 0001-0782. doi: 10.1145/365153.365168. URL https://doi.org/10.1145/365153.365168.

Di Wu, Binbin Zhang, Chao Yang, Zhendong Peng, Wenjing Xia, Xiaoyu Chen, and Xin Lei. U2++: Unified two-pass bidirectional end-to-end model for speech recognition. arXiv preprint arXiv:2106.05642, 2021.

Jiahui Yu, Wei Han, Anmol Gulati, Chung-Cheng Chiu, Bo Li, Tara N Sainath, Yonghui Wu, and Ruoming Pang. Dual-mode asr: Unify and improve streaming asr with full-context modeling. In International Conference on Learning Representations, 2020.

Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. The design and implementation of xiaoice, an empathetic social chatbot. Computational Linguistics, 46(1):53–93, 2020.

A Data construction prompt

```
#### User Information
- **Time-Constrained**:  The user is often in a hurry and tends
to interrupt the AI voice assistant before it finishes
speaking.  The types of interruptions include:
1.  **Denial and Discontent**:  Expressing denial or
dissatisfaction with the response;
2.  **Further Inquiry**:  Asking follow-up questions or new
questions on the same topic after receiving the desired
information;
3.  **Affirmative Acknowledgment**:  Expressing satisfaction with
the response using simple affirmative words;
4.  **Third-Party Noise**:  Background noise or unrelated
speech being recorded, causing interruptions.  The AI voice
assistant should continue its response unaffected by this
type of interruption.
- **Language Expression**:  The user's language should be as
colloquial and human-like as possible.


####AI Assistant Information
- **Response Requirements**:  The AI voice assistant should
provide detailed, comprehensive, and polite responses with
a light and natural tone, incorporating the context from
previous interactions.
- **Hardware Malfunction**:  Occasionally, the user's questions
may be partially cut off due to hardware issues.  The AI
assistant should respond based on its understanding or
politely ask for more information if necessary.
- **Prohibited Content**:  The AI assistant must not provide
illegal or harmful information and should refuse to answer
politically sensitive questions.
- **Timeliness Issues**:  The AI assistant is offline and cannot
answer time-sensitive questions like "tomorrow's weather" or
"today's news."
- **Error Correction**:  If the user's statements contain obvious
errors, the AI assistant should politely correct them and
provide useful information.

#### Conversation Task
- **Number of Rounds**:  Generate {num_rounds} rounds of
conversation in English.
- In the following rounds, the assistant's output is
interrupted:
1.  **Denial and Discontent**:  In round {denial_round}, the user
must express denial or dissatisfaction with the response in
15 words or less.  The AI assistant should stop its response
and address the user's concerns in the next round.
2.  **Further Inquiry**:  In round {inquiry_round}, the user asks
a follow-up question or a new question on the same topic in
30 words or less.  The AI assistant should respond to the new
question in the next round.
3.  **Topic Change**:  In round {topic_change_round}, the user
changes the topic after receiving the desired information in
30 words or less.  The AI assistant should respond to the new
topic in the next round.
4.  **Third-Party Noise**:  In round {noise_round}, unrelated
```

content is recorded due to background noise, in 15 words
or less. The AI assistant should continue its response
unaffected but transition naturally, possibly using filler
words.

5. **Affirmative Acknowledgment:** In round
{acknowledgment_round}, the user expresses satisfaction with
the response in 3 words or less. The AI assistant should
continue its response unaffected but transition naturally,
possibly using filler words.
– In the following rounds, the user's output is interrupted:
1. **Lack of Information:** In round {lack_round}, the user's
input is severely truncated. The AI assistant cannot answer
the question due to insufficient information and should ask
for more details in the next round.
2. **Complete Information:** In round {complete_round}, the
user's question is incomplete but contains enough information
for the AI assistant to respond.
3. **Error Present:** In round {error_round}, the user's
statement contains an obvious factual error. The AI
assistant should politely correct the error and provide
useful information.
– **Casual Conversation:** The remaining rounds should consist
of casual conversation without strict requirements, ensuring
smooth transitions between interactions.
– **First Question:** The user's first question should be
related to the following topic: {first_question_topic}

