Enhancing Text-Audio Generation By Music Classification And Retrieval-Augmented Generation

Runyu He, Bochen Wang, Junyi Zhu, Yixuan Yin

May 1, 2024

Abstract

Recent advancements in deep learning have propelled the development of AI systems capable of generating music that resonates with human emotions and preferences. However, current music generation models still struggle to align generated music with detailed textual descriptions and maintain consistency, especially for longer compositions. This paper presents an innovative approach to address these challenges by integrating genre classification and retrieval-augmented generation (RAG) into the music generation pipeline. We train advanced CNN architectures, including ResNet-50, GoogleNet, and VGG16, for accurate genre classification. The classifier is then incorporated into a RAG framework, where the most relevant pre-classified music piece is retrieved based on the input text query. The retrieved audio and the text description are then fed into the MUSICGEN model to generate a new music piece that inherits attributes from both inputs. We evaluate our system through a double-blind human study, comparing the outputs of the original MUSICGEN model with our RAG-enhanced model. The results demonstrate a significant improvement in the ability of the RAG-enhanced model to generate music embodying specific stylistic elements, as evidenced by higher average confidence scores from participants. Our work represents a significant step towards more personalized and context-aware AI-generated musical experiences, laying the foundation for future advancements in this exciting field.

Code and Resources

The source code for this project is available at MusicMindMeld: Music Generation Enhanced By RAG and Classified Knowledge Base.

1 Introduction

The advent of deep learning has revolutionized the field of music generation, giving rise to AI systems capable of composing music that resonates with human emotions and preferences. Pioneering models like OpenAI's Jukebox (Dhariwal et al., 2020) and Google's Magenta (Engel et al., 2020) have demonstrated significant capabilities in generating music across various styles. More recently, MUSICGEN by Meta AI (Défossez et al., 2022) has made strides in simplifying the music generation pipeline and enhancing quality through efficient token interleaving patterns.

Despite these advancements, current music generation AI still faces shortcomings in aligning generated music to detailed textual descriptions and maintaining consistency, especially for longer compositions. Creating music based on unstructured text remains a complex task due to the wide range of possible descriptions covering various genres, instruments, tempos, scenarios, and subjective emotions (Agostinelli et al., 2023).

Retrieval-augmented generation (RAG) has shown promise in supporting generative models, particularly for text-based tasks. By integrating a retrieval mechanism, RAG enables models to access and utilize relevant information from external knowledge bases, enhancing the quality and coherence of generated outputs (Lewis et al., 2020). Recent studies have demonstrated RAG's effectiveness in improving the factual accuracy and fluency of language models (Shuster et al., 2021).

Our project, "Enhancing Text-to-Audio Generation by Genre Classification and Retrieval-Augmented Generation," seeks to leverage the power of RAG to refine the interface between technology and music. By incorporating a retrieval component based on genre classification, we aim to enhance the fidelity

and applicability of generated audio, enabling the creation of music that more closely aligns with targeted text descriptions.

Our approach involves training advanced CNN architectures like ResNet-50, GoogleNet, and VGG16 for accurate genre classification. This classifier is then integrated into a retrieval-augmented generation pipeline, where the most relevant pre-classified music piece is retrieved based on the input text query. The retrieved audio is then fed into the MUSICGEN model along with the text description to generate a new music piece that inherits attributes from both inputs.

Through this innovative fusion of genre classification and RAG, our system demonstrates improved ability to generate music embodying specific stylistic elements, as evidenced by human evaluations. This work represents a significant step towards more personalized and emotionally connected AI-generated musical experiences, laying the foundation for future advancements in this exciting field.

2 Background

2.1 Music Classification

A common technique involves converting audio signals into spectrograms—visual representations that depict time-frequency information (Wyse, 2017). These spectrograms are traditionally processed using Convolutional Neural Networks (CNNs) like VGG (Simonyan, 2014), a model widely implemented through the torchvision library for classifying various music characteristics.

