CS505: Final Report

Christopher Ziko and **Rakin Munim**
Project Github

Abstract

This document is the final report for CS505's Final Project. In it we will discuss our project as of its current state along with some reflections and areas of future work. The goal of this document is to provide an understanding of where we started, what we learned, and what's next. The project was centered around generative music and what the state of the art is.

1 Introduction

1.1 Motivations

Our initial motivation in this project was to figure our how to use AI in music. Humans spend lots of time interacting with music since it is a form of coded information. Music is natural language that helps us convey ideas, emotions, memories, etc. However there exists a barrier in music. Music is generated using instruments, something that takes lots of time to learn. Not everyone is fortunate enough to have the time to learn instruments, and even those who are can only fully master a small amount comparatively. This problem of expert knowledge isn't just present in music. Image and writing are also mediums that require lots of energy to become fluent at. However, the past two years have seen a development of language models that have completely changed what was possible in other fields. For example, DALL-E, an image generation model, is able to generate near in-decipherable, at least at first glance, images from simple prompts. Styles that took artists years to master can be now be learned by models and used to generate new versions in minutes, effectively letting the artist live on in a static form. With DALL-E's example in mind we sought out to find out what was happening in the music space.

2 Milestone Answers

2.1 Goal

The goal of our project was to generate music using NLP models. Our initial goal was to just input an extremely natural language prompts that generate normal music. Normal music is multimodal music that fuses instruments and the human voice. For a concrete example refer to ABBA's Dancing queen. The song has both human vocals and instrumentals. The song also makes use of various instruments. To list a few: Grand Piano, Electric Guitar, etc Saxophone, etc. Most of popular music is similar to this

Back to our goal, what would a simple prompt look like? In our first milestone an example prompt we had was:

'Sing a song that is similar to California Love by Tupac but for Connecticut. Add East Asian Instrumental Bells and whistles throughout the music'

Let's dissect this prompt: It references California Love, a preexisting popular song. This requires a model to have been trained on popular songs. It also says "but for Connecticut". Take a look at the lyrics on Genius and you will see that California love is a song about different aspects of the state of CA. For the model to succeed at fulfilling the conversion to Connecticut prompt, the model must have at least a language component than can generate language similar to that of the original song. Lastly it says "add East Asian ... bells and whistles". This part means the model must have an understanding of music at the granularity of instruments.

If the model we have in mind is successful it would ask a few more follow up questions to understand any specific details about the request then output a song that is high quality, harmonious, preserves enough of the old song, and adds all of the new items requested. It is left to the reader as an exercise to generate a possible output to the prompt

above. Not many readers will be able to do so, or at least that is our misguided assumption. And of those who can it would be very time consuming. Hence this is a perfect entry into how it might be helpful to people.

Unfortunately this goal isn't something we we able to achieve in our current version of the model. Instead we made a pivot and reduced our scope to just the instrumentals of songs. Even within this goal we had to make a few adjustments about the specifics our data. Because we made use of an off the shelf tokenizer, all of our songs translated into only one instrumental class: piano. The label for each song became the artist of the song. This was simply done during the pipeline to reach a full generation pipeline. However our overall architecture will still work with modified labeling systems or perhaps post processing. It will be out of the scope of this paper to write of these matters.

2.2 Method

Our idea to solving this problem was using neural models. We examined the current model space and found a leading model named Anticipatory Music Transformer. This model takes an auto regressive approach to predicting music. The prediction tokens are tokens in the midi space. Because Professor Andrew taught us to not be 'monkey see monkey do' type thinkers in class, we picked our goal is to generate comparable works but using a diffusion based approach.

What are diffusion based models? What are auto regressive models? Diffusion based models work by using noising dynamics. These models take in some input, usually an image, add some noise iteratively and learn the mappings. There are two processes, a forward and a backward and the models job is to learn the backward given the forward. Auto regressive models work in a different way, they learn next token prediction given the previous window of tokens. Afterwords, the model feeds the output into itself until it reaches an end token. Both of these models use transformers under the hood since they are a powerful model.

