Universidade Estadual de Campinas
Faculdade de Engenharia Elétrica e de Computação
IA376L - Deep Learning Aplicado a Síntese de Sinais
Profª. Drª. Paula Dornhofer Paro Costa

Estudo de caso: Geração de áudio musical através de GANs, LSTM e Transformers.

Alunos: Gabriel Santos Martins Dias. RA: 172441.
Gleyson Roberto do Nascimento. RA: 043801.
Patrick Carvalho Tavares R. Ferreira. RA: 175480.

Julho/2022

Contextualização

- Grande parte do trabalho em modelos generativos profundos para áudio tende a se concentrar na síntese de fala.

- Geração de áudio sintético para música é uma tarefa desafiadora para Machine Learning, pois a audição humana é acurada, tanto no sentido de coerência global, quanto na percepção de formas de onda em escala fina. A subjetividade atrelada à avaliação de modelos generativos trata-se de um problema difícil.

- Ainda não existe um consenso na comunidade científica sobre a melhor forma de avaliação nesse contexto.

Objetivo

- O objetivo do nosso projeto é realizar a síntese e avaliação de áudio voltado para música.
- Para isso, tentamos reproduzir os resultados encontrados no artigo GANSynth, além de termos realizado um estudo de caso acerca de outras possibilidades de arquiteturas para geração de áudio, como LSTM e Transformer.
- Como métricas avaliativas quantitativas foram utilizados o KS Test, considerando que o áudio sintetizado também pode ser considerado um dado tabular, e, também, os resultados de análise sensorial por parte da turma de IA376L. Como métricas avaliativas qualitativas foram utilizadas análises gráficas (gráficos log-log e histograma) e Music Information Retrieval (cromagrama, espectrograma, STFT, etc).

Motivação para o uso da GANSynth

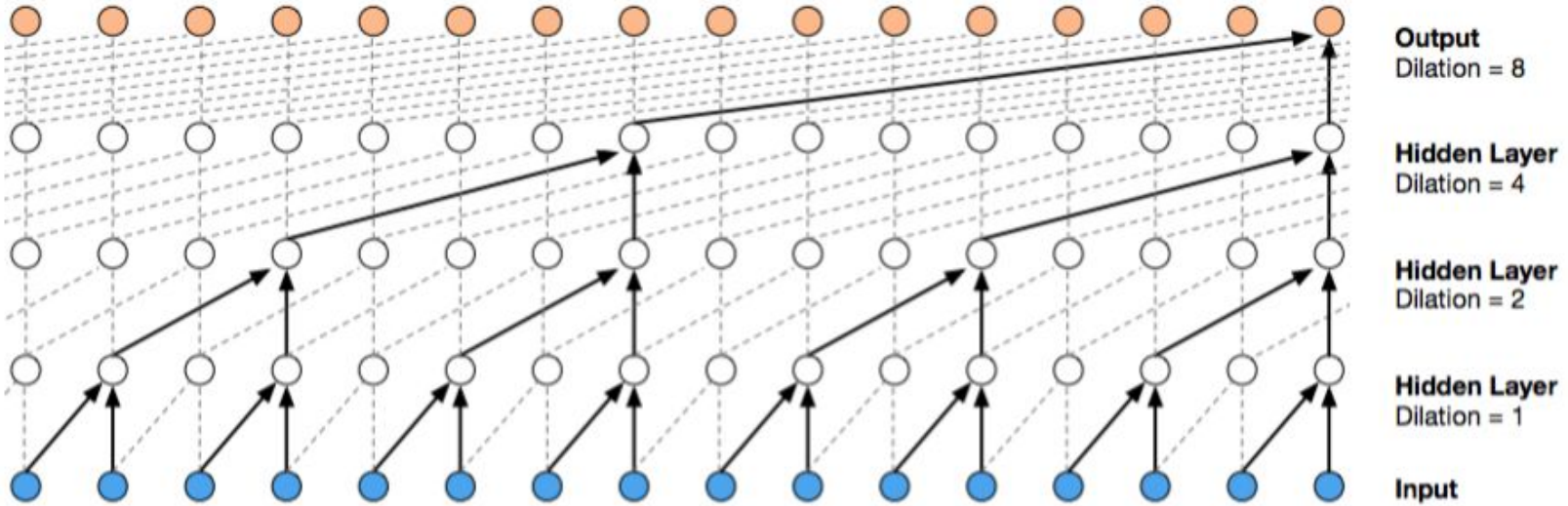
- As GANs desbloquearam transformações de domínio intrigantes para imagens, cultivando a expectativa de que poderia ter resultado satisfatório análogo à áudio.

- No entanto, tentativas de adaptar arquiteturas de GANs aplicadas à imagem para gerar formas de onda de maneira direta não atingem um bom nível de fidelidade perceptual.

- A arquitetura GANSynth desenvolvida pela Google AI para geração de áudio consegue criar sons de instrumentos musicais com alta fidelidade.

Descrição do problema

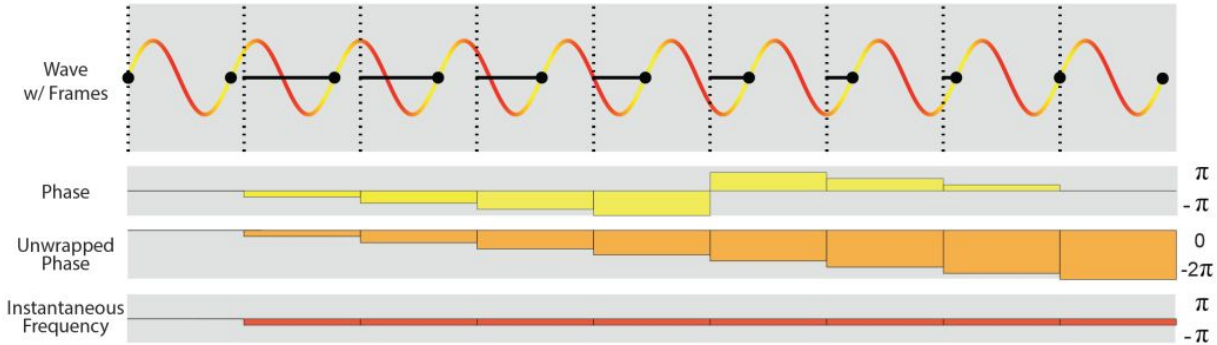
Grandes avanços no estado da arte de síntese de áudio foram iniciados quase exclusivamente por modelos autorregressivos, como WaveNet. No entanto, essa rede perde em termos de coerência global do áudio gerado, além de ter baixa taxa de amostragem, devido ao processo iterativo utilizado.



