

Análise da aplicação de Rede Adversária Generativa para composição de músicas de compositores do período barroco

Luiz Eduardo Pereira^{1*}; Mateus Modesto²

¹ Usp/Esalq. MBA Data Science e Analytics. Av. Pádua Dias, 11 - 13418-900 Piracicaba, São Paulo, Brasil

² Usp/Esalq. Msc. Engenharia de Produção e de Manufatura. Av. Pádua Dias, 11 - 13418-900 Piracicaba, São Paulo, Brasil

*autor correspondente: luizedupereira000@gmail.com

Análise da aplicação de Rede Adversária Generativa para composição de músicas de compositores do período barroco

Resumo

Este trabalho aborda a composição de música barroca por meio de inteligência artificial, visando a contribuição de métodos de geração para a produção musical inspirada em um estilo escolhido. O objetivo geral é criar músicas originais utilizando uma abordagem de aprendizado de máquina, empregando um conjunto de dados contendo composições do compositor Antonio Vivaldi, do período barroco. A metodologia adotada combina as redes neurais “Generative Adversarial Network” [GAN] e “Wasserstein GAN with Gradient Penalty” [WGAN-GP] com a técnica de modelagem de sequências temporais “Long Short-Term Memory” [LSTM] para gerar composições que evocam o rico estilo barroco. Os resultados revelaram músicas que capturam a essência do período, apresentando algumas semelhanças com o conjunto de dados original. Além disso, as músicas geradas foram submetidas a uma análise comparativa utilizando o método “t-Distributed Stochastic Neighbor Embedding” [t-SNE], revelando correspondências significativas com as composições de Vivaldi. Esses achados demonstram o potencial da abordagem proposta em criar peças musicais que refletem fielmente os padrões e características do estilo barroco. Além disso, surge a oportunidade de utilizar a inteligência artificial juntamente com as habilidades artísticas de um compositor, para dar origem a composições musicais geradas por IA que são posteriormente aprimoradas pela expertise de um profissional do ramo.

Palavras-Chave: Música, LSTM, GAN, WGAN-GP, t-SNE

Analysis of the application of Generative Adversarial Network for composing music by baroque period composers

Abstract

This work addresses the composition of Baroque music through artificial intelligence, aiming to contribute with generation methods for music composition inspired by a chosen style. The overall objective is to create original music using a machine learning approach, employing a dataset containing compositions by baroque composer Antonio Vivaldi. The adopted methodology combines the Generative Adversarial Network [GAN] and Wasserstein GAN with Gradient Penalty [WGAN-GP] neural networks with the Long Short-Term Memory [LSTM] temporal sequence modeling technique to generate compositions that evoke the rich Baroque style. The results revealed music that captures the essence of the period, presenting some similarities with the original dataset. Furthermore, the generated music underwent a comparative analysis using the t-Distributed Stochastic Neighbor Embedding [t-SNE] method,

revealing significant correspondences with Vivaldi's compositions. These findings demonstrate the potential of the proposed approach to create musical pieces that faithfully reflect the patterns and characteristics of the baroque style. Furthermore, the opportunity arises to use artificial intelligence alongside the artistic skills of a composer, to give rise to AI-generated musical compositions that are subsequently enhanced by the expertise of a professional in the field.

Keywords: Music, LSTM, GAN, WGAN-GP, t-SNE

Introdução

A música é uma das formas mais expressivas de arte e é capaz de evocar uma ampla gama de emoções e sensações em seus ouvintes. Através da combinação de ritmo, melodia e harmonia, a música pode contar histórias, expressar sentimentos e transmitir mensagens poderosas. Desde os primórdios da humanidade, a música tem sido uma parte integral da vida e das tradições culturais em todo o mundo. Com o tempo, ela evoluiu e se transformou, incorporando novas técnicas, instrumentos e estilos musicais (Buelow, 2004; Dong, 2018.).

No contexto da música barroca, que floresceu no século XVII, é comum que as composições incluam múltiplos instrumentos ou vozes, que interagem entre si e se desenvolvem de maneira complexa. Essas vozes são geralmente divididas em diferentes seções, como violinos, violas, violoncelos, cravos e outros instrumentos, cada um contribuindo com uma parte única para a textura sonora geral da obra. A teoria musical barroca aborda várias disciplinas de composição, incluindo harmonia e contraponto, que permitem relacionar os sons e as vozes de maneira coesa e expressiva (Buelow, 2004; Dong, 2018.).

Nos últimos anos, o conteúdo gerado por inteligência artificial, também conhecido como “Artificial Intelligence Generated Content” [AIGC], tem ganhado cada vez mais atenção da sociedade. As pessoas estão começando a se interessar pelos diversos produtos de geração de conteúdo desenvolvidos por grandes empresas de tecnologia, como o ChatGPT, um modelo de linguagem, e o DALL-E-2, utilizado para criar imagens a partir de textos descritivos. AIGC refere-se ao conteúdo que é gerado por meio de técnicas avançadas de inteligência artificial generativa, em contraposição à criação por autores humanos. Essa abordagem permite a automatização da criação de grandes quantidades de conteúdo em um curto período (Cao et al., 2018).

Na composição musical tradicional, o compositor combina conhecimentos musicais, emoção e experiência criativa para criar músicas. Com o avanço tecnológico, várias tecnologias relacionadas à música foram desenvolvidas, mas a maioria delas se concentrou na edição de músicas existentes, como arranjo e mixagem. Nestas aplicações, a música pode ser desmembrada em diversas partes e reorganizada com base em conhecimentos musicais

para criar novas peças. No entanto, recentemente, aprendizado de máquina tem sido usado para compor músicas com base em características musicais mais profundas (Li et al., 2019).

Um exemplo disso é o estudo realizado por Kulkarni et al. (2019), que teve como objetivo analisar diversas técnicas de composição musical baseadas em aprendizado de máquina, incluindo “Convolutional Neural Network” [CNN], “Recurrent Neural Network” [RNN], “Long Short-Term Memory” [LSTM] e “Generative Adversarial Network” [GAN]. Por meio da análise comparativa dos resultados obtidos, os pesquisadores concluíram que a GAN se mostrou a melhor opção para a geração de músicas, principalmente quando utilizada em conjunto com as outras redes mencionadas, potencializando ainda mais a qualidade da composição.

O presente trabalho tem como principal objetivo investigar e demonstrar como a Inteligência Artificial pode ser uma ferramenta poderosa na composição musical. A arte da música requer criatividade, habilidade técnica e experiência para criar algo novo e original, entretanto, o avanço tecnológico tem permitido que a IA seja usada para resgatar e reinterpretar estilos musicais do passado, como o barroco, por exemplo, e criar novas composições baseadas nessas características. Além disso, a IA pode auxiliar os músicos com sugestões de harmonias e melodias de acordo com suas preferências e estilos musicais, o que pode ajudar a explorar novas ideias criativas e expandir seus horizontes musicais. Outro benefício da colaboração entre humanos e IA é que ela pode democratizar o processo de criação musical, tornando-o mais acessível a pessoas com diferentes níveis de habilidade. Isso pode ser especialmente útil para aqueles que estão começando na música e não têm o conhecimento técnico necessário para criar uma composição do zero. Desse modo, a IA pode fornecer sugestões e ideias iniciais que os músicos podem utilizar para aprimorar suas composições.

Material e Métodos

Teoria Musical

O período musical barroco, que abrange aproximadamente os séculos XVII e XVIII, é caracterizado por uma profunda evolução na linguagem musical, marcada pelo uso extensivo de ornamentos, contrastes dinâmicos e uma ênfase na expressão emocional. Compositores como Johann Sebastian Bach, George Frideric Handel e Antonio Vivaldi são figuras proeminentes deste período. Essa era musical é conhecida por sua complexidade e sofisticação, e seu legado continua a influenciar a música clássica até os dias de hoje (Buelow, 2004).

Uma importante parte da música é a sua notação na forma de partitura. A Figura 1 apresenta um trecho do primeiro movimento de Inverno, de Vivaldi, para exemplificar alguns dos elementos básicos de uma partitura.

