Title: Few-Shot Conditional Generation of Music using GANs and/or Diffusion Models

By: Nafis Abeer

ECE faculty supervisor: Brian Kulis

Abstract

This project aims to explore the potential of using Generative Adversarial Networks (GANs) and/or diffusion models for few-shot conditional generation of music. Starting with a pre-trained model for piano music generation, we intend to leverage a small dataset of a different musical instrument for fine-tuning and conditioning. The objective is to then generate realistic and coherent music of the other instrument, such as the violin, without explicitly training a model on a large violin dataset. Successful outcomes of this project would mean establishing a method for domain adaptation in the field of music generation. Part of the challenge will be evaluating the generated music, but we will lay out a few mitigations for this issue. This proposal outlines the problem statement, prior work, approaches, and a plan for implementing and evaluating the project.

Clear and precise problem statement

The generation of music through deep learning models has been an area of active research, and has yielded impressive results for various musical styles and instruments. However, a lot remains to be achieved in this field due to the creative and variable nature of music. Addressing some of these challenges requires a huge amount of data. Data scarcity presents a problem not just for music generation tasks, but for the entire field of data science. This project seeks to overcome the limitations of data scarcity by leveraging pre-trained models on abundant data (piano music) and adapting them to generate music for less-represented instruments (violin) using few-shot learning [1]. Successfully generating music of different instruments will provide variable and realistic data which can then be used to improve upon existing generative models, or be applied towards other machine learning tasks.

Our primary goal is to investigate and develop a method for generating music using Generative Adversarial Networks (GANs) and diffusion models, starting with a model pre-trained for piano music generation. The pre-trained model architectures are suggested by faculty supervisor, Brian Kulis, and they

will have to be trained on available piano music on the SCC in order to be considered our base “pre-trained” models. We will then be fine-tuning these pre-trained models on a small violin dataset and conditioning it to create coherent and realistic violin music. The choice of this other instrument is subject to change.

A significant challenge in this project is evaluating the generated music. Assessing the quality and creativity of music is subjective, and conventional evaluation metrics might not capture the nuances of creative outputs properly. To tackle this issue, we'll explore different evaluation strategies, including a mix of quantitative metrics like objective measures of musical structure, and qualitative assessments, such as human evaluations from listeners.

By successfully generating top-quality music of a target instrument through domain adaptation, this project will contribute to advancing music generation techniques, particularly for underrepresented instruments with limited data availability. Moreover, the methodology we develop in this project will contribute to foundations for domain adaptation in generative models beyond music, opening up new research possibilities in various fields where data scarcity is an obstacle. This broader applicability can inspire further innovation in the world of generative models and deep learning.

Discussion of prior work by others as reported in the relevant literature

Various deep learning models have been proposed and have shown promising results in the field of music generation and few-shot generation. One notable attempt at few-shot music generation is by Liang et al. in their paper titled "Dawson: A domain adaptive few-shot generation framework" [1]. Their work presents “Music Matinee” for generating music in underrepresented domains using only a few examples. It is based on Dawson, which has learned to generate new digits given only four samples in the MNIST dataset. Though their work is promising, our project aims to use different base models, namely WavGAN, DiffWave, or MusicLM. Liang et al. makes use of Meta-learning algorithms, such as MAML, for few-shot generation and we currently do not have clear plans to implement this ourselves. Seeing how they claim to be the first to propose a few-shot music generation framework, we will be referring to their work for guidance.

WaveNet [7], MusicLM [4], and Sashimi [10] represent notable advancements in the field of music generation. These generative models, designed for raw audio synthesis, have showcased high-performance in producing realistic and coherent music. While we proceed with our GAN or diffusion approaches, we acknowledge the achievements of these models and may refer to them later to enhance our own selected method. We are particularly interested in the MusicLM architecture, and we will spend the early days of our research understanding it and deciding if we should proceed with this as our base model.

