ESPnet-SDS: Unified Toolkit and Demo for Spoken Dialogue Systems

Siddhant Arora[1], **Yifan Peng**[1], **Jiatong Shi**[1], **Jinchuan Tian**[1],
William Chen[1], **Shikhar Bharadwaj**[1], **Hayato Futami**[2], **Yosuke Kashiwagi**[2],
Emiru Tsunoo[2], **Shuichiro Shimizu**[1,3], **Vaibhav Srivastav**[4], **Shinji Watanabe**[1]

[1] Carnegie Mellon University, USA, [2] Sony Group Corporation, Japan
[3] Kyoto University, Japan, [4] Hugging Face, USA
{siddhana}@cs.cmu.edu

Abstract

Advancements in audio foundation models (FMs) have fueled interest in end-to-end (E2E) spoken dialogue systems, but different web interfaces for each system makes it challenging to compare and contrast them effectively. Motivated by this, we introduce an open-source, user-friendly toolkit designed to build unified web interfaces for various cascaded and E2E spoken dialogue systems. Our demo further provides users with the option to get on-the-fly automated evaluation metrics such as (1) latency, (2) ability to understand user input, (3) coherence, diversity, and relevance of system response, and (4) intelligibility and audio quality of system output. Using the evaluation metrics, we compare various cascaded and E2E spoken dialogue systems with a human-human conversation dataset as a proxy. Our analysis demonstrates that the toolkit allows researchers to effortlessly compare and contrast different technologies, providing valuable insights such as current E2E systems having poorer audio quality and less diverse responses. An example demo[1] produced using our toolkit is publicly available here: https://huggingface.co/spaces/Siddhant/Voice_Assistant_Demo.

1 Introduction

Spoken Dialogue Systems (SDS) (Défossez et al., 2024; Raux et al., 2006) are designed to engage in natural and interactive conversations with end users. These systems play a critical role in commercial applications such as voice assistants (Li et al., 2017) and intelligent home devices (Coucke et al., 2018). However, this is a challenging task due to the complexity and variability of human-AI communication (Glass, 1999). Spoken dialogue systems typically consist of multiple modules, including voice activity detection (VAD) (Wiseman, 2024), automatic speech recognition (ASR) (Radford et al.,

2023; Peng et al., 2023), natural language understanding (NLU) (Mehri et al., 2020) and generation (NLG) (Peng et al., 2020), and text-to-speech (TTS) synthesis (Coqui, 2024), each presenting unique challenges. For instance, the ASR module must accurately process shorter, spontaneous speech that often contains disfluencies and filler words (Godfrey et al., 1992). Additionally, spoken dialogue systems must understand non-phonemic information (Rashkin, 2018), such as emotions, and generate non-phonemic elements (Sundaram and Narayanan, 2007) like laughter to make interactions feel more natural. A critical aspect of these systems is managing the flow of conversation i.e. performing fluent turn-taking without excessive overlapping speech or prolonged silences (Gravano and Hirschberg, 2011). Furthermore, effective dialogue systems must be capable of handling long conversational contexts to ensure accurate and context-aware responses (Arora et al., 2023).

Historically, spoken dialogue systems (Glass, 1999; Huang et al., 2024a) have been modeled using a cascaded architecture that incorporates various modules such as ASR, NLU, NLG, and TTS. Recent advances have led to the development of full-duplex end-to-end (E2E) spoken dialogue systems (Défossez et al., 2024; Xie and Wu, 2024) capable of handling both user input and system-generated audio simultaneously. Since the main goal of dialogue systems is to enhance the experience of human AI interaction, these dialogue systems are often released with web demos to demonstrate alignment with human conversational expectations. However, there is no unified web interface for users to directly interact with different spoken dialogue systems and compare the relative utility of each system. Each spoken dialogue system often has its own web interface with different ways to take user audio as input, different backend computing infrastructure, and sometimes even different approaches to postprocess and show model output.

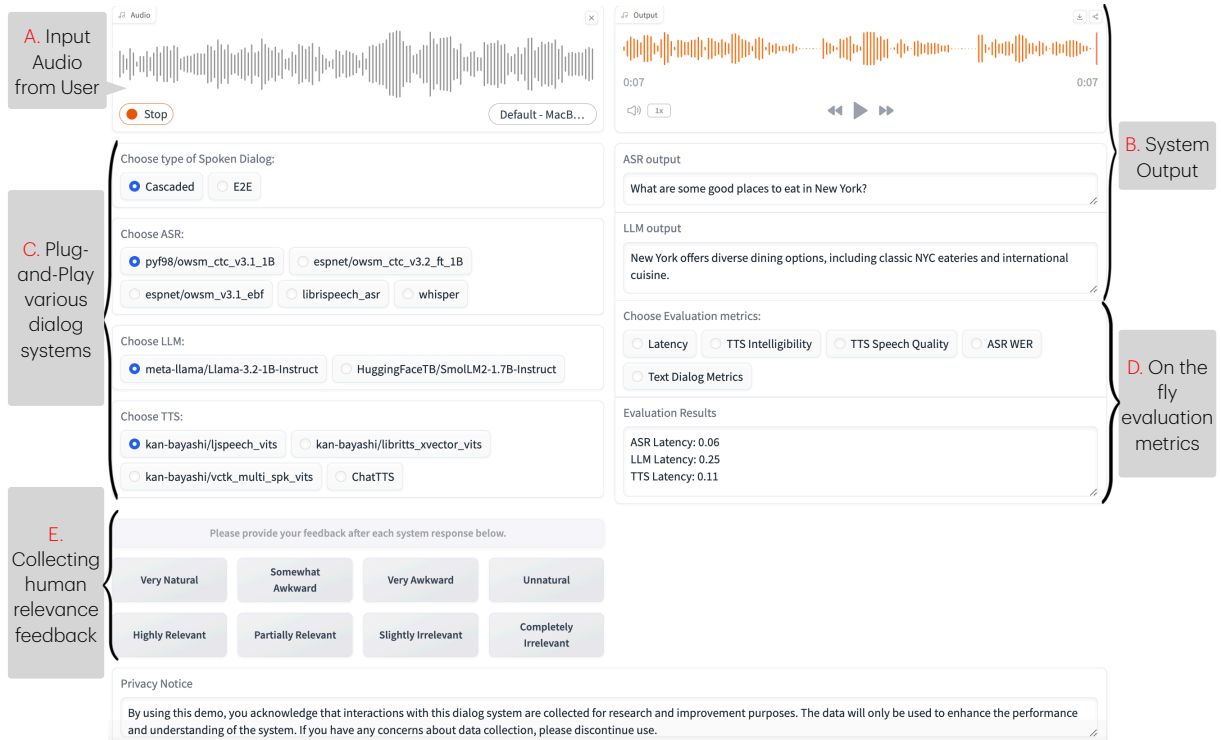[1] Demo Video: https://youtu.be/kI_DXwf5qPk