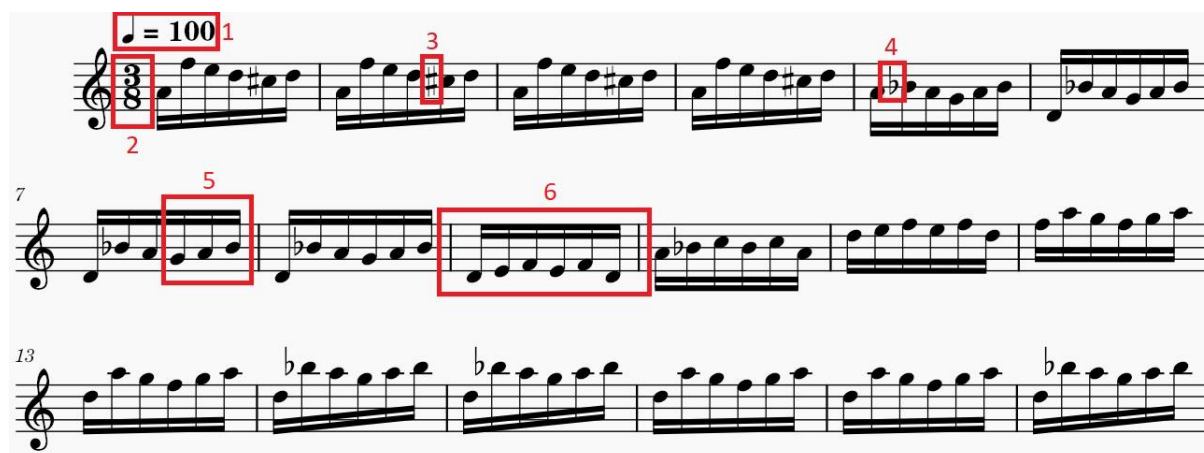


Figura 1. Partitura de Inverno, 1º movimento de Vivaldi

Fonte: ClassicalArchives (2023)

A 1ª marcação se refere ao andamento da música. O andamento define a velocidade em que as notas serão tocadas. A 2ª marcação se refere ao espaço do estilo de compasso e do tom da música, o tom é definido pelos símbolos de acidente bemol ou sustenido, mas nesse caso, a ausência dos símbolos de acidentes se refere ao fato de a música estar no tom natural, dó maior.

A 3ª e 4ª marcações se referem ao símbolo do acidente sustenido e bemol, respectivamente. Um sustenido eleva uma nota meio tom acima, enquanto um bemol baixa meio tom em uma nota.

A 5ª marcação apresenta um conjunto de notas, que sendo representadas pelo símbolo de semicolcheias representam as notas sol, lá e si, com duração de 0,25 tempos dentro do compasso. A 6ª marcação é o compasso, a música é feita de vários compassos, e geralmente um conjunto de compassos são organizados para formar frases em uma música.

Coleta de Dados

Este trabalho consiste na utilização de concertos para violino em formato “Musical Instrument Digital Interface” [MIDI] do compositor barroco Antonio Vivaldi. Os dados foram extraídos da base de dados ClassicalArchives (2023), que contém uma ampla coleção de MIDIs de música clássica e barroca.

Neste estudo, foi selecionado um conjunto de dados composto por um total de 83 obras para violino de Vivaldi. É importante ressaltar que esse tamanho de “dataset” pode ser considerado menor do que o geralmente utilizado em treinamento de redes neurais. A escolha

de um conjunto de dados menor neste caso específico é devido à natureza do trabalho, que se concentra em obras de um compositor específico e de um tipo específico, ou seja, concertos para violino de Vivaldi. Como o objetivo é explorar e analisar as características musicais e estilísticas dessas composições específicas, um conjunto de dados mais focado e limitado se fez necessário.

Tratamento de dados

A formação instrumental mais comum em um concerto para violino compreende um violino solo, acompanhado por violinos, violas, violoncelos, contrabaixos e cravo. No entanto, neste trabalho, foram utilizados apenas a melodia principal executada pelo violino solo e o acompanhamento dos violinos.

As músicas extraídas dos MIDIs passaram por processamento e tratamento no software de edição de partituras MuseScore 4. Com o auxílio desse software, realizou-se a seleção dos instrumentos a serem utilizados. Também foi realizada a transposição de todas as peças para o tom de Dó Maior, visando estabelecer uma padronização entre as composições. Além disso, foram feitos ajustes para preencher ou eliminar pausas prolongadas, assegurando a continuidade da melodia a ser criada. Essas etapas de edição e refinamento no MuseScore 4 contribuíram para adaptar as músicas às necessidades e especificações do estudo.

Outra modificação foi a separação de cada música em conjuntos de 100 notas. Essa separação foi necessária devido à natureza da rede neural utilizada, onde é necessário ter um tamanho fixo de entradas para treino. Para cada nota foi associado um número inteiro para representá-la. Todas as notas naturais receberam um número seguindo a ordem crescente das notas e em seguida foi atribuído números para as notas com acidentes. Isso foi feito pois músicas no tom de Dó Maior tem majoritariamente notas naturais, então as notas com acidentes foram movidas para o final para garantir que fossem utilizadas somente se necessário, e não por estarem no meio de duas notas naturais.

Para a utilização da biblioteca de Python para manipulação de músicas, Music21, foi necessário converter os arquivos MIDIs para o formato musicxml, uma variação do formato xml para música. Isso foi necessário pois notas com durações muito curtas estavam sendo ignoradas pelo formato MIDI quando combinado com essa biblioteca.

LSTM

A “Long Short-Term Memory” [LSTM] é um tipo de “Recurrent Neural Network” [RNN]

desenvolvida por Hochreiter e Schmidhuber (1997). Ela foi criada com o objetivo de superar as limitações das RNNs tradicionais na modelagem de dependências de longo prazo em sequências de dados.

A principal característica que diferencia as LSTMs das RNNs convencionais é a introdução de células de memória. Essas células são capazes de armazenar informações ao longo do tempo e controlar o fluxo dessas informações através da rede. Essa capacidade de reter informações relevantes ao longo do tempo é fundamental para o desempenho das LSTMs em várias tarefas, como processamento de linguagem natural e sequências temporais.

A Figura 2 apresenta uma unidade recorrente da LSTM. Essas unidades recorrentes são conectadas em cadeia para processar sequências de dados ao longo do tempo. Cada unidade recorrente recebe uma entrada, que pode ser uma observação em uma determinada etapa de tempo, e produz uma saída, que pode ser passada para a próxima unidade recorrente na sequência. Essa conexão em cadeia permite que as unidades recorrentes troquem informações e aprendam dependências de longo alcance nas sequências.

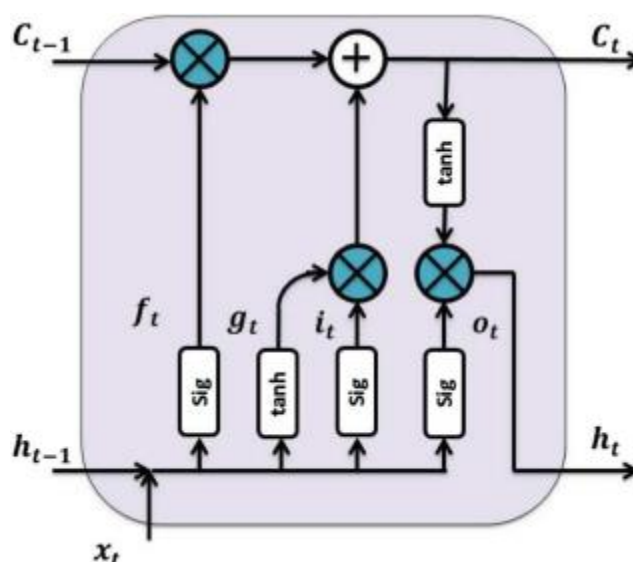


Figura 2. Representação de uma unidade recorrente da LSTM
Fonte: Smagulova e James (2019)

Em cada iteração a unidade recorrente da LSTM utiliza h_{t-1} (Estado oculto da etapa anterior) e x_t (Entrada da etapa atual). Essas entradas são combinadas e passam por diversas portas que controlam o fluxo de informações.

O f_t (Porta de esquecimento) determina quais informações devem ser esquecidas de c_{t-1} (Estado anterior da célula). Ele recebe a entrada combinada de h_{t-1} e x_t e em seguida aplica a função sigmóide, que gera valores entre 0 e 1. Então, f_t é multiplicado pelo valor de c_{t-1} , indicando quais partes devem ser descartadas (multiplicadas por 0), lembradas (multiplicadas por 1) ou parcialmente lembradas (multiplicadas por um valor entre 0 e 1).

