Explorando Modelos com Janelas Temporais para Previsão de Acordes Musicais a Partir de Melodias

Marlon Duarte
Universidade Federal do Ceará (UFC)
Fortaleza, Brazil
marlongduarte@alu.ufc.br

Raylander Marques
Universidade Federal do Ceará (UFC)
Fortaleza, Brazil
raylandermarques@alu.ufc.br

Gabriel Rudan
Universidade Federal do Ceará (UFC)
Fortaleza, Brazil
gabrielrudan@alu.ufc.br

Zairo Bastos
Universidade Federal do Ceará (UFC)
Fortaleza, Brazil
zairo.vianahd@alu.ufc.br

César Lincoln
Universidade Federal do Ceará (UFC)
Fortaleza, Brazil
cesarlincoln@dc.ufc.br

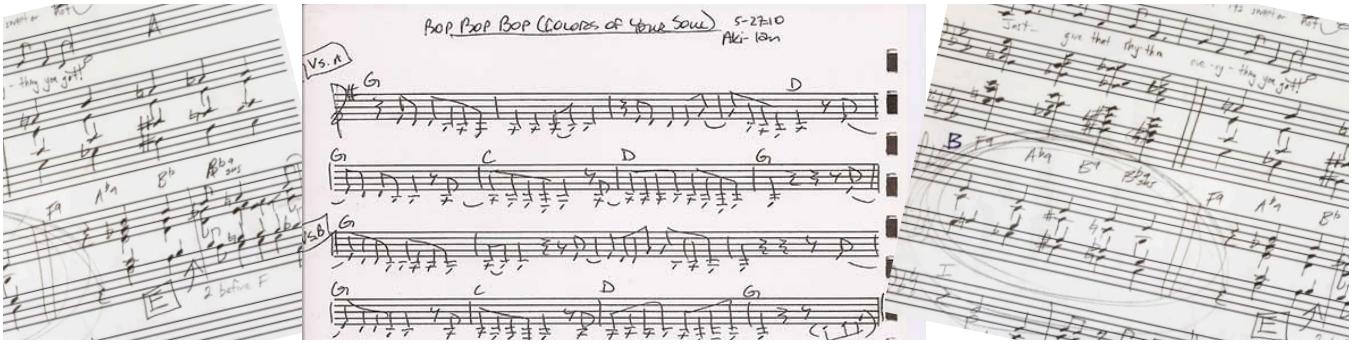

Figure 1: Exemplos de compassos musicais escritos à mão.

ABSTRACT

The task of predicting chords in music is highly important in the careers of composers and professional musicians. Composers create new melodies that require harmony to make them, among many other things, more marketable. Professional arranger musicians frequently need to learn new songs for which they lack supporting material. This work explores chord prediction in melodies using Machine Learning. Utilizing the POP909 dataset [1], composed of 909 songs in MIDI format, Random Forest (RF) and Long Short-Term Memory (LSTM) models were trained and compared, both with and without bidirectionality, employing features such as melody, note intensity, musical key, and the delta time of each melodic interval's execution. The RF model achieved an overall average accuracy of approximately 77%, performing well on common chords. Conversely, the LSTM without bidirectionality achieved around 61% accuracy, and with the use of the technique (BLSTM), it obtained approximately 73% accuracy. Both models demonstrated challenges in predicting rare chords, such as diminished chords. The contributions aim to advance AI-assisted musical analysis, focusing on applications for composition and live accompaniment.

[1]https://github.com/music-x-lab/POP909-Dataset

In: Proceedings of the Brazilian Symposium on Multimedia and the Web (WebMedia'2025). Rio de Janeiro, Brazil. Porto Alegre: Brazilian Computer Society, 2025.

ISSN 2966-2753

KEYWORDS

Chord Prediction, Musical Sequences, Machine learning, Time Series

1 INTRODUÇÃO

A música exerce uma profunda influência no comportamento e no bem-estar humano, desempenhando papéis fundamentais no desenvolvimento cognitivo, emocional e motor [3]. Segundo Benenzon [3], atividades musicais podem fortalecer vínculos sociais, melhorar a concentração e estimular a coordenação motora, como é evidente na prática de instrumentos que exigem sincronização corporal, seja na bateria ou na execução coletiva de peças musicais que demandam precisão rítmica e sensibilidade interpessoal.

A tarefa de músicos profissionais, como arranjadores e compositores, necessita de uma experiência, muitas vezes baseada na mnemônica, consideravelmente complexa. Isso se dá pelo fato de necessitar inserir harmonia em uma melodia nunca antes ouvida ou estudada, no caso dos compositores. No caso de músicos arranjadores profissionais, estes precisam ser capazes de ouvir uma música pela primeira vez e construir um arranjo para a mesma. Os músicos que atuam nessa linha precisam saber "tirar música de ouvido", ou seja, prever os acordes que melhor acompanham uma determinada melodia [11].

Com o avanço das tecnologias digitais, a interseção entre música e computação tornou-se cada vez mais forte. Ferramentas como *Digital Audio Workstations* (DAWs), protocolos *Musical Instrument*

Digital Interface (MIDI) e ambientes de produção digital transformaram a forma como a música é composta, executada e analisada [2].

Em meados do século XX, quando as primeiras gravações musicais estavam sendo produzidas, o perfil de cantores, por exemplo, era muito claro devido à forma como as gravações eram feitas [8]. Não havia sistema de gravação separado em trilhas; um único captador tinha a tarefa de capturar todas as fontes de áudio, violões ou pianos, além do cantor, que necessitava ter uma voz muito forte para se destacar perto de instrumentos com sonoridade tão intensa. Nesse contexto, surgem também novas possibilidades de pesquisa envolvendo técnicas de aprendizado de máquina aplicadas à música [5].

O aprendizado de máquina (ML), conceito que evoluiu da inteligência artificial (IA) consolidada na década de 50, permite que os sistemas aprendam e melhorem automaticamente a partir da experiência, sem programação explícita, como observado por [18]. Recentemente, o ML tem sido aplicado a problemas musicais como "*thumb-nailing*" [14], geração de música [23] e transferência de estilo (LU, 2019), com pesquisadores frequentemente disponibilizando as saídas de áudio em seus sites. Apesar do avanço na aplicação da IA no processo criativo, especialistas questionam os limites de sua atuação nesse campo [19].

