FACULDADE DE TECNOLOGIA DE SÃO PAULO

GABRIEL NIERI

Aspectos Técnicos dos Algoritmos e Arquiteturas de IA na Produção Musical

SÃO PAULO
2025

FACULDADE DE TECNOLOGIA DE SÃO PAULO

GABRIEL NIERI

Aspectos Técnicos dos Algoritmos e Arquiteturas de IA na Produção Musical

Trabalho submetido como exigência parcial para a obtenção do Grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientadora: Professora Mestre Grace Anne Pontes Borges

SÃO PAULO
2025

RESUMO

Este trabalho explora os aspectos técnicos de algoritmos e arquiteturas de Inteligência Artificial (IA) aplicados à produção musical, contextualizando a evolução da composição algorítmica. O estudo aborda a necessidade do conhecimento musical humano no desenvolvimento dessas tecnologias e os desafios tecnológicos, criativos e legais inerentes à automação completa do processo. O objetivo principal é analisar propostas de algoritmos e sistemas de IA na geração de conteúdo musical, bem como os conceitos e princípios musicais empregados. A metodologia empregada é a revisão bibliográfica, com análise de obras acadêmicas e artigos científicos de bases de dados como *Scopus* e *Google Scholar*. São apresentados estudos de caso que demonstram a aplicação de diferentes abordagens de IA, como sistemas algorítmicos, redes neurais recorrentes e autoencoders variacionais, na criação musical. Conclui-se que a IA possui um potencial significativo na produção musical, mas a intervenção humana e o conhecimento musical são cruciais para superar os desafios existentes e aprimorar a qualidade das composições geradas.

Palavras-chave: Inteligência Artificial; Produção Musical; Algoritmos; Arquiteturas de IA; Composição Algorítmica.

ABSTRACT

This work explores the technical aspects of Artificial Intelligence (AI) algorithms and architectures applied to music production, contextualizing the evolution of algorithmic composition. The study addresses the need for human musical knowledge in the development of these technologies and the technological, creative, and legal challenges inherent in the complete automation of the process. The main objective is to analyze proposals for AI algorithms and systems in musical content generation, as well as the musical concepts and principles employed. The methodology used is a bibliographic review, with analysis of academic works and scientific articles from databases such as Scopus, and Google Scholar. Case studies are presented demonstrating the application of different AI approaches, such as algorithmic systems, recurrent neural networks, and variational autoencoders, in musical creation. It is concluded that AI has significant potential in music production, but human intervention and musical knowledge are crucial to overcome existing challenges and improve the quality of generated compositions.

Keywords: Artificial Intelligence; Music Production; Algorithms; AI Architectures; Algorithmic Composition.

SUMÁRIO

1. INTRODUÇÃO

A música possui grande poder de influenciar o estado emocional das pessoas e, com os avanços da computação, surgiram diversos sistemas baseados em inteligência artificial capazes de atuar em áreas como entretenimento e saúde (Dash; Agres, 2024).

Em 2024, o cenário da indústria musical mundial já se mostrava predominantemente definido por plataformas de *streaming* digital, que chegaram a totalizar 69% dos lucros globais (IFPI, 2025). Esses dados indicam a crescente democratização do acesso à música e a relevância da tecnologia para alcançar tais resultados.

Com os serviços de *streaming* e a popularização do consumo musical, inovações e experimentações foram incentivadas. Todavia, apenas recentemente passou-se a considerar o processo completo de produção musical por meio de automação (Kwiecień et al., 2024).

Para compreender esse processo, é necessário que, durante a intervenção humana, haja conhecimento em conceitos musicais como melodia, acorde, harmonia e ritmo, colaborando com a autonomia do algoritmo (Morais, 2019).

Ampliando a abordagem, ao se considerar os algoritmos e arquiteturas utilizados na geração musical automatizada, é possível citar desde métodos simples, como cadeias de Markov, até sistemas mais complexos baseados em redes neurais, como redes diretas, convolucionais, recorrentes, transformadores, autocodificadores variacionais, entre outros (Civit et al., 2024).

Desse modo, é relevante avaliar a qualidade da geração dessas músicas, considerando a naturalidade das mudanças na história humana e nos estilos musicais (Kwiecień et al., 2024). A música precisa transmitir emoção, visto que carrega um elemento essencial em sua criação, independentemente da presença de um compositor (Juslin; Sloboda, 2001 apud Civit et al., 2024, p. 8). Essa expressividade contribui para uma geração menos mecânica e com mais nuances (Briot et al., 2024).

1.1. Hipóteses

Modelos de inteligência artificial podem estar relacionados à geração de músicas complexas, equiparáveis a composições humanas.

Algoritmos específicos podem ser mais adequados para contextos musicais que exigem soluções simples.

Sistemas de inteligência artificial podem contribuir para a produção musical, mantendo a qualidade das composições.

A familiaridade com conceitos musicais pode influenciar positivamente o desenvolvimento e o uso de sistemas voltados à criação automatizada de músicas.

Questões legais, criativas e tecnológicas podem representar obstáculos para a automação completa da produção musical por inteligência artificial.

1.2. Objetivo

O objetivo deste trabalho é analisar propostas de algoritmos e sistemas no âmbito técnico, bem como o seu funcionamento na geração de conteúdo musical. Além disso, busca-se estudar conceitos relacionados à composição, às regras e aos princípios musicais utilizados nesses algoritmos.

1.3. Justificativas

A abordagem da integração entre a tecnologia e a música torna-se essencial para permitir a ampliação do acesso a novos meios de produção artística e otimizar os processos que envolvem a criação musical. Nesse contexto, é necessário aprofundar a compreensão sobre como essas tecnologias funcionam.

A viabilidade também se apoia na crescente popularização de serviços de música baseados em IA, moldando novas relações entre a arte e a sociedade. A pesquisa realizada pela Federação Internacional da Indústria Fonográfica (IFPI) em 2023 revela o crescimento do *streaming*, que ocupa 48,90% do mercado global de música, demonstrando a importância de estudos relacionados à automação musical.

1.4. Metodologia

Este trabalho será conduzido por revisão bibliográfica, com base na análise de obras acadêmicas, artigos científicos e publicações recentes sobre aspectos técnicos de algoritmos e arquiteturas de Inteligência Artificial aplicados à produção musical, com foco em fontes indexadas em bases como *Scopus* e *Google Scholar*.

2. FUNDAMENTOS MUSICAIS

Este capítulo apresenta uma compreensão da música ao longo da história, articulando natureza, matemática, filosofia e tecnologia, além dos primeiros sons inspirados por fenômenos naturais. Mencionam-se também Pitágoras, Aristóteles e Boécio como figuras relevantes no desenvolvimento musical, culminando nas atuais tecnologias digitais que expandem o campo da produção.

2.1. Contexto histórico

Apesar da ausência de informações concretas sobre a origem da primeira música ou suas inspirações, elementos musicais como melodia e ritmo manifestam-se tanto na influência de sons produzidos por criaturas quanto em fenômenos naturais ao nosso redor, como o vento, as ondas e os trovões. A maioria dos sons é composta por harmônicas complexas ou sobretons, formando séries harmônicas que seguem uma ordem natural de ocorrência (Burgess, 2014).

O termo “série harmônica” é um conceito com aplicações tanto na música quanto na matemática. Ela consiste em uma sequência de múltiplos inteiros de um som fundamental, como, por exemplo, as frequências 1000, 2000, 3000, 4000, 5000 e 6000 Hz. A frequência fundamental é o primeiro valor da série, enquanto as frequências superiores são chamadas sobretons, harmônicas ou parciais (Bain, 2003).

Pitágoras, no século VI a.C., descreveu a relação matemática entre a largura de uma corda esticada e sua vibração ao ser puxada, identificando proporções simples que originam a oitava (2:1), a quinta (3:2), a quarta (4:3) e a tônica. O ciclo das quintas foi alcançado por meio desses cálculos e, com pequenos ajustes - conhecidos como coma pitagórica -, culminou na atual escala temperada (Burgess, 2014).

A oitava é formada por pequenas razões inteiras que ocorrem naturalmente entre sons harmônicos e complexos, ou entre sub-harmônicos de um tom periódico, seja ele puro ou composto. Tons com frequência nessa razão podem produzir uma oitava harmônica quando tocados simultaneamente, ou melódica, quando em sequência (Cheveigné, 2023).

Aristóteles (420b 10), em sua obra filosófica Da Alma (350 a.C.), afirma que “o som é um certo movimento do ar”, comparando-o ao impacto de objetos sobre superfícies lisas. Ele também explica que as emoções são princípios de movimento - assim como a música -, capazes de afetar homens e animais sensorialmente,

provocando sentimentos como tristeza, luto, entusiasmo ou relaxamento (Schoen-Nazzaro, 1978).

No ano 510 d.C., o filósofo e teólogo romano Boécio, figura central na tradução de obras clássicas gregas para o latim, desenvolveu o tratado *De Institutione Musica*. Segundo Mendes (2020), Boécio divide a música em três genera, por ordem de importância: *mundana*, *humana* e *instrumentalis*. A música mundana representa o nível mais elevado e sagrado, sendo imperceptível ao ouvido humano. A música humana relaciona-se à existência dos seres e do mundo terreno, expressando a medida musical presente na essência humana. Por fim, a música *instrumentalis* refere-se à forma audível da música, manifestada por instrumentos ou executantes que reproduzem a harmonia celeste, desde que obedeçam às proporções.

O canto gregoriano, junto ao seu sistema de notação musical - inalterado em aparência por sete séculos -, contribuiu para o desenvolvimento da música ocidental. Embora seu auge tenha ocorrido entre os séculos V e VIII, os *neumas* só aparecem em manuscritos a partir do século IX. Derivado das tradições grega e romana, o termo *neuma* significa “sinal” e é composto por dois sinais básicos: *virga*, uma linha ascendente indicando elevação vocal, e *punctum*, um ponto que representa sua queda. Ambos eram originalmente chamados *acutus* e *gravis* - a linha descendente - conforme os gramáticos clássicos (Parrish, 1978).

De acordo com Martin (2014), a notação musical não é apenas um sistema técnico de representação sonora, mas também uma forma de comunicação visual. Por meio de símbolos, linhas e palavras, constroem-se significados, sendo uma maneira criativa de atribuir sentido à música. A notação musical funciona como uma linguagem: ela determina o que pode ser dito, e o que se deseja expressar define a linguagem utilizada, no caso, a composição (Cardew, 1961).

Franco de Colônia contribuiu para a evolução da notação musical ao criar um sistema que especificava a duração de notas individuais por meio de formatos visuais distintos. Ele também definiu com mais clareza as pausas e, em sua obra *Ars cantus mensurabilis*, escrita entre 1260 e 1280, caracterizou as longas, breves e semibreves como unidades sonoras com durações específicas (Strayer, 2013).

O princípio de representar graficamente a duração do som permanece essencial na produção musical contemporânea, especialmente em ambientes digitais, como as DAWs (*Digital Audio Workstations*). Uma DAW é um *software* que permite gravar, organizar, editar e processar sons em trilhas (*tracks*) separadas, que podem

estar em formato de áudio ou MIDI, representando instrumentos ou vozes. Essas plataformas oferecem controle visual e técnico sobre cada elemento da produção. Exemplos como *Logic Pro*, *Pro Tools* e *Ableton Live* possibilitam manipular a duração, como o início e o fim das trilhas, com base em sistemas de tempo, compassos ou *ticks*, que são unidades de tempo digital equivalentes às subdivisões rítmicas medievais ou aos cantos gregorianos (Hosken, 2014).

A tecnologia MIDI permite controlar altura, intensidade, duração e articulações sem a necessidade de gravação acústica. Além da composição, a DAW contribui para o refinamento sonoro por meio de processos como mixagem e masterização. Na mixagem, é possível ajustar a densidade sonora (conjunto de ondas simultâneas), o posicionamento estéreo (canais de áudio) e aplicar efeitos como equalização, *reverb* e compressão - tudo isso sem alterar o arquivo original. Na masterização, o projeto final é renderizado em um único arquivo, exportado de acordo com configurações específicas, como formato e frequência (Hz), para posterior distribuição (Hosken, 2014).

Além das DAWs, outras ferramentas também contribuíram para a evolução da produção musical. Um exemplo são os *trackers* - *softwares* voltados à composição que se popularizaram nos anos 1980 em computadores como *Commodore* 64, *Amiga* e *Atari ST* (Reunanen, 2024). Os *trackers* utilizam uma interface em grade, onde o tempo é representado verticalmente e os eventos musicais (notas, efeitos e comandos) são inseridos por letras e números no teclado (Möllenkamp, 2014). Mesmo com recursos limitados, esse sistema permitia reorganizar composições com maior precisão.

Outro marco tecnológico na produção musical são as *grooveboxes*: instrumentos autônomos para criação de músicas eletrônicas em diversos estilos, como *techno*, *hip-hop*, *jungle* e *acid* (Tjora, 2009). As *grooveboxes* reúnem, em um único dispositivo, sequenciadores, controladores (botões, teclas e pads sensíveis à pressão), monitor de controle (display *LCD*), indicadores em *LED* e geradores de som - como sintetizadores, baterias e *samplers*. O objetivo é permitir a construção de sequências musicais em tempo real (Réveillac, 2018).

2.2. Conceitos musicais

O som é gerado pela vibração de um corpo e apresenta quatro características essenciais: frequência, amplitude, timbre e duração. A música resulta da organização

do som no tempo, estruturando-se em três componentes básicos - melodia, harmonia e ritmo - que são registrados por meio da notação musical (Souza Filho, 2015).

Segundo Mitra e Zualkernan (2025), alguns conceitos fundamentais da teoria musical podem ser definidos da seguinte forma: a nota representa um único som indicado por um símbolo; acordes são conjuntos de notas tocadas simultaneamente; o ritmo é o padrão de notas executadas; a melodia consiste na sucessão dessas notas; e a harmonia é o processo de combinação dos sons em uma unidade composicional. Já a altura (*pitch*) refere-se à propriedade do som baseada em sua frequência. O compasso é a unidade de tempo que agrupa um determinado número de batidas (*beats*), enquanto a partitura (*score*) é uma forma específica de notação musical que representa uma parte da composição, independentemente do instrumento. Ela contém informações como altura, notas, acordes, entre outras.

O termo “seções” (*sections*) refere-se a estruturas unitárias perceptíveis pelo ouvinte quando diversos elementos da música - como harmonia, melodia e ritmo - coincidem. As seções também podem formar unidades menores dentro de uma obra, compondo estruturas maiores a partir de suas partes (Spencer; Temko, 1988).

Nesse contexto, o *accompaniment* (acompanhamento) é o complemento da melodia por meio de suporte harmônico ou rítmico, podendo incluir a adição de registros e timbres ausentes no solo, sustentação metro-rítmica, acordes e enriquecimento sonoro. O acompanhamento pode ser vocal ou instrumental - como a percussão - e contribui para a identidade das seções musicais (Davidova; Zavadska, 2019).

A articulação, por sua vez, é um fator expressivo responsável por unir ou separar os sons dentro de uma frase musical, regulando como os tons são executados por meio de técnicas específicas (Rimas; Jr, 2024). Complementando esse aspecto da frase, uma definição mais recente de contraponto destaca-o como “a combinação de linhas musicais, soando simultaneamente, de acordo com um sistema de regras” (Sachs & Dahlhaus, 2011, apud Frigatti, 2020, p. 551). Assim, enquanto a articulação atua nos sons individuais, o contraponto organiza essas unidades em uma estrutura coerente.

Mudanças de volume ou intensidade, flutuações no tempo, variações de altura e ajustes no timbre são exemplos de dinâmicas na música (Vines; Nuzzo; Levitin, 2005). Além disso, quando uma frase musical não é interpretada como um motivo, ela tende a ser repetitiva ou cíclica, decorativa e com papel secundário - características

da figura. Os motivos (*motifs*), por sua vez, são as menores unidades musicais ou pequenos grupos de notas, surgindo antes da frase. Motivos melódicos apresentam identidade clara e podem ser a base a partir da qual o restante da composição se desenvolve (Roger, 1999).

A forma musical é caracterizada por estratégias voltadas ao equilíbrio entre repetição constante e mudança excessiva (Scholes, 1977 apud Huron, 2013). A frase, por sua vez, é definida como uma unidade maior que um motivo, porém menor que uma sentença. Rousseau complementa essa definição ao descrevê-la como uma progressão harmônica ou melódica contínua, que expressa uma ideia relativamente completa e termina com uma cadência. Outro conceito relevante para a teoria musical envolve as escalas e seus graus. Escalas consistem em sucessões de alturas ascendentes ou descendentes, semelhantes a uma escada. Já os graus representam os valores específicos de altura - determinando também sua posição numérica - usados sistematicamente em uma mesma composição para formar melodias e harmonias (Nikolsky, 2022).

A progressão harmônica pode ser definida sob dois enfoques: matemático ou tonal. Na abordagem matemática, é entendida como uma sucessão de números reais não nulos cuja sequência de inversos forma uma progressão aritmética (Tavares, 2014). Já na perspectiva tonal, trata-se de uma sequência de acordes que se move de forma intencional, criando direção e conduzindo o ouvinte por uma estrutura harmônica (Kostka; Payne; Almén, 2012).

Por fim, o *tremolo* é a repetição rápida de uma única nota em instrumentos de corda, ou uma variação de intensidade sonora que produz um efeito de vibração no volume - o que pode ser confundido com *vibrato* (Latham, 2011).

2.3. Gêneros musicais

A noção de gênero vai além de uma simples classificação estilística - é compreendida como um conjunto de eventos musicais organizados por regras socialmente aceitas, que se manifestam de forma distinta entre comunidades ou contextos históricos. Nessa perspectiva, o gênero musical não é definido apenas por aspectos formais e técnicos, revelando-se como uma construção cultural, e não uma estrutura fixa (Fabbri; Pinho, 2017).

Essa complexidade se evidencia ainda mais ao se considerar o papel central dos gêneros na mediação entre produção e recepção musical, contribuindo para

experiências que unem dimensões estéticas e sociais. Nesse sentido, os gêneros atuam como elementos de comunicação entre os envolvidos na prática musical, moldando gostos e identidades por meio do reconhecimento de convenções sonoras compartilhadas (Trotta, 2008).

Ao se levar em conta essa complexidade, a compreensão dos gêneros como construções culturais pode ser aprofundada. Estilos com alta diversidade e baixa uniformidade instrumental tendem a atrair músicos mais especializados. No entanto, à medida que um gênero se populariza, seus elementos passam a ser repetidos para atender às demandas do mercado, o que leva a uma tendência de simplificação - nesse caso, a uniformidade aumenta e a complexidade diminui. Esses fatores evidenciam uma tensão constante entre inovação e padronização na produção musical contemporânea (Percino; Klimek; Thunder, 2014).

Embora a tendência de simplificação seja frequentemente associada à popularização de estilos, a diversidade musical nos Estados Unidos não apresentou um declínio linear ao longo das décadas. Entre 1960 e 2010, observam-se alterações significativas na música popular: mesmo com períodos de queda, ocorreram fases de recuperação. Por exemplo, em 1980, estilos como *new wave*, *disco* e *hard rock* dominaram, resultando em uma redução na variedade musical. No entanto, esse cenário se reverteu com o surgimento de gêneros como o *rap*, que introduziram novos padrões sonoros e ampliaram a diversidade. Os dados indicam que, mesmo sob pressões comerciais, a música atravessa ciclos de inovação (Mauch et al., 2015).

Esse panorama sugere, ao traçar um paralelo com a produção musical por Inteligência Artificial, a importância de desenvolver sistemas capazes de operar com adaptabilidade e criatividade, sem se limitar à reprodução de padrões estabelecidos. A repetição excessiva sonora pode comprometer a identidade estética de criações automatizadas, tornando-as mais previsíveis.

3. INTELIGÊNCIA ARTIFICIAL: ALGORITMOS E ARQUITETURAS

São apresentados os principais fundamentos técnicos sobre modelos de redes neurais, com ênfase em algoritmos, arquiteturas e estruturas de dados. Os tópicos abordam algoritmos de otimização, retropagação, funções de perda, transformadores, autocodificadores e outras abordagens que ampliam a compreensão dos elementos da inteligência artificial.

3.1. Algoritmos

Algoritmos desempenham papel essencial no aprendizado de máquina, pois orientam os processos de otimização que tornam o treinamento de modelos mais eficaz. No caso das redes neurais, o algoritmo de gradiente descendente e suas variantes têm sido amplamente utilizados para minimizar funções de erro, ajustando os parâmetros do modelo. Muitas vezes, esses algoritmos são aplicados sem uma compreensão clara de seus comportamentos, limitações ou benefícios. Por isso, entender conceitos como os três tipos de gradiente descendente - *batch*, *stochastic* e *mini-batch* - além da escolha adequada da taxa de aprendizado, torna-se fundamental para aplicações práticas (Ruder, 2017).

O gradiente do tipo *batch* calcula o gradiente da função de custo utilizando todo o conjunto de dados de treinamento. Embora apresente convergência consistente, torna-se inviável em volumes muito grandes de dados. Já o tipo *stochastic* atualiza os parâmetros a cada novo exemplo de treinamento, reduzindo o tempo de processamento, mas dificultando a convergência - exigindo ajustes cuidadosos na taxa de aprendizado. Por fim, a técnica *mini-batch* busca equilibrar os dois casos anteriores, realizando atualizações com pequenos subconjuntos de dados (geralmente entre 32 e 256 amostras), resultando em convergência mais estável. Essa abordagem é amplamente empregada no treinamento de redes neurais profundas (Ruder, 2017).

Segundo a IBM (2024), a taxa de aprendizado é um hiperparâmetro que define o quanto o modelo ajusta seus parâmetros a cada etapa do algoritmo de otimização. Parâmetros são informações que o modelo aprende automaticamente a partir dos dados de treinamento - sendo essenciais para que ele realize previsões corretas. Já os hiperparâmetros são configurações externas ao modelo, não aprendidas automaticamente, e controlam como os parâmetros são ajustados durante o treinamento (Huawei, 2023).

O gradiente é um vetor de derivadas parciais em relação a todas as variáveis independentes. No contexto do aprendizado de máquina, ele indica como os valores devem ser ajustados para reduzir erros e melhorar o desempenho do modelo (Google Developers, 2025).

Uma técnica de otimização conhecida é o algoritmo ganancioso (*greedy*), no qual cada iteração seleciona a amostra com o maior ganho imediato, mesmo que essa escolha possa não ser a melhor solução. Esse procedimento garante eficiência computacional, porém é limitado pela tendência de escolher a opção mais viável a curto prazo, desconsiderando possibilidades de longo prazo (Bian; Zhou; Qian, 2022). Em contraste, o algoritmo estocástico adota aleatoriedade ou probabilidade, selecionando amostras de modo randômico em um subconjunto de dados a cada iteração (Youvan, 2023).