O i_t (Porta de entrada) ajuda a identificar elementos importantes que devem ser adicionados em c_t (Estado da célula). Ele recebe a entrada combinada de h_{t-1} e x_t e aplica a função sigmóide. O resultado de i_t é multiplicado por g_t (Candidato a estado da célula), que é calculado aplicando a função \tanh (Tangente hiperbólica) à entrada combinada de h_{t-1} e x_t , dessa forma, somente as informações consideradas importantes por i_t são adicionadas à c_t .

O valor de c_t é dado por c_{t-1} multiplicado por f_t e em seguida adicionando o resultado de i_t multiplicado por g_t .

Por último, é atualizado o valor de h_t (Estado oculto). Para isso, c_t é passado por uma função de ativação \tanh e multiplicado por o_t , resultado da entrada combinada de h_{t-1} e x_t aplicado a uma função sigmóide.

Após isso, c_t e h_t são retroalimentados para a próxima unidade recorrente, e o processo se repete na próxima etapa. Esse “loop” continua até que a LSTM processe todos os elementos na sequência.

GAN

“Generative Adversarial Network” [GAN] por Goodfellow et al. (2014) é uma arquitetura de rede neural composta por duas partes principais: o gerador e o discriminador. Essas duas redes são treinadas em conjunto de forma adversarial, ou seja, uma rede tenta enganar a outra. A Figura 3 apresenta a arquitetura de uma GAN.

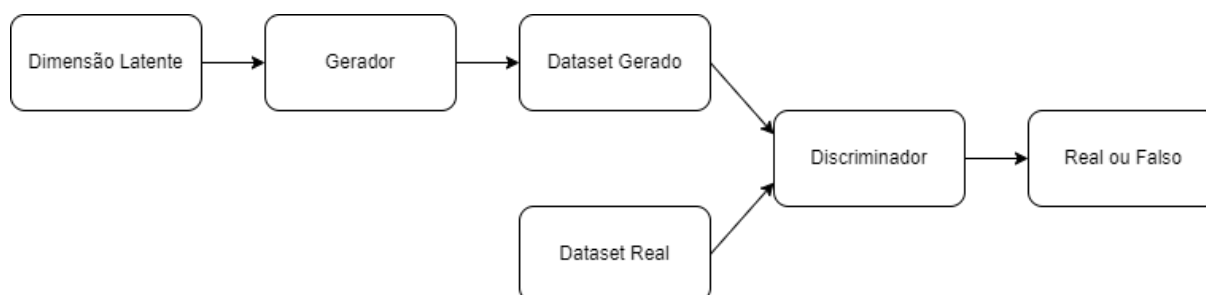


Figura 3. Arquitetura de uma GAN

Fonte: Elaborado pelo autor, baseado em Kulkarni et al. (2019)

O objetivo do gerador é criar composições sintéticas que se assemelham aos dados reais fornecidos durante o treinamento. Ele recebe uma entrada aleatória, chamada de dimensão latente, geralmente um vetor de números aleatórios, e a transforma em uma composição sintética. O objetivo é que essas composições sejam tão realistas quanto possível, a ponto de enganar o discriminador.

Por outro lado, o discriminador é treinado para distinguir entre as composições reais e as sintéticas produzidas pelo gerador. Ele recebe uma composição e tenta determinar se ela

é real ou sintética. O discriminador é uma rede neural que aprende a classificar as composições corretamente, fornecendo um “feedback” para o gerador.

Durante o treinamento, as duas redes são atualizadas em um jogo de “minimax”, onde o gerador busca melhorar suas habilidades de geração de dados e enganar o discriminador, enquanto o discriminador busca se tornar cada vez mais habilidoso em distinguir entre composições reais e sintéticas. Essa competição entre as redes resulta em um refinamento contínuo do gerador, tornando-o capaz de gerar dados sintéticos de alta qualidade.

A ideia central de uma GAN é que, ao final do treinamento, o gerador seja capaz de gerar dados que sejam indistinguíveis dos dados reais, de acordo com o discriminador. Isso significa que o gerador aprendeu as características e complexidades do conjunto de dados fornecido durante o treinamento e é capaz de criar novas composições que se parecem genuinamente reais.

WGAN-GP

Uma “Wasserstein Generative Adversarial Network with Gradient Penalty” [WGAN-GP] por Gulrajani et al. (2017) é uma variação da arquitetura de GAN que aborda algumas limitações e desafios associados às GANs convencionais. Embora as GANs tradicionais busquem gerar dados realistas através da competição entre o gerador e o discriminador, elas podem enfrentar problemas como instabilidade no treinamento, dificuldade na convergência e mensuração inadequada da qualidade da geração. A WGAN-GP aborda essas questões de maneira diferente.

Nas GANs convencionais, a função de perda é baseada na divergência de Jensen-Shannon [JS] ou na entropia cruzada (“cross-entropy”), e o objetivo é minimizar essa divergência para que o gerador produza saídas que se assemelham às amostras reais. No entanto, essa abordagem tem algumas limitações. Uma delas é que a divergência JS pode ser difícil de otimizar e levar ao problema de “mode collapse”, onde o gerador produz apenas um subconjunto limitado de saídas, ignorando a diversidade das amostras reais.

A WGAN-GP propõe uma abordagem diferente, usando a distância de Wasserstein como medida de desempenho. Ao minimizar essa distância de Wasserstein, estamos ensinando o gerador a criar amostras que se pareçam o máximo possível com as amostras reais, de forma que a transformação entre elas seja suave e gradual. Isso ajuda a evitar problemas como “mode collapse”, no qual o gerador produz repetidamente um subconjunto limitado de saídas e melhora a qualidade geral da geração.

Na WGAN-GP, queremos que o discriminador seja suave e não tenha variações abruptas. Isso é chamado de “lipschitz continuous”, uma propriedade matemática que mantém os valores dos gradientes do discriminador sob controle.

O “Gradient Penalty” [GP] é uma técnica que usamos para fazer com que o discriminador não desenvolva gradientes muito grandes, que podem causar instabilidade no treinamento. A ideia é simples: quando o discriminador tem gradientes muito grandes, ele pode reagir exageradamente a pequenas mudanças nas entradas, o que pode resultar em problemas no treinamento. O GP impõe uma penalidade nos gradientes exagerados, forçando o discriminador a ser mais suave e estável.

Em resumo, a WGAN-GP é uma evolução das GANs tradicionais que visa resolver problemas de instabilidade, convergência e mensuração de qualidade. A introdução da distância de Wasserstein como função de perda, juntamente com a penalização do gradiente, proporciona um treinamento mais estável e confiável, resultando em melhores resultados de geração de dados.

Arquiteturas dos modelos avaliados

A Figura 4 apresenta a estrutura do discriminador. Ele possui 4 camadas LSTM, capazes de lidar com sequências de dados. As camadas LSTM permitem que a rede aprenda padrões e dependências temporais nos dados de entrada.

Model: "Discriminator"

| Layer (type) | Output Shape | Param # |
|---------------------------|------------------|---------|
| lstm (LSTM) | (None, 100, 256) | 265216 |
| lstm_1 (LSTM) | (None, 100, 256) | 525312 |
| dense (Dense) | (None, 100, 128) | 32896 |
| leaky_re_lu (LeakyReLU) | (None, 100, 128) | 0 |
| lstm_2 (LSTM) | (None, 100, 256) | 394240 |
| dense_1 (Dense) | (None, 100, 128) | 32896 |
| leaky_re_lu_1 (LeakyReLU) | (None, 100, 128) | 0 |
| lstm_3 (LSTM) | (None, 100, 256) | 394240 |
| dense_2 (Dense) | (None, 100, 1) | 257 |

=====
Total params: 1,645,057
Trainable params: 1,645,057
Non-trainable params: 0

Figura 4. Arquitetura do Discriminador
Fonte: Elaborado pelo autor

Além disso, a rede possui 2 camadas densas, também conhecidas como camadas totalmente conectadas, que processam as informações provenientes das camadas anteriores. Essas camadas densas conectam todos os neurônios da camada anterior a todos os neurônios da camada atual.

