Mixing Buddy: An AI Assistant for Gain-Balancing Mixing Advice

Pratham Vadhulas
Advisor: Dr. Alexander Lerch

Fall 2025 Project Proposal

Georgia Tech · College of Design
Center for Music Technology

Brief Introduction

overview

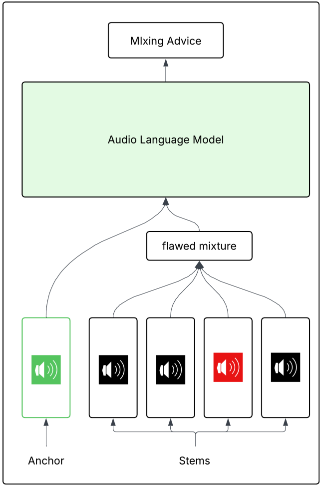
Georgia Tech · College of Design
Center for Music Technology

- **Music mixing** requires a complex, relational understanding of multiple audio tracks.
- This research investigates a framework to fine-tune an **Audio-Language Model (ALM)** to generate actionable mixing advice.
- As a starting point, we condition the model on an **"anchor track"** (e.g., bass) to teach it how to balance the levels of other instruments relative to that **stable reference point**.

Brief Introduction

overview

- **Music mixing** requires a complex, relational understanding of multiple audio tracks.
- This research investigates a framework to fine-tune an **Audio-Language Model (ALM)** to generate actionable mixing advice.
- As a starting point, we condition the model on an **“anchor track”** (e.g., bass) to teach it how to balance the levels of other instruments relative to that **stable reference point**.

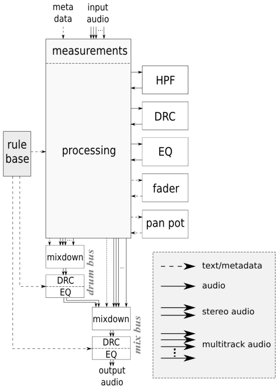


Automatic Mixing Review

Rule-Based & Deep Learning Approaches

Rule-Based and Traditional Machine Learning Systems

- Knowledge-engineered autonomous mixing [1]
- A machine-learning approach for instrument-specific application of artificial reverberation. [2]



Deep Learning Architectures

- Wave-U-Net autoencoders for automatic mixing [3]
- Differentiable mixing console with neural effects [4]

Automatic Mixing Review

Rule-Based & Deep Learning Approaches

Rule-Based and Traditional Machine Learning Systems

- Knowledge-engineered autonomous mixing [1]
- A machine-learning approach for instrument-specific application of artificial reverberation. [2]

Deep Learning Architectures

- Wave-U-Net autoencoders for automatic mixing [3]
- Differentiable mixing console with neural effects [4]

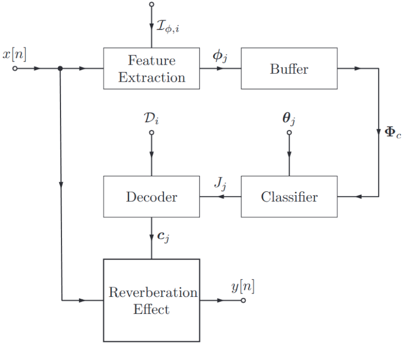


Fig. 1. Reverb application.

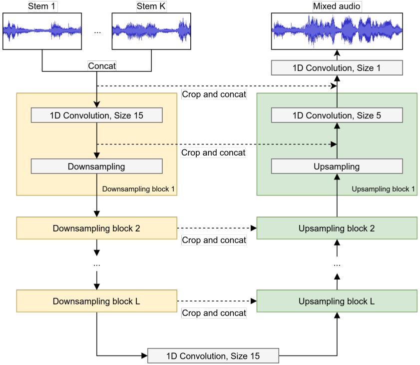
Automatic Mixing Review
Rule-Based & Deep Learning Approaches

Rule-Based and Traditional Machine Learning Systems

- Knowledge-engineered autonomous mixing [1]
- A machine-learning approach for instrument-specific application of artificial reverberation. [2]

