OmniCharacter: Towards Immersive Role-Playing Agents with Seamless Speech-Language Personality Interaction

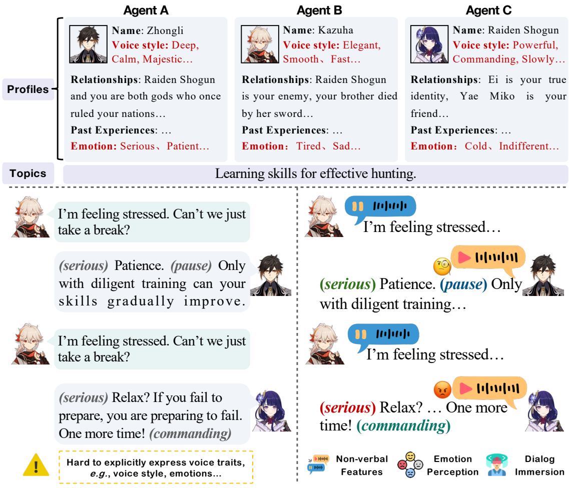
Anonymous ACL submission

Abstract

Role-Playing Agents (RPAs), benefiting from large language models, is an emerging interactive AI system that simulates roles or characters with diverse personalities. However, existing methods primarily focus on mimicking dialogues among roles in textual form, neglecting the role's voice traits (*e.g.,* voice style and emotions) as playing a crucial effect in interaction, which tends to be more immersive experiences in realistic scenarios. Towards this goal, we propose ***OmniCharacter***, a first seamless speech-language personality interaction model to achieve immersive RPAs with low latency. Specifically, OmniCharacter enables agents to consistently exhibit role-specific personality traits and vocal traits throughout the interaction, enabling a mixture of speech and language responses. To align the model with speech-language scenarios, we construct a dataset named ***OmniCharacter-10K***, which involves more distinctive characters (20), richly contextualized multi-round dialogue (10K), and dynamic speech response (135K). Experimental results showcase that our method yields better responses in terms of both content and style compared to existing RPAs and mainstream speech-language models, with a response latency as low as 289ms.[1]

1 Introduction

The rapid advancement of Large language models (LLMs) (Wang et al., 2024b; Park et al., 2023; OpenAI, 2022) has demonstrated strong potential in interactive AI, enabling more natural and engaging human-computer interactions. One of the most promising and widely adopted research directions is the development of role-playing agents (RPAs) (Wang et al., 2024a; Tu et al., 2024; Shao et al., 2023), which have been applied in various domains, such as virtual assistants, AI-driven storytelling, and intelligent non-player characters (NPCs) in video games. The focus of RPAs allows LLMs to simulate diverse personas and mimic human-like behavior via predefined role profiles.


Agent A
Agent B
Agent C
Profiles
Name: Zhongli
Voice style: Deep, Calm, Majestic…
Relationships: Raiden Shogun and you are both gods who once ruled your nations…
Past Experiences: …
Emotion: Serious、Patient…
Name: Kazuha
Voice style: Elegant, Smooth、Fast…
Relationships: Raiden Shogun is your enemy, your brother died by her sword…
Past Experiences: …
Emotion: Tired、Sad…
Name: Raiden Shogun
Voice style: Powerful, Commanding, Slowly…
Relationships: Ei is your true identity, Yae Miko is your friend…
Past Experiences: …
Emotion: Cold、Indifferent…
Topics
Learning skills for effective hunting.
I'm feeling stressed. Can't we just take a break?
(serious) Patience. (pause) Only with diligent training can your skills gradually improve.
I'm feeling stressed. Can't we just take a break?
(serious) Relax? If you fail to prepare, you are preparing to fail. One more time! (commanding)
Hard to explicitly express voice traits, e.g., voice style, emotions…
I'm feeling stressed…
(serious) Patience. (pause) Only with diligent training…
I'm feeling stressed…
(serious) Relax? … One more time! (commanding)
Non-verbal Features
Emotion Perception
Dialog Immersion

(a) Text-based PRP (b) Our Method

Figure 1: (a) Existing role-playing agents (RPAs) concentrate on engaging dialogue in textual form, whereas the role's voice traits are often ignored. (b) Our method considers the importance of the characters' vocal persona, *e.g.,* voice style and emotions, delivering a more seamless and immersive speech-language interaction.

Despite significant advances in this field, most existing studies primarily concentrate on replicating dialogues solely in textual form, while often overlooking the role's crucial voice traits such as tone, style, emotions, and pauses. The voice traits are important paralinguistic concepts that deeply reveal individual differences in conversations, enabling a more immersive and personalized role-playing interaction. As shown in Figure 1 (a), for the same user query, although both agents provide reasonable responses, their unique tones, *i.e.*, Agent A's serious yet calmed and Agent B's angry and commanding, are difficult to convey through text alone, limiting their role-specific expression. Therefore, incorporating personalized voice traits

[1] Code, dataset, and demo are available at https://anonymous.4open.science/r/OmniCharacter-BB80/.

into the current RPAs can help create more realistic and immersive interactions.

To bridge this gap, we introduce ***OmniCharacter***, a first step in realizing speech-language collaborative RPAs for immersive and seamless interaction. Specifically, a **Speech-Language Collaborative Model** is first built as a base model to perceive both language inputs (*i.e.*, role profile, dialogue contexts, and user text input) and speech input (*i.e.*, user speech input) for semantic alignment. Based on this, we propose a **Role Speech Decoder** to generate role-specific speech responses containing their unique voice traits. In particular, it mainly consists of two major components: 1) Role-context Guided Speech Token Prediction, which leverages the textual representations from LLM as prior, ensuring the alignment of discrete speech token generation with the corresponding textual contexts. 2) Role-aware Speech Synthesis, which allows to generate waveform containing rich character-related voice traits from speech tokens specific to the given conditions (*i.e.*, profile, dialogue contexts, current query, and speaker embedding). It is worth noting that the proposed model is able to generate text and speech responses in an auto-regressive manner, resulting in a latency as low as 289ms, ensuring an immersive and seamless dialogue interaction.