A função “Leaky ReLU” (“Rectified Linear Unit”) é uma modificação da função ReLU que permite um pequeno gradiente quando a entrada é negativa. Isso evita que os neurônios sejam desativados e melhora o fluxo de informação. Ao utilizar a “Leaky ReLU” em camadas específicas, a rede pode lidar melhor com valores negativos e tomar decisões mais precisas.

A camada de saída da rede é uma camada densa com uma única unidade. Essa camada utiliza a função de ativação sigmóide, que comprime a saída entre 0 e 1, representando a probabilidade de a música de entrada ser real ou gerada.

A Figura 5 apresenta a estrutura do gerador. Ele é composto por 4 camadas LSTM para lidar com sequências, 3 camadas densas e 2 camadas de “Leaky ReLU”. Além disso, o gerador utiliza 2 camadas de “batch normalization”.

Model: "Generator"

| Layer (type) | Output Shape | Param # |
|---|-------------------|---------|
| lstm (LSTM) | (None, None, 256) | 365568 |
| lstm_1 (LSTM) | (None, None, 256) | 525312 |
| dense (Dense) | (None, None, 128) | 32896 |
| leaky_re_lu (LeakyReLU) | (None, None, 128) | 0 |
| batch_normalization (Batch Normalization) | (None, None, 128) | 512 |
| lstm_2 (LSTM) | (None, None, 256) | 394240 |
| dense_1 (Dense) | (None, None, 128) | 32896 |
| leaky_re_lu_1 (LeakyReLU) | (None, None, 128) | 0 |
| batch_normalization_1 (Batch Normalization) | (None, None, 128) | 512 |
| lstm_3 (LSTM) | (None, None, 256) | 394240 |
| dense_2 (Dense) | (None, None, 200) | 51400 |
| reshape (Reshape) | (None, 100, 2) | 0 |
| Total params: 1,797,576 | | |
| Trainable params: 1,797,064 | | |
| Non-trainable params: 512 | | |

Figura 5. Arquitetura do Gerador

Fonte. Elaborado pelo autor

As camadas de “batch normalization” são responsáveis por normalizar as entradas de uma camada, o que acelera o treinamento da rede neural e melhora sua capacidade de generalização. Essa técnica ajusta e padroniza os valores de entrada, garantindo que eles tenham média zero e variância unitária. Isso facilita o processo de aprendizado da rede.

A camada de “reshape”, por sua vez, tem a função de modelar os dados de saída para um formato que represente a música. Essa etapa de remodelagem é necessária para garantir que a saída do gerador esteja na forma adequada para representar uma música.

As redes neurais GAN e WGAN-GP compartilham uma estrutura de rede semelhante, mas com as seguintes especificidades. Enquanto ambas seguem a arquitetura de um gerador e um discriminador, a WGAN-GP difere ao não empregar a função de ativação sigmóide na saída do discriminador, o que permite uma melhor estabilidade no treinamento. Além disso, na abordagem WGAN-GP, o discriminador é submetido a mais iterações de treinamento do que o gerador, aprimorando ainda mais a qualidade da rede geradora. Quanto às funções de perda, a GAN utiliza “binary cross-entropy”, que se concentra na divergência de distribuições, enquanto a WGAN-GP emprega a distância de Wasserstein, uma métrica que mede a distância entre as distribuições de probabilidade e oferece uma representação mais suave e estável do treinamento da rede geradora. Além disso, a WGAN-GP possui a parte de penalização de gradiente, que é uma característica fundamental para garantir a suavidade e a estabilidade no treinamento. A inclusão da penalização de gradiente ajuda a mitigar problemas de convergência lenta e a promover uma melhor correspondência entre as distribuições de probabilidade do gerador e do conjunto de dados real.

T-SNE

O “t-Distributed Stochastic Neighbor Embedding” [t-SNE] é um algoritmo de aprendizado de máquina não supervisionado que é amplamente utilizado para visualização de dados e redução de dimensionalidade. Foi desenvolvido por Maaten e Hinton (2008) e tornou-se uma ferramenta popular para análise exploratória de dados.

A redução de dimensionalidade é um processo usado para representar dados complexos em um espaço de dimensões menores, mantendo as relações e estruturas importantes dos dados originais. Quando lidamos com conjuntos de dados de alta dimensionalidade, pode ser difícil visualizar ou analisar os padrões e relações intrínsecas presentes neles. Nesse sentido, o t-SNE é uma técnica eficaz para representar os dados em um espaço bidimensional ou tridimensional, facilitando sua visualização.

O t-SNE realiza a redução de dimensionalidade mapeando pontos de dados de alta dimensão em um espaço de menor dimensão, onde as distâncias entre os pontos são preservadas o máximo possível. Ele usa uma abordagem probabilística para modelar as

similaridades entre os pontos de dados nas duas dimensões. O algoritmo calcula uma probabilidade condicional para cada par de pontos de dados, que representa a similaridade entre eles. Essas probabilidades são convertidas em uma distribuição conjunta em um espaço de baixa dimensão e o mesmo processo é repetido nesse novo espaço até que uma solução estável seja alcançada.

Uma das características importantes do t-SNE é a tendência de preservar estruturas locais e capturar agrupamentos em densidades diferentes. Isso significa que os pontos que estão próximos uns dos outros nos dados originais também tendem a estar próximos na visualização reduzida. Portanto, o t-SNE é especialmente útil para revelar agrupamentos e estruturas não lineares em conjuntos de dados complexos.

Além da visualização de dados, o t-SNE também pode ser usado para comparação de dados. Neste trabalho, inspirado por Kolokolova et al. (2020), o t-SNE foi utilizado para comparar as músicas de Vivaldi com as composições geradas e verificar se elas têm padrões semelhantes ou diferentes. Se os dados forem visualmente semelhantes nas projeções do t-SNE, isso pode indicar uma semelhança nas estruturas subjacentes. Por outro lado, se os dados aparecerem diferentes nas projeções do t-SNE, isso pode sugerir diferenças significativas entre os conjuntos de dados. A Figura 6 apresenta um exemplo gráfico do t-SNE, nele podemos notar que no geral, dados semelhantes se agrupam em clusters, indicando que cada área de cluster possui diferentes características em comparação com os outros.

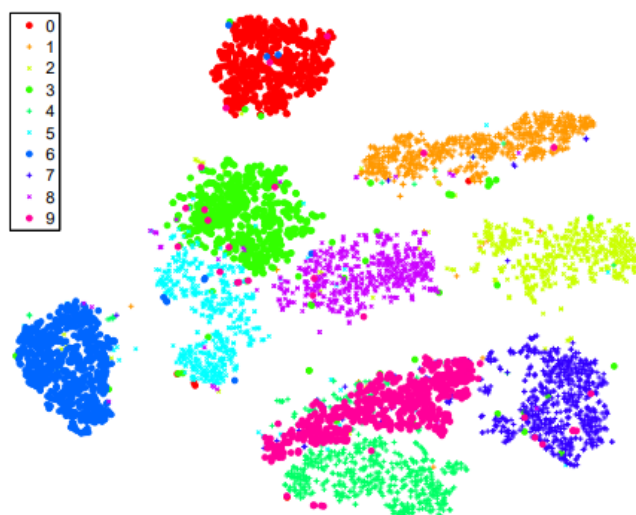


Figura 6. Exemplo do t-SNE para um dataset de escrita manual
Fonte. Maaten e Hinton (2008)

Pós-processamento

O pós-processamento é o momento em que apresenta uma oportunidade para o artista ou produtor criar em cima da música gerada pela Inteligência Artificial. Ao ter a estrutura musical básica fornecida pela GAN ou WGAN-GP, o artista pode adicionar camadas adicionais de instrumentação, como solos, contrapontos, e variações rítmicas. Essa abordagem permite que o artista transforme a ideia inicial gerada pela GAN em uma composição musical completa e refinada.

Assim, o pós-processamento permite ao artista exercer sua criatividade e expressão, adicionando nuances, ajustes e personalização à composição. É nesse estágio que a música ganha vida própria, transcendendo a mera ideia gerada pela Inteligência Artificial e se tornando uma obra musical autêntica.

Resultados e Discussão

Programação e Arquivo de Áudios

O código fonte deste trabalho está disponível no GitHub® através do link <<https://github.com/luizedu9/BaroqueGAN>>, bem como o Arquivo de Áudios, onde estão disponibilizados vários exemplos de composições geradas pela GAN e WGAN-GP.

