Inno-Sama

Nikita Kurkulskiu
Innopolis University
Innopolis, Russia
n.kurkulskiu@innopolis.university

***Abstract*—This project aims to develop a neuro avatar that combines speech recognition, emotional analysis, and real-time animation driven by Large Language Models (LLMs). The system will interpret vocal inputs, generate natural responses, and reflect emotional states through synchronized body language, offering a highly immersive user experience. Inspired by AI-driven VTubers like Neuro-sama, the avatar will enable live, voice-based conversations with real-time context awareness. By integrating advanced voice processing and dynamic character animation, the project seeks to create more empathetic, spontaneous, and personalized digital interactions, setting a new standard for communication with virtual entities.**

I. Introduction

This project focuses on creating an advanced neuro avatar using Large Language Model (LLM) technology. By interpreting vocal input and analyzing emotional cues, the system will deliver engaging, context-aware responses while exhibiting animated motions synchronized to the conversation's mood. Our main objective is to merge speech recognition, dynamic character animation, and robust language modeling in one cohesive platform. Unlike standard chatbots, this avatar will interpret user speech, respond naturally, and convey emotional cues through body language, making each interaction highly immersive. By leveraging methods similar to Neuro-sama—an AI-driven VTuber—our system can stream live, engage with viewers in real-time, and provide spontaneous yet contextually relevant replies. The planned enhancements include a speech-to-speech capability, which will allow an entirely voice-based conversation flow. Through advanced voice processing and real-time language generation, the neuro avatar will not only speak with users but also respond to their vocal inflections, enabling deeper and more empathetic engagements. This immersive approach can transform how audiences experience live streams, virtual tutoring, or interactive storytelling. By harnessing the power of LLM technology, we envision a future where communication with digital personalities becomes increasingly seamless, personalized, and emotionally resonant. Ultimately, this innovation aspires to redefine how humans interface with virtual entities.

II. Related work

Existing approaches in conversational AI range from simple rule-based chatbots to complex language models like GPT-4 or LLaMA. VTubers such as Neuro-sama demonstrate how AI can engage with audiences in real-time. However, few solutions offer the integration of voice recognition, emotional analysis, and real-time animated gestures in a single framework. Our approach expands on state-of-the-art language modeling techniques, adding streaming voice input/output capabilities and emotional cues inspired by prior studies in speech-driven avatar animation [1].

III. Methodology

To realize a fully functional neuro avatar, we focused on the following steps:

- **Speech-to-Text (STT):** We employed *vosk* for real-time speech recognition in both English and Russian, leveraging a lightweight model for minimal latency.
- **Language Model (LLM) Integration:** We launched an uncensored LLaMA 3.1 8B model and fine-tuned it using an instruction-tuning format. The dataset combined chat-like data from Neuro-sama's streams and Reddit-based conversations. LoRA was utilized to accommodate limited computational resources.
- **Text-to-Speech (TTS):** We integrated *XTTS-v2* (English) and *MeloTTS* for generating voice outputs, aiming for minimal mispronunciations while maintaining a semi-natural sound.
- **Emotion and Context Analysis:** Preliminary emotion recognition is based on pitch and volume variations of the input audio. The system maps these cues to a simplified emotional state, influencing avatar gestures in real-time.
- **Real-Time Animation:** A custom avatar pipeline synchronizes speech-based triggers (e.g., excitement, laughter) with corresponding animations or gestures, inspired by established VRM/VTuber frameworks. This allows dynamic facial expressions, head and body movements.
- **Live Streaming Setup:** The project aims to integrate with streaming platforms via tools like OBS WebSocket API for direct broadcasting of the avatar's gestures and voice output.