The Conditional WaveGAN paper [2] shows us how we can potentially adapt a generative model for different tasks while using the WaveGAN model as a baseline. The paper explores the generation of audio samples conditioned on class labels, similar to the way conditional GANs have been used for image synthesis. Although WaveGAN generates random human-recognizable audio samples with relatively good inception scores, the work presented in [2] aims to control the generated audio waveform based on class labels. This approach aligns somewhat closely with our goal of conditioning music to resemble a different instrument, as it demonstrates the potential for controlling generated audio in a more directed manner.

Our project also aims to also employ a model like DiffWave [3], a diffusion model for raw audio synthesis. DiffWave builds upon the principles of WaveNet and addresses some of its limitations by providing a non-autoregressive, flexible, and efficient approach to music generation using diffusion processes. This model is particularly relevant to our project as it offers a diffusion-based counterpart to WaveGAN. However, we might be more interested in a model that builds upon Sashimi [10], an improved version of WaveNet. DiffWave-like models have demonstrated high-quality audio synthesis in both conditional and unconditional settings, making them promising candidates for our domain adaptation task.

For preprocessing music, SoX is a valuable tool that can be utilized in our project. SoX, also known as Sound eXchange, is an open-source command-line utility for converting, processing, and playing audio files [6]. It allows for various operations, such as format conversion, resampling, and applying audio effects, which can be useful in preparing our music dataset for training and evaluation.

Approach(es) to be used in carrying out the project

We will first decide whether to start with WaveGAN, DiffWave, or MusicLM as our base model. We will then train the model on a large piano dataset before applying few-shot learning techniques to adapt it to generate violin music. We will spend our early days deciding if the violin is even our target instrument.

After the initial training, we will apply some kind of few-shot learning method, such as meta-learning, transfer-learning or memory-augmented neural networks, to adapt the model to generate music of the target instrument. Transfer learning leverages the knowledge acquired from a related task to improve the performance on the target task, even with limited training data [8]. In our sample case, the related task is the generation of piano music, and the target task is the generation of violin music. Meta-learning, or learning to learn, is a strategy where the model learns to generalize from a small number of examples by optimizing its internal learning process [9]. By using these few-shot learning techniques, we aim to effectively adapt our chosen model, initially trained on piano music, to generate realistic music of our target instrument. It should be noted that our few-shot learning method may integrate Dual Contrastive Learning [11], which was utilized as sort of a loss function for Few-shot Image Generation.

Evaluating the quality of the generated violin music is essential for this project. To assess the quality, we will employ both objective and subjective measures:

Objective measures:

1. Quantitative metrics: We will use metrics like Inception Score (IS) or Frechet Inception Distance (FID) to evaluate the generated samples' diversity and quality. These metrics have been widely used in evaluating generative models and can be adapted to the music domain.
2. Music theory-based analysis: We can analyze the generated music's coherence and structure using music theory, assessing aspects like harmony, melodic patterns, and rhythmic consistency. This is not a preferred method.

Subjective measures:

1. Human evaluation: We will gather feedback from human evaluators, such as musicians or music enthusiasts, to rate the quality, coherence, and realism of the generated violin music through

surveys or listening tests. This approach is rather unrealistic given our timeline, so we may resort to:

2. A/B testing: Human evaluators are presented with a mix of real and generated violin music samples. Evaluators will be asked to identify the generated samples or rate the samples' quality without knowing their origin, helping gauge how well the generated music blends with real music.

Plan and schedule

Week 1 (May 23 - May 29):

- Familiarize with project requirements and objectives. Review relevant literature.
- Set up the development environment and required tools (SoX, etc.).

Week 2 (May 30 - June 5):

- Research and select a base model (WaveGan, DiffWave, or MusicLM) and target dataset.
- Obtain the pre-trained piano models and prepare the small target instrument dataset by preprocessing it into a compatible format.