To facilitate the development of OmniCharacter, we further present the ***OmniCharacter-10K***, a speech-language RPAs dataset with detailed profile annotations, diverse dialogues, and vivid audio responses tailored to each role's characteristic. This dataset presents several appealing properties: 1) ***Large Vocabulary***: It includes a total of 20 characters along with 10,493 multi-turn dialogues, along with 135K audio responses. 2) ***Rich Annotations***: Each character is accompanied by a rich profile, highlighting aspects such as personality, voice style, relationships, and past experiences. Further, each dialogue turn is annotated with corresponding speech responses for both users and characters via TTS models (Du et al., 2024; Kim et al., 2021). 3) ***Dynamic Curation***: The dialogue data is generated through interactive conversations between two chatbots, guided by the predefined character profiles. The main contributions are as follows:

- We introduce ***OmniCharacter***, a pioneering step in realizing speech-language collaborative RPAs.
- We construct ***OmniCharacter-10K***, a multi-turn dialogue dataset with detailed role profiles, conversion data, and high-quality audio annotations for advancing RPAs development.
- Experimental results demonstrate that our method achieves superior performance on the role-playing benchmark, *i.e.*, CharacterEval, as well as providing robust results on general speech benchmarks, *i.e.*, LibriSpeech, AISHELL-2, and LibriTTS.

2 Methodology

In this section, we present an in-depth presentation of OmniCharacter, the first RPAs that perceive both speech-language understanding while generating character-specific audio-text responses with low latency. The framework is illustrated in Figure 2.

2.1 Overview

The traditional RPAs are designed to make LLMs interact with users or other agents by emulating specific characters. To achieve this, RPAs utilize a character profile denoted as P, along with the ongoing dialogue contexts $C_n = [q_1, r_1, ..., q_n]$ to generate a response r_n consistently with character sets:

$$r_n = \mathtt{RPAs}(C_n, P), \tag{1}$$

where q_i represent the i-th input query and r_i indicates the corresponding response. However, this paradigm relies on LLMs' behavior cloning ability and focuses mainly on textual form, neglecting essential vocal traits like voice style, tone, and emotions of different characters.

To bridge this gap, we introduce OmniCharacter, the first model considering the role-related vocal persona by collaborating speech and language for immersive interaction. Specifically, OmniCharacter mainly contains two components: (1) a Speech-Language Collaborative model to align semantics by processing both language inputs (role profile, dialogue contexts, and text query) and speech input, (2) a Role Speech Decoder that aims to generate role-specific speech responses that reflect their unique voice traits. Thereby, the dialogues contexts C_n can be redefined as $U_n = [X_1, Y_1, \ldots, X_n]$, where $X_i \in \{X_i^S, X_i^T, X_i^{S+T}\}$ represents the i-th input query, supporting three types: audio-only (X_i^S), text-only (X_i^T), or text-audio (X_i^{S+T}), resulting in flexible interaction during dialogue. Similarly, $Y_n \in \{Y_n^S, Y_n^T\}$ indicates the response, owning of two types, *i.e.*, audio response (Y_n^S) and

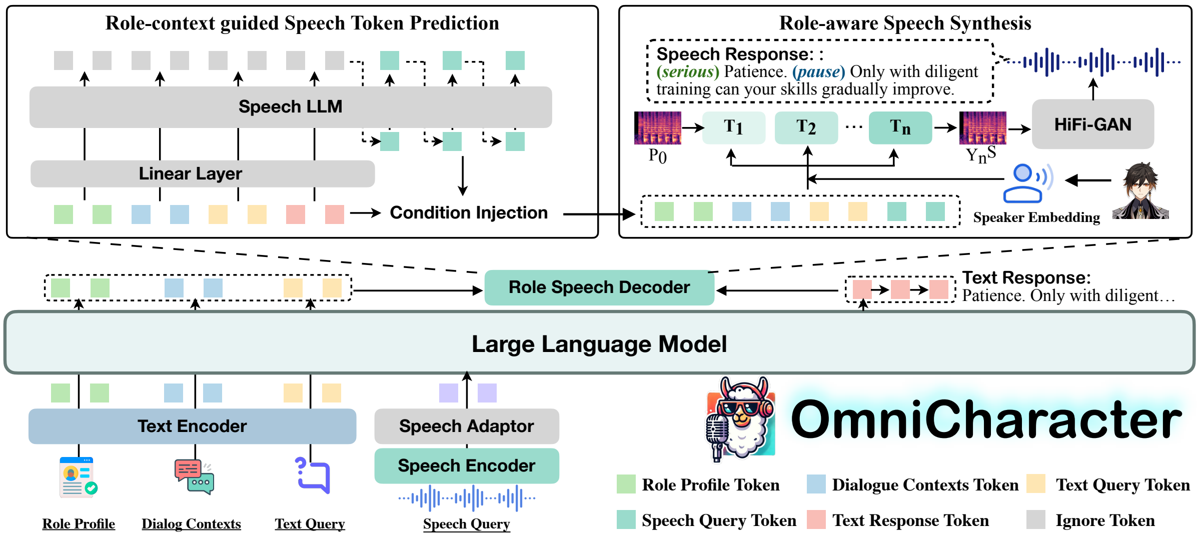


Figure 2: **The overview of our OmniCharacter framework**. We first build a speech-language collaborative model, a large language model that receives both speech and language inputs for unified modeling. Furthermore, we propose a role speech decoder to synthesize speech responses containing vocal traits of different characters by devising two innovative modules: i) Role-context Guided Speech Token Prediction, which aims to enhance the generation of speech tokens by leveraging the textual representations of the base model as a contextual prior. iii) Role-aware Speech Synthesis, which generates speech responses from speech tokens maintaining unique character personality and voice traits based on several conditions.

text response (Y^T_n), which maintains the consistent and distinct personality of characters. Next, we elaborate on the details of two components in the subsequent subsections.

