



# Towards a Japanese Full-duplex Spoken Dialogue System

Atsumoto Ohashi, Shinya Iizuka, Jingjing Jiang, Ryuichiro Higashinaka

Graduate School of Informatics, Nagoya University, Japan

ohashi.atsumoto.c0@s.mail.nagoya-u.ac.jp

## Abstract

Full-duplex spoken dialogue systems, which can model simultaneous bidirectional features of human conversations such as speech overlaps and backchannels, have attracted significant attention recently. However, the study of full-duplex spoken dialogue systems for the Japanese language has been limited, and the research on their development in Japanese remains scarce. In this paper, we present the first publicly available full-duplex spoken dialogue model in Japanese, which is built upon Moshi, a full-duplex dialogue model in English. Our model is trained through a two-stage process: pre-training on a large-scale spoken dialogue data in Japanese, followed by fine-tuning on high-quality stereo spoken dialogue data. We further enhance the model’s performance by incorporating synthetic dialogue data generated by a multi-stream text-to-speech system. Evaluation experiments demonstrate that the trained model outperforms Japanese baseline models in both naturalness and meaningfulness.<sup>1</sup>

**Index Terms:** spoken dialogue system, full-duplex

## 1. Introduction

Full-duplex spoken dialogue systems are attracting attention as a way to achieve natural voice interaction [1, 2, 3]. In dialogue, *full-duplex* refers to having simultaneous bidirectional features such as speech overlaps and backchannels. Full-duplex spoken dialogue systems need to be researched for addressing the limitations of conventional *half-duplex* dialogue systems [4, 5] that wait for the other speaker to finish speaking before responding.

Moshi [6] is a prominent full-duplex spoken dialogue model that achieves full-duplex dialogue by modeling both its own and the user’s speech streams in parallel. Additionally, research on full-duplex spoken dialogue systems is increasing, particularly in English [7, 8, 9]. However, there is no full-duplex spoken dialogue system for other languages including Japanese, and insights into full-duplex spoken dialogue systems are lacking compared to in English.

The purpose of this study is to provide insights and baseline performance through the development of the first Japanese full-duplex spoken dialogue model. In this study, we adapt Moshi to Japanese through pre-training and fine-tuning using Japanese spoken dialogue data. In the pre-training stage, we use the J-CHAT corpus [10], which contains approximately 69,000 hours of monophonic spoken dialogue, to acquire basic capabilities of spoken dialogues in Japanese. Then, in the fine-tuning stage, we model full-duplex dialogue in Japanese using 344 hours of high-quality stereo spoken dialogue data where two speakers’

<sup>1</sup>Our training codebase, fine-tuned models, and speech samples are available at <https://nu-dialogue.github.io/j-moshi>

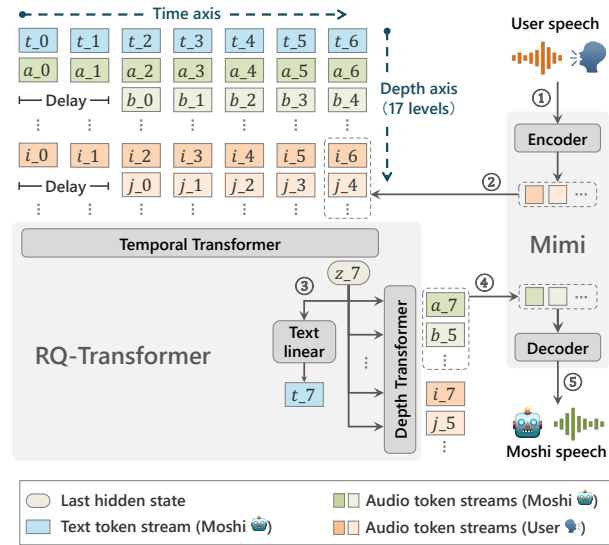


Figure 1: *Moshi’s model architecture.* It consists of the neural audio codec *Mimi*, which encodes speech waveforms into discrete audio tokens, and *RQ-Transformer*, which autoregressively models sequences of text tokens and audio tokens.

voices are recorded on separate channels. Furthermore, we aim to improve dialogue capabilities through data augmentation using 602 hours of synthetic spoken dialogue generated by multi-stream text-to-speech (TTS). Through evaluation experiments, we demonstrate that the adapted model can acquire a certain level of Japanese speech capabilities. Moreover, through comparative analysis between English and Japanese, we show that the model can acquire Japanese-specific characteristics, such as having more speech overlaps [11, 12].