The AMT also makes use of special type of auto regression. Instead of the model learning how to predict a token based only on future token, the model learn to gain a little bit of information leakage, up to some parameter, delta. This is essentially gives the mechanism the name anticipation, it learns how to predict tokens based on the the past tokens, and the window into the future, the anticipated tokens. It can be summed up as controlled information leakage.

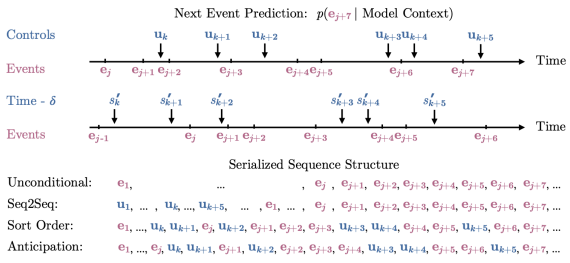

Figure 1: The Anticipation Sequence Structure Credit:AMT

Anticipation actually is great idea, so why aren't normal models trained this way? To some degree masked pertaining does this. Anyways this doesn't make sense in language based models like GPT. In language generation we usually want models to respond to some prompt that should grow linearly. We may not have the tokens required to respond in our input. But it is AMT's goal to actually generate melodic accompaniments. These are just songs that complement some underlying baseline song. In this process the model has the song as a whole must include the baseline tokens hence looking at them is no problem.

For the diffusion based approach the features of anticipation and accompaniment generation is not really preserved. However we diffusion models allow us to gain a feature that AMT doesn't come with, class based sampling. At the heart our model maps images to labels. The images are piano rolls generated baby an open source tokenizer known as midipic. The labels are artist name. The AMT requires input tokens, whereas our model requires just the class.

Diffusion based systems have guidance mechanisms that allow images to be generated using labels. It is our hypothesis that his guidance system can be engineered to include multi class guidance and fuse different classes together. We defend the last sentence by saying, models like DALLE are able to mix styles of genres of art, and our approach uses their system.

3 Experiments

3.1 Data

To train our model we made use of the Lakh Midi dataset. This is a large dataset compiled by Colin