2.2 Speech-Language Collaborative Model

To fully align speech and language semantics, we first devise a unified speech-language collaborative model. Specifically, it comprises a speech encoder, a speech adaptor, and an LLM.

Speech Encoder. To tackle user or character speech input, we adopt a speech encoder to encode the speech input X^S_n of the n-th dialogue turn to speech sequence $M^S_n = [M^S_{n,1}, ..., M^S_{n,N_f}]$, where N_f indicate the number of audio frames.

Speech Adaptor. To address the high temporal redundancy of the encoded speech sequence above, a down-sampling strategy is further applied by grouping every k consecutive frame into compact sequence $Z^S_n = [Z^S_{n,1}, ..., Z^S_{n,\lfloor N_s/k \rfloor}]$, where

$$Z^S_{n,i} = M^S_{k*i} \oplus M^S_{k*i+1} \oplus ... \oplus M^S_{k*i+k-1}. \quad (2)$$

This sequence is then encoded by a speech adapter τ to make it compatible with the embedding space of LLM:

$$E^S_n = \tau(Z^S_n). \quad (3)$$

Large Language Model. We utilize Qwen2.5-7B-Instruct (Yang et al., 2024) as the LLM, a state-of-the-art large language model renowned for its advanced reasoning capabilities to align text and audio sequences for unified modeling. In particular, we process the text inputs, including the role profile P, dialogue contexts U_n, and the text query X^T_n to form an overall sequence $E^T_n = \{E^T_{n,i}\}^{N_t}_{i=1}$ via a text encoder. This sequence is then concatenated with the speech sequence $E^S_n = \{E^S_{n,i}\}^{N_s}_{i=1}$ and sent into LLM. The N_t and N_s indicate the length of text and speech sequences respectively. The prompt template for organizing the overall inputs is showcased in the Appendix A.2. Finally, the LLM is trained via an auto-regressive manner:

$$\mathcal{L}_{language} = -\sum_{i=1}^{N_t} \log P(O^T_i \mid O^T_{<i}). \quad (4)$$

2.3 Role Speech Decoder

Role-context Guided Speech Token Prediction. Intuitively, it is straightforward to use the LLM to predict both text tokens O^T and speech tokens O^S. However, we find this strategy makes the model difficult to train, causing duplicate speech token outputs and hallucinations that are inconsistencies with the semantics of text tokens. To facilitate stable and accurate speech token prediction, we propose an innovative module known as role-context guided speech token prediction built upon the speech-language collaborative model. Specifically, it employs a lightweight language model (denoted as SpeechLLM) that effectively utilizes the context representations $H = [H_1, ..., H_{N_s+N_t}]$ output from LLM to generate speech tokens:

$$O^S = \text{SpeechLLM}(O^S_m | O^S_{1:m-1}, \phi(H)), \qquad (5)$$

where ϕ is a linear projection layer to map the context representations into the SpeechLLM space. We find that this design not only stabilized the training but also ensured that the generated speech tokens are highly aligned with the contextual semantics, effectively preventing the speech token repetitive output problem. Note that the SpeechLLM also follows an auto-regressive training objective:

$$\mathcal{L}_{speech} = -\sum_{i=1}^{N_s} \log P(O^S_i \mid O^S_{<i}). \qquad (6)$$

Role-aware Speech Synthesis. A high-quality speech response that reflects the character's voice traits and persona is important for immersive RPAs. To achieve this, we propose a role-aware speech synthesis module, which generates the audio response containing voice traits of characters from the speech tokens. Following (Zeng et al., 2024), we first decode the speech tokens into the Mel spectrogram and then synthesize the waveform with the generated Mel spectrogram as input. To achieve this, we adopt a conditional flow matching (CFM) model to sample the Mel spectrogram specified by the context representations E, speech tokens O^S, and speaker embedding v (Wang et al., 2023b) extracted from character voice as conditions. The sampling process can be defined by a time-dependent vector field $T = \{T_1, ..., T_n\}$. To improve efficiency, the optimal-transport (OT) flow is used. We summarize the above process as:

$$Y^S_n = \text{OT-CFM}(p_0 | H, v, O^S), \qquad (7)$$

where p_0 is the initial Mel spectrogram. Ultimately, after obtaining the generated Mel spectrogram, a HiFi-GAN vocoder serves to synthesize a continuous waveform from the Mel spectrogram, facilitating the conversion of spectral representations into high-fidelity audio signals Y^S_n containing characters' voice traits such as voice style and emotions.

2.4 Training Strategy

Following (Fang et al., 2024), we adopt a two-stage training strategy for OmniCharacter. In the first stage, we train the speech adapter and LLM to generate text responses based on the text and speech inputs using the objective $\mathcal{L}_{language}$ in Equation (4). In the second stage, we only train the linear layer and speech LLM in role-context guided speech token prediction module for speech token prediction by using the objective $\mathcal{L}_{speech}$ in Equation (6). For role-aware speech synthesis, we utilize the pre-trained conditional flow matching and HiFi-GAN weights from GLM-4-Voice (Zeng et al., 2024) and fine-tune them on high-quality role speech data[2].

3 The OmniCharacter-10K Dataset

To facilitate the model with speech-language scenarios, we construct OmniCharacter-10K, a multi-turn dialogue dataset with distinctive characters with rich and expressive text and speech annotations. In this section, we describe the data collection, annotation, and verification of OmniCharacter-10K. We also introduce the statistics and distribution of it. The illustration of OmniCharacter-10K is showcased in Figure 3.