## 2. Moshi

This section explains the model architecture of Moshi [6] (Figure 1), which consists of the neural speech codec *Mimi* and the large-scale speech language model *RQ-Transformer*.

### 2.1. Mimi

*Mimi* is a neural speech codec [13] consisting of a SEANet [14] autoencoder and a residual vector quantizer [15]. The encoder discretizes 24,000Hz speech waveform data into audio tokens at a frame rate of 12.5Hz. To each frame, eight layers of tokens are assigned, where layer 1 is trained to retain semantic information of speech, while layers 2-8 are trained to retain acoustic

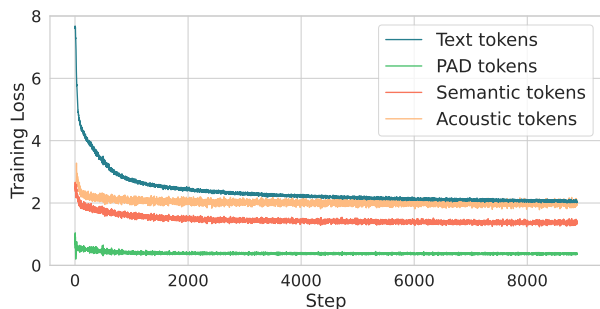


Figure 2: Moshi’s Loss curve during pre-training on J-CHAT

information. A token of layer 1 is called the semantic token, and those of layers 2-8 are called acoustic tokens.

As shown in Figure 1, at each timestep (i.e., one frame), Mimi encodes the user’s input speech to output eight user audio tokens and feeds them to RQ-Transformer. Simultaneously, it decodes eight Moshi audio tokens generated by RQ-Transformer to generate Moshi’s output speech.

## 2.2. RQ-Transformer

RQ-Transformer consists of a Temporal Transformer based on a 7B-parameter large language model (LLM), which has been pre-trained with a large amount of web text, and a smaller Depth Transformer. This architecture allows Moshi to generate natural spoken dialogue by leveraging the high level of linguistic capabilities of the LLM.

The Temporal Transformer models token sequences along the time dimension at a rate of 12.5Hz. The token sequences include 17 layers: Moshi’s text token sequence (1 layer), Moshi’s audio token sequence (8 layers), and the user’s audio token sequence (8 layers). Here, text tokens represent the textual content to be spoken as inner monologue. At each timestep  $s$ , it outputs a hidden vector  $z_s$  from  $(s - 1) \times 17$  tokens up to the previous step  $s - 1$ . Then, the Text Linear layer samples the text token at  $s$  from  $z_s$ . The Depth Transformer models audio tokens along the depth dimension at  $s$ . Taking  $z_s$  as input, it autoregressively samples eight Moshi audio tokens and eight user audio tokens. Note that a delay of one timestep is applied to acoustic tokens to stabilize the quality of generated speech.

To align the timing of text tokens with audio tokens, token-level transcriptions created by Whisper [16] (i.e., timestamp information of which timestep each text token corresponds to) are utilized, and PAD tokens are inserted into timesteps where no text tokens are assigned.

The training procedure of Moshi consists of pre-training with 7 million hours of monophonic audio data, fine-tuning with 2,000 hours of stereo spoken dialogue, and instruction tuning with 20,000 hours of stereo spoken dialogue synthesized by the multi-stream TTS system. For more detailed information about Moshi, refer to Moshi’s technical paper [6].

## 3. Training in Japanese

In this study, we adapt Moshi to Japanese spoken dialogue. This section explains the text vocabulary adaptation and two-stage training steps, consisting of pre-training and fine-tuning, that were implemented to adapt original Moshi<sup>2</sup> to Japanese.

<sup>2</sup><https://huggingface.co/kyutai/moshiko-pytorch-bf16>

We also explain data augmentation using multi-stream TTS. Through early experiments, we found that Mimi has a certain level of ability to encode and resynthesize Japanese speech without additional training, so we froze Mimi’s parameters and trained only RQ-Transformer’s parameters (Mimi’s Japanese performance is shown in Section 4).