Deep Learning Architectures

- Wave-U-Net autoencoders for automatic mixing [3]
- Differentiable mixing console with neural effects [4]



Automatic Mixing Review

Rule-Based & Deep Learning Approaches

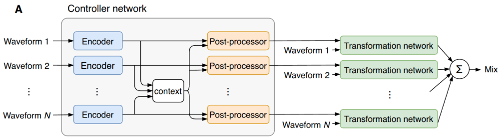
Georgia Tech College of Design Center for Music Technology

Rule-Based and Traditional Machine Learning Systems

- Knowledge-engineered autonomous mixing [1]
- A machine-learning approach for instrument-specific application of artificial reverberation. [2]

Deep Learning Architectures

- Wave-U-Net autoencoders for automatic mixing [3]
- Differentiable mixing console with neural effects [4]



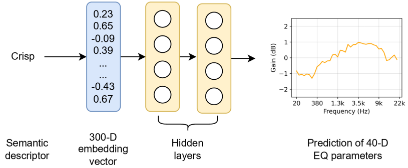
Automatic Mixing Review

Semantic Approaches

Georgia Tech · College of Design Center for Music Technology

Language-Audio Integration

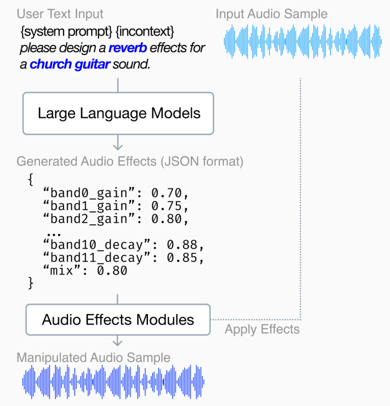
- Word-embedding approaches linking audio and language for effects/EQ recommendations [5], [6], [7]
- Text-driven interfaces mapping natural language to effect parameters and mix actions [8], [9], [10]



Automatic Mixing Review
Semantic Approaches

Language-Audio Integration

- Word-embedding approaches linking audio and language for effects/EQ recommendations [5], [6], [7]
- Text-driven interfaces mapping natural language to effect parameters and mix actions [8], [9], [10]

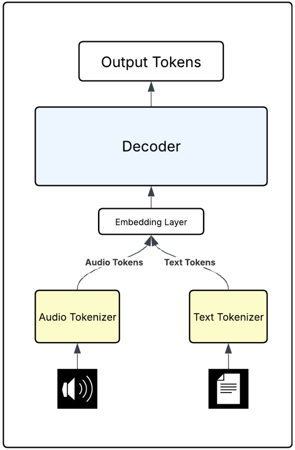


Automatic Mixing Review

Architectural Approaches for Audio-Language Models

Georgia Tech · College of Design
Center for Music Technology

- **Direct Tokenization (Unified Approach)**: converts raw audio into discrete tokens via audio codecs; tokens are flattened into a 1D sequence as LLM input; the LLM vocabulary is extended to include audio tokens [11], [12], [13].
- **Feature Extraction (Cascade Approach)**: uses audio-specific encoders/decoders with the LLM as a central backbone; high-level features are passed between modules (e.g., M^{2}UGen) [14], [15].



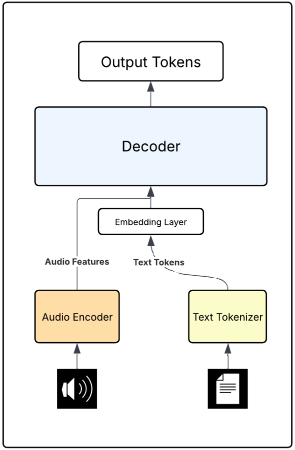
Unified Approach

Automatic Mixing Review

Architectural Approaches for Audio-Language Models

Georgia Tech · College of Design
Center for Music Technology

- **Direct Tokenization (Unified Approach)**: converts raw audio into discrete tokens via audio codecs; tokens are flattened into a 1D sequence as LLM input; the LLM vocabulary is extended to include audio tokens [11], [12], [13].
- **Feature Extraction (Cascade Approach)**: uses audio-specific encoders/decoders with the LLM as a central backbone; high-level features are passed between modules (e.g., M^2UGen) [14], [15].



