Might change how I summarize the paper. Use bullet points, often with dumbed down analogies to explain a point. Break down the points into format given by Tom Collins. Include author names.

- Type of reference?
- What did this paper study?
- What was their approach?
- What was the problem question?
- Was there an ML approach? If yes, describe.
- What did the research/solution achieve?
- What did they not achieve a.k.a future scope?
- Possible use of this reference would be?
- Where does it stand in relation to other studies? Agree with or contradict?
- How, and how effectively, does this source address the topic?

"This paper studied <blank>. They used data from <blank>. ML approach was <blank>. Results indicated that <blank>. Strengths/weaknesses were <blank>. Potential for me to extend/alter/reconsider this approach in my own research: <blank>"

A. Barahona-Rıos, "Deep Learning for the Synthesis of Sound Effects".

-> Ch.4 -

[1]

A. Gungormusler, N. Paterson-Paulberg, and M. Haahr, "barelyMusician: An Adaptive Music Engine for Video Games," in *Audio Engineering Society Conference: 56th International Conference: Audio for Games*, Feb. 2015. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=17598

The paper introduces barelyMusician, an adaptive music engine for video games that focuses on the interactivity of musical elements and offers features for real-time generation and manipulation of audio samples and musical parameters.

-> What is it – Middleware tool, an adaptive music engine.

-> Utilizes a "hybrid approach" -> combines real time music generation and affective musical mood transition capabilities.

-> Focuses on practical aspects of composition transformation techniques.

-> Develop a generic comprehensive framework that abstracted the low-level musical properties and structures -> create intuitively without music knowledge.

-> Adaptive Audio techniques -> Vertical layering & Horizontal re-sequencing.

-> Architecture loosely based on “Flexible Music Composition Engine” by Hoeberechts et. Al, 2011.

-> Real-time sound synthesis and sampling techniques for better sound quality.

-> Desired audio data is generated procedurally from scratch or loaded from a pre-existing soundbank sample by sample by the “instruments” in composition phase.

-> Implemented in C# Unity for portability and ease of use.

-> Limitation – a major simplification was done by restricting music transformation to micro level to prevent complexity in real time. “Key note” was kept intact throughout because of difficulty in preserving the consonance of a particular music pattern.

-> 2 scenes for evaluation, a) Scene with GUI for interactive results b) Simple game scenario to observe how system can be used by a game developer. To assess level of interactivity and recognition of mood that the user may experience.

-> Combined real-time music generation + affective transformation methods -> manipulation of sounds in real-time.

-> No ML approach; All looks like a blackbox sort of thing, no information about what exactly goes inside the micro-macro level. Found the GitHub though. Also found the thesis from the first author: https://publications.scss.tcd.ie/theses/diss/2014/TCD-SCSS-DISSERTATION-2014-013.pdf

-> Potentially useful for me to refer on the “adaptive” side of things. Using the two techniques or perhaps improving them or adding to them using a different method.

GitHub: https://github.com/anokta/barelymusician

Link to demo: http://youtu.be/Qs3yDVtqt9s.

M. F. Rodrigues and F. R. S. Coutinho, “A Tool to Implement Adaptive Audio Techniques on Unity Games,” in *2021 20th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, Gramado, Brazil: IEEE, Oct. 2021, pp. 1–8. doi: 10.1109/SBGames54170.2021.00011.

-> The development of the Adaptive Audio Manager tool for Unity provides a solution for implementing adaptive audio techniques in Unity games, making it easier for beginner developers to create effective and dynamic audio systems. The tool offers classes and methods for vertical layering and horizontal re-sequencing techniques, allowing developers to control the reproduction of audio tracks and create a more immersive gaming experience.

-> The evaluation of the tool showed promising results, with programmers of different expertise levels finding it beneficial for game development. However, it was noted that the tool has a limitation of only allowing one layer to be played simultaneously, which restricts the control over which instrument is played at each game state.

-> Whole paper is a good reference for understanding what Vertical Layering and Horizontal Re-Sequencing is.

-> Vertical Layering: Vertical layering is an adaptive audio technique that involves segmenting an audio track into individual layers, each composed of one or more instruments or sound effects. This technique allows the user to activate or deactivate specific layers, introducing or removing desirable sounds based on in-game events.

-> So it is dependent on the "Game State", when it changes, a cue is sent to the Layer and it changes the music by adding or removing layers. Plus, gives the user more control (usually).