Este trabalho propõe uma abordagem de treinamento de dois modelos de *Machine Learning* (ML) para a predição de acordes em melodias a partir de arquivos MIDI[2]. Em termos musicais, a melodia pode ser compreendida como a sucessão organizada de notas no tempo, cuja linearidade define a identidade principal de uma peça. Já a harmonia, refere-se à combinação simultânea de sons, isto é, à sobreposição de notas que formam acordes e sustentam o discurso melódico. Ambos os parâmetros, embora distintos, mantêm uma relação de interdependência: a harmonia frequentemente emerge de agrupamentos de notas melódicas, enquanto a melodia tende a se apoiar nas estruturas harmônicas subjacentes. O ritmo, por sua vez, organiza a disposição temporal desses elementos, influenciando a percepção de estrutura e movimento. Nesse contexto, a tarefa de prever acordes a partir de melodias consiste em modelar computacionalmente as relações entre padrões melódicos e contextos harmônicos; um desafio que requer a representação adequada de informações de intensidade das notas, tonalidade e janelas de sequência melódicas que resultam em um determinado acorde.

As principais contribuições deste estudo incluem: a análise do modelo *Random Forest* na predição de acordes a partir de uma melodia com base em janelas de tempo e estrutura tabular; uma comparação dos resultados de um modelo LSTM com um modelo RF realizando a mesma tarefa de predição de acordes dada uma melodia; outra comparação entre as predições dos dois modelos em relação à predição humana; e a discussão sobre possíveis aplicações em sistemas inteligentes de apoio à composição, improvisação ou acompanhamento musical.

2 TRABALHOS RELACIONADOS

A previsão de características musicais por meio de modelos de redes neurais atualmente vem se estabelecendo como um expansivo campo de pesquisa [27]. Pesquisas como a de Araujo et al. [1] evidenciam, inclusive, que essa abordagem pode extrapolar o âmbito puramente musical, permitindo, por exemplo, prever o sucesso de músicas no mercado, utilizando como base correlações entre menções no Twitter, métricas de popularidade no Spotify, gênero musical e a ocorrência de artistas ou movimentos de destaque.

No entanto, apesar de sua popularidade, a aplicação desses modelos para tarefas como a previsão de acordes em sequências musicais ainda apresenta espaço para novas descobertas e aprimoramentos [4]. A identificação de variáveis relevantes e a otimização da precisão do aprendizado são, portanto, desafios contínuos que impulsionam a investigação nesta área.

Pensar na previsão de acordes sob a ótica de um problema de série temporal é um caminho promissor para o treinamento de modelos, dado que a ocorrência de um acorde é intrinsecamente dependente dos dados musicais que o precedem [10], como notas e até mesmo acordes anteriores, estabelecendo uma sequência temporal clara. Nesse contexto, utilizar as Redes de Memória de Longo Curto Prazo (LSTMs) se mostra altamente vantajoso, visto sua eficácia na previsão de dados sequenciais e na captura de dependências de longo prazo, características essenciais para a compreensão de estruturas musicais. A aplicabilidade de modelos baseados em LSTMs para tarefas ligadas à teoria musical é evidenciada por trabalhos como os de Hardwick e Lim et al..

O trabalho de Hardwick [12] investiga a habilidade do modelo LSTM em aprender sequências e dependências de longo prazo, com o objetivo de gerar progressões de acordes coerentes. As camadas LSTM são empregadas para assimilar as dependências de longo prazo presentes em um conjunto de dados MIDI, resultando na criação de progressões harmônicas fluidas e consistentes. Uma distinção importante de sua arquitetura é a iteração do modelo unicamente na direção para frente, ao contrário de algumas arquiteturas como as BLSTM (LSTMs Bidirecionais), que processam a melodia em ambas as direções, o que pode torná-las menos adequadas para aplicações em tempo real. A avaliação do sistema demonstrou a geração de harmonia majoritariamente diatônica e algumas relações cadenciais básicas. Porém, o estudo de caso efetuado para a validação evidencia uma forte preferência pelos acordes originais (5.93 vs. 3.6 de adequação). Embora o trabalho tenha evidenciado o potencial das representações de histograma de croma para a tarefa, a coerência musical de longo prazo e a preferência dos usuários ainda são desafios notáveis.

Corroborando essa linha de pesquisa, Lim et al. [15] explorou o uso de redes BLSTM para a geração automática de progressões de acordes a partir de melodias simbólicas. Neste caso, o modelo BLSTM demonstrou um aumento de desempenho de 23,8% sobre HMMs (Modelos Ocultos de Markov) e 11,4% sobre DNN-HMMs (Redes Neurais Profundas) na previsão de acordes. Apesar de o artigo apontar certas limitações, como o número restrito de classes de acordes e a não preservação integral do sentido sequencial da melodia em todos os contextos, a pesquisa é seminal ao demonstrar a robustez desses modelos recorrentes na captura de dependências temporais estendidas em dados musicais.

Além disso, outras metodologias de *machine learning* também têm sido empregadas na análise e classificação de características musicais. Monnier et al. [16] propôs uma metodologia para processar e reconhecer acordes musicais a partir de arquivos de áudio,

[2]Os arquivos MIDI armazenam dados de performance musical, como notas, tempo e intensidade, e não o áudio em si. [22]

utilizando a Transformada Rápida de Fourier (FFT) e o algoritmo *Treebagger*. Inicialmente, a precisão do reconhecimento era baixa (20-30%), mas, após otimizações na FFT, a precisão final ficou entre 67% e 89% (dependendo da semente do dispositivo). Esta pesquisa destacou a viabilidade de usar o aprendizado de máquina para reconhecer notas musicais, efetuando o devido pré-processamento, além de apontar a necessidade de técnicas mais sofisticadas para lidar com a complexidade de oitavas e inversões em acordes, a destacar o *Random Forest*.

Modelos como RF são reconhecidos por sua capacidade de lidar com alta dimensionalidade e capturar relações não lineares, sendo aplicáveis a problemas de classificação em diversos domínios, incluindo o musical. Pesquisas como [7], [20] e [24] ilustram a diversidade de técnicas e objetivos na análise computacional da música, abrangendo desde a geração de elementos rítmicos e harmônicos até a classificação de gêneros musicais e características específicas de instrumentos.

Mais recentemente, arquiteturas baseadas em *Transformers*, como o *Music Transformer*[13] e o *MuseNet*, têm se destacado por sua capacidade de modelar dependências musicais de longo alcance, aprimorando tarefas de geração e previsão harmônica. Tais estudos reforçam a relevância da exploração de diferentes modelos de aprendizado em conjunto com representações de dados apropriadas, para avançar no estudo e na previsão de estruturas musicais.