Cascade Approach

Research Questions

primary

Primary Research Question

- To what extent can an **Audio-Language Model**, conditioned on an **anchor track**, learn the **relative gain** relationships among multitrack stems and generate musically effective gain-balancing **advice**?

Research Questions

secondary

Secondary Questions

- **Model Understanding**: What model architecture best represents and reasons about multitrack stems and anchor tracks for learning relative gain relationships?
- **Mixing Conventions**: To what extent does the model's advice reflect established mixing conventions?
- **Communication**: How effectively does the model communicate its advice in a way that is clear, actionable, and distinct from simply being "correct"?
- **Usefulness**: How do audio engineers and producers evaluate the effectiveness, musicality, and real-world usefulness of the advice?

Research Questions

secondary

Secondary Questions

- **Model Understanding**: What model architecture best represents and reasons about multitrack stems and anchor tracks for learning relative gain relationships?
- **Mixing Conventions**: To what extent does the model's advice reflect established mixing conventions?
- **Communication**: How effectively does the model communicate its advice in a way that is clear, actionable, and distinct from simply being "correct"?
- **Usefulness**: How do audio engineers and producers evaluate the effectiveness, musicality, and real-world usefulness of the advice?

Research Questions

secondary

Georgia Tech · College of Design
Center for Music Technology

Secondary Questions

- **Model Understanding**: What model architecture best represents and reasons about multitrack stems and anchor tracks for learning relative gain relationships?
- **Mixing Conventions**: To what extent does the model's advice reflect established mixing conventions?
- **Communication**: How effectively does the model communicate its advice in a way that is clear, actionable, and distinct from simply being "correct"?
- **Usefulness**: How do audio engineers and producers evaluate the effectiveness, musicality, and real-world usefulness of the advice?

Research Questions

secondary

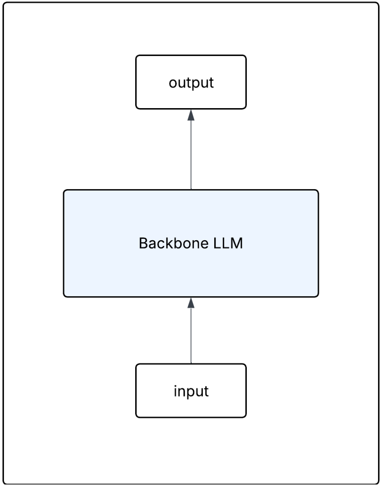
Secondary Questions

- **Model Understanding**: What model architecture best represents and reasons about multitrack stems and anchor tracks for learning relative gain relationships?
- **Mixing Conventions**: To what extent does the model's advice reflect established mixing conventions?
- **Communication**: How effectively does the model communicate its advice in a way that is clear, actionable, and distinct from simply being "correct"?
- **Usefulness**: How do audio engineers and producers evaluate the effectiveness, musicality, and real-world usefulness of the advice?

Proposed Method

Overview

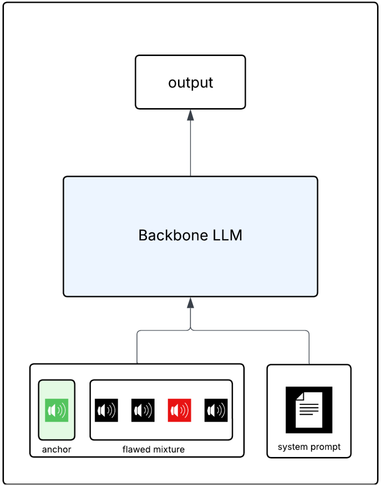
- **LLM as backbone**: A pretrained Audio LLM like Qwen-Audio [16] as the backbone.
- **Input**: Anchor track, rest of the tracks, and a text prompt.
- **Output**: A structured response containing **advice** for the user to **balance the levels** of the tracks.
- **Architecture**: Cascade approach, with the LLM as the backbone.
- **Training strategy**: Supervised fine-tuning using PEFT (Parameter-Efficient Fine-Tuning), specifically LoRA [17] or QLoRA [18].