-> Horizontal Re-Sequencing: Horizontal re-sequencing is an adaptive audio technique used to reproduce queued audio tracks, similar to a playlist. It allows for smooth transitions between different audio tracks based on the current state of the game, creating a "seamless and immersive" audio experience. Each track can have a different duration, long tracks or short fragments, as long as they can musically connect with each other. Each track fragment is played according to the current state of the game.

-> The tool has a limitation of only allowing one layer to be played simultaneously, which restricts the control over which instrument is played at each game state.

Ringer, C., Missaoui, S., Hodge, V. J., Chitayat, A. P., Kokkinakis, A., Patra, S., ... & Walker, J. A. (2023). Time to die 2: Improved in-game death prediction in dota 2. *Machine Learning with Applications*, *12*, 100466.

-> The paper focuses on "moment-to-moment prediction" in esports matches, specifically predicting if a player will die within a set number of seconds in the game Dota 2. The prediction is based on telemetry data gathered directly from the game itself. An enhanced dataset is used, covering 9,822 Dota 2 matches and employing techniques such as learned embeddings for categorical features and modeling the temporal element of the telemetry data using **recurrent neural networks.** These enhancements improve the predictive power of the model, achieving an F1 score of 0.54 compared to 0.17 for the previous model. The paper also explores the suitability of the model as a broadcast aid for esports reporting.

-> Need to fully comprehend how their model works, felt a bit high-level as of now.

-> Data categorisation quite similar to Adrian's paper. This could be used as a great reference to find out the "future state" of the game without manually declaring it.

-> Although the whole gist of the paper is to aid broadcaster commentary, this prediction-based system could be utilized to proactively generate or ready a set of audio samples to match with “what may happen” in a game scene.

-> One thing I found useful is that there is also a possibility of a relationship between the learned spaces of the items and heroes in Dota 2, where certain items are more often used by a subset of heroes, leading to their co-location in their respective spaces. These potential relationships can be explored further to gain insights into gameplay strategies and decision-making processes. For different competitive games like Counter-Strike or Valorant, which are heavily strategy-based FPS competitive team games, there could be a possible implementation of game-state and future-state adaptive audio. Not in those games particularly but in a different game or game audio engine perhaps (story-based games where every decision you make changes outcome, etc).

S. Amiriparian, N. Cummins, M. Gerczuk, S. Pugachevskiy, S. Ottl, and B. Schuller, “‘Are You Playing a Shooter Again?!’ Deep Representation Learning for Audio-Based Video Game Genre Recognition,” *IEEE Trans. Games*, vol. 12, no. 2, pp. 145–154, Jun. 2020, doi: 10.1109/TG.2019.2894532.

-> The paper introduces a novel computer audition task: audio-based video game genre classification.

-> The aim of the study is to check the feasibility of the proposed task, to introduce a new “corpus” called G2AME (Game Genre by Audio Multimodal Extracts), and to compare the efficacy of various acoustic feature spaces for classifying the corpus into six game genres.

-> The paper highlights the extraction of three different feature representations from game audio files: knowledge-based acoustic features, DEEP SPECTRUM features, and quantized DEEP SPECTRUM features using Bag-of-Audio-Words.

-> DEEP SPECTRUM features are derived from forwarding the visual representations of audio instances through deep task-independent pretrained CNNs, specifically GoogLeNet, AlexNet, and VGG16.

-> The suitability of the deep learning approach for the audio-based video game genre classification task is studied, with the best method achieving an accuracy of up to 66.9% unweighted average recall using tenfold cross-validation. BODF is a combination of DEEP SPECTRUM and Bag of Audio Words.

-> To create BODF representations, the DEEP SPECTRUM features are transformed into images by creating color mapped plots. These plots are then fed to convolutional neural networks (CNNs) pretrained on ImageNet. The activations of a specific fully-connected layer in the CNNs are extracted as large feature vectors, which are denoted as DEEP conv and ch for convolutional layers and channels, respectively.

-> “The classification paradigms explored for the G2 AME corpus have been adopted from techniques successfully used in other acoustic-detection tasks. Our baseline system is based on acoustic feature sets which are extracted from audio clips of different video games using our openSMILE toolkit [16]. These feature sets are primarily used in computational paralinguistics based detection tasks [17]. However, they have also been used for movie genre classification [18], and music genre classification [19], [20]."

