GAME-TIME: EVALUATING TEMPORAL DYNAMICS IN SPOKEN LANGUAGE MODELS

Kai-Wei Chang[*1], *En-Pei Hu*[*2], *Chun-Yi Kuan*[2], *Wenze Ren*[2], *Wei-Chih Chen*[2],
Guan-Ting Lin[2], *Yu Tsao*[3], *Shao-Hua Sun*[2], *Hung-yi Lee*[2], *James Glass*[1]

[1] Massachusetts Institute of Technology, USA
[2] National Taiwan University, Taiwan
[3] Academia Sinica, Taiwan

ABSTRACT

Conversational Spoken Language Models (SLMs) are emerging as a promising paradigm for real-time speech interaction. However, their capacity of temporal dynamics, including the ability to manage timing, tempo and simultaneous speaking, remains a critical and unevaluated challenge for conversational fluency. To address this gap, we introduce the **Game-Time Benchmark**, a framework to systematically assess these temporal capabilities. Inspired by how humans learn a language through language activities, Game-Time consists of basic instruction-following tasks and advanced tasks with temporal constraints, such as tempo adherence and synchronized responses. Our evaluation of diverse SLM architectures reveals a clear performance disparity: while state-of-the-art models handle basic tasks well, many contemporary systems still struggle with fundamental instruction-following. More critically, nearly all models degrade substantially under temporal constraints, exposing persistent weaknesses in time awareness and full-duplex interaction. The Game-Time Benchmark provides a foundation for guiding future research toward more temporally-aware conversational AI. Demos and datasets are available on our project website[1].

***Index Terms*—** Spoken Language Models, Temporal Dynamics, Full-Duplex Speech, Conversational AI, Benchmark

1. INTRODUCTION

In the pursuit of human-like conversation with machines, the research frontier is moving beyond text-based Large Language Models (LLMs). The next challenge lies in mastering conversational dynamics in real-time speech, which has given rise to the field of conversational Spoken Language Models (SLMs) [1, 2, 3, 4, 5, 6, 7]. This marks a critical shift from rigid turn-by-turn dialogues to fluid spoken interactions. Achieving such dynamics requires SLMs to operate in a *real-time full-duplex* manner [8, 9], where models must listen and speak simultaneously while producing seamless responses. This is inherently difficult, demanding synchronous speech generation, continual intent recognition, and precise control over both what to respond and when to respond. A further challenge lies in modeling the temporal dynamics of spoken interaction, where contemporary systems often fail to capture the fine-grained timing that is essential for advanced conversational fluency. For instance, they struggle to process user speech while planning a coherent reply that aligns with user-specified timing and tempo. This limitation underscores a fundamental deficiency: *the lack of time-awareness*.

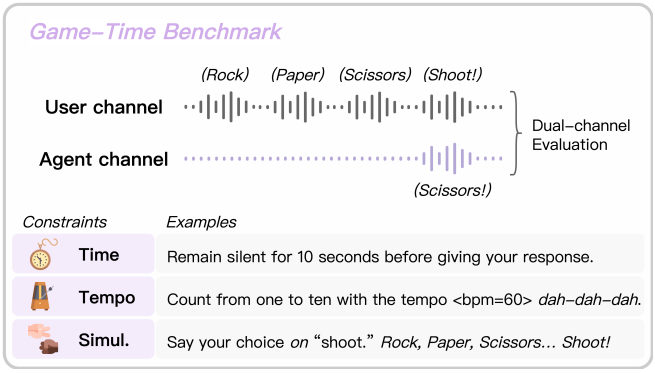

Fig. 1. Overview of the *Game-Time Benchmark*, evaluating temporal dynamics in conversational Spoken Language Models (SLMs).