3 METODOLOGIA

Nesta seção serão explanados desde a fonte do *dataset* até os hiperparâmetros dos modelos utilizados.

3.1 Conjunto de Dados

O *dataset* POP909 [26] é uma coleção especializada, desenvolvida com o propósito de impulsionar pesquisas em geração automática de arranjos musicais com ênfase em arranjos para piano. A base de dados é composta por 909 canções populares da China, totalizando aproximadamente 60 horas de arranjos em formato MIDI. Esses arranjos foram elaborados por músicos profissionais e cobrem um período de cerca de 60 anos, abrangendo composições desde a década de 1950 até aproximadamente 2010.

Cada arquivo MIDI do POP909 contém três trilhas distintas: "MELODY", que representa a transcrição da melodia vocal principal; "BRIDGE", dedicada à melodia secundária ou instrumentos solistas que fazem ornamentos e introduções nas músicas; e "PIANO", que corresponde ao corpo principal do acompanhamento, incluindo acordes, arpejos e texturas harmônicas complexas. A combinação das trilhas "BRIDGE" e "PIANO" forma o arranjo de acompanhamento completo para piano. Além disso, os arquivos incluem informações expressivas detalhadas, como controle de velocidade (*velocity*) de cada nota, extraídas com base na análise do áudio original.

O POP909 oferece anotações ricas e detalhadas, como curvas de tempo rotuladas manualmente; acordes e tonalidades extraídos por meio de algoritmos de *Music Information Retrieval* (MIR). As anotações de batida e acordes foram extraídas tanto dos arquivos MIDI quanto dos áudios, enquanto as mudanças de tonalidade foram extraídas exclusivamente dos áudios. O alinhamento temporal entre os arranjos em MIDI e os arquivos de áudio foi feito com precisão, assegurando coerência temporal entre as diferentes modalidades de representação musical.

O diretório raiz do POP909 contém 909 pastas, uma para cada música. Dentro de cada pasta encontram-se o arquivo MIDI final do arranjo, além de: arquivos de texto contendo as anotações de batida, acorde e tonalidade; um arquivo de índice em formato *excel* que agrega metadados relevantes como nome da canção, nome do artista e número de revisões realizadas.

As entradas para o treinamento dos modelos foram selecionadas de modo a refletir os principais parâmetros musicais discutidos anteriormente. A melodia é representada pela variável `note_midi`, que indica a altura das notas ao longo da sequência. A intensidade ou velocity descreve a força com que a nota é executada, refletindo nuances expressivas e dinâmicas da performance. O tempo, por sua vez, é representado pela variável `time_delta`, que expressa o intervalo entre eventos sucessivos. Essa variável é resultado de uma adaptação metodológica: no conjunto de dados original, o tempo é contabilizado de forma contínua, o que pode introduzir variações pouco significativas e prejudicar o aprendizado do modelo. Para mitigar esse efeito, optou-se por transformar a contagem contínua em intervalos discretos, calculando o tempo relativo entre eventos sucessivos, o que resultou no `time_delta`. Por fim, a harmonia é representada pelo acorde (*chord*), que constitui a saída a ser predita [6].

Essas escolhas se justificam pela interdependência estrutural entre os elementos musicais: a harmonia se forma, em muitos casos, a partir das notas melódicas e, ao mesmo tempo, orienta seu desenvolvimento; a intensidade confere peso expressivo e hierarquia perceptiva às notas; e o tempo organiza esses eventos em padrões reconhecíveis [6]. Assim, as entradas do modelo foram construídas com três variáveis contínuas: `note_midi`, *velocity* e `time_delta` — normalizadas via *Z-score*, enquanto a representação da tonalidade (*key*) e da saída (*chord*) foi codificada por *one-hot encoding*.

Nas próximas seções e subseções, explicamos as abordagens e modelos utilizados. Tratamos os eventos como séries temporais, como visto na Figura 2.

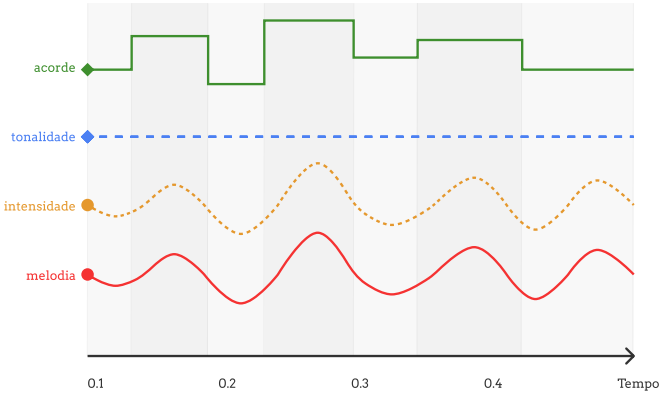

Figure 2: Estrutura das músicas como série temporal.

3.2 Arquitetura LSTM aplicada à Previsão de Acordes

A arquitetura *Long Short-Term Memory* (LSTM) foi escolhida por sua reconhecida capacidade de modelar sequências temporais longas com dependências complexas, o que é particularmente adequado para tarefas musicais em que a relação entre notas e acordes se estende ao longo do tempo. Este tipo de rede neural recorrente supera as limitações das RNNs tradicionais ao incorporar mecanismos de memória e esquecimento, possibilitando o aprendizado de padrões temporais de longo alcance, essenciais na progressão harmônica.

O modelo construído segue uma arquitetura sequencial composta por duas camadas LSTM: uma primeira camada com 128 unidades e `return_sequences` setado para "True", permitindo o fluxo completo da sequência temporal para a próxima camada; uma segunda camada LSTM com 64 unidades, seguida por camadas densas e de regularização via *Dropout* com valor 0.3.

A criação das sequências para entrada no modelo foi realizada com base em janelas deslizantes, utilizando a função `create_sequences`[3]. Essa abordagem estrutura as entradas como janelas temporais de tamanho fixo (`window_size`), com avanço determinado por um passo (stride).