-> The above paragraph in the Introduction is intriguing. If a genre can be classified using audio, the reverse is also possible. This particular piece of research may very well help in setting up a layout for semantic input based on the game genre that the developer can define from the get-go. This would aid in certain aspects of the generation of audio/music for the particular game. As this paper has clearly concluded that their proposed method has a statistically significant difference from other proposed methods (?), perhaps building upon this method to have it implemented in conjunction with a possible generative audio system is feasible?

C. Bossalini, W. Raffe, and J. Andres Garcia, "Generative Audio and Real-Time Soundtrack Synthesis in Gaming Environments: An exploration of how dynamically rendered soundtracks can introduce new artistic sound design opportunities and enhance the immersion of interactive audio spaces.," in *32nd Australian* *Conference on Human-Computer Interaction*, Sydney NSW Australia: ACM, Dec. 2020, pp. 281–292. doi: 10.1145/3441000.3441075.

-> Explores the use of dynamically rendered soundtracks in gaming environments to enhance immersion and provide new artistic sound design opportunities.

-> The challenge faced by generative audio is the disconnect between experimental technologies and their incorporation into industry production pipelines. This is due to the creation of ground-up systems instead of leveraging existing frameworks.

-> The paper presents a system that allows for the adjustment of individual audio track parameters and implements a region-based system to change global music parameters such as reverb, tempo, key, and pattern. Custom collider regions are used to create parameter control regions.

-> The development of cross-platform and efficient virtual instruments is a hurdle in generative audio. Licensing and cross-platform capability are often lacking in virtual instruments created for specific DAW programs.

-> "The hybridized approach between fully generative and fully composed music allows for customization and control by creatives in the pipeline. It provides greater runtime customizability and the ability to change note quality, timbre, and pitch." Interesting.

-> Is a good article at most. Summarizes a lot of what is "needed" or "could haves, should haves, can be dones".

M. Lopez Ibanez, N. Alvarez, and F. Peinado, "Assessing The Effect of Adaptive Music On Player Navigation In Virtual Environments," Proceedings of the 21st International Conference on Digital Audio Effects (DAFx-18), Aveiro, Portugal, September4-8, 2018.

- The study aimed to explore the impact of adaptive music on player navigation in virtual environments.
- A video game with a 3D labyrinth was created, where players had to retrieve objects in the shortest time possible.
- Two groups of subjects played the game, with one group having the ability to choose musical attributes that influenced the spatialized music they heard during gameplay, while the other group had a default non-adaptive soundtrack.
- The time taken to complete the task was measured to assess user performance.
- A live music system called LitSens is the audio foundation for the game. LitSens automatically combined short fragments of music composed by a human and responded in real-time to player decisions.
- They aimed to evaluate how users' context affects auditive preferences and how these preferences impact performance.
- To evaluate the results, the researchers used a parametric, unpaired test (Student's t-test) and also asked subjects from group A to explain their attribute selection.
- The results showed a statistically significant correlation between player performance and the inclusion of adaptive music.
- The study concluded that there is a lack of firm musical criteria for making sounds prominent and easy to track for users, and an adaptive music system proved useful and effective for a complex user base.
- There is a reference to another paper from the same authors - "Towards an emotion driven adaptive system for video game music" which basically explores and suggests an adaptive engine based on the emotinal state of the player through dialogue choices.

C. Plut and P. Pasquier, "Generative music in video games: State of the art, challenges, and prospects," Entertainment Computing, vol. 33, p. 100337, Mar. 2020, doi: 10.1016/j.entcom.2019.100337.

- The study presents a laydown of generative music in video games and provides a survey of the current state of the art in this field.
- There is good discussion on how different video games employ generative audio and adaptive/procedural audio systems.
- Adaptive music is generally a piece of music composed in a way that, when broken in sections creates unique compositions, so more like derivations of one track. The track is broken into multiple groups of instruments and arrangements which results in multiple arrangements of one track. Adaptive music addresses the "linear" use of music - music is directly tied to the current level and/or game state. With linear com- posed music, the music begins playing through a musical piece when the associated level is loaded.