### 3.1. Japanese Text Vocabulary Adaptation

While Moshi’s text tokenizer is a SentencePiece [17] trained on English data with a 32,000-word vocabulary, it does not include Japanese vocabulary and is therefore inefficient for tokenizing Japanese text. In this study, we adopted the SentencePiece model of Japanese GPT-2<sup>3</sup> with a 32,000-word vocabulary as the Japanese tokenizer. Additionally, as we changed the vocabulary by replacing the tokenizer, we initialized some weights of RQ-Transformer that are tied to the text vocabulary. Specifically, we randomly initialized the text token embedding tables in the Temporal Transformer and Depth Transformer, as well as the Text Linear parameters.

### 3.2. Pre-training

The purpose of pre-training is to acquire fundamental capabilities of Japanese spoken dialogue through large-scale Japanese spoken dialogue data. In this study, we adopted the J-CHAT corpus [10], which contains 69,000 hours of Japanese spoken dialogue collected from YouTube and Podcasts.

**J-CHAT Preprocessing.** Since J-CHAT’s audio data is monophonic with all speakers’ voices recorded in the same channel, it cannot be used directly for a two-channel spoken dialogue model (i.e., Moshi itself and the user). Therefore, following the method used for the original Moshi, we applied speaker diarization [18] to each audio file and created stereo spoken dialogues by randomly selecting one speaker for Moshi’s channel and the others for the user’s channel. Next, transcriptions for each channel were created using an automatic speech recognition (ASR) system.<sup>4</sup> Furthermore, we obtained timestamps for each text token using WhisperX [19] and inserted PAD tokens into timesteps without assigned text tokens in order to match the lengths of text token sequences and audio token sequences. The final data contained 3 billion text tokens, of which approximately 88% were PAD tokens.

**Training on J-CHAT.** We trained one epoch on the train set of the preprocessed J-CHAT (approximately 60,000 hours). We adopted ZeRO-3 parallelism implemented in the DeepSpeed library [20] and conducted training on 128 NVIDIA V100 32GB GPUs. Mixed precision (float16) and activation checkpointing for each Transformer layer were used. The maximum length of each input sample was set to 2.7 minutes (2,048 tokens in the time direction), with a total batch size of 512 samples. Using AdamW [21] and following Llama 2 7B [22], we set  $\beta_1 = 0.9$ ,  $\beta_2 = 0.95$ ,  $\epsilon = 1e - 5$ , and weight decay to 0.1. The learning rate was set to  $3e - 5$  with 500 steps of linear warmup. Similar to the original Moshi, in loss calculation weighting, PAD token losses were reduced by 50%, and the loss ratio between semantic tokens and acoustic tokens was set to 100 : 1. The total number of optimization steps was 8, 880, requiring 36 hours. Figure 2 shows the loss curve during training.

<sup>3</sup><https://huggingface.co/rinna/japanese-gpt2-medium>

<sup>4</sup><https://huggingface.co/reason-research/reasonspeech-espnet-v2>

Table 1: List of dialogue data used in this study

Corpus	#Dialogues	Hours
<b>Pre-training data</b>		<b>68,892</b>
J-CHAT [10]	4,937,497	68,892
<b>Fine-tuning data</b>		<b>344</b>
Japanese Callhome [23]	120	16
CSJ [24]	58	12
Travel Agency Dialogue Corpus [25]	330	115
Casual Dialogue Corpus	500	148
Consultation Dialogue Corpus	100	53
<b>Synthesized data by multi-stream TTS</b>		<b>602</b>
JapanesePersonaChat [26]	4,983	94
JapaneseEmpatheticDialogues [26]	20,000	102
Japanese Daily Dialogue [27]	5,246	44
RealPersonaChat [28]	13,510	362

### 3.3. Fine-tuning

The training data used in pre-training was forcibly divided into two channels from monophonic audio and therefore does not contain natural turn-taking such as speech overlaps and backchannels. To model actual full-duplex spoken dialogue, we performed fine-tuning using stereo spoken dialogue data where each speaker’s voice was recorded in separate channels. We prepared a total of 344 hours of spoken dialogue data using five relatively large stereo spoken dialogue corpora in Japanese:

**Japanese Callhome [23]** A corpus of casual telephone conversations. We used the remaining 16 hours, excluding some audio with insufficient transcriptions.

**CSJ [24]** A corpus containing 660 hours of Japanese speech. We used only 12 hours of two-speaker spoken dialogue.

**Travel Agency Dialogue Corpus [25]** A corpus containing dialogues recorded via Zoom meetings between travel agency operator roles and customer roles. It contains a total of 115 hours of dialogue audio.

