Incorporating Dialogue State Tracking into Japanese Full-duplex Task-oriented Spoken Dialogue Model

Yuya Chiba[1], **Ryuichiro Higashinaka**[2]
[1]NTT Communication Science Laboratories, NTT, Inc., Japan
[2]Graduate School of Informatics, Nagoya University, Japan
yuya.chiba@ntt.com, higashinaka@i.nagoya-u.ac.jp

Abstract

Full-duplex spoken dialogue models, which process audio input and output simultaneously, have been actively studied for their ability to naturally model turn-taking and non-verbal phenomena in addition to generating responses. Although these models enable natural conversational flow, they lack mechanisms for language understanding and dialogue management, making them difficult to apply to task-oriented dialogue systems. We propose a method for incorporating dialogue state tracking in task-oriented dialogue into Moshi, aiming to achieve a multi-channel, full-duplex task-oriented spoken dialogue model. We evaluated the proposed method on JMultiWOZ, a benchmark corpus for Japanese task-oriented dialogue, focusing on dialogue state tracking and response generation.

1 Introduction

Research on spoken dialogue models has been actively pursued not only to improve response generation but also enhance the naturalness of speech communication (Nguyen et al., 2023; Veluri et al., 2024; Yu et al., 2025). Full-duplex spoken dialogue models can naturally handle turn-taking phenomena, such as utterance overlaps and backchannels, by processing audio input and output in parallel.

The representative full-duplex spoken dialogue model, Moshi (Défossez et al., 2024; Ohashi et al., 2025) represents user and system audio input/output as multi-channel streams. This design enables the model to be trained directly on spoken dialogue data and has been shown to enable smooth turn-taking. However, Moshi lacks a mechanism for dialogue control, therefore has difficulty adapting to complex task-oriented dialogues that require interaction with external knowledge sources such as a domain-dependent database.

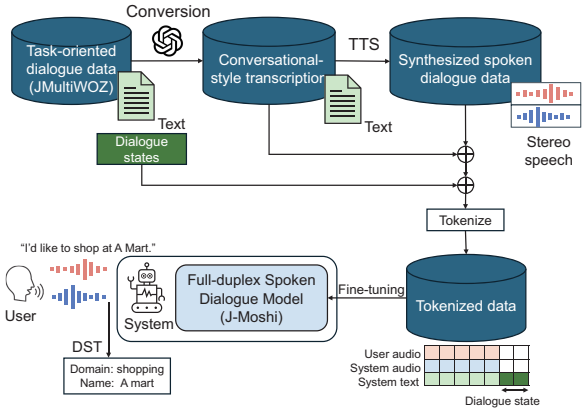

Figure 1: Data construction and training full-duplex spoken dialogue model for dialogue state tracking

To address this issue, we propose a method for incorporating dialogue state tracking (DST) into Moshi by using its text stream. DST is a comprehensive task that encompasses essential functions of task-oriented dialogue systems, including user's intent inference and response generation based on external information (i.e., database search results). With the proposed method, the model continuously observes user and system utterances and estimates the dialogue state at appropriate times. The model then generates system responses on the basis of the database search results. An overview of the model training is shown in Fig. 1. For training, we use pseudo-spoken dialogue data synthesized from text-based task-oriented dialogues. To the best of our knowledge, this is the first study to incorporate DST into a full-duplex spoken dialogue model.

2 Related Studies

Various spoken dialogue models have been proposed that process user and system speech using large language models (LLMs). Zhang et al. (2023) introduced SpeechGPT, which operates on discretized audio tokens. Models such as LLaMA-