Fonte: Ref. Bibliográfica [1]

Descrição do problema

- Mesmo redes neurais com forte coerência local, como as redes convolucionais, têm dificuldade em realizar a modelagem de áudio, já que as múltiplas frequências que compõem as amostras não coincidem com o stride utilizado nestas camadas, gerando batimento que aumenta o erro de reprodução em fase, conforme estendemos a geração.
- Este é um desafio para uma rede de síntese, pois ela deve aprender todas as combinações apropriadas de frequência/fase e ativá-las na combinação certa para produzir uma forma de onda coerente.



Fonte: Ref. Bibliográfica [1]

NSynth dataset

- Pesquisadores de GANs fizeram um rápido progresso na modelagem de imagens avaliando modelos em conjuntos de dados focados em graus de liberdade limitados, e gradualmente, avançando para domínios menos restritos.

- O conjunto de dados NSynth foi introduzido com motivação semelhante para áudio. Em vez de conter todos os tipos de áudio, o NSynth consiste apenas em notas individuais de instrumentos musicais em uma variedade de tons, timbres e volumes.

- NSynth contém 300.000 notas musicais de 1.000 instrumentos diferentes alinhados e gravados isoladamente. Cada amostra tem quatro segundos de duração e é amostrada em 16kHz, dando 64.000 dimensões.

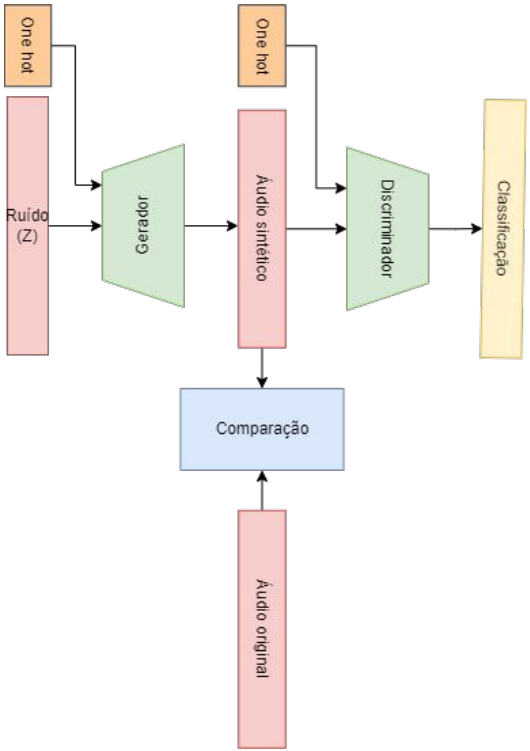
- No estudo da GANSynth, o dataset foi restringido ao subconjunto de instrumentos acústicos, pois estes timbres são mais propensos a soar naturais ao ouvido humano. Isso deixou 70.379 exemplos de instrumentos que são principalmente cordas, sopros, metais e mallets.

Metodologia GAN

Diferentes arquiteturas podem ocupar o lugar do gerador e do discriminador, mas o modelo geral é apresentado na figura a seguir. Este tipo de estrutura é classificado como uma CGAN (as GANs cuja classe de saída é controlável).

Temos como entrada um tensor de ruído cuja rede geradora mapeia um hiperespaço de representação de áudios em tempo de treinamento, além de classes de timbres e notas em one-hot encoding, as quais definem o tipo de saída desejada.

Um áudio de mesma categoria (timbre e nota) é inserido para o discriminador posteriormente, o qual serve de comparação para a rede discriminadora. Seu objetivo é classificar o áudio real como verídico, enquanto o áudio sintético deve ser dado como falso.



Fonte: Ref. Bibliográfica [1]

Metodologia GAN

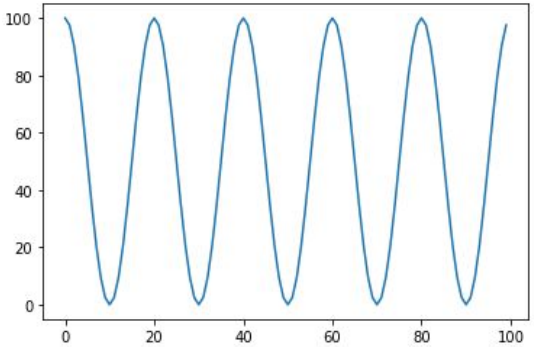
Aprender a reconhecer um áudio como falso é uma tarefa consideravelmente mais simples do que aprender a gerar um áudio de boa qualidade. Essa discrepância de dificuldades entre gerador e discriminador é motivo pelo qual muitos projetistas optam por uma estratégia baseada em deixar o discriminador aprender ao longo dos primeiros mini-batches e congelar seus pesos em seguida, permitindo que o gerador aprenda o mapeamento.

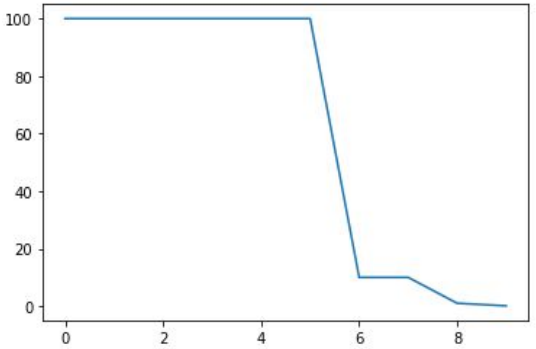
Fazemos isso através de dois escalonadores de Learning Rate, um para o gerador, e outro para o discriminador.

Metodologia GAN

O discriminador utiliza o cosine annealing como escalonador, de forma que seu step de gradiente varie entre valores adequados ao treinamento e valores tão pequenos que praticamente o impeçam de variar, de forma a aguardar o desenvolvimento do gerador neste meio tempo.