A variação do parâmetro `window_size` afeta diretamente a quantidade de contexto musical fornecido à rede para prever o próximo acorde. Janelas maiores (e.g., 48) oferecem mais contexto harmônico e rítmico, potencialmente favorecendo a previsão de acordes dependentes de estruturas melódicas mais longas; contudo, podem contribuir com alguma entropia na hora da troca de acordes. No caso de janelas curtas (e.g., 3), promovem respostas mais locais, podendo ser úteis para acordes com forte dependência imediata das notas anteriores, o que acontece em alguns casos, mas na música não é regra.

O *stride* determina o intervalo de avanço da janela de entrada ao gerar os pares entrada-saída. Como evidenciado por Nayebi et al. [17], e também testado aqui, um *stride* pequeno (e.g., 1) gera sequências mais sobrepostas e detalhadas, aumentando a densidade dos dados de treino, o que pode beneficiar o aprendizado, mas torna a quantidade de dados e processamento mais complexa. Os *strides* maiores (e.g., 6) reduzem o custo computacional, mas podem impactar negativamente a generalização se o modelo for treinado com menos diversidade de contexto.

Ao projetar redes neurais, a seleção das funções de ativação para as camadas densas é crucial. A função Tanh (tangente hiperbólica) se destaca por sua capacidade de preservar o gradiente próximo à origem, o que é benéfico para modelar relações contínuas com maior simetria. Em contrapartida, a função ReLU (Unidade Linear Retificada), embora acelere o aprendizado e minimize a saturação, pode levar a "neurônios mortos". Isso ocorre quando os neurônios param de aprender, um problema especialmente comum em tarefas com dados esparsos ou altamente codificados.

Para a camada de saída em modelos de classificação multiclasse, a escolha da função de ativação é fundamental. A função Softmax é a mais indicada nesse cenário, pois transforma os valores de saída em uma distribuição de probabilidade, na qual a soma das probabilidades de todas as classes é igual a 1. Isso impõe uma decisão exclusiva, ou seja, a escolha de um único acorde por instante.

No caso da classificação de um único acorde por janela temporal, a *Softmax* é a escolha mais apropriada. Ela garante que o modelo forneça uma probabilidade para cada classe, permitindo a seleção da classe com maior probabilidade como a previsão final, o que se alinha com a natureza da tarefa.

3.3 Arquitetura *Random Forest* aplicada à Previsão de Acordes

Enquanto redes LSTM são particularmente adequadas para modelar dependências temporais complexas, algoritmos de aprendizado de máquina tradicionais como *Random Forest* também podem ser aplicados à tarefa de previsão de acordes, oferecendo vantagens em termos de interpretabilidade e eficiência computacional [21]. O *Random Forest* foi escolhido como abordagem comparativa por sua capacidade de lidar com relações não lineares entre features e por sua robustez a outliers e dados não normalizados.

O modelo de *Random Forest* implementado segue uma arquitetura de ensemble, combinando múltiplas árvores de decisão para reduzir overfitting e melhorar a generalização. Diferentemente da abordagem LSTM, que processa sequências temporais diretamente, o *Random Forest* requer que as relações temporais sejam explicitamente codificadas como *features* através de janelas deslizantes.

Os dados foram organizados de forma similar à abordagem LSTM, utilizando os mesmos arquivos MIDI do *dataset* POP909. A construção das janelas temporais foi adaptada para o contexto do *Random Forest*:

- `window_size`: Adotou-se uma janela simétrica de 16 amostras (8 pontos anteriores e 8 posteriores ao instante atual), permitindo a captura de contexto temporal bidirecional para as predições.
- *Feature vector*: Cada instância foi representada como um vetor achatado (*flattened*) contendo todas as features da janela temporal, concatenado com a codificação *one-hot* da tonalidade atual.

O modelo *Random Forest* foi configurado com 400 árvores de decisão (*n_estimators*), sem restrição de profundidade máxima e exigindo um mínimo de 2 amostras para a divisão de nós.

A seleção de hiperparâmetros foi realizada por meio de Grid Search com validação cruzada (3-fold). Para a construção das árvores da floresta, o critério de divisão dos nós adotado foi a impureza de Gini, padrão em algoritmos de classificação.

4 RESULTADOS E DISCUSSÕES

Esta seção apresenta os resultados dos modelos Random Forest, LSTM e BLSTM na previsão de acordes musicais, com análise quantitativa e até qualitativa de seu desempenho e capacidade de representar padrões temporais e coerência harmônica.

4.1 Experimentos com o *Random Forest*

O *dataset* foi utilizado da mesma forma que para o modelo LSTM, com algumas variações na estruturação do código. A tonalidade da música foi adicionada como uma *feature* categórica, codificada por meio de *One-Hot Encoding*. Em seguida, o conjunto de dados foi

[3]Função desenvolvida pelos autores para segmentar os dados em janelas temporais de tamanho fixo e avanço controlado (*stride*), agrupando características numéricas e tonais de cada trecho musical.

dividido em 80% para treinamento e 20% para teste, aplicando-se amostragem estratificada para garantir uma distribuição balanceada das classes de acordes em ambas as subamostras. Por fim, as variáveis numéricas foram normalizadas utilizando o *Z-score*.

A otimização dos hiperparâmetros foi realizada por meio de *Grid Search*. Conforme a saída do terminal, os melhores parâmetros encontrados foram: 400 estimadores, `max_depth=None` e `min_samples_split=2`. Com esses hiperparâmetros, o modelo foi avaliado no conjunto de teste, obtendo uma acurácia global de 77,01%.

A média ponderada do *F1-score* foi de 0,77, indicando desempenho robusto e equilibrado, considerando o suporte de cada classe. A média macro do *F1-score* foi de 0,67, sugerindo certa variabilidade entre as classes, como também observado no modelo LSTM.

O modelo demonstrou excelente capacidade de generalização para os acordes maiores e menores mais frequentes no conjunto. Por exemplo, os acordes "F#:min" (*F1-score* de 0,83), "D:min" (0,83), "B:min" (0,82) e "C:min" (0,80) foram classificados com alta eficácia.

Acordes com sétima, como "A#:min7" (0,72) e "B:min7" (0,77), tiveram desempenho dentro da média geral, o que é positivo, considerando que esses acordes compartilham contexto com suas versões sem a sétima. No entanto, acordes maiores com sétima, como "C#:maj7" (0,42) e "E:maj7" (0,39), apresentaram resultados inferiores, possivelmente devido à sua sonoridade mais "tensa", característica que os torna distintos de seus equivalentes comuns.