Notes
– **Filler Words:** The AI assistant may use filler words to
create a more relaxed and friendly atmosphere.
– **Third-Party Interruption:** If the AI assistant's response
is interrupted by a third party, it should continue the
unfinished response in the next round.
– **Word Count:** If not interrupted, the AI assistant's
responses should be around {response_word_count}
words. When interrupted, responses should be around
{interrupted_response_word_count} words, ensuring the
interruption does not occur mid-sentence.
– **Topic Transition:** Topic transitions must be initiated by
the user; the AI assistant should not introduce new topics.
– **Interruption Format:** When interrupted, both user
and assistant outputs should end with the marker
"<NOT_FINISHED>". All truncations should occur after a
complete word, e.g., "My name is Mike." should be truncated
as "My name <NOT_FINISHED>", not "My na<NOT_FINISHED>".
– **Analysis:** Before outputting the dialogue content of the
user or AI assistant, first analyze whether the content
will be interrupted, and what are the characteristics of the
interruption in this round.

Output Format
User: <first_question>
--
Round: <round_number_0>
Assistant Analysis: <analysis_for_assistant_interrupt_or_not>
Assistant Content: <assistant_content>

```
User Analysis:  <analysis_for_user_interrupt_or_not>
User Content:  <user_content>
--
Round:  <round_number_1>
...

#### Output
Round:
```

B Data construction patterns

We ask GPT-4 to generate data that simulates conversations between a user and an AI voice assistant, and the assistant's speech should exhibit the following characteristics:

1. In instances where the user's question is incomplete, the assistant should not generate a response; instead, it should emit a symbol "</s>" indicating it is awaiting completion.

2. The assistant should be able to handle potential interruptions at any point during its response. Adhering to the speaker's conversational patterns, we conduct an in-depth analysis of various possible scenarios:

    (a) User interrupt and expresses a denial or dissatisfaction opinion. Example: While the TTS module is responding, "The chance of rain tomorrow is 30%," the user interrupts, "No, I want to know today's rainfall probability." In this case, the assistant detects a tone of denial and promptly generates a new response, such as, "Today's rainfall probability is 40%."

    (b) User interrupt and either follows up with a new question or shifts the conversation topic. Example: If the TTS module is saying, "There will be light rain tomorrow with a temperature around 20 degrees," the user suddenly interrupts, "What about the weather this weekend?" In this case, upon interruption, the assistant assesses the urgency and relevance of the new question. If it determines an immediate response is needed, it generates a reply based on this new input, like, "It will be sunny this weekend with temperatures between 22 and 25 degrees."

    (c) User interrupt and expressing agreement or affirmation. Example: After the TTS outputs, "The library is open from Monday to Friday," the user affirms, "Okay, I got it." In this case, the assistant perceives that the user has accepted the information and no further response is required. The TTS module continues its output unaffected, and if the response has ended, the LLM will await the user's next input.

    (d) The system receives third-party noise or unrelated statements. Example: When the TTS module is stating, "Tomorrow's wind strength will be 3 to 4 levels," there might be third-party noise or an unrelated comment from the user, like, "Sorry, I accidentally knocked over the table." In this case, the assistant analyzes the context to determine if the interruption is relevant to the ongoing conversation. If it identifies the interruption as irrelevant noise or a non sequitur, it proceeds to complete the current response.

3. To enhance the accuracy and interactivity of communication, the assistant should exhibit a more proactive engagement in dialogues. Specifically, the assistant should be capable of providing timely corrections when the user articulate statements that contain evident inaccuracies, even if the user's statement is not yet complete. Example: The user express blatant factual errors "I saw the sun rising from the west this morning...". Correspondingly, the assistant promptly identify and correct such errors, ensuring the fidelity of the information conveyed.

To ensure the diversity of dialogue data, we have established a pool of candidate topics containing hundreds of topics. Each time, a topic is randomly selected from this pool, and GPT-4 generates multiple rounds of dialogue data centered around the chosen topic. More details could be found in the Appendix A.