**Casual Dialogue Corpus (in-house)** A casual dialogue corpus created within our laboratory. Each dialogue was recorded via Zoom meetings. It contains a total of 148 hours of dialogue audio from 32 speakers.

**Consultation Dialogue Corpus (in-house)** A consultation dialogue corpus created within our laboratory, similar to the casual dialogue corpus. It contains a total of 53 hours of dialogue audio from 32 speakers.

Since the above corpora span a wide range of domains, we believe that the model can handle diverse domains. Table 1 shows the breakdown of the data.

We tokenized the total 344 hours of the above data using the same method as in Section 3.2 and split it into train/valid/test sets at a ratio of 94:3:3. We then trained for 3 epochs on 320 hours of data included in the train set using 16 V100 32GB GPUs. The total number of optimization steps was 1,423, requiring 2 hours. While basic hyperparameters were the same as during pre-training, the total batch size was set to 16 samples, and the learning rates for the Temporal Transformer and Depth Transformer were set to  $2e - 6$  and  $4e - 6$ , respectively. Hereafter, we call the resulting model “J-Moshi.”

### 3.4. Data Augmentation using Multi-stream TTS

The original Moshi was trained using 20,000 hours of stereo spoken dialogue synthesized from text dialogue using multi-stream TTS [6]. We also synthesized spoken dialogue from text

dialogue using multi-stream TTS and included it in the fine-tuning data. This is expected to increase the diversity of dialogues in the training data and acquire more versatile dialogue capabilities. Following Défossez et al.’s implementation [6], we set J-Moshi’s semantic token delay to 25 timesteps and acoustic token delay to 27 timesteps, then implemented the TTS model through pre-training on J-CHAT and fine-tuning on 344 hours of stereo spoken dialogue data. All training settings were the same as in Sections 3.2 and 3.3. Below, we explain the text dialogue data and speech synthesis procedure.

**Text Dialogue Data Preparation.** As text dialogue data for speech synthesis using multi-stream TTS, we used four existing text dialogue corpora: JapanesePersonaChat [26], JapaneseEmpatheticDialogues [26], Japanese Daily Dialogue Corpus [27], and RealPersonaChat [28]. Since these corpora were collected in text chat, they contain many written language expressions and are not suitable for synthesizing spoken dialogue. Therefore, following previous research [5], we used an LLM<sup>5</sup> to rewrite the text dialogues to include spoken language-specific expressions. As a result, we rewrote all 43,739 dialogues contained in the above four corpora.

**Stereo Spoken Dialogue Generation.** Using multi-stream TTS, we generated 10 speech samples with different seed values for each text dialogue obtained in the previous step. Then, from these 10 samples, we selected the sample with the lowest Word Error Rate (WER) between the ASR result and the original dialogue text as the final speech sample for that dialogue. As a result, 602 hours of stereo spoken dialogue were synthesized. Table 1 shows the breakdown of the synthesized speech data. The overall WER of the data was 24.6%.

We fine-tuned the model pre-trained on J-CHAT using a total of 946 hours of speech data, including 344 hours of the fine-tuning data and 602 hours of the synthesized speech data. The training settings were the same as in Section 3.3, and the total number of optimization steps was 2,401. Hereafter, we refer to this model trained on the augmented data as “J-Moshi-ext.”

## 4. Experiments

We adopted the prompted dialogue continuation task [1, 7], commonly used for evaluating full-duplex spoken dialogue models, to verify the spoken dialogue generation performance of J-Moshi and J-Moshi-ext through both automatic and human evaluations. This task involves generating a continuation from a few seconds of spoken dialogue prompt. We divided each dialogue in the test set of the fine-tuning data (see Section 3.3) into 30-second chunks, and for each of the resulting 709 audio samples, we used the first 10 seconds as a prompt and had the model generate the subsequent 20 seconds.

### 4.1. Baselines

As a comparison, we adopted dGSLM [1], the typical full-duplex spoken dialogue model. For dGSLM training, like J-Moshi, we conducted pre-training using J-CHAT and fine-tuning using 344 hours of stereo spoken dialogue data. During inference, following Nguyen et al.’s settings [1], we set top- $k$  to 20. Additionally, to evaluate Mimi’s performance in Japanese spoken dialogue, we included simple re-synthesis of actual 20-second audio by Mimi (Re-synthesis). We also used the original 20-second audio (Ground-truth) as upper bound. As with the original Moshi, we tried three temperature parameters  $\tau$  of 0.8, 0.9, and 1.0 during generation.