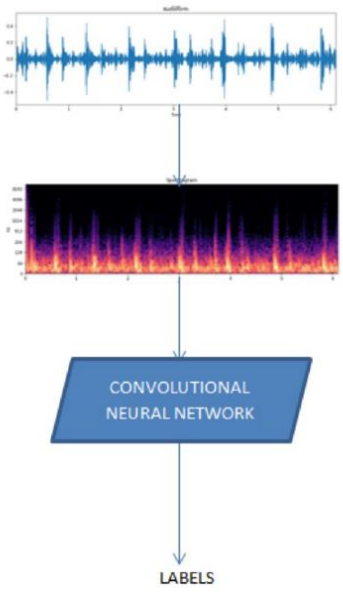

Figure 1: CNN Classification Through Spectrogram

Recent advancements have, however, extended beyond traditional CNN architectures due to their limitations in handling highly diverse and complex music tracks, which often led to a plateau in model accuracy and adaptability (Simonyan, 2014). Newer architectures like ConvNextV2, introduced in 2020, incorporate residual connections to mitigate the issues of exploding or diminishing gradients commonly observed in deep networks (Liu, 2022). EfficientNetV2, unveiled in 2021, enhances training speed and model efficiency, optimizing resource allocation during model training (Tan, 2021). Moreover, the Vision Transformer architecture shifts focus towards leveraging attention mechanisms, reinforcing the notion that "attention is all you need" for significant improvements in image and, by extension, audio classification tasks (Dosovitskiy, 2020) (Vaswani, 2017).

Our research builds upon these innovative approaches by exploring their application in music classification. Initially, we replicate existing classification methodologies using these advanced models to assess their efficacy. Our classification tasks utilize openly available text-audio datasets to broaden

our training scope. For instance, the GTZAN dataset, which consists of 1,000 audio clips each lasting 30 seconds and labeled by one of ten genres, serves as a foundational dataset. We also consider the Google Audio Set, comprising 10-second sound clips extracted from 2.1 million YouTube videos annotated across 527 classes, which include a diverse range of sounds from instruments to animal noises.

2.2 Music Generation

The landscape of music generation has been profoundly reshaped by the advent of advanced neural networks, which now enable the creation of music across various styles and the simulation of human-like vocal textures. Pioneering models like OpenAI's Jukebox (Dhariwal, 2020) and Google's Magenta (Engel, 2020) have demonstrated significant capabilities in generating music by learning complex patterns from extensive musical datasets. These models employ deep neural architectures to generate compositions that are both innovative and reflective of learned styles, though they often struggle with aligning closely to detailed textual descriptions and maintaining consistency throughout longer compositions.

One of the significant strides in addressing these challenges is the development of MUSICGEN by Meta AI (Défossez, 2022) (Agostinelli, 2023). This model represents a paradigm shift in music generation, utilizing a single-stage transformer language model that processes multiple streams of compressed discrete music representations. Unlike traditional models that required cascading stages for effective generation, MUSICGEN uses efficient token interleaving patterns to streamline the generation process, allowing for enhanced control over the produced musical pieces. This approach not only simplifies the music generation pipeline but also enhances the quality of the audio output, making it possible to generate music that closely adheres to given textual or melodic inputs.

MUSICGEN incorporates advanced text conditioning capabilities, which play a crucial role in aligning the generated music with textual descriptions. Text conditioning in MUSICGEN involves encoding textual descriptions into embeddings that guide the music generation process (Wu, 2023). This is achieved through a sophisticated architecture that integrates both the audio and text inputs, allowing the model to attend to relevant features from both modalities. The text encoder processes the descriptions and converts them into a format that the music generation model can utilize effectively, ensuring that the generated music not only exhibits high fidelity but also aligns with the textual cues provided by the user.

While MUSICGEN can generate music based on simple textual descriptions, its ability to understand and incorporate more complex narrative contexts or emotional subtleties into the music remains limited. Improving text conditioning in MUSICGEN could further enhance its utility and applicability in diverse music generation scenarios. As the technology evolves, focusing on these areas could lead to more nuanced and context-aware music generation systems that better serve the creative intentions of users.

3 Method

3.1 Music Classification

1. Dataset Compilation

   In our study, we have developed a specialized dataset dedicated to Jay Chou's music, known as the Jay Chou Dataset, comprising 42 carefully selected instrumental tracks. This dataset was specifically curated to focus solely on the musical elements of the tracks, excluding vocals. To ensure compatibility with existing music classification frameworks, each track was processed into 30-second WAV format segments, mirroring the structure of the well-established GTZAN dataset. This preprocessing was accomplished through a systematic pipeline that converted the original MP3 files into the desired format, facilitating easier ingestion by our classification models.