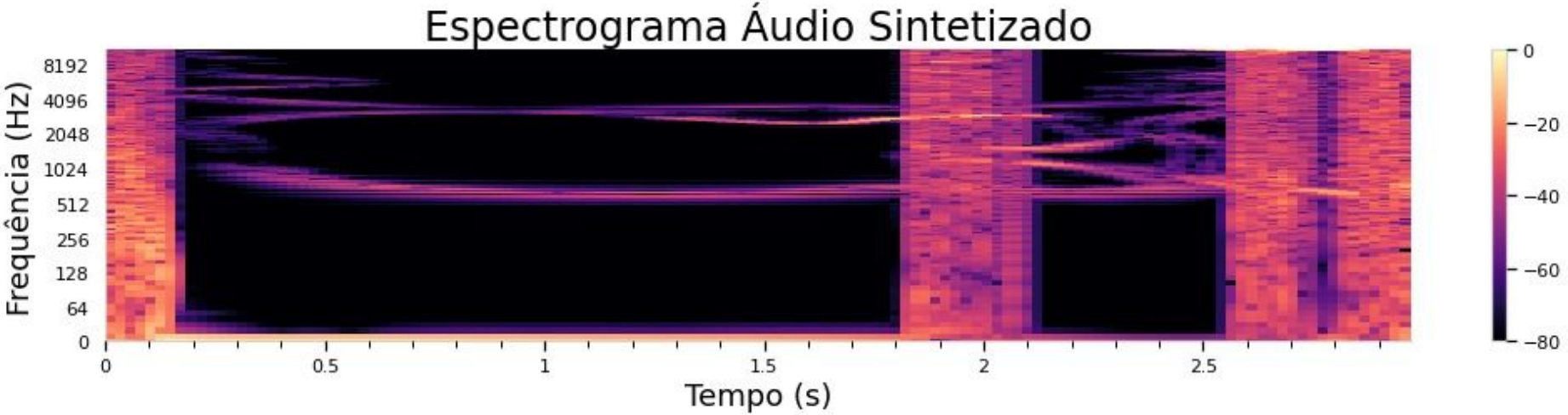
O escalonador do gerador, por outro lado, é baseado em MultiStep, começando com um valor elevado, que favorece a fuga de mínimos locais ruins, e decaindo para valores menores, que proporcionam ajuste fino de parâmetros.





Resultado GAN

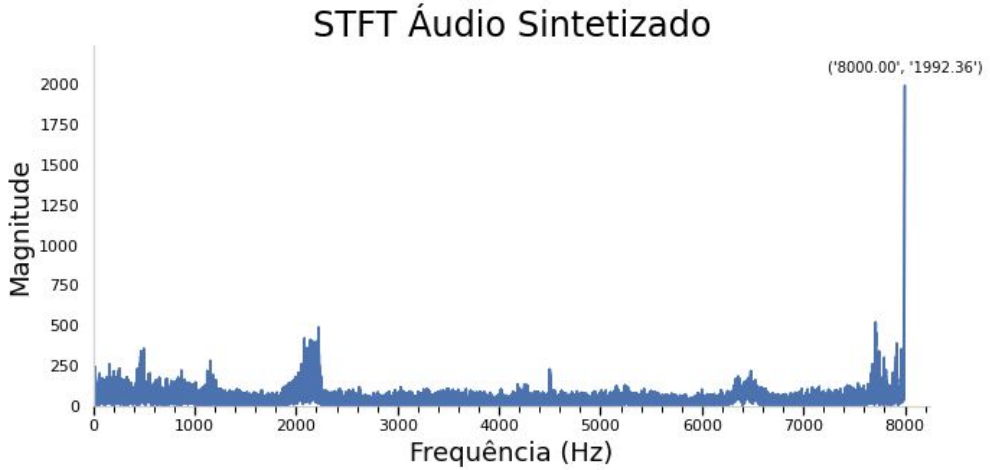
- Após o treinamento das GANs, os resultados da geração de áudio são analisados sob o ponto de vista de espectrograma;
- Nota-se que a rede tenta gerar faixas coerentes de áudio, mas ainda ocorrem momentos de ruído intenso no gráfico, caracterizados pelas faixas verticais.



Fonte: Notebook de Python próprio desenvolvido para avaliação.

Resultado GAN

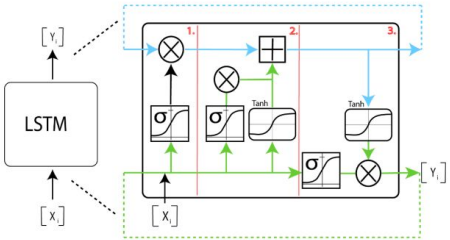
Um exemplo de áudio gerado pela rede é exibido abaixo:

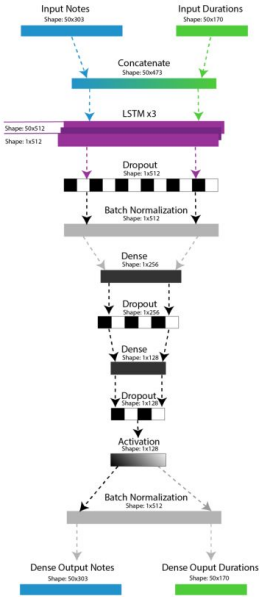


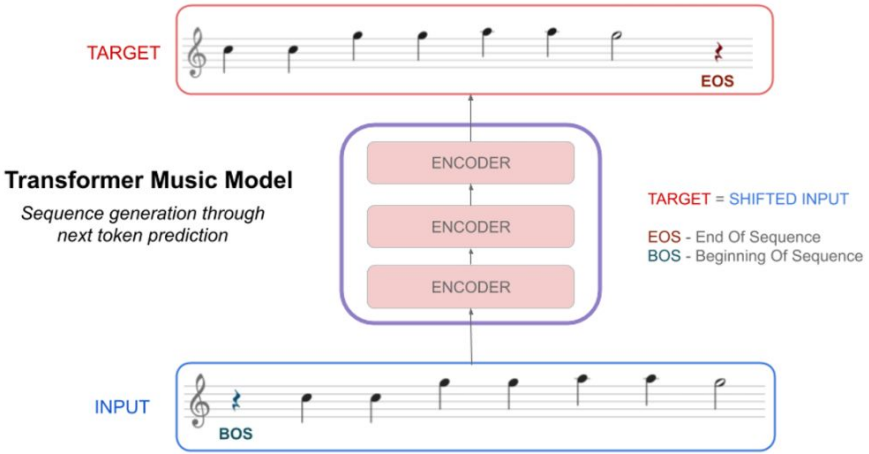
Mais áudios podem ser verificados no notebook no repositório.