<sup>5</sup><https://huggingface.co/google/gemma-2-27b-it>

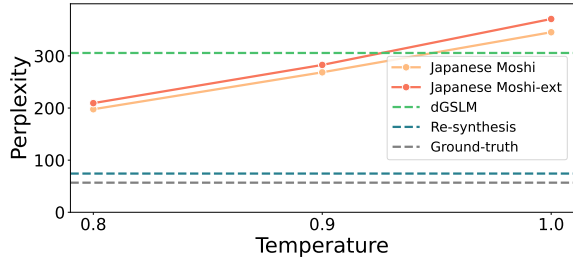


Figure 3: Perplexity of speech generated by comparison models

#### 4.2. Automatic Evaluation Results

Following previous studies [1, 6], we evaluated the model’s fluency by measuring the perplexity (PPL) of a language model<sup>6</sup> on the ASR results of speech samples generated by each model.

Results are shown in Figure 3. The PPL of J-Moshi and J-Moshi-ext revealed that, similar to the original Moshi, lower  $\tau$  values enable more fluent speech to be generated. Particularly with the  $\tau = 0.8$  setting, PPL improved by about 100 points compared to dGSLM trained from scratch on Japanese data. This suggests that adapting Moshi to Japanese positively affected speech fluency. However, compared to Re-synthesis and Ground-truth, the PPL significantly deteriorated, indicating the need for further improvements. Notably, Re-synthesis degraded minimally from Ground-truth. This suggests that Mimi can be applied directly to Japanese spoken dialogue for the first step of adapting Moshi to Japanese.

#### 4.3. Human Evaluation Results

In the human evaluation<sup>7</sup>, we used 50 audio samples randomly selected from the 709 dialogue continuations of each model. For J-Moshi and J-Moshi-ext, we used  $\tau = 0.8$ , which performed best in Section 4.2. We recruited a total of 125 crowdworkers. Each worker evaluated ten samples. Following previous studies [1, 10], they evaluated naturalness (how natural the dialogue sounds) and meaningfulness (how well the speech can be understood) on a 5-point scale.

Table 2 shows the results. For naturalness, both Japanese Moshi and J-Moshi-ext outperformed dGSLM. They also significantly surpassed dGSLM in meaningfulness. Notably, J-Moshi-ext further improved meaningfulness compared to J-Moshi, indicating that dialogue data augmentation through multi-stream TTS contributed to improving language capability. However, compared to Re-synthesis, both naturalness and meaningfulness deteriorated by more than 1 point, suggesting RQ-Transformer itself has significant room for improvement. Additionally, Re-synthesis scores degraded by about 0.5 points compared to Ground-truth, indicating that Mimi will also need to be adapted to Japanese in the future.

### 5. Comparison of Japanese and English

To analyze what kind of Japanese-specific behaviors J-Moshi acquired through our training, we compared it with the original Moshi. Specifically, we used four turn-taking statistics [1]: Inter-Pausal Units (IPU) (speech segments separated by at least 0.2 seconds of silence), Pause (silence between IPUs from the same speaker), Gap (silence between IPUs from different

<sup>6</sup><https://huggingface.co/l1lm-jp/l1lm-jp-3-3.7b>

<sup>7</sup>The experiment was approved by the ethical review committee of the Graduate School of Informatics, Nagoya University.

Table 2: Five-point evaluation scores and 95% confidence intervals for generated spoken dialogues.  $\tau$  indicates the temperature parameter during generation.

Model	$\tau$	Naturalness	Meaningfulness
dGSLM		2.44±0.12	1.76±0.09
J-Moshi	0.8	<b>2.67±0.13</b>	2.19±0.12
J-Moshi-ext	0.8	2.66±0.13	<b>2.30±0.13</b>
Re-synthesis		3.90±0.12	3.92±0.13
Ground-truth		4.46±0.09	4.45±0.10

Table 3: Turn-taking metrics per minute in generated spoken dialogues.  $\tau$  indicates the temperature parameter during generation. † indicates values reported by Défossez et al. [6].