Week 3 (June 6 - June 12):

- Start implementing few-shot learning techniques for the selected base model.
- Begin adapting the base model for target instrument music generation using few-shot learning techniques.

Week 4 (June 13 - June 19):

- Continue fine-tuning the base model on the target instrument dataset.
- Develop evaluation strategies, including quantitative metrics (IS, FID).

Week 5 (June 20 - June 26):

- Complete fine-tuning of the base model.
- Refine evaluation strategies, including the development of subjective measures.

Week 6 (June 27 - July 3):

- Conduct A/B testing with human evaluators (might require crowdsourcing).
- Continue fine-tuning if not completed yet. Perform objective evaluation when completed.

Week 7 (July 4 - July 10):

- Analyze evaluation results from objective and subjective measures.
- Identify areas for improvement in the model based on evaluation results.

Week 8 (July 11 - July 17):

- Refine and iterate the model based on identified areas for improvement.
- Continue the fine-tuning and evaluation process.

Week 9 (July 18 - July 24):

- Finalize the model and perform a last round of evaluations.
- Prepare a comprehensive report on the project, including methodology and results.

Week 10 (July 25 - July 31):

- Review the report and incorporate necessary revisions.
- Prepare a presentation to showcase project outcomes and key findings.

Week 11 (August 1 - August 7):

- Present the project to the faculty supervisor.
- Collect feedback and discuss potential future work in music generation and domain adaptation.

Week 12 (August 8 - August 14):

- Address any remaining feedback and finalize project documentation.
- Submit the final report and project deliverables.

Citations:

[1] Liang, W., Liu, Z., & Liu, C. (2020, January 2). *Dawson: A domain adaptive few shot generation framework*. arXiv.org. Retrieved April 18, 2023, from https://arxiv.org/abs/2001.00576

[2] Lee, C. Y., Toffy, A., Jung, G. J., & Han, W.-J. (2018, September 27). *Conditional wavegan*. arXiv.org. Retrieved April 18, 2023, from https://arxiv.org/abs/1809.10636

[3] Kong, Z., Ping, W., Huang, J., Zhao, K., & Catanzaro, B. (2021, March 30). *DiffWave: A versatile diffusion model for audio synthesis*. arXiv.org. Retrieved April 18, 2023, from https://arxiv.org/abs/2009.09761

[4] *Generating Music from text*. MusicLM. (n.d.). Retrieved April 18, 2023, from https://google-research.github.io/seanet/musiclm/examples/

[5] Jthickstun. (n.d.). *Jthickstun/pytorch_musicnet: Pytorch dataset and Jupyter demos for MusicNet*. GitHub. Retrieved April 18, 2023, from https://github.com/jthickstun/pytorch_musicnet

[6] *Sound exchange: Homepage*. SoX. (n.d.). Retrieved April 18, 2023, from https://sox.sourceforge.net

[7] Oord, A. van den, Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., & Kavukcuoglu, K. (2016, September 19). *WaveNet: A generative model for raw audio*. arXiv.org. Retrieved April 18, 2023, from https://arxiv.org/abs/1609.03499

[8] Pan, S. J., & Yang, Q. (2010). A Survey on Transfer Learning. IEEE Transactions on Knowledge and Data Engineering, 22(10), 1345–1359. From https://doi.org/10.1109/TKDE.2009.191

[9] Finn, C., Abbeel, P., & Levine, S. (2017). Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. Proceedings of the 34th International Conference on Machine Learning, 70, 1126–1135. From https://arxiv.org/abs/1703.03400

[10] Goel, K., Gu, A., Donahue, C., & Ré, C. (2022, February 20). *It's raw! audio generation with state-space models*. arXiv.org. Retrieved April 20, 2023, from https://arxiv.org/abs/2202.09729

[11] Zhao, Y., Ding, H., Huang, H., & Cheung, N.-M. (2023, April 15). *A closer look at few-shot image generation*. arXiv.org. Retrieved April 20, 2023, from https://arxiv.org/abs/2205.03805