   Subsequently, we integrated this custom dataset with the GTZAN dataset, which consists of 1,000 audio tracks, each lasting 30 seconds and evenly distributed across ten distinct musical genres. For our study, we introduced an eleventh category labeled 'JAY CHOU' to encompass the unique genre represented by the tracks in our custom dataset. This integration allows us to not only maintain the diversity of musical genres but also to enrich the dataset with a distinctive style embodied by Jay Chou's music.

2. Baseline Comparison

   For our baseline model, we adopted the architecture from the study published two year ago, "Music Genre Classification using Machine Learning Techniques," which aimed at automating the organization of music libraries through genre classification. This study leveraged the extensive Audio set dataset, which includes over two million human-labeled sound clips from YouTube, classified into 632 audio event classes. Inspired by this work, we implemented a ResNet-50 model, known for its depth and efficiency in handling complex image classifications, adapted here for audio data. The model was configured with Cross-Entropy as the loss function and optimized using Adam with specified learning rates and weight decay parameters, all executed on Google Colab's GPU environment.

   Our goal was to exceed the 65% accuracy benchmark set by the previous study, aiming for a 70% accuracy threshold. We successfully surpassed this target, achieving an average accuracy of 75% across the ten standard genres, further validated by a 79% accuracy on our extended test set that included the Jay Chou category. These results not only demonstrate the efficacy of the ResNet-50 model in handling diverse musical data but also underscore the potential of integrating distinctive musical styles into traditional genre classification frameworks.

3. Model Implementation

   In our exploration, we implemented three different CNN architectures to refine our understanding of the most effective approaches for music genre classification:

   (a) ResNet-50: Previously described, it forms the backbone of our comparative analysis.

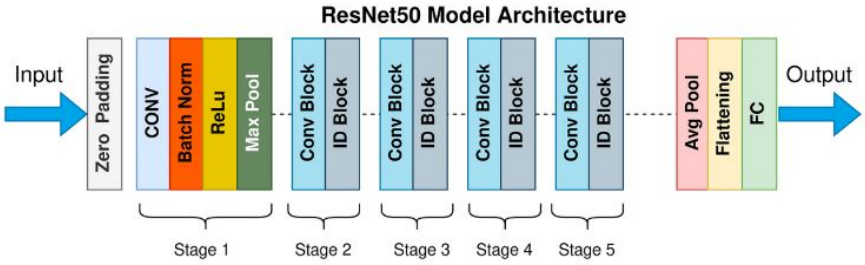

Figure 2: ResNet50 Architecture

   (b) GoogleNet: Introduced in "Going Deeper with Convolutions", GoogleNet is a deeper network that uses inception modules to effectively capture complex patterns in the data, significantly enhancing the model's ability to discern nuanced differences between genres.

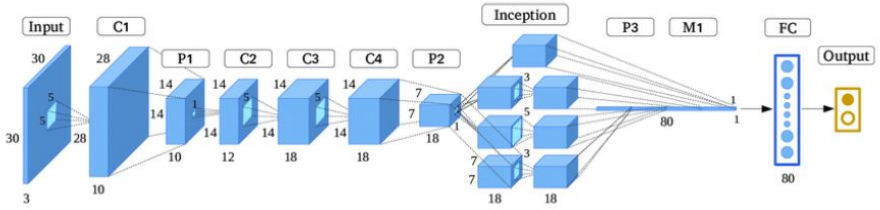

Figure 3: GoogleNet Architecture

   (c) VGG16: Known for its simplicity and depth, VGG16 was also adapted for our classification tasks, employing the same loss function and optimizer configurations as the other models.

4. Model Improvement

   To fine-tune the models effectively, we adopted Cross-Entropy loss as our primary loss function due to its efficacy in handling multi-class classification problems, like those found in music genre

classification. This choice is reinforced by the Adam Optimizer's ability to adaptively adjust learning rates based on the training data, providing a robust mechanism for model updates. Specifically, we set the learning rate to 0.0001 and applied a weight decay of 1e-4 to regularize the model and prevent overfitting. These parameters were chosen to balance the speed of convergence with the stability of the training process, ensuring that the models learn detailed features without memorizing the noise inherent in the training data.