De forma semelhante à abordagem estocástica, redes neurais exigem uma função de otimização que guie o processo de aprendizado. Essa função, conhecida como função de perda, quantifica o quão distante está a predição da rede em relação ao valor real (*ground truth*); quanto menor seu valor, menor o erro da previsão comparado aos dados de treinamento. Nesse contexto, a entropia cruzada (*cross-entropy*) é um tipo de função de perda, semelhante à divergência de Kullback-Leibler, que mede a diferença entre duas distribuições de probabilidade. Ela é expressa como a soma ponderada dos logaritmos negativos das probabilidades previstas em relação às classes verdadeiras. Assim, a entropia cruzada não apenas avalia a precisão das previsões, mas também incorpora um grau de incerteza em cada predição (Spindelböck; Ranftl; Linden, 2021).

A divergência de Kullback-Leibler, conceito fundamental da teoria da informação, fornece uma medida quantitativa da diferença entre duas distribuições de probabilidade, avaliando o quanto uma distribuição se desvia da outra (a referência). Essa métrica é amplamente adotada em áreas como aprendizado de máquina (*machine learning*), estatística e ciência de dados, a fim de verificar a qualidade da distribuição prevista por um modelo em relação aos dados reais (Nawa; Nadarajah, 2024).

É fundamental considerar o tipo de codificação empregada para representar variáveis em tarefas de classificação, dada a influência que isso exerce no desempenho dos modelos. Nesse sentido, a codificação *one-hot* é amplamente utilizada para converter categorias em representações binárias, nas quais cada valor

distinto é transformado em uma nova coluna contendo 1 para presença ou 0 para ausência. Essa abordagem evita más interpretações que os algoritmos poderiam cometer ao analisar categorias (Alsowail; Al-Shehari, 2021). Da mesma forma que a entropia cruzada considera incertezas nas predições ao lidar com distribuições de probabilidade, a codificação *one-hot* preserva a integridade dos dados categóricos, contribuindo para o processo de aprendizado.

A métrica AUC-ROC (*Area Under the Receiver Operating Characteristic Curve*) é responsável por avaliar modelos de classificação sem interferir no processo de treinamento. Ela considera a taxa de verdadeiros positivos em relação aos falsos positivos, indicando a chance de o modelo errar ao classificar algo como verdadeiro. Sua representação gráfica forma uma curva: a AUC corresponde à área sob essa curva e expressa o desempenho geral, variando de 0 a 1 - quanto maior, melhor (Tafvizi; Avci; Sundararajan, 2022).

O Limite Inferior da Evidência (ELBO) surge como solução para a dificuldade de calcular a probabilidade dos dados. Ao buscar aproximar a distribuição real de variáveis ocultas, utiliza-se uma distribuição alternativa mais simples. Nesse processo, o ELBO atua como uma função que contribui para encontrar a melhor aproximação possível. Ele serve como um limite inferior para a evidência dos dados, e sua maximização equivale à minimização da divergência de Kullback-Leibler, mantendo a distribuição aproximada próxima da real (Blei; Kucukelbir e McAuliffe, 2018). O ELBO, portanto, estabelece-se como uma ferramenta importante em algoritmos que dependem de aproximações probabilísticas.

Ainda no campo das incertezas em algoritmos, a lógica *fuzzy* representa uma expansão matemática ao permitir valores de verdade contínuos entre 0 e 1, em vez de se restringir ao verdadeiro ou falso. Essa lógica é aplicada para lidar com situações de imprecisão e ambiguidade, pois aceita valores menos limitados e determinísticos, favorecendo seu uso em sistemas inteligentes (Son Il; Gyong Il; Min, 2016).

É necessário também considerar possíveis problemas durante o processo de aprendizado de máquina, como *underfitting* e *overfitting*. O *underfitting* ocorre quando o modelo é incapaz de capturar de forma eficaz a relação entre variáveis de entrada e saída, gerando alto índice de erro tanto em dados novos quanto nos dados de treinamento. Geralmente, esse fenômeno ocorre apenas em modelos excessivamente simples, exigindo menor regularização, maior número de atributos de entrada ou mais tempo de treinamento. Já o *overfitting* representa o extremo oposto: o modelo se

ajusta demais aos dados de treinamento, resultando em altas taxas de erro em dados de teste. Enquanto o *underfitting* costuma ser identificado com facilidade, o *overfitting* pode não ser perceptível durante o treinamento, mas falhar nos testes. Torna-se, portanto, fundamental buscar o equilíbrio entre esses dois cenários (IBM, 2021).

Outro método de otimização no processo de aprendizagem é o algoritmo ADAM (*Adaptive Moment Estimation*), aplicado em redes neurais. Ele ajusta os passos de aprendizagem com base em informações anteriores, como a média e a variância dos gradientes, permitindo que cada parâmetro do modelo aprenda em seu próprio ritmo. O algoritmo ADAM foi desenvolvido com base em outros métodos - AdaGrad e RMSProp - combinando seus pontos fortes (Kingma; Ba, 2015).

A inferência variacional (VI) é voltada à aproximação de distribuições de probabilidade, especialmente no contexto da estatística bayesiana, onde a inferência - dedução que parte de uma premissa para chegar a uma conclusão - se baseia na distribuição dos dados após a observação. Dada a complexidade dos cálculos exatos sobre essa distribuição, a inferência variacional surge como uma alternativa eficiente (Blei; Kucukelbir; McAuliffe, 2018).

O truque de reparametrização é uma técnica que separa a fonte de aleatoriedade dos parâmetros do modelo por meio da reformulação de uma variável aleatória como uma transformação parametrizada de outra, mais simples de amostrar. Em outras palavras, escolhe-se uma variável mais simples para gerar uma mais complexa. Essa abordagem é comum em autocodificadores variacionais, pois desloca a aleatoriedade para fora do modelo e permite calcular o gradiente dentro da expectativa (Sjölund, 2023).

Ademais, a retropropagação é um algoritmo de treinamento de redes neurais que permite o uso do gradiente descendente para ajustar seus pesos - multiplicadores que aumentam ou diminuem a contribuição de um neurônio em determinada camada. Calcula-se, de forma eficiente, o impacto da alteração dos pesos nos erros do modelo, por meio da aplicação da regra da cadeia (um princípio do cálculo diferencial) ao cálculo do gradiente da função de perda, determinando pesos maiores para boas predições, ou o contrário (IBM, 2024).

Algoritmos genéticos partem de princípios da evolução natural, trabalhando com uma população de soluções que evoluem ao longo das gerações. A cada nova geração, os indivíduos mais aptos são selecionados para reprodução, combinando suas características por meio de cruzamento e, ocasionalmente, passando por

pequenas mutações. A ideia é que essas soluções melhorem progressivamente, promovendo melhores resultados ao imitar o processo de seleção natural (Thede, 2004).

Para lidar com problemas de segurança e confiabilidade, uma das soluções adotadas é o algoritmo BFGS (*Broyden-Fletcher-Goldfarb-Shanno*), desenvolvido em 1970. Esse algoritmo é conhecido por empregar métodos de predição e monitoramento de falhas na saída, ao mesmo tempo em que combina métodos iterativos de direção conjugada e quasi-Newton. No entanto, a presença de valores incomuns nos dados prejudica seu desempenho (Xue; Zhang; Xiao, 2022).

Além dos métodos de otimização e inferência aplicados em redes neurais, também existem abordagens voltadas à tomada de decisão, como os sistemas baseados em regras (*rule-based systems*). Eles consistem em um conjunto de regras no formato se-então (*if-else*), desenvolvidas a partir do aprendizado com dados reais e fundamentadas em técnicas de aprendizado de máquina (Liu; Gegov; Cocea, 2016).

3.2. Arquiteturas

As redes neurais são modelos inspirados no funcionamento do cérebro e aprendem ajustando sua arquitetura ou os pesos das conexões, por meio de exemplos ou padrões de entrada e saída. De modo semelhante ao aprendizado biológico, que envolve mudanças nas conexões entre neurônios, as redes neurais melhoram seu desempenho ajustando ligações internas (Islam; Chen; Jin, 2019).

A arquitetura de redes neurais refere-se à forma como os neurônios artificiais são organizados em camadas e ao modo como interagem entre si. Existem dois tipos principais de arquiteturas: as redes diretas (*feedforward*), conhecidas pela sigla FFN, que percorrem as camadas em uma única direção; e as redes com retroalimentação (*feedback*), que podem seguir mais de uma direção, denominadas redes neurais recorrentes (RNN). Essas redes podem conter múltiplas camadas e são classificadas de acordo com o método de aprendizado e a função empregada. Para resolver diferentes tipos de problemas computacionais, diversos algoritmos de treinamento são aplicados (Ahamed; Akthar, 2016).

A rede neural *feedforward* (FNN) é considerada a mais simples, composta por camadas de neurônios que processam os dados em apenas uma direção: da entrada, passando por uma ou mais camadas ocultas, até a saída. Sua aplicação é comum em problemas de classificação e regressão, e o algoritmo mais utilizado para seu

treinamento é a retropropagação. Embora eficiente, essa arquitetura apresenta limitações, como a lentidão no processo de treinamento, o que impulsiona o desenvolvimento de alternativas mais avançadas (Nguyen et al., 2020).

Uma extensão das redes neurais diretas são as redes neurais recorrentes (RNN), que possuem conexões entre neurônios estendidas ao longo do tempo, aproveitando informações anteriores durante o processamento. Diferentemente das redes diretas, as redes recorrentes possibilitam que a saída de um neurônio seja reutilizada como entrada, concedendo uma espécie de memória para lidar com dados sequenciais. Essa arquitetura permite que o modelo aprenda padrões ao longo do tempo, em conjunto com operações matemáticas que ajustam os pesos das conexões (Lipton; Berkowitz; Elkan, 2015).

Outra arquitetura relevante é a Unidade Recorrente Fechada (*Gated Recurrent Unit*), derivada da rede neural recorrente, que possui a capacidade de lembrar ou esquecer informações ao longo do tempo utilizando mecanismos conhecidos como portas de *reset* e atualização. A porta de *reset* controla quando o modelo deve ignorar o estado anterior e se concentrar em uma nova entrada, enquanto a porta de atualização define a quantidade de informação a ser mantida no estado atual. Essa flexibilidade permite lidar com informações de curto e longo prazo (Cho et al., 2014). A Unidade Recorrente Fechada foi desenvolvida originalmente com o objetivo de superar o problema de dissipação do gradiente - *vanishing gradient problem* (Rehmer e Kroll, 2020). Nesse sentido, a arquitetura tem apresentado resultados promissores (Hu et al., 2018).

A dissipação do gradiente é um problema que pode ocorrer em redes neurais recorrentes durante o treinamento do modelo, por meio de métodos como o gradiente descendente ou a retropropagação. Nessa situação, os valores do gradiente tornam-se progressivamente menores à medida que se retrocede pelas camadas, tornando o aprendizado das camadas iniciais muito mais lento que o das finais, o que prejudica o desempenho geral do modelo. O problema pode evoluir a ponto de os parâmetros serem atualizados com valores quase irrelevantes, travando o aprendizado (IBM, s.d.).

Ao compreender as limitações das redes neurais recorrentes, outras abordagens podem ser exploradas, como os autocodificadores variacionais (VAE). Esses modelos são capazes de reduzir a dimensionalidade dos dados (número de atributos), organizando as camadas de forma empilhada. Reduz-se a quantidade de neurônios nas camadas ocultas, o que permite identificar e representar as principais

características dos dados de modo compacto, além de reconstruir os dados originais com maior precisão. Outros aspectos envolvem a incorporação de probabilidades na arquitetura para detectar anomalias, minimizando a necessidade de definir limites fixos para identificar comportamentos inesperados (An; Cho, 2015).

Prosseguindo com as diferentes arquiteturas utilizadas em redes neurais, destacam-se as camadas totalmente conectadas (*fully connected layers*), também chamadas de camadas densas. Essas estruturas são amplamente adotadas por sua versatilidade, conectando cada neurônio com todos os de camadas anteriores e seguintes, gerando novas combinações de informações. O número de dimensões pode aumentar, diminuir ou permanecer inalterado, e tais camadas, embora lineares, são ajustadas com funções que inserem não linearidade (Kalayci; Asan, 2022).

Diferentemente de modelos anteriores que analisavam palavra por palavra, um tipo avançado de rede neural para processamento de sequências são os transformadores, os quais consideram todas as partes de uma sequência simultaneamente, identificando com maior precisão informações relevantes e gerando respostas coerentes e contextualizadas (AWS Amazon, s.d.). A codificação posicional relativa (*Relative Positional Encoding* - RPE) é uma técnica nos transformadores que insere informações sobre a distância entre palavras diretamente no mecanismo de atenção. Ela pode ser implementada de duas maneiras: por meio de vetores que representam a posição relativa entre pares de palavras ou por valores escalares que ajustam o cálculo da atenção. Dessa forma, o modelo consegue reconhecer a posição das palavras em relação umas às outras (Chen; Varoquaux; Suchanek, 2023).

Os coeficientes mel-cepstrais (MFCCs) fornecem representações espectrais da fala considerando aspectos da percepção auditiva humana e são úteis por reduzirem a quantidade de informações necessárias a serem processadas pelas redes neurais, sendo adotados como entradas com o objetivo de prever um espectro limpo da fala (Razani et al., 2017).

3.3. Estruturas de dados

Estruturas de dados são formatos específicos utilizados para organizar, processar, recuperar e armazenar informações de forma eficiente em um computador. Seu objetivo é facilitar o acesso e a manipulação de dados, permitindo que esses processos sejam realizados rapidamente, ocupando menos espaço na memória. Existem diversos tipos de estruturas, tanto básicas quanto mais complexas, com

finalidades específicas. Em geral, classificam-se em dois grupos: estruturas lineares - organizadas em sequência, como listas, pilhas e filas - e estruturas não lineares, que apresentam relações mais complexas entre os elementos, como árvores, grafos e tabelas (Padhya; Yadav, 2023).

A pilha é uma estrutura de dados com acesso limitado, na qual os elementos são inseridos e removidos segundo o princípio *last in*, *first out* (último a entrar, primeiro a sair). As operações básicas consistem em inserir um novo item no topo (*push*) e remover o que está no topo (*pop*). Há um limite de capacidade, e, caso se tente adicionar um item com a pilha cheia, ocorre um estado de *overflow*. O contrário ocorre no *underflow*, quando se tenta remover um item de uma pilha vazia. Essa estrutura impõe uma ordem específica, na qual os primeiros itens inseridos só podem ser acessados após a remoção dos mais recentes (Rai, 2014).

Já a lista pode ser compreendida como uma coleção de elementos organizados em uma sequência linear, pertencentes ao mesmo tipo de dado, podendo aparecer múltiplas vezes. A lista mantém os itens em uma ordem definida e sistemática (Tripathy; Gantayat, 2012). A fila, por sua vez, realiza a inserção de elementos em uma extremidade e a remoção pela outra, como em uma fila comum de espera. O primeiro elemento a entrar é o primeiro a sair - *first in*, *first out* (Jain; Kumar, 2014).

A estrutura não linear de árvore organiza os elementos de forma semelhante a organogramas ou árvores genealógicas. Ela é composta por elementos chamados nós, sendo um deles o nó raiz, que origina a hierarquia. Cada nó pode ter outros nós conectados a ele, denominados filhos, formando subárvores. Esse funcionamento é recursivo, ou seja, uma única raiz já é considerada uma árvore, e a partir dela constroem-se árvores maiores conectando novos nós (Aho; Hopcroft; Ullman, 1983).

Um grafo é elaborado por dois conjuntos: nós finitos não vazios e seus pares de nós. Os grafos se dividem em duas categorias: grafos não direcionados, em que os pares de nós representam arestas sem direção (sendo, portanto, desordenados), e grafos direcionados, nos quais cada aresta é representada por um par ordenado de nós (Horowitz; Sahni, 1983).

As tabelas de dispersão (*hash tables*) são estruturas do tipo dicionário ou mapa, que armazenam objetos identificados por chaves. Associam cada dado a uma chave pertencente a um conjunto, ao qual se aplica uma função que determina a posição onde o dado será armazenado. Quando duas ou mais chaves são mapeadas para a

mesma posição, ocorre uma colisão. A função de dispersão busca minimizar essas colisões (Brass, 2008).

É importante destacar o papel dos conjuntos organizados de dados, conhecidos como *datasets*, que constituem coleções de itens resultantes de processos como medição, coleta, análise ou observação. Esses itens podem incluir números, imagens, sentenças ou até mesmo outros *datasets* (Alrashed et al., 2021).

Por fim, *arrays* são estruturas de dados amplamente utilizadas em diversas linguagens de programação, devido à sua simplicidade e eficiência. Permitem armazenar múltiplos elementos e oferecem acesso rápido e direto a qualquer posição. No entanto, embora o acesso aleatório pareça vantajoso, esse aspecto pode dificultar a leitura e a análise dos dados (Åkerblom; Castegren, 2024).

4. DESAFIOS TECNOLÓGICOS, CRIATIVOS E LEGAIS DA IA

São analisados os desafios da aplicação da Inteligência Artificial na geração musical, que envolvem questões como complexidade, representação de dados e escalabilidade. Em seguida, exploram-se as dificuldades em produzir música com expressividade, criatividade e naturalidade. Por fim, consideram-se as implicações legais da criação musical por IA.

4.1. Desafios tecnológicos

Um dos principais desafios tecnológicos na aplicação de Inteligência Artificial para a produção musical está na representação adequada dos dados musicais. Existem diversas opções disponíveis, como *waveform*, MIDI*, piano-roll* e partituras, cada uma com suas vantagens e limitações. O *waveform* mantém a expressividade sonora do som, mas exige grande capacidade computacional e não permite a extração direta de parâmetros como tons ou acordes. O formato MIDI possibilita acesso direto a propriedades como altura, duração e intensidade das notas, porém não inclui o áudio propriamente dito. A escolha da representação impacta diretamente a performance dos modelos de geração e análise musical, o que exige decisões complexas (Mycka; Mańdziuk, 2024).

Outro aspecto relevante é a escalabilidade dos modelos, pois a geração de músicas com longa duração e estrutura coerente demanda arquiteturas hierárquicas complexas com alta capacidade de memória. Redes neurais recorrentes (RNN), autocodificadores e transformadores têm sido adaptados para capturar padrões de longo prazo, mas ainda podem apresentar limitações para manter a consistência, especialmente na produção de música polifônica, em que múltiplas vozes precisam ser coerentes entre si em tempo e harmonia (Mycka; Mańdziuk, 2024).

Modelos probabilísticos, como as cadeias de Markov, normalmente usados para geração melódica, possuem dificuldades em criar peças com dependências musicais mais longas ou estruturadas, devido à sua limitação de memória e à consideração apenas de estados anteriores imediatos. Mesmo versões mais robustas, como o modelo oculto de Markov (HMM), não são suficientes para lidar com a complexidade harmônica de estilos como o coral de Bach ou o contraponto renascentista. Essas limitações dificultam a geração de composições musicais coerentes ao longo do tempo (Siphocly; Salem, 2021).

A disponibilidade de dados rotulados com emoções é relativamente limitada, assim como sua qualidade, o que dificulta o treinamento de modelos robustos de IA. Além disso, os rótulos emocionais tendem a ser subjetivos e inconsistentes entre avaliadores, comprometendo a confiabilidade. Outro fator relevante é a avaliação das músicas geradas: muitos sistemas não adotam critérios formais de análise e, quando o fazem, geralmente se limitam a julgamentos subjetivos dos próprios desenvolvedores, o que introduz vieses e compromete a validade das conclusões (Dash; Agres, 2024).

4.2. Desafios criativos

Segundo a análise de Civit et al. (2022), a maioria dos sistemas de geração musical apresentou falta ou dificuldade em alcançar uma expressividade emocional convincente - apenas 18 dos 118 sistemas observados consideraram esse aspecto. Essa limitação restringe o uso da IA em contextos nos quais a resposta afetiva é essencial. Outro desafio relevante é o excesso de previsibilidade ou a limitação estilística das músicas geradas, consequência direta das bases de dados utilizadas no treinamento. Boa parte dos sistemas teve sua criatividade comprometida pela mera reprodução de estilos, sem inovação.

Arquivos MIDI podem soar de forma mecânica, comprometendo a naturalidade da música e afastando-a da complexidade ou das nuances presentes em composições humanas. Capturar sutilezas e sentimentos torna-se, portanto, um obstáculo significativo para os modelos de IA. Ademais, a ausência de consenso sobre como avaliar a criatividade e a originalidade de um modelo dificulta a comparação entre abordagens distintas, inviabilizando a identificação de quais sistemas produzem os resultados mais satisfatórios (Mitra; Zualkernan, 2025).

A composição musical computadorizada busca emular a criatividade humana, representando um desafio considerável para a IA (Siphocly; Salem, 2021). Nessa perspectiva, inovar não se resume à geração de notas ou ritmos, mas exige que a música consiga ressoar emocionalmente com o ouvinte.

4.3. Desafios legais

Um desafio significativo é a ausência de métodos unificados capazes de proteger conteúdos gerados e que possam ser aplicados em diferentes modelos, considerando que, atualmente, os mecanismos de proteção são ajustados apenas a

arquiteturas específicas, o que evidencia a falta de adaptabilidade. Como marcas d'água não são aplicáveis nesse contexto, o desenvolvimento de estratégias eficazes para a proteção dos direitos autorais musicais torna-se um problema recorrente. Além disso, as legislações variam de país para país, dificultando ainda mais a implementação de um método universal (Mycka; Mańdziuk, 2024).

Outro obstáculo relevante é a potencial violação de direitos autorais por sistemas de IA treinados com grandes volumes de dados musicais existentes. De acordo com a lei de propriedade intelectual dos Estados Unidos, se um modelo de IA for alimentado com músicas protegidas por *copyright* sem as devidas licenças e, como resultado, gerar composições semelhantes ou derivadas, isso pode ser caracterizado como infração (Poland, 2023).

Mesmo ao considerar o conceito de uso justo (*fair use*), a modificação de uma obra existente por meio da IA não garante automaticamente a aplicação desse princípio - especialmente se houver finalidade comercial e não houver distinção substancial entre as criações. Além disso, os responsáveis pela criação do *software* generativo ou sistema podem ser responsabilizados pelas infrações, mesmo que o modelo não seja considerado uma pessoa jurídica ou física (Poland, 2023).

5. ESTUDO DE CASO

Este capítulo reúne três estudos de caso desenvolvidos de forma distinta na geração musical por meio do uso de inteligência artificial. Cada estudo adotou métodos próprios de composição e avaliação, contribuindo para uma compreensão mais ampla das possibilidades técnicas e criativas da aplicação desses modelos.