Observou-se dificuldade na classificação de acordes com baixo suporte e maior complexidade harmônica. A classe "F#:dim" teve o pior desempenho (*F1-score* de 0,10), seguida por "A:aug" (0,25). Esses acordes, por sua sonoridade marcante e função harmônica instável, são pouco utilizados em músicas populares (como no *dataset* Pop909), o que limita a quantidade de exemplos disponíveis para aprendizado. Historicamente, acordes com essas características chegaram a ser considerados "pagãos" ou "satânicos" pela igreja, dada a tensão que provocam e a dificuldade de harmonização vocal [9]. Esses resultados indicam que, apesar da boa performance geral, o modelo apresenta limitações ao generalizar padrões de classes sub-representadas no conjunto de dados.

A matriz de confusão construída ao final da execução deste modelo, como demonstra a Figura 3 mostra que o mesmo confunde o acorde de C:maj com G:maj e F:maj. Todos esses acordes pertencem a um mesmo campo harmônico e, portanto, podem aparecer em muitas melodias semelhantes acompanhadas por algum desses três acordes, o que explicaria a confusão. Contudo, essa confusão é pequena (o maior valor é 249 para o C:maj e G:maj) frente ao número total de acertos na classe C:maj, que foi de 5460.

Ao analisar a matriz de confusão, percebeu-se que era possível melhorar os resultados para acordes diminutos aumentando a amostra sem adicionar nada aos dados. As amostras de acordes diminutos são poucas e eles ainda aparecem no *dataset* com nomes diferentes, reduzindo ainda mais as amostras. Por exemplo, um acorde de "C:dim" é formado pelas notas: C + D#(Eb) + F#(Gb) + A. Esse conjunto de notas é o mesmo nos acordes de D#:dim, F#:dim e A, conforme a Tabela 1:

Dessa forma, é possível reduzir o número de acordes diminutos sem diminuir o número de amostras, ou seja, aumentar o número de amostras para cada acorde diminuto dada a semelhança. Por exemplo, as amostras para D#:dim, F#:dim e A:dim migrariam todas para C:dim, pois a construção dos acordes é a mesma. Isso teria impacto na forma como a coluna da tonalidade deveria ser construída, mas é contornável. Perceba que, ao invés de termos doze acordes diminutos, agora teremos apenas três, pois cada acorde diminuto em formato de tetrade contém 4 notas; na escala cromática são doze, que, divididos por quatro, resultam em 3, conforme pode ser mais facilmente compreendido analisando a Tabela 2.

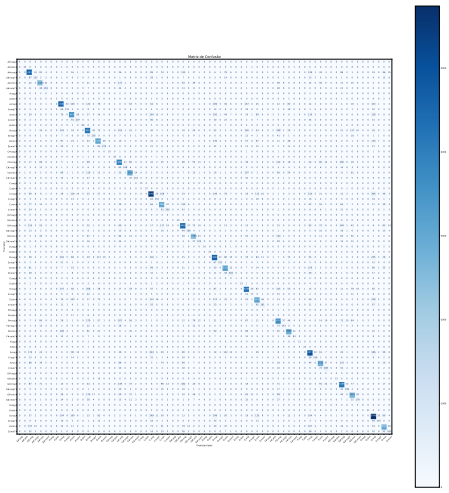

Figure 3: Matriz de confusão do *Random Forest*

Table 1: Notas que compõem os acordes diminutos.

Acorde	Tônica	3ª	5ª	7ª
C:dim	C	D#	F#	A
D#:dim	D#	F#	A	C
F#:dim	F#	A	C	D#
A:dim	A	C	D#	F#

Table 2: Acordes diminutos com semelhança.

Acorde 1	Acorde 2	Acorde 3	Acorde 4
C:dim	D#:dim	F#:dim	A:dim
C#:dim	E:dim	G:dim	A#:dim
D:dim	F:dim	G#:dim	B:dim

4.2 Resultados no modelo LSTM

O modelo LSTM proposto foi submetido a algumas configurações experimentais, nas quais se variaram parâmetros fundamentais como o tamanho da janela temporal (`window_size`), o passo de avanço (*stride*) e as funções de ativação utilizadas nas camadas densas. O objetivo foi avaliar o impacto dessas escolhas sobre o desempenho do modelo na tarefa de previsão de acordes musicais, utilizando como base a sequência melódica e a tonalidade. Contudo, não foi possível testar um número muito grande de parâmetros por conta de limitações computacionais durante os experimentos. Pretende-se continuar nesta pesquisa e melhorar o modelo utilizando um *Grid*

Search melhor orquestrado, de forma que se possa extrair a real melhor configuração para o presente trabalho.

A avaliação do modelo foi conduzida com base em métricas clássicas de classificação multiclasse, incluindo a acurácia geral, como métrica agregada de desempenho, *Precision*, *Recall* e *F1-score*, calculados por cada classe e também de forma macro e ponderada (*weighted average*). Também foram realizadas análises qualitativas da matriz de confusão, com atenção especial ao comportamento do modelo frente às classes mais frequentes e às de menor ocorrência. Tais métricas são particularmente relevantes em domínios com alto desbalanceamento entre classes, como é o caso da distribuição de acordes no *dataset* POP909, que apresenta uma predominância de acordes maiores e menores em detrimento de acordes diminutos e aumentados.

Na primeira rodada, a configuração foi para janela com tamanho 16 e um passo de 1 com ativação tangente hiperbólica. O treinamento progrediu por 69 épocas antes de ser interrompido pelo *EarlyStopping*. A acurácia final de validação atingiu aproximadamente 61%. A avaliação no conjunto de teste resultou em uma acurácia geral de 0.61 e *F1-score* (*weighted avg*): 0.60. Enquanto que o *F1-score* (*macro avg*): 0.44.

A notável diferença entre a média ponderada e a macro média do *F1-score* pode indicar um desequilíbrio no desempenho entre as classes, isso porque o *dataset* é desbalanceado e músicas que utilizam acordes diminutos e/ou aumentados são muito raras. O modelo obteve bom desempenho em acordes com alto suporte (muito comuns no *dataset*), como F#:min (*F1-score* de 0.69) e C#:min (*F1-score* de 0.67). Contudo, demonstrou dificuldade em prever acordes raros, como D#:aug, que obteve um *F1-score* de 0.00.