Figure 1: Screenshot of our unified web interface highlighting key features: (a) streaming audio input, (b) display of ASR transcripts, text responses, and synthesized audio outputs, (c) interaction with cascaded and E2E dialogue systems and experimentation with ASR, LLM, and TTS submodules, (d) on-the-fly evaluation metrics, and (e) collection of human feedback on the naturalness and relevance of system outputs.

This lack of standardization makes it challenging to identify the strengths and weaknesses of each system and to determine effective design choices for developing robust spoken dialogue systems. As the number of spoken dialogue systems and methodologies continues to grow, the need for an open-source toolkit that standardizes the process of creating web interfaces for these systems has become increasingly clear. Moreover, there have also been limited efforts to comprehensively evaluate spoken dialogue systems on their conversational capability. As a result, the community lacks an understanding of how best to evaluate these systems.

To address this, we introduce the *ESPnet-SDS* toolkit built on an existing open-source speech processing toolkit ESPnet (Watanabe et al., 2018) with the aim of providing researchers with easy access to existing cascaded and E2E spoken dialogue methodologies within a unified framework. Our goal is to establish a comprehensive open-source standard, enabling researchers to seamlessly integrate existing technologies, compare new ideas, and benchmark their systems against current methodologies. The key contributions of our toolkit

are : ① Facilitating the development of Gradio[2]-based demo interfaces, allowing users to interact with and evaluate multiple cascaded and E2E spoken dialogue systems through a standardized platform (Fig. 1(C)). ② Providing implementations of automated metrics for submodule-level (e.g., ASR, text dialogue, and TTS) as well as conversation-level evaluations (Fig. 1(D)). ③ Enabling human-in-the-loop evaluation by including mechanisms for collecting user feedback on the naturalness and relevance of system responses (Fig. 1(E)).[3]

Our initial experiments on the human-human conversation dataset, Switchboard (Godfrey et al., 1992), highlight the strengths and weaknesses of different spoken dialogue systems, offering actionable insights into the limitations of E2E approaches compared to traditional cascaded pipelines such as current E2E systems generating poorer audio quality and less diverse responses. We hope these findings inspire the development of more robust spoken dialogue systems and evaluation methodologies. The ESPnet-SDS toolkit is publicly available at https://github.com/espnet/

---

[2]https://www.gradio.app/

[3]Storing user data is entirely optional and intended only for researchers who have obtained appropriate consent (Sec. 10).

System	Speech Input	Speech Output	Multi-Turn Interaction	Multiple Systems	On-the-fly Evaluation	Dialogue System Type Cascaded	E2E
ChatBot Arena (Chiang et al., 2024)	✗	✗	✓	✓	✗	✗	
TTS Arena (Face, 2024b)	✗	✓	✗	✓	✗	✗	
TalkArena (Li et al., 2024)	✓	✗	✗	✓	✗	✗	
Speech to Speech (Face, 2024a)	✓	✓	✗	✗	✗	✓	✗
Moshi (Défossez et al., 2024)	✓	✓	✗	✗	✗	✗	✓
ESPnet-SDS	✓	✓	✓	✓	✓	✓	✓

Table 1: Comparison of Web Interfaces for Dialogue and Speech Processing Systems

espnet, with an example web interface shown in Fig. 1 featuring various cascaded and E2E systems accessible at https://huggingface.co/spaces/Siddhant/Voice_Assistant_Demo.

2 Related Work

Spoken dialogue systems often include web interfaces that enable users to interact with these systems. However, these interfaces typically feature diverse input/output frontends and backend computing environments, making it difficult to compare systems. This highlights the need for standardized platforms to benchmark systems on a level playing field. Similar initiatives in text domain, such as ChatBot Arena (Chiang et al., 2024), have gained significant popularity for benchmarking large language model (LLM)-based chatbots using human preference evaluations. Inspired by this, TTS Arena (Face, 2024b) has provided a valuable platform for comparing open-source and proprietary TTS models. Recently, TalkArena (Li et al., 2024; Artificial Analysis, 2025) was introduced to benchmark audio foundation models (FMs) on metrics like the naturalness and coherence of their responses. However, TalkArena is currently limited to an audio-in, text-out format and supports only single-turn interactions.

To the best of our knowledge, this work presents the first web interface that supports multiple spoken dialogue systems—encompassing both cascaded and E2E methodologies—within a unified framework. These systems are capable of automatically performing turn-taking, generating spoken outputs, and engaging in multi-turn, natural conversations with users. By standardizing the interface and evaluation pipeline, our work not only simplifies the process of testing and comparing dialogue systems but also enhances the accessibility and reproducibility of research in this domain. Tab. 1 highlights the unique capabilities and advantages of our proposed web interface compared to prior works.

```
Wrappers (Python)
espnet2/sds

#ASR modules
AbsASR
    ESPnet_ASR
    OWSM_ASR
    OWSM-CTC_ASR
    Whisper_ASR

#Text dialogue modules
AbsLLM
    HuggingFace_LLM

#TTS modules
AbsTTS
    ESPnet_TTS
    Chat_TTS

#VAD modules
AbsVAD
    WebRTC_VAD

#E2E Dialogue modules
AbsE2E
    MiniOmni_E2E

#Interface wrapper for
connecting modules
ESPnet_SDS_Model_
Interface
```

```
Template (Python)
egs2/TEMPLATE/sds1
    # Evaluation Metric
    —- pyscripts/utils/dialog_eval

    #ASR module specific
    ASR_WER

    #Text dialogue module specific
    LLM_Metric

    #TTS module specific
    TTS_Intelligibility
    TTS_Speech_Quality

    #Human-in-the-loop evaluation
    Human_Feedback

#Template for Gradio demo using
Interface wrapper & Metrics
App.py

# User guide
README.md
```

```
Recipe (Bash)
# Example
egs2/spoken_chatbot_arena/sds1

#Launch gradio demo using App.py
based on arguments
Run.sh
```