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 829–836

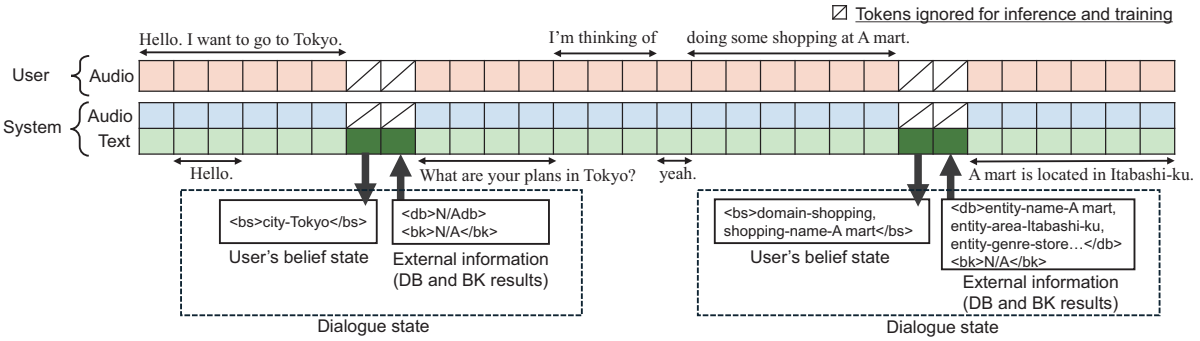

Figure 2: Token sequences for full-duplex task-oriented dialogue model. Dialogue state consists of user's belief state and external knowledge used to control task-oriented dialogue system's response. Each component of dialogue state is separated by special tokens that mark beginning and end of each segment (e.g., <bs> and </bs>).

Omni (Fang et al., 2024), Qwen2.5-Omni (Xu et al., 2025), and GLM-4-Voice (Zeng et al., 2024) have also been developed to handle both text and audio tokens as input and output. These models are designed for turn-based interactions and process input and output in a half-duplex manner.

In contrast, models that process user and system speech in parallel are referred to as full-duplex models. Nguyen et al. (2023) proposed dGSLM, a model that simultaneously processes user and system audio tokens. Recent studies have actively explored models for incorporating linguistic information such as SALMONN-omni (Yu et al., 2025). Moshi (Défossez et al., 2024) integrates text tokens of system utterances in addition to audio tokens. Benchmarking methods for evaluating such models have also recently been introduced (Lin et al., 2025; Arora et al., 2025). In this study, we used J-Moshi (Ohashi et al., 2025), a version of Moshi fine-tuned for Japanese. Although this paper focuses on Japanese, similar experiments can be conducted in English.

3 Method

Moshi's spoken language model consists of two Transformers: the Temporal Transformer and Depth Transformer. The input and output consist of sequences of text and audio tokens from the system, along with audio tokens from the user. The Temporal Transformer models token sequences along the time dimension and autoregressively outputs a temporal context vector z_s for each time step. This vector is then converted into text tokens representing the system's utterance via a text-linear layer. The Depth Transformer takes z_s as input and sequentially generates audio tokens for both system and user speech. The Temporal Transformer is based on a 7B LLM and is further fine-tuned on large-scale spoken dialogue data to bridge the modality gap between text and audio.

With our proposed method, both text generation and DST are executed within Moshi's text stream. Figure 2 shows an example of the token sequences handled with the proposed method. The model infers system and user audio tokens autoregressively, but when a token indicating the start of a belief state (<bs>) is generated, it begins estimating the belief state and continues until the token indicating the end (</bs>) is generated. External information (such as database search results) is then provided to the text stream, and text and audio token inference resumes. During DST, the Depth Transformer is deactivated, and only text tokens are input and output.

We fine-tune J-Moshi using dialogue data annotated with dialogue states to equip it with DST capability.

4 Data Construction

MultiWOZ (Budzianowski et al., 2018) is a well-known benchmark for task-oriented dialogue systems. We used JMultiWOZ (Ohashi et al., 2024), a Japanese version of MultiWOZ, to evaluate our proposed method. Since the dialogues in JMultiWOZ are in text format, they must be converted into stereo spoken dialogue data for fine-tuning. Below, we describe the procedure for synthesizing spoken dialogues and inserting dialogue states into the token sequences.

4.1 Overview of JMultiWOZ

JMultiWOZ consists of dialogues in which a traveler (user) and operator (system) plan a trip through conversation. Users engage in conversa-

Original:

User: I'd like you to look up tourist spots in Kyoto. Are there any places with free parking?
System: There are no tourist attractions in Kyoto that offer free parking.

Dialogue State:
Belief state: general-city-Kyoto, parking-available (free)
DB result: N/A **BK result:** N/A

Converted:

User: Um, I'd like you to look up places to visit in Kyoto.
System: Sure.
User: Are there any places with free parking?
System: Well, in Kyoto, there actually aren't any tourist spots with free parking.