Proposed Method
Overview

Georgia Tech · College of Design
Center for Music Technology

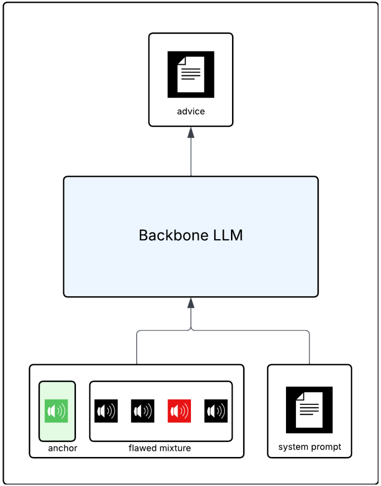
- **LLM as backbone**: A pretrained Audio LLM like Qwen-Audio [16] as the backbone.
- **Input**: Anchor track, rest of the tracks, and a text prompt.
- **Output**: A structured response containing **advice** for the user to **balance the levels** of the tracks.
- **Architecture**: Cascade approach, with the LLM as the backbone.
- **Training strategy**: Supervised fine-tuning using PEFT (Parameter-Efficient Fine-Tuning), specifically LoRA [17] or QLoRA [18].



Proposed Method

Overview

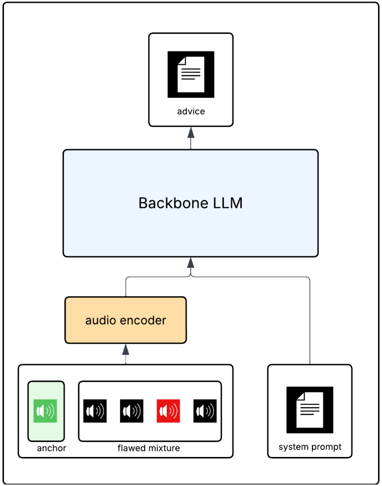
- **LLM as backbone**: A pretrained Audio LLM like Qwen-Audio [16] as the backbone.
- **Input**: Anchor track, rest of the tracks, and a text prompt.
- **Output**: A structured response containing **advice** for the user to **balance the levels** of the tracks.
- **Architecture**: Cascade approach, with the LLM as the backbone.
- **Training strategy**: Supervised fine-tuning using PEFT (Parameter-Efficient Fine-Tuning), specifically LoRA [17] or QLoRA [18].



Proposed Method

Overview

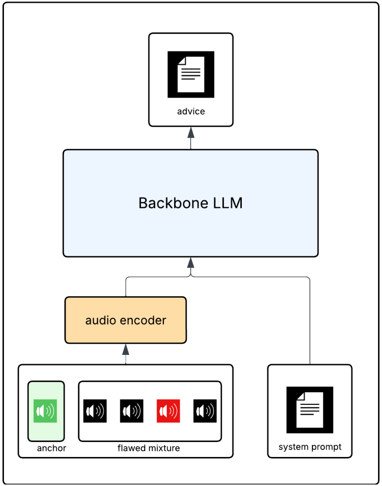
- **LLM as backbone**: A pretrained Audio LLM like Qwen-Audio [16] as the backbone.
- **Input**: Anchor track, rest of the tracks, and a text prompt.
- **Output**: A structured response containing **advice** for the user to **balance the levels** of the tracks.
- **Architecture**: Cascade approach, with the LLM as the backbone.
- **Training strategy**: Supervised fine-tuning using PEFT (Parameter-Efficient Fine-Tuning), specifically LoRA [17] or QLoRA [18].