- Great example of Mozart’s Musikalisches würfelspiel, or musical dice game, a well-known piece of generative music, in which the score is made up of multiple musical sections, each of which can transition to any other section. Player/performer rolls the dice which determines the next musical section to be played.
- Paragraph to use as a good reference or literature - “For our purposes, music can be considered generative within a video game if the music is produced by a systemic automation that is partially or completely independent of the gameplay. This independence can have a large range of possibilities. A generative linear system may be almost completely independent of the gameplay – a piece of music can be requested, and is then linearly played through regardless of the gameplay. A highly adaptive generative music system may use a large array of game variables to inform the generation of musical content”.
- A taxonomical diagram that lists all aspects of generative and adaptive audio in games is given, which is a good cornerstone to my thesis topic. Currently deliberating between all the possible contexts.
- All but one of the generative music systems surveyed are using Markov chains models. Markov models are indeed simpler than deep learning/neural networks models, but they face the risk of plagiarism, by recopying too long sequences from the corpus.
- The only surveyed game music generation system based on artificial neural networks is Adaptive Music System (AMS) by Hutchings and McCormack.
- Apparently generative and procedural audio is interchangeable in this particular context.
- The paper suggests that future work should focus on leveraging the strengths of both human-composed and generative music together. It proposes that generative techniques should be increasingly used in games music, but in combination with human-composed music. This approach can ensure “smooth transitions between different musical states and layers”, while still allowing for the expressive qualities of human-composed music. The authors acknowledge that generative music requires more labor than composed music, as it involves creating algorithms and assigning music to different game states. However, they believe that the potential benefits of generative music in enhancing player experiences make it worth exploring further.

C. Plut and P. Pasquier, “Music Matters: An empirical study on the effects of adaptive music on experienced and perceived player affect,” in 2019 IEEE Conference on Games (CoG), London, United Kingdom: IEEE, Aug. 2019, pp. 1–8. doi: 10.1109/CIG.2019.8847951. ct

- The study shows that adaptive music can significantly increase a player's reported experienced feeling of tension. Players recognize and value music, and they recognize and value adaptive music over linear music.
- Enjoyment of the game increases with the introduction of music, and it slightly decreases when the music tension adapts inversely to the game tension. The adaptivity of the music is more important than the emotional suitability of the music. Players perceive and value music, and they perceive and value adaptive music more
- A game – Galactic Escape was made for this study: https://youtu.be/3vxXbMeJGkw
- The study they did is just about making the players play the game with their Independent Variable being the musical tension as it relates to game tension and Dependent Variable being the participant’s affective

response to the game. They fill out a questionnaire that had different moods and emotions as experienced after each “condition” of the game.

- The study found that players reported enjoying the game more when music was introduced, indicating a positive impact of music on player enjoyment.
- Enjoyment slightly decreased when the music tension adapted inversely to the game tension, but increased more when the music tension matched the game tension. This suggests that a better alignment between music and gameplay tension enhances player enjoyment.
- Perceived emotion responses showed a consistent trend, with players reporting that the music added to their overall experience. This effect increased as the music tension adapted to the game tension and further increased when the music tension matched the game tension.
- The experienced affective dimensions of valence and arousal followed a similar pattern to the perceived dimensions. Reported values for valence and arousal increased when music was introduced, and these values further rose when the music was adaptive and matched the gameplay tension.
- The player's reported feeling of tension differed from the pattern observed in valence, arousal, and perceived emotion responses. As music was introduced, player tension was reduced. However, as the music tension adapted to the game tension, player tension increased, and it rose further when the music tension matched the game tension.
- While it provides an important step in understanding the relationship between music and video games, there are still areas that need further exploration. The effects of measured variables may change with variations in interaction speed, different camera or avatar representations, and other unexplored facets such as listening environment, previous gaming experience, input device, and narrative elements.
- Biofeedback data is suggested as a means of getting emotional response, but it would be difficult.
- Overall, the study supports the hypothesis that music matters in video games for affective impact, and adaptive music has a stronger effect in increasing player enjoyment and strengthening the affective impact of a game.

P. E. Hutchings and J. McCormack, “Adaptive Music Composition for Games,” IEEE Trans. Games, vol. 12, no. 3, pp. 270–280, Sep. 2020, doi: 10.1109/TG.2019.2921979.

- The paper suggests that unpredicted, emergent moments in games should have unique musical identities. Combining visual content, narrative events, and emotive music can have documented effects on memory, immersion, and emotion perception in games.
- Music in games is also evaluated for narrative and functional fit. Current music composition practices for games rely on event triggers, leading to repetition and a lack of uniqueness for unpredictable events.
- The paper introduces an **adaptive music system** based on cognitive models and multi-agent algorithms to create unique, emotionally fitting music for video games. It emphasizes the need for a communicative model to relate game events, content, and moods to music.
- The **spreading activation model** is adopted to combine relationships between emotions, objects, and environments in games. Context descriptions from this model are used to inform a multi-agent music composition system that combines neural networks and evolutionary classifiers.