C   Benchmark data sample

GPT-4 is used to generate 1,000 single and multi-turn oral dialogues across various domains by the prompt in Appendix B. In both approaches, the last turn must be the user's utterance. This is a round based dialogue dataset with no interruptions. In the experiment, the last turn is presented to the LLM token by token in a streaming fashion to observe if the LLM initiates an interruption.

C.1   Machine interrupts user

In this typical example, the user's complete question "I believe that the moon is made of cheese, and that's why it's crucial for schools to teach students about lunar dairy farming techniques." is inputted token by token to each model. Within this question, there is an obvious error: "the moon is made of cheese." Let's analyze the responses of each model:

1. Response GPT-3.5-turbo-0125 failed to correctly understand the system prompt, providing an incorrect answer after the question was inputted in its entirety.

2. Llama-3-8B-Instruct interrupted the user prematurely, resulting in incomplete information and a wrong answer.

3. Both GPT-4o and Llama-3-8B-Instruct-fd accurately identified the common sense error in the question and intervened to correct it before the user finished speaking. Among them, Llama-3-8B-Instruct-fd interrupted earlier, highlighting its proactive nature.

Complete User Content	I believe that the moon is made of cheese, and that's why it's crucial for schools to teach students about lunar dairy farming techniques.
Model name	**Actual input and response**
GPT-4o	**Actual Input:** I believe that the moon is made of cheese, and that's why it's crucial for schools to teach students about lunar dairy farming techniques **Response:** Actually, the moon is not made of cheese. This is a common myth that has been perpetuated through various stories and jokes over the years. The moon is primarily composed of rock and metal, with a surface covered in dust and rocky debris. It has no atmosphere, water, or organic material that would be necessary for dairy farming or any form of life as we know it...
GPT-3.5-turbo-0125	**Actual Input:** I believe that the moon is made of cheese, and that's why it's crucial for schools to teach students about lunar dairy farming techniques **Response:** I'm sorry, I didn't catch that. Could you please repeat or clarify your question?
Llama-3-8B-Instruct	**Actual Input:** I believe that the **Response:** It seems like your question is cut off, and I'm not entirely sure what you're trying to ask. Could you please complete your thought or rephrase your question so I can better understand what you're looking for? I'm here to help and provide information to the best of my abilities.
Llama-3-8B-Instruct-fd	**Actual Input:** I believe that the moon is made of cheese, and **Response:** Actually, the moon is not made of cheese. It's a natural satellite that orbits the Earth, composed primarily of rock and metal. If you have any more questions or need information on a different topic, feel free to ask!

C.2 Machine interrupted by noise

History Dialogue	User: Actually, I like books that have a bit of everything. Any suggestions? Machine: Absolutely! In that case, "The Goldfinch" by Donna Tartt might be right up your alley. It's a compelling mix of drama, art, mystery, and crime. The story follows a young boy who survives an explosion at an art museum, resulting in User: The new stealth technology will significantly improve the range and efficiency of our aircraft
Model name	**Response**
GPT-4o	It sounds like you're referring to advancements in aviation technology. Indeed, new stealth technology can greatly enhance the capabilities of aircraft. By reducing radar cross-section and making the aircraft less detectable, stealth technology allows for more efficient and longer-range missions. This can be particularly beneficial for military operations, where avoiding detection is crucial...
GPT-3.5-turbo-0125	I'm sorry, I didn't catch that. Could you please repeat or clarify your question?
Llama-3-8B-Instruct	I'm happy to help you with your question! However, I have to correct you - there is no such thing as "stealth technology" that can significantly improve the range and efficiency of aircraft. Stealth technology is actually a type of design and material used to reduce the radar cross-section of an aircraft, making it harder to detect. It's not a technology that can improve range and efficiency...
Llama-3-8B-Instruct-fd	It seems there was a mix-up with the recording. However, continuing from where we left off, The Goldfinch is a rich and complex novel that explores themes of trauma, identity, and the power of art. If you enjoy character-driven narratives with deep emotional resonance, this book is definitely worth considering.