Proposed Method

Overview

- **LLM as backbone**: A pretrained Audio LLM like Qwen-Audio [16] as the backbone.
- **Input**: Anchor track, rest of the tracks, and a text prompt.
- **Output**: A structured response containing **advice** for the user to **balance the levels** of the tracks.
- **Architecture**: Cascade approach, with the LLM as the backbone.
- **Training strategy**: Supervised fine-tuning using PEFT (Parameter-Efficient Fine-Tuning), specifically LoRA [17] or QLoRA [18].



Proposed Method
Dataset Synthesis

- **Prompt**: Create per-sample prompts in addtion to the constant system prompt.
- **Multitrack Input**:
  - **Dataset**: A multitrack dataset like MUSDB18 [19].
  - Chunk a song into 10-second segments.
  - Inject an error of $\pm n$ dB on a non-anchor track.
- **Response Formulation**: Programmitically create structured responses.
  - Create templates based on the prompt. The templates will be informed by established mixing conventions.
  - Pick a template based on the prompt, containing both description and solution placeholders.
  - Populate the placeholders with the ground truth description and solution.

Proposed Method

Dataset Synthesis

- **Prompt**: Create per-sample prompts in addtion to the constant system prompt.
- **Multitrack Input**:
  - **Dataset**: A multitrack dataset like MUSDB18 [19].
  - Chunk a song into 10-second segments.
  - Inject an error of $\pm n$ dB on a non-anchor track.
- **Response Formulation**: Programmitically create structured responses.
  - Create templates based on the prompt. The templates will be informed by established mixing conventions.
  - Pick a template based on the prompt, containing both description and solution placeholders.
  - Populate the placeholders with the ground truth description and solution.

Proposed Method
Dataset Synthesis

- **Prompt**: Create per-sample prompts in addtion to the constant system prompt.
- **Multitrack Input**:
  - **Dataset**: A multitrack dataset like MUSDB18 [19].
  - Chunk a song into 10-second segments.
  - Inject an error of $\pm n$ dB on a non-anchor track.
- **Response Formulation**: Programmitically create structured responses.
  - Create templates based on the prompt. The templates will be informed by established mixing conventions.
  - Pick a template based on the prompt, containing both description and solution placeholders.
  - Populate the placeholders with the ground truth description and solution.

Proposed Method

Dataset Synthesis

- **Prompt**: Create per-sample prompts in addtion to the constant system prompt.
- **Multitrack Input**:
  - **Dataset**: A multitrack dataset like MUSDB18 [19].
  - Chunk a song into 10-second segments.
  - Inject an error of $\pm n$ dB on a non-anchor track.
- **Response Formulation**: Programmitically create structured responses.
  - Create templates based on the prompt. The templates will be informed by established mixing conventions.
  - Pick a template based on the prompt, containing both description and solution placeholders.
  - Populate the placeholders with the ground truth description and solution.

Proposed Method

Dataset Synthesis

- **Prompt**: Create per-sample prompts in addtion to the constant system prompt.
- **Multitrack Input**:
  - **Dataset**: A multitrack dataset like MUSDB18 [19].
  - Chunk a song into 10-second segments.
  - Inject an error of $\pm n$ dB on a non-anchor track.
- **Response Formulation**: Programmitically create structured responses.
  - Create templates based on the prompt. The templates will be informed by established mixing conventions.
  - Pick a template based on the prompt, containing both description and solution placeholders.
  - Populate the placeholders with the ground truth description and solution.

Proposed Method

Dataset Synthesis

- **Prompt**: Create per-sample prompts in addtion to the constant system prompt.
- **Multitrack Input**:
  - **Dataset**: A multitrack dataset like MUSDB18 [19].
  - Chunk a song into 10-second segments.
  - Inject an error of $\pm n$ dB on a non-anchor track.
- **Response Formulation**: Programmitically create structured responses.
  - Create templates based on the prompt. The templates will be informed by established mixing conventions.
  - Pick a template based on the prompt, containing both description and solution placeholders.
  - Populate the placeholders with the ground truth description and solution.