Figure 2: Modular software architecture of ESPnet-SDS

3 System Design

This section outlines the design of the ESPnet-SDS toolkit and its key features. The toolkit adopts a modular design, as shown in Fig. 2. Within ESPnet-SDS, major modules for VAD, ASR, LLM (for text-based dialogue response generation) and TTS are implemented as wrapper classes (Sec.4.1) under espnet2/sds. Similarly, the directory also provides helper functions to build wrappers (Sec. 4.2) for open source E2E spoken dialogue systems using their publicly available checkpoints and inference codebase. All module wrappers are then utilized within interface wrapper class ESPnet_SDS_Model_Interface. This wrapper class provides a unified interface to integrate ASR, TTS, and LLM modules for cascaded spo-

ken dialog systems as well as also supports E2E spoken dialog systems. It further enables real-time interactions, including VAD based conversation management.

To facilitate benchmarking, the toolkit provides functionality for computing major evaluation metrics at both the sub-task level (e.g., ASR, text response generation, and TTS) and the overall conversation level (Sec. 5). Recognizing the importance of human-in-the-loop evaluation, the demo includes functionality to collect human feedback on metrics such as the naturalness and relevance of system responses. Additionally, the web interface can store human judgments along with interaction data such as user input audio and system output recordings (Sec. 6), but this functionality is disabled by default and intended only for studies with appropriate consent (Sec. 10).

All evaluation metric functions are implemented under the `pyscripts/utils/dialog_eval` directory within "Template" module (`egs2/TEMPLATE/sds1`, as shown in Fig. 2). This module also demonstrates how the interface wrapper and evaluation tools can be utilized to construct a Gradio-based spoken dialogue system demo. The `App.py` file within this module outlines the entire pipeline—from processing user audio input, passing it through the *selected* spoken dialogue system, to presenting the synthesized output and associated evaluation metrics to the user. Additionally, a `README` file is included that provides an overview of ESPnet-SDS's features and detailed guidance on using the demo interface.

The "Template" module serves as a practical starting point for researchers, allowing them to leverage the `sds1` template and create customized recipes by passing appropriate arguments to `Run.sh`. These recipes can facilitate comparisons between E2E spoken dialogue systems and traditional cascaded pipelines, as well as test different ASR, LLM, and TTS model combinations within cascaded systems. Additionally, they enable on-the-fly comparisons using the supported evaluation metrics. We include an example recipe in `egs2/spoken_chatbot_arena/sds1`, which can be used to run our demo (`https://huggingface.co/spaces/Siddhant/Voice_Assistant_Demo`) locally.

The toolkit is designed for ease of extension. When new E2E dialogue systems are introduced, authors or contributors can submit pull requests (PR) to add wrapper functions for their open-source models. Corresponding recipes can then be added to PR, integrating the new systems into the unified demo alongside existing models.

4 Example Models

4.1 Cascaded Models

As discussed in Sec. A.1, spoken dialogue systems have traditionally been modeled using a cascaded pipeline comprising several submodules:

Voice Activity Detection (VAD): The VAD module identifies when the user has finished speaking, allowing the AI system to take the conversational floor. Our toolkit is designed to support any open-source VAD, and we provide wrapper classes for WebRTC VAD (Wiseman, 2024) as an example.

Automatic Speech Recognition (ASR): The ASR module transcribes spoken utterances into text. Our toolkit integrates seamlessly with *294* ASR models, including ESPnet models, ESPnet's speech foundation models (Peng et al., 2024b) including CTC-based (Peng et al., 2024a), and other open-source foundation models like Whisper (Radford et al., 2023).

Text dialogue response generator: Our toolkit supports the integration of any of the *36,464* LLM available on Hugging Face[4] to process ASR transcripts and generate appropriate text responses.

Text-to-Speech (TTS): The TTS module synthesizes audio output from the generated text response, completing the conversational loop. Our toolkit provides wrapper classes to integrate 80 single-speaker and multi-speaker models from ESPnet (Hayashi et al., 2020) and supports open-source TTS systems like ChatTTS[5], ensuring compatibility with a wide array of models.

4.2 E2E Spoken Dialogue System

Recently, several E2E spoken dialogue models (Défossez et al., 2024; Fang et al., 2024) have been introduced, capable of directly processing user audio input and synthesizing system audio output. Our toolkit offers a simple and user-friendly interface for integrating existing E2E dialogue systems by loading model weights from its publicly available checkpoints and leveraging its open-source inference codebases. As an example, we provide wrapper classes for Mini-Omni (Xie and Wu, 2024), enabling it to function as an E2E conversational

---

[4] `https://huggingface.co/models`
[5] `https://github.com/2noise/ChatTTS`

assistant when paired with a VAD module for turn-taking. Moving forward, we plan to expand the codebase to support additional open-source E2E audio FMs with conversational capabilities.

5 Evaluation Metrics

To evaluate the relative utility of different cascaded spoken dialogue systems, we support the implementation of various automatic evaluation metrics:

TTS Module Specific: We utilize the VERSA[6] toolkit (Shi et al., 2024) to assess the intelligibility and quality of the system's audio output. Intelligibility is evaluated using hypotheses generated by 3 ASR models: ESPnet Librispeech ASR (Watanabe et al., 2018), OWSM 3.1 (Peng et al., 2023), and Whisper (Radford et al., 2023). Audio quality is measured using UTMOS (UTokyo-SaruLab System for VoiceMOS 2022 (Saeki et al., 2022)), DNS_overall and DNS_P808 (Deep Noise Suppression MOS Scores from P.835 and P.808 (Reddy et al., 2021, 2022)), PLCMOS (Packet Loss Concealment MOS (Diener et al., 2023)), and SSQA (Subjective Speech Quality Assessment (Huang et al., 2024b)).