Metodologia LSTM e Transformer

Considerando a dificuldade de treinamento e implementação da GAN layer by layer, consideramos a utilização de redes pré-treinadas sendo elas:







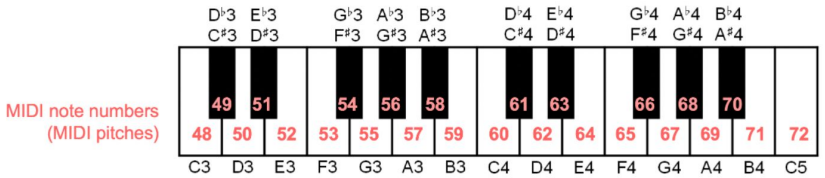
Fontes: Referências Bibliográficas [2] e [3].

Metodologia LSTM e Transformer

Além do uso das arquiteturas pré-treinadas, um ponto fundamental mudança de paradigma neste processo é que os dados de treinamento, normalmente arquivos MIDI, são transformados em tokens de música, numa estrutura denominada **note sequence**:

Arquivo MIDI

index	note_name	start_time	duration	velocity	tempo
0	C4	31.0	0.6	100	168.0
1	E-4	13.4	0.6	100	168.0
2	E-4	25.8	0.5	100	168.0
3	C4	25.7	0.5	100	168.0
4	G#4	29.4	0.6	100	168.0
5	G4	12.9	0.5	100	168.0
6	C4	31.0	0.5	100	168.0
7	F4	27.3	1.5	100	168.0



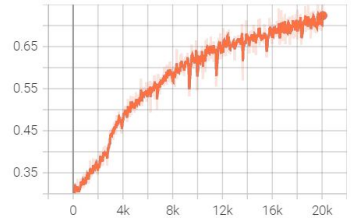
Note sequence

index	note_name	start_time	duration	velocity	tempo
0	C4	31.0	0.6	100	168.0

Fonte: Referência Bibliográfica [4].

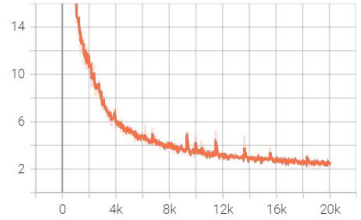
Resultados Treinamento LSTM

metrics/accuracy
tag: metrics/accuracy



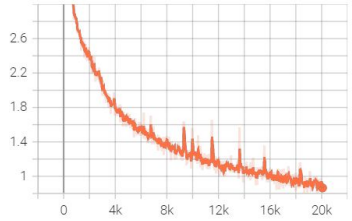
Name	Smoothed	Value	Step
logdir/run1/train	0.7237	0.7566	20.07k
run1/train	0.7237	0.7566	20.07k

metrics/perplexity
tag: metrics/perplexity



Name	Smoothed	Value	Step
logdir/run1/train	2.384	2.157	20.07k
run1/train	2.384	2.157	20.07k

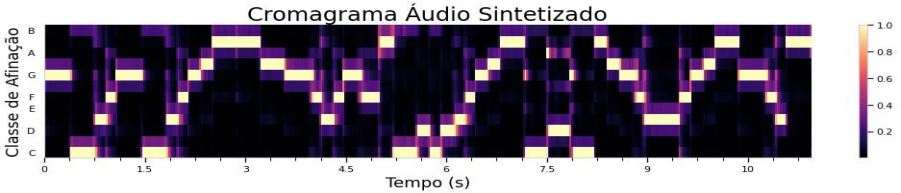
loss
tag: loss



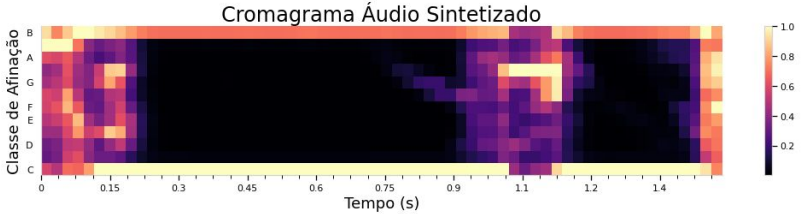
Name	Smoothed	Value	Step
logdir/run1/train	0.8654	0.7688	20.07k
run1/train	0.8654	0.7688	20.07k

Definições importantes

Cromagrama: Representação das frequências existentes no áudio em termos de notas musicais, sendo uma ferramenta de MIR para verificar se uma **melodia** segue o campo harmônico, isto é, visualmente apresenta padrão geométrico e cíclico.



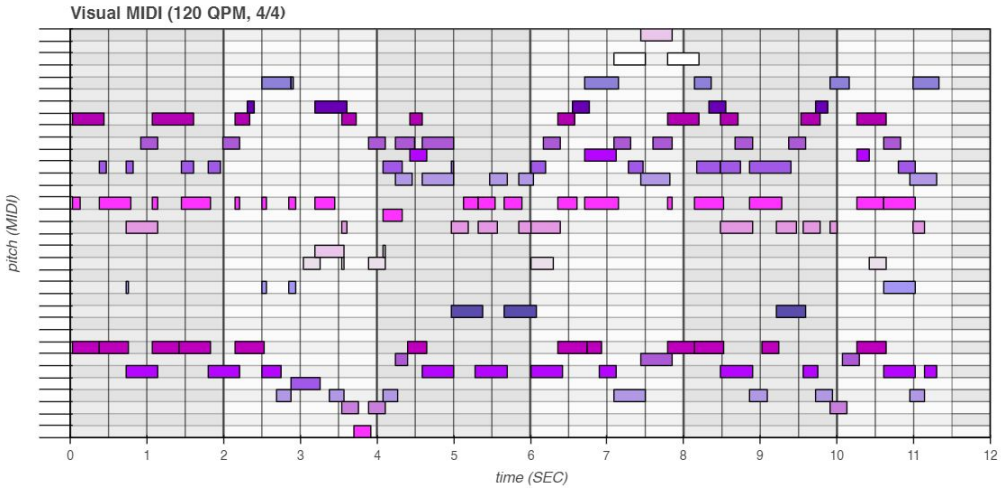
Exemplo de melodia que segue o campo harmônico.



Exemplo de áudio que representa ruído e sem definição de campo harmônico.