Proposed Method

Dataset Synthesis

- **Prompt**: Create per-sample prompts in addtion to the constant system prompt.
- **Multitrack Input**:
  - **Dataset**: A multitrack dataset like MUSDB18 [19].
  - Chunk a song into 10-second segments.
  - Inject an error of $\pm n$ dB on a non-anchor track.
- **Response Formulation**: Programmitically create structured responses.
  - Create templates based on the prompt. The templates will be informed by established mixing conventions.
  - Pick a template based on the prompt, containing both description and solution placeholders.
  - Populate the placeholders with the ground truth description and solution.

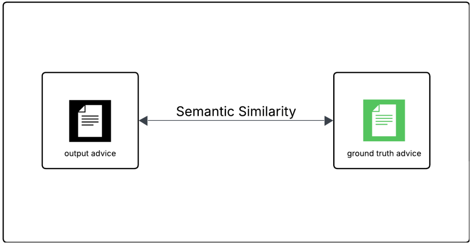
Evaluation Framework

Human Evaluation

- **Participants**: Semi-professional audio engineers and producers
- **Evaluation Criteria**:
  - **Effectiveness**: How well does the advice address the mixing challenge?
  - **Actionability**: How clear and implementable is the advice?
  - **Adherence to Conventions**: How well does the advice follow established mixing practices?
- **Methodology**: Rating scales and qualitative feedback collection

Evaluation Framework
Automated Evaluation

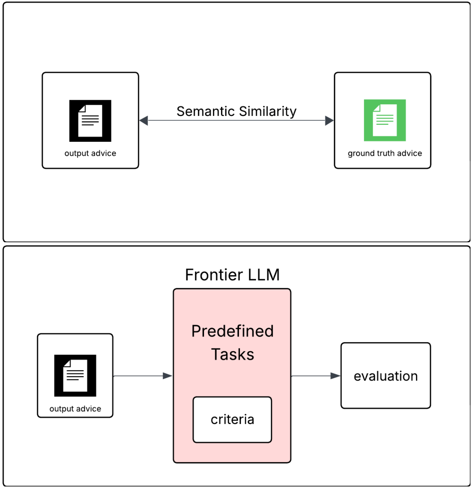
- **Semantic Similarity**: Compare advice to expert annotations
- **LLM-as-a-Judge**: Use LLMs to rate advice quality and relevance
- **Gain Advice Accuracy**:
  - **Direction accuracy**: Increase vs. decrease correctness
  - **Magnitude accuracy**: Correctness of categorical intensity labels (e.g., "too loud")



Evaluation Framework

Automated Evaluation

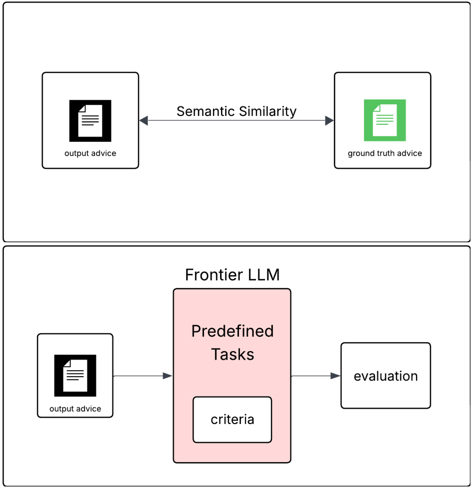
- **Semantic Similarity**: Compare advice to expert annotations
- **LLM-as-a-Judge**: Use LLMs to rate advice quality and relevance
- **Gain Advice Accuracy**:
  - **Direction accuracy**: Increase vs. decrease correctness
  - **Magnitude accuracy**: Correctness of categorical intensity labels (e.g., "too loud")