A critical component of our model tuning strategy was the implementation of the EarlyStopping technique. This method monitors the validation loss during training and halts the training process if there is no appreciable improvement in model performance over a predetermined number of epochs. This approach is particularly valuable for several reasons:

(a) Prevention of Overfitting:

EarlyStopping is instrumental in preventing our models from overfitting. By monitoring performance on a validation set—not seen by the model during training—we can detect when the model starts to memorize rather than generalize from the training data. This is evidenced by good performance on the training set but poor performance on the validation set.

(b) Computational Efficiency:

Training deep neural networks is computationally intensive. By stopping the training early when no further improvements are observed, we save computational resources, which is crucial when working with large datasets and complex models.

(c) Hyperparameter Tuning:

During the hyperparameter tuning phase, EarlyStopping allows us to more efficiently explore the hyperparameter space. It reduces the time spent on less promising parameter combinations and redirects focus towards more potentially fruitful configurations. This not only accelerates the experimental process but also enhances the overall quality of the model tuning.

By integrating these strategic elements into our model tuning process, we ensure that our music classification models are not only accurate but also robust and efficient. This meticulous approach to model optimization underscores our commitment to developing a classification system that is both practical and scalable, capable of handling the complex nuances of diverse music genres.

3.2 Retrieval Augmented Generation

1. Data Annotation & Organization

To construct a robust database for retrieval, we utilized a classical music dataset complete with unique identifiers (IDs), music titles, and descriptive keywords or comments. These annotations serve as the foundation for accurately matching classical music pieces to user queries based on semantic content.

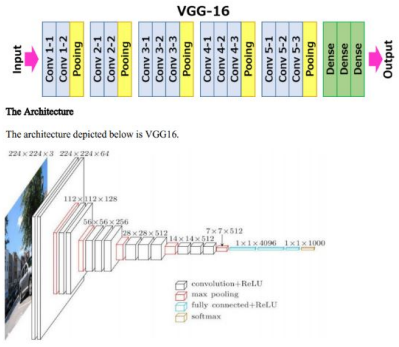

Figure 4: VGG16 Architecture

We employed a music classifier trained on advanced CNN architectures such as ResNet-50, GoogleNet, and VGG16. This classifier processes preprocessed audio data in the form of spectrograms to categorize the music by genres, artists, or styles. The classified pieces are then organized into specific databases, facilitating targeted retrieval that aligns with user preferences.

The unique identifiers associated with each piece in the dataset are crucial for linking metadata to the actual audio files. These IDs ensure efficient location and retrieval of corresponding WAV files, streamlining the user experience in accessing selected music pieces.

2. Retrieval Augmented Generation

For efficient management and retrieval of music documents, we implemented a document store utilizing the FAISS library, known for its high-performance similarity search and clustering of dense vectors (Johnson, 2017). Each classical music piece's metadata is transformed into a structured document format, with descriptive comments as content and music titles and IDs as metadata. These documents are indexed in the store, enabling rapid and precise retrieval based on semantic similarity.

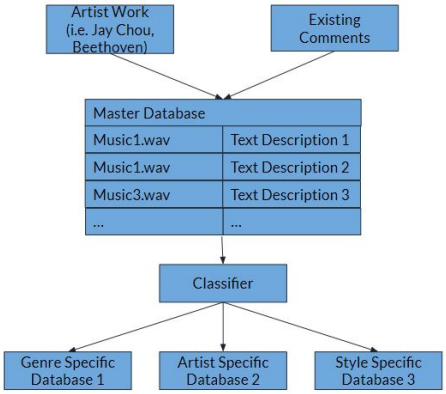

Figure 5: Classified Music Database Structure

To retrieve the most relevant music piece from a user's text query, we utilized a dense passage retriever (Karpukhin, 2020). This tool, equipped with pre-trained encoder models, maps the text query and music descriptions into a shared embedding space. By calculating cosine similarity between the query embedding and document embeddings, it identifies the music piece that best matches the semantic content of the query. Upon retrieving the most relevant document, we extract the music ID from its metadata. This ID is used to locate and preprocess the corresponding WAV file from our classical music collection.

For the music generation phase, we employed the MusicGen model from the Hugging Face Transformers library. MusicGen, a transformer-based autoregressive model, processes input audio and text descriptions, generating new music by predicting the next audio samples. The model attentively blends the audio and text inputs, allowing for the generation of music that reflects the characteristics of both the retrieved classical piece and the user's text description.