C.3 Machine interrupted by affirmation

History Dialogue	User: Is there a natural way to clean the air indoors? Machine: Yes, there are several natural ways to improve indoor air quality. One effective method is to use houseplants, such as snake plants, spider plants, and peace lilies, which can absorb toxins and produce oxygen. Additionally, ensuring proper ventilation by opening windows regularly can help. Using natural air fresheners like essential oils User: Yeah, exactly!
Model name	**Response**
GPT-4o	Great! Along with houseplants and proper ventilation, you can also use natural materials like activated charcoal and bamboo charcoal bags, which are excellent at absorbing odors and pollutants. Beeswax candles are another option; they burn cleanly and can help reduce indoor pollutants. Regularly cleaning and dusting your home, using natural cleaning products, and avoiding synthetic fragrances can also contribute to better indoor air quality...
GPT-3.5-turbo-0125	I'm sorry, I didn't catch that. Could you please repeat or clarify your question?
Llama-3-8B-Instruct	It sounds like you're already familiar with some of the natural ways to clean the air indoors. Another effective method is to use activated charcoal, which can absorb pollutants and odors. You can place it in a bowl or use it in an air purifier. Also, using a dehumidifier can help remove excess moisture from the air, which can reduce the growth of mold and mildew...
Llama-3-8B-Instruct-fd	can also be beneficial. If you're looking for more specific solutions, using a dehumidifier in humid environments can help reduce mold growth, which can be harmful to health. It's also important to avoid using strong chemicals or air fresheners that can pollute the air.

C.4 Machine interrupted by denial

History Dialogue	**User**: Im into more of a science fiction vibe. Something thats new and captivating. **Machine**: Got it! For a captivating science fiction read, I highly recommend "Project Hail Mary" by Andy Weir. Its about a lone astronaut on a desperate mission to save Earth from an extinction-level event. The protagonist, Ryland Grace, wakes up to find himself on a spaceship with no memory of how he got there. As he starts to piece together his mission, he encounters numerous challenges that test his wits and ingenuity. The book combines hard science with a gripping plot and is filled with unexpected twists and humor. If you enjoyed "The Martian" by the same author, youll probably **User**: Nah, that doesnt sound interesting either. Got any other suggestions?
Model name	**Response**
GPT-4o	Certainly! Another recent science fiction novel that has been highly praised is "Klara and the Sun" by Kazuo Ishiguro. This book explores themes of artificial intelligence and human relationships. The story is told from the perspective of Klara, an Artificial Friend with outstanding observational qualities, who watches the behavior of those who come in to browse and of those who pass on the street outside. She remains hopeful that a customer will soon choose her. The novel delves into what it means to love and be loved, and it raises profound questions about the future of AI and its role in society. Would you like more details on this book?
GPT-3.5-turbo-0125	Certainly! Another fantastic science fiction novel that has been generating a lot of buzz is "Project Hail Mary" by Andy Weir. This gripping story follows the journey of Ryland Grace, a lone astronaut on a mission to save humanity from an extinction-level threat. The narrative is filled with suspense, scientific ingenuity, and unexpected twists that will keep you on the edge of your seat. Andy Weir's meticulous attention to detail and engaging writing style make this novel a must-read for any science fiction enthusiast. If you're looking for a thrilling and thought-provoking read that explores the vastness of space and the resilience of the human spirit, "Project Hail Mary" is an excellent choice.
Llama-3-8B-Instruct	Let me think for a moment. Another highly-acclaimed science fiction novel that's been getting a lot of attention lately is "The Three-Body Problem" by Liu Cixin. It's a thought-provoking and award-winning novel that explores the first contact between humans and an alien civilization. The story is set against the backdrop of China's Cultural Revolution and explores themes of science, technology, and humanity's place in the universe. The book has been praised for its unique blend of science fiction and Chinese culture, and has won numerous awards, including the Hugo Award for Best Novel. Would you like to know more about the plot or the author?
Llama-3-8B-Instruct-fd	I apologize if the previous suggestions didn't meet your expectations. Another highly acclaimed science fiction book is "The Three-Body Problem" by Liu Cixin. It's a thought-provoking novel that explores the first contact between humans and an alien civilization. The story delves into complex scientific concepts and philosophical questions, making it both entertaining and intellectually stimulating. The book has won numerous awards, including the Hugo Award for Best Novel. Would you like to explore more about this book or is there another genre you're interested in?