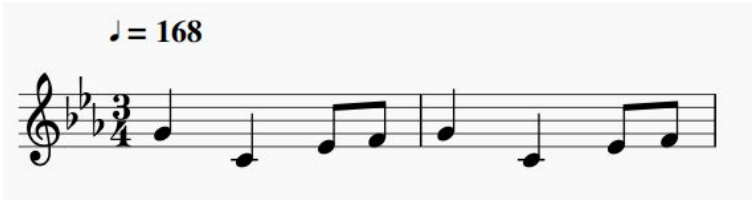
Definições importantes

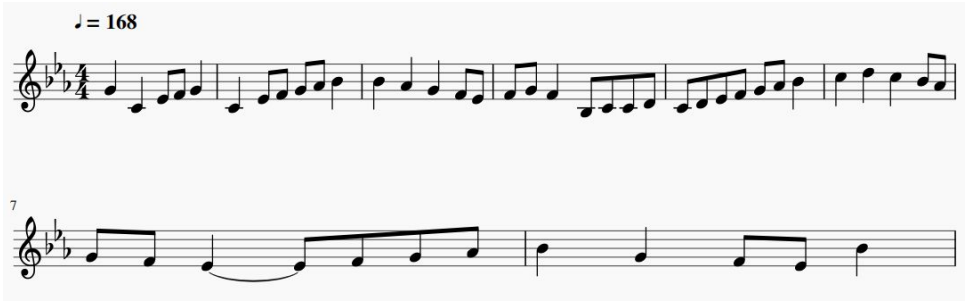
Mapa de MIDI: a **harmonia** representa uma composição musical com mais de duas notas sendo tocadas simultaneamente, desta forma, o cromagrama não é adequado para visualizar se segue o campo harmônico, desta forma, o Mapa de MIDI visualmente apresenta se há padrão geométrico e cíclico.

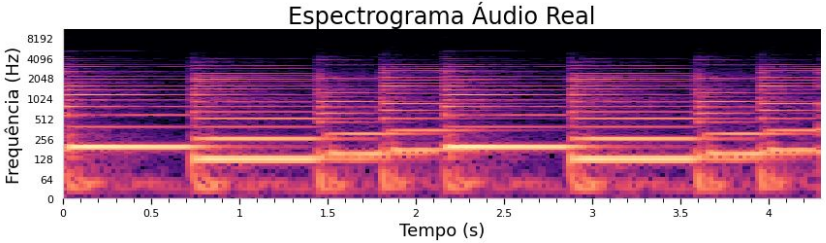


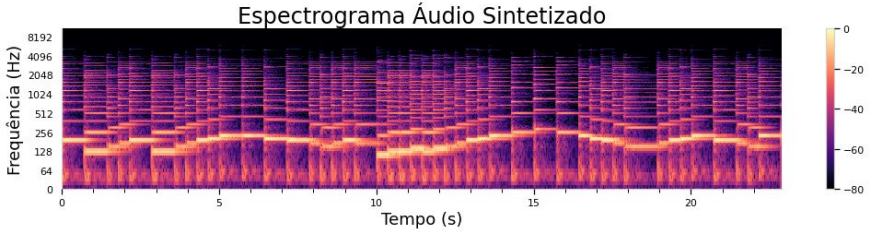
Exemplo de harmonia que segue o campo harmônico.