Language	Model	Samples	$\tau$	IPU	Pause	Gap	Overlap
Japanese	J-Moshi	709	0.8	53.2s	6.3s	4.5s	<b>5.0s</b>
	J-Moshi-ext	709	0.8	50.9s	7.0s	4.6s	4.2s
	Ground-truth	709		59.7s	3.5s	4.0s	8.1s
English	Moshi†	1,000	0.8	35.1s	13.2s	12.5s	1.2s
	Ground-truth†	1,000		51.1s	6.4s	4.2s	3.3s

speakers), and Overlap (time where IPUs from different speakers overlap). For the 709 dialogue continuations generated by J-Moshi in Section 4, we calculated each measure as an average per minute. For the original Moshi, we used the values reported by Défossez et al. [6]. While the experimental conditions are not strictly identical, the comparison is considered valid as key settings such as temperature parameters are the same.

Results are shown in Table 3. We can see that spoken dialogues generated by J-Moshi have more IPUs and overlaps than Moshi. It has been reported that Japanese dialogues have more overlaps and backchannels than English [11, 12], and this tendency is supported by Ground-truth shown in Table 3. These results suggest that J-Moshi has acquired Japanese-specific turn-taking characteristics. Another difference between J-Moshi and Moshi is the sparseness of text tokens relative to audio tokens. While it was reported that the PAD token ratio was 65% in Moshi’s training data, the PAD token ratio in J-CHAT was 88% as we found in Section 3.2. This is likely due to the characteristics of Japanese where each token contains more phonemes, such as kanji characters, making the density of text tokens sparse relative to audio. This suggests the need for learning settings and objective functions that consider Japanese characteristics, such as adjusting the loss weights for PAD tokens.

### 6. Conclusion

In this study, we developed J-Moshi, a Japanese adaptation of Moshi, as a Japanese full-duplex spoken dialogue model. We conducted pre-training with 60,000 hours of Japanese speech data and fine-tuning with stereo spoken dialogue data and further attempted to improve performance using synthetic data generated through multi-stream TTS. In our experiments, we evaluated the naturalness and meaningfulness of dialogue speech generated by J-Moshi. Furthermore, through comparative analysis with Moshi, we demonstrated that J-Moshi acquired Japanese-specific behavior such as increased speech overlaps. We hope our findings will contribute to research and development of full-duplex spoken dialogue systems across multiple languages.

## 7. Acknowledgements

This work was supported by JST Moonshot R&D, Grant number JPMJMS2011. We used the computational resources of the supercomputer “Flow” at the Information Technology Center, Nagoya University.