5.1. Technical, Musical, and Legal Aspects of an AI-Aided Algorithmic Music Production System (Kwiecień et al, 2024)

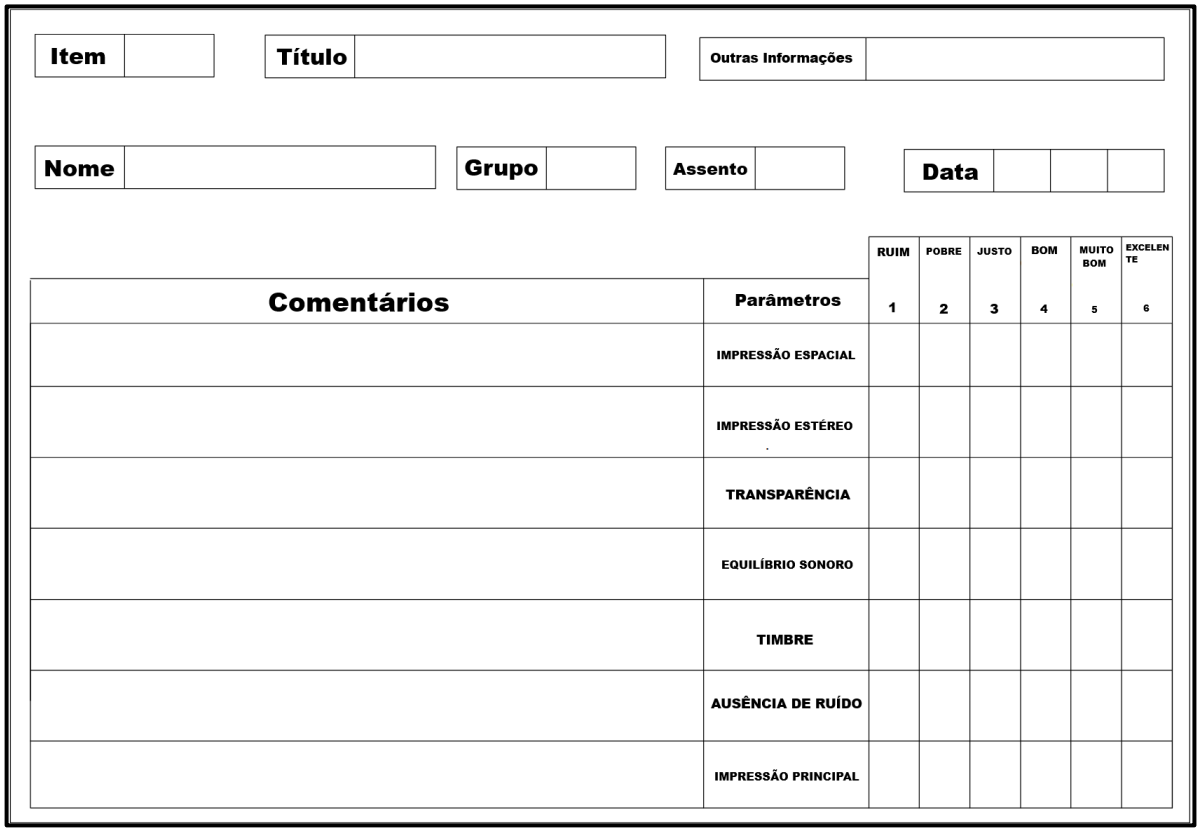
Um sistema composto por duas estruturas é apresentado em um dos estudos, sendo capaz de compor com base em parâmetros definidos pelo usuário em diversos estilos. Foram empregadas técnicas específicas para alcançar o resultado desejado, o qual foi avaliado tanto pelo próprio sistema quanto por um grupo seleto de ouvintes humano.

5.1.1 Metodologia

Desenvolveu-se um sistema gerador-crítico com diferentes algoritmos implementados, em que o componente denominado Gerador é responsável por compor músicas e criar arquivos de áudio em três estilos distintos: relaxante, eletrônico ou de dança. Após essa etapa, entra em ação o Crítico -uma rede neural treinada com os dados produzidos -que valida os resultados por meio da classificação das criações do Gerador, atribuindo uma pontuação de 1 para “correto” e 0 para “incorreto”. Em caso de classificação como incorreta, uma nova solicitação é enviada ao Gerador.

Posteriormente à classificação, a qualidade do conteúdo final produzido pelo sistema é avaliada por meio de testes auditivos realizados com um grupo de 20 ouvintes, utilizando métodos estabelecidos pela União Europeia de Radiodifusão (EBU), conforme ilustrado na Figura 1.

Figura 1 - Formulário de avaliação da qualidade do som



Item		Título		Outras Informações	
Nome		Grupo		Assento	Data

Comentários	Parâmetros	RUIM 1	POBRE 2	JUSTO 3	BOM 4	MUITO BOM 5	EXCELENTE 6
	IMPRESSÃO ESPACIAL						
	IMPRESSÃO ESTÉREO						
	TRANSPARÊNCIA						
	EQUILÍBRIO SONORO						
	TIMBRE						
	AUSÊNCIA DE RUÍDO						
	IMPRESSÃO PRINCIPAL						

Fonte: EBU TECH 3286, 1997

O sistema gerou 50 áudios para cada um dos gêneros musicais, dos quais 5 foram selecionados aleatoriamente por categoria. Adicionalmente, outras 5 gravações de artistas humanos também foram escolhidas, totalizando 30 áudios - 10 por gênero - com duração de 30 segundos, extraídos a partir do meio da gravação, e apresentados aos ouvintes para avaliação.

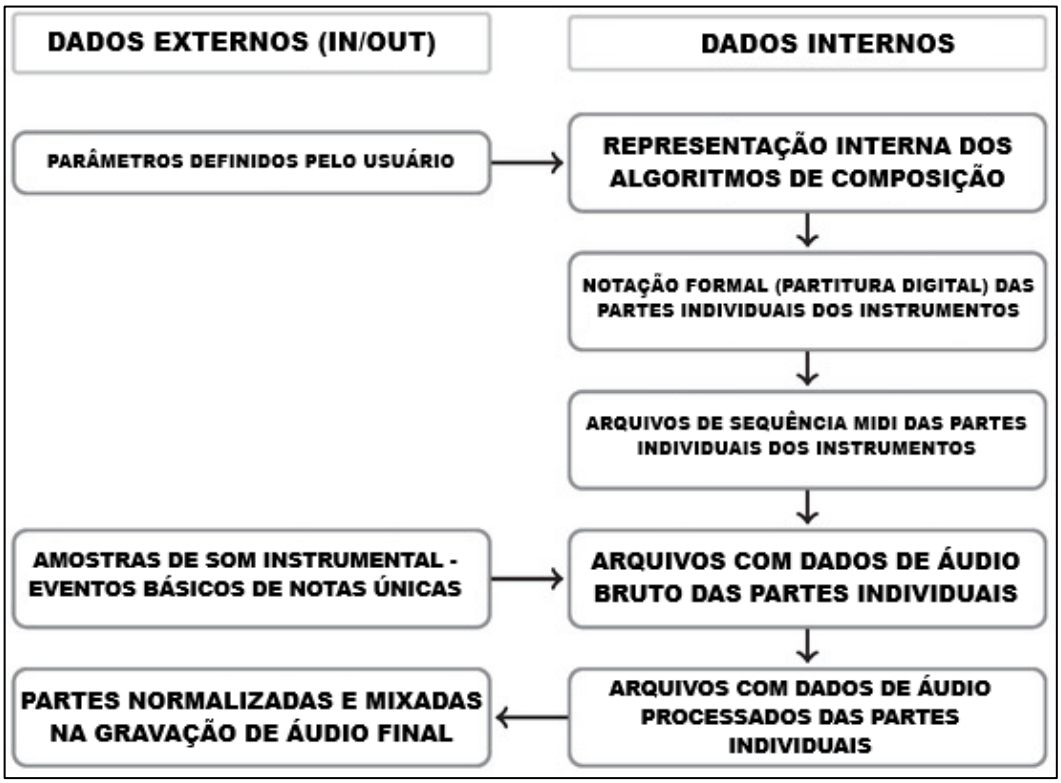
Os ouvintes avaliaram os áudios em uma escala de seis níveis: 1 para ruim, 2 pobre, 3 justo, 4 bom, 5 muito bom e 6 excelente, seguindo as recomendações da EBU. Caso a nota atribuída seja igual ou superior a 3, o áudio é aceito; do contrário, é rejeitado. Também foram coletadas informações sobre idade, experiência musical e possíveis limitações auditivas dos participantes..

5.1.2 O Gerador

O processo criativo de organização da estrutura sonora da música ao longo do tempo foi simulado pelo Gerador, partindo inicialmente de parâmetros definidos pelo usuário, conforme o fluxo de dados exibido na Figura 2. Esses dados foram estruturados em *arrays* (valores de um único tipo), listas (valores de tipos diversos),

mapas (pares chave-valor) ou conjuntos de regras aplicadas em algoritmos de composição baseados na teoria musical.

Figura 2 - Fluxo de dados de automação da produção musical

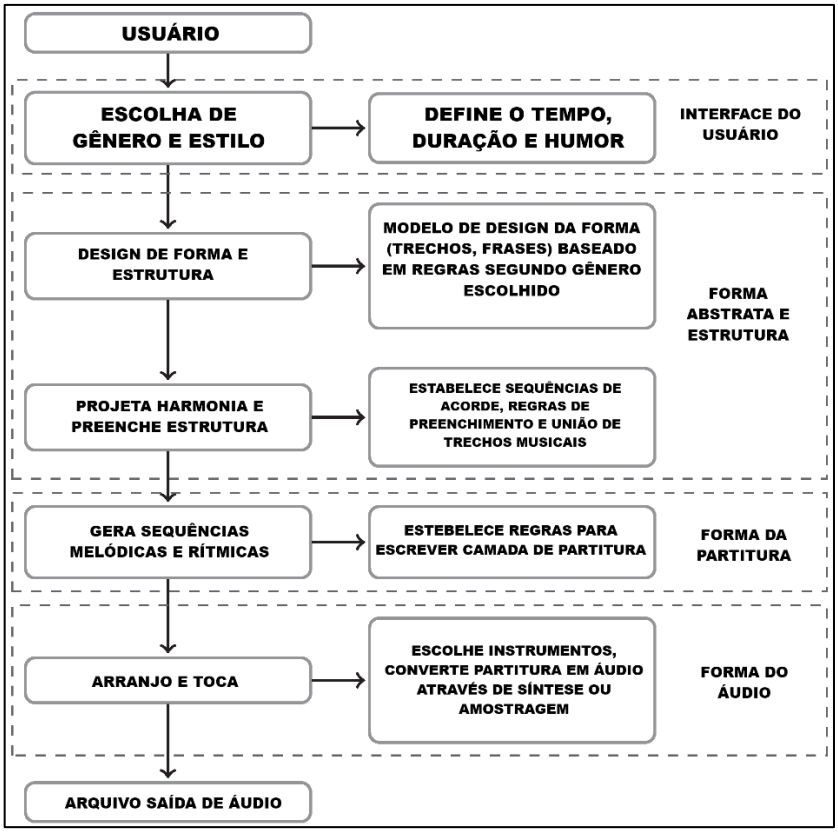


Fonte: Kwiecień et al, 2024

Ideias abstratas ou conceitos foram transformados pelos algoritmos em partituras simbólicas para cada instrumento digital. Em seguida, a interpretação feita pelo Gerador acrescentou articulação (modo como a nota deve ser tocada) e dinâmica (intensidade do volume). As partituras foram convertidas em dados no formato *Musical Instrument Digital Interface* (MIDI) e passaram por um processo de afinação das notas. O MIDI equivale a uma execução musical automatizada, resultando em um arquivo de áudio que passa pelas etapas de mixagem, balanceamento dos instrumentos, masterização e polimento, a fim de conferir um som profissional. As faixas podem, então, ser armazenadas digitalmente e reproduzidas pelo usuário.

A Figura 3 apresenta o processo de produção musical do Gerador. No início, o usuário define o gênero musical, a duração em segundos, o tempo em batidas por minuto (BPM), o tipo de emoção (de triste a feliz), o nível de excentricidade (de normal a excêntrico) e, por fim, o ajuste de afinação em hertz.

Figura 3 - Modelo do processo de geração musical



Fonte: Kwiecień et al, 2024

Um exemplo de como o sistema responde aos parâmetros definidos pelo usuário pode ser observado no controle de emoção: ao selecionar uma emoção triste, o sistema tende a empregar acordes, escalas e ritmos diferentes em comparação a uma emoção mais alegre. Já o parâmetro de excentricidade foi incorporado com o objetivo de introduzir variações musicais, permitindo o uso de escalas incomuns e controlando a probabilidade de ocorrência de sequências harmônicas atípicas.

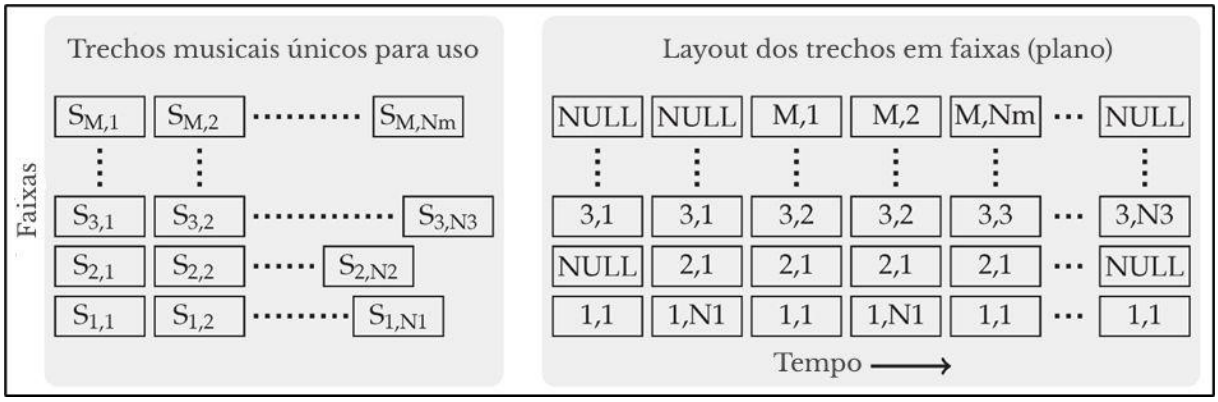
No caso dos *motifs*, a excentricidade gera melodias mais interessantes, porém difíceis de memorizar. Quanto maior esse valor, mais irregulares se tornam os ritmos, maiores as durações das notas com frequência e mais variadas as estruturas de bateria.

Na segunda fase - conforme ilustrado na Figura 3 - o sistema cria uma partitura formal, organizando a música como um compositor humano, em uma estrutura hierárquica com múltiplas camadas: harmonia, melodia, ritmo e instrumentação. As formas musicais também integram essa etapa, tanto no sentido de a produção se basear em estruturas comuns de um determinado gênero quanto na adoção de decisões guiadas por parâmetros definidos pelo usuário.

As decisões seguem probabilidades definidas conforme o gênero musical e consideram a relação entre harmonia e forma, como: quantidade de acordes em uma sequência, duração dos acordes em relação ao tamanho da frase musical e repetição. A forma musical é construída a partir de padrões e ordens adaptados de *trackers* (*softwares* de produção musical) e *grooveboxes* (dispositivos eletrônicos também utilizados para produção), ou seja, busca-se representar trechos musicais (*sections*) ao longo de faixas (*tracks*) em um plano bidimensional.

Esses trechos são segmentos recorrentes em compassos de uma faixa; cada faixa pode conter diversas seções exclusivas, organizadas em diferentes ordens, com bancos reutilizáveis desses mesmos elementos, permitindo repetição ou omissão. A forma musical - rondó, variação, verso-coro, entre outras - é materializada no *layout* apresentado na Figura 4, incorporando pausas, entradas, saídas de instrumentos e variações. A lógica *fuzzy* é aplicada nas decisões baseadas em probabilidade.

Figura 4 - À esquerda, mostra o conjunto de trechos musicais M para cada faixa única de áudio N em formato Sm,n. À direita segue um exemplo de como as faixas se organizam com o tempo.



Fonte: Kwiecień et al, 2024

A Antes de aplicar ritmos e melodias aos trechos musicais, considera-se a harmonia - uma sequência de distâncias entre tônicas, em semitons, a partir de uma nota inicial. Se a tônica for C (dó), por exemplo, 5 semitons acima está F e 7 semitons acima, G. Essa sequência pode ser repetida com durações variadas, e cada trecho de uma faixa é associado a um passo da sequência harmônica. Três tipos de sequência são usados, conforme o gênero musical: a mais simples, comum em músicas relaxantes e eletrônicas, repete um único acorde em uma escala; a intermediária, aplicável a todos os gêneros, alterna dois acordes entre graus da escala, como dominante, subdominante, mediante ou grau seguinte; por fim, a terceira

sequência emprega acordes mais longos, selecionados livremente conforme o gênero e influenciados pela emoção.

O Gerador utiliza três técnicas de IA, conforme a tarefa. A lógica *fuzzy* é aplicada no design da forma musical, na progressão harmônica, geração de *lead motifs* (motivos melódicos principais), linhas de baixo e faixas de acompanhamento. O algoritmo genético atua nos padrões de bateria e frases melódicas, enquanto o sistema baseado em regras (*rule-based system*, RBS) contribui tanto na criação de frases quanto em padrões de bateria, faixas de apoio, efeitos e mixagem.

A lógica *fuzzy* é a técnica mais empregada, usada para suavizar transições, como as variações nos parâmetros de humor e excentricidade, ou na criação da melodia principal, incluindo a introdução de inversões melódicas quando a linha melódica se aproxima do limite de alcance de um instrumento.

As duas últimas técnicas são responsáveis pela geração de trechos de partitura voltados às partes instrumentais, podendo operar de forma combinada com a lógica *fuzzy*. O sistema baseado em regras tem maior influência sobre a melodia principal e o design das frases, aplicando contrapontos para avaliar critérios como a existência de um único clímax, ausência de repetições excessivas e evitação de certas sequências. Trata-se de uma técnicas robustas de IA empregada em todo o sistema.

O algoritmo genético é o principal responsável pela geração das partes mais complexas, como as frases melódicas. Estas são construídas a partir de um conjunto limitado de motivos, garantindo coesão estrutural. Esses motivos são transformados e combinados de diversas formas para compor frases cuja parametrização é otimizada pelo algoritmo, com base nas regras contrapontísticas definidas pelo RBS. Na bateria, o algoritmo atua diretamente sobre representações em arrays, que indicam a presença ou ausência de som, utilizando tanto padrões inéditos quanto variações de estruturas pré-definidas, modificadas por operações como cruzamentos e mutações. Nas frases melódicas, porém, não há intervenção direta do algoritmo devido à hierarquia em múltiplas camadas: frases, motivos, notas, ritmo e dinâmica. Nesse caso, um conjunto de parâmetros é repassado a outros algoritmos responsáveis pela geração da partitura.

A partitura gerada é convertida em áudio por meio do MIDI, com dados extraídos diretamente da notação, e pelo *sampler*, que utiliza amostras sonoras, carregando o conteúdo de áudio em um arquivo. Após essa conversão, o sistema aplica efeitos sonoros, como modulação, filtros, compressão e distorção - sendo a

modulação e alguns filtros sincronizáveis ao ritmo. Os arquivos processados são mixados conforme a lista de faixas (*track list*), e tanto os efeitos quanto a mixagem são definidos pelo RBS.

O gerador e todos os algoritmos foram implementados em Python, utilizando as bibliotecas *os*, *random*, *time*, *math* e Mido para manipulação de dados MIDI, além da *pyo* para o processamento de áudio. As partituras digitais, tanto em formato gráfico quanto em MIDI, são geradas a partir do Lilypond, um interpretador de música escrita em texto. A síntese sonora é realizada no Fluidsynth, com amostras instrumentais em formato SoundFont, sendo o resultado final processado e mixado pelo programa SoX.

A escolha dos instrumentos em cada faixa é definida por um arquivo editável pelo usuário, que especifica onde cada instrumento será alocado, também em formato SoundFont. Cada faixa deve conter ao menos um instrumento, mas é possível atribuir múltiplos instrumentos a uma mesma faixa ou reutilizar o mesmo instrumento em diferentes faixas. Atualmente, o sistema opera com onze faixas distintas: duas melodias principais, uma faixa de ambiente e várias de acompanhamento.

5.1.3 O Crítico

O sistema proposto utilizou uma rede neural *feedforward* do tipo *multilayer perceptron* (MLP) como avaliador da qualidade das músicas geradas. O treinamento dessa rede consiste no ajuste dos pesos das conexões entre os neurônios.

Para desenvolver um método eficaz de classificação musical, dois aspectos são destacados: um conjunto de dados suficientemente grande para o aprendizado adequado e a escolha de boas características (*features*). Além disso, ressalta-se a importância da participação de especialistas na análise das músicas geradas, a fim de orientar o modelo e definir as principais etapas do processo de classificação. Essas etapas envolvem a criação dos conjuntos de dados para treinamento e teste, seguidas pela construção, compilação e treinamento da rede com os dados previamente organizados. Por fim, o modelo treinado é avaliado em um conjunto de testes para verificar sua capacidade de classificar corretamente as gravações geradas.

Inicialmente, as gravações são coletadas com avaliações feitas por especialistas. A partir delas, o sistema extrai descritores de áudio, por meio da biblioteca Essentia, organizados em três categorias: rítmicos (BPM), de baixo nível (como coeficientes cepstrais em frequência de mel-ressonância - MFCCs) e tonais

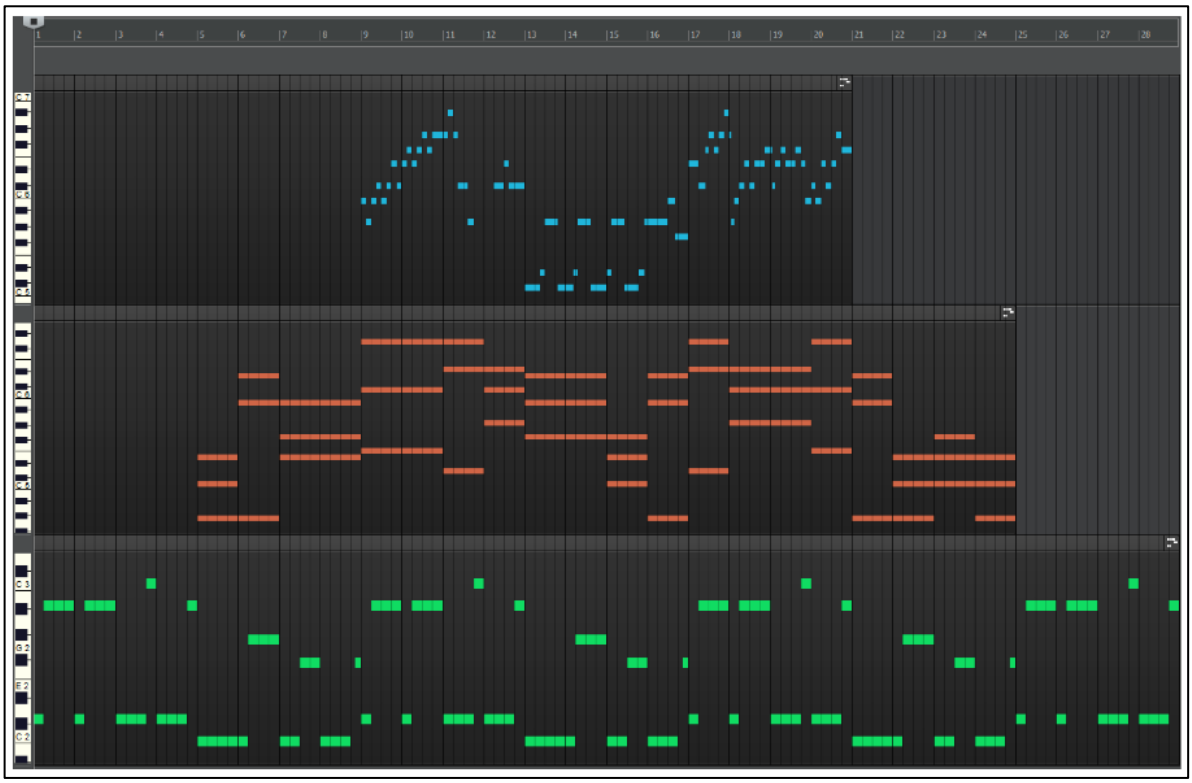
(como a taxa de mudança de acordes). Também são calculadas medidas estatísticas - média, mediana e desvio padrão - totalizando 54 variáveis.