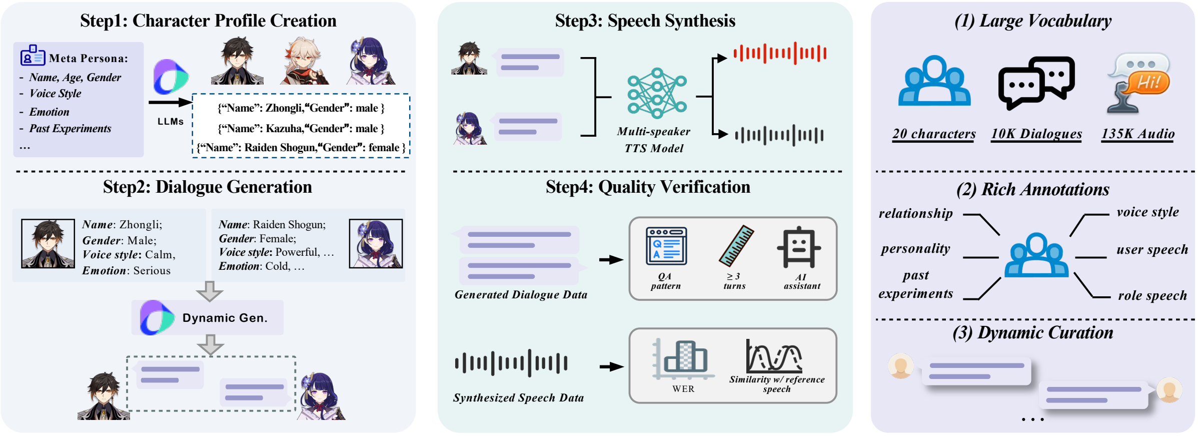
3.1 Data Collection, Annotation, and Verification

Step-1: Character Profile Creation. To make sure the collected characters have distinct traits as well as contain high-quality audio, we select from games since (1) the availability of rich information (*e.g.,* personality, background, and vocal tone), and (2) clear, noise-free character audio recorded by professional voice actors. We end up with 20 foundational characters (10 Chinese and 10 English) from Genshin Impact to construct our dataset. Subsequently, we use Doubao [3] to summarize the meta information for each character, and then order the model to expand the information to ensure the diversity and complexity. At last, all character profiles are subjected to rigorous human checks to ensure accuracy and reliability.

Step-2: Dialogue Generation. After selecting the characters and character profiles, we proceed to generate multi-turn dialogues. Specifically, we deploy two models (*i.e.*, `Doubao-pro-32k`), where one model assumes the role of the character and the other as the user or another character. Each model is provided with its respective profile and instructed to engage in multi-turn dialogues, ensuring the conversations are aligned with the predefined

[2] https://genshin.hoyoverse.com/en/

[3] https://www.volcengine.com/product/doubao/



(a) Dataset Construction Pipeline

(b) Dataset Properties

Figure 3: **Illustration of OmniCharacter-10K.** (a) Dataset Construction Pipeline, which consists of four steps: (1) Characters Profile Creation, (2) Dialogue Generation, (3) Speech Synthesis, and (4) Quality Verification. (b) Dataset Properties, which has three appealing properties of our dataset: (1) Large Vocabulary, the dataset includes 20 characters, 10K multi-turn dialogues, and 135K audio responses, (2) Rich Annotations, each character is equipped with rich text and speech annotations, and (3) Dynamic Curation, we generate dialogue data through chatbot interactions based on predefined character profiles.

Splits	# Characters	Avg. Turns/Conv.	# Samples	# Speech Hours. (user/character)
Training	20	15	10, 068	353.96 (183.5/170.46)
Test	20	15	425	7.08 (3.56/3.52)

Table 1: The statistic of OmniCharacter-10K dataset.

traits and backgrounds of the characters.

Step-3: Speech Synthesis. The generated multi-turn dialogues consist of open-domain text without corresponding audio annotations. To handle this, we first collected 40K high-quality audio samples for 20 characters, which are professionally recorded by voice actors. Then, we utilize these audios as seed data and train a multi-speaker text-to-speech (TTS) model, *i.e.*, VITS (Kim et al., 2021) for the audio synthesis of open-text in dialogue. For the user side, we use the advanced TTS model CosyVoice (Du et al., 2024) to synthesize audio. Each dialogue sample has a 50% ratio of being assigned a generic male or female voice, ensuring its balance and diverse distribution.

Step-4: Quality Verification. To ensure the quality of the constructed dataset, we manually filter samples based on several criteria. For text data, we performed data filtering by retaining the samples following ABAB (user-role or role1-role2) pattern. Besides, we preserved dialogues longer than three turns, filtering out shorter ones. Furthermore, we remove data that mimics the style of general AI assistance such as '*I am a helpful AI assistant...*', as well as unnecessary explanatory prefaces and postfaces. For speech data, we first use the Whisper-large-v3 (Radford et al., 2023) to convert the synthesized audio into text and compute the WER metrics with the corresponding text, where the WER greater than 10 are rejected. Then, we use WavLLM (Hu et al., 2024) to compute the similarity between the synthesized audio and reference audio, where the similarity less than 0.8 threshold is also rejected.

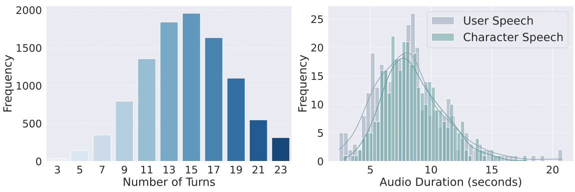


Figure 4: **Left:** Distribution of dialogue turns across samples in the OmniCharacter-10K dataset. **Right:** Distribution of audio duration for user speech and character speech respectively.