Dialogue State:
Belief state: general-city-Kyoto, parking-available (free)
DB result: N/A **BK result:** N/A

Figure 3: Example of original and conversational-style dialogue (translated from Japanese). Belief state, DB result, and BK result represent corresponding dialogue state.

tions to achieve randomly assigned dialogue goals, while the operator searches a travel-information database and provides detailed information about the retrieved entities.

Each user turn is annotated with a dialogue state. The dialogue state consists of three components: the user's belief state, database search results (DB result), and success or failure of facility booking (BK result). The upper part of Fig. 3 shows an example of an exchange between the user and system along with the corresponding dialogue state as key-value pairs.

The corpus contains a total of 4,254 dialogues. The dialogue data are divided into train/dev/test sets, with 3,654/300/300 dialogues, respectively.

4.2 Synthesizing spoken dialogue

Since the original dialogue text does not contain speech-specific phenomena such as fillers and backchannels, it is converted into a conversational style using `gpt-4o-mini`. The prompts for the conversion are shown in the Appendix A. We design the conversion instructions on the basis of a previous study (Shimazu et al., 2014). The instructions include Japanese speech-specific phenomena and their explanations, as well as examples of conversion. We also instruct that interjections be inserted at appropriate points within the utterance using angle brackets. The utterance is then split at the position of the angle brackets, with the interjection treated as a separate utterance of the other speaker. The lower part of Fig. 3 presents a converted dialogue and the corresponding dialogue states.

Next, we synthesize stereo audio using the conversational-style dialogue. In Moshi's training, it is desirable for the system's speech to be fixed to a single speaker, while the user's speech is synthesized from an unspecified number of speakers. Therefore, we use OuteTTS[1], a zero-shot text-to-speech (TTS) system capable of synthesizing conversational Japanese speech. For system-speech synthesis, we use the speech of a single female speaker selected from an internal corpus as an acoustic prompt. For user-speech synthesis, we generate the first utterance of each dialogue without using a prompt. For subsequent utterances, we use the first utterance as a prompt to maintain speaker consistency throughout the dialogue. Stereo spoken dialogue data are generated by combining these synthesized speech signals using a method described in previous studies (Reece et al., 2023; Lee et al., 2025). With that method, the interval between the preceding and following utterances is determined probabilistically. The synthesized dialogues are then tokenized after aligning the text and audio in time.

4.3 Insertion of dialogue state

The dialogue state is first converted into a key-value format then tokenized using J-Moshi's text tokenizer. Each dialogue state is marked with special tokens indicating the start and end, which are inserted after the user utterance from which the dialogue state was derived. To exclude audio tokens within the dialogue state segments from the loss calculation, mask tokens are used in place of audio tokens in those segments.

5 Experiment

5.1 Experimental settings

We objectively evaluated the DST and response generation of the model trained with the proposed method on the test data. For the evaluation of DST, gold data are provided from the beginning of the dialogue up to the <bs> token, and the output is defined as the sequence generated until the </bs> token appears. The dialogue state generated during this interval is compared with the reference dialogue state. To reduce the number of tokens, we exclusively retain the key-value pairs with non-null values as dialogue states.

For response generation, the gold data are provided for the segment from the beginning of the dialogue up to the </bk> token, which marks the

[1] https://github.com/edwko/OuteTTS

Base model	Ratio [%]	DST		Response generation					DST timing
		JGA	Slot F1	BLEU	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	MAE [sec.]
J-Moshi-ext (Proposed)	25	34.13	94.12	0.073	0.266	0.108	0.232	0.658	5.407
	50	50.55	96.45	0.092	0.302	0.138	0.271	0.676	5.069
	75	60.40	97.43	0.109	0.325	0.156	0.295	0.685	5.088
	100	**65.69**	**97.78**	**0.129**	**0.340**	**0.173**	**0.312**	**0.689**	**4.900**
T5-base	100	74.07	98.52	0.334	0.585	0.434	0.564	0.817	–
T5-large	100	84.45	99.16	0.364	0.614	0.463	0.592	0.830	–

Table 1: Average scores of objective evaluations. Bold text indicates best evaluation results for model trained from J-Moshi-ext (proposed). T5-base and T5-large show results representing text-based upper bound.

end of the external information input. The output is generated until the next <bs> token appears. For response evaluation, the pad tokens (i.e., PAD and EPAD tokens of the original Moshi (Défossez et al., 2024)) are removed from both the generated responses and reference responses. The model must also infer the timing of <bs> token generation during actual deployment. Therefore, we evaluated the timing of DST initiation by calculating the mean absolute error (MAE) between the time the <bs> token was generated during response generation and the reference time.