Na segunda rodada, foram feitas algumas mudanças para avaliar o impacto da função de ativação. O experimento foi repetido, substituindo a ativação de tangente hiperbólica pela ReLU na camada densa. O treinamento estendeu-se por 86 épocas. Os resultados no conjunto de teste foram ligeiramente inferiores aos da configuração base. A acurácia geral ficou em torno de 60% e *F1-score* (*weighted avg*) e *F1-score* (*macro avg*) ficaram em: 0.58 e 0.44, respectivamente. A comparação sugere que, para esta arquitetura específica, a função de ativação *Tanh* proporcionou um desempenho marginalmente superior à *ReLU*, porém, ao repetir essas rodadas percebeu-se que os valores chegaram a se igualar em alguns momentos.

Na terceira rodada, o passo foi aumentado para 2, mantendo a janela em 16, reduzindo a sobreposição entre as sequências de treinamento e, consequentemente, o tamanho do conjunto de dados. O treinamento foi realizado com a ativação tangente hiperbólica. O modelo foi treinado por 67 épocas, alcançando uma acurácia geral de 61%. A performance final mostrou um *F1-score* (*weighted avg*) em 0.59.

Na matriz de confusão gerada pelo modelo, na Figura 4, observam-se 24 células com coloração mais intensa, indicando um desempenho superior nessas classes. Essas 24 células correspondem aos 12 acordes de cada uma das notas da escala cromática, nos formatos maior e menor. Esse resultado evidencia a capacidade do modelo em prever acordes maiores e menores, que também são os mais abundantes no conjunto de dados, em contraste com acordes diminutos, aumentados, com 7ª maior, entre outros.

Houve uma pequena queda de performance em comparação com o passo de tamanho 1, indicando que a maior densidade de dados

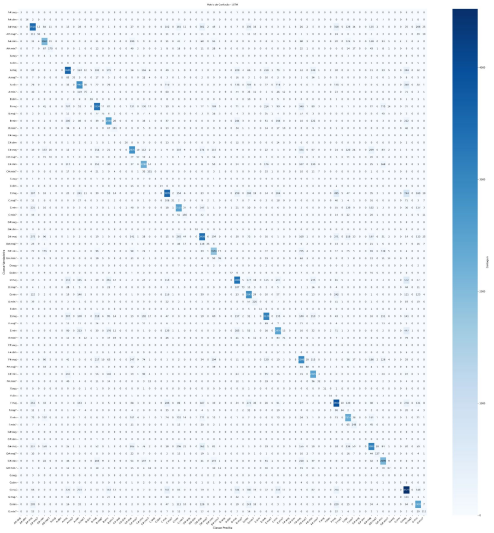

Figure 4: Matriz de confusão do modelo LSTM

gerada por um passo menor pode ser benéfica para o aprendizado do modelo, mesmo que à custa de um maior tempo de processamento. Contudo, os valores não são demasiadamente discrepantes.

Na 4ª rodada, o tamanho da janela foi aumentado para 32, fornecendo ao modelo um contexto histórico mais longo para cada previsão. Esta rodada utilizou um passo de tamanho 2 e a ativação foi a *ReLU*. Os resultados mostraram uma queda de desempenho um pouco maior, dado que a acurácia geral agora ficou em torno de 55%. Já o *F1-score* (*weighted avg*) ficou em 0.54.

Contrariamente ao esperado pela equipe da pesquisa, dobrar o tamanho da janela não melhorou a capacidade de previsão do modelo, resultando no desempenho mais baixo entre as configurações testadas. Isso pode sugerir que um contexto mais longo introduziu ruído ou que a arquitetura do modelo não foi capaz de extrair eficientemente informações relevantes de sequências tão longas. Do ponto de vista musical, esse ruído pode ser uma confusão quando em uma janela grande aparece um número maior de acordes acompanhando a melodia. Um compasso pode ter desde uma única nota até dezenas, distribuídas similarmente à demonstração da Figura 5, mas no caso do *dataset* POP909, as músicas apresentam compassos com valores que variam muito, mas raramente ultrapassando 20 notas por compasso. Janelas grandes de tamanho maior que 20, por exemplo, podem acabar trazendo mais acordes para um mesmo contexto melódico dentro da janela, e isso pode acabar gerando ruído para o modelo.

A análise dos resultados indica que a configuração com janela de tamanho 16 e passo tamanho 1 (`window_size=16, stride=1`), forneceu o melhor desempenho geral, atingindo uma acurácia de 0.61 e um *F1-score* ponderado de 0.60. Para avaliar os impactos da função de ativação, se *ReLU* ou *Tanh*, seriam necessários mais experimentos avaliando os casos, mas, nos experimentos deste trabalho, a função *Tanh* apresentou melhores resultados como era de se esperar.

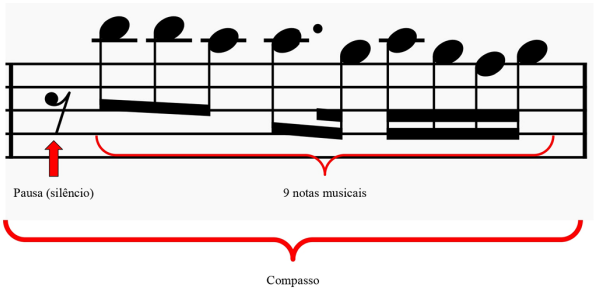

Figure 5: Compasso no qual se encontram 9 notas.

4.3 Experimentos com o modelo BLSTM

A avaliação dos modelos *Random Forest*, LSTM e BLSTM revelou percepções importantes sobre a tarefa de reconhecimento de acordes. O modelo LSTM demonstrou sensibilidade notável aos hiperparâmetros, com a configuração de janela, passo e função de ativação semelhantes ao BLSTM alcançando o melhor desempenho, com 0,61 de acurácia e 0,60 de *F1-score* ponderado. Percebeu-se que a ampliação da janela de contexto para 32 eventos impactou negativamente o desempenho, sugerindo, possivelmente, que um contexto excessivamente longo não é benéfico para a arquitetura atual. Um desafio persistente para o LSTM, assim como para os demais modelos, foi o desbalanceamento de classes que pode ser observado na Tabela 3, resultando em classificações imprecisas para acordes menos frequentes.

Table 3: Distribuição de classes no dataset nomeadas por tipo de acorde.

Acorde	G:maj	C:maj	F:maj	...	G#:aug	A#:aug	E:aug
Freq.	36989	35867	32366	...	22	19	7

Já o modelo *Random Forest* mostrou-se eficaz, particularmente para as categorias de acordes mais comuns. A acurácia geral e os *F1-scores* para as classes majoritárias foram promissores. No entanto, o desempenho em acordes raros e complexos reiterou a necessidade de estratégias que mitiguem o impacto do desbalanceamento de classes.