A arquitetura da rede neural conta com 54 neurônios na camada de entrada. Já a quantidade na camada oculta varia conforme o gênero musical: para música de relaxamento, a melhor configuração foi com 42 neurônios; nos gêneros dança e eletrônico, os melhores resultados ocorreram com 35 e 22, respectivamente. A função tangente hiperbólica foi usada na camada oculta, e a logística, na camada de saída. A rede foi treinada com o algoritmo de otimização numérica BFGS (*Broyden-Fletcher-Goldfarb-Shanno*).

5.1.4 Resultados

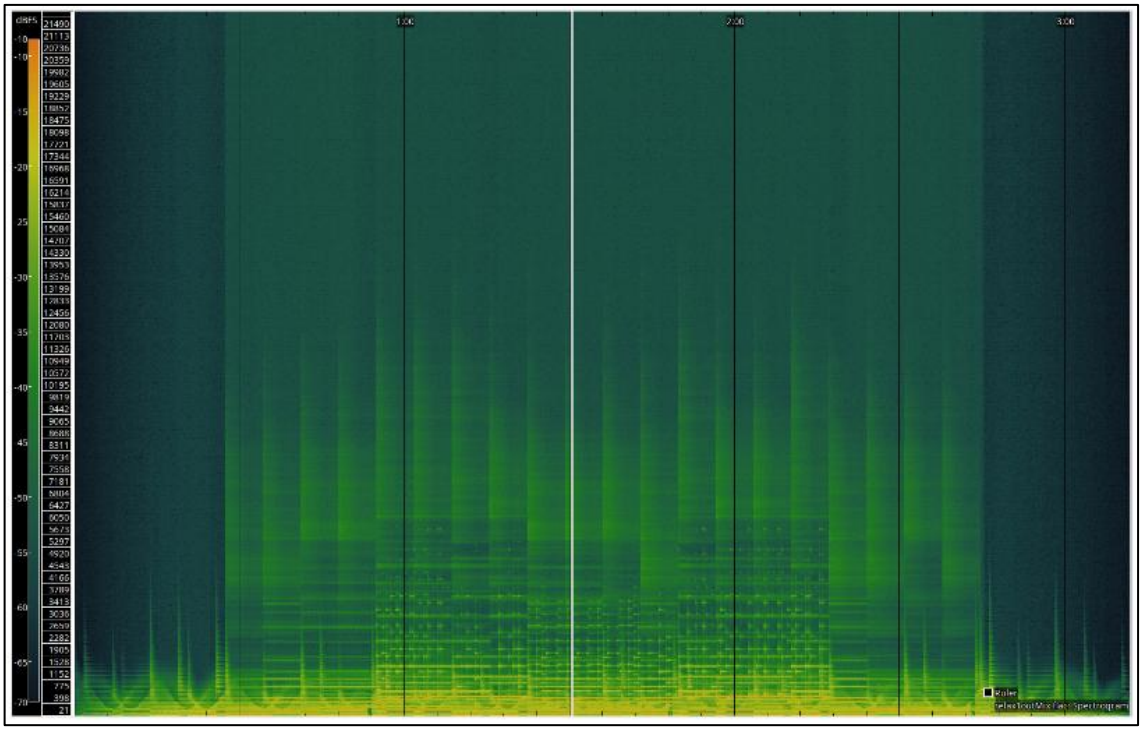
Nas Figuras 5, 6, 7 e 8 são observadas representações gráficas do gênero relaxamento e música eletrônica, produzidas pelo sistema.

Figura 5 - Visão de um *piano roll* de uma música gerada em relaxamento



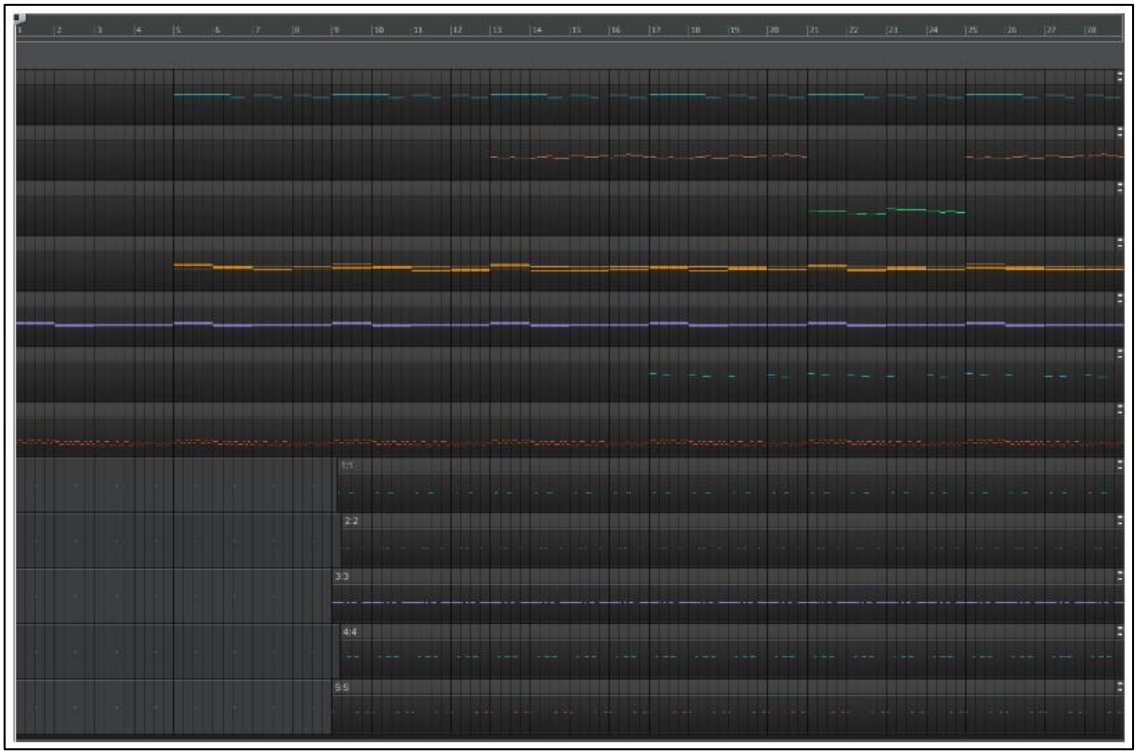
Fonte: Kwiecień et al, 2024

Figura 6 - Espectrograma da música de relaxamento



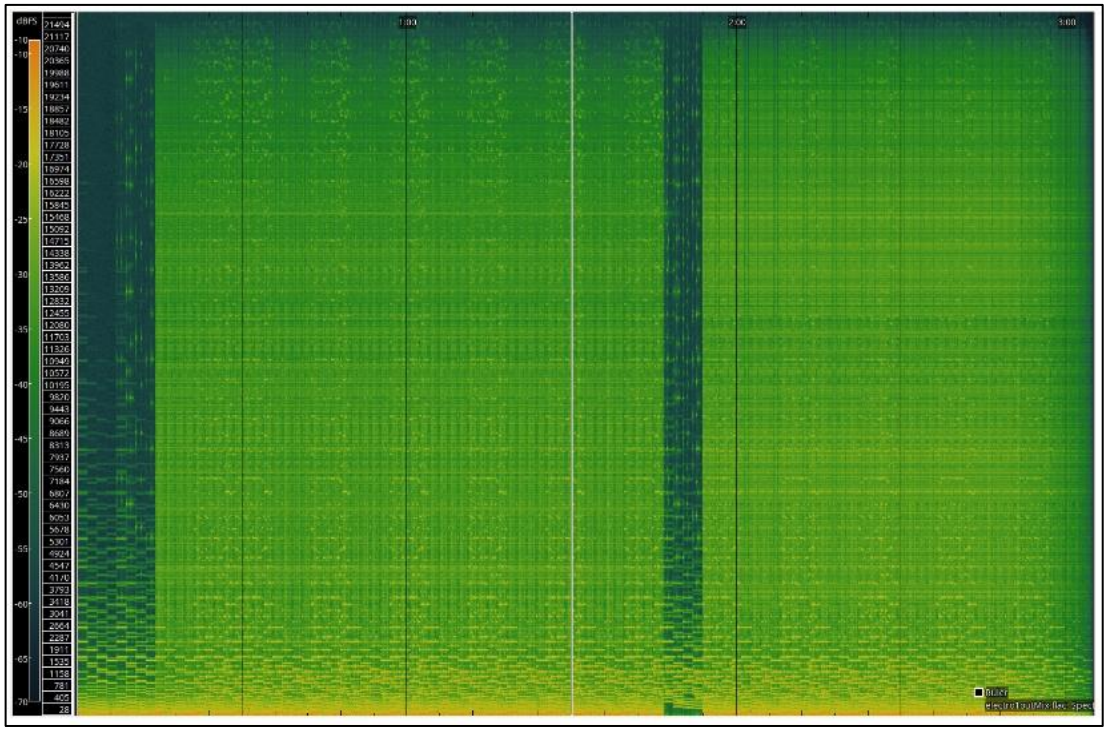
Fonte: Kwiecień et al, 2024

Figura 7 - Visão de um *piano roll* de uma música gerada em eletrônica



Fonte: Kwiecień et al, 2024

Figura 8 - Espectrograma da música eletrônica



Fonte: Kwiecień et al, 2024

Em uma música relaxante com trecho musical de 2 compassos, o Gerador produziu um conjunto de três motivos musicais que se apresentam na Tabela 1:

Tabela 1 - Conjunto de motivos usados na construção da primeira frase. Os ritmos são 2 - mínima (2 tempos), 4 - semínima (1 tempo), 8 - colcheia (meio tempo), 16 - semicolcheia (1/4 de tempo)

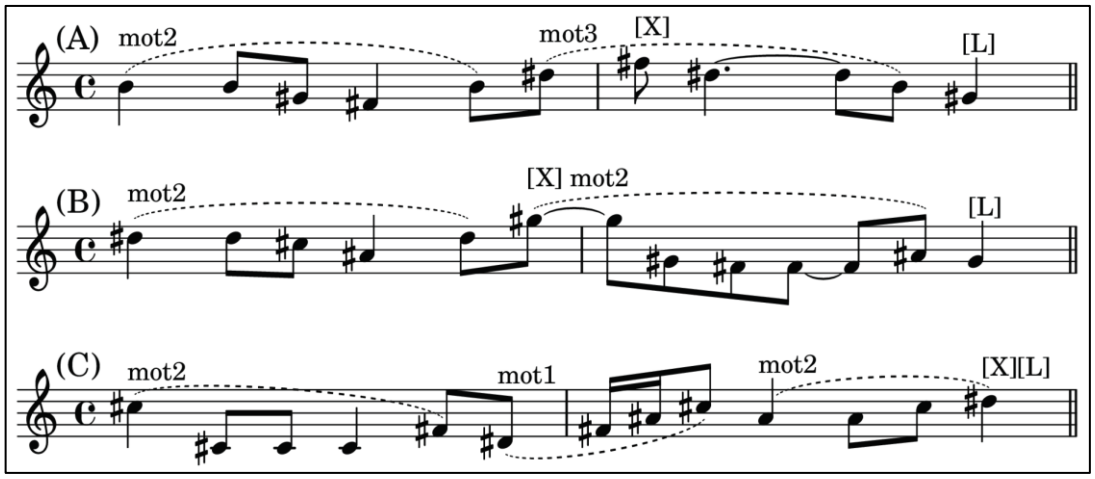
Nome	Duração da nota/ritmo	Intervalos
Mot1	8, 16, 16, 8	1, 2, 1
Mot2	4, 8, 8, 4, 8	5, 1, 1, -2
Mot3	8, 8, 2, 8	1, -1, -2

Fonte: Kwiecień et al, 2024

Os intervalos representam a diferença de altura entre uma nota e a seguinte, sendo expressos em número de passos conforme a escala musical utilizada. Se o valor for positivo, a próxima nota é mais aguda; caso contrário, é mais grave.

Com base nos três motivos, o Gerador criou três frases, ilustradas na Figura 9. Raramente os utiliza de forma literal, preferindo ajustá-los para adequar os intervalos e obter melhor adaptação. O algoritmo genético é responsável apenas pelas características melódicas.

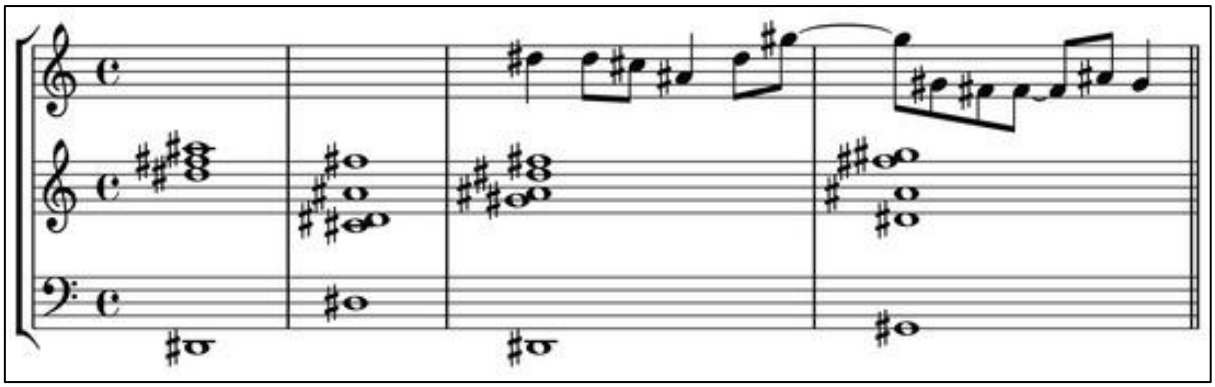
Figura 9 - 3 frases diferentes, de A a C produzidos pelo Gerador usando motivos da Tabela 1, o [X] marca o climax melódico e o [L] o fim da nota



Fonte: Kwiecień et al, 2024

A frase musical foi então complementada com outras faixas, como acordes e baixo, conforme ilustrado na Figura 10. Nessa estrutura, duas vozes também foram adicionadas: a primeira (L1) e a segunda (L2), que se alternam entre si e com o acompanhamento, na ordem L1-L2-Acc. Essa organização estabelece uma dinâmica de interação entre as vozes, promovendo maior variedade rítmica e harmônica, além de conferir fluidez ao conjunto.

Figura 10 - Faixas de acorde e baixo (acompanhamento)



Fonte: Kwiecień et al, 2024

O classificador (Crítico), para avaliar o sucesso do modelo, utilizou a separação dos três gêneros musicais em verdadeiro positivo (TP), verdadeiro negativo (TN), falso positivo (FP) e falso negativo (FN), com 2.000 arquivos de áudio por gênero. No caso da música de relaxamento, 540 de 940 amostras foram corretamente classificadas, resultando em uma taxa de acerto de 57%. Para o gênero dança, 312 de 659 amostras

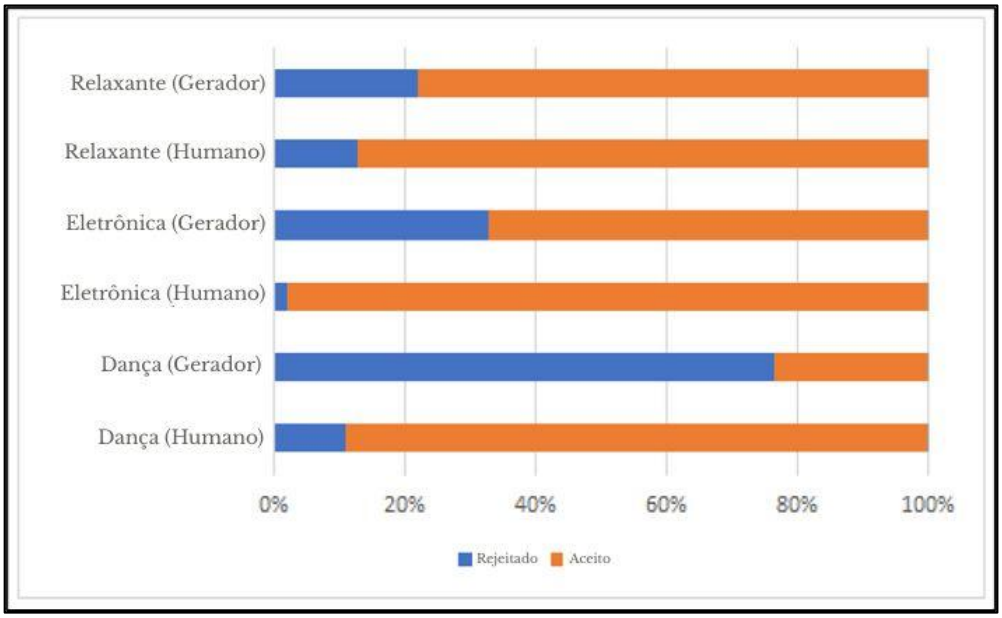
foram identificadas corretamente, com 47% de precisão; já na eletrônica, apenas 31 de 311 foram classificadas corretamente - cerca de 10%.

Nas classificações incorretas, observou-se que o gênero relaxamento obteve 765 acertos entre 1.060 amostras avaliadas, o que representa aproximadamente 72% de precisão. Comparado aos demais estilos, esse gênero apresentou maior dificuldade para o sistema. Por outro lado, a música de dança teve desempenho superior, com taxa de acerto de 89%, enquanto o gênero eletrônico atingiu o melhor resultado, com 99% de classificações corretas.

Com a fórmula (TP + TN) / (TP + TN + FP + FN), mediu-se a porcentagem de amostras corretamente classificadas. Os resultados foram: 0,65 para relaxamento, 0,75 para dança e 0,85 para eletrônica. Outro indicador utilizado foi a sensibilidade - proporção de positivos corretamente classificados -, calculada por TP / (TP + FN). As pontuações foram, respectivamente: 0,65 (relaxamento), 0,69 (dança) e 0,67 (eletrônica).

Nos testes de escuta, 55% dos ouvintes tinham entre 20 e 29 anos, 36% entre 30 e 39, e 9% entre 40 e 49 anos. Entre os 20 participantes, 55% possuíam experiência profissional em produção ou distribuição musical, 18% eram amadores e 27% tinham experiência mínima. Segundo a Figura 11, o gênero dança apresentou o menor nível de aceitação, com 24%, em contraste com o alto índice registrado para o gênero relaxamento, com 78%.

Figura 11 - Taxa de aceitação e rejeição das gravações criadas por humanos e por sistemas



Fonte: Kwiecień et al, 2024

5.1.5 Discussões

Como observado na Figura 11, percebe-se uma diferença de complexidade musical entre faixas classificadas como relaxantes e de dança, considerando a quantidade de camadas presentes em cada uma. Essa disparidade evidencia que o sistema teve melhor desempenho com músicas mais simples e apresentou dificuldade na elaboração de composições mais sofisticadas e amplamente aceitas. Ainda assim, os resultados foram satisfatórios, reforçando o potencial de aplicação prática.

Outro fator que contribuiu para a alta rejeição das músicas do gênero dança foi o processo de seleção aleatória a partir de um repositório comercial fornecido por artistas. Após essa etapa, uma faixa classificada como “dança” pode não soar assim para o ouvinte, levando a interpretações ambíguas quanto ao gênero ou até à rejeição total - o que impacta diretamente o desempenho do sistema Crítico.

Apesar dessas limitações, a proposta de utilizar um sistema híbrido - que não dependa exclusivamente de uma única técnica de IA - facilita o controle de detalhes específicos na produção musical, inclusive pelo próprio usuário, além de permitir a execução local sem necessidade de conexão externa.

No aspecto legal e ético, o sistema não infringe direitos autorais, pois gera composições sem copiar obras humanas, baseando-se apenas em algoritmos programados e lógica musical. O artigo também destacou a ausência de legislação específica para obras geradas dessa forma, considerando a novidade e a rápida evolução da área.

5.2. EmotionBox: a music-element-driven emotional music generation system using Recurrent Neural Network (Zheng et al, 2021)

O estudo propõe o modelo EmotionBox, voltado à geração de músicas com base em emoções. O sistema utiliza características como densidade sonora e histograma de altura, passando por uma etapa de avaliação com ouvintes humanos, na qual se identifica a emoção percebida em cada amostra.

5.2.1 Metodologia

O sistema desenvolvido para a geração de música emocional neste estudo é uma Rede Neural Recorrente (RNN), mais especificamente uma versão aprimorada conhecida como *Gated Recurrent Unit* (GRU). Essa variante busca reter informações de etapas anteriores de forma mais eficiente.

O modelo foi treinado com um conjunto de dados MIDI sem rótulos emocionais. Esse banco de dados, de código aberto (*open-source*), foi escolhido devido à escassez de conjuntos rotulados disponíveis publicamente, além de oferecer uma alternativa que dispensa a rotulagem para gerar música emocional.

Realizou-se uma avaliação comparativa entre modelos treinados com rótulos (*label-based*) e o modelo proposto, envolvendo um grupo de 26 participantes. Os indivíduos foram questionados sobre qual emoção sentiram ao ouvir as amostras, entre as opções: felicidade, tranquilidade ou paz, tristeza e tensão. O objetivo foi verificar se o modelo atual apresentava maior precisão na evocação emocional.

Foram selecionadas três amostras de seis segundos para cada emoção e para cada método, totalizando 24 amostras. Os participantes puderam pausar ou repetir as faixas, a fim de ouvi-las com clareza.

5.2.2 EmotionBox

O modelo gerador proposto utilizou como entrada (*input*) arquivos MIDI polifônicos, ou seja, com múltiplas notas tocadas simultaneamente, compondo melodia e acompanhamento. A codificação dos dados MIDI, de forma a serem compreendidos e treinados pela rede neural, consistiu em uma estrutura de eventos: *note-on*, *note-off*, *time-shift* e *velocity*. Como o conjunto de dados era exclusivo para piano, o intervalo de altura das notas musicais (*pitch*), que pode variar de 0 a 127, foi limitado entre 21 e 108 - abrangendo 88 notas, de A0 a C8.