Treinamento

GAN e WGAN-GP se destacam como abordagens poderosas no campo da aprendizagem de máquina, especialmente ao lidar com a criação de conteúdo, como imagens, músicas e texto. Durante o processo de treinamento desses modelos, é crucial ter uma compreensão clara dos resultados que estão sendo obtidos ao longo do caminho. Obter resultados satisfatórios demanda ajustes de diversos parâmetros ao longo do processo de treinamento, um processo que foi marcado por testes para descobrir a combinação ideal. Além disso, foi crucial considerar o tempo de execução dos modelos como um fator determinante na seleção desses valores.

Um parâmetro importante é a dimensão latente, que representa o espaço a partir do qual o gerador produz os dados. Esse processo envolve a criação de um vetor de valores aleatórios que, por meio de múltiplas camadas de transformações, é convertido em dados que se assemelham às amostras reais desejadas. Neste contexto, o tamanho escolhido para a dimensão latente foi de 1000.

O número de épocas (“epochs”) determina quantas vezes o conjunto de treinamento completo é apresentado ao modelo, para esse trabalho foi utilizado um tamanho 100. Foi

utilizado também um “batch size” de 64, que significa que a cada iteração do treinamento, o modelo atualiza seus parâmetros com base em 64 exemplos por vez.

A GAN começa a ter resultados que realmente representam uma música a partir da época 70, enquanto a WGAN-GP começa a ter resultados com boa qualidade a partir da época 10. O tamanho das épocas e do “batch size” foi escolhido para que a WGAN-GP consiga dar resultado em tempo satisfatório. Enquanto a GAN consegue treinar todas as 100 épocas em média de 5 minutos, a WGAN-GP leva em média 12 horas.

A diferença no tempo de treinamento entre GAN e WGAN-GP é significativa. A WGAN-GP é uma variação da GAN projetada para resolver problemas de convergência e estabilidade. No entanto, ela envolve cálculos mais complexos durante o treinamento, como o cálculo do gradiente do discriminador em relação às entradas. Isso explica por que o treinamento da WGAN-GP é mais demorado do que o da GAN convencional.

A WGAN-GP possui alguns parâmetros adicionais. Nessa técnica, o discriminador passa por um treinamento mais frequente em comparação com o gerador, adotando uma relação de 5 vezes mais iterações para o discriminador. Adicionalmente, é aplicado um peso para regular a penalização do gradiente, o qual exerce controle sobre o impacto dessa penalização na função de perda durante o processo de treinamento. Neste contexto, o valor definido para esse peso foi estabelecido como 10.

Perda

Duas métricas frequentemente utilizadas para avaliar a eficácia tanto de GANs quanto de WGAN-GPs são as perdas (“Loss”) associadas ao discriminador e ao gerador. O monitoramento dessas perdas ao longo do treinamento oferece informações cruciais sobre o processo de aprendizado e a interação entre as redes adversárias. A perda do discriminador reflete o quão bem ele é capaz de distinguir entre amostras reais e geradas, enquanto a perda do gerador indica quão eficazmente ele está produzindo dados que enganam o discriminador. O equilíbrio entre essas perdas é um indicativo importante da qualidade das amostras geradas e da convergência do treinamento.

Na Figura 7, observamos a variação da perda da GAN ao longo do tempo. No início do treinamento, é comum ter uma perda considerável. Essa perda alta indica que o gerador não está produzindo amostras de alta qualidade que enganam o discriminador. Nesse estágio inicial, o gerador ainda está aprendendo a mapear adequadamente o espaço de entrada para o espaço de saída. Enquanto isso, o discriminador está tentando aprender a distinguir entre amostras reais e sintéticas.

Conforme o treinamento progride, em torno de aproximadamente 70 épocas, a perda da GAN começa a se estabilizar. Isso indica que o gerador está se aprimorando e se tornando mais eficiente em enganar o discriminador. A perda mais baixa sugere que o gerador está gerando músicas que são mais difíceis de serem diferenciadas das músicas reais. Isso é um sinal positivo e indica que o treinamento está progredindo adequadamente.

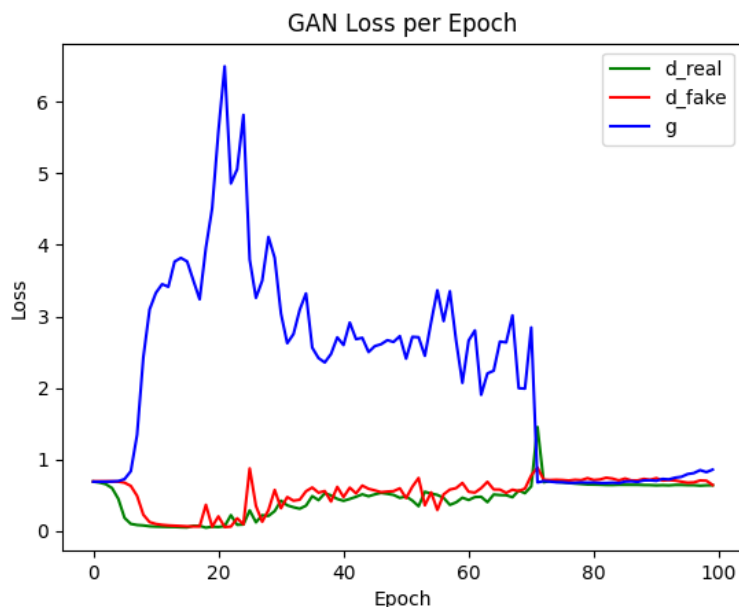


Figura 7. Perda do gerador [g], do discriminador para dados reais [d_real] e do discriminador para dados sintéticos [d_fake]

Fonte: Resultados originais da pesquisa

A Figura 8 ilustra a dinâmica das perdas na WGAN-GP ao longo do treinamento. A WGAN-GP difere das GANs tradicionais ao utilizar a distância de Wasserstein para medir a discrepância entre as distribuições de amostras reais e geradas.

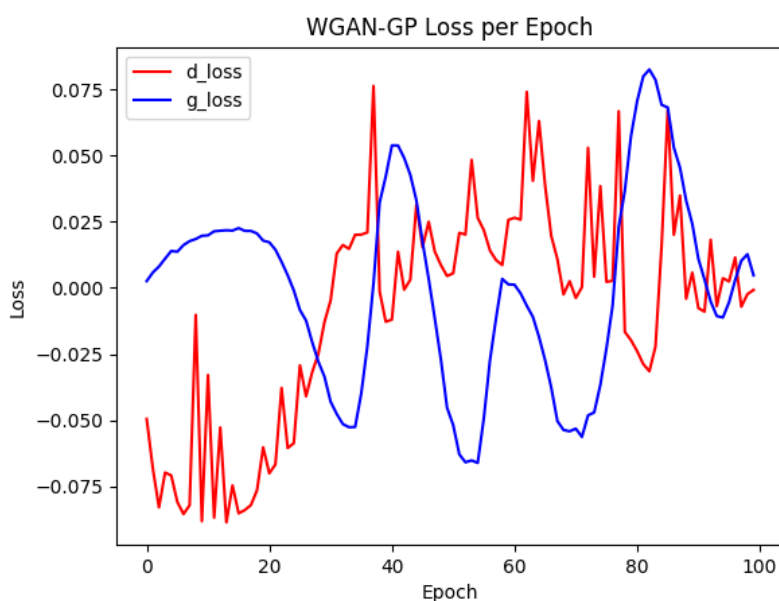


Figura 8. Perda do gerador [g_loss] e do discriminador [d_loss]

Fonte: Resultados originais da pesquisa

A perda do discriminador mede a distância entre as distribuições de dados reais e gerados, com o objetivo de minimizar a distância de Wasserstein. Ao observar na Figura 8 a evolução dos valores do discriminador ao longo do treinamento, notamos que inicialmente eles são predominantemente negativos e conforme o treinamento avança, esses valores gradualmente se tornam positivos. Idealmente, um valor próximo de zero indica que as distribuições estão se aproximando e o gerador está produzindo dados mais similares aos reais, sinalizando um treinamento bem-sucedido.