We conducted the experiment by varying the amount of training data to evaluate performance improvement. We further assessed the language capability of the model trained with the proposed method by comparing its performance with that of T5, a state-of-the-art text-based model for JMultiWOZ (Ohashi et al., 2024).

5.2 Training conditions

We used `J-Moshi-ext`, the best-performing variant of J-Moshi, as the base model for fine-tuning. The text-embedding and text-linear layers of the Temporal Transformer, as well as the text-embedding layer of the Depth Transformer, were extended to handle special tokens representing dialogue states. Each speech channel and the system text of the dialogue data were tokenized using J-Moshi's text and audio tokenizers. Text and audio were aligned using `whisper-large-v3`.

The training data were randomly split into proportions of 25, 50, 75, and 100%, and the splitting and training process was repeated three times. The experimental results are reported as the average of these trials. Under each condition, the model was fully fine-tuned using 12 NVIDIA V100 32-GB GPUs. The optimizer was AdamW with the learning rates of 2e-6 and weight decay of 0.01. We set the warmup steps to 30 and gradient accumulation step to 1. The number of training epochs was set to 10. The total batch size was 24.

We used `retrieva-jp/t5-base-long` and `retrieva-jp/t5-large-long` as base models for the text-based upper bound. These models were fine-tuned on the same text data as used with the proposed method, except that the PAD and EPAD tokens were removed.

5.3 Experimental results

Table 1 presents the objective evaluation results. For DST, we used joint goal accuracy (JGA) and Slot F1 as metrics. Since Slot F1 is calculated over all slots, including those with empty values, it generally yields a high score. The results indicate that both metrics tend to improve as the amount of training data increases with the proposed method. The final scores reached 65.69 for JGA and 97.78 for Slot F1, indicating that the model trained with the proposed method can achieve a reasonable level of DST performance. Since the proposed method processes only text tokens using the Temporal Transformer during DST, it likely effectively leverages the underlying language model in J-Moshi.

For response generation, we evaluated with BLEU[2], ROUGE[3], and BERTScore[4]. We used `cl-tohoku/bert-base-japanese` as the reference model for calculating BERTScore. The results indicate that while these metrics improve with more training data, the BLEU remained low at 0.129. Among all evaluation metrics, response generation showed the greatest room for improvement compared with the text-based upper bound. This is likely due to the amount of training data used in this study being insufficient to fully capture the relationship between text and audio tokens in response generation. Therefore, it is necessary to explore more effective methods for bridging the gap between text and audio modalities.

A substantial discrepancy is observed between

[2] https://pypi.org/project/evaluate/
[3] https://pypi.org/project/rouge-score/
[4] https://pypi.org/project/bert-score/

Positive sample:

User: I'd like to make a reservation for February 21st at 1:20 p.m. for two people.
(2/21の13:20で予約お願いしまーす。人数は2人です。)
BK result: restaurant-success-True, restaurant-ref-**36832**

Generated: Certainly. Your reservation has been completed. Your reservation number is **36832**.
(かしこまりました。予約が完了しました。予約番号は**36832**になります。)
Reference: Certainly. Your reservation has been made. Your reservation number is **36832**.
(かしこまりました。ご予約しました。予約番号は**36832**です。)

Negative samples:

User: I'd like to know the opening hours there.
(そこの営業時間、知りたいんですが。)
DB result: city-sendai, name-rakutei, genre-izakaya, genre-okonomiyaki, genre-nabe,area-taihaku-ku, pricerange-normal, station-nagamachi, wifi-yes (free), parking-yes(free), opentime-**17:00-23:00**, phone-0227960307, accesstime-nagamachi-taxi-5, closed-tuesday, domain-restaurant