- THIS PAPER HAS A GREAT LITERATURE REVIEW.
- The authors acknowledge that there were hardly any evaluations of AMSs through user studies or predefined metrics of quality to judge the effectiveness of the implementation or underlying concepts.
- The system consists of the following three key components: 1) a spreading activation model of game context; 2) the six-category model of emotion for describing game context; 3) an adaptive system for composing and arranging expert composed music fragments using these two models.
- To test the effectiveness of the AMS, for the intended task of real-time scoring of game-world events in video games, the model was implemented in two games: 1) Zelda Mystery of Solarus (MoS); and 2) StarCraft II. The games were selected for having different mechanics, visual styles, and settings, as well as having open source code or exposed game-states that could be used to model game-world contents.
- Using a Wilcoxon signed-rank nondirectional test, a signif- icant positive increase in rank sum ($p < 0.05$) was observed for both the groups in reporting the perceived effect of music on immersion when AMS was used, independent of the game choice. Furthermore, in no cases did the participants report that the AMS-generated music "greatly" or "somewhat" reduced immersion.
- OSC was used for communicating game states. Read Spreading Activation 6. A. Python's NetworkX library is used for implementing model of spreading activation as a weighted, undirected graph G = (V, E), where V is a set of vertices and E a set of edges.
- RNN is used for the Harmony agent, XCS for the musical agent (Wilson's eXtended classifier), another RNN for the precussive layer.
- The results of this game implementation study suggest that using affect categories within a knowledge activation model as a communicative layer between game-world events and the music composition system can lead to improved immersion and perceived correlation of events with music for gamers, compared to current standard video-game scoring approaches

G. A. C. dos Santos, A. Baffa, J.-P. Briot, B. Feijó, and A. L. Furtado, "An adaptive music generation architecture for games based on the deep learning Transformer mode." arXiv, Sep. 10, 2022. Accessed: Oct. 01, 2023. [Online]. Available: http://arxiv.org/abs/2207.01698

- The paper presents an architecture for generating music for video games based on the Transformer deep learning model. The motivation is to customize the music generation according to the player's preferred musical style. Their architecture is similar to Hutchings and McCormack's, in that it uses both neural network-based generation and an emotion reference mode.
- The system generates various musical layers following the standard layering strategy used in video game music composition. An arousal-valence model of emotions is used to adapt the music generated to the gameplay and player's situation.
- It uses the technique of layering, with the activation of layers controlled by an emotion model, in order to adapt it to the gameplay.
- FUTURE SCOPE OF THIS RESEARCH: The objective is to decouple the generation and the adaptation of musical components from the way to coordinate and orchestrate them, in order to refine the control of music adaptation according to the game play, independently of the music personalization

- Markov models are not used: Markov models are indeed simpler than deep learning/neural networks models, but they face the risk of plagiarism, by recopying too long sequences from the corpus. Some interesting solution is to consider a variable order Markov model and to constrain the generation
- **IMPORTANT POINT REFERRING TO C.PLUT AND P. PASQUIER SURVEY:** ***Generative music is more general and adaptive than pre-composed composed adaptive music, but is also more difficult to control and more computing demanding. As, noted by [31]: "Another reason that generative music may not have received widespread attention in the games industry is that it is often unpredictable and can be difficult to control. The audio director of "No Man's Sky" game, Paul Weir, notes that generative music was used in the game with an acknowledgment that it could produce "worse" music than composed music." [41]. Actually, that distinction between adaptive and generative music is not that clear, as often systems classified as generative are not completely generative and include adaptation components. This is for instance the case of the Adaptive Music System. Note that it is classified as generative in [31], although its very name claims it as adaptive! In fact we need systems to be both generative and adaptive.***
- The architecture of the Adaptive Music System (AMS) is multi-agent and multi-technique, consisting of several agents responsible for different musical aspects - sadness, happiness, threat, anger, tenderness and excitement, whose selection is triggered by current game state (every 30ms, the list of messages received by the Open Sound Control (OSC) is used to update the activation values.
- The harmony role agent generates a chord progression using an RNN trained on a corpus of symbolic chord sequences.
- The melody role agents, one for each instrument, instantiate characteristics of pre-existing melodies using an evolutionary rule system and adapt them to the harmony.
- The rhythm role agent uses another RNN model to generate the rhythm. AMS incorporates a model of six emotions (sadness, happiness, threat, anger, tenderness, and excitement) that modulate the selection among melodies. The selection of melodies is based on the currently highest activated concept or affect, managed by a spreading activation model.
- **As explained in Adaptive Music Composition For Games (Hutchings, McCormack): "Spreading activation models don't require logical structuring of concepts into classes or defining features, making it possible to add content based on context rather than structure. For example, if a player builds a house out of blocks in Minecraft, it does not need to be identified as a house. Instead, its position in the graph could be inferred by time characters spend near it, activities carried out around it, or known objects stored inside it."**
- The authors straightforwardly represent the emotional intensity in music generation, and secondly, they better support layering music. The proposed architecture aims to support collaborative and interactive environments, with a focus on capturing long-term coherence in music using the Transformer deep learning architecture.
- The Transformer architecture was selected for its ability to enforce consistency and structure, by better handling long-term correlations.
- Authors are considering the Skini platform for interactive and collaborative control of musical elements.
- User's client requests a music; 2. The server maps the user feeling through the arousal valence parameters; 3. It fetches, from memory, a song correspondent to the mapped emotion 4. If no associated music has