The processed audio data and text query are fed into the MusicGen processor, which handles necessary preprocessing steps like tokenization and feature extraction. The model then generates a new piece of music that inherits the attributes of both the retrieved audio and the text query through the transformer's attention mechanism.

This retrieval-augmented generation approach not only enhances the quality and coherence of the generated music but also creates a personalized audio experience that resonates with the user's preferences while maintaining the essence of the original classical music. The final output can be played directly within the environment or saved as a WAV file for further use and distribution, providing a seamless and enriching user experience.

4 Results

4.1 Music Classification

Upon testing, it was evident that the architectures exhibited significant variability in performance. Notably, the VGG16 model underperformed in comparison to its counterparts, achieving the lowest scores in both recall and F1-score. This could be attributed to VGG16's architecture potentially being less adept at capturing the nuanced features necessary for accurate music genre classification.

In contrast, ResNet-50 emerged as the superior model, outperforming the other architectures across all tested metrics. The higher performance of ResNet-50 suggests that its deeper and more complex structure, characterized by residual connections, is more effective at processing the spectral complexities inherent in musical data. This capability makes it particularly suited for tasks where distinguishing subtle differences between genres is crucial.

GoogleNet, while not performing as poorly as VGG16, still lagged behind ResNet-50. Its intermediate results may reflect the trade-offs inherent in its inception modules, which, while reducing parameter count, might not capture as detailed features as the more straightforward, deeper approach of ResNet-50.

Architecture	F1-score	Recall
ResNet-50	0.81	0.77
GoogleNet	0.57	0.50
VGG16	0.44	0.18

Table 1: Results on Testing Data: Performance Comparison of Different Architectures

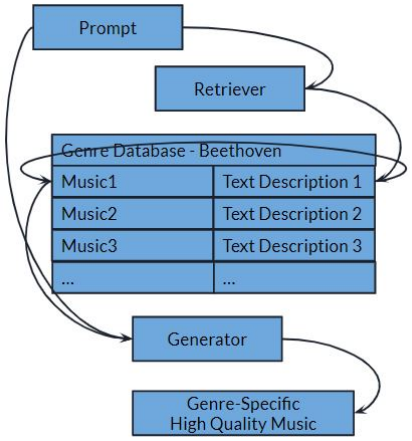

Figure 6: Retrieval Augmented Generation Pipeline

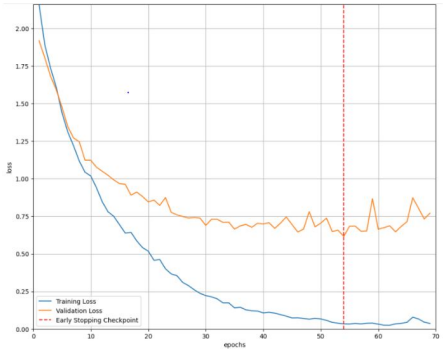

Figure 7: ResNet50 Training and Validation Loss

4.2 Music Generation

We conducted a double-blind human evaluation between the outputs form original MusicGen model and our enhanced model incorporating Retrieval-Augmented Generation (RAG) with a database specifically curated with Beethoven's music. The goal was to assess the effectiveness of each model in generating music that embodies Beethoven's stylistic elements.

Each of three individual subjects was asked to blindly evaluate 20 music samples—10 generated by the original MusicGen model and 10 by our RAG-enhanced model. The subjects rated their confidence on a scale from 1 to 5, where 5 indicates strong confidence that the generated music piece contained Beethoven-like elements.

The evaluation results are summarized in the table below:

Model	**Average Confidence Score**
Original MusicGen	2.4
RAG-enhanced MusicGen	4.2

Table 2: Average confidence scores of perceived Beethoven elements in music samples

The data indicates a significant improvement in the ability of the RAG-enhanced MusicGen model to generate music that participants believe retains the characteristic elements of Beethoven's style. The average confidence score for the RAG model was notably higher, suggesting that the integration of a targeted retrieval mechanism effectively aligns the generated music with the specific stylistic attributes of Beethoven.

5 Discussion

This project represents a significant advancement in the fusion of deep learning and music generation, particularly through the innovative application of music classification and retrieval-augmented generation pipeline. The use of advanced models has proven crucial in developing a system that not only accurately identifies various musical elements but also tailors the music generation process to reflect the diverse inputs from users. Our implementation of retrieval-augmented generation represents a particularly notable advancement, enabling dynamic selection of music pieces that closely align with user-specified text. This method has shown great promise in personalizing the music generation process, thereby enhancing user engagement and satisfaction.