Para cada nota, a dinâmica foi registrada no formato MIDI com valores de 0 a 127, sendo convertida para uma escala de 0 a 32 no parâmetro *velocity*, por conveniência. Os eventos foram descritos da seguinte forma: 88 eventos *note-on* (para iniciar novas notas), 88 *note-off* (para indicar o término da nota), 32 *time-shif*t (para representar avanços temporais entre 15 milissegundos e 1 segundo) e, por fim, *velocity*, responsável por ajustar a intensidade sonora - totalizando 240 possíveis eventos.

O modelo EmotionBox foi alimentado manualmente com duas características musicais, descritas como histograma de altura (*pitch histogram*) e densidade sonora (*note density*). O histograma de altura é um vetor (*array*) com 12 valores inteiros, correspondentes aos 12 semitons da escala cromática (oitava), representando a frequência de ocorrência de cada semitom na música. A Tabela 2 do artigo apresenta um exemplo prático desses dados extraídos.

Tabela 2 - Exemplo de um histograma de altura na escala C maior (Dó maior)

Nome da nota	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
Histograma	2	0	1	0	1	2	0	2	0	1	0	1
Distribuição de probabilidade	0,2	0	0,1	0	0,1	0,2	0	0,2	0	0,1	0	0,1

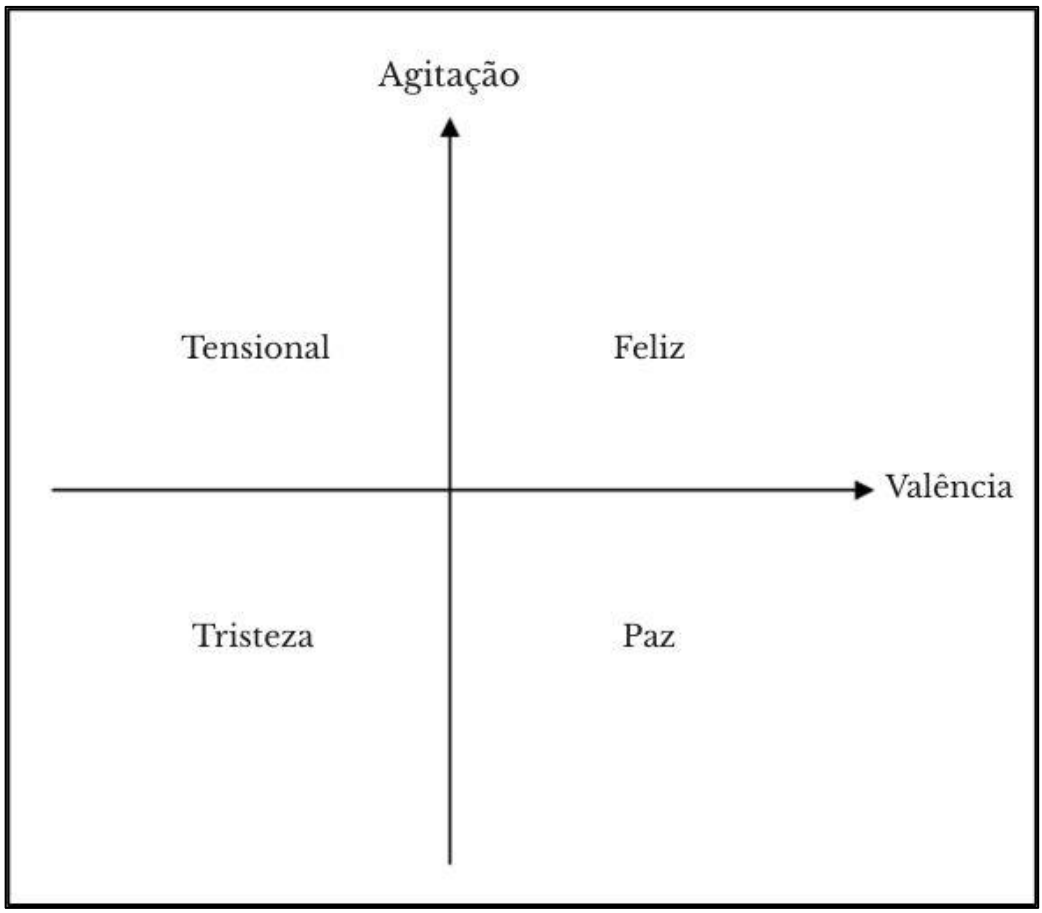
Fonte: Zheng et al, 2021

Os autores aplicaram a teoria musical segundo a qual notas sustenidas não são incluídas na escala do exemplo, sendo marcadas como 0 para indicar que nunca devem ser tocadas. Considerando que a nota C é a tônica, F a subdominante e G a dominante, seus valores foram atribuídos como 2, dobrando a probabilidade de ocorrência em relação às demais, por serem consideradas notas principais.

A densidade sonora representa a quantidade de notas tocadas dentro de uma janela de tempo - dois segundos, no caso do EmotionBox. Quanto mais notas, mais rápida ou agitada é a música; quanto menos, mais lenta ou calma. Ao modificar o histograma de altura e a densidade sonora, torna-se possível alterar o modo e o andamento da peça, provocando variações na percepção emocional.

Em relação às emoções, o artigo adotou o modelo de Russell, embora em versão simplificada, conforme ilustrado na Figura 12. Esse modelo é composto por quatro quadrantes e dois eixos: valência (*valence*), que varia do desagradável ao agradável, e agitação (*arousal*), do calmo ao intenso. Essa representação gráfica permite uma categorização mais clara e objetiva dos estados emocionais, facilitando o processo de geração musical.

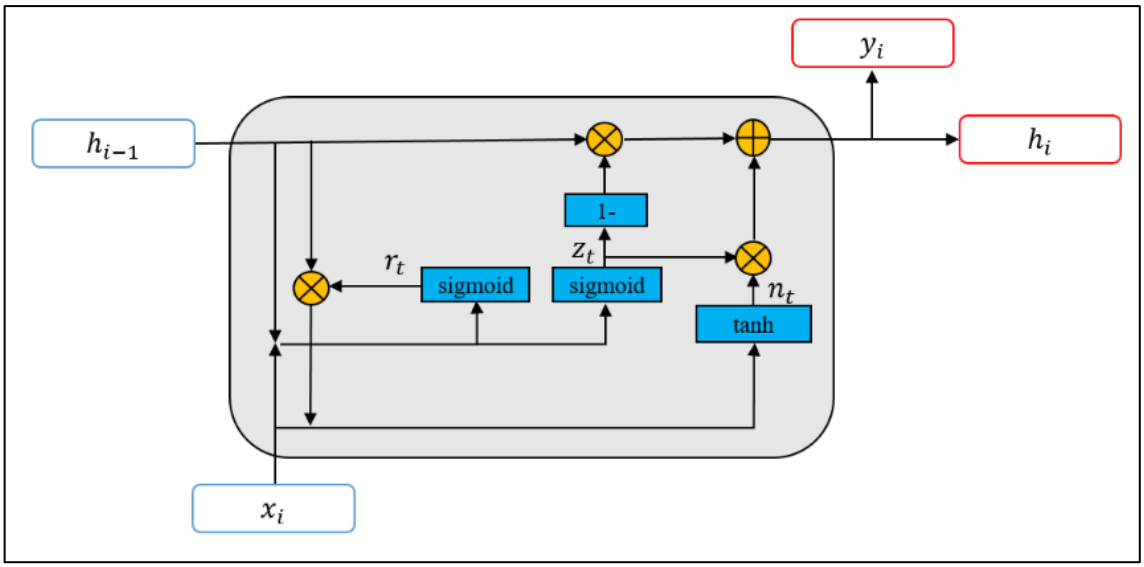
Figura 12 - Modelo simplificado de Russell



Fonte: Zheng et al, 2021

Em relação à arquitetura da rede neural, o GRU resolve um problema comum das RNN - redes neurais utilizadas para trabalhar com dados sequenciais - relacionado ao desvanecimento do gradiente durante a retropropagação, ou seja, a dificuldade em lembrar eventos antigos. No artigo, a Figura 13 ilustra o funcionamento do modelo GRU: X indica a entrada (*input*) e Y a saída (*output*) do mecanismo de portas (*gates*), enquanto h representa a memória - informação anterior ou atual. As portas R e Z correspondem, respectivamente, às funções de *reset* (esquecer informação) e *update* (manter informação).

Figura 13 - Representação da arquitetura do GRU

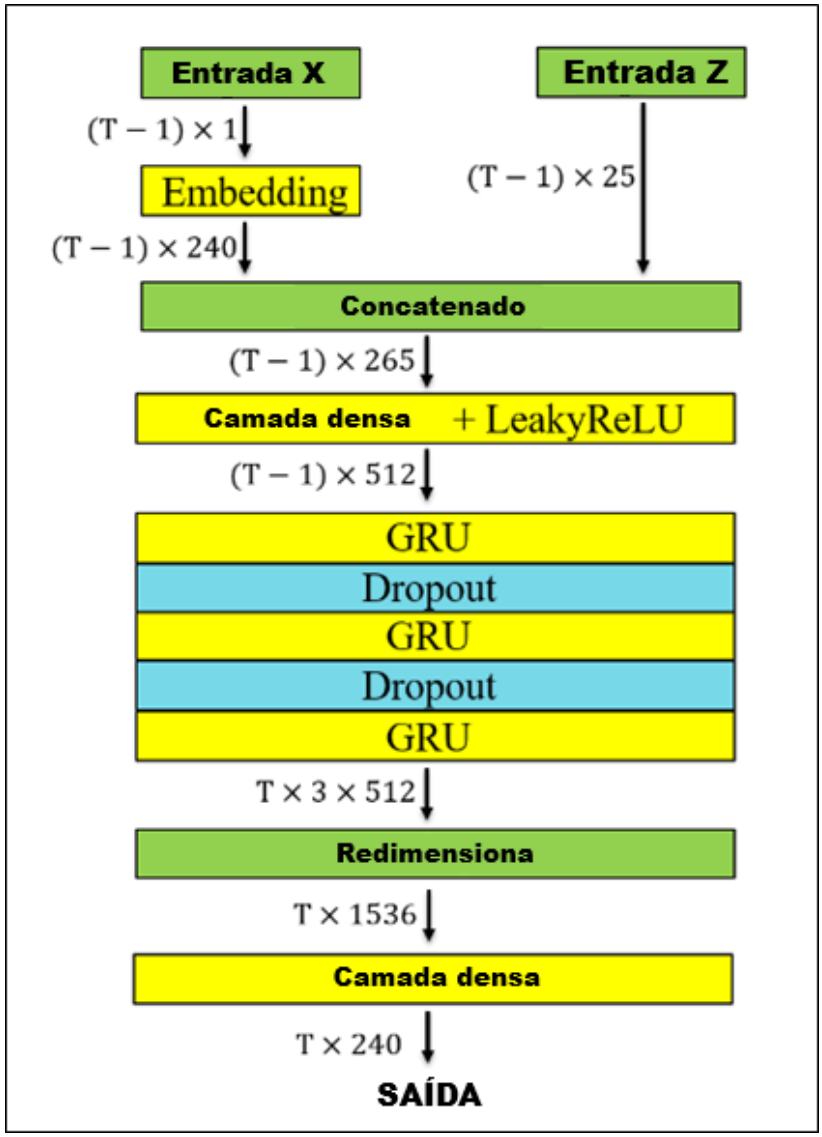


Fonte: Zheng et al, 2021

No contexto da arquitetura do EmotionBox, conforme apresentado na Figura 14, a entrada X representa os eventos de uma sequência na qual o último é removido (mascarado). O modelo, então, busca gerar a sequência completa de forma recursiva para calcular a diferença entre as duas sequências. A entrada X equivale ao tamanho T da sequência menos um (T - 1) e cada evento é convertido em um vetor de 240 linhas por 240 colunas, configurando uma camada de representação numérica.

A entrada Z consiste em um vetor formado pela soma do histograma de altura e da densidade sonora, cada um com 12 valores, acrescido de um vetor de zeros para aumentar a estabilidade da rede, totalizando tamanho 25. Essas duas entradas são concatenadas, resultando em um vetor de 265 elementos, que alimenta a camada densa (*fully connected layer*) com capacidade para 512 valores na saída, aplicando a função unidade linear retificada (ReLU). Essa função melhora o desempenho computacional ao transformar valores negativos em zero, mantendo os positivos.

Figura 14 - Diagrama de arquitetura do EmotionBox, a entrada X é sequência de eventos e a entrada Z o histograma de altura e densidade sonora



Fonte: Zheng et al, 2021

Em seguida, o vetor de 512 características é enviado a um GRU de três camadas, com *dropout* - técnica que desativa 30% dos neurônios nas duas primeiras camadas para evitar *overfitting*. A saída do GRU passa por uma camada densa com 240 unidades, gerando um novo vetor de dimensão T por 240. Essa saída representa a probabilidade de cada evento em cada passo, sendo então comparada com a sequência real por meio da função de perda de entropia cruzada, responsável pelo cálculo dos erros.

A geração da música inicia-se ao especificar um histograma de altura e densidade sonora. O primeiro evento é selecionado aleatoriamente, e os três

elementos são enviados ao modelo para criar novos eventos entre as 240 possibilidades de forma recursiva. Os autores mencionaram o risco de repetição excessiva caso os eventos com maiores probabilidades fossem sempre escolhidos; para evitar isso, definiram um limite entre 0 e 1. Para cada evento, um número aleatório é gerado nesse intervalo; se o valor ultrapassar o limite, o evento com maior probabilidade é selecionado (*greedy*); caso contrário, a escolha ocorre de maneira mais aleatória, caracterizando um processo estocástico (*stochastic*).

A abordagem para controlar o grau de incerteza ou aleatoriedade é denominada temperatura, um hiperparâmetro que configura o comportamento do algoritmo ou da rede neural, escalando (multiplicando ou dividindo) os valores de saída da última camada (*logits*) antes da aplicação da função *softmax*, que transforma esses valores em probabilidades. Temperaturas menores resultam em eventos previsíveis, enquanto temperaturas maiores geram eventos mais inesperados.

Para comparar com um modelo treinado por rótulos, os autores selecionaram um conjunto homogêneo de dados MIDI para piano, composto por 329 peças instrumentais de 23 compositores clássicos. Esse conjunto foi rotulado por Zhao et al. (2019) com as quatro emoções básicas e autorizado para uso no treinamento do modelo comparativo. Para extrair informações das notas dos arquivos MIDI, utilizou-se a ferramenta Pretty-MIDI, desenvolvida por Raffel & Ellis (2014).

Durante o treinamento, a sequência completa de eventos foi dividida em blocos de 200 eventos, com avanço de 10 eventos a cada iteração. Os modelos foram treinados utilizando o otimizador ADAM (*Adaptive Moment Estimation*), com função de perda de entropia cruzada e taxa de aprendizado de 0,0002, considerada lenta. Foram aplicadas 100 epochs e *batches* de tamanho 64, com a implementação dos modelos em PyTorch.

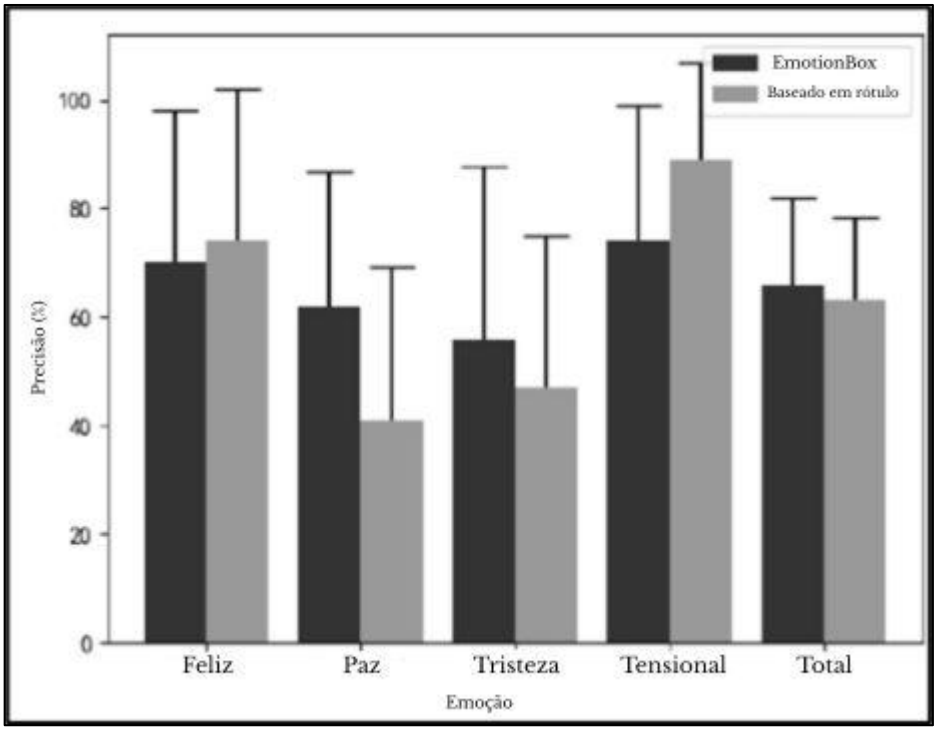
A arquitetura das redes neurais permaneceu idêntica, com a única modificação na entrada Z do modelo comparativo, que foi substituída pelo método baseado em rótulos, convertendo as variáveis das quatro emoções para o formato binário (*one-hot encoding*). No modelo baseado em rótulos, a rede neural foi treinada para mapear os tipos emocionais em uma música a partir de rótulos bem definidos.

5.2.3 Resultados

Conforme mostrado na Figura 15, o modelo proposto e o modelo baseado em rótulos apresentaram resultados semelhantes em precisão para classificação das

emoções. Em ambos, a emoção tensional teve o maior índice de reconhecimento, enquanto tristeza e paz apresentaram menor precisão, respectivamente no modelo proposto e no baseado em rótulos, embora o EmotionBox tenha superado o segundo em taxa de acerto.

Figura 15 - Acurácia média e desvio padrão pelos ouvintes em classificar as amostras por emoções



Fonte: Zheng et al, 2021

A Tabela 3, acompanhada do teste de postos sinalizados de Wilcoxon (*Wilcoxon signed-rank test*) para comparação de amostras, indicou significativa diferença entre os métodos para emoções tensionais e de paz. O modelo baseado em rótulos apresentou maior precisão em emoções tensionais, mas desempenho inferior em emoções de paz.

Tabela 3 - Comparação de cada emoção entre os dois modelos, valores menores que 0.05 apresentam uma diferença estatística significante

EmotionBox	Método baseado em rótulos	Valor P
Felicidade	Felicidade	0,4389
Tensional	Tensional	0,0026
Tristeza	Tristeza	0,3007
Paz	Paz	0,0111
Todos	Todos	0,2855

Fonte: Zheng et al, 2021

Nas Tabelas 4 e 5, analisou-se como as emoções foram percebidas pelos ouvintes. Destaca-se que ambas as amostras de músicas felizes, 28% e 23%, foram avaliadas como tensionais, indicando uma área comum de agitação, apesar da diferença em valência..

Tabela 4 - Resultados do *EmotionBox*

Amostras \ Classificação	Feliz	Tensional	Triste	Paz
Feliz	71%	28%	0%	1%
Tensional	17%	74%	5%	4%
Triste	1%	8%	56%	35%
Paz	8%	4%	26%	63%

Fonte: Zheng et al, 2021

No EmotionBox, músicas de diferentes níveis de agitação raramente foram classificadas em emoções opostas (por exemplo, feliz como paz), contrastando com o modelo baseado em rótulos, no qual 26% das músicas pacíficas foram classificadas como felizes e 28% como tensionais - desempenho considerado insatisfatório.

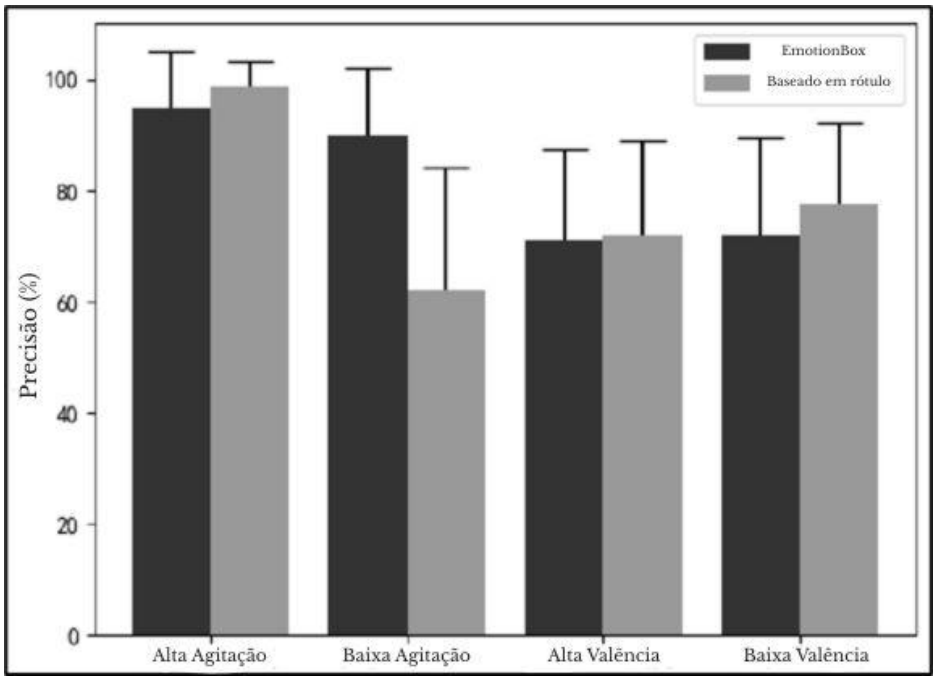
Tabela 5 - Resultados do modelo baseado em rótulos

Amostras \ Classificação	Feliz	Tensional	Triste	Paz
Feliz	71%	23%	0%	3%
Tensional	10%	90%	0%	0%
Triste	4%	18%	47%	31%
Paz	26%	28%	5%	41%

Fonte: Zheng et al, 2021

Durante o treinamento, caso o ouvinte percebesse a amostra musical com valência ou agitação semelhante à esperada, o resultado era considerado correto. Por exemplo, uma amostra feliz classificada como tensional era aceita, pois ambas pertencem à mesma área de agitação, conforme demonstrado na Figura 16.

Figura 16 - Acurácia média e desvio padrão pelos ouvintes sobre as amostras em agitação e valência



Fonte: Zheng et al, 2021

5.2.4 Discussões

Os resultados evidenciaram um desempenho do EmotionBox superior e mais equilibrado em comparação aos modelos dependentes de rótulos na geração de músicas com emoções de baixa agitação: tristeza e paz. Constatou-se maior eficiência temporal devido à ausência da categorização manual das emoções, indicando que a metodologia proposta reforça a importância da teoria musical na geração musical.