3.2 Statistics of OmniCharacter-10K

Table 1 presents the statistics of the OmniCharacter-10K dataset, which includes a total of 20 characters with 10, 068 training samples and 425 test samples. The training and test sets are rich in speech annotations, with 353.96 hours and 7.08 hours of audio data, respectively. In addition, Figure 4 shows the distribution of dialogue turns (left) and audio duration (right) across the dataset. On one hand, more than 80% of the dialogues are longer than 10 turns, highlighting the dataset's focus on multi-turn interactions. On the other hand, the audio

Model	Character Consistency						Conversational Ability				Role-playing Attractiveness				
	KE	KA	KH	PB	PU	**Avg.**	Flu.	Coh.	Cons.	**Avg.**	HL	CS	ED	Emp.	**Avg.**
Proprietary Models															
BC-NPC-Turbo	1.802	2.964	2.993	2.910	3.151	2.764	3.578	3.898	3.916	3.798	3.836	2.643	2.336	2.971	2.946
MiniMax	1.835	2.910	2.944	2.774	3.125	2.718	3.609	3.932	3.811	3.784	3.768	2.672	2.150	3.017	2.902
GPT-3.5	1.716	2.339	2.212	1.921	2.316	2.101	2.629	2.917	2.700	2.749	2.565	2.422	1.660	2.526	2.293
GPT-4	2.250	2.855	2.785	2.721	2.873	2.697	3.332	3.669	3.343	3.448	3.143	3.184	2.153	3.010	2.873
Open-sourced Models															
ChatGLM3-6B	2.016	2.792	2.704	2.455	2.812	2.556	3.269	3.647	3.283	3.399	3.064	2.932	1.969	2.993	2.739
Baichuan2-7B	1.813	2.849	2.929	2.830	3.081	2.700	3.551	3.894	3.827	3.757	3.670	2.728	2.115	2.984	2.874
Baichuan2-13B	1.802	2.869	2.946	2.808	3.081	2.701	3.596	3.924	3.864	3.759	3.700	2.703	2.136	3.021	2.890
InternLM-7B	1.782	2.800	2.781	2.719	3.016	2.620	3.527	3.823	3.744	3.698	3.546	2.622	2.070	2.897	2.784
InternLM-20B	1.945	2.916	2.920	2.753	3.041	2.715	3.576	3.943	3.717	3.745	3.582	2.885	2.132	3.047	2.911
CharacterGLM	1.640	2.819	2.738	2.301	2.969	2.493	3.414	3.717	3.737	3.623	3.738	2.265	1.966	2.812	2.695
Qwen-7B	1.956	2.728	2.633	2.605	2.780	2.540	3.187	3.564	3.229	3.327	3.036	2.791	2.052	2.838	2.679
Qwen-14B	1.988	2.800	2.811	2.744	2.900	2.649	3.351	3.765	3.510	3.542	3.354	2.871	2.237	2.970	2.858
Qwen2-7B-Instruct	1.785	2.649	2.489	1.978	2.577	2.296	2.995	3.360	3.048	3.014	3.223	2.262	1.622	2.679	2.446
OmniCharacter	2.322	3.021	2.879	3.664	2.956	2.968	3.342	3.713	3.384	3.480	3.225	3.317	2.997	3.094	3.158

Table 2: **Performance comparison with state-of-the-art methods on *CharacterEval*.** We evaluate the models across three dimensions including: (A) **Character Consistency**: (a-1) KE: Knowledge-Exposure, (a-2) KA: Knowledge-Accuracy, (a-3) KH: Knowledge-Hallucination, (a-4) PB: Persona-Behavior, (a-5) PU: Persona-Utterance. (B) **Conversational Ability**: (b-1) Flu.: Fluency, (b-2) Coh.: Coherency, (b-3) Cons.: Consistency. (C) **Role-playing Attractiveness**: (c-1) HL: Human-Likeness, (c-2) CS: Communication Skill, (c-3) ED: Expression Diversity, (c-4) Emp.: Empathy.

duration statistics further reveal that the speech in the dataset is sufficiently long to enable immersive dialogue experiences. These statistics emphasize the dataset's capability to support the evaluation of RPAs in supporting immersive dialogues.

4 Experiments

4.1 Experimental Setups

Model Configuration. We employ the Whisper-large-v3 as the speech encoder and Qwen-2.5-7B-Instruct as the LLM in speech-language collaborative model. In role-context guided speech token prediction, we employ the lightweight Qwen2.5-0.5B-Instruct (Team, 2024) model as SpeechLLM to generate speech tokens. To facilitate this process, a speech tokenizer derived from GLM-4-Voice (Zeng et al., 2024) with 16K vocabulary size is utilized to extract discrete speech tokens. We resample all speech data to a frequency of 16Khz. The speech-language collaborative model and role-context guided speech token prediction are initialized using pre-trained parameters from OpenOmni (Luo et al., 2025), and we conduct fine-tuning on the OmniCharacter-10K train set. The weights of role-aware speech synthesis are initialized from pretrained checkpoint of GLM-4-Voice.

Training Details. The OmniCharacter employs a two-stage training strategy as described in Section 2.4. In the first stage, we utilize the AdamW optimizer with a warmup ratio of 0.3, gradually increasing the learning rate until it reaches 5e-4. The second stage retains the parameters established in the first stage, and we only adjusted the final learning rate to 5e-5. We set the batch size as 32 for both two stages. The entire training process is finished within 3 epochs on 8×A100 GPUs.

Datasets. The evaluation leverages the following datasets: CharacterEval (Tu et al., 2024) is used for basic language understanding to assess the text comprehensive ability of RPAs. OmniCharacter-10K test set is employed for character-aware speech-language evaluation to test speech generation and alignment abilities. LibriSpeech (Panayotov et al., 2015), AISHELL-2 (Du et al., 2018), and LibriTTS (Zen et al., 2019) are general speech datasets for model's generalization test.

4.2 Performance Comparison

Basic Language Understanding. To evaluate the effectiveness of our OmniCharacter in basic language understanding, we compare our method with state-of-the-art RPAs on CharacterEval (Tu et al., 2024) benchmark as shown in Table 2. Note that the characters tested in CharacterEval differ from those in our training data. The results illustrate that our model outperforms the proprietary and open-source models on most metrics. Particularly, compared with models specifically designed for role-playing task including BC-NPC-Turbo and MiniMax, our method achieves considerate im-

Model	S2TIF		S2SIF		Alignment	
	Content ↑	Style ↑	Content ↑	Style ↑	WER ↓	CER ↓
SpeechGPT	2.07	1.81	1.75	1.72	57.90	47.01
LLaMA-Omni	3.79	2.67	2.43	1.79	48.9	49.2
OmniCharacter	4.15	3.28	3.76	2.16	12.66	11.57