Generated: It's open **from 10 a.m. to 8 p.m.** Yes.
(**10**時から**8**時ですはい。)
Reference: It says it's open **from 5 p.m. to 11 p.m**.
(夕方の**5**時から夜の**11**時までってなってますよ。)

User: Um, I'm thinking of going shopping for stationery and miscellaneous goods in Nagoya next time. Could you recommend a good store near Nagoya Station?
(えっと、今度名古屋で文房具や雑貨の買い物したいなーって思ってて、名古屋駅近くで良いお店教えてもらえますか)
Belief state: general-active_domain-shopping, general-city-nagoya shopping-genre-stationery, shopping-station-nagoya
DB result: city-nagoya, name-**tokyu hands nagoya**, genre-stationery, genre-fashion, genre-pharmaceuticals & cosmetics, genre-travel goods genre-other, area-nakamura-ku, station-nagoya, parking-yes (free), opentime-10:00-20:00, phone-0525660109, accesstime-Nagoya-on foot-1, closed-irregular holidays, domain-shopping

Generated: Yes, money A chiyaku and Nagoya taxi can be used, how about Mankai?
(はい、お金 ア地訳と名古屋タクシーが使えて漫会なんかどうですか?)
Reference: When it comes to stationery and general goods near the station, **Tokyu Hands Nagoya** is definitely the best recommendation.
(駅近くの文房具雑貨って言ったら 東急ハンズ名古屋店これが一番おすすめですね。)

Figure 4: Positive and negative examples of response generation. Parentheses indicate original Japanese sentences. *User* denotes user's utterance, *Generated* denotes generated response to it, and *Reference* denotes correct response. Figure shows only dialogue states related to response generation (i.e., DB result and BK result). Bold text in utterance indicates that information should be referred to from dialogue states.

DST timings of the predicted result and reference. This is considered due to the fact that the timing of <bs> generation depends on the user utterances predicted with the model. In task-oriented dialogue, user utterances are determined on the basis of the user's goals, making them difficult to predict with Moshi's architecture. To accurately assess the model's turn-taking performance, it is necessary to conduct evaluations on the basis of interactions with real users.

Figure 4 presents positive and negative examples of response generation. In the positive example, the model successfully generates a sentence almost identical to the reference and correctly refers to the reservation number contained in the BK result (the information about booking; see Section 4.1). The remaining two are negative examples. In the first case, although the model produces a response similar to the reference, it fails to correctly retrieve the opening-hour information from the DB result. In the second case, the model generates a linguistically incorrect response. One possible reason for such outputs is the limited performance of the speech synthesis model for Japanese that was used to construct the dialogue data. Because the speech synthesizer used in this study was a multilingual model, it tended to generate Japanese speech with reduced clarity. Consequently, recognition errors were also observed in the acquisition of speech recognition results with time annotations. These errors are considered to have hindered both the alignment between the database search results and system utterances, as well as the learning of linguistically coherent token sequences.

6 Conclusions

This study introduced the DST task into a full-duplex spoken dialogue model. The results indicate that the model trained with the proposed method can achieve a reasonable level of DST performance. However, more effective methods are needed to bridge the modality gap in response generation and DST-timing prediction.

Limitations

For this study, the experiment was conducted in Japanese. The base model, J-Moshi, was trained with less target-language data compared with the original Moshi. Therefore, although the results of this experiment are considered consistent with the overall trend, higher performance in DST and response generation may be achievable by using Moshi with English corpora such as Multi-

WOZ. Spoken dialogue datasets based on MultiWOZ, such as SpokenWOZ (Si et al., 2023) and DSTC11 (Soltau et al., 2023), have also been constructed. We plan to conduct experiments using these corpora and the original Moshi to examine language dependency. While the proposed method is considered generally applicable to task-oriented dialogue systems, the experiment was limited to the domains included in JMultiWOZ. Therefore, we also plan to apply our method to more diverse domains by using task-oriented dialogue datasets such as CamRest (Wen et al., 2016) and the Schema-Guided Dialogue dataset (Rastogi et al., 2020) to address domain-specific issues individually. This paper reported only objective evaluations. In practical deployment, each step in the system, including DST, database search, and external information insertion, involves additional processing time. It is thus necessary to investigate how these processing delays affect user evaluation, for example, by comparing our method with conventional real-time spoken dialogue systems (Michael, 2020; Chiba and Higashinaka, 2025; Kennington et al., 2025).