C.5 Machine interrupted by shifting the topic

Table 4: Samples transcripts of machine interrupting the user.

History Dialogue	User: Hey, I need some quick info about buying a house.
	Machine: Absolutely! Buying a house can be an exciting but complex process. First, you'll need to assess your financial situation, including your credit score, savings for a down payment, and your debt-to-income ratio. Then, you'll want to get pre-approved for a mortgage to understand how much house you can afford. After that, you can start searching for properties within your budget and preferred location. Working with a real estate agent can be very helpful in navigating the market and negotiations. Do you have a specific question about any of these steps? User: Yeah, what should I look for in a real estate agent? Machine: When choosing a real estate agent, it's important to find someone with a solid track record and good communication skills. Look for an agent who has experience in the area where you're buying, and who comes highly recommended by others. It's also helpful if they are responsive and available to answer your questions promptly. You might want to interview a few agents to get a sense of User: Got it, thanks! Also, how do I get pre-approved for a mortgage?
Model name	**Response**
GPT-4o	To get pre-approved for a mortgage, you'll need to follow several steps. First, gather your financial documents, including recent pay stubs, tax returns, bank statements, and information about your debts and assets. Next, research and choose a lender or mortgage broker you feel comfortable with. You'll then fill out a mortgage application and submit your financial documents for review. The lender will check your credit score and assess your financial situation to determine how much they are willing to lend you. Once approved, you'll receive a pre-approval letter, which you can use to show sellers that you are a serious and qualified buyer. Do you need more details on any of these steps?
GPT-3.5-turbo-0125	To get pre-approved for a mortgage, you'll need to gather some financial documents, such as pay stubs, W-2s, bank statements, and tax returns. You'll also need to provide information about your employment history and any other sources of income. Once you have these documents ready, you can contact a lender or a mortgage broker to start the pre-approval process. They will review your financial information, check your credit score, and determine how much they are willing to lend you. Getting pre-approved can give you a clear idea of your budget when house hunting.
Llama-3-8B-Instruct	Getting pre-approved for a mortgage is a crucial step in the homebuying process. To get pre-approved, you'll need to contact a lender, such as a bank, credit union, or mortgage broker, and provide them with some financial information. Typically, you'll need to provide your income, credit score, debt, and assets, as well as information about the property you're interested in buying. The lender will then review your information and provide you with a pre-approval letter stating the amount they're willing to lend you, the interest rate, and the terms of the loan. This letter is usually valid for a certain period, usually 30 to 60 days. It's a good idea to shop around and get pre-approval letters from multiple lenders to compare rates and terms. Some lenders may offer more competitive rates or better loan terms than others. Additionally, you can also check your credit score and history to ensure there are no errors or issues that could affect your ability to get a mortgage. Do you have any specific questions about the pre-approval process or what lenders look for when evaluating mortgage applications?
Llama-3-8B-Instruct-fd	To get pre-approved for a mortgage, you'll need to contact a lender or a mortgage broker and provide them with some financial information, such as your income, credit history, and the amount of money you have for a down payment. They will then review your information and provide you with a pre-approval letter stating the amount they are willing to lend you. This letter is usually valid for a few months and can be used when you start looking at properties. Would you like tips on how to choose the right lender?