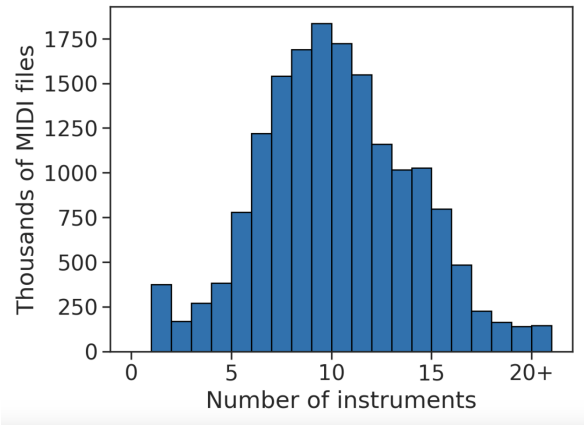

Figure 2: Instrument Distribution

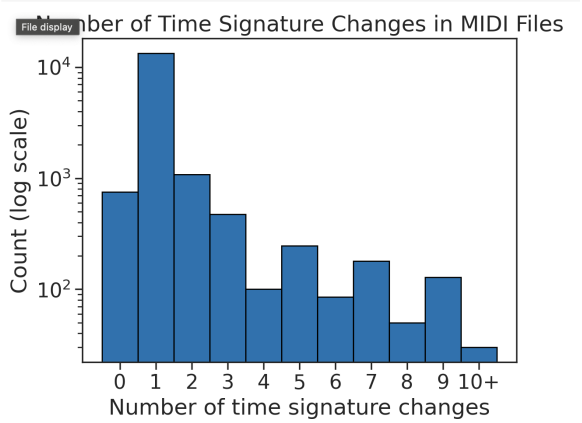

Figure 3: Time Signature Changes in the Clean Midi Dataset

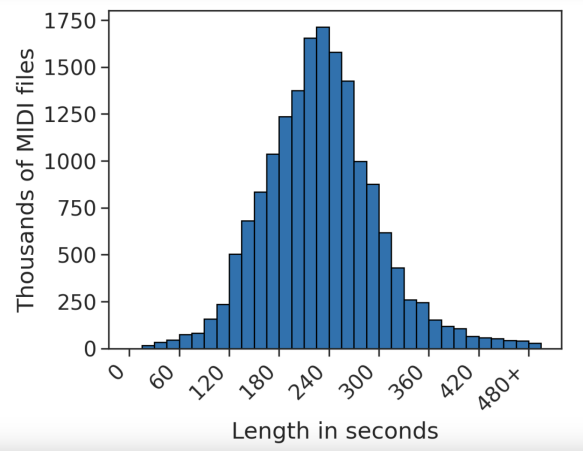

Figure 4: Instrument Distribution

Raffel of midi files. Midi files are digital scores, that embedded all information needed to play back the music given you have a synthesizer. Midi files are just what machines see, but they can also be turned into regular scores with the staff in grade school band class. Midi files essentially encode music as a sequence of ONs and OFFs, along with velocity, pitch, etc. They occur over a time domain. Midi files are somewhat of a nuisance to work with. We ran into various problems that had musical information manipulation Particularly when we were using python libraries to extract the information we found there are two types of midi files. Between these two files there are different methods of encoding music. There can be multiple tracks, kind of similar to a musical score. However one of the file types puts all instruments into one track, whereas the other puts them into different tracks. Hence we implemented code that quickly combine music from multiple tracks into a single track. This obviously was adhoc and had information manipulation which impacts the quality of our results.

The Lakh Midi dataset has roughly 100K midi files of various files. It is actually an exceptionally well curated dataset that has been well documented. See figures 2,3,and 4 to give you a taste of the music corpus. We used a smaller subset that was cleaned and deduplicated. It was distributed as a part of the original dataset.

Overall the songs are high quality and the corpus is definitely large enough to support what we do. In fact during the prepossessing we had to use a smaller subset than than clean midi subset, and our model still was able to learn. Lastly, I'll give you an example of a song which was in the corpus: Dancing Queen by ABBA.

A note on data redundancy in Lakh: Usually models augment data to make bigger samples, but we noticed that the corpus actually comes with multiple versions of some song. For Dancing Queen there way more than 10 versions. Each of these were slightly varied. The effects of data redundancy needs to be studied maybe it helped the model, or hurt it. There might also be biases for different artists in case not all of them have multiple songs

3.2 Diffusion Model

In our code we use the open source and weights model DiT. It is an open source model that was released by Facebook research. We train the model by fine tuning using pretraining weights. Our image sizes are 256 by 256. There is enough fidelity here to encode music that sounds realistic, and this was a default used by the tokenizer itself. Additionally there are various sizes of the underlying image model. There a N, L and XL sizes. We found the optimal point model to being L. We trained on var-

1. Bbp - Prism
    - image-size 256
    - model : DiT-L/4
    - epochs : 32
    - steps : 88700
    - GPUS utilized: 4x V100-SXM2-16GB = 64GB
    - Total training hours from start to finish: 3.93 hours
        - Start Time: 34m2024-12-06 09:18:50
        - End Time: 34m2024-12-06 13:14:57
        - Difference : 3.93 hours