Áudio Resultante LSTM

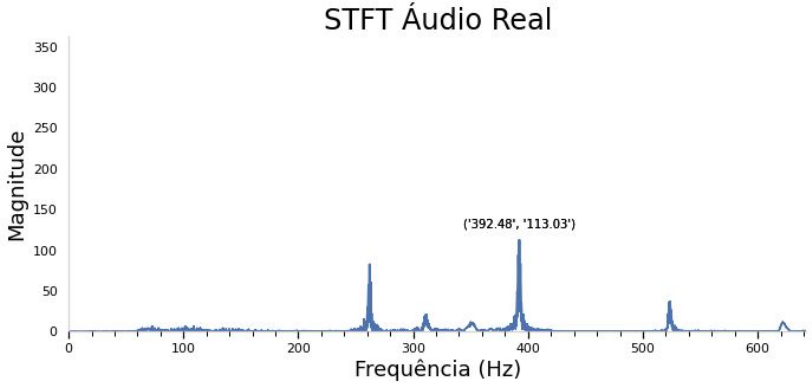


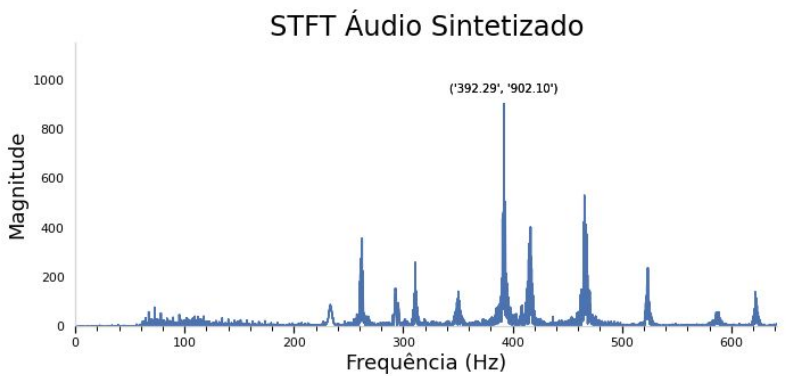


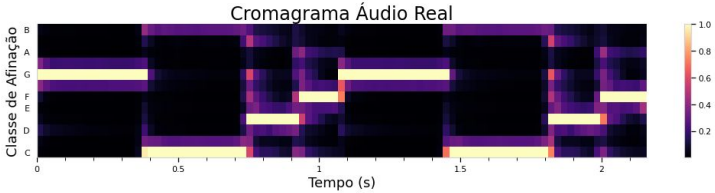


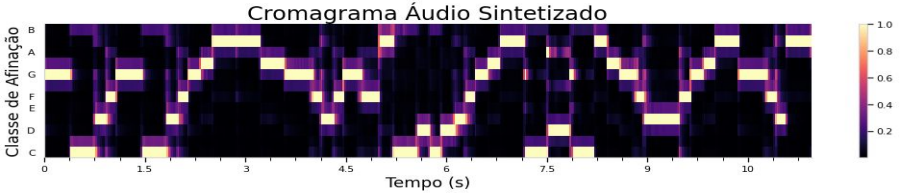


Áudio Resultante LSTM

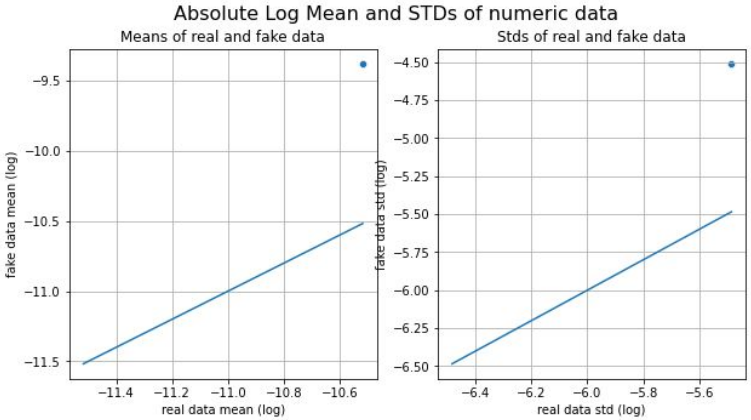


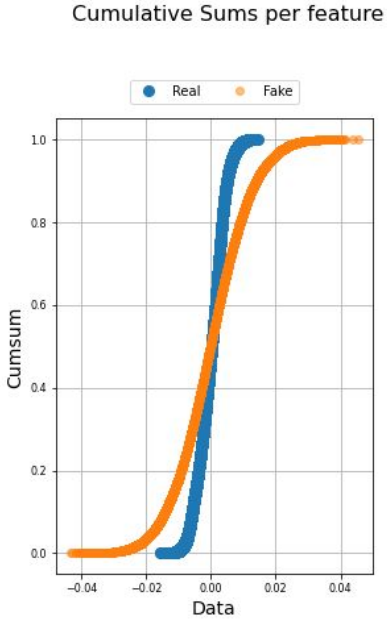


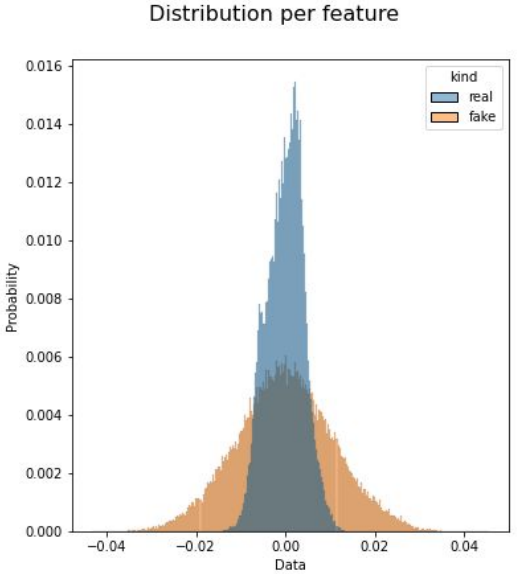




Áudio Resultante LSTM



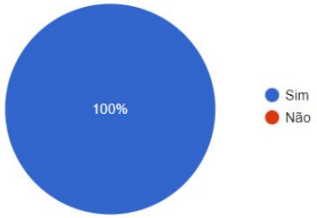




KS Test = 0,7780642732875434

Áudio Resultante LSTM

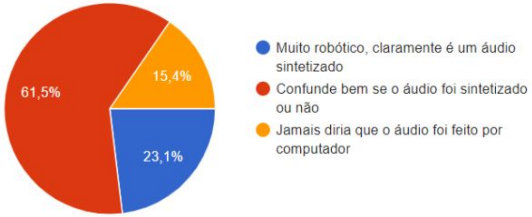
Você considera este áudio uma música?



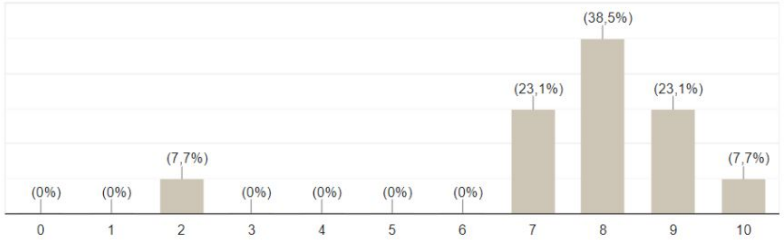
Você considera que este áudio

Sobre a composição e o áudio de input

Sobre a síntese



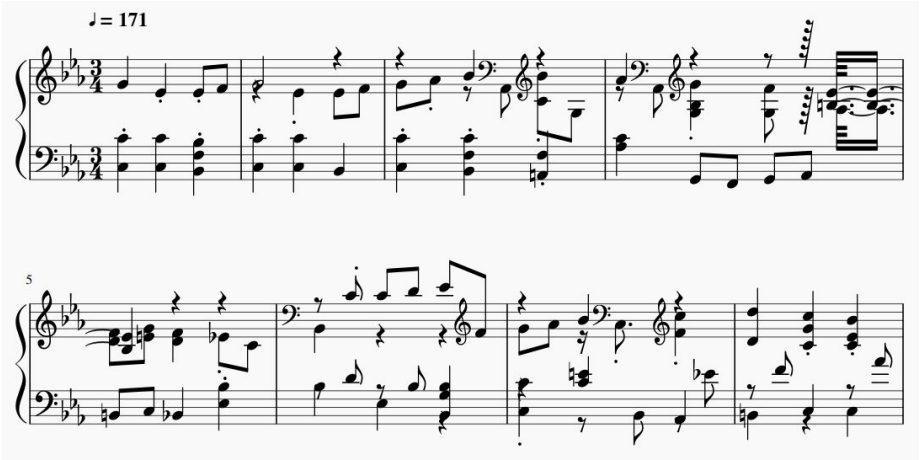
Se você tivesse que dar uma nota para este áudio, qual seria?