Despite these advancements, the project also highlighted several challenges inherent in automated music generation. The variability of text descriptions and the subjective interpretation of music often complicate the generation process, making it difficult to produce universally satisfying musical outputs. Addressing these challenges will be crucial for further advancements in the field and will involve refining the models' ability to interpret and process complex textual inputs and subtle musical nuances.

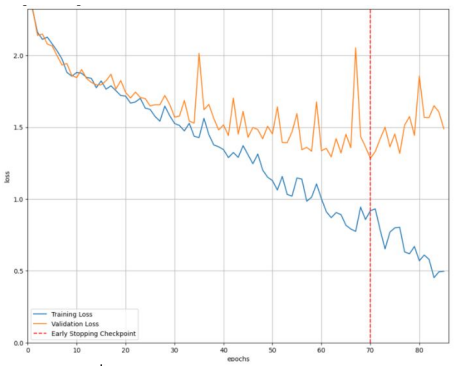

Figure 8: GoogleNet Training and Validation Loss

6 Conclusion

This project has not only contributed valuable insights into the integration of technology and music but also laid a robust foundation for future innovations in automated music generation. The techniques developed through this project could revolutionize the way music is generated, offering more personalized and emotionally connected musical experiences through artificial intelligence.

Looking forward, there are several exciting directions for further development:

1. Integration with Generative Models:

   Integrating the classification-based retrieval system with advanced generative models like Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) could enable the production of more nuanced and stylistically coherent music pieces. Leveraging genre-specific features identified by the music classifier could guide these generative models to produce music that captures the intricacies of different genres and styles effectively.

2. Incorporation of User Feedback and Interaction:

   Enhancing the system's adaptability to individual preferences through user feedback and interactive features could significantly improve the music generation process. Allowing users to provide feedback on the generated music and to interact with the generation process by adjusting parameters or selecting preferred music segments could lead to richer user engagement and better alignment with users' expectations.

By continuing to explore these avenues, the potential for creating more refined and user-responsive music generation systems becomes increasingly attainable, promising a future where AI-generated music can truly mimic the depth and dynamism of human composition.

7 References

1. Greenwade, G. D. (1993). The Comprehensive Tex Archive Network (CTAN). *TUGBoat*, 14(3), 342–351.

2. Wyse, L. (2017). Audio spectrogram representations for processing with convolutional neural networks. *arXiv preprint arXiv:1706.09559*.

3. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

4. Liu, Z., Mao, H., Wu, C. Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 11976–11986).

5. Tan, M., & Le, Q. (2021). EfficientNetV2: Smaller models and faster training. In *International Conference on Machine Learning* (pp. 10096–10106). PMLR.

6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

7. Dhariwal, P., et al. (2020). Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*.

8. Engel, J., Hantrakul, L., Gu, C., & Roberts, A. (2020). DDSP: Differentiable Digital Signal Processing. *arXiv preprint arXiv:2001.04643*.

9. Défossez, A., Copet, J., Synnaeve, G., & Adi, Y. (2022). High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*.

10. Wu, Y., Chen, K., Zhang, T., Hui, Y., Berg-Kirkpatrick, T., & Dubnov, S. (2023). Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 1-5). IEEE.

11. Agostinelli, A., Denk, T. I., Borsos, Z., Engel, J., Verzetti, M., Caillon, A., ... & Frank, C. (2023). MusicLM: Generating music from text. *arXiv preprint arXiv:2301.11325*.

12. Johnson, J., Douze, M., & Jégou, H. (2017). Billion-scale similarity search with GPUs. *arXiv preprint arXiv:1702.08734*.

13. Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., ... & Yih, W. T. (2020). Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

14. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems* (pp. 5998–6008).

15. Engel, J., Agrawal, K. K., Chen, S., Gulrajani, I., Donahue, C., & Roberts, A. (2020). GANSynth: Adversarial neural audio synthesis. *arXiv preprint arXiv:1902.08710*.

16. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *arXiv preprint arXiv:2005.11401*.

17. Shuster, K., Poff, S., Chen, M., Kiela, D., & Weston, J. (2021). Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.