Figure 5: Training metadata for our model

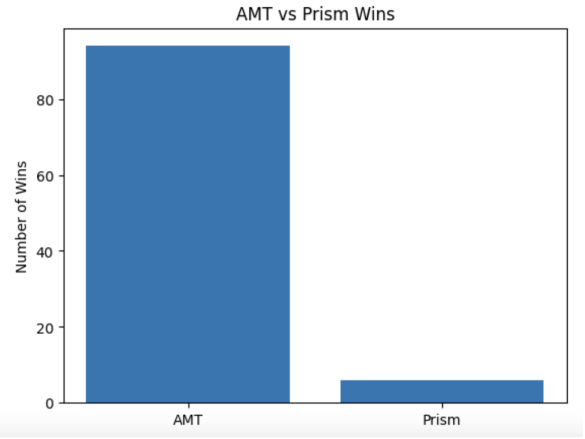

Figure 6: Wins Hist

ious hyper parameter for the model, using different learning rates, patch sizes, and training steps. A learning rate of .0001 as too slow and .001 to large. Additionally training for too many steps on some models led to learning rate to drop for a little bit until eventual spiraling out. Lastly we would like to say that training this model require access to high quality GPU. Prism was trained in 4 GPUs with 16GB each. When we attempted to train the model on a single GPU we quickly were met with CUDA out of ram issues. This highlights a limitation of our model since doing inference will require access to expensive machines. Our final model prism trained for about 4 hours.

3.3 Performance Evaluation

Quantitative Evaluation: Our method of evaluation was to use cosine similarity. We got the idea from a blog post. Particularly, fed our model into a CNN which extracted a latent feature. After this latentization we compare cosine similarities of some piano roll from the corpus and each of the two generative models' outputs. We do this for about 100 runs, sampling randomly from each set. We report the average of the cosine similarity for each model. Our model had a cosine average cosine similarity of .38 whereas our baseline had about .66. We also keep track of wins for each comparison, and find that in this metric we severely are outperformed by AMT. We believe this originates from two biases in our evaluation system. Firstly, the CNN we used was trained on ImageNet making it not the best option musical datasets, rather it would have been a much better idea to train the CNN on the corpus piano rolls. This was actually pointed out by one of the viewers during our poster presentation. Secondly because we didn't do any post processing after generating the diffusion image there was lots of small perturbations, we believe this is the main reason the cosine similarities were low for our model. Qualitative Evaluation: Aside from the quantitative metric, we listened to the music that was created by our model and found it to be somewhat harmonious. It definitely learns underlying patterns about harmony. On some songs it knows how to dwindle down toward the end. In this paper we have included some of the piano rolls that were generated. If you look at figure 7, you will see how our model learns underlying patterns like a dark background, and it learns sequential patters. However there is a lot of noise, which makes sense considering diffusion models take a noise based approach. If you look at the baseline AMT, you can see there is no noise present, but the overall patterns are quite similar to what we have. Post processing can give us an easy boost in generative quality.

3.4 Future Work

Let us now explore possible future work. We would like to propose three new "parameters" that we need to focus on during the next iteration. Firstly, we need to create better labels/indexes on the data and have a better understanding of the distributions of what is actually going into the model. Our first iteration made use of lots of adhoc methods in pulling data to train from. We may have oversampled for some artists and undersampled for others. Additionally, we should make use of labels about the subjective experiences of of the song, e.g. dramatic, soft, etc. If we wish to really build a model that can take natural language and make granular changes we will need to utilize new labels.

Secondly we need to study tokenizatoin techniques. Now that we have a model pipleline work-

Figure 7: Prism Sample(Ours)