A perda do gerador em uma WGAN-GP não é tão crucial quanto em outras arquiteturas GAN. Isso se deve à abordagem única da WGAN-GP, que prioriza a minimização da distância de Wasserstein entre as distribuições de dados reais e gerados, em vez de depender exclusivamente da perda geradora. A penalidade do gradiente na WGAN-GP ajuda a evitar problemas de treinamento, como o desvanecimento do gradiente, melhorando a estabilidade do processo. Isso significa que o gerador é guiado pelo gradiente do discriminador para gerar dados mais realistas, mas a perda geradora em si não é o foco central do treinamento. A ênfase recai na minimização da distância de Wasserstein e na criação de gradientes suaves para ambos os gerador e discriminador, garantindo que o treinamento seja mais controlado e os resultados mais consistentes.

Frequência de notas e duração

As músicas reais foram padronizadas no tom de dó maior, para ter uma base tonal comum entre todas as músicas. Isso irá se refletir na forma de picos de frequência nas notas naturais e uma baixa frequência nas notas com acidentes (bemóis e sustenidos). Essa distribuição de frequências é esperada, pois, em uma peça musical tonal, é comum que a maioria das notas pertença à escala principal.

A Figura 9 representa a frequência das notas musicais nos conjuntos de dados real, gerados pela GAN e gerados pela WGAN-GP. A análise dessa imagem permite observar se as duas técnicas conseguem replicar bem a distribuição de notas.

Podemos observar que a abordagem da WGAN-GP apresentou uma melhor capacidade de se aproximar das frequências das notas presentes nos dados reais. Em contraste, ao examinar a representação da GAN, observa-se a presença de picos mais proeminentes de discrepância em relação aos dados reais. Essa diferença no comportamento das duas abordagens fornece informações interessantes sobre a qualidade de suas saídas.

A análise permite concluir que a WGAN-GP alcançou um desempenho superior no que diz respeito à representação das frequências das notas musicais. A proximidade notável entre as frequências geradas pela WGAN-GP e as frequências presentes nos dados reais sugere

que essa abordagem conseguiu capturar as nuances e os padrões tonais de forma mais precisa e coesa.

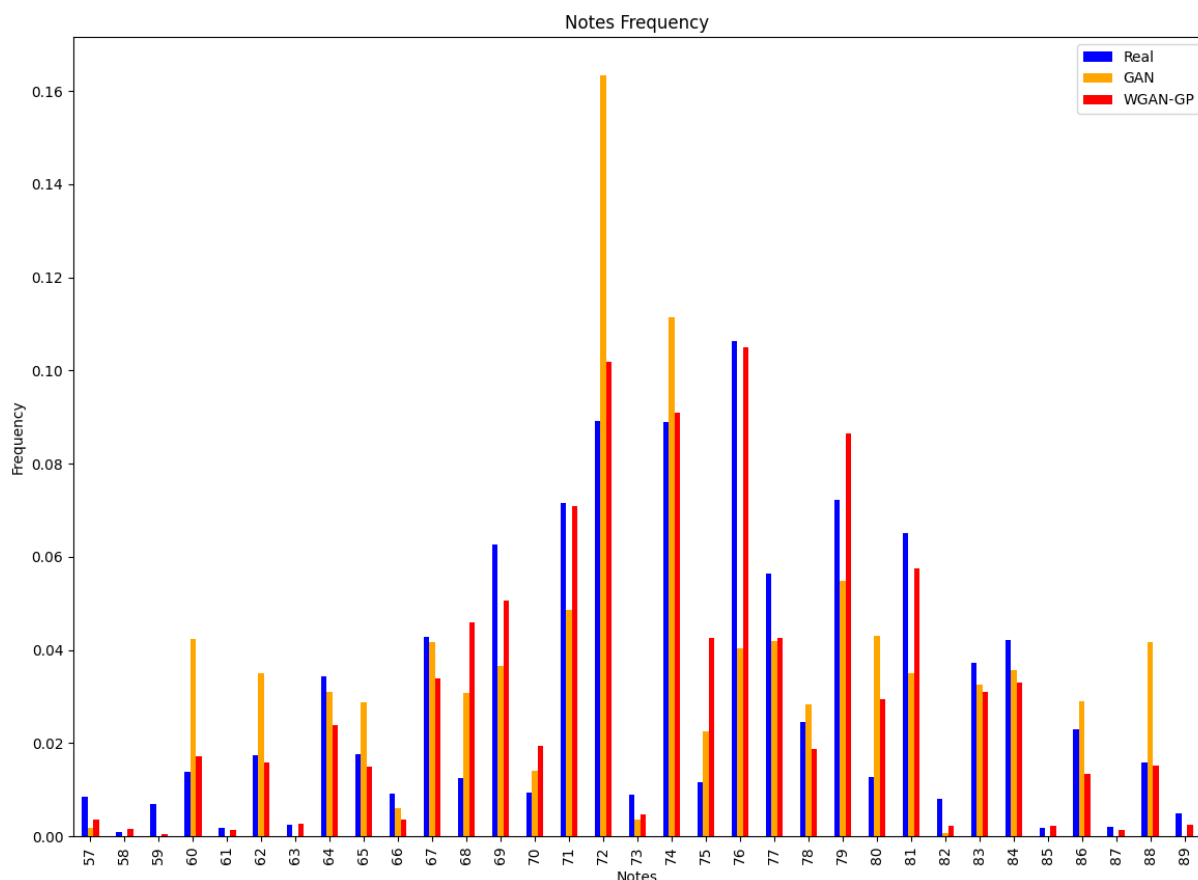


Figura 9. Frequência de notas das músicas reais, das geradas pela GAN e geradas pela WGAN-GP

Fonte: Resultados originais da pesquisa

Por outro lado, os picos mais significativos de diferença nas frequências das notas geradas pela GAN indicam uma menor fidelidade em reproduzir com precisão as características tonais presentes nos dados reais. Isso pode ser atribuído a variações na geração das notas, resultando em um desvio perceptível das frequências tonais esperadas.

Na Figura 10, observamos que ambas as técnicas não conseguiram replicar de maneira equivalente os padrões de duração presentes nos dados reais. A GAN exibe uma tendência a distribuir as durações de forma mais uniforme, mas se perdendo um pouco para as notas com durações maiores que 0,5. Por outro lado, a abordagem WGAN-GP apresenta uma inclinação para repetir mais frequentemente as durações de 0,25, possivelmente gerando uma certa monotonia temporal que não reflete a diversidade das durações nas composições reais.

Além disso, ao examinar mais profundamente a comparação das notas geradas, podemos notar que a GAN parece ter gerado uma variedade mais ampla nas durações musicais. Por outro lado, a WGAN-GP parece ter conseguido capturar certas repetições de

durações específicas, indicando uma aderência maior a um subconjunto nas características musicais.

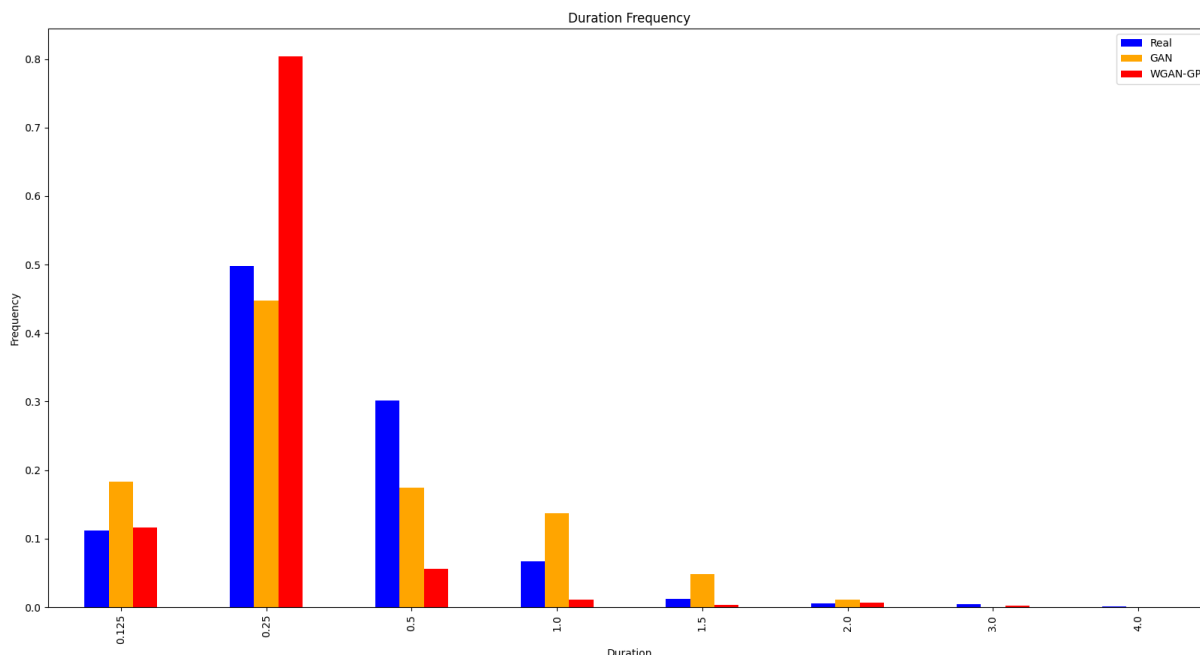


Figura 10. Frequência das durações das músicas reais, das geradas pela GAN e geradas pela WGAN-GP