Model	Flu.	Cons.	Emo.	Cla.	App.	Imm.
SpeechGPT	6.12	3.86	3.75	6.89	4.23	3.64
LLaMA-Omni	6.88	4.27	3.44	6.69	4.78	4.68
OmniCharacter	7.97	6.84	6.23	7.88	5.63	8.52

Table 3: **Performance comparison with state-of-the-art methods on OmniCharacter-10K test split. Top:** ChatGPT scores for S2TIF, S2SIF tasks, and alignment scores between speech and text responses. **Bottom:** Human score (averaged from 5 experts) across six voice-related dimensions: fluency (Flu.), consistency (Cons.), emotional expression (Emo.), clarity (Cla), appropriateness (App.), and immersion (Imm.).

provements. Moreover, OmniCharacter surpasses open-sourced models with larger scales such as InterLM-20B and Qwen-14B, achieving superior performance in both character consistency and role-playing attractiveness. However, our model fails to bring clear performance gains in conversational ability. Possible reason is that other methods have been pre-trained on more large-scale dialogue data, providing strong conversational capability.

Character-aware Speech-Language Evaluation. To thoroughly evaluate the speech-language collaboration ability of OmniCharacter for immersive and personalized role-playing interaction, we conduct evaluation from two perspectives: metrics-based and human-based evaluations.

The metrics-based evaluation consists of three tasks (Fang et al., 2024): (1) speech-to-text instruction-following (S2TIF): whether the model's text response correctly answers the input speech query; (2) speech-to-speech instruction-following (S2SIF): whether the model's speech response correctly answers the input speech query; (3) speech-text alignment (Alignment): alignment between text and speech responses. For S2TIF and S2SIF, we ask GPT-4o to score the response in a range from 1 to 5 from content and style aspects. For Alignment, we transcribe the speech response into text and calculate word error rate (WER) and character error rate (CER). As shown in Table 3 (Top), our model consistently enhances performance across all metrics, achieving superior results compared to SpeechGPT and LLaMA-Omni. These results confirm that OmniCharacter is capable of effectively handling user or characters' instructions, while simultaneously generating consistent text and speech responses.

Model	LibriSpeech (ASR-WER)		AISHELL-2 (ASR-CER)	LibriTTS (TTS-WER)
	test-clean	test-other	test	test-clean
CosyVoice 1.0	-	-	-	3.17
Whisper-large-v3	2.50	4.53	-	-
Qwen2-Audio	2.00	4.50	3.30	-
OmniCharacter	3.26	4.23	6.62	7.23

Table 4: **Performance comparison with the state-of-art methods on general speech benchmarks** including LibriSpeech, AISHELL-2, and LibriTTS. The model is evaluated on two tasks, *i.e.*, Automatic Speech Recognition (ASR) and Text-to-Speech (TTS). WER: Word Error Rate. CER: Character Error Rate.

Model	Character Consistency	Conversational Ability	Role-playing Attractiveness
OmniCharacter *w/o* audio	2.948	3.475	3.148
OmniCharacter	2.956	3.480	3.158

Table 5: Ablation study of model training with or without audio modality. For simplicity, we only report the average score across three dimensions on CharacterEval.

In terms of human-based evaluation, five experts are assigned to examine the model's outputs across six voice-related dimensions as shown in Table 3 (bottom). Each expert is asked to score the responses on a scale from 1 to 10 for each dimension. From the table, we figure out that our method reaches the best performance across all dimensions, proving the superiority of our model which is capable of providing immersive interactions for users.

Generalization on General Speech Benchmarks. To better prove the robust ability of OmniCharacter, we evaluated the performance of our model on three widely used speech datasets, LibriSpeech, AIShell-2, and LibriTTS, focusing on both Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) tasks. As presented in Table 4, our model achieves comparable performance compared with previous models, demonstrating its generalization ability. These results further highlight the robustness of OmniCharacter in generating coherent and accurate outputs, showcasing its potential in real-world speech applications.

4.3 Impact of Audio Modality

To investigate the advancement of audio modality for RPAs, we conduct ablation by excluding audio from our model and evaluate on CharacterEval in Table 5. Note that we report average score for clarity. As observed, equipped with audio modality improves the performance in character consistency and conversational ability, and yields comparable results in role-playing attractiveness. This highlights the significant contribution of audio modality in enhancing basic language understanding.

Method	GP-4o	LLaMA-Omni	OmniCharacter
Latency (ms)	320	226	289

Table 6: Speech response latency of different models.

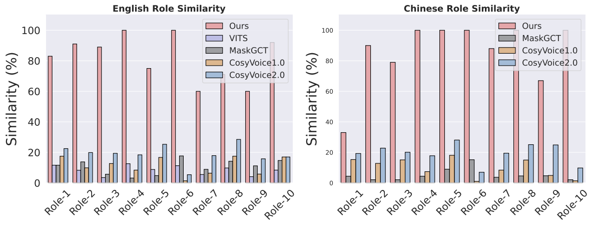


Figure 5: Comparison of character voice similarity on OmniCharacter-10K test set. Our model generates audio responses that better match the timbre of characters.

4.4 Similarity of Synthesized Voice for Characters

To verify the model's capability of synthesizing character-specific speech containing their voice traits, we use WavLLM to compute the cosine similarity between the reference speech and the generated speech on OmniCharacter-10K test set, as shown in Figure 5. The comparison methods include state-of-the-art TTS systems, *i.e.*, VITS, MaskGCT, CosyVoice1.0, and CosyVoice2.0. For our model, we directly input the corresponding text to generate the speech output. For other models, we leverage their voice cloning capabilities by providing reference speech, along with the given text, to synthesize the corresponding speech. If the cosine similarity between the output speech and the reference speech is greater than 0.8, we classify them as belonging to the same character. As observed, our model consistently achieves higher cosine similarity for both Chinese and English characters' voices. These results highlight the effectiveness of our approach in preserving the voice traits of characters, achieving a more interaction.