Para o modelo baseado em rótulos, a ausência da densidade sonora, segundo as experimentações dos autores, resultou em músicas com andamento mais acelerado. Esse fator dificultou a correta classificação das amostras de baixa agitação pelos ouvintes.

Apesar dos aspectos positivos, a valência foi identificada como um eixo desafiador, dada a insuficiência no índice de acertos, o que representa um obstáculo e abre espaço para futuras pesquisas sobre métodos de representação da valência visando aprimorar a expressividade. Outro ponto a ser considerado é a limitação da geração musical pela dependência exclusiva de um conjunto de dados para piano, restringindo a abordagem mais ampla de percepção emocional relativa a outros instrumentos e estilos musicais.

O EmotionBox utilizou um conjunto de dados de compositores clássicos cujas obras estão em domínio público, uma vez que mais de 70 anos se passaram desde o

falecimento dos autores. Dessa forma, o uso desses dados para fins de pesquisa foi considerado ético e não infringiu direitos legais, justificando também a exploração restrita a dados de piano.

Estudos como o de Kwiecień et al. (2024) incluíram critérios de avaliação mais abrangentes nos testes subjetivos, diferentemente do presente artigo, que careceu de análises sobre a qualidade musical. Uma música gerada pode expressar emoção, porém ser percebida pelos ouvintes como desagradável ou apresentar elevado nível de satisfação.

5.3. Music Generation and Classification of 8-Bit Tracks Using Variational Autoencoder and Music Transformer (Zhuang et al, 2025)

O estudo integra duas arquiteturas para geração e classificação de músicas em estilo *8-bit*, a partir de um conjunto de dados da Nintendo, destacando o uso de métricas quantitativas para avaliar a performance do sistema.

5.3.1 Metodologia

Foram utilizados os modelos autocodificador variacional (VAE) e *Music Transformer* para um sistema automatizado de geração e classificação. O VAE resumiu informações MIDI complexas de uma música em representações simplificadas, preservando os principais elementos musicais, reduzindo redundâncias e permitindo gerações coerentes por meio do espaço latente - representação abstrata dos dados, similar a um conjunto de recursos.

O *Music Transformer* foi escolhido por sua capacidade avançada de lidar com sequências, compreendendo relações entre notas, padrões e estruturas graças ao mecanismo de autoatenção (*self-attention*) e à técnica de codificação posicional relativa, que identifica a distância entre as notas em vez de sua posição absoluta. O modelo complementa o VAE ao usar as informações extraídas para gerar música de forma coerente no estilo *8-bit*.

A base de dados utilizada foi a NES-MDB (*Nintendo Entertainment System Music Database*), composta por 5.278 músicas de 397 jogos e 296 compositores, juntamente com a biblioteca *pretty_midi*. Os arquivos MIDI foram pré-processados em sequências de tamanho fixo, com 200 passos de tempo, e codificados em *one-hot* para representar 88 notas únicas (de A0 a C8). Aplicou-se truncamento ou preenchimento nas sequências, conforme seu tamanho, para padronização, além da

divisão dos dados em conjuntos de treinamento, validação e teste (80:10:10). A geração dos arquivos MIDI foi avaliada usando o *software Ableton Live 12 Suite*, que inclui compasso, notas iniciais e configuração das faixas.

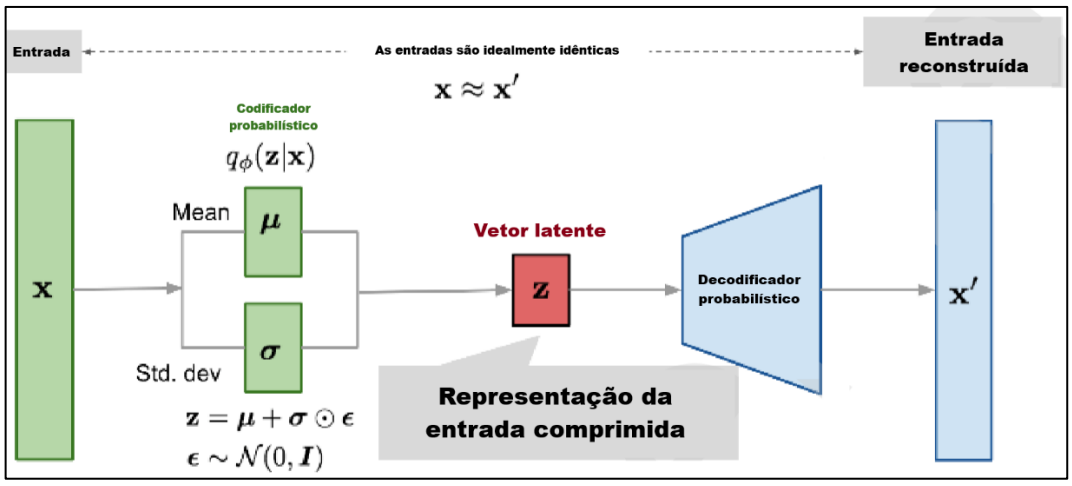
A avaliação do sistema baseou-se em métricas quantitativas e técnicas de visualização. As métricas incluíram: acurácia, proporção de acertos do modelo sobre o total de previsões; precisão, proporção de verdadeiros positivos entre todos os positivos; recall, proporção de acertos em relação ao total de ocorrências reais; *F1-Score*, média harmônica entre precisão e recall; e ROC-AUC, que mede a capacidade do modelo de distinguir classes por meio da curva ROC (*Receiver Operating Characteristic*), representando a taxa de verdadeiros positivos contra falsos positivos, e da área sob essa curva (*Area Under the Curve*), cujo valor ideal é próximo de 1.

5.3.2 VAE e *Music Transformer*

O autocodificador variacional (VAE) é um modelo generativo baseado em redes neurais que aprende a representar dados latentes (não visíveis) enquanto reconstrói a entrada. Ele é composto por um codificador, que transforma os dados de entrada X através de múltiplas camadas de rede neural direta (*Feedforward Neural Network*) em duas saídas: média e logaritmo da variância; o espaço latente, que representa os dados por distribuições probabilísticas e permite amostragens aleatórias - cuja retropropagação é viabilizada pelo truque de reparametrização, uma reorganização matemática das amostras em uma variável latente Z - e é regularizado pela divergência de Kullback-Leibler na função de perda, organizando o espaço latente; e o decodificador, que reconstrói a entrada X a partir da variável Z. A arquitetura do VAE está ilustrada na Figura 17, dividida pelos autores em quatro etapas, conforme descrito.

No modelo VAE, a função de perda é constituída pelo Limite Inferior da Evidência (ELBO), uma equação com dois termos que busca equilibrar a reconstrução dos dados de entrada e a regularização do espaço latente. O primeiro termo avalia o desempenho do modelo na semelhança entre a entrada X e a saída reconstruída, enquanto o segundo corresponde à divergência de Kullback-Leibler, que quantifica a diferença entre duas distribuições probabilísticas (aprendida e normal padrão).

Figura 17 - Arquitetura do VAE, a entrada X passa pelo codificador, segue em uma variável latente Z pelo truque da reparametrização e é reconstruído pelo decodificador. No final a função de perda minimiza erros de reconstrução e alinha o espaço latente



Fonte: Zhuang et al, 2025

O *Music Transformer* é um modelo de Inteligência Artificial baseado na arquitetura *Transformer*, adaptado especificamente para o processamento de sequências de dados musicais. O mecanismo de autoatenção permite ao modelo analisar a relação entre as notas, mesmo quando distantes entre si, contribuindo para o reconhecimento de progressões harmônicas, repetições melódicas e temas musicais recorrentes.

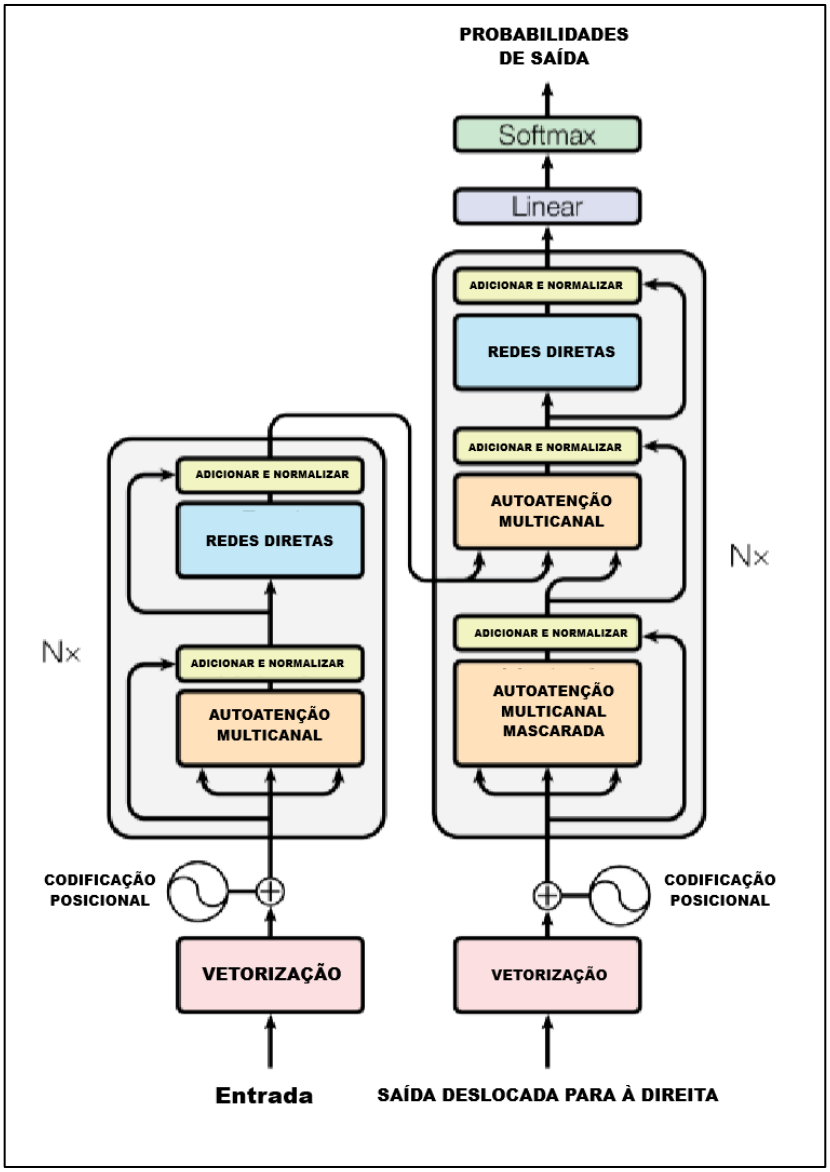
A autoatenção no *Music Transformer* calcula a relevância de cada nota em relação às demais por meio de uma equação que envolve os elementos da matriz de consulta (*Query* - Q), representando as perguntas, e da matriz de chave (*Key* - K), que compara essas informações com as consultas por multiplicação. Em seguida, aplica-se a função *softmax* para transformar os valores em pesos. O valor (*Value* - V) constitui, então, a saída do modelo.

O *Music Transformer* foi treinado utilizando a função de perda de entropia cruzada, e sua arquitetura é apresentada na Figura 18. O processo inicia com uma camada de entrada que transforma cada sequência de notas em formato MIDI em vetores numéricos, por meio da camada *embedding* com codificação posicional. Em seguida, uma camada de autoatenção multicanal permite ao modelo analisar várias relações musicais simultaneamente, otimizando sua capacidade de processar estruturas complexas.

Posteriormente, é aplicada a codificação posicional relativa. A saída da camada de autoatenção passa por transformações não lineares realizadas pela camada

feedforward, com o objetivo de aprimorar a detecção de padrões complexos em sequências musicais. Na última etapa, o decodificador com autoatenção mascarada (onde posições futuras da sequência são ocultadas) permite a geração musical nota por nota, com base nas notas anteriores ou atuais.

Figura 18 - Diagrama de Arquitetura do Transformador

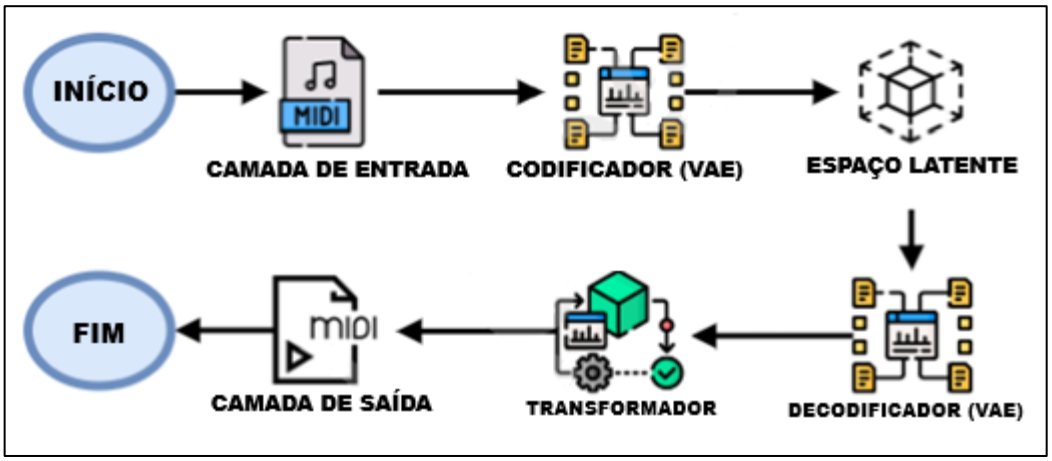


Fonte: Zhuang et al, 2025

O conjunto de dados NES-MDB foi utilizado tanto para análise quanto para geração de músicas em *8-bits*, sendo também um recurso essencial em estudos de musicologia computacional e inteligência artificial. Os arquivos MIDI foram interpretados e analisados pela biblioteca *pretty_midi*, que extrai informações como altura das notas e duração.

A integração entre o autocodificador variacional (VAE) e o *Music Transformer* ocorre por meio do vetor Z gerado pelo VAE, que serve como entrada para o *Transformer*, o qual, então, gera sequências musicais nota por nota. A arquitetura desse sistema é ilustrada na Figura 19, dividida em seis partes: camada de entrada, codificador VAE, espaço latente, decodificador VAE, *Transformer* e camada de saída.

Figura 19 - Diagrama da arquitetura do modelo de integração



Fonte: Zhuang et al, 2025

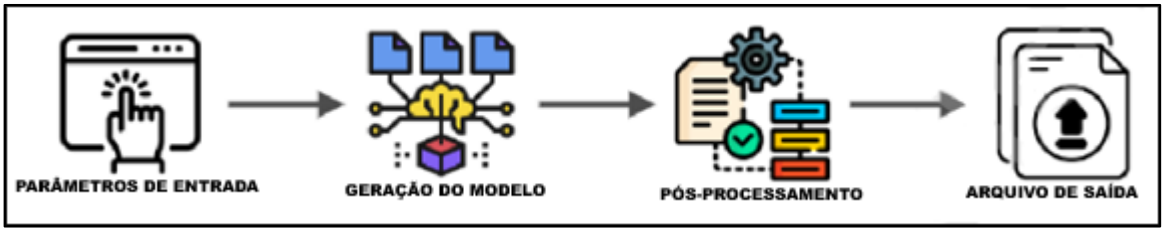
O treinamento do modelo foi dividido em quatro etapas: processo de treino, cálculo da perda, retropropagação e monitoramento da convergência. O treino foi conduzido utilizando a técnica de Gradiente Estocástico Descendente com *Minibatch*, associada ao otimizador ADAM e aplicação de 16 *batches*, ao longo de 5 *epochs*. No cálculo da perda, mediu-se a diferença entre a sequência prevista e a original (perda de reconstrução), alinhando a distribuição aprendida à normal padrão (perda de regularização). A perda total foi obtida pela soma desses dois termos.

Durante a retropropagação, os gradientes da perda total foram calculados via diferenciação automática, sendo propagados de volta para o VAE e o *Transformer*. O otimizador ADAM operou com taxa de aprendizado de 0,001, assegurando uma convergência estável. A perda de reconstrução e a divergência de Kullback-Leibler foram monitoradas separadamente, a fim de avaliar a convergência e identificar possíveis casos de overfitting ou underfitting.

A etapa final do sistema é a conversão da saída em um arquivo MIDI. Como as notas não são geradas diretamente, mas sim como uma sequência de tokens (eventos musicais), aplica-se a decodificação e a conversão para o formato “.mid” com o auxílio da biblioteca *pretty_midi*, conforme demonstrado na Figura 20. O sistema oferece ao

usuário opções de personalização do arquivo gerado, como o tipo de trilha - podendo ser um único instrumento ou múltiplos instrumentos distribuídos em várias trilhas -, o número de compassos (calculado com base na quantidade de batidas) e as notas iniciais, que guiam a tonalidade e os motivos melódicos.

Figura 20 - Processo de geração de arquivo MIDI

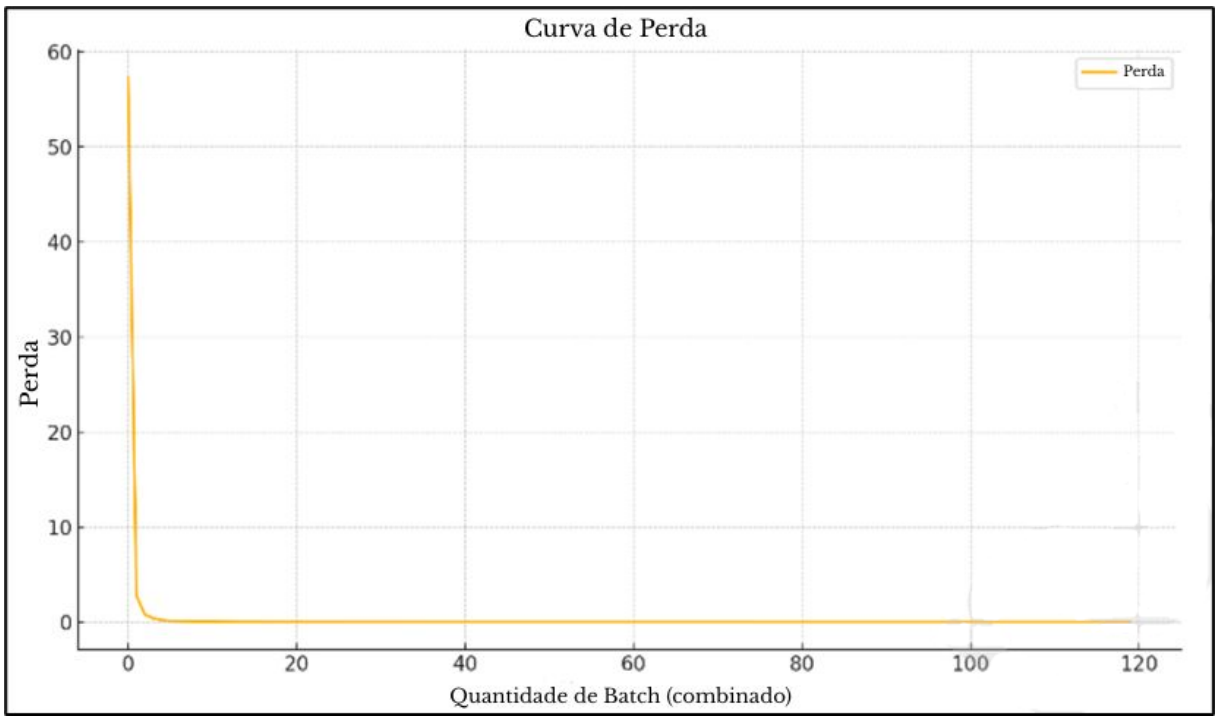


Fonte: Zhuang et al, 2025

5.3.3 Resultados

No âmbito dos resultados de classificação de notas musicais, a curva de perda do modelo durante o treinamento demonstrou, conforme a Figura 21, um aprendizado rápido e boa convergência. O eixo horizontal representa os *batches*, enquanto o eixo vertical indica a perda, que decai rapidamente.