Existing benchmarks for SLMs focus on content and style generation quality [10, 11], mimicking human dialogue behaviors (e.g. back-channeling) [12], and turn-taking [13]. However, they lack the focus on temporal dynamics in the conversation. To address this critical gap, we introduce the ***Game-Time Benchmark***, which is designed to assess the temporal capabilities of SLMs, focusing on their ability to perceive, predict, and produce speech in sync with the user.

This work is inspired by childhood language acquisition [14]. Children learn to talk through language activities that require not only understanding the meaning of vocabulary, such as counting numbers and naming objects, but also acquiring a sense of timing and tempo in coordination games [15, 16]. For example, playing "rock-paper-scissors" relies on sharing a common tempo and acting precisely on a specific cue (e.g., "shoot"). This inspiration led us to design the Game-Time Benchmark, which contains two categories of tasks. The **Basic Tasks** evaluate an SLM's foundational ability to follow simple instructions, a challenge that can still prove difficult for modern SLMs. The **Advanced Tasks** build upon this foundation by augmenting the core instructions with temporal constraints. Here, the SLM must perform the basic tasks while fulfilling requirements for timing and synchronicity. The Game-Time Benchmark provides a framework for evaluating whether a model can move beyond mere content generation to acquire the temporal dynamics of conversational fluency. It proposes a novel perspective on evaluation, focusing not just on *what to say*, but critically, on *when to say*.

In this paper, we evaluate various SLMs with different design philosophies, including Moshi [17], Unmute [18], Freeze-Omni [19], Gemini-Live [20], and GPT-realtime [21]. Our results

*Co-first authors
[1] https://ga642381.github.io/Game-Time

Table 1. Game-Time Task Families. Basic Task contains fundamental tasks. Advanced Task contains temporal dynamics paired with Basic Task. $\mathcal{N}$: Number of subtasks. *The game of Rock paper scissors is itself an advanced task.

Category	Task Family	$\mathcal{N}$	Subtask / Paired Basic Tasks	Description
Basic	1 - Sequence	3	Number; Alphabet; Spell	Generate sequential items in order from n_{start} to n_{end}.
	2 - Repeat	2	Word; Sentence	Repeat user-provided content $\mathcal{C}$.
	3 - Compose	2	Word; Scenario	Compose response including a target word w or fitting a scenario $\mathcal{S}$.
	4 - Recall	3	Vocabulary; Letter; Rhyme	Name N items that satisfy property ϕ.
	5 - Open-Ended	2	Empathy; QA	Provide helpful and contextually appropriate content.
	6 - Role-Play	2	Scenario; Persona	Act within an imagined scenario $\mathcal{S}$ or play a given persona $\mathcal{P}$.
Adv.	A - Time-Fast	10	[Multiple Basic Tasks]	Complete the task quickly, within a specified duration τ_{fast}.
	B - Time-Slow	7	[Multiple Basic Tasks]	Perform the task slowly, taking at least the specified duration τ_{slow}.
	C - Time-Silence	4	Repeat; Recall; Open-Ended	Insert a silent interval of s seconds before the response.
	D - Tempo-Interval	4	Sequence; Recall	Follow a specified tempo with δ-second space between each word.
	E - Tempo-Adhere	4	Sequence; Recall	Adhere to the tempo specified by the user's spoken example $\mathcal{C}_{\text{tempo}}$.
	F - Simul.-Shadow	1	Repeat	Repeat each word with immediate, word-by-word overlap.
	G - Simul.-Cue	1	Rock paper scissors*	Overlap with the user by speaking at a designated timing or cue.

show a performance disparity even on Basic Tasks: while state-of-the-art models generally excel, many contemporary SLMs still struggle with fundamental instruction-following. Furthermore, the performance of nearly all models degrades significantly when temporal constraints are introduced. Our findings indicate that models especially struggle with tasks requiring time awareness and real-time full-duplex capability, revealing a critical gap in the capabilities of current systems. For reproducibility, datasets and results are available on our website.