4.5 Learning Discriminative Character Speech Embeddings

The OmniCharacter generates the character-specific voice by utilizing the speech embedding as one of the conditions. To evaluate the discrimination of speaker embedding between different characters, we compute their cosine similarity as shown in Figure 6. Specifically, we randomly sampled three sets of data of 20 characters for experimental comprehensiveness. We also included the user's speaker embedding for comparison. The results indicate that the similarity values between the speech embeddings of different characters and users are very low, demonstrating their strong discriminability. This confirms the robustness of our method in controlling character's voice traits.

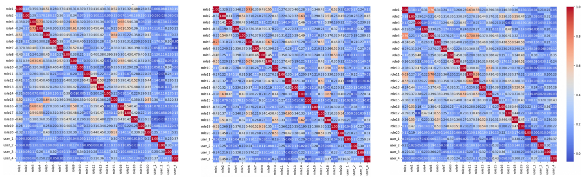


Figure 6: Speech embedding discriminability for 20 characters and 4 users. We randomly sampled three sets of data for experimental comprehensiveness. The low similarity values demonstrate strong separation of character voices, enabling precise voice control.

4.6 Speech Response Latency

To demonstrate the seamless interaction capability of the proposed model, we evaluate the speech response latency of different models, as shown in Table 6. From the table, we observe that our model outperforms the GPT-4o, achieving a lower response latency. It is observed that the latency of our method is higher than LLaMA-Omni. The possible reason is that LLaMA-Omni only supports general female voice, while we introduce additional modules considering voice traits of different roles, thus the increased latency is acceptable. Overall, these results underscore the seamless dialogue ability of our method, demonstrating its effectiveness in real-time speech-language interaction for RPAs.

5 Conclusion

In this paper, we present OmniCharacter, the first step toward creating a seamless and immersive RPAs model that integrates both speech and language personality interaction. Unlike existing methods that primarily focus on text-based dialogues, our approach emphasizes the importance of character voice traits, *e.g.,* voice style and emotions, which enable agents to reflect both personality and vocal traits throughout interactions, delivering a harmonious blend of speech and language responses. To facilitate the speech-language scenarios for RPAs, we construct the OmniCharacter-10K dataset, which includes distinct characters, contextually rich multi-turn dialogues, and dynamic speech responses. Experimental results show that OmniCharacter surpasses current RPAs and general speech-language models in both content and style, achieving a seamless and immersive experience.

6 Limitations

The proposed OmniCharacter advances immersive role-playing agents (RPAs) by considering characters' voice traits via a speech-language personality interaction. Despite effectiveness, several limitations remain. First, its performance depends on the quality of the pre-trained LLMs and speech synthesis model, which may be influenced by their alignment and training data quality. Second, the current implementation is limited to dialogues with two characters, posing challenges for scaling to multi-role conversations. Third, while the OmniCharacter-10K dataset offers diverse scenarios, it may not fully capture the variety of real-world dialogues. Future work will expand the dataset to improve applicability. Lastly, while latency is low, further optimization is needed for real-time interactions.

References

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. 2023. Audiolm: a language modeling approach to audio generation. *Preprint*, arXiv:2209.03143.

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*.

Jiayu Du, Xingyu Na, Xuechen Liu, and Hui Bu. 2018. Aishell-2: Transforming mandarin asr research into industrial scale. *arXiv preprint arXiv:1808.10583*.

Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, et al. 2024. Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *arXiv preprint arXiv:2407.05407*.

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*.

Shujie Hu, Long Zhou, Shujie Liu, Sanyuan Chen, Lingwei Meng, Hongkun Hao, Jing Pan, Xunying Liu, Jinyu Li, Sunit Sivasankaran, et al. 2024. Wavllm: Towards robust and adaptive speech large language model. *arXiv preprint arXiv:2404.00656*.

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

Run Luo, Ting-En Lin, Haonan Zhang, Yuchuan Wu, Xiong Liu, Min Yang, Yongbin Li, Longze Chen, Jiaming Li, Lei Zhang, et al. 2025. Openomni: Large language models pivot zero-shot omnimodal alignment across language with real-time self-aware emotional speech synthesis. *arXiv preprint arXiv:2501.04561*.

Openai OpenAI. 2022. Openai: Introducing chatgpt. *URL https://openai. com/blog/chatgpt*.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22.

Letian Peng and Jingbo Shang. 2024. Quantifying and optimizing global faithfulness in persona-driven role-playing. *arXiv preprint arXiv:2405.07726*.

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

Paul K Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, et al. 2023. Audiopalm: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925*.

Nafis Sadeq, Zhouhang Xie, Byungkyu Kang, Prarit Lamba, Xiang Gao, and Julian McAuley. 2024. Mitigating hallucination in fictional character role-play. *arXiv preprint arXiv:2406.17260*.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. Character-llm: A trainable agent for role-playing. In *EMNLP*, pages 13153–13187.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Quan Tu, Shilong Fan, Zihang Tian, and Rui Yan. 2024. Charactereval: A chinese benchmark for role-playing conversational agent evaluation. *arXiv preprint arXiv:2401.01275*.

Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al. 2023a. Neural codec language models are zero-shot text to speech synthesizers. *arXiv preprint arXiv:2301.02111*.

Hui Wang, Siqi Zheng, Yafeng Chen, Luyao Cheng, and Qian Chen. 2023b. Cam++: A fast and efficient network for speaker verification using context-aware masking.

Lei Wang, Jianxun Lian, Yi Huang, Yanqi Dai, Haoxuan Li, Xu Chen, Xing Xie, and Ji-Rong Wen. 2024a. Characterbox: Evaluating the role-playing capabilities of llms in text-based virtual worlds. *arXiv preprint arXiv:2412.05631*.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. 2024b. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345.

Tianrui Wang, Long Zhou, Ziqiang Zhang, Yu Wu, Shujie Liu, Yashesh Gaur, Zhuo Chen, Jinyu Li, and Furu Wei. 2024c. Viola: Conditional language models for speech recognition, synthesis, and translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Xintao Wang, Yunze Xiao, Jen-tse Huang, Siyu Yuan, Rui Xu, Haoran Guo, Quan Tu, Yaying Fei, Ziang Leng, Wei Wang, et al. 2024d. Incharacter: Evaluating personality fidelity in role-playing agents through psychological interviews. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1840–1873.