already being generated, it starts the generation; 5. After the music is fetched, it attaches metadata such as instruments; 6. It delivers the request response with the music to the final user; 7. The memory is refreshed.

- Not sure if the authors tested or measured the validation or response from any conducted user tests to compare their architecture and approach.

A. McDonagh, J. Lemley, R. Cassidy, and P. Corcoran, "Synthesizing Game Audio Using Deep Neural Networks," in 2018 IEEE Games, Entertainment, Media Conference (GEM), Galway: IEEE, Aug. 2018, pp. 1–9. doi: 10.1109/GEM.2018.8516448.

- Audio synthesis using GANs is "examined", and is compared to traditional methods of composing and synthesizing audio, creation of unique sounds using deep learning methods is proposed. They employ this technique to generate samples of in-game audio which could be used to improve in-game immersive experiences.
- This paper explores the idea of using a generative model, trained on raw audio data, to automatically generate convincing in-game sounds that are unique to the game experience.
- RELATED WORK USEFUL TO REFER: Deep learning approaches to audio synthesis, such as WaveNet and WaveRNN, have shown advancements in recent years but are not as sophisticated as imaging tasks. WaveNet is a deep neural network used for generating raw audio waveforms and has been successful in speech and music generation tasks. WaveRNN achieves faster than real-time audio synthesis and demonstrates the feasibility of high fidelity audio generation on low-power mobile CPUs.
- The researchers used the Audio Set dataset, which contains 1.7 million 10-second audio clips labeled on 632 audio event classes. The dataset was manually labeled by humans and is available in CSV files or as 128-dimensional audio features stored as TensorFlow Record files.
- Cross-synthesis is the most comparable method to GANs in terms of output characteristics. It involves taking two signals, a carrier, and a modulator, and performing a magnitude spectrum transform to impose the spectral characteristics of the modulator onto the carrier. However, this approach has some drawbacks, such as the requirement for a broadband carrier and the difficulty of extending it to more than two input signal types.
- Other methods of audio synthesis include augmenting existing sounds with techniques like pitch shifting, time-stretching, or filtering. However, these methods typically apply to a single sound of a single type, unlike GANs, which can combine multiple sounds of different signal types. The researchers compare the performance of their GAN model against these traditional methods in terms of synthesizing unique audio.
- GANs can produce unique audio samples by learning from a training dataset. If the training dataset contains multiple types of audio, GAN outputs can exhibit characteristics of all these types combined in a hybrid manner, similar to cross-synthesis.
- This technique can be extended to generate hybrids of sounds from more than two types by adding sounds of these types in equal or differing proportions in the training dataset.
- The relative proportion of a given type of sound in the dataset can be adjusted to magnify or lessen its effect in the resulting hybrid output. In the experiments conducted, a WaveGAN model was trained on classes of audio from the Audio Set dataset, which contains a significant amount of audio data.