2. RELATED WORKS

2.1. Full-duplex Spoken Language Models

Recent work has explored how SLMs can move beyond turn-based interaction toward full-duplex conversation [12, 22, 23, 24, 25, 26], where listening and speaking occur simultaneously. Two main modeling strategies have emerged to achieve full-duplex capability [3]:

(1) Dual-channel SLMs [22, 17, 27, 28] use two input channels: a listening channel for user speech and a speaking channel for the model's own output. At each time step, the model simultaneously processes both inputs and generates output speech, which may be either voiced or explicitly silent. Although this architecture significantly increases modeling complexity, it naturally supports real-time listening and speaking concurrently.

(2) Time-multiplexing SLMs [29, 19, 18] include a state prediction mechanism [30] that decides whether to speak or remain silent. During user speech, the model withholds output and monitors for an appropriate moment to take the turn and respond. During its own speech, the model generates in an autoregressive manner, and this generation is often interrupted by incoming user speech.

In the Game-Time benchmark, we evaluate both model designs and commercial voice agent APIs, comparing their ability to manage timing and overlap, and discussing the trade-offs of each design.

2.2. Benchmarks for Conversational Spoken Language Models

A number of benchmarks have been proposed to evaluate SLMs. Some focus on the spoken language understanding and paralinguistics generation [10, 31, 32, 33, 34, 35]. Others, including *Full-Duplex-Bench* [12], *Talking Turns* [13], and *FD-Bench* [36] primarily assess dialogue behaviors including latency, back-channeling, and turn-taking, with an emphasis on the conversational naturalness. However, these benchmarks do not directly test a model's ability to comprehend and act upon explicit temporal requirements. Game-Time complements these existing works by shifting the focus from *what to say* to *when to say*, introducing a framework for evaluating timing, tempo, and simultaneous speaking within a conversation.

3. GAME-TIME BENCHMARK

We introduce the *Game-Time Benchmark* to evaluate SLMs on their understanding of *time*, *tempo*, and timely *simultaneously speaking*. In this section, we define the task families, describe how the benchmark is constructed, and outline the evaluation protocol.

3.1. Task Families

Inspired by how humans learn a language with language activities and games, the Game-Time benchmark comprises two categories: Basic Tasks, testing fundamental speech capabilities, and Advanced Tasks, which paired temporal constraints with suitable Basic Tasks to assess the model's time-awareness. The full taxonomy of families, subtasks, and temporal requirements is summarized in Table 1.

Game-Time Basic Tasks: The 6 Basic Task families reflect fundamental capabilities of spoken interaction. They are inspired by the kinds of activities through which humans first practice language: reciting ordered sequences (*Sequence*), repeating spoken content (*Repeat*), composing sentences that meet specific criteria (*Compose*), recalling items from memory (*Recall*), responding helpfully and openly in conversation (*Open-ended*), and navigating hypothetical scenarios or adopting specific personas (*Role-play*). Together, they capture a spectrum from structured behaviors (e.g., counting) to open-ended and social behaviors (e.g., *Empathy* and *Role-Play*).

Game-Time Advanced Tasks: Advanced tasks introduce constraints that move beyond *what* to say and focus on *when* to say it. Here, the philosophy is to test temporal and interactive fluency — skills that come naturally to humans but remain underexplored in SLMs. *Time* tasks examine whether models can modulate overall duration of speaking time, which require coordinating not only the speaking rate but also the content; *Tempo* tasks probe their ability to sustain rhythmic consistency or synchronize with an external beat; *SimulSpeak* tasks challenge them to overlap with the user's speech, listening and synchronizing with the user in real time. These constraints are abstractions of conversational dynamics such as timing, tempo and coordination, which are central to human conversation.