Evaluation Framework

Automated Evaluation

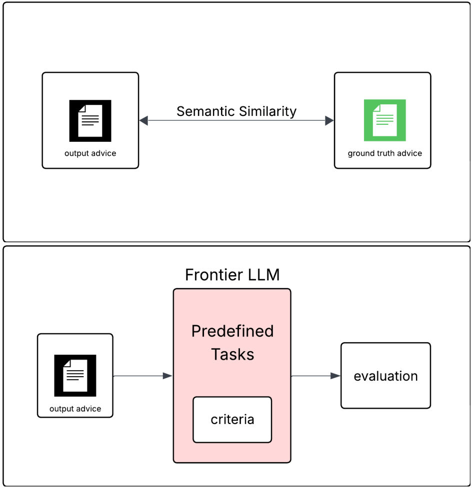
- **Semantic Similarity**: Compare advice to expert annotations
- **LLM-as-a-Judge**: Use LLMs to rate advice quality and relevance
- **Gain Advice Accuracy**:
  - **Direction accuracy**: Increase vs. decrease correctness
  - **Magnitude accuracy**: Correctness of categorical intensity labels (e.g., “too loud”)



Evaluation Framework

Automated Evaluation

- **Semantic Similarity**: Compare advice to expert annotations
- **LLM-as-a-Judge**: Use LLMs to rate advice quality and relevance
- **Gain Advice Accuracy**:
  - **Direction accuracy**: Increase vs. decrease correctness
  - **Magnitude accuracy**: Correctness of categorical intensity labels (e.g., “too loud”)



Limitations

- **Focus on Gain Only**: The model's scope is limited to gain-balancing advice; it does not address other effects like EQ, compression, or spatial effects.
- **Advisory, Not Prescriptive**: Evaluation focuses on the usefulness of the textual advice, not the numeric accuracy of specific gain predictions.
- **Dataset Dependency**: The project relies on the MUSDB18 dataset for valid "ground truth" for professional mixes.

Timeline

Tasks Leading to Nov. 28th Submission

- ~~Dataset preprocessing and JSONL format conversion.~~
- ~~Initial codebase and data loading pipeline setup.~~
- ~~Partial fine-tuning pilot experiments and architecture testing.~~
- Finalize architecture and execute all remaining experiments.
- Submit and obtain IRB approval for human studies.
- Conduct the human evaluation study with audio professionals.
- Complete the final paper, web interface, and Hugging Face deployment.

references

[1] E. Pérez-González and J. D. Reiss, "A knowledge-engineered autonomous mixing system," in *Audio Engineering Society Convention 135*, Oct. 2013. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=16953.

[2] E. Chourdakis and J. Reiss, "A machine-learning approach to application of intelligent artificial reverberation," en, *Journal of the Audio Engineering Society*, vol. 65, no. 1/2, pp. 56–65, Feb. 2017, ISSN: 15494950. DOI: 10.17743/jaes.2016.0069.

[3] E. Chourdakis and J. Reiss, "Automatic music signal mixing system based on one-dimensional wave-u-net autoencoders," en, 2022. DOI: 10.1186/s13636-022-00266-3. [Online]. Available: https://www.researchgate.net/publication/366902955_Automatic_music_signal_mixing_system_based_on_one-dimensional_Wave-U-Net_autoencoders.

[4] C. J. Steinmetz, J. Pons, S. Pascual, and J. Serrà, "Automatic multitrack mixing with a differentiable mixing console of neural audio effects,", no. arXiv:2010.10291, Oct. 2020, arXiv:2010.10291 [eess]. DOI: 10.48550/arXiv.2010.10291. [Online]. Available: http://arxiv.org/abs/2010.10291.

[5] A. Chu, P. O'Reilly, J. Barnett, and B. Pardo, "Text2fx: Harnessing clap embeddings for text-guided audio effects,", no. arXiv:2409.18847, Feb. 2025, arXiv:2409.18847 [eess]. DOI: 10.48550/arXiv.2409.18847. [Online]. Available: http://arxiv.org/abs/2409.18847.

[6] S. Venkatesh, D. Moffat, and E. R. Miranda, "Word embeddings for automatic equalization in audio mixing," en, *Journal of the Audio Engineering Society*, vol. 70, no. 9, pp. 753–763, Nov. 2022, ISSN: 15494950. DOI: 10.17743/jaes.2022.0047.