- The researchers aimed to use as close to an equal proportion as possible between classes when multiple classes were included in the training data, for simplicity of demonstration.
- The authors used the WaveGAN experiment tests and methods, and used the steps from the initial experiment to new data sources and a mix of sources. Specifically, they retrieved the 'dog' and 'violin' classes from the AudioSet dataset and used them individually and in combination to train WaveGAN models. The authors used data from multiple classes, without labels, to train a single model, aiming to examine the potential of WaveGAN in synthesizing audio that sounds like a mixture of multiple classes.
- Once a generator is trained, it is possible to learn the reverse mapping, allowing the finding of an input vector that, when given to a GAN, produces a sound closest to a provided sound sample. This technique enables interpolation between two or more sounds in the latent space, similar to its successful use in computer vision tasks. In an experiment, interpolation in the latent space is used to generate a range of sound samples between two chosen sounds, showcasing the potential of this approach.
- Their results from this "mixed" training set resulted in a representation of all the characteristics of their training signals, which upports the idea that GANs can be used to synthesize audio with multi-class characteristics as a desirable trait.
- "Our methods, which employed a single GAN model, per- formed well on multiple datasets. This is an encouraging step as it is common for GANs to perform poorly when a new dataset is used without hyperparameter tuning".
- Basically - GANs can generate a wide range of audio without relying on spectrograms or predefined filters. However, subjective sound quality can be improved and objective assessment is still an open problem. These methods are useful for producing unique audio for games with fantasy or extra-terrestrial themes.

C. Donahue, J. McAuley, and M. Puckette, "Synthesizing audio with generative adversarial networks," CoRR, vol. abs/1802.04208, 2018. [Online]. Available: http://arxiv.org/abs/1802.04208

- WaveGAN.

Prechtl, Anthony (2016). Adaptive Music Generation for Computer Games. PhD thesis The Open University.

- Proposing a new algorithm for adaptive music generation.
- Has a comprehensive questionnaire that determines the "immersivity" of adaptive music vs linear music.

L. J. Paul, "Using Pure Data as a Game Audio Engine," ***New York*****, 2015.**

- Extensive process on using Pure Data as an audio engine in Unity along with examples of use.

- Great resource to understand the implementation of Pd in game engines, and Leonard Paul is a very well known game audio designer and programmer in the field. The School of Video Game Audio has great resources also.

F. Nadeem, "Multi-Modal Reinforcement Learning with Videogame Audio to Learn Sonic Features".

-

Deep Dive: A Framework for Generative Music in Video Games: [Web Article]

-

Everything Merges with the Game: A Generative Music System Embedded in a Videogame Increases Flow: [Web Article]

-

[1]

K. Worrall and T. Collins, "Considerations and Concerns of Professional Game Composers Regarding Artificially Intelligent Music Technology," IEEE Trans. Games, pp. 1–13, 2023, doi: 10.1109/TG.2023.3319085.

-

F. Rumsey, "Generative music, emotions, and realism," J. Audio Eng. Soc., vol. 63, no. 4, 2015.

-

/*D. Herremans and E. Chew, "MorpheuS: Generating Structured Music with Constrained Patterns and Tension," IEEE Trans. Affective Comput., vol. 10, no. 4, pp. 510–523, Oct. 2019, doi: 10.1109/TAFFC.2017.2737984. */

- Worth mentioning?

/* I. Wallis, T. Ingalls, and E. Campana, "COMPUTER-GENERATING EMOTIONAL MUSIC: THE DESIGN OF AN AFFECTIVE MUSIC ALGORITHM," 2008. */

-

D. M. Young, "THE FUTURE OF ADAPTIVE GAME MUSIC: THE CONTINUING EVOLUTION OF DYNAMIC MUSIC SYSTEMS IN VIDEO GAMES," 2013.

-

D. Moffat and J. D. Reiss, “Objective Evaluations of Synthesized Environmental Sounds,” 2018.

-

Z. Xiang and Y. Guo, “Controlling Melody Structures in Automatic Game Soundtrack Compositions With Adversarial Learning Guided Gaussian Mixture Models,” IEEE Trans. Games, vol. 13, no. 2, pp. 193–204, Jun. 2021, doi: 10.1109/TG.2020.3035593.

-

R. D. Gaina and M. Stephenson, “‘Did You Hear That?’ Learning to Play Video Games from Audio Cues.” arXiv, Jun. 11, 2019. Accessed: Oct. 01, 2023. [Online]. Available:http://arxiv.org/abs/1906.04027

-

/* Jean-Luc Dugelay, Atilla Baskurt, and Mohamed Daoudi. 2008. 3D Object Processing: Compression, Indexing and Watermarking. Wiley Publishing. */

- Should I cite this?