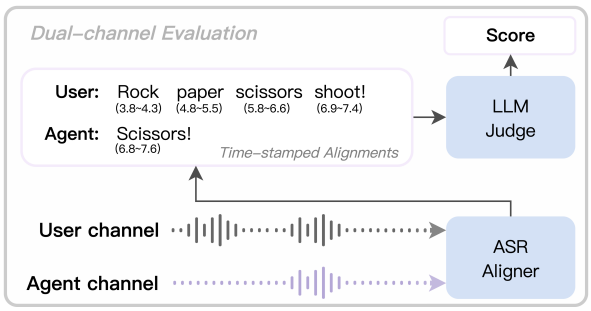

Fig. 2. Dual-channel Evaluation with LLM-as-a-judge.

3.2. Task Formalization

We formalize our tasks within an Instruction-Following (IF) framework [37, 38]. Each IF instance is specified by a base task t and a set of constraints $\mathcal{C}$. Performing an IF task requires the model to *perform a base task t while satisfying all constraints in $\mathcal{C}$*. Each constraint is a predicate over variables that are typically numeric or symbolic.

Example: Consider the user instruction, "Please count from one to ten in 10 seconds." Here, the base task is *sequential generation* t_{seq}. Also, this instruction implies two constraints: (i) a range constraint $c_{\text{range}}$with variables $(n_{\text{start}}, n_{\text{end}}) = (1, 10)$ requiring the spoken sequence to be $1, 2, \ldots, 10$, and (ii) a duration constraint $c_{\text{dur}}$with variable $\tau_{\text{fast}} = 10$s requiring the task performing time to be less than 10 seconds. For this instance, the model should perform the base task t_{seq} while satisfying $\mathcal{C} = \{c_{\text{range}}, c_{\text{dur}}\}$. Building upon this formalization, we can systematically generate the dataset by creating natural language templates, instantiating them with diverse variables, and synthesizing the resulting instructions into speech.

3.3. Dataset Construction Pipeline

The Game-Time benchmark dataset is constructed through a four-stage pipeline designed to generate a diverse and high-quality set of spoken instructions. **(i) Seed Instruction Creation**: We begin by manually writing a set of seed instructions for Basic Tasks, defining the base task and corresponding variables. **(ii) Linguistic Diversification**: An LLM paraphrases seed instructions to create a variety of linguistic templates. We then populate these templates by varying the defined variables, resulting in a large and diverse set of instruction texts for the Basic Tasks. The Advanced Tasks are derived by augmenting a subset of Basic Tasks with temporal constraints C. This controlled approach ensures that corresponding basic and Advanced Tasks share the same underlying base task t, allowing us to focus on the performance variation when imposing constraints. **(iii) Speech Synthesis**: These text-based instructions are synthesized into audio using a TTS system with multiple voices to ensure vocal diversity [2]. **(iv) Quality Control**: Finally, we use an ASR model to transcribe the synthesized audio. Instructions whose transcriptions do not closely match the original text are filtered out. This automated check is supplemented by manual listening verification on a majority of the samples to ensure high perceptual quality.

In the end we have a total of 1,475 test instances: 700 samples for the Basic tasks (14 subtasks, 50 samples each) and 775 samples for the Advanced tasks (31 subtasks, 25 samples each).

[2]We primarily use CosyVoice [39] for speech synthesis. For tasks requiring precise tempo control, the audio was edited manually. Google TTS was used for the "Rock, Paper, Scissors" task, as we found it produced higher-quality output in this case.

Table 2. Comparison of SLMs in Game-Time Benchmark.

Model	Full-Duplex Method	Open	Frozen LLM
Freeze-Omni	Time-Multiplexing	✓	✓
Unmute	Time-Multiplexing	✓	✓
Moshi	Dual Channel	✓	✗
Gemini-Live	–	✗	-
GPT-realtime	–	✗	-
SSML-LLM	Non-causal Completion	–	✓

3.4. Dual-channel Evaluation

Our evaluation protocol, illustrated in Figure 2, leverages an LLM to score model performance. For each dialogue, we first transcribe the dual-channel audio (user and model) to obtain time-aligned text. This transcription is then provided to an LLM judge, which assesses the model's performance on the criteria of instruction following [3].