Fonte: Resultados originais da pesquisa

A análise de frequências destaca a importância de considerar tanto as notas quanto as durações ao avaliar músicas geradas pela GAN e WGAN-GP. O aprimoramento simultâneo dessas dimensões é essencial para capturar os padrões temporais e melódicos das composições originais, aproximando-as das características desejadas. Mas deve-se reconhecer que a avaliação não deve se basear exclusivamente nas frequências, uma vez que essas podem representar apenas uma parte das preferências musicais. É necessário introduzir técnicas de comparação mais abrangentes que capturem as nuances das composições originais e geradas para uma avaliação mais precisa da qualidade e autenticidade das composições geradas.

t-SNE

A Figura 11 é uma representação visual da técnica de t-SNE. O t-SNE reduz a dimensionalidade dos dados, permitindo que eles sejam visualizados em um espaço bidimensional. Na Figura 11, podemos observar que a dimensionalidade dos dados foi reduzida para que as músicas reais e geradas pela GAN e WGAN-GP pudessem ser comparadas. O t-SNE mapeou cada música em um ponto no espaço visual.

Observa-se que os dados reais apresentam uma maior diversidade e variabilidade, preenchendo uma região mais ampla no espaço bidimensional. No entanto, tanto as músicas geradas pela GAN quanto as geradas pela WGAN-GP parecem estar concentradas em regiões mais específicas dentro do espaço, indicando uma representação mais restrita dos padrões musicais presentes no conjunto de dados real.

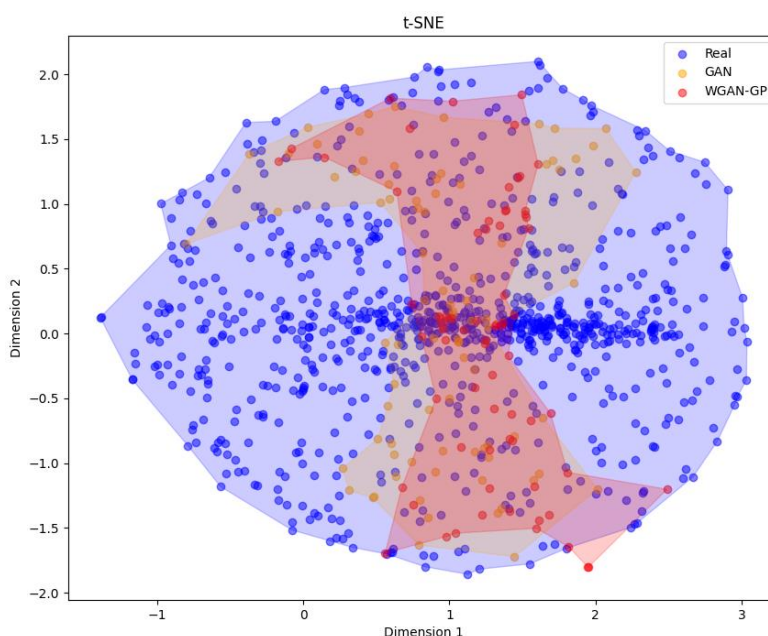


Figura 11. t-SNE para as músicas reais, geradas pela GAN e WGAN-GP
Fonte: Resultados originais da pesquisa

Apesar disso, é notável que ambas as técnicas (GAN e WGAN-GP) conseguiram capturar e representar uma parcela dos padrões musicais presentes nos dados reais. Embora não tenham conseguido abranger toda a variedade do conjunto de músicas reais, os resultados sugerem que as músicas geradas conseguem imitar, em algum grau, certos aspectos das músicas reais. As regiões em que as músicas geradas estão localizadas indicam que elas compartilham semelhanças com as músicas reais, possivelmente capturando certos traços tonais, rítmicos ou estruturais do estilo barroco.

Músicas

O repositório das músicas geradas pela GAN e WGAN-GP está disponível no Arquivo de Áudio (Link na subseção “Programação e Arquivo de Áudios”). Essa coleção reúne exemplos de áudios gerados por ambas as técnicas, exibindo tanto momentos em que elas obtiveram sucesso na criação de peças musicais com características da música barroca, como momentos em que apresentaram resultados menos satisfatórios.

Dentro da seção "Good Samples" para cada técnica, é possível encontrar exemplos de áudios nos quais tanto a GAN quanto a WGAN-GP demonstra uma notável musicalidade e conseguem incorporar aspectos típicos da música barroca. Esses exemplos servem como demonstrações bem-sucedidas das capacidades das técnicas em recriar o estilo musical desejado.

Por outro lado, na seção "Bad Samples", são apresentados exemplos em que as técnicas não conseguiram reproduzir uma qualidade sonora satisfatória ou não conseguiram capturar os elementos característicos da música barroca. Esses exemplos evidenciam os desafios enfrentados pelas técnicas em termos de geração musical consistente e autêntica.

Na seção "Full Sample" para cada técnica, é disponibilizada uma música completa gerada pela respectiva técnica. Essas músicas incluem partes que demonstram uma boa musicalização e incorporam os aspectos da música barroca, bem como partes que podem não ser tão bem-sucedidas em termos de qualidade ou fidelidade ao estilo musical desejado.

A Figura 12 retrata um segmento de composição gerado pela WGAN-GP, com algumas nuances características do estilo barroco. O padrão rítmico estabelece uma coesão que é presente em composições deste período musical. A presença sutil, mas perceptível, de repetições de motivos melódicos específicos apresenta uma familiaridade ao estilo. Neste exemplo, a técnica implementada pela WGAN-GP demonstra habilidade ao preservar o tom musical, sem incorrer na geração indesejada de acidentes nas notas.



Figura 12. Trecho da partitura "WGAN-GP Full Sample" disponível no arquivo de áudio.
Fonte: Resultados originais da pesquisa

Na Figura 13 podemos observar um exemplo de um trecho em que a GAN não conseguiu capturar os elementos distintivos da música barroca. De forma mais específica, esse trecho musical apresenta uma notável ausência de estrutura rítmica sólida e de coesão entre as ideias musicais. A falta de padrões rítmicos característicos do estilo barroco contribui para uma sensação de desconexão e irregularidade no desenrolar da composição. Além

disso, a falta de coesão nas ideias musicais resulta em uma composição fragmentada, onde os motivos melódicos e harmônicos não se interligam harmoniosamente.



Figura 13. Trecho da partitura “GAN Full Sample” disponível no arquivo de áudio.
Fonte: Resultados originais da pesquisa

GAN vs WGAN-GP

Quando comparamos a GAN com a abordagem WGAN-GP na tarefa de gerar composições musicais, distintas diferenças surgem em relação ao processo de treinamento e à qualidade resultante. A WGAN-GP se destaca ao produzir músicas de boa qualidade desde os estágios iniciais do treinamento, aprimorando essa qualidade à medida que avança. Por outro lado, a GAN convencional tende a alcançar resultados musicais de maior qualidade somente em fases finais do treinamento, exibindo um progresso inicial mais lento. Isso pode ser atribuído à formulação mais avançada da WGAN-GP, que incorpora conceitos como a distância de Wasserstein e a penalização de gradientes. Essas melhorias na estrutura da WGAN-GP contribuem para uma estabilidade de treinamento e uma aprendizagem mais eficiente dos padrões musicais desejados, em comparação com a GAN convencional. Além disso, é importante notar que a GAN apresenta um processo de execução mais rápido devido à sua estrutura mais simples, enquanto a WGAN-GP requer mais tempo para executar um número equivalente de épocas.