Por fim, a arquitetura BLSTM proposta exibiu proficiência no reconhecimento de acordes em sequências musicais, apresentando um desempenho consistente para as classes mais representativas do conjunto de dados, como pode ser visto na matriz de confusão na Figura 6. Mas também se mostrou interessante com classes mais raras, como os acordes diminutos.

Em síntese, todos os modelos (*Random Forest*, LSTM e BLSTM) apresentaram desempenho relevante na previsão das classes de acordes, especialmente considerando o total de 72 classes avaliadas. O principal desafio observado foi o desbalanceamento entre classes, um aspecto que talvez possa ser mitigado por meio de técnicas de balanceamento de dados, mesmo diante da complexidade do conjunto. Acordes diminutos e aumentados, por exemplo, são naturalmente raros em composições musicais e apresentam menor intercambiabilidade com outros acordes. Além disso, melodias associadas a esses acordes costumam exibir características peculiares,

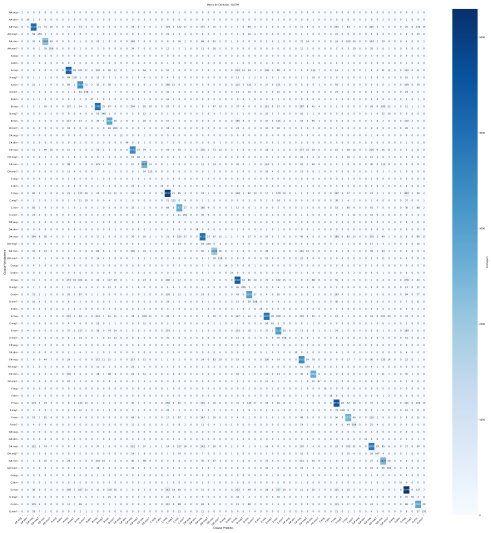

Figure 6: Matriz de confusão do BLSTM

como repetição de uma única nota ao longo de todo o compasso. Assim, as dificuldades decorrentes do desbalanceamento configuram um ponto crítico de aprimoramento para trabalhos futuros, independentemente da arquitetura de modelo utilizada.

4.4 Análise Comparativa entre Modelo e Predições Humanas

Após as últimas execuções do processo de treinamento do modelo e, consequentemente, a avaliação da qualidade dos mesmos através dos dados do conjunto de teste, decidiu-se avaliar a capacidade dos modelos em predizer uma melodia completamente diferente das melodias que compunham o *dataset* utilizado no processo.

Para esta tarefa, foi elaborada uma melodia de estrutura bastante simples, composta por compassos contendo apenas três figuras rítmicas: uma mínima pontuada e duas colcheias. Em cada compasso, todas as figuras executam a mesma nota, com exceção do penúltimo compasso, que recebeu uma nota de passagem (no caso, um B) para criar uma sensação de finalização na frase. A sequência, formada por nove compassos, apresentou as seguintes notas: C, D, E, F, G, F, E, D, (B de passagem) e, por fim, C. A partitura da melodia pode ser vista na Figura 7

Foram selecionados dois músicos com experiência em acompanhamento e harmonização de melodias por percepção auditiva (harmonização de ouvido), a fim de propor arranjos harmônicos para a melodia de teste. O procedimento foi conduzido de forma independente, sem qualquer comunicação ou acesso prévio às anotações do outro participante, de modo a evitar influências cruzadas e garantir a autenticidade das escolhas harmônicas. Como consequência, as harmonizações resultantes apresentaram variações, ainda que pequenas, evidenciadas na Figura 7.

Embora tenham sido observadas diferenças entre as harmonizações propostas, com exceção das realizadas pelos músicos humanos, que apresentaram elevada similaridade entre si, é importante destacar que a melodia em questão admite um número considerável de possibilidades harmônicas. Inclusive, a manutenção de

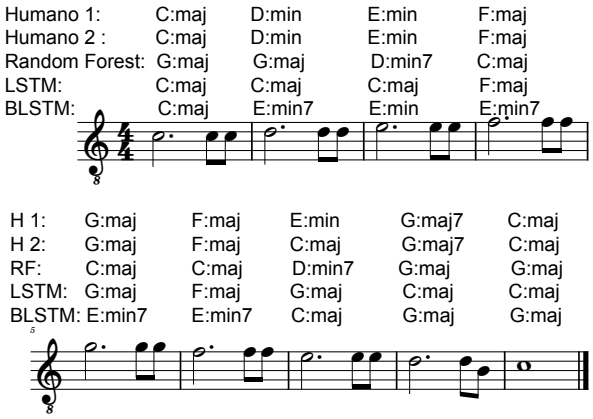

Figure 7: Resultados da predição dos acordes para a melodia proposta.

um único acorde de C:maj ao longo de toda a peça constitui uma alternativa musicalmente válida. Essa característica de flexibilidade harmônica é reconhecida no meio musical e ensino informal de música, sendo comum a utilização dessa melodia, por professores de música em redes sociais, como exemplo para demonstrar as múltiplas abordagens possíveis na harmonização ou para exercícios vocais.

A harmonização de uma peça musical tende a apresentar configuração rígida apenas quando o ouvinte já está condicionado a uma versão específica, dificultando o desprendimento daquela sonoridade. No entanto, não existe uma forma única e definitiva de construir uma harmonia, nem para a melodia proposta, nem para qualquer outra. O processo depende da percepção individual do arranjador e de sua interpretação harmônica, bem como de seu conhecimento técnico: quanto maior a familiaridade com conceitos de harmonia e com diferentes estilos musicais, maior será a capacidade de propor variações harmônicas para uma mesma obra [25].

Diante dos resultados apresentados na Figura 7, é importante destacar que, por operarem com janelas temporais, os modelos não dispõem do mesmo referencial adotado pelos músicos — o início da música. Na prática, músicos tendem a assumir que o primeiro acorde corresponde à tonalidade da peça, o que influencia diretamente o encadeamento harmônico subsequente. As tensões sonoras geradas pelo acorde inicial podem, portanto, direcionar a escolha dos acordes seguintes.