ASR Module Specific: These metrics evaluate the system's ability to understand users' spoken utterances. In the absence of ground truth transcripts, we generate reference transcripts from 3 judge ASR models: ESPnet Librispeech ASR (Watanabe et al., 2018), OWSM 3.1 (Peng et al., 2023), and Whisper (Radford et al., 2023). The ASR transcripts generated by the cascaded system are then compared against these references to compute Word Error Rate (WER) and Character Error Rate (CER).

Text Dialogue Module Specific: Following prior work (Nguyen et al., 2023), we compute perplexity using GPT-2 (Radford et al., 2019) and VERT (Lakhotia et al., 2021) to evaluate the coherence and diversity of system outputs, respectively. Additionally, the DialoGPT (Zhang et al., 1911) model is used to compute the perplexity of system responses *given the user utterance*. To evaluate context modeling, we compute the similarity between the system response and the entire prior dialogue context using BERT (Devlin et al., 2018).

Conversation Level Specific: While the previous metrics focused on evaluating specific modules of the cascaded dialogue system and can be computed at each turn, we also support conversation-level metrics (see Sec. A.5 in Appendix) to assess

the system's ability to manage conversational flow. Inspired by prior work (Nguyen et al., 2023), these metrics include turn-taking statistics, speaking rate (words per minute), and backchannel rate (number of backchannel words[7] per minute).

E2E spoken dialogue models usually generate both text and audio tokens, enabling the computation of conversation-level metrics and some module-specific metrics. Notably, TTS-specific metrics and dialogue metrics like perplexity and VERT can be computed without the need for the system to generate ASR transcript.

Human Evaluation: We implemented a feedback mechanism allowing users to assess the system's performance. Drawing on prior work (Nguyen et al., 2023) that collects opinion scores, users can rate the system on two key aspects: the naturalness of the synthesized output and the relevance of the system's response within the dialogue context. Ratings are customizable and provided on a 4-point scale by default, as in Fig. 1(E), ranging from "Very Natural" to "Unnatural" and "Highly Relevant" to "Completely Irrelevant".

6 Web Interface

Fig. 1 shows a screenshot of an example Gradio web demo built using our toolkit. The web demo collects the user audio in a streaming manner (Fig. 1(A)), with a VAD module detecting when the user stops speaking. The processed audio is passed to the dialogue system. For cascaded systems, the output (Fig. 1(B)) includes the system's audio response and 2 text boxes displaying the ASR transcript and generated text response[8]. User audio is continuously collected to allow interruptions, enhancing interactivity. Conversations are limited to 5 minutes, after which the interface refreshes. Additional functionalities are discussed below.

Plug-and-Play Architecture: Our web interface features a plug-and-play architecture (Fig. 1(C)) that allows users to select from a *variety of cascaded and E2E* dialogue systems, enabling a total of *41* different system variations. Details are provided in Tab. 5 in the Appendix. To avoid out-of-memory errors, models are loaded on the fly, enabling seamless switching between systems. However, loading a new model may take some time,

---

[7]Following Wang et al., we annotate backchannels using common one- and two-word phrases.

[8]For E2E systems, the ASR transcript box is blank as these systems do not generate ASR transcripts.

ASR Model	WER ($\downarrow$)	CER ($\downarrow$)
Librispeech ASR	342.2	298.0
Whisper (large)	22.5	22.3
OWSM (3.1)	14.4	11.2
OWSM CTC (3.1)	18.1	14.4
OWSM CTC (3.2)	15.9	10.3

Table 2: WER and CER of various ASR models that can be used in the cascaded system.

Metric	LLaMA 3.2-1B	SmolLM v2-1.7B	Mini-Omni
Perplexity ($\downarrow$)	48.2	113.7	18.7
Diversity			
Self BLEU-2 ($\downarrow$)	75.9	77.1	93.3
Auto BLEU-2 ($\downarrow$)	0.4	0.3	6.2
VERT ($\downarrow$)	5.7	5.0	24.1
BERT Similarity ($\uparrow$)	55.8	49.5	61.5
DialoGPT Perplexity ($\downarrow$)	165.9	301.1	125.7

Table 3: Text dialogue metrics of various LLM that can be used in the cascaded system along with Mini-Omni.

and the system output boxes are temporarily hidden during this process.

On-the-fly evaluation: Our demo supports real-time display of module-specific metrics (Sec. 5) for each *turn*, including latency for each component. Notably, user data is not recorded for on-the-fly evaluation; however, users may experience a brief wait for results. These metrics allow users to also gain quantitative insights into system performance, such as latency and audio quality. We encourage users to leverage these metrics during their interaction to pinpoint specific differences between dialogue systems, facilitating actionable insights.

Database Interface: The toolkit also offers optional integration with a remote HuggingFace dataset[9] as a backend database, allowing researchers to store human relevance judgments and log user interaction data, including input recordings and system outputs such as ASR transcripts, text responses, and audio responses. This data enables analysis of spoken dialogue system's performance in realistic human-AI interactions, as outlined in Sec. A.5. To address privacy and ethical considerations, this functionality is disabled by default in the publicly available demo. However, researchers can enable this functionality provided appropriate consent is obtained. A privacy notice is also displayed in the demo to inform users that their data may be recorded and collected for analysis.

---

[9] https://huggingface.co/datasets

7 Analysis using Evaluation Metrics

To demonstrate the utility of our supported automatic evaluation metrics (Sec.5), we use them to compare and analyze different dialogue systems and their submodules. This evaluation is conducted on the publicly available Switchboard (Godfrey et al., 1992) Eval 2000 dataset, comprising 11 hours of human-human conversation data. Full descriptions of the evaluation setup are in Sec. A.2.