No geral, ambas as técnicas conseguiram atingir o objetivo de criar músicas com características do período barroco. No entanto, a GAN exibe maior probabilidade de gerar trechos com qualidade inferior, saindo do estilo barroco. Por outro lado, a WGAN-GP demonstra uma inclinação para produzir composições de alta qualidade, embora com uma estrutura temporal mais fixa, favorecendo a geração de notas mais rápidas. No entanto, essa abordagem pode apresentar limitações ao capturar a essência do estilo, uma vez que também são características desse estilo a presença de notas mais longas.

Contudo, é crucial destacar que ambas as técnicas enfrentam um desafio na geração musical: a capacidade de desenvolver uma ideia musical ao longo da composição. Enquanto ambas as abordagens são capazes de criar uma ideia inicial, ambas parecem encontrar dificuldades em seguir e expandir essa ideia de forma coesa ao longo da música. Esse desafio reflete a natureza da música barroca, na qual as ideias musicais são frequentemente elaboradas e desenvolvidas ao longo da peça.

Colaboração de IA com o Artista

A era da inteligência artificial trouxe consigo um muitas de possibilidades no campo da criação musical. Não se trata apenas da capacidade de gerar composições originais com algoritmos sofisticados, mas também da oportunidade de colaboração entre a tecnologia e os artistas da área. Para exemplificar a colaboração entre IA e artista, foi utilizado a música “WGAN-GP Full Sample”, disponível no Arquivo de Áudio para chegar em algumas conclusões.

Primeiramente, como pontos negativos, é evidente uma falta de variedade rítmica, como foi possível ver na subseção de “Frequências de notas e durações”, onde a WGAN-GP mostrou predisposição a uma duração específica (0,25). Também é possível ver que a música introduz um tema principal, mas possui dificuldade em continuar ou finalizar essa linha de ideia. Outro ponto, é a introdução de alguns poucos acidentes musicais, que apesar de comum no estilo barroco, podem ter sido utilizados de forma errada na composição, trazendo uma estranheza ao tom da música.

Como pontos positivos temos o preenchimento correto dos compassos musicais, durações adequadas nas figuras e tonalidade definida dentro da harmonia. Também foi possível observar a presença de uma escala pentatônica, uma sequência de cinco notas que são escolhidas de forma específica dentro de uma oitava musical. Essas cinco notas são selecionadas de maneira a criar um padrão que soa bem e é amplamente utilizado na música em geral.

Outro ponto positivo é a presença de células com elementos que remetem a ornamentos musicais, como trinados, mordentes e grupetos, muito utilizados na música barroca. A presença de notas de pedal, juntamente com o ritmo acelerado e os outros elementos citados conseguem demonstrar uma certa influência dos concertos de Vivaldi no qual a GAN e WGAN-GP foram treinadas.

No Arquivo de Áudios está disponível a composição “WGAN-GP + Artist”, onde foi dado a liberdade ao artista de utilizar elementos e ideias disponíveis na composição “WGAN-GP Full Sample” para criar uma composição inteiramente nova. O artista utilizou alguns

compassos da composição original que combinavam elementos de harmonia e fraseado e os reorganizou. Foram feitas algumas mudanças em alturas e durações, para aumentar a variedade rítmica da música. Para manter a essência da música barroca, foi repetido algumas células musicais. Por fim, foi acrescentado contrastes nas intensidades e adicionado um acompanhamento de baixo contínuo.

O caso apresentado demonstra como a inteligência artificial pode ser combinada de forma colaborativa com o pensamento humano, resultando na criação de novas obras musicais. Esse processo não apenas amplia as possibilidades na composição musical, mas também lança as portas para a literatura, as artes visuais e diversos outros meios criativos.

Conclusão

Através do presente trabalho foi explorado a aplicação da “Generative Adversarial Network” [GAN] e de sua melhoria “Wasserstein Generative Adversarial Network with Gradient Penalty” [WGAN-GP] na composição de músicas inspiradas pelo compositor barroco Antonio Vivaldi. Ao longo deste trabalho, foi evidenciado a capacidade dessas técnicas de capturar a essência do período barroco, criando composições que evocam a estética e a emoção característica da época.

A avaliação metódica, realizada através da comparação entre as composições reais e geradas, revelou “insights” valiosos. A análise das frequências das notas e durações, combinada com a técnica de t-SNE, proporcionou uma compreensão mais profunda da relação entre as músicas geradas e as obras originais de Vivaldi. Como resultado, foi possível gerar composições que são um subconjunto das obras originais.

Destaca-se a eficácia distinta das duas abordagens GAN e WGAN-GP, cada uma apresentando suas próprias características distintivas. Enquanto a GAN demonstrou tempos de execução mais curtos, mas demandando múltiplas iterações para alcançar um nível satisfatório de qualidade musical, a WGAN-GP se destacou ao produzir composições de alta qualidade desde as primeiras iterações, apesar de requerer um tempo de execução mais prolongado por iteração. Essas nuances indicam a importância de escolher a abordagem adequada com base nas necessidades e restrições específicas do projeto.

Além disso, é crucial ressaltar que a colaboração entre a inteligência artificial e um artista não apenas é possível, mas altamente enriquecedora. Ao usar as composições geradas como fonte de inspiração, o artista pode dar vida a criações musicais mais profundas e significativas, aproveitando a fusão entre a criatividade humana e as capacidades inovadoras da tecnologia.

Em resumo, este estudo demonstrou que a aplicação de GAN e WGAN-GP na composição musical é uma abordagem promissora para criar peças que capturam a essência de um período artístico específico. Ao mesmo tempo, enfatizou a complementaridade entre a inteligência artificial e a expressão humana, abrindo caminho para futuras explorações no campo da música e da arte em geral.

Agradecimentos

Gostaria de expressar meus agradecimentos a Sâmea Alves, que me ajudou com a parte musical deste projeto, ao orientador Mateus Modesto, Eduardo Miranda e a Lynx Process, em especial ao Ulisses Teixeira e Rafaela Martins pelo apoio ao longo deste curso.

Referências

Buelow, G.J. 2004. A history of baroque music. Indiana University Press, Bloomington, Indiana, EUA.

Cao, Y.; Li, S.; Liu, Y.; Yan, Z.; Dai, Y.; Yu, P.S.; Sun, L. 2018. A comprehensive survey of AI-Generated Content (AIGC): A history of generative AI from GAN to ChatGPT. Association for Computing Machinery 37(4).

ClassicalArchives. 2023. Antonio Vivaldi (1678-1741). Disponível em: <<https://www.classicalarchives.com/midi/composer/3521.html>>. Acesso em: 16 mai. 2023.

Dong, H.W.; Hsiao, W.Y.; Yang, L.C.; Yang, Y.H. 2018. MuseGAN: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment. Proceedings of the AAAI Conference on Artificial Intelligence 32(1).

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. 2014. Generative Adversarial Networks. Disponível em: <<https://arxiv.org/abs/1406.2661>>. Acesso em: 17 ago. 2023.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A. 2017. Improved Training of Wasserstein GANs. Disponível em: <<https://arxiv.org/abs/1704.00028>>. Acesso em: 17 ago. 2023.

Hochreiter, S; Schmidhuber, J. 1997. Long Short-Term Memory. Neural Computation, 9(8): 1735-1780.

Kolokolova, A.; Billard, M.; Bishop, R.; Elsisy, M.; Northcott, Z.; Graves, L.; Nagisetty, V.; Patey, H. 2020. GANs & Reels: Creating Irish Music using a Generative Adversarial Network. Disponível em: <<https://arxiv.org/abs/2010.15772>>. Acesso em: 18 mar. 2023.

Kulkarni, R.; Gaikwad, R.; Sugandhi, R.; Kulkarni, P.; Kone, S. 2019; Survey on deep learning in music using GAN. International Journal of Engineering Research & Technology 8(9).

Li, S.; Jang, S.; Sung, Y. 2019. Automatic melody composition using enhanced GAN. Mathematics 7(10).

Maaten, L. van der; Hinton, G. 2008. Visualizing data using t-SNE. Journal of machine learning research 9(11).

Smagulova, K.; James, A.; 2019. A survey on LSTM memristive neural network architectures and applications. The European Physical Journal Special Topics. 228: 2313-2324.