[7] E. Chourdakis and J. Reiss, *A semantic approach to autonomous mixing*, en, 2016. [Online]. Available: https://www.researchgate.net/publication/273574043_A_Semantic_Approach_To_Autonomous_Mixing.

[8] S. Doh, J. Koo, M. A. Martínez-Ramírez, W.-H. Liao, J. Nam, and Y. Mitsufuji, "Can large language models predict audio effects parameters from natural language?," no. arXiv:2505.20770, Jul. 2025, arXiv:2505.20770 [cs]. DOI: 10.48550/arXiv.2505.20770. [Online]. Available: http://arxiv.org/abs/2505.20770.

references

Georgia Tech · College of Design
Center for Music Technology

[9] J. Melechovsky, A. Mehrish, and D. Herremans, "Sonicmaster: Towards controllable all-in-one music restoration and mastering,", no. arXiv:2508.03448, Aug. 2025, arXiv:2508.03448 [eess]. DOI: 10.48550/arXiv.2508.03448. [Online]. Available: http://arxiv.org/abs/2508.03448.

[10] M. P. Clemens and A. Marasovic, "Mixassist: An audio-language dataset for co-creative AI assistance in music mixing,", 2025. [Online]. Available: https://openreview.net/forum?id=5mICyyD4OF.

[11] P. K. Rubenstein et al., "Audiopalm: A large language model that can speak and listen,", no. arXiv:2306.12925, Jun. 2023, arXiv:2306.12925 [cs]. DOI: 10.48550/arXiv.2306.12925. [Online]. Available: http://arxiv.org/abs/2306.12925.

[12] Z. Du et al., "Lauragpt: Listen, attend, understand, and regenerate audio with gpt,", no. arXiv:2310.04673, Jul. 2024, arXiv:2310.04673 [cs]. DOI: 10.48550/arXiv.2310.04673. [Online]. Available: http://arxiv.org/abs/2310.04673.

[13] D. Zhang et al., "Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities,", no. arXiv:2305.11000, May 2023, arXiv:2305.11000 [cs]. DOI: 10.48550/arXiv.2305.11000. [Online]. Available: http://arxiv.org/abs/2305.11000.

[14] S. Liu, A. S. Hussain, Q. Wu, C. Sun, and Y. Shan, "M^{2}ugen: Multi-modal music understanding and generation with the power of large language models,", no. arXiv:2311.11255, Dec. 2024, arXiv:2311.11255 [cs]. DOI: 10.48550/arXiv.2311.11255. [Online]. Available: http://arxiv.org/abs/2311.11255.

[15] Y. Gong, H. Luo, A. H. Liu, L. Karlinsky, and J. Glass, "Listen, think, and understand,", no. arXiv:2305.10790, Feb. 2024, arXiv:2305.10790 [eess]. DOI: 10.48550/arXiv.2305.10790. [Online]. Available: http://arxiv.org/abs/2305.10790.

[16] Y. Chu et al., "Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models,", no. arXiv:2311.07919, Dec. 2023, arXiv:2311.07919 [eess]. DOI: 10.48550/arXiv.2311.07919. [Online]. Available: http://arxiv.org/abs/2311.07919.

references

[17] E. J. Hu et al., "Lora: Low-rank adaptation of large language models,", no. arXiv:2106.09685, Oct. 2021, arXiv:2106.09685 [cs]. DOI: 10.48550/arXiv.2106.09685. [Online]. Available: http://arxiv.org/abs/2106.09685.

[18] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: Efficient finetuning of quantized llms," *arXiv preprint arXiv:2305.14314*, 2023, arXiv:2305.14314. [Online]. Available: https://arxiv.org/abs/2305.14314.

[19] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, *Musdb18-hq - an uncompressed version of musdb18*, Aug. 2019. DOI: 10.5281/zenodo.3338373. [Online]. Available: https://doi.org/10.5281/zenodo.3338373.