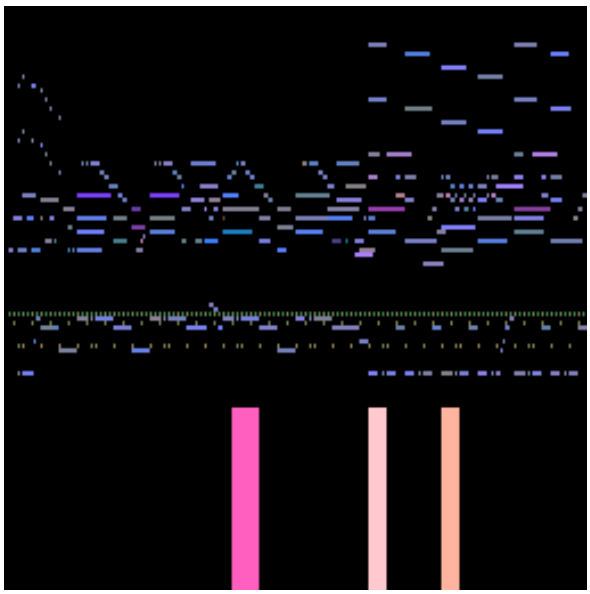

Figure 8: Corpus Sample

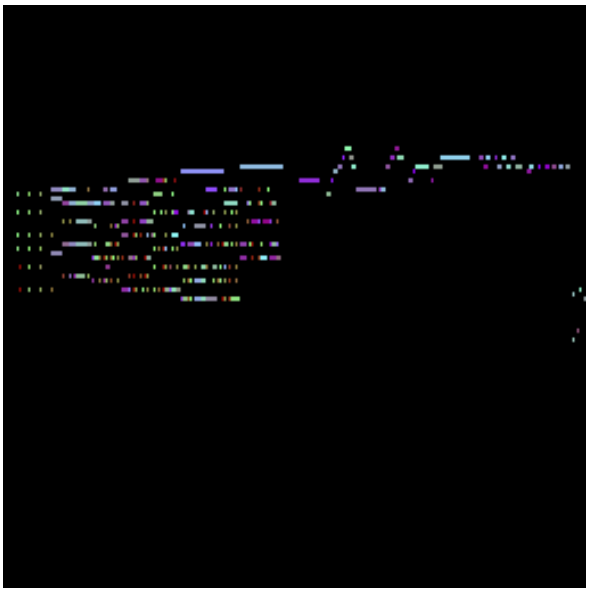

Figure 9: AMT Sample: Baseline

ing, we can start doing small iterative changes in the tokenization system and test it immediately. Our new tokenization method should formally preseve all instrumental data when converting to images.

Lastly, we would like to explore a more multimodal model that integrates lyrics. Spoken words play an important role in connecting to music and it would be worthy to create a model that knows when to "deploy" words in a song based on the instrumentals. Again doing so would allow us to achieve our original goal.

4 Discussion

4.0.1 Is the use of generative AI in art ethical?

The growing use of generative AI in art raises complex questions about authorship, creativity, and the role of technology in the creative process. Many artists are grappling with the potential of AI to reshape how their art is made, and there is no single viewpoint that prevails. Some artists, like Matt Saunders, see AI as a valuable tool for challenging conventions and expanding artistic possibilities. He believes that the role of the artist remains central, even when working with AI, as the artist brings their vision and intentionality to the work. He sees AI not as replacing human creativity, but as offering new ways to think about and interact with art.

However, not all artists are as comfortable with the idea of AI-generated art. The Artist Rights Alliance (ARA), which includes famous artists like Pearl Jam, Billie Eilish, and Stevie Wonder, has ex-

pressed strong concerns about AI in music. They argue that AI-generated music undermines the value of human creativity and devalues the work of songwriters and performers. The ARA has called for protections against AI tools that replace musicians, ensuring that artists are compensated fairly and their rights are respected. From the perspective of many musicians, especially lesser known ones, the livelihood and integrity of their craft are at risk with the rise of AI.

From my perspective, using generative AI in art is not inherently wrong, provided that the artist is transparent about it. If the public is aware that a work was generated with the assistance of AI, then there is room for this technology to be integrated into the creative process without undermining the artist's integrity. I am, however, wary of AI's use in visual art. The authenticity and emotional depth can feel diminished if AI is used in the process of making it. The "hand-drawn" elements of visual art are the primary medium of expression for these artists. On the other hand, I see more room for AI in music. While I am comfortable with AI being used as a tool for composition, I still believe that human creativity must remain at the core of the song's creation. Music, like visual art, is a deeply personal form of expression, but it also allows for a more collaborative relationship between human creators and technology. Instrumentation has evolved greatly over the last century, so in a sense, AI can be seen as the next iteration of that.