Ethical Considerations

We used publicly available datasets, models, and a TTS system. All artifacts were used in accordance with their respective licenses and terms of use. The speakers of the speech used as prompts for the TTS system were selected from an internal corpus, and consent was obtained for the use of their speech for research purposes. Therefore, there are few ethical concerns regarding the experiment we conducted. However, if a task-oriented dialogue system capable of natural interaction, as envisioned in this study, is made possible, it may raise concerns such as impersonation or identity fraud. When applying the proposed method, it is essential to implement measures to prevent misuse including the generation of offensive or privacy-invasive outputs and voice cloning.

Acknowledgments

This work was supported by JST Moonshot R&D Grant Number JPMJMS2011.

References

Siddhant Arora, Zhiyun Lu, Chung-Cheng Chiu, Ruoming Pang, and Shinji Watanabe. 2025. Talking turns: Benchmarking audio foundation models on turn-taking dynamics. *arXiv preprint arXiv:2503.01174*, pages 1–28.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ–a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proc. EMNLP*, pages 5016–5026.

Yuya Chiba and Ryuichiro Higashinaka. 2025. Investigating the impact of incremental processing and voice activity projection on spoken dialogue systems. In *Proc. COLING*, pages 3687–3696.

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. *arXiv preprint arXiv:2410.00037*, pages 1–66.

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*, pages 1–18.

Casey Kennington, Pierre Lison, and David Schlangen. 2025. Prior lessons of incremental dialogue and robot action management for the age of language models. *arXiv preprint arXiv:2501.00953*, pages 1–33.

Sehun Lee, Kang-wook Kim, and Gunhee Kim. 2025. Behavior-SD: Behaviorally aware spoken dialogue generation with large language models. In *Proc. NAACL*, pages 9574–9593.

Guan-Ting Lin, Jiachen Lian, Tingle Li, Qirui Wang, Gopala Anumanchipalli, Alexander H Liu, and Hung-yi Lee. 2025. Full-duplex-bench: A benchmark to evaluate full-duplex spoken dialogue models on turn-taking capabilities. *arXiv preprint arXiv:2503.04721*, pages 1–18.

Thilo Michael. 2020. RETICO: An incremental framework for spoken dialogue systems. In *Proc. SIGDIAL*, pages 49–52.

Tu Anh Nguyen, Eugene Kharitonov, Jade Copet, Yossi Adi, Wei-Ning Hsu, Ali Elkahky, Paden Tomasello, Robin Algayres, Benoit Sagot, Abdelrahman Mohamed, and Emmanuel Dupoux. 2023. Generative spoken dialogue language modeling. *Transactions of the Association for Computational Linguistics*, 11:250–266.

Atsumoto Ohashi, Ryu Hirai, Shinya Iizuka, and Ryuichiro Higashinaka. 2024. JMultiWOZ: A large-scale Japanese multi-domain task-oriented dialogue dataset. In *Proc. LREC-COLING*, pages 9554–9567.

Atsumoto Ohashi, Shinya Iizuka, Jingjing Jiang, and Ryuichiro Higashinaka. 2025. Towards a Japanese full-duplex spoken dialogue system. *Proc. INTERSPEECH*, pages 1783–1787.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proc. AAAI*, volume 34, pages 8689–8696.

Andrew Reece, Gus Cooney, Peter Bull, Christine Chung, Bryn Dawson, Casey Fitzpatrick, Tamara Glazer, Dean Knox, Alex Liebscher, and Sebastian Marin. 2023. The CANDOR corpus: Insights from a large multimodal dataset of naturalistic conversation. *Science Advances*, 9(13):eadf3197.

Akira Shimazu, Mikio Nakano, Koji Dosaka, and Masahito Kawamori. 2014. *Computational model of spoken dialogue*. The Institute of Electronics, Information and Communication Engineers. (in Japanese).