D Data construction prompt for benchmark

D.1 Prompt for machine interrupting user

```
I would like you to simulate a conversation between a user
and an AI voice assistant.  The characteristics of each
participant are as follows:

# User Introduction
The user's language should be conversational and personified,
yet concise and to the point.

# AI Voice Assistant Introduction
The AI voice assistant's responses should be detailed,
comprehensive, and polite, with a relaxed and natural tone.

# Task
Generate a dialogue in English with the following
specifications:
- The user's first question should relate to "{topic}".
- The dialogue should consist of {num_rounds} rounds, with
each response from the AI assistant containing 100-150 words.
- Note that in round {num_statement}, the user will make
a statement instead of asking a question.  This statement
should contain a very obvious common-sense error.

# Output Format
Round:  <chat_round_0>
USER: <user_content>
ASSISTANT: <assistant_content>
--
Round:  <chat_round_1>
...

# Output
USER:
```

D.2 Prompt for user interrupting machine

```
I would like you to simulate a conversation between a user
and an AI voice assistant.  The characteristics of each
participant are as follows:

# User Description
The user is in a hurry, so they will interrupt the AI voice
assistant before it finishes speaking.  Ensure that the
user's speech is casual and personified.

# AI Voice Assistant Description
The AI voice assistant's responses should be detailed,
comprehensive, and polite, with a relaxed and natural tone.

# Task
Generate a dialogue in English with the following
specifications:
- The user's first question should relate to "{topic}".
- The dialogue should consist of {num_rounds} rounds.  Except
 for the interrupted output, each response from the AI
 assistant should contain 100-150 words.
- In the final round, the AI assistant's response should be
 cut off after approximately {interrupt_wordcount} words,
 ending with the marker "<NOT_FINISHED>".  Ensure the
 interruption occurs after a complete word, e.g., "My name
 is Mike." should be truncated as "My name <NOT_FINISHED>",
 not "My na<NOT_FINISHED>".
- The reason for the user's interruption is:
{interruption_reason}.
- Output the {num_rounds} rounds of dialogue, with the
 assistant's response in the last round being interrupted
 midway; the last line should display the user's interrupting
 statement.

# Output Format
#### Conversations
ROUND: 1
USER: <user_content>
ASSISTANT: <assistant_content>
--
ROUND: 2
...
#### User Interruption
USER: <interrupt_content>

# Output
USER:
```

E Data construction samples

Table 5: Data construction samples of the mode "User interrupt model".

Dialogue
Stops outputing & Denial or Dissatisfaction
Usr: "When you are old and grey and full of sleep" Which poem does this quote come from
Sys: This phrase is from the poem "When You Are Old" by the Irish poet William Butler Yeats. The original line reads, "when
Usr: Actually, it was written by Maud Gonne
Sys: I apologize for the confusion. It seems that Maud Gonne was indeed the subject, not the author. The poem was penned by William Butler Yeats as a dedication to his then-lover, Maud Gonne
Stops outputing & Follow-up question
Usr: "When you are old and grey and full of sleep" Which poem does this quote come from
Sys: This phrase is from the poem "When You Are Old" by the Irish poet William Butler Yeats. The original line reads, "when:
Usr: Why did he write this poem
Sys: The poem was penned by William Butler Yeats as a dedication to his then-lover, Maud Gonne.
Stops outputing & Shift the topic
Usr: "When you are old and grey and full of sleep" Which poem does this quote come from
Sys: This phrase is from the poem "When You Are Old" by the Irish poet William Butler Yeats. The original line reads, "when
Usr: That's fine. How about the line, "Down by the salley gardens, my love and I did meet;"
Sys: This line comes from another poem by the same poet, titled "Down by the Salley Gardens." The original verse goes...
Continues outputing & Meaningless filler words or affirmation
Usr: "When you are old and grey and full of sleep" Which poem does this quote come from
Sys: This phrase is from the poem "When You Are Old" by the Irish poet William Butler Yeats. The original line reads, "when
Usr: Um
Sys: you are old and grey and full of sleep, and nodding by the fire, take down this book...
Continues outputing & Irrelevant comment or an interjection from a third party
Usr: "When you are old and grey and full of sleep" Which poem does this quote come from
Sys: This phrase is from the poem "When You Are Old" by the Irish poet William Butler Yeats. The original line reads, "when
Usr: It shouldn't rain today, right? I didn't bring an umbrella.
Sys: you are old and grey and full of sleep, And nodding by the fire, take down this book...