IV. Experiments and Evaluation

A. Experimental Setup

We trained and tested our fine-tuned model on a local Linux server. The LoRA-based fine-tuning was executed using *Unsloth* and *TRL* libraries:

```
from trl import SFTTrainer
from transformers import TrainingArguments
from unsloth import is_bfloat16_supported

trainer = SFTTrainer(
        model=model,
        tokenizer=tokenizer,
        train_dataset=custom_dataset,
        dataset_text_field="text",
        max_seq_length=max_seq_length,
        dataset_num_proc=2,
        packing=False,
        args=TrainingArguments(
                per_device_train_batch_
            size=2,
                gradient_accumulation
            _steps=4,
                warmup_steps=5,
                num_train_epochs=1,
                learning_rate=2e-4,
                fp16=not is_bfloat16
                _supported(),
                bf16=is_bfloat16
                _supported(),
                logging_steps=5,
                optim="adamw_8bit",
                weight_decay=0.01,
                lr_scheduler_type="linear",
                seed=3407,
                output_dir="outputs",
                report_to="none",
        ),
)
```

B. Datasets experiments

Initially, we experimented with a scrapped dataset from Twitch chat logs, containing raw conversations between streamers and viewers. However, this dataset presented several challenges: frequent off-topic chatter, incomplete sentences, and a generally chaotic structure. As a result, the model trained on this raw data demonstrated inconsistent behavior, often struggling to maintain context or producing fragmented, incoherent responses.

To address this, we performed additional preprocessing:

- Removed all messages shorter than three words.
- Filtered out non-English messages and excessive emoji usage.
- Grouped user queries and streamer replies into instruction-response pairs suitable for instruction-tuning.

Despite these cleaning steps, the quality improvement was marginal, likely due to the informal and noisy nature of Twitch conversations.

Additionally, we explored generating a synthetic dataset by prompting GPT-3.5 to simulate streamer-like dialogues, creating higher-quality instruction-response examples. This synthetic data preserved casual tone while maintaining better coherence. Although promising, due to resource constraints, we could not fully integrate this synthetic dataset into the final training pipeline.

Final Dataset: We used a scrapped dataset of questions and answers from the Neuro-Sama stream and combined it with the self scrapped Reddit dataset. All the data was converted into instruction-tuning format.

C. Evaluation Metrics

- **Language Model Quality:** We tracked training loss, which ranged from 0.45 to 2.58 depending on inclusion of system prompts and data format.
- **Audio Processing Latency:** Measured latency between a spoken query and avatar's spoken response, which averaged around 1.5–2 seconds.
- **User Perception:** Informal feedback suggested that emotional cues and real-time animation significantly enhanced the sense of immersion.

D. Results

The system demonstrated stable performance for both STT and TTS, though occasional transcription errors emerged with rapid or accented speech. The LLM produced mostly coherent, contextually relevant replies. Fine-tuning improved model behavior, enabling a more casual style suitable for streaming. The neural avatar effectively synchronized basic gestures with emotional states.

V. Analysis and Observations

- **Model Behavior:** Despite achieving a lower loss, the model occasionally reverted to "assistant-like" behavior. This highlights the need for further domain-specific data.
- **Computational Constraints:** Using LoRA was critical for fine-tuning on limited hardware, but it constrained potential gains from deeper hyperparameter tuning.
- **Voice Naturalness:** Our chosen TTS models provided sufficiently clear audio, though the system occasionally generated unnatural pacing or emphasized the wrong words.
- **System Integration:** The pipeline from STT to LLM response to TTS was robust, but further optimization is needed for real-time streaming scenarios with minimal delay.

VI. Conclusion

We successfully integrated STT, TTS, and an LLM-based response generator into a single neuro avatar system, showcasing real-time speech interactions and basic emotional animation cues. Although computational limitations restricted the scope of fine-tuning, the prototype performed effectively as a proof-of-concept for a fully autonomous, voice-interactive AI streamer. Future work will focus on refining the emotional detection pipeline, expanding the fine-tuned dataset for a more conversational style, and enhancing real-time avatar gestures with advanced motion capture methods. The inclusion of singing or retrieval-based voice conversion is also planned to further enhance user engagement and entertainment value.

REFERENCES

[1] Vedal, "Neuro-sama YouTube Channel," YouTube, [Online]. Available: https://www.youtube.com/@Neurosama. [Accessed: 10-Apr-2025].

[2] Vedal, "Neuro-sama Twitch Channel," Twitch, [Online]. Available: https://www.twitch.tv/vedal987. [Accessed: 10-Apr-2025].