We also explored alternative methods: using an *audio-LLM-as-a-judge* [40] and employing rule-based automatic metrics. We found the audio-LLM approach is also effective but more costly and less aligned with human's evaluation (discussed in Sec.5.2). Meanwhile, rule-based metrics are often too rigid for the interpretive nature of spoken dialogue. For instance, a rigid script would penalize a model for including a natural conversational preamble (e.g., "Okay, I'll start now...") in a time-constrained task. In contrast, an LLM can perform reasoning to recognize and evaluate the core speech embedded in the whole dialogue and give a reasonable evaluation[4].

Overall, we find this text-based LLM judge to be a unified, simple, yet effective method. To validate this approach, we conduct subjective human evaluations and confirm that its assessments align with human preferences. Furthermore, this method can effectively evaluate other behaviors such as turn-taking. Due to space limitations, these additional results are available on our project website.

4. EXPERIMENTAL SETUP

We evaluate various SLMs on the Game-Time Benchmark with different full-duplex strategies (see Table 2). This includes **Time-Multiplexing** models (Freeze-Omni [19], Unmute [18]) which use a modular pipeline of a streaming encoder, a frozen LLM, and a streaming decoder; and a **Dual-channel** model (Moshi [17]) where a fine-tuned LLM directly processes and generates speech. We also include **commercial voice agents**: Gemini-Live[5] and GPT-realtime[6]

Oracle System: We introduce *SSML-LLM* as our benchmark's oracle topline. This is a non-streaming and non-causal system that operates with future knowledge. It receives the full word-level alignments of the user's utterance as input, and an LLM then generates a dialogue counterpart with timing that is precisely controlled and synchronized with the user's speech via Speech Synthesis Markup Language (SSML). This SSML is then synthesized into

[3]For "Open-Ended" Basic Tasks, which lack an explicit instruction, the LLM judge evaluates "response appropriateness" instead.

[4]While we could have designed instructions to be unequivocally precise, doing so would result in unnatural conversations. Our approach prioritizes evaluating a model's ability to interpret natural instructions and time-related cues, rather than its ability to follow overly constrained commands.

[5]Gemini-Live [20]. API model name: *gemini-live-2.5-flash-preview*

[6]GPT-realtime [21]. API model name: *gpt-realtime*

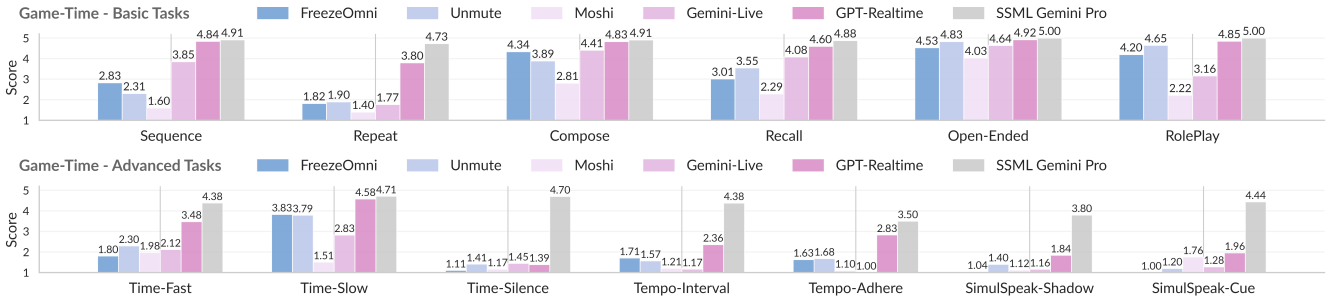

Fig. 3. Game-Time benchmark scores evaluated with LLM-as-a-judge. **Top**: results on Basic Tasks. **Bottom**: results on Advanced Tasks.