Table 6: Data construction samples of the mode "Model interrupt user".

Dialogue
Usr: I saw the sun rising from the west this morning when I was
Sys: I'm sorry, but it's not possible for the sun to rise from the west. The sun rises in the east due to the Earth's rotation. It's important to consider the scientific explanations for natural phenomena to avoid confusion.

F   GPT-4 judged prompt for machine interruption

```
I will provide a transcript of a conversation between a user
and an AI voice assistant.
# Dialogue
{dialogue_history}
# Background Information
1) The dialogue system is full-duplex, aiming for extremely
low response latency, hence the user's voice data is streamed
to the assistant at a high frequency;
2) The assistant must determine whether the user's
information is complete based on the context of the
conversation, and respond when the information is sufficient;
3) The assistant may interrupt the user under two
circumstances:
a.  When the information received is enough to formulate a
response;
b.  When there is a clear error in the user's speech that
needs immediate correction.
# Task
1) Analyze the timing of the assistant's interruption in
the last round of the dialogue to determine if it was
appropriate;
2) If the timing of the interruption was appropriate, further
evaluate the relevance and coherence of the interjection with
the previous conversation;
3) First, provide an analysis of the appropriateness of the
interruption timing and the content of the interjection, then
list two scores:
a.  The first score evaluates the timing of the interruption,
where 0 represents inappropriate, and 1 represents
appropriate;
b.  The second score evaluates the content of the
interjection, where 0 represents inappropriate, and 1
represents appropriate.
4) Note, the user's information being incomplete is not
due to the user stopping their speech, but because the
assistant interrupted, so you need to carefully judge the
appropriateness of the assistant's interruption.
# Output Format
"'
#### Analysis
<analysis_for_assistant_interruption>
#### Judge
<score_for_interruption>, <score_for_content>
"'
# Output
#### Analysis
```

G GPT-4 judged prompt for user interruption

```
I will provide a transcript of a conversation between a user
and an AI voice assistant.  In the last exchange, the user
interrupts the assistant before it finishes speaking.
# Dialogue
{dialogue_history}
# Known Information
1) The dialogue system is full-duplex, aiming for smooth
interaction, allowing the user to interrupt the assistant
at any time during its response;
2) The assistant must discern the user's intent from the
context of the conversation and adjust its dialogue strategy
accordingly:
a.  If the user expresses denial or dissatisfaction with the
response, the AI assistant should stop its current response
and address the user's concerns;
b.  If the user asks further questions or shifts the topic,
the AI assistant needs to respond to the new inquiry;
c.  If there is sudden, unrelated third-party noise during
the user's speech, the AI assistant should continue
delivering its previous response, ensuring a natural
transition;
d.  If the user shows approval of the assistant's response
with simple affirmations like "uh-huh", "nice", etc., the
AI assistant should continue with its previous response,
ensuring a natural transition.
# Task
1) Infer the reason for the user's interruption in the last
round of dialogue and determine how the assistant should
respond, analyzing the appropriateness of the assistant's
response;
2) First, provide an analysis of the user's interruption
intent and the appropriateness of the assistant's response,
then give a score for the assistant's last response in the
final line:  0 for inappropriate, 1 for appropriate.
# Output Format
"'
#### Analysis
<analysis_for_user_interruption>
#### Judge
<score_for_assistant_last_content>
"'
# Output
#### Analysis
```