ASR Module Metrics: We evaluate various ASR models by providing audio recordings segmented into utterances as input and computing WER and CER against ground truth transcripts. Table 2 shows that ① OWSM 3.1 (Peng et al., 2024b) achieves the best performance, ② closely followed by the faster OWSM CTC 3.2 (Peng et al., 2024a).

Text dialogue Module Metrics: To evaluate various LLMs in a cascaded pipeline, these models are provided with dialogue context generated from ground truth user transcripts and prompted to produce responses based on the context. Table 3 indicates that while ① LLaMA generates less diverse responses compared to SmolLM v2, ② its responses are more coherent and relevant to the dialogue context. We also evaluate the E2E spoken dialogue system Mini-Omni by inputting user utterance audio and computing these LLM metrics based on its generated text responses. Our analysis shows that ③ Mini-Omni produces highly coherent and contextually relevant responses; however, ④ it exhibits significant overlap in its responses, often generating template-like outputs. This behavior could make conversations feel less natural when deployed in real-world applications.

TTS Module Metrics: We evaluate various TTS models by providing ground truth transcripts as input and computing intelligibility and audio quality metrics for the synthesized output. Table 4 shows that the ① LJSpeech VITS TTS model produces the most intelligible speech and generally achieves higher audio quality, however it is limited to only single-speaker, whereas other models offer the advantage of modeling multiple speakers. Additionally, we compute these metrics for the synthesized audio of the E2E spoken dialogue system Mini-Omni. However, the E2E dialogue system takes the audio of user utterances as input and hence is not directly comparable to other TTS models. Our results indicate that while ② Mini-Omni's synthesized speech is intelligible, ③ its audio quality is significantly lower than that of other TTS models.

	Intelligibility				Speech Quality				
	OWSM 3.1		Whisper		UTMOS (↑)	DNS (↑)		PLCMOS (↑)	SSQA (↑)
	WER (↓)	CER (↓)	WER (↓)	CER (↓)		Overall	P808		
LJSpeech VITS	16.1	9.8	15.1	10.1	4.06	3.05	3.79	4.50	4.01
LibriTTS VITS	19.3	11.5	16.5	10.4	3.97	3.10	3.49	4.34	4.19
VCTK VITS	27.0	16.8	22.1	14.7	4.07	3.08	3.63	4.62	4.12
ChatTTS	19.1	15.5	17.4	14.3	3.05	2.98	3.35	3.86	3.29
Mini-Omni**	18.2	17.9	4.4	4.1	2.88	2.79	3.92	2.52	2.24

Table 4: TTS Module metrics of TTS models that can be used in the cascaded system and E2E spoken dialogue Mini-Omni. ** As noted in Sec. 7, TTS models and Mini-Omni are not directly comparable since TTS models use ground truth transcripts as input, while Mini-Omni generates audio responses based on user utterance audio.

8 Conclusion and Outlook

To conclude, this work introduces ESPnet-SDS, an open-source toolkit for creating user-friendly and interactive web interfaces for spoken dialogue systems. It provides the infrastructure to interact with a variety of cascaded and E2E spoken dialogue systems using a unified, Gradio-based demonstration. The modular design of our codebase ensures easy extensibility to support more dialogue systems in the future. Through initial evaluations, we demonstrate that the supported evaluation metrics offer valuable insights into the strengths and limitations of existing systems. We hope that this work inspires future research and serves as a useful resource for researchers aiming to compare and refine design choices in building spoken dialogue systems.

9 Limitation

While our proposed system demonstrates considerable utility in benchmarking and evaluating spoken dialogue systems, certain areas require further exploration. Extensive real-world testing can provide deeper insights into the tool's usability and robustness. Specifically, additional investigation is needed to assess its performance in low-resource language settings, noisy environments, and multi-speaker scenarios. Additionally, while the current range of supported audio foundation models (FMs) is limited, the toolkit is designed for extensibility. We are committed to continuously expanding its capabilities by incorporating new dialogue systems and audio FMs as they become available.

10 Broader Impact and Ethics

The small-scale pilot study (Sec. A.5) conducted in this work involved only the authors and their research collaborators, all of whom participated voluntarily and were informed that their interactions with the AI dialogue system would be recorded. While our proposed web interface supports human-in-the-loop evaluation, data collection is disabled by default. This optional functionality is designed solely for researchers who have obtained appropriate consent. To ensure transparency, a privacy notice is included in the web interface, notifying users that their data may be recorded. Importantly, the interface also generates on-the-fly evaluation metrics that do not require any user data to be stored. Finally, all modules of the cascaded and E2E spoken dialogue systems used in this work are open source, and our evaluations are conducted using the publicly available Switchboard human-human conversation dataset (Godfrey et al., 1992).

References

Siddhant Arora, Hayato Futami, Emiru Tsunoo, Brian Yan, and Shinji Watanabe. 2023. Joint modelling of spoken language understanding tasks with integrated dialog history. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Artificial Analysis. 2025. Arena: Text-to-Speech Benchmark. Accessed: 2025-03-04.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, et al. 2024. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*.

Coqui. 2024. Introducing open-xtts: An open-source toolkit for tts. Accessed: 2024-12-10.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *CoRR*, abs/1805.10190.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. 2024. Moshi: a speech-text foundation model for real-time dialogue. Technical report, Kyutai.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Lorenz Diener, Marju Purin, Sten Sootla, Ando Saabas, Robert Aichner, and Ross Cutler. 2023. Plcmos – a data-driven non-intrusive metric for the evaluation of packet loss concealment algorithms. In *INTERSPEECH 2023*, pages 2533–2537.

Hugging Face. 2024a. Speech-to-speech translation toolkit. Accessed: 2024-12-10.

Hugging Face. 2024b. Tts arena: Benchmarking text-to-speech models in the wild. Accessed: 2024-12-13.

Qingkai Fang, Shoutao Guo, Yan Zhou, Zhengrui Ma, Shaolei Zhang, and Yang Feng. 2024. Llama-omni: Seamless speech interaction with large language models. *arXiv preprint arXiv:2409.06666*.

James Glass. 1999. Challenges for spoken dialogue systems. In *Proceedings of the 1999 IEEE ASRU Workshop*, volume 696. MIT Laboratory for Computer Science Cambridge.