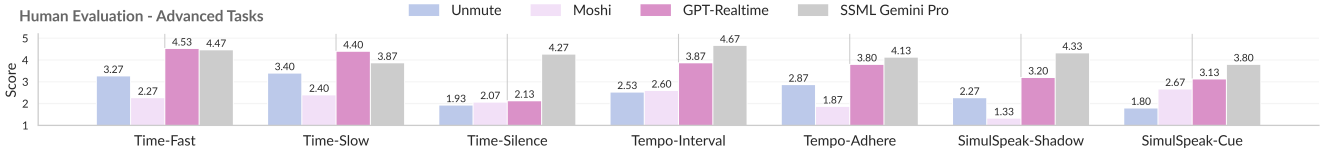

Fig. 4. Human evaluation on Game-Time Advanced Tasks.

audio by a TTS system. For example, after processing the user's speech "Rock paper scissors shoot!", it generates `<ssml><break time="6.9s"/>scissors!</ssml>` to make the response "Scissors!" overlap with user's "Shoot!". Although not feasible in real-time, SSML-LLM helps calibrate our LLM judge and human evaluations by providing a theoretical performance ceiling. We use Gemini 2.5 Pro [41] with reasoning and Google TTS.

Dual-channel Evaluation: Our LLM-as-a-judge framework requires time-stamped data for its analysis. To obtain word-level alignments, we use the Whisper-medium model. Based on a preliminary study, we selected Gemini 2.5 Pro as the LLM judge, which we found superior to other open-source and commercial models at processing these time-stamped transcripts, leveraging its strong reasoning capabilities for evaluating complex temporal behaviors.

5. RESULTS

5.1. Main Results

Basic Tasks: As shown in Fig. 3 (Top), the oracle topline consistently achieves the best performance across all tasks. GPT-realtime shows strong performance on most Basic Tasks, and it is worth noting that in *Repeat*, it is the only model that delivers reasonable performance. On the other hand, we observe that time-multiplexing models (Freeze-Omni and Unmute), which rely on a frozen LLM, generally outperform the dual-channel model (Moshi). This suggests that fine-tuning a text LLM to model speech signals remains challenging in spoken conversation scenarios. Overall, although Basic Tasks can be handled by the most advanced model (GPT-realtime), there is still room for improvement in modern academic models.

Advanced Tasks: As shown in Fig. 3 (Bottom), introducing temporal constraints results in a substantial drop in performance. Among the Advanced Tasks, models perform comparatively better on *Time-Fast* and *Time-Slow*, but fail on *Time-Silence*, suggesting that they can adjust their speaking rate in response to user instructions but still fail to grasp precise temporal requirements. Similarly, adhering to tempos (*Tempo*) and synchronizing speech with users (*SimulSpeak*) remains difficult for modern SLMs, even for SOTA commercial voice agents such as GPT-realtime. This performance disparity suggests that current SLMs do not possess time-awareness, highlighting the need to focus on this capability in future research.

Table 3. Correlation between human judge with LLM and ALLM judge, both using Gemini 2.5 Pro.

	Spearman's ρ	Pearson's r
Human - LLM	0.677	0.675
Human - ALLM	0.643	0.625

5.2. Human Evaluation

Fig. 4 shows the result of the human evaluation. For each task, there are 20 samples, with each evaluated by three human judges via Prolific. We observe a similar trend in the performance of SLMs as in LLM dual-channel evaluation. Table 3 presents the correlation between LLM-as-a-judge scores and human evaluations for Advanced Tasks. We also list the Audio LLM judge score for reference. Across 4 models $\times$ 35 data scores, we observe a reasonably high correlation (Spearman's $\rho = 0.677$, Pearson's $r = 0.675$) for our dual-channel evaluation method. These results suggest that the LLM-as-a-judge is reliable and well aligned with human evaluation. We also find that for tasks requiring precise measurements like maintaining a ten-second silence, the LLM may be more objective than humans, as it can leverage time-stamped alignment data for evaluation.