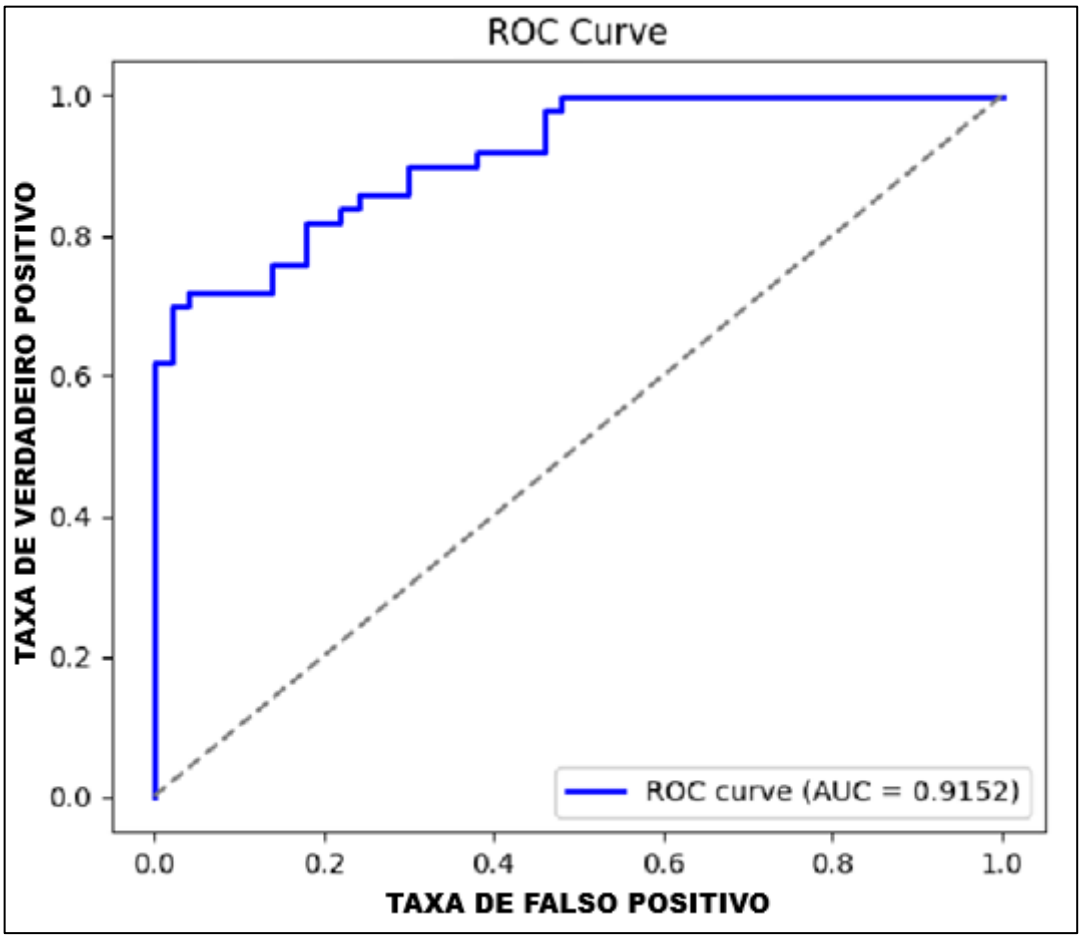
Figura 21 - Curva de perda do treino do modelo



Fonte: Zhuang et al, 2025

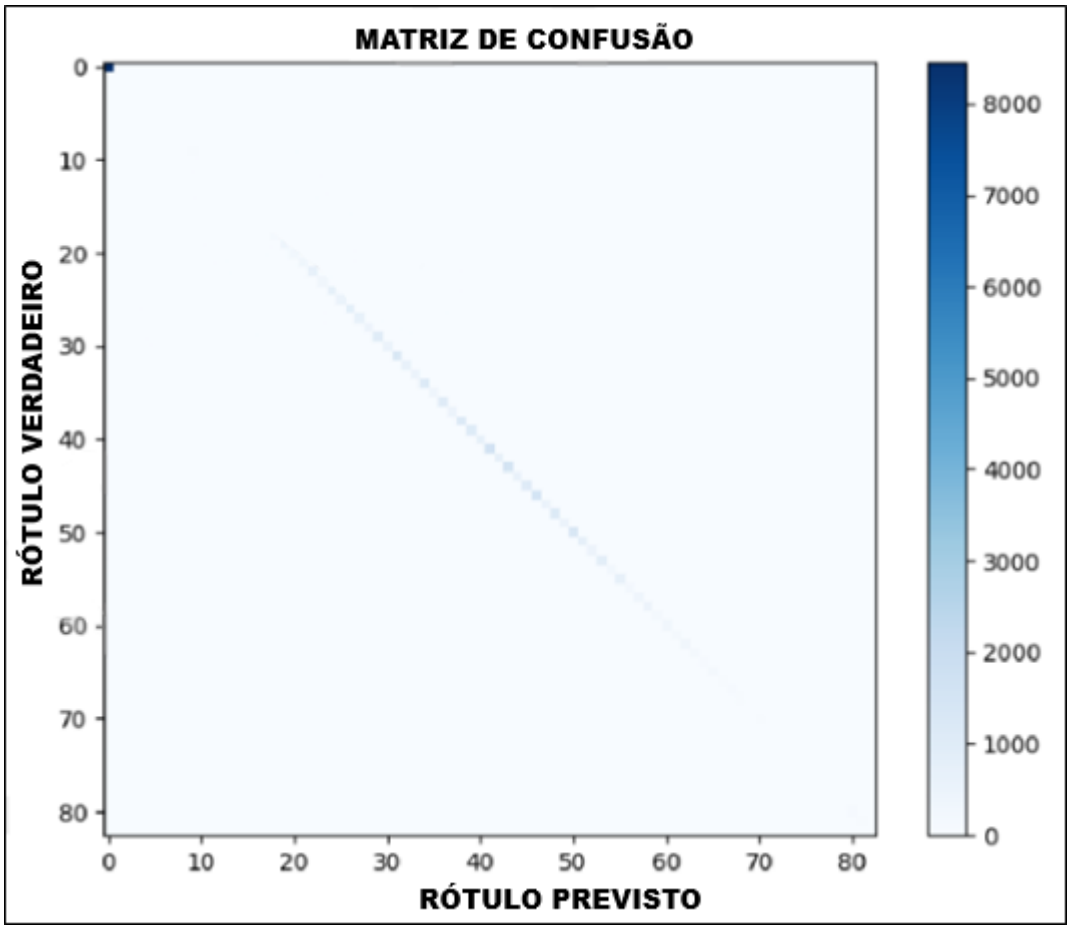
A matriz de confusão - tabela composta por valores de verdadeiro positivo (TP), verdadeiro negativo (TN), falso positivo (FP) e falso negativo (FN), na qual previsões fora da diagonal são consideradas incorretas - apresentou uma acurácia de 92,55% e uma pontuação ROC-AUC de 0,9152, considerada quase perfeita, conforme ilustrado na Figura 22. Já a Figura 23 evidenciou a robustez do modelo, visto que a maioria dos dados se concentrou na diagonal, com algumas notas ultrapassando 8.000 ocorrências.

Figura 22 - Curva ROC



Fonte: Zhuang et al, 2025

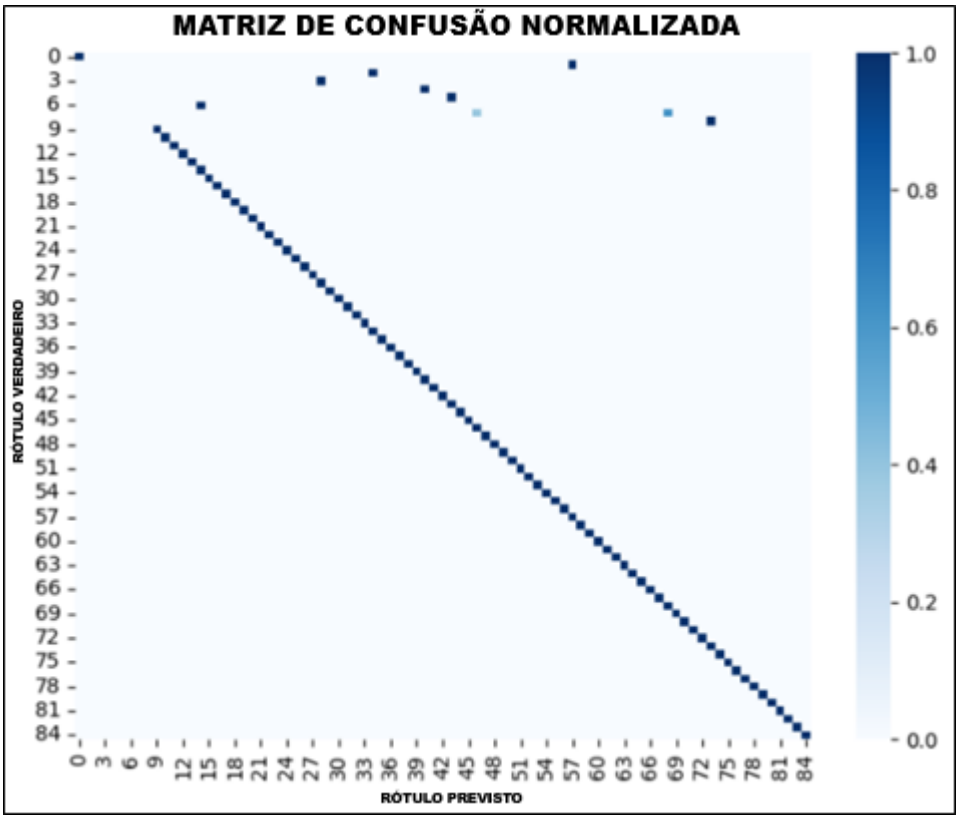
Figura 23 - Matriz de confusão



Fonte: Zhuang et al, 2025

Para aprimorar a visualização da precisão do modelo quanto à proporção entre acertos e erros, os autores normalizaram a matriz de confusão, como mostrado na Figura 24. Essa versão apresenta, por meio da intensidade de cor (de 0 a 1), a acurácia das previsões. Os principais erros de classificação ocorreram em notas extremamente graves ou agudas.

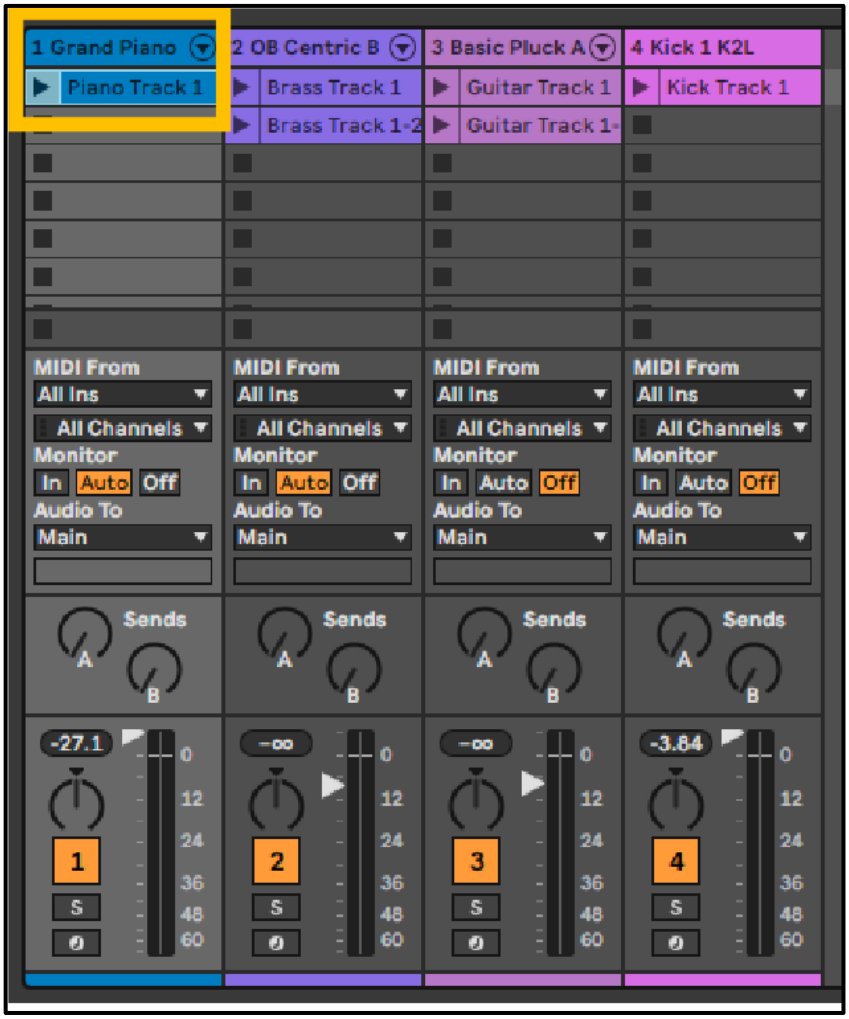
Figura 24 - Matriz de confusão normalizada



Fonte: Zhuang et al, 2025

O artigo segmentou as notas por faixas para aprofundar a análise da classificação. As notas entre E2 e B3 apresentaram maior concentração ao longo da diagonal da matriz, indicando precisão elevada. Já as notas de C-1 a G0 (registros graves) apresentaram 26.000 ocorrências no rótulo 0 - indicando alta acurácia -, ainda que com pequenos desvios em outros rótulos. A Figura 25 demonstra a aplicação do MIDI gerado em uma faixa única de piano de cauda, destacada em laranja, integrada a outras três faixas adicionadas manualmente: sintetizador Oberheim metálico (instrumento de sopro), violão e bumbo ou tambor.

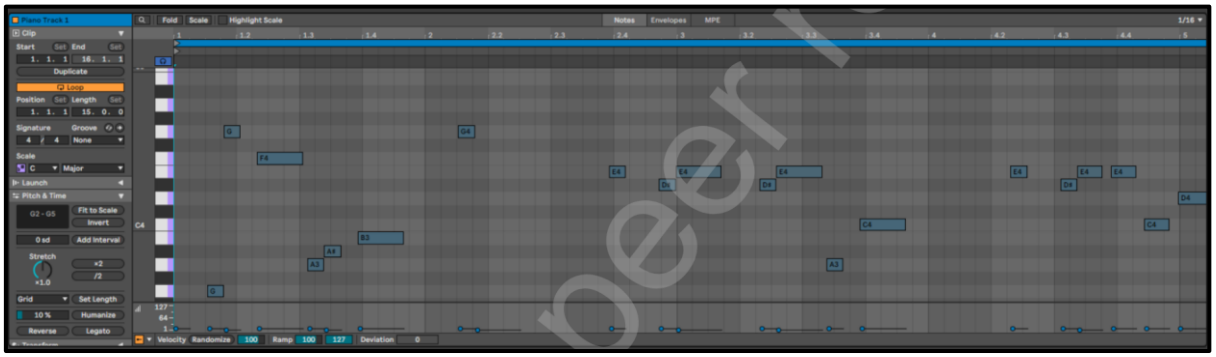
Figura 25 - Arranjo e integração da faixa MIDI



Fonte: Zhuang et al, 2025

A Figura 26 detalha as notas geradas em escala de Dó maior (C maior), com compasso 4/4 e variações de duração, altura e intensidade, compondo uma sequência coerente e bem estruturada.

Figura 26 - Visualização detalhada da faixa MIDI gerada



Fonte: Zhuang et al, 2025

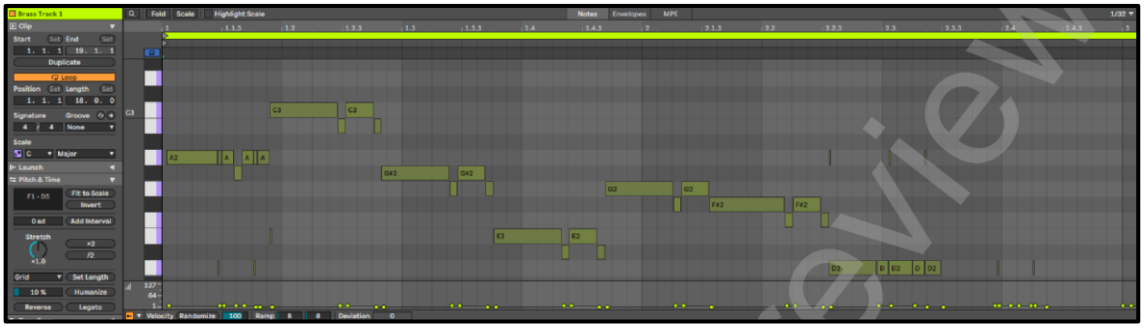
Para a geração de múltiplas faixas, conforme apresentado na Figura 27, foram elaboradas quatro trilhas: sintetizador metálico, piano eletrônico, conjunto de bateria eletrônica e bumbo eletrônico. Cada instrumento possui controle individual de volume e monitoramento, possibilitando edições futuras.

Figura 27 - Múltiplas faixas geradas pelo modelo

Fonte: Zhuang et al, 2025

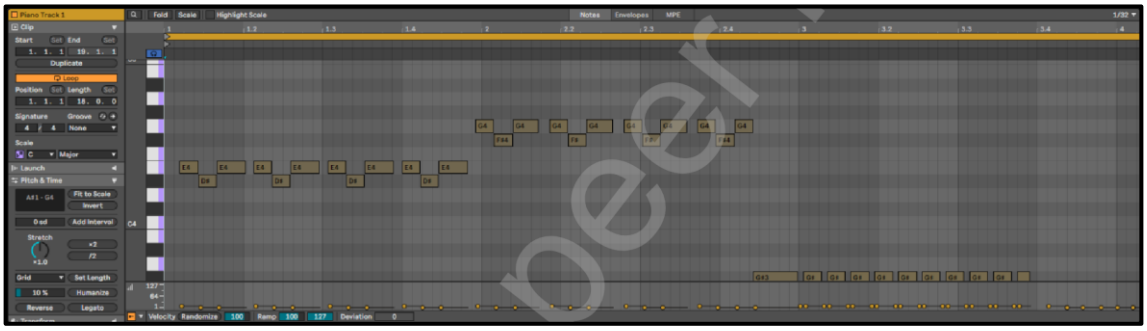
É possível visualizar as notas geradas em cada faixa nas Figuras 28, 29, 30 e 31, observando aspectos como altura, duração, velocidade e estrutura rítmica. As Figuras 28 e 31 evidenciam progressões melódicas; a Figura 29 apresenta repetições de motivos com variações melódicas; e a Figura 30 destaca variações rítmicas com aumento da largura das notas, gerando maior impacto percussivo. O ambiente em que essas notas são visualizadas é conhecido como estação de trabalho de áudio digital (*Digital Audio Workstation* – DAW).

Figura 28 - Notas MIDI do sintetizador de metal



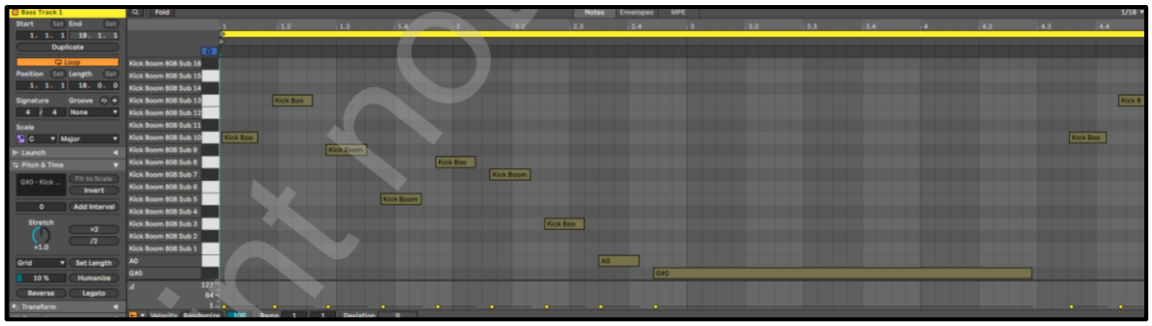
Fonte: Zhuang et al, 2025

Figura 29 - Notas MIDI do piano eletrônico



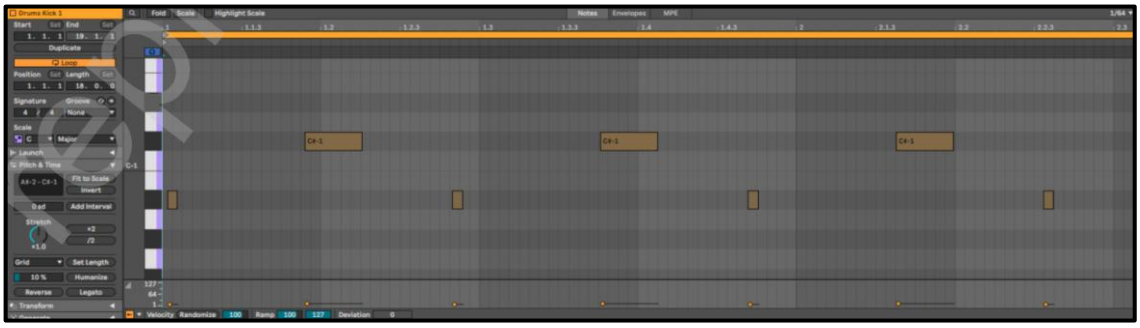
Fonte: Zhuang et al, 2025

Figura 30 - Notas MIDI da bateria



Fonte: Zhuang et al, 2025

Figura 31 - Notas MIDI do bumbo



Fonte: Zhuang et al, 2025

5.3.4 Discussões

Os autores compararam os resultados de classificação e geração do modelo com o estudo de Angioni et al., utilizando o mesmo conjunto de dados NES-MDB. O desempenho foi contrastado com modelos como redes neurais (NN), floresta aleatória (*Random Forest*), máquinas de vetores de suporte (SVM), LSTM, CNN e transformadores. O modelo proposto superou os demais nas métricas de classificação, incluindo precisão por tipo musical e médias aritmética e ponderada por gênero, como mostram as Tabelas 6, 7 e 8, evidenciando a efetividade da integração entre VAE e *Music Transformer*.

Tabela 6 - Comparação de desempenho dos modelos de classificação

Modelo	Pontuação de Acurácia	Pontuação ROC-AUC
NN	0,40	0,53
RF	0,42	0,54
SVM	0,26	0,50
LSTM	0,65	0,68
CNN	0,62	0,65
Transformadores	0,73	0,78
Modelo proposto	**0,92**	**0,91**

Fonte: Zhuang et al, 2025

Tabela 7 - Métricas de avaliação para classificação musical baseado em gênero

Modelo	RPG	Esporte	Combate	Tiro	Quebra-cabeça
NN	0,45	0,008	0,06	0,49	0,05
RF	0,42	0,004	0,002	0,34	0,01
SVM	0,35	0,08	0,16	0,29	0,12
LSTM	0,72	0,62	0,64	0,74	0,60
CNN	0,70	0,61	0,59	0,71	0,58
Transformadores	0,78	0,64	0,69	0,79	0,90
Modelo proposto	**0,93**	**0,89**	**0,90**	**0,92**	**0,95**

Fonte: Zhuang et al, 2025

Tabela 7 - Classificação de gêneros musicais: médias aritmética e ponderada

Modelo	Média Aritmética	Média Ponderada
NN	0,21	0,34
RF	0,15	0,27
SVM	0,20	0,27
LSTM	0,65	0,72
CNN	0,61	0,67
Transformadores	0,70	0,75
Modelo proposto	**0,91**	**0,94**

Fonte: Zhuang et al, 2025

As métricas quantitativas adotadas no desenvolvimento do artigo proporcionaram uma análise objetiva do sistema proposto, que demonstrou ser eficaz e robusto em tarefas de geração e classificação musical. Além disso, a verificação da estrutura e coerência das músicas por meio da DAW reforça sua aplicabilidade prática. No entanto, assim como o modelo EmotionBox, a ausência de avaliações subjetivas - como testes auditivos com ouvintes humanos - limitou a compreensão quanto aos aspectos expressivos, criativos e de recepção musical.

Ademais, a falta de considerações jurídicas e éticas sobre o uso do conjunto de dados NES-MDB levanta preocupações relacionadas a direitos autorais, uma vez que não houve menção a licenciamento ou autorização formal de uso. Mesmo que não haja intenção de comercialização da música gerada, ainda persistem riscos legais associados à proteção das composições originais e seu uso indevido.

Esse cuidado torna-se ainda mais relevante diante do histórico rigoroso da Nintendo em proteger sua propriedade intelectual. A empresa é reconhecida por adotar uma postura ativa contra o uso não autorizado de seus conteúdos, o que reforça a necessidade de que futuras pesquisas sejam conduzidas com maior atenção aos aspectos legais.

6. CONCLUSÃO

O presente estudo abordou os principais aspectos técnicos e conceituais relacionados à aplicação de algoritmos e arquiteturas de Inteligência Artificial (IA) na produção musical automatizada. Foram analisados os fundamentos musicais essenciais, os tipos de algoritmos empregados, as redes neurais, os desafios inerentes e estudos de caso que aprofundaram a compreensão da interação entre modelos computacionais e o processo criativo. O foco principal foi estabelecer a conexão entre música e tecnologia, examinando o funcionamento de sistemas capazes de gerar composições musicais.

A relevância deste tema se acentua no contexto das transformações digitais que impactam crescentemente o setor musical, gerando a necessidade de otimizar processos e explorar novas formas de expressão artística. A análise de três estudos de caso permitiu identificar como cada um influenciou a criação sonora, contribuindo para alcançar os objetivos propostos: investigar algoritmos, compreender conceitos musicais e avaliar sistemas.