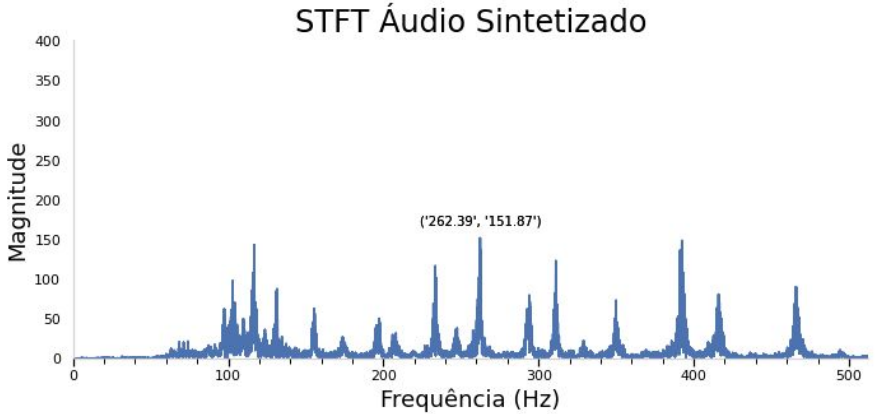


Nota média: 7,7

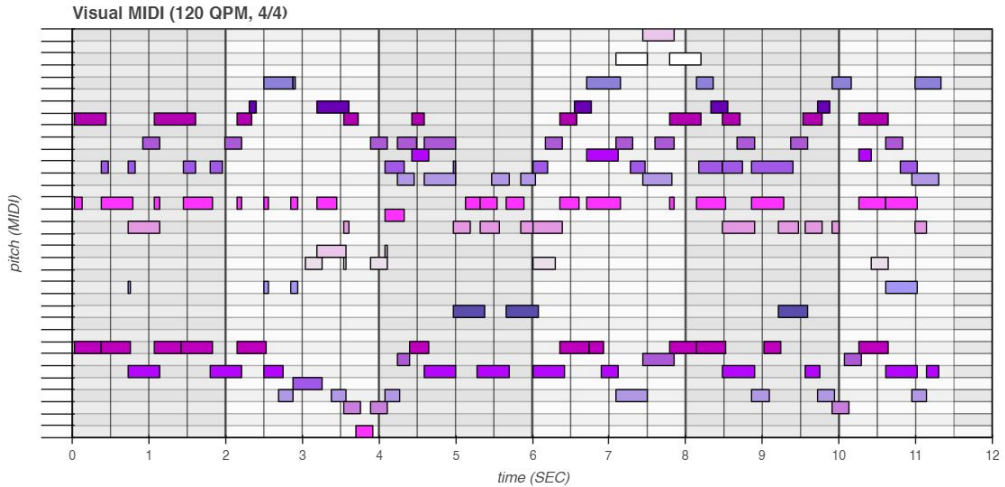
Fonte: Google Forms

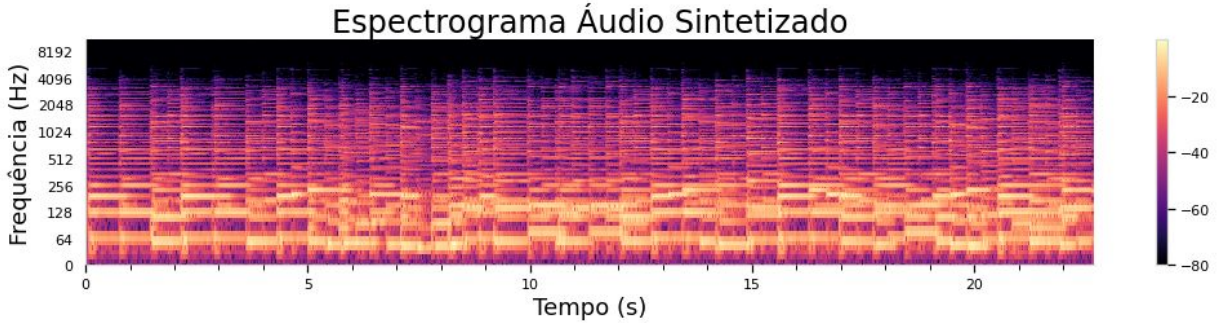
Resultado Transformer



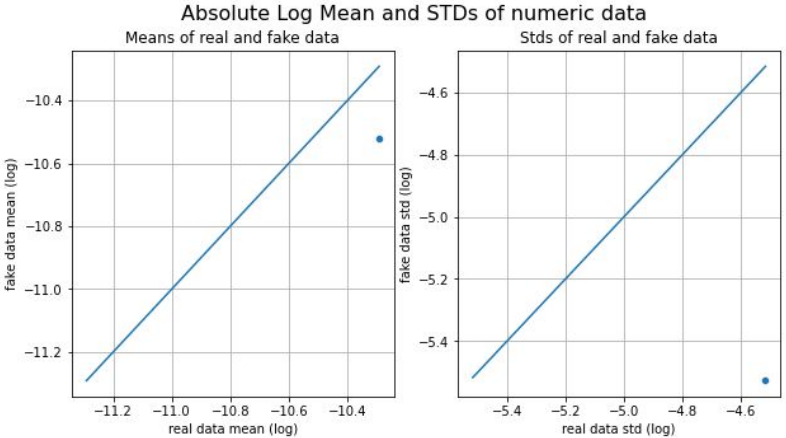


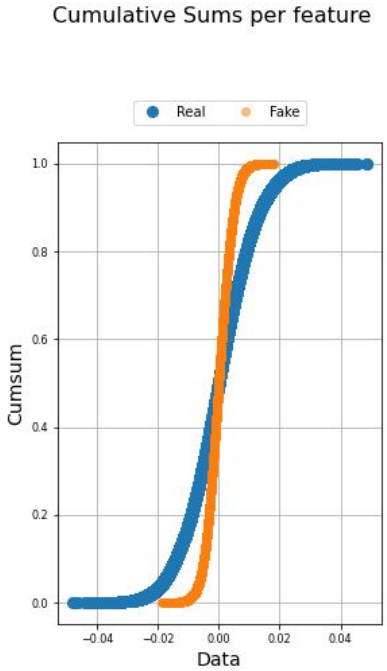
Resultado Transformer

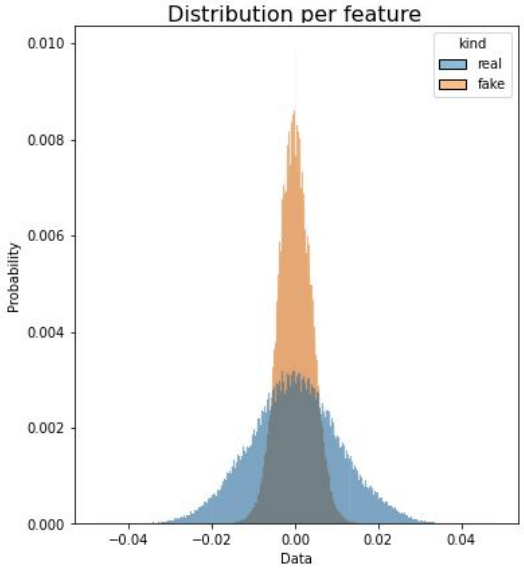




Resultado Transformer



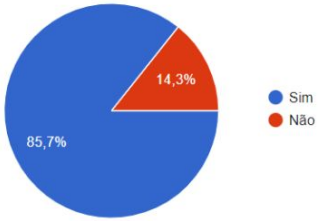




KS Test = 0,7722103025240228

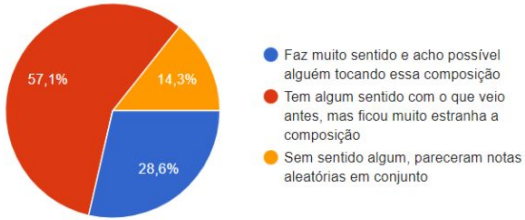
Resultado Transformer

Você considera este áudio uma música?

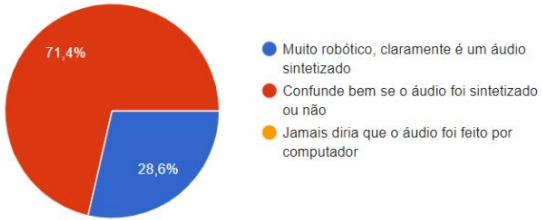


Você considera que este áudio

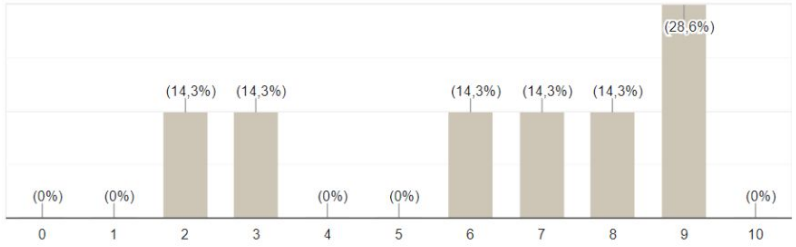
Sobre a composição e o áudio de input



Sobre a síntese



Se você tivesse que dar uma nota para este áudio, qual seria?



Nota Média = 6,3

Fonte: Google Forms

Discussão dos Resultados

- A GANSynth não trouxe resultados com características musicais, tendo outputs apenas com frequência fundamental em 8kHz, sendo basicamente ruído;

- A resposta da LSTM, pela opinião do grupo, foi bastante proveitosa em termos de Music Information Retrieval, sendo apenas considerável a mudança do compasso 3/4 da música original para 4/4 e o aumento significativo da energia no áudio de saída, com magnitude aproximadamente 9x maior que o áudio original, além disso, o resultado esperado respeita o campo harmônico original e o bit (168), e, conforme a pesquisa sensorial, resultou em um áudio agradável, sem ruído e plausível enquanto composição melódica;

- Já o transformer, resultou numa sequência de melodia e acompanhamento, que de forma geral conseguiu trazer o compasso 3/4 da música original, mas alterou o bit da música, ainda que levemente (171), mas mantendo o campo harmônico e, segundo a pesquisa sensorial, resultou em um áudio pouco agradável, sem ruído e pouco plausível enquanto composição musical.

Conclusões

- Recriar a GANSynth layer by layer, embora bastante didático, demonstrou ser desafiador e nos deparamos com entraves não necessariamente citados no artigo de referência, de forma que o grupo tomou a decisão de tentar também outras arquiteturas e metodologias. Todavia tentamos várias alternativas com relação aos hiperparâmetros, tempo de treinamento, alterações básicas de arquitetura, mas infelizmente esta montagem não resultou em áudio enquanto música, somente áudio com fundamental em 8kHz, de forma que podemos caracterizar como ruído;