6. CONCLUSION

This paper introduced the Game-Time Benchmark to address a critical gap in the evaluation of the temporal dynamics of conversational Spoken Language Models (SLMs). We evaluated various SLMs with a series of tasks testing temporal capabilities of timing, tempo, and simultaneous speaking. Our results reveal a clear gap, with some models able to handle basic instructions, but nearly all failing once temporal constraints are introduced. This widespread inability to manage precise timing or synchronize with a user reveals a persistent lack of time-awareness in current SLMs, even in the most advanced systems. Our dual-channel evaluation utilizing an LLM for reasoning was shown to be a reliable method, offering a unified and scalable way to measure these complex behaviors. We hope the Game-Time Benchmark will motivate the community to build the next generation of diverse and time-aware SLMs.

7. ACKNOWLEDGMENT

We are grateful to Yi-Cheng Lin and Cheng-Han Chiang for their valuable discussions on evaluation methods, and to Shih-Yun Shan Kuan for assistance with commercial API usage.

8. REFERENCES

[1] Shengpeng Ji et al., "Wavchat: A survey of spoken dialogue models," *arXiv preprint arXiv:2411.13577*, 2024.

[2] Wenqian Cui et al., "Recent advances in speech language models: A survey," in *ACL (1)*. 2025, pp. 13943–13970, Association for Computational Linguistics.

[3] Siddhant Arora et al., "On the landscape of spoken language models: A comprehensive survey," *arXiv preprint arXiv:2504.08528*, 2025.

[4] Haibin Wu et al., "Towards audio language modeling–an overview," *arXiv preprint arXiv:2402.13236*, 2024.

[5] Siddique Latif et al., "Sparks of large audio models: A survey and outlook," *arXiv preprint arXiv:2308.12792*, 2023.

[6] Ke Hu et al., "Efficient and Direct Duplex Modeling for Speech-to-Speech Language Model," in *Interspeech 2025*, 2025, pp. 2715–2719.

[7] Chaoyou Fu et al., "Vita-1.5: Towards gpt-4o level real-time vision and speech interaction," *arXiv preprint arXiv:2501.01957*, 2025.

[8] Ziyang Ma et al., "Language model can listen while speaking," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025, vol. 39, pp. 24831–24839.

[9] Qian Chen, Yafeng Chen, Yanni Chen, et al., "MinMo: A multimodal large language model for seamless voice interaction," *arXiv preprint arXiv:2501.06282*, 2025.

[10] Ruiqi Yan et al., "URO-Bench: A comprehensive benchmark for end-to-end spoken dialogue models," *arXiv preprint arXiv:2502.17810*, 2025.

[11] Chih-Kai Yang et al., "Towards holistic evaluation of large audio-language models: A comprehensive survey," *arXiv preprint arXiv:2505.15957*, 2025.

[12] Guan-Ting Lin et al., "Full-duplex-bench: A benchmark to evaluate full-duplex spoken dialogue models on turn-taking capabilities," *arXiv preprint arXiv:2503.04721*, 2025.

[13] Siddhant Arora et al., "Talking turns: Benchmarking audio foundation models on turn-taking dynamics," in *The Thirteenth International Conference on Learning Representations*, 2025.

[14] Jerome Bruner, "Child's talk: Learning to use language," *Child Language Teaching and Therapy*, vol. 1, pp. 111–114, 1985.

[15] Nancy Ratner and Jerome Bruner, "Games, social exchange and the acquisition of language," *Journal of child language*, vol. 5, no. 3, pp. 391–401, 1978.

[16] David Whitebread et al., *The role of play in children's development: A review of the evidence*, LEGO Fonden Billund, Denmark, 2017.

[17] Défossez et al., "Moshi: a speech-text foundation model for real-time dialogue," *arXiv preprint arXiv:2410.00037*, 2024.