As hipóteses foram confirmadas ao observar que os modelos transformadores proporcionaram maior complexidade harmônica e melódica. Em sistemas gerador-crítico, algoritmos genéticos demonstraram eficácia em tarefas mais simples e de criatividade localizada. Contudo, o conhecimento musical prévio revelou-se indispensável para aprimorar os resultados, especialmente na configuração de parâmetros musicais.

Os sistemas examinados foram capazes de gerar músicas com características estruturais e sonoras semelhantes às de composições humanas. No entanto, foram observados exemplos de elevada insatisfação em casos específicos do gerador-crítico, como na música eletrônica (*dance*). Apesar da possibilidade de resultados negativos, a IA possui o potencial de atuar como uma ferramenta auxiliar, complementando a ação humana em vez de substituí-la.

Desafios relacionados à criatividade, regulamentação e considerações emocionais no fluxo criativo indicam a necessidade de mais pesquisas e estudos para a superação desses obstáculos. Trabalhos futuros podem explorar essas questões e desenvolver sistemas mais transparentes.

REFERÊNCIAS

AHAMED, K. I; AKTHAR, S. A Study on Neural Network Architectures. International Knowledge Sharing Platform, 2016. Disponível em: <https://iiste.org/Journals/index.php/CEIS/article/view/32857/33754>. Acesso em: 9 de jun. 2025.

AHO, A. V; HOPCROFT, J. E; ULLMAN, J. D. Data Structures and Algorithms. Pearson, 1983. Disponível em: <https://users.dcc.uchile.cl/~voyanede/cc4102/dS&A%20Book%20By%20Alfred%20Aho.pdf>. Acesso em: 10 de jun. 2025.

ÅKERBLOM, B; CASTEGREN, E. Arrays in Practice An Empirical Study of Array Access Patterns on the JVM. arXiv preprint, 2024. Disponível em: <https://arxiv.org/pdf/2403.02416>. Acesso em: 10 de jun. 2025.

ALRASHED, T. et al. Dataset or Not? A Study on the Veracity of Semantic Markup for Dataset Pages. Springer, 2021. Disponível em: <https://link.springer.com/chapter/10.1007/978-3-030-88361-4_20>. Acesso em: 10 de jun. 2025.

ALSOWAIL, R. A; AL-SHEHARI, T. An Insider Data Leakage Detection Using One-Hot Encoding, Synthetic Minority Oversampling and Machine Learning Techniques. Entropy, 2021. Disponível em: <https://pmc.ncbi.nlm.nih.gov/articles/PMC8535057/>. Acesso em: 8 de jun. 2025.

AN, J; CHO, S. Variational Autoencoder based Anomaly Detection using Reconstruction Probability. SNU Data Mining Center, 2015. Disponível em: <http://dm.snu.ac.kr/static/docs/TR/SNUDM-TR-2015-03.pdf>. Acesso em: 9 de jun. 2025.

ANGIONI, SIMONE, et al. A transformers-based approach for fine and coarse-grained classification and generation of MIDI songs and soundtracks. PeerJ Computer Science, vol.9, pp. e1410, 2023, doi: 10.7717/peerj-cs.1410. Disponível em: <https://peerj.com/articles/cs-1410/>. Acesso em: 10 de jun. 2025.

ASAN, U; KALAYCI, T. A. Improving Classification Performance of Fully Connected Layers by Fuzzy Clustering in Transformed Feature Space. Istanbul Technical University, 2022. Disponível em: <https://www.mdpi.com/2073-8994/14/4/658>. Acesso em: 9 de jun 2025.

AWS AMAZON. O que são transformadores em inteligência artificial?. Amazon, [S.D.]. Disponível em: <https://aws.amazon.com/pt/what-is/transformers-in-artificial intelligence/>. Acesso em: 9 de jun. 2025.

BIAN, C; ZHOU, Y; QIAN, C. Robust Subset Selection by Greedy and Evolutionary Pareto Optimization. Nanjing University, 2022. Disponível em: <https://arxiv.org/pdf/2205.01415>. Acesso em: 8 de jun. 2025.

BISWAS, A. et al. Advances in Speech and Music Technology. Disponível em: <https://link.springer.com/book/10.1007/978-3-031-18444-4>. Acesso em: 01 de dez. 2024.

BLEI, D. M; KUCUKELBIR, A; MCAULIFFE, J. D. Variational Inference: A Review for Statisticians. arXiv preprint, 2018. Disponível em: <https://arxiv.org/pdf/1601.00670>. Acesso em 8 de jun. 2025.

BRASS, P. Advanced Data Structures. City College of New York, 2008. Disponível em: <https://helloplanetcpp.wordpress.com/wp-content/uploads/2018/07/advanced-data-structures.pdf>. Acesso em: 10 de jun. 2025.

BRIOT, J. P. et al. Deep Learning Techniques for Music Generation - A Survey. Paris. Disponível em: <https://arxiv.org/pdf/1709.01620>. Acesso em: 01 de dez. 2024.

CARDEW. C. Notation-Interpretation, Etc. Tempo, 1961. Disponível em: <https://www.cambridge.org/core/journals/tempo/article/abs/notationinterpretation etc/3A48A734758BD7E449B3C9574855676E>. Acesso em: 3 de jun. 2025.

CHEN, L; VAROQUAUX, G; SUCHANEK, F. M. The Locality and Symmetry of Positional Encodings. Institut Polytechnique de Paris, 2023. Disponível em: <https://arxiv.org/pdf/2310.12864>. Acesso em: 9 de jun. 2025.

CHO, K. et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. Association for Computational Linguistics, 2014. Disponível em: <https://aclanthology.org/D14-1179.pdf>. Acesso em: 9 de jun. 2025.

CIVIT, M. et al. A systematic review of artificial intelligence-based music generation: Scope, applications, and future trends. Seville. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417422013537#b16>. Acesso em: 01 de dez. 2024.

DASH, A; AGRES, K. AI-Based Affective Music Generation Systems: A Review of Methods and Challenges. ACM, 2024. Disponível em: <https://dl.acm.org/doi/pdf/10.1145/3672554>. Acesso em: 10 de jun. 2025

EBU TECH. Assessment Methods for the Quality of Sound Material - Music. Tech 3000 Series, 1997. Disponível em: <https://tech.ebu.ch/publications/tech3286>. Acesso em: 8 de mai. 2025.

FABBRI, F; PINHO, M. G. Uma teoria dos gêneros musicais: duas aplicações. Revista Vórtex, 2017. Disponível em: <periodicos.unespar.edu.br/vortex/article/view/2161>. Acesso em: 7 de jun 2025.

FRIGATTI, E. F. L. Polifonia e Contraponto: análise do uso de procedimentos contrapontísticos durante o século XX para criação de novas obras musicais. USP, 2020. Disponível em: <https://www.teses.usp.br/teses/disponiveis/27/27158/tde-26022021-223142/pt-br.php>. Acesso em: 7 de jun. 2025.

GOOGLE DEVELOPERS. Glossário de machine learning. Machine Learning, 2025. Disponível em: <https://developers.google.com/machine-learning/glossary?hl=pt-br>. Acesso em: 7 de jun. 2025.

HOROWITZ, E; SAHNI, S. Fundamentals of Data Structures. Computer Science Press, 1983. Disponível em: <https://ggnindia.dronacharya.info/Downloads/Subinfo/RelatedBook/Data-Structure-Algorithms-Text-Book-1.pdf>. Acesso em: 10 de jun. 2025.

HOSKEN, D. An Introduction to Music Technology. Nova Iorque, 2014. Disponível em: <https://www.taylorfrancis.com/books/mono/10.4324/9780203539149/introduction-music-technology-dan-hosken>. Acesso em: 5 de jun. 2025.

HU, Y. et al. Overcoming the vanishing gradient problem in plain recurrent networks. University of Zurich, 2018. Disponível em: <https://arxiv.org/abs/1801.06105>. Acesso em: 9 de jun. 2025.

HUANG, C. Z. A. et al. MUSIC TRANSFORMER: GENERATING MUSIC WITH LONG-TERM STRUCTURE. Disponível em: <https://arxiv.org/pdf/1809.04281>. Acesso em: 01 de dez. 2024.

HUAWEI TECHNOLOGIES. Machine Learning. Springer, 2023. Disponível em: <https://link.springer.com/chapter/10.1007/978-981-19-2879-6_2#citeas>. Acesso em: 7 de jun. 2025.

HURON. D. A Psychological Approach to Musical Form: The Habituation-Fluency Theory of Repetition. Current Musicology, 2013. Disponível em: <https://journals.library.columbia.edu/index.php/currentmusicology/article/view/5312>. Acesso em: 7 de jun. 2025.

IBM. What is gradient descent?. IBM, [S.D.]. Disponível em: <https://www.ibm.com/think/topics/gradient-descent>. Acesso em: 9 de jun. 2025.

IBM. What is underfitting?. IBM, 2021. Disponível em: <https://www.ibm.com/think/topics/underfitting>. Acesso em: 8 de jun. 2025.

International Federation of the Phonographic Industry. STATE OF THE INDUSTRY. Disponível em: <https://globalmusicreport.ifpi.org/>. Acesso em: 01 de dez. 2024.

ISLAM, M; CHEN, G; JIN, S. An Overview of Neural Network. ResearchGate, 2019. Disponível em: <https://www.researchgate.net/publication/337137421_An_Overview_of_Neural_Network>. Acesso em: 9 de jun. 2025.

JAIN, A; KUMAR, U. Research Paper on Queues. Dronacharya College of Engineering India, 2014. Disponível em: <https://ijirt.org/publishedpaper/IJIRT100828_PAPER.pdf>. Acesso em: 9 de jun. 2025.

KINGMA, D. P; BA, J. L. ADAM: A Method for Stochastic Optimization. arXiv preprint, 2015. Disponível em: <https://arxiv.org/pdf/1412.6980>. Acesso em: 8 de jun. 2025.

KOSTKA, S; Payne, D; Almén, B. Tonal Harmony: With an Introduction to Twentieth Century Music. McGraw-Hill, 2012. Disponível em: <https://www.docdroid.net/j83Dfj8/kostka-stefan-payne-dorothy-almen-byron-tonal-harmony-with-an-introduction-to-twentieth-century-music-mcgraw-hill-humanities-social-sciences-languages-2012-pdf>. Acesso em: 7 de jun. 2025.

KWIECIEŃ, J. et al. Technical, Musical, and Legal Aspects of an AI-Aided Algorithmic Music Production System. Cracóvia. Disponível em: <https://www.mdpi.com/2076-3417/14/9/3541#B18-applsci-14-03541> Acesso em: 30 nov. 2024.

LATHAM, A. The Oxford Companion to Music. Oxford University Press, 2011. Disponível em: <https://www.oxfordreference.com/display/10.1093/acref/9780199579037.001.0001/cref-9780199579037-e-6912>. Acesso em: 7 de jun. 2025.

LIPTON, Z. C; BERKOWITZ, J. A Critical Review of Recurrent Neural Networks for Sequence Learning. arXiv preprint, 2015. Disponível em: <https://arxiv.org/pdf/1506.00019>. Acesso em: 9 de jun. 2025.

LIU, H; GEGOV, A; COCEA, M. Rule-based systems: a granular computing perspective. Springer, 2016. Disponível em: <https://link.springer.com/article/10.1007/s41066-016-0021-6>. Acesso em 9 de jun. 2025.

MARTIN, J. L. Semiotic resources of music notation: Towards a multimodal analysis of musical notation in student texts. 2014. Disponível em: <https://www.degruyterbrill.com/document/doi/10.1515/sem-2014-0006/html>. Acesso em: 3 jun. 2025.

MAUCH, M. et al. The Evolution of Popular Music: USA 1960-2010. Queen Mary University of London, 2015. Disponível em: <https://arxiv.org/pdf/1502.05417>. Acesso em: 7 de jun. 2025.

MITRA, R; ZUALKERNAN, I. Music Generation Using Deep Learning and Generative AI: A Systematic Review. American University of Sharjah, 2025. Disponível em: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10845168>. Acesso em: 7 de jun. 2025.

MÖLLENKAMP, A. Paradigms of Music Software Development. Universidade de Rostock Alemanha, 2014. Disponível em: <https://www.academia.edu/16727745/Paradigms_of_Music_Software_Development>. Acesso em: 6 de jun. 2025.

MORAIS, G. F. Composição Algorítmica de Música - Desenvolvimento de Software (HarmoniMIDI). Universidade Federal de Minas Gerais. Disponível em: <https://monografias.dcc.ufmg.br/wp-content/uploads/GabrielFerreiraMorais.pdf>. Acesso em: 01 de dez. 2024.

MYCKA, J; MAŃDZIUK, J. Artificial intelligence in music: recent trends and challenges. Springer, 2024. Disponível em: <https://link.springer.com/article/10.1007/s00521-024 10555-x>. Acesso em: 10 de jun. 2025.

NADARAJAH, S; NAWA, V. Exact Expressions for Kullback-Leibler Divergence for Univariate Distributions. Entropy, 2024. Disponível em: <https://www.mdpi.com/1099-4300/26/11/959>. Acesso em: 8 de jun. 2025.

NGUYEN, Q. H. et al. A Novel Hybrid Model Based on a Feedforward Neural Network and One Step Secant Algorithm for Prediction of Load-Bearing Capacity of Rectangular Concrete-Filled Steel Tube Columns. Molecules, 2020. Disponível em: <https://www.mdpi.com/1420-3049/25/15/3486>. Acesso em: 9 de jun. 2025.

NIKOLSKY, A. Glossary of some important musical terms. ResearchGate, 2022. Disponível em: <https://www.researchgate.net/publication/363696216_Glossary_of_some_importan _musical_terms>. Acesso em: 7 de jun. 2025.

PADHYA, J; YADAV, A. Data Structures. Mumbai, 2023. Disponível em: <https://www.ijarsct.co.in/Paper16219.pdf>. Acesso em: 9 de jun. 2025.

PERCINO, G; KLIMEK, P; THURNER, S. Instrumentational complexity of music genres and why simplicity sells. University of Vienna, 2014. Disponível em: <https://arxiv.org/pdf/1405.5057>. Acesso em: 7 de junho de 2025

POLAND, C. M. Generative AI and US Intellectual Property Law. Texas USA, 2023. Disponível em: <https://arxiv.org/pdf/2311.16023>. Acesso em 7 de junho de 2025.

RAFFEL, C; ELLIS, D. P. Intuitive analysis, creation and manipulation of midi data with pretty midi. In 15th international society for music information retrieval conference late breaking and demo papers (pp. 84-93) 2014. Disponível em: <https://colinraffel.com/publications/ismir2014intuitive.pdf>. Acesso: 10 de jun. 2025.

RAZANI, R. et al. A Reduced Complexity MFCC-based Deep Neural Network Approach for Speech Enhancement. University St Montreal, 2017. Disponível em: <https://www.ece.mcgill.ca/~bchamp/Papers/Conference/ISSPIT2017.pdf>. Acesso em: 9 de jun. 2025.

REHMER, A; KROLL, A. On the vanishing and exploding gradient problem in Gated Recurrent Units. Elsevier, 2020. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2405896320317481>. Acesso em 9 de jun. 2025.

REUNANEN, M. Trackers: The Rise, Bloom and Later Developments of a Paradigm. WiderScreen, 2024. Disponível em: <http://widerscreen.fi/numerot/ajankohtaista/trackers-the-rise-bloom-and-later-developments-of-a-paradigm/>. Acesso em: 6 de jun. 2025.

RÉVEILLAC, J. M. Electronic Music Machines: The New Musical Instruments. ISTE, 2019. Disponível em: <https://onlinelibrary.wiley.com/doi/book/10.1002/9781119618089>. Acesso em: 6 de jun. 2025.

RIMAS, J; Jr, J. R. The Difference Between Articulation and Phrasing. Palgrave Macmillan, 2024. Disponível em: <https://link.springer.com/chapter/10.1007/978-3-031-63965-4_12>. Acesso em: 7 de jun. 2025.

ROGER, S. The Aesthetics of Music. Oxford Nova Iorque, 1999. Disponível em: <https://drmarcjeanbernard.weebly.com/uploads/3/7/5/0/37501827/147163534-roger-scruton-the-aesthetics-of-music-html-edition-1999.pdf>. Acesso em: 7 de jun. 2025.

RUDER, S. An overview of gradient descent optimization algorithms. Dublin, 2017. Disponível em: <https://arxiv.org/pdf/1609.04747>. Acesso em: 7 de jun. 2025.

SIPHOCLY, N; SALEM, A. B. M. Top 10 Artificial Intelligence Algorithms in Computer Music Composition. ResearchGate, 2021. Disponível em: <https://www.researchgate.net/publication/355913039_Top_10_artificial_intelligence algorithms_in_computer_music_composition>. Acesso em: 10 de jun. 2025.

SJÖLUND, J. A Tutorial on Parametric Variational Inference. arXiv preprint, 2023. Disponível em: <https://arxiv.org/pdf/2301.01236>. Acesso em: 8 de jun. 2025.

SON Il, K; GYONG Il, H; MIN, P. J. An Analysis of General Fuzzy Logic and Fuzzy Reasoning Method. arXiv preprint, 2016. Disponível em: <https://arxiv.org/abs/1604.03210>. Acesso em: 8 de jun. 2025.

SOUZA FILHO, N. E; GONÇALVES, B. A; OLIVEIRA. V. T. Música para estudantes de engenharia: Síntese sonora de tema de jazz. Revista Brasileira de Ensino de Física, 2015. Disponível em: <https://www.scielo.br/j/rbef/a/t3qnqKNLmXQmgFjWzNHn7wK>. Acesso em: 7 de jun. 2025.

SPENCER. P; TEMKO, P. M. A Practical Approach to the Study of Form in Music. Prentice-Hall, 1988. Disponível em: <https://www.google.com.br/books/edition/A_Practical_Approach_to_the_Study_of_ or/IXUfAAAAQBAJ?hl=en&gbpv=0>. Acesso em: 7 de jun. 2025.

SPINDELBÖCK, T; RANFTL, S; VON DE LINDEN, W. Cross-Entropy Learning for Aortic Pathology Classification of Artificial Multi-Sensor Impedance Cardiography Signals. National Library of Medicine, 2021. Disponível em: <https://pmc.ncbi.nlm.nih.gov/articles/PMC8700029/>. Acesso em: 8 de jun. 2025.

STRAYER, H. R. From Neumes to Notes: The Evolution of Music Notation. Cedarville University, 2013. Disponível em: <https://digitalcommons.cedarville.edu/musicalofferings/vol4/iss1/1/>. Acesso em: 3 de jun. 2013.

STRYKER, C; BELCIC, I. What is learning rate in machine learning?. IBM, 2024. Disponível em: <https://www.ibm.com/think/topics/learning-rate>. Acesso em: 7 de jun. 2025.

STRYKER, C; BERGMANN, D. What is backpropagation?. IBM, 2024. Disponível em: <https://www.ibm.com/think/topics/backpropagation>. Acesso em: 8 de jun. 2025.

SUNIL RAI, A. A Comparative Study of Stack and Queues in Data Structure. Dronacharya College of Engineering Gurgaon, 2014. Disponível em: <https://ijirt.org/publishedpaper/IJIRT101022_PAPER.pdf>. Acesso em: 9 de jun. 2025.

TAFVIZI, A; AVCI, B; SUNDARARAJAN, M. Attributing AUC-ROC to Analyze Binary Classifier Performance. arXiv preprint, 2022. Disponível em: <https://arxiv.org/pdf/2205.11781>. Acesso em: 9 de jun. 2025.

TAVARES, J. N. Progressão harmónica. Universidade do Porto, 2014. Disponível em: <https://rce.casadasciencias.org/rceapp/art/2014/266/>. Acesso em: 7 de jun. 2025.

THEDE, S. An introduction to genetic algorithms. ResearchGate, 2004. Disponível em: <https://www.researchgate.net/publication/228609251_An_introduction_to_genetic_algorithms>. Acesso em: 9 de jun. 2025.

TJORA, A. H. The groove in the box: a technologically mediated inspiration in electronic dance music. Cambridge University Press, 2009. Disponível em: <https://www.cambridge.org/core/journals/popular-music/article/abs/groove-in-the-box-a-technologically-mediated-inspiration-in-electronic-dance-music/7BFC620177B2DC76D85D2133A0BF61D9>. Acesso em: 6 de jun. 2025.

TRIPATHY, B. K; GANTAYAT, S. S. On The Implementation Of Data Structures Through Theory Of Lists. International Journal of Information Technology Convergence and Services, 2012. Disponível em: <https://airccse.org/journal/ijitcs/papers/2512ijitcs05.pdf>. Acesso em: 9 de jun. 2025.

TROTTA, F. Gêneros Musicais e Sonoridade: construindo uma ferramenta de análise. Universidade Federal de Pernambuco, 2008. Disponível em: <periodicos.ufpe.br/revistas/icone/article/view/230128>. Acesso em: 7 de jun. 2025.

VINES, B. W; NUZZO, R. L; LEVITIN, D. Quantifying and analyzing musical dynamics: Differential calculus, physics and functional data techniques. ResearchGate, 2005. Disponível em: <https://www.researchgate.net/publication/233421178_Quantifying_and_analyzing_musical_dynamics_Differential_calculus_physics_and_functional_data_techniques>. Acesso em: 7 de jun. 2025.

XUE, C; ZHANG, T; XIAO, D. An Advanced Broyden-Fletcher-Goldfarb-Shanno Algorithm for Prediction and Output-Related Fault Monitoring in Case of Outliers. Journal of Chemistry, 2022. Disponível em: <https://onlinelibrary.wiley.com/doi/10.1155/2022/7093835>. Acesso em: 9 de jun. 2025.

YOUVAN, D. C. Stochastic Elements in Language Models: Exploring Variability and Randomness in AI Responses. ResearchGate, 2023. Disponível em: <https://www.researchgate.net/publication/375922473_Stochastic_Elements_in_Language_Models_Exploring_Variability_and_Randomness_in_AI_Responses>. Acesso em: 8 de jun. 2025.

ZAVADSKA, G; DAVIDOVA, J. Composition of Song Accompaniment as a Form of Developing Future Music Teachers' Harmonic Hearing. Editora europeia, 2019. Disponível em: <https://www.europeanpublisher.com/en/article/10.15405/ejsbs.250>. Acesso em: 7 de jun. 2025.

ZHAO, K. et al. An emotional symbolic music generation system based on LSTM networks. In Proceedings of 2019 ieee 3rd information technology, networking, electronic and automation control conference, itnec 2019. Disponível em: <https://www.researchgate.net/publication/345425604_An_Emotional_Symbolic_Music_Generation_System_based_on_LSTM_Networks>. Acesso em: 10 de jun. 2025.

ZHENG, K. et al. EmotionBox: a music-element-driven emotional music generation system using Recurrent Neural Network. China, 2021. Disponível em: <https://arxiv.org/pdf/2112.08561>. Acesso em: 19 de mai. 2025.

ZUANG, Q. et al. Music Generation and Classification of 8-Bit Tracks Using Variational Autoencoder and Music Transformer. SSRN, 2025. Disponível em: <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5195892>. Acesso em: 21 de mai. 2025.