Zekun Moore Wang, Zhongyuan Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, et al. 2023c. Rolellm: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. *arXiv preprint arXiv:2310.00746*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Tao Yang, Yuhua Zhu, Xiaojun Quan, Cong Liu, and Qifan Wang. 2025. Psyplay: Personality-infused role-playing conversational agents. *arXiv preprint arXiv:2502.03821*.

Yeyong Yu, Runsheng Yu, Haojie Wei, Zhanqiu Zhang, and Quan Qian. 2024. Beyond dialogue: A profile-dialogue alignment framework towards general role-playing language model. *arXiv preprint arXiv:2408.10903*.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. 2019. Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*.

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*.

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*.

Dong Zhang, Xin Zhang, Jun Zhan, Shimin Li, Yaqian Zhou, and Xipeng Qiu. 2024a. Speechgpt-gen: Scaling chain-of-information speech generation. *arXiv preprint arXiv:2401.13527*.

Qinglin Zhang, Luyao Cheng, Chong Deng, Qian Chen, Wen Wang, Siqi Zheng, Jiaqing Liu, Hai Yu, Chaohong Tan, Zhihao Du, et al. 2024b. Omniflatten: An end-to-end gpt model for seamless voice conversation. *arXiv preprint arXiv:2410.17799*.

Jinfeng Zhou, Zhuang Chen, Dazhen Wan, Bosi Wen, Yi Song, Jifan Yu, Yongkang Huang, Libiao Peng, Jiaming Yang, Xiyao Xiao, et al. 2023. Characterglm: Customizing chinese conversational ai characters with large language models. *arXiv preprint arXiv:2311.16832*.

A Appendix

A.1 Related Works

Role-Playing Agents. Benefiting from the powerful summarization and imitation capabilities of large language models (LLMs), role-playing agents (RPAs) have made significant progress in recent years. Prior research primarily focuses on replicating the knowledge, experiences, and intent of characters (Yang et al., 2025; Sadeq et al., 2024; Yu et al., 2024; Shao et al., 2023; Zhou et al., 2023). For instance, (Sadeq et al., 2024) reduces hallucination by adjusting parametric knowledge influence based on a pre-calibrated confidence threshold. (Shao et al., 2023) seeks to train an agent with profile and experience perception, replacing limited prompts for LLM instruction. Due to the lack of comprehensive benchmarks, later works focused on building character-specific datasets and new evaluation metrics. (Peng and Shang, 2024; Wang et al., 2024d,a, 2023c). (Wang et al., 2024d) aims to evaluate the personality fidelity of RPAs using psychological scales. (Wang et al., 2024a) devises a simulation sandbox to generate fine-grained character behavior trajectories to evaluate RPAs capabilities. However, most existing studies primarily focus on replicating dialogues in textual form, often overlooking the role's essential voice traits, such as tone, style, emotions, and pauses. To bridge this gap, we propose OmniCharacter, a model that seamlessly combines speech and language to ensure immersive RPAs interactions.

Speech-Language Models. Recent advancements in speech-language models have significantly improved human-machine interactions (Fang et al., 2024; Zhang et al., 2023; Zeng et al., 2024; Zhang et al., 2024a). With the advancement of language models in the natural language processing field, early work attempted to enable language models to generate both speech tokens and text tokens via a decoder-only architectural (Wang et al., 2024c; Borsos et al., 2023; Wang et al., 2023a). With the development of large language models (LLMs), more recent studies have integrated speech capabilities into LLMs by either adding speech tokens to the text vocabulary or incorporating a speech encoder before the LLMs such as SpeechGPT (Zhang et al., 2024a) and AudioPaLM (Rubenstein et al., 2023). Beyond that, some methods fine-tune LLMs for speech understanding such as LLaMA-Omni (Fang et al., 2024) and GLM-4-Voice (Zeng et al., 2024), enabling it to perform general speech tasks with low response latency. Additionally, models like Moshi (Défossez et al., 2024) and Omni-Flatten (Zhang et al., 2024b) handle real-time interactions by managing simultaneous speech and text streams. Despite progress, most existing speech-language models explore general audio-language capabilities. Compared with the above methods, we focus on modeling the unique voice traits of different roles in RPA tasks for a more immersive human-machine interaction.

A.2 Prompt

Due to space limitations, we present the complete prompt template for input data of the speech-language collaborative model in the appendix, as shown in Figure 7. Specifically, we use the role profile as a system prompt to enable the LLM to mimic the character. Then, we feed the model the dialogue contexts. The model is required to generate a reasonable text-audio response based on the current text or audio queries.

[System Prompt]

You are Zhongli, a character from the game “Genshin Impact". You are the former God of Contracts and now live as a mortal in Liyue.

Personality Traits: Calm, composed, wise, and responsible.

Voice Style: You speak in a slow and deliberate manner, often using formal language and quoting proverbs or ancient sayings. Your tone is gentle and soothing, yet authoritative.

Catchphrase: "As it should be".

Relationships: Hu Tao is your friend; Ningguang respects you; Childe is your acquaintance; Xiao is your old friend.

Past Experiences:

(1) As the God of Contracts, you ruled over Liyue for thousands of years, ensuring the prosperity and stability of the region.

(2) You decided to step down from your position as a god and hand over the reins of power to mortals.

(3) You now work as a consultant at the Wangsheng Funeral Parlor, providing guidance and advice to others.

(4) You have witnessed the rise and fall of many civilizations and have a deep understanding of the nature of humanity.

Emotions: Serious and Patient.

<Dialogue Contexts>

<Text Query> or <Speech Query>

PLEASE ANSWER THE QUESTIONS IN THE USER/CHARACTERS’ INPUT TEXT OR SPEECH.

Figure 7: The prompt template used to organize our input data including role profile, dialogue contexts, text input, and speech input to train our speech-language collaborative model.