Shuzheng Si, Wentao Ma, Haoyu Gao, Yuchuan Wu, Ting-En Lin, Yinpei Dai, Hangyu Li, Rui Yan, Fei Huang, and Yongbin Li. 2023. SpokenWOZ: A large-scale speech-text benchmark for spoken task-oriented dialogue agents. *Proc. NeurIPS*, 36:39088–39118.

Hagen Soltau, Izhak Shafran, Mingqiu Wang, Abhinav Rastogi, Wei Han, and Yuan Cao. 2023. DSTC-11: Speech aware task-oriented dialog modeling track. In *Proc. The Eleventh Dialog System Technology Challenge*, pages 226–234.

Bandhav Veluri, Benjamin N Peloquin, Bokai Yu, Hongyu Gong, and Shyamnath Gollakota. 2024. Beyond turn-based interfaces: Synchronous LLMs as full-duplex dialogue agents. In *Proc. EMNLP*, pages 21390–21402.

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, David Vandyke, and Steve Young. 2016. Conditional generation and snapshot learning in neural dialogue systems. In *Proc. EMNLP*, pages 2153–2162.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. 2025. Qwen2.5-Omni technical report. *arXiv preprint arXiv:2503.20215*, pages 1–20.

Wenyi Yu, Siyin Wang, Xiaoyu Yang, Xianzhao Chen, Xiaohai Tian, Jun Zhang, Guangzhi Sun, Lu Lu, Yuxuan Wang, and Chao Zhang. 2025. SALMONN-omni: A standalone speech LLM without codec injection for full-duplex conversation. *arXiv preprint arXiv:2505.17060*, pages 1–20.

Aohan Zeng, Zhengxiao Du, Mingdao Liu, Kedong Wang, Shengmin Jiang, Lei Zhao, Yuxiao Dong, and Jie Tang. 2024. GLM-4-Voice: Towards intelligent and human-like end-to-end spoken chatbot. *arXiv preprint arXiv:2412.02612*, pages 1–14.

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. In *Findings of EMNLP*, pages 15757–15773.

A Prompt

Please rewrite the following transcription of a conversation between two speakers, A and B, originally written in formal written language, into a natural spoken dialogue.

The following features are known to be characteristic of spoken language. Please take them into account during the conversion.
You may exaggerate the degree of spoken language if necessary.
Note that spoken language is not the same as casual language, so avoid representing it merely by adding elongated vowels ("ー") or exclamation marks.

Characteristic Features of Spoken Language

(i) Lexical phenomena
- 1. Phonological contractions: Written expressions transformed into more colloquial forms
- 2. Sentence-final particles: Particles such as "ne", "ka", "yo ne", "yo", "no", "kana", "na", "kke" often appear at the end of utterances, or inserted mid-utterance (e.g., "ne", "sa"). In polite expressions, "desu ne" may also be used.
- 3. Colloquial particles: Particles like "tte" or "toka" which appear only in spoken language
- 4. Interjections: Expressions of surprise ("ah", "eh"), backchannels ("hai", "ee"), and fillers ("etto", "anoo")

(ii) Omissions
- 1. Case particle omission: So-called "bare case" nouns without postpositional particles
- 2. Predicate omission: Omitting copulas like "da" or "desu"

(iii) Redundant expressions
- 1. Restatements and repetitions: Canceling and rephrasing utterances

(iv) Other phenomena
- 1. Noun or particle-only utterances
- 2. Utterance ending with a conjunctive particle
- 3. Twisted construction: Grammatically one sentence, but semantically awkward
- 4. Anticipation: One speaker starts and another finishes the utterance
- 5. Inversion: Later parts of the utterance semantically modify earlier parts
- 6. Counting items
- 7. Spelling explanation: Explaining how to write characters or spell English words

Below is an example of such a conversion.
If inserting backchannels like "hai" or "ee" from the other speaker during one speaker's utterance, please use angle brackets (e.g., <hai>) as in the example.
Actively insert backchannels and paraphrasing expressions at appropriate places.
Please preserve the number of utterance lines.

Input Example
{Written dialogue example}

Converted Example
{Spoken dialogue example}

Input
{input_text}

Figure 5: Prompt for converting written dialogue into natural conversational-style dialogue (translated from Japanese).