## 8. References

- [1] T. A. Nguyen, E. Kharitonov, J. Copet, Y. Adi, W.-N. Hsu, A. Elkahky, P. Tomasello, R. Algayres, B. Sagot, A. Mohamed, and E. Dupoux, “Generative Spoken Dialogue Language Modeling,” *Transactions of the Association for Computational Linguistics*, pp. 250–266, 2023.
- [2] P. Wang, S. Lu, Y. Tang, S. Yan, W. Xia, and Y. Xiong, “A Full-duplex Speech Dialogue Scheme Based On Large Language Model,” in *Proceedings of the Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [3] Z. Ma, Y. Song, C. Du, J. Cong, Z. Chen, Y. Wang, Y. Wang, and X. Chen, “Language Model Can Listen While Speaking,” *arXiv preprint arXiv:2408.02622*, 2024.
- [4] D. Zhang, S. Li, X. Zhang, J. Zhan, P. Wang, Y. Zhou, and X. Qiu, “SpeechGPT: Empowering Large Language Models with Intrinsic Cross-Modal Conversational Abilities,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 15 757–15 773.
- [5] Q. Fang, S. Guo, Y. Zhou, Z. Ma, S. Zhang, and Y. Feng, “LLaMA-Omni: Seamless Speech Interaction with Large Language Models,” in *Proceedings of the Thirteenth International Conference on Learning Representations*, 2025.
- [6] A. Défossez, L. Mazaré, M. Orsini, A. Royer, P. Pérez, H. Jégou, E. Grave, and N. Zeghidour, “Moshi: a speech-text foundation model for real-time dialogue,” *arXiv preprint arXiv:2410.00037*, 2024.
- [7] B. Veluri, B. N. Peloquin, B. Yu, H. Gong, and S. Gollakota, “Beyond Turn-Based Interfaces: Synchronous LLMs as Full-Duplex Dialogue Agents,” in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024, pp. 21 390–21 402.
- [8] Q. Zhang, L. Cheng, C. Deng, Q. Chen, W. Wang, S. Zheng, J. Liu, H. Yu, and C. Tan, “Omniflatten: An end-to-end gpt model for seamless voice conversation,” *arXiv preprint arXiv:2410.17799*, 2024.
- [9] W. Yu, S. Wang, X. Yang, X. Chen, X. Tian, J. Zhang, G. Sun, L. Lu, Y. Wang, and C. Zhang, “SALMONN-omni: A Code-free LLM for Full-duplex Speech Understanding and Generation,” *arXiv preprint arXiv:2411.18138*, 2024.
- [10] W. Nakata, K. Seki, H. Yanaka, Y. Saito, S. Takamichi, and H. Saruwatari, “J-CHAT: Japanese Large-scale Spoken Dialogue Corpus for Spoken Dialogue Language Modeling,” *arXiv preprint arXiv:2407.15828*, 2024.
- [11] R. Hayashi, “Simultaneous talk—from the perspective of floor management of English and Japanese speakers,” *World Englishes*, vol. 7, no. 3, pp. 269–288, 1988.
- [12] M. Stubbe, “Are you listening? cultural influences on the use of supportive verbal feedback in conversation,” *Journal of Pragmatics*, vol. 29, no. 3, pp. 257–289, 1998.
- [13] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, “High Fidelity Neural Audio Compression,” *Transactions on Machine Learning Research*, 2023.
- [14] M. Tagliasacchi, Y. Li, K. Misiunas, and D. Roblek, “SEANet: A multi-modal speech enhancement network,” *arXiv preprint arXiv:2009.02095*, 2020.
- [15] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, “Soundstream: An end-to-end neural audio codec,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 495–507, 2021.
- [16] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. Mcleavey, and I. Sutskever, “Robust Speech Recognition via Large-Scale Weak Supervision,” in *Proceedings of the 40th International Conference on Machine Learning*, 2023, pp. 28 492–28 518.
- [17] T. Kudo and J. Richardson, “SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2018, pp. 66–71.
- [18] H. Bredin, “pyannote.audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe,” in *Proceedings of the 24th INTERSPEECH Conference*, 2023, pp. 1983–1987.
- [19] M. Bain, J. Huh, T. Han, and A. Zisserman, “WhisperX: Time-Accurate Speech Transcription of Long-Form Audio,” *arXiv preprint arXiv:2303.00747*, 2023.
- [20] S. Rajbhandari, J. Rasley, O. Ruwase, and Y. He, “ZeRO: memory optimizations toward training trillion parameter models,” in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2020.
- [21] I. Loshchilov and F. Hutter, “Decoupled Weight Decay Regularization,” in *Proceedings of the Sixth International Conference on Learning Representations*, 2019.
- [22] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esioiu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom, “Llama 2: Open foundation and fine-tuned chat models,” *arXiv preprint arXiv:2307.09288*, 2023.
- [23] A. Canavan and G. Zipperlen, “CALLHOME Japanese Speech LDC96S37,” Philadelphia: Linguistic Data Consortium, 1996.
- [24] K. Maekawa, “Corpus of Spontaneous Japanese: Its design and evaluation,” in *Proceedings of ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2003.
- [25] M. Inaba, Y. Chiba, Z. Qi, R. Higashinaka, K. Komatani, Y. Miyao, and T. Nagai, “Travel agency task dialogue corpus: A multimodal dataset with age-diverse speakers,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 23, no. 9, pp. 1–23, 2024.
- [26] H. Sugiyama, M. Mizukami, T. Arimoto, H. Narimatsu, Y. Chiba, H. Nakajima, and T. Meguro, “Empirical Analysis of Training Strategies of Transformer-Based Japanese Chat-Chat Systems,” in *Proceedings of 2022 IEEE Spoken Language Technology Workshop*, 2023, pp. 685–691.
- [27] R. Akama, Y. Isobe, J. Suzuki, and K. Inui, “Construction of a Japanese Daily Dialogue Corpus,” in *Proceedings of the 29th Annual Conference of the Association for Natural Language Processing*, 2023, pp. 108–113, (in Japanese).
- [28] S. Yamashita, K. Inoue, A. Guo, S. Mochizuki, T. Kawahara, and R. Higashinaka, “RealPersonaChat: A realistic persona chat corpus with interlocutors’ own personalities,” in *Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation*, 2023, pp. 852–861.