[18] Neil Zeghidour et al., "Streaming sequence-to-sequence learning with delayed streams modeling," *arXiv preprint arXiv:2509.08753*, 2025.

[19] Xiong Wang et al., "Freeze-omni: A smart and low latency speech-to-speech dialogue model with frozen llm," in *Forty-second International Conference on Machine Learning*, 2025.

[20] Google, "Gemini live: A more helpful, natural and visual assistant," Aug. 2025.

[21] OpenAI, "Introducing gpt-realtime and realtime api updates for production voice agents," Aug. 2025.

[22] Tu Anh Nguyen et al., "Generative spoken dialogue language modeling," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 250–266, 2023.

[23] Chen Chen, Ke Hu, Chao-Han Huck Yang, Ankita Pasad, et al., "Reinforcement learning enhanced full-duplex spoken dialogue language models for conversational interactions," in *Second Conference on Language Modeling*, 2025.

[24] Wenyi Yu, Siyin Wang, et al., "Salmonn-omni: A standalone speech llm without codec injection for full-duplex conversation," *arXiv preprint arXiv:2505.17060*, 2025.

[25] Jin Xu et al., "Qwen2. 5-omni technical report," *arXiv preprint arXiv:2503.20215*, 2025.

[26] Qinglin Zhang et al., "OmniFlatten: An end-to-end GPT model for seamless voice conversation," in *Proc. ACL*, 2025.

[27] Qichao Wang et al., "NTPP: Generative speech language modeling for dual-channel spoken dialogue via next-token-pair prediction," in *Forty-second ICML*, 2025.

[28] Anne Wu et al., "Aligning spoken dialogue models from user interactions," in *Forty-second ICML*, 2025.

[29] Xinrong Zhang et al., "Beyond the turn-based game: Enabling real-time conversations with duplex models," *arXiv preprint arXiv:2406.15718*, 2024.

[30] Peng Wang et al., "A full-duplex speech dialogue scheme based on large language model," *Advances in Neural Information Processing Systems*, vol. 37, pp. 13372–13403, 2024.

[31] Junyi Ao, Yuancheng Wang, et al., "SD-Eval: A benchmark dataset for spoken dialogue understanding beyond words," *Advances in Neural Information Processing Systems*, 2024.

[32] Yiming Chen et al., "Voicebench: Benchmarking llm-based voice assistants," *arXiv preprint arXiv:2410.17196*, 2024.

[33] Kuofeng Gao et al., "Benchmarking open-ended audio dialogue understanding for large audio-language models," in *ACL (1)*. 2025, Association for Computational Linguistics.

[34] Heyang Liu, Yuhao Wang, et al., "Vocalbench: Benchmarking the vocal conversational abilities for speech interaction models," *arXiv preprint arXiv:2505.15727*, 2025.

[35] Jian Zhang et al., "Wildspeech-bench: Benchmarking audio llms in natural speech conversation," *arXiv preprint arXiv:2506.21875*, 2025.

[36] Yizhou Peng et al., "FD-Bench: A Full-Duplex Benchmarking Pipeline Designed for Full Duplex Spoken Dialogue Systems," in *Interspeech 2025*, 2025, pp. 176–180.

[37] Jeffrey Zhou et al., "Instruction-following evaluation for large language models," *arXiv preprint arXiv:2311.07911*, 2023.

[38] Valentina Pyatkin et al., "Generalizing verifiable instruction following," *arXiv preprint arXiv:2507.02833*, 2025.

[39] Zhihao Du et al., "Cosyvoice 2: Scalable streaming speech synthesis with large language models," *arXiv preprint arXiv:2412.10117*, 2024.

[40] Cheng-Han Chiang et al., "Audio-aware large language models as judges for speaking styles," *arXiv preprint arXiv:2506.05984*, 2025.

[41] Gheorghe Comanici et al., "Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities," *arXiv preprint arXiv:2507.06261*, 2025.