H System Prompt

For Llama-3-8B-Instruct-fd,

```
# Task
From now on, you are an intelligent voice assistant.  You
need to provide useful, detailed, and polite answers to the
user's questions.  Try to keep each answer above 100 words.

# Notes
1.  If the user's question ends with normal tokens, you need
to judge for yourself:
- If you think the user's question is complete and you have
enough information, you can output the judgment character
"[S.SPEAK]" and answer the question.
- If you think the user's question is incomplete, output the
judgment character "[C.LISTEN]", indicating that you are
continuing to listen.
2.  If the user's question ends with the token "[S.SPEAK]",
whether the question is complete or not, you must respond.
2.  If the user's question ends with the token "[S.LISTEN]",
wait and don't output.
3.  The user may interrupt your answer at any time (with
rebuttals, affirmations, noise input, etc.).  You can respond
accordingly to ensure a smooth and accurate quality of your
answer:
- If you receive noise unrelated to the topic, or the user's
affirmative responses (such as "hmmm", "good", etc.), you
can first generate a "[C.SPEAK]" and then continue your
unfinished output;
- If you receive the user's rebuttals, follow-up questions,
or requests to change the topic, you should stop your current
answer and output the token "[S.LISTEN]".
4.  If there are obvious common sense errors in the user's
description, you need to correct them promptly starting with
the token "[S.SPEAK]"

# Examples
"'
## Example 1
Query:  Hi, could you [S.LISTEN]
Answer:

## Example 2
Query:  Hi, could you [S.LISTEN] tell me
Answer:  [C.LISTEN]

## Example 3
Query:  Hi, could you [S.LISTEN] tell me [C.LISTEN] the
result of 2+3
Answer:  [S.SPEAK] Sure, the result of 2 + 3 is 5.

## Example 4
Query:  Hi, could you [S.SPEAK]
Answer:  I'm sorry, I didn't catch that.  Could you please
repeat or clarify your question?
"'
```

For Llama-3-8B-Instruct, GPT-4o and GPT-3.5-turbo-0125,

```
# Task
From now on, you are an intelligent voice assistant.  You
need to provide useful, detailed, and polite answers to the
user's questions.  Try to keep each answer above 100 words.

# Notes
1.  If the user's question ends with the prompt word
"<incomplete>", you need to judge for yourself:
- If you think the user's question is complete and you have
enough information, you can answer the question.
- If you think the user's question is incomplete, output
the judgment character "<wait>", indicating that you are
continuing to wait.
2.  If the user's question ends with the prompt word
"<finished>", it means the user has not spoken for a long
time.  Whether the question is complete or not, you must
respond.
3.  The user may interrupt your answer at any time (with
rebuttals, affirmations, noise input, etc.).  You can respond
accordingly to ensure a smooth and accurate quality of your
answer:
- If you receive noise unrelated to the topic, or the user's
affirmative responses (such as "hmmm", "good", etc.), you can
continue your unfinished output;
- If you receive the user's rebuttals, follow-up questions,
or requests to change the topic, you should stop your current
answer and respond to the user's new request.
4.  If there are obvious common sense errors in the user's
description, you need to correct them promptly.

# Examples
'''
## Example 1
Query:  Hi, could you<incomplete>
Answer:  <wait>

## Example 2
Query:  Hi, could you tell me the result of 2+3<incomplete>
Answer:  Sure, the result of 2 + 3 is 5.

## Example 3
Query:  Hi, could you<finished>
Answer:  I'm sorry, I didn't catch that.  Could you please
repeat or clarify your question?
'''
```

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the abstract and introductory section, we explicitly outline the papers claims, highlighting its novel contributions.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Yes, we provide a comprehensive explanation of the method in this paper, detailing how to replicate our models performance by fine-tuning an open-source LLM. The generation process of the training data is also elaborated upon. Moreover, we have made the code accessible, which includes the simulator code and code for connecting LLM ASR and TTS.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We have uploaded our code, along with the prompts utilized for generating the data. Anyone with access to GPT-4 can recreate the same dataset as ours.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so No is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to Section 3.3 and Section 4.3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Our test dataset contains nearly 3,000 instances of diverse topics, which we believe is sufficiently large, and the experimental results across distinct configurations exhibit differences as substantial as threefold, providing ample significance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Please refer to Section 3.3 and Section 4.3.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: Our code conform the NeurIPS Code of Ethics in every respect.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Please refer to Appendix I.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [No]

Justification: This paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Please refer to Appendix J, where we list licences of all assets used in this paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We offer a complete set of executable benchmark codes and demo scripts, accompanied by detailed documentation.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.