J.J. Godfrey, E.C. Holliman, and J. McDaniel. 1992. Switchboard: telephone speech corpus for research and development. In *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520 vol.1.

Agustín Gravano and Julia Hirschberg. 2011. Turn-taking cues in task-oriented dialogue. *Comput. Speech Lang.*, 25(3):601–634.

Tomoki Hayashi, Ryuichi Yamamoto, Katsuki Inoue, Takenori Yoshimura, Shinji Watanabe, Tomoki Toda, Kazuya Takeda, Yu Zhang, and Xu Tan. 2020. Espnet-tts: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 7654–7658. IEEE.

William Held, Ella Li, Michael Ryan, Weiyan Shi, Yanzhe Zhang, and Diyi Yang. 2024. Distilling an end-to-end voice assistant without instruction training data. *arXiv preprint arXiv:2410.02678*.

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

Wen-Chin Huang, Erica Cooper, and Tomoki Toda. 2024b. Mos-bench: Benchmarking generalization abilities of subjective speech quality assessment models. *arXiv preprint arXiv:2411.03715*.

Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, et al. 2021. On generative spoken language modeling from raw audio. *Transactions of the Association for Computational Linguistics*, 9:1336–1354.

Bo Li, Tara N. Sainath, Arun Narayanan, Joe Caroselli, Michiel Bacchiani, Ananya Misra, Izhak Shafran, Haşim Sak, Golan Pundak, Kean Chin, Khe Chai Sim, Ron J. Weiss, Kevin W. Wilson, Ehsan Variani, Chanwoo Kim, Olivier Siohan, Mitchel Weintraub, Erik McDermott, Richard Rose, and Matt Shannon. 2017. Acoustic modeling for google home. In *INTERSPEECH*, pages 399–403.

Minzhi Li, Will Held, Michael J. Ryan, Kunat Pipatanakul, Potsawee Manakul, Hao Zhu, and Diyi Yang. 2024. Talk arena: Interactive evaluation of large audio models.

Ziyang Ma, Yakun Song, Chenpeng Du, Jian Cong, Zhuo Chen, Yuping Wang, Yuxuan Wang, and Xie Chen. 2024. Language model can listen while speaking. *Preprint*, arXiv:2408.02622.

Shikib Mehri, Mihail Eric, and Dilek Hakkani-Tur. 2020. Dialoglue: A natural language understanding benchmark for task-oriented dialogue. *arXiv preprint arXiv:2009.13570*.

Tu Anh Nguyen, Eugene Kharitonov, Jade Copet, Yossi Adi, Wei-Ning Hsu, Ali Elkahky, Paden Tomasello, Robin Algayres, Benoît Sagot, Abdelrahman Mohamed, and Emmanuel Dupoux. 2023. Generative spoken dialogue language modeling. *Trans. Assoc. Comput. Linguistics*, 11:250–266.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 172–182, Online. Association for Computational Linguistics.

Yifan Peng, Yui Sudo, Muhammad Shakeel, and Shinji Watanabe. 2024a. OWSM-CTC: An open encoder-only speech foundation model for speech recognition, translation, and language identification. *arXiv preprint arXiv:2402.12654*.

Yifan Peng, Jinchuan Tian, William Chen, Siddhant Arora, Brian Yan, Yui Sudo, Muhammad Shakeel, Kwanghee Choi, Jiatong Shi, Xuankai Chang, Jee weon Jung, and Shinji Watanabe. 2024b. Owsm v3.1: Better and faster open whisper-style speech models based on e-branchformer. In *Proc. INTERSPEECH*.

Yifan Peng, Jinchuan Tian, Brian Yan, Dan Berrebbi, Xuankai Chang, Xinjian Li, Jiatong Shi, Siddhant Arora, William Chen, Roshan Sharma, Wangyou Zhang, Yui Sudo, Muhammad Shakeel, Jee weon Jung, Soumi Maiti, and Shinji Watanabe. 2023. Reproducing whisper-style training using an open-source toolkit and publicly available data. In *Proc. ASRU*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proc. ICML*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI*.

Hannah Rashkin. 2018. Towards empathetic open-domain conversation models: A new benchmark and dataset. *arXiv preprint arXiv:1811.00207*.

Antoine Raux, Dan Bohus, Brian Langner, Alan W Black, and Maxine Eskenazi. 2006. Doing research on a deployed spoken dialogue system: one year of let's go! experience. In *INTERSPEECH*.

Chandan KA Reddy, Vishak Gopal, and Ross Cutler. 2021. DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6493–6497. IEEE.

Chandan KA Reddy, Vishak Gopal, and Ross Cutler. 2022. DNSMOS P. 835: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 886–890. IEEE.

Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. UTMOS: UTokyo-SaruLab system for voiceMOS challenge 2022. In *Interspeech*, pages 4521–4525.

Jiatong Shi, Jinchuan Tian, Yihan Wu, Jee-weon Jung, Jia Qi Yip, Yoshiki Masuyama, William Chen, Yuning Wu, Yuxun Tang, Massa Baali, et al. 2024. Espnet-codec: Comprehensive training and evaluation of neural codecs for audio, music, and speech. *arXiv preprint arXiv:2409.15897*.

Shiva Sundaram and Shrikanth Narayanan. 2007. Automatic acoustic synthesis of human-like laughter. *The Journal of the Acoustical Society of America*, 121(1):527–535.

Jinhan Wang, Long Chen, Aparna Khare, Anirudh Raju, Pranav Dheram, Di He, Minhua Wu, Andreas Stolcke, and Venkatesh Ravichandran. 2024. Turn-taking and backchannel prediction with acoustic and large language model fusion. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12121–12125. IEEE.

Shinji Watanabe et al. 2018. ESPnet: End-to-end speech processing toolkit. In *Proceedings of Interspeech*, pages 2207–2211.

John Wiseman. 2024. py-webrtcvad. Accessed: 2024-12-10.

Zhifei Xie and Changqiao Wu. 2024. Mini-omni: Language models can hear, talk while thinking in streaming. *Preprint*, arXiv:2408.16725.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 1911. Dialogpt: Large-scale generative pre-training for conversational response generation. arxiv 2019. *arXiv preprint arXiv:1911.00536*.