Ultimately, I believe that as long as AI is used to augment rather than replace human creativity, it can enrich the artistic process. However, the preservation of human creativity is essential. Without it, the arts risk becoming hollow, as they are, at their core, a reflection of human experiences, emotions, and perspectives. Artists like the ones part of the ARA and Matt Saunders remind us that while technology can be a tool for innovation, it should not erase the unique qualities that make art meaningful for us.

4.1 Reflection: Christopher

This project provided an exciting yet challenging opportunity to delve into music generation. I enjoy listening to music on a daily basis, so I was emphatic at the opportunity to use my technical knowledge to create something that could generate music. It also provided a contrast to the text-based work that was being done in the homework assignments, while still falling under the umbrella of NLP. As the project progressed, however, I quickly realized how complex it is to make a model understand the nuances of music. The amount of attention to detail needed to make a model properly generate music, is immense, even more so than I originally thought.

The cosine loss comparison with the AMT was particularly revealing. I had hoped the diffusion model would show similar results, but it quickly became apparent that AMT had a better grasp of musical structure, having longer notes with very little noise. This was a humbling reminder of how much the choice of model architecture can influence results. That being said, the diffusion model we created was promising, and even though there was significant noise in the final project, the fact that it even sounded like music was a great personal accomplishment. It felt like we were on the cusp of something great, but time was not on our side, as well as our lack of experience in music generation.

Looking forward, I would like to experiment more with hybrid models that combine the best of both the diffusion model and AMT. One of the major regrets of this project is that we couldn't quite get our diffusion model to sync with anticipation. I also think adding diversity to the dataset, like incorporating different genres or styles, could push the model further in terms of creativity and musical expression. This project not only expanded my understanding of music generation but also gave me an appreciation for the subtle challenges involved in creating something artistic in NLP. It was a great experience overall, and I'm excited to continue exploring this space beyond this course.

4.2 Reflection: Rakin

Overall, this project and course were a pleasure to take and be a part of. Being able to train a diffusion model and work with musical data were not things I imagined myself doing when I initially started this class. The results of this project, with my small knowledge of music, tells me that how we listen to music and what we listen to will change. I hope to continue this project even after this class ends with a few of my friends and peers. This project taught me about some of the difficulties of research and things not always going the way you want them to. Thanks to this project I got to explore many new topics and ideas and I also learned a lot about myself. Overall this experiment was a

success. Onto the next one and hopefully well see a continuation here. ;)

5 Conclusion

TLDR: We fine tune a diffusion model on images that were created as representations of songs. Our pipeline works, although our model output is quite noisy. However the results are very promising and this work lays a foundation for creating a more general model.

5.0.1 Acknowledgemnets

We would like to thank Professor Wood, Can, and Akshat for teaching us what was necessary for making this project happen. Also thanks to everyone else not mentioned, within and outside of this class that helped us get to this final output. Research like this are a result of many people guidance and work.

References

[1] Colin Raffel. *Learning-Based Methods for Comparing Sequences, with Applications to Audio-to-MIDI Alignment and Matching*. PhD Thesis, Columbia University, 2016. https://colinraffel.com/projects/lmd/.

[2] Matt Saunders. *Art, Artificial Intelligence, and Authorship*. The Harvard Gazette, August 2023. https://news.harvard.edu/gazette/story/2023/08/is-art-generated-by-artificial-intelligence-real-art/.

[3] Artist Rights Alliance. *Artist Rights Alliance calls for tech and AI developers not to devalue music*. NYS Music, April 18, 2024. https://nysmusic.com/2024/04/18/artist-rights-alliance-calls-for-tech-and-ai-developers-not-to-devalue-music/.

[4] Peebles, Bill. *The Diffusion Transformer Model by Facebook Research*. https://github.com/facebookresearch/DiT/tree/main.

[5] Jean, Louis. *MidiPic, an midi-to-image tokenizer*. https://github.com/Bycob/midipic.

[6] Okonji, Onyeka. *Cosine similarity — measuring similarity between multiple images*. https://onyekaokonji.medium.com/cosine-similarity-measuring-similarity-between-multiple-images-f289aaf40c2b.