- A mudança de paradigma para note sequence, isto é, token de música, demonstra ser um fator bastante significativo com relação ao Music Information Retrieval, já que esta estrutura mantém as propriedades do arquivo MIDI e isto se reflete na qualidade do áudio;

- Mesmo com a perplexidade próxima a 2, o grupo considera que os resultados da LSTM foram bastante representativos, acontecendo o mesmo com o Transformer, de forma que ambos os áudios estavam em um range plausível para a entrada;

Conclusões

- Embora em termos estruturais o resultado do Transformer seja mais fiel ao padrão de entrada, isto é manteve o compasso 3/4, a magnitude da energia e o padrão de composição, o LSTM teve maior aceitação perante o público conforme a pesquisa sensorial, de forma que fica evidente que embora tecnicamente o Transformer tenha um desempenho superior, lhe falta criatividade de composição e definição de áudio.

- Procuramos compreender resultados parciais da entrega anterior em busca de respostas para tentar corrigir as mudanças de compasso, bit, magnitude que apareceram no LSTM, contudo, nossos esforços pessoais não resultaram em melhora e, também, a equipe do Magenta, responsável direto pelos modelos pré-treinados não deu retorno com relação aos questionamentos, além disso, após o trabalho realizado, ocorreram atualizações no Google Colab e no Tensorflow, de forma que isso prejudicou bastante o desempenho dos modelos pré-treinados, havendo a necessidade de downgrade para versões anteriores.

Bibliografia:

[1] Engel, J.; Agrawal, K. K.; Chen, S.; Gulrajani, I.; Donahue, C.; Roberts, A.;" **Gansynth: Adversarial Neural Audio Synthesis**"; ICLR; 2019;

[2] Conner, M.; Gral, L.; Adams, K.; Hunger, D.; Strelow, R.; Neuwirth, A.; "**Music Generation Using an LSTM**"; MICS; 2022;

[3] Huang, C.; Vaswani, A.; Uszkoreit, J.; Shazeer, N.; Simon, I.; Hawthorne, C.; Dai, A.; Hoffman, M., Dinculescu, M.; Eck, D.; "**Music Transformer**"; 2018;

[4] Muller, M.; "**Information Retrieval for Music and Motion**"; 1ª Edição; Editora Springer; 2010.