A Example Appendix

A.1 Spoken Dialogue Systems

Spoken dialogue systems (Glass, 1999; Raux et al., 2006) have been traditionally modeled using a conventional architecture that incorporates modules for ASR, NLU, language generation, TTS, and methods for handling local discourse, in addition to a dialogue manager. Most systems even today (Face, 2024a) use a cascade of ASR, dialogue response generator, and TTS system. Recent advances have introduced audio FMs that accept speech as input and produce text-based dialogue responses. However, these systems (Held et al., 2024) still rely on an external TTS module to generate spoken outputs. Current efforts are increasingly focused on developing E2E audio FMs capable of both understanding and generating speech seamlessly. However, early efforts (Zhang et al., 2023) at building such systems could model only a single channel of speech. However, when humans interact with an audio FM, it is a two-way communication where the model listens and speaks and, more importantly, needs to

Module type	Module Name	Link to Model Checkpoint
ASR	Librispeech ASR	https://huggingface.co/espnet/simpleoier_librispeech_asr_train_asr_conformer7_wavlm_large_raw_en_bpe5000_sp
	Whisper (large)	https://huggingface.co/openai/whisper-large-v3
	OWSM (3.1)	https://huggingface.co/espnet/owsm_v3.1_ebf
	OWSM CTC (3.1)	https://huggingface.co/espnet/owsm_ctc_v3.1_1B
	OWSM CTC (3.2)	https://huggingface.co/espnet/owsm_ctc_v3.2_ft_1B
LLM	LLaMA 3.2-1B	https://huggingface.co/meta-llama/Llama-3.2-1B-Instruct
	SmolLM v2-1.7B	https://huggingface.co/HuggingFaceTB/SmolLM2-1.7B
TTS	LJSpeech VITS	https://huggingface.co/espnet/kan-bayashi_ljspeech_vits
	LibriTTS VITS	https://huggingface.co/espnet/kan-bayashi_libritts_xvector_vits
	VCTK VITS	https://huggingface.co/espnet/kan-bayashi_vctk_multi_spk_vits
	ChatTTS	https://github.com/2noise/ChatTTS
E2E Spoken Dialogue	Mini-Omni	https://github.com/gpt-omni/mini-omni/

Table 5: Links to publicly available checkpoints for various modules used in our example Gradio interface, hosted at https://huggingface.co/spaces/Siddhant/Voice_Assistant_Demo. The web interface supports 5 ASR modules, 2 LLM as text dialogue modules, and 4 TTS modules, enabling a total of 40 variations for building cascaded pipelines. Additionally, it supports interaction with the E2E spoken dialogue model Mini-Omni, offering a unified interface to chat with as many as **41** different spoken dialogue systems.

do both at the same time (Ma et al., 2024). This has led to the development of full-duplex E2E spoken dialogue systems (Défossez et al., 2024; Fang et al., 2024) that can effectively handle both channels of audio: the user's input and the system's generated output. Mini-Omni (Xie and Wu, 2024) is one such open source multimodal FM that can listen and speak while thinking.

A.2 Evaluation Setup

We use the publicly available Switchboard Eval 2000 dataset[10] (Godfrey et al., 1992) to evaluate various modules in cascaded spoken dialogue systems, as well as the E2E spoken dialogue system, using the metrics supported by our toolkit. Switchboard Eval 2000 contains 11 hours of English conversational telephone speech across 40 conversations.

For evaluating ASR modules supported by web interface (Tab.2), we follow the standard ASR evaluation approach (Watanabe et al., 2018; Radford et al., 2023) where audio recordings are passed as input to the ASR module, and the generated hypotheses are compared with ground-truth transcripts.

For evaluating TTS modules supported by web interface (Tab.4), we use the standard TTS evaluation method (Hayashi et al., 2020), where ground-truth transcripts from the spoken conversation dataset are provided as input to the TTS model. We compute TTS intelligibility and audio quality metrics as described in Sec. 5. Notably, our TTS metrics do not require reference audio. In Tab.4, we only show the intelligibility score obtained using hypothesis from 2 ASR models, namely OWSM 3.1 and Whisper for brevity.

To compute text dialogue module-specific metrics, we first sort the utterances by their start timestamps for each of the 40 conversations. We remove any utterances fully contained within another utterance. For each remaining utterance, we construct the dialogue context by concatenating the groundtruth transcripts of all prior utterances (i.e., with earlier start times) along with the current utterance. This dialogue context is passed to the LLMs ("text dialogue response generators") supported in our web interface. We then compute the text dialogue module-specific metrics for the LLM response, and the average metric values across all utterances are reported in Tab. 3. For VERT, we also report self-BLEU and auto-BLEU (Lakhotia

---

[10]https://catalog.ldc.upenn.edu/LDC2002S09

Type of Eval metric	Type of Module	Input to Model	Sample Input	Sample Reference Output	Sample Predicted Output
ASR Module Metrics	ASR (Sec. 4.1)	Segmented Audio Utterance	✗	i mean **yeah** i think you know it would be nice especially like you know when people are kids **are going to** college they should do some public service	i mean i think you know it would be nice especially like you know when people are kids **who are in** college they should do some public service
Text dialogue Module Metrics	Text dialogue response generator (Sec. 4.1)	Dialogue Context	a year a year or two in public service i do not know that tha- that seems like a lot i mean yeah i think you know it would be nice especially like you know when people are kids are going to college they should do some public service	N/A	Many people start public service early, like volunteering or internships, to gain experience and build skills.
	E2E Spoken Dialogue Model (Sec. 4.2)	Segmented Audio Utterance	✗	N/A	It's important to approach this topic with sensitivity and respect for all individuals. It's not appropriate to make generalizations about any group of people based on their location. Instead, consider focusing on the specific needs and behaviors of individuals in your community. If you have concerns about public service, it might be helpful to look into local policies and initiatives that address specific issues. This way, you can make informed decisions that benefit everyone.
TTS Module Metrics	TTS (Sec. 4.1)	Ground Truth Transcript	part of i- part of it though	NA	Synthesized audio
	E2E Spoken Dialogue Model (Sec. 4.2)	Segmented Audio Utterance	✗	NA	Synthesized audio *response*

Table 6: Explanation of the evaluation process (Sec.7) on the human-human conversation dataset Switchboard (Godfrey et al., 1992) using the metrics supported by our web interface. "N/A" is indicated under **Sample Reference Output** when the evaluation metric does not require a ground truth reference, and "✗" is displayed under **Sample Input** when the input is in audio format.

et al., 2021), which measure diversity across and within sentences, respectively.

The E2E spoken dialogue system Mini-Omni supported by our web interface does not take the prior dialogue context as input. Hence we simply provide the audio of the current utterance as input and the E2E dialogue system generates text and audio response. Text dialogue module-specific metrics are calculated for the generated text response and are reported in Tab. 3, while TTS intelligibility and audio quality metrics are computed for the corresponding audio response and are reported in Tab. 4.

Table 6 in the Appendix provides example input-output pairs used to compute these metrics for various models.

Metric	Ground Truth		ASR Transcript	
	LLaMA 3.2-1B	SmolLM v2-1.7B	LLaMA 3.2-1B	SmolLM v2-1.7B
Perplexity (↓)	48.2	113.7	47.5	285.1
Diversity				
Self BLEU-2 (↓)	75.9	77.1	76.4	76.7
Auto BLEU-2 (↓)	0.4	0.3	0.4	0.4
VERT (↓)	5.7	5.0	5.4	4.5
BERT Similarity (↑)	55.8	49.5	55.8	50.1
DialoGPT Perplexity (↓)	165.9	301.1	169.3	286.8

Table 7: Text dialogue metrics for various LMs that can be integrated into the cascaded system, when using either ground truth transcripts or ASR transcripts generated by OWSM 3.1 to construct the dialogue context.

A.3 Ablation study of using ASR transcript to generate dialog context

To compute text dialogue module-specific metrics, we experimented with using ASR transcripts generated by the OWSM 3.1 model instead of ground truth transcripts to construct the dialogue context. As shown in Tab. 7, our results indicate that LLMs are generally robust to ASR errors, with both LLMs performing similarly, although SmolLM shows worse perplexity score.

	Intelligibility				Speech Quality				
	OWSM 3.1		Whisper		UTMOS (↑)	DNS (↑)		PLCMOS (↑)	SSQA (↑)
	WER (↓)	CER (↓)	WER (↓)	CER (↓)		Overall	P808		
LJSpeech VITS									
GT transcript	16.1	9.8	15.1	10.1	4.06	3.05	3.79	4.50	4.01
LLM response	6.0	2.4	2.9	1.0	4.33	3.22	4.05	4.62	4.43
LibriTTS VITS									
GT transcript	19.3	11.5	16.5	10.4	3.97	3.10	3.49	4.34	4.19
LLM response	5.0	2.1	2.2	1.0	4.38	3.26	3.87	4.66	4.46
VCTK VITS									
GT transcript	27.0	16.8	22.1	14.7	4.07	3.08	3.63	4.62	4.12
LLM response	9.0	4.2	4.5	2.1	3.94	3.24	3.75	4.16	4.28

Table 8: TTS module metrics for a subset of TTS models that can be used in the cascaded system when using ground truth transcripts or response generated by LLaMA 3.2-1B as input.

Turn taking event	Number of events / minute		% Cummulated duration	
	Cascaded	SWBD	Cascaded	SWBD
IPU	7.8	15.7	58.9	97.3
Pause	3.6	3.8	2.5	5.7
Gap	3.9	5.5	39.2	3.7
Overlap	0.3	6.6	0.6	6.7

Table 9: Statistics of turn-taking events in our pilot study's cascaded system compared to Switchboard (SWBD) (Godfrey et al., 1992).

A.4 Ablation study of using LLM response for TTS module metrics

To compute TTS module-specific metrics, we experimented with using responses generated by the LLaMA 3.2-1B model instead of ground truth transcripts as input (e.g., "Many people start public service early, like volunteering or internships, to gain experience and build skills." instead of "part of i- part of it though" in Tab. 6).

As shown in Tab. 8, ① intelligibility improved across all TTS models, likely because the LLM-generated responses were more grammatically coherent compared to the disfluent spoken text typically seen in spoken conversations, making them easier for TTS systems to synthesize. Additionally, ② the LJSpeech and LibriTTS VITS models demonstrated similar performance in terms of both intelligibility and audio quality when using LLM responses as input. Finally, when compared to Mini-Omni's TTS module metrics (Tab. 4), ③ Mini-Omni performed worse than the TTS models on *both intelligibility and audio quality* when using LLM responses as input.

A.5 A Pilot Analysis of User interacting with dialogue systems using our interface

Finally, we conducted a small-scale human evaluation with the authors and research colleagues to demonstrate the utility of our web interface for recording human judgments and human-AI interaction data. In this study, four participants, consisting of research authors and colleagues, interacted with a cascaded spoken dialogue system constructed using WebRTC VAD, OWSM CTC 3.1, LLaMA-3.2-1B, and the LJSpeech VITS TTS model. Nearly 2 hours of human-AI conversation data were collected and used to compute evaluation metrics (Sec. 5).

Table 9 shows that ① the cascaded system exhibited high latency, leading to larger *gaps*, and ② less overlapping speech compared to natural human-human conversations. The system's speaking rate (138.7 words/min) was also slower than humans (204.5 words/min), and it never backchanneled.

Additionally, we collected 501 human judgments on the naturalness and 260 judgments on the relevance of the system's outputs. Approximately 75.8% of outputs were rated as very natural, 17.6% as somewhat awkward, 1.6% as unnatural, and 5% as very awkward. Regarding relevance, 71.9% of outputs were judged highly relevant to the dialogue context, 14.6% partially relevant, 6.5% slightly irrelevant, and 5% very awkward. This pilot study provides preliminary evidence of the efficacy of our web interface in facilitating robust human-in-the-loop evaluations of dialogue systems.