Ao analisar a sequência produzida pelo modelo *Random Forest*, observa-se que ele inferiu o primeiro acorde como sendo G:maj, divergindo de todas as demais harmonizações. Essa escolha inicial possivelmente impactou todo o restante da sequência. Do ponto de vista tonal, essa seleção pode ser considerada inadequada, já que a música está em C:maj e o primeiro compasso é inteiramente ocupado pela nota C, a qual não integra o acorde de G:maj. Essa discrepância se repete em grande parte dos acordes sugeridos pelo modelo, com exceção dos compassos cinco e oito, nos quais as notas da melodia estão presentes nos acordes gerados.

Os modelos que produziram harmonizações mais próximas das sugeridas pelos músicos foram o LSTM e o BLSTM. No entanto, o LSTM apresentou desempenho superior ao concluir a melodia com o acorde C:maj, enquanto o BLSTM optou por G:maj, acorde que não corresponde à tonalidade da peça e que, além disso, não contém a nota C em sua composição. Por outro lado, no segundo compasso, o BLSTM sugeriu o acorde E:min7, cuja estrutura contém a nota D em função da 7ª menor adicionada, ajustando-se adequadamente ao contexto harmônico e conferindo à progressão uma sonoridade mais sofisticada.

5 CONCLUSÃO

A predição de acordes é fortemente dependente da percepção humana, uma vez que, apesar de seus fundamentos físicos, a harmonia é definida pela experiência auditiva. Neste trabalho, não foi adotada avaliação perceptiva humana, mas os modelos enfrentaram o desafio de sugerir harmonias complexas a partir das melodias e as comparamos com anotações humanas. Os resultados apontam para um caminho bom, mas que ainda carece de muito avanço e amadurecimento. O conjunto de classes é muito grande e, à medida que o estilo de música muda, pode crescer ainda mais.

Os modelos LSTM e BLSTM apresentaram desempenhos semelhantes na tarefa de previsão de acordes, com o LSTM obtendo resultados ligeiramente superiores quando comparado às anotações humanas de referência. Observou-se que o BLSTM tende a permanecer por mais tempo em um único acorde, demonstrando menor variação harmônica ao longo da sequência, enquanto o LSTM apresentou maior dinamicidade e acertos distribuídos em boa parte dos compassos, conforme ilustrado na Figura 7. O modelo Random Forest, embora tenha obtido métricas numéricas mais elevadas durante o treinamento, mostrou maior divergência em relação às anotações humanas, indicando menor aderência musical nas predições. De modo geral, verificou-se que os modelos são capazes de identificar corretamente os acordes, porém com predições levemente deslocadas no tempo, sugerindo um atraso na correspondência entre o acorde previsto e o acorde real.

Os modelos avaliados neste trabalho demonstram boa capacidade de aprendizado para a tarefa proposta. No entanto, o conjunto de dados ainda não é capaz de fornecer uma grande generalização para as 72 classes que se pretendia avaliar. Acordes menos comuns precisam receber mais atenção em trabalhos futuros.

Para as próximas pesquisas, sugere-se o aumento do conjunto de dados visando ofertar uma quantidade mais equilibrada de amostras de ambas as classes, de forma que se possa fazer um balanceamento dos dados. Dessa forma, certamente os resultados serão mais estáveis e poderão apresentar predições mais confiáveis. Também, e ainda na linha do conjunto de dados, preencher os mesmos com músicas de culturas mais variadas, o que pode tornar os modelos ainda mais genéricos nos resultados.

Além disso, futuras investigações podem explorar questões de generalização, como: até que ponto o modelo manteria seu desempenho ao ser exposto a músicas em outros idiomas e contextos culturais? Haveria limitações associadas à diversidade cultural das amostras ou às características linguísticas das letras? Tais análises podem contribuir para uma compreensão mais ampla da aplicabilidade e robustez do modelo em cenários multiculturais.

REFERENCES

[1] Carlos V.S. Araujo, Rafael Giusti, and Marco A.P. Cristo. 2018. Um Modelo de Predição para o Sucesso no Mercado Musical. In *Anais Estendidos do XXIV Simpósio Brasileiro de Sistemas Multimídia e Web* (Salvador). SBC, Porto Alegre, RS, Brasil, 37–42. https://doi.org/10.5753/webmedia.2018.4559

[2] Adam Patrick Bell. 2018. *Dawn of the DAW: The studio as musical instrument.* Oxford University Press, Oxford, United Kingdom.

[3] Rolando Benenzon. 1989. *Teoria da musicoterapia.* Grupo Editorial Summus, São Paulo, SP.

[4] Jean-Pierre Briot and François Pachet. 2020. Deep learning for music generation: challenges and directions. *Neural Computing and Applications* 32, 4 (2020), 981–993. https://doi.org/10.1007/s00521-018-3813-6

[5] Kleberson Calanca. 2021. Entre a Era do Rádio e a Bossa-Nova: O avanço das tecnologias de captação, gravação e difusão do som como meios criadores. *Revista Brasileira de Música* n.d., n.d. (2021), n.d. Informações de volume, número e páginas não disponíveis online.

[6] Fabio Adour da Camara. 2008. *Sobre Harmonia: uma proposta de perfil conceitual.* Ph. D. Dissertation.

[7] Cedric De Boom, Stephanie Van Laere, Tim Verbelen, and Bart Dhoedt. 2019. Rhythm, chord and melody generation for lead sheets using recurrent neural networks. In *Joint European conference on machine learning and knowledge discovery in databases*. Springer, Berlin / Heidelberg, Germany, 454–461. https://doi.org/10.48550/arXiv.2002.10266

[8] Agostino Di Scipio. 2000. The technology of musical experience in the 20th century. *Rivista italiana di musicologia* 35, 1/2 (2000), 247–275.

[9] Mateus Rocha Grain. 2023. *Das memórias que cercam os intervalos de três tons: entre sons e conotações.* Ph. D. Dissertation. Universidade do Estado de Santa Catarina - UDESC. Disponível em: https://sistemabu.udesc.br/pergamumweb/vinculos/0000c9/0000c931.pdf.

[10] Mark Granroth-Wilding and Mark Steedman. 2014. A robust parser-interpreter for jazz chord sequences. *Journal of New Music Research* 43, 4 (2014), 355–374. https://doi.org/10.1080/09298215.2014.910532

[11] Lucy Green, Flávia Motoyama Narita, and Luciana Fernandes Hamond. 2022. O que os professores podem aprender com os músicos populares? *Revista Orfeu* 7, 1 (2022), 1–14. https://doi.org/10.5965/2525530407012022e0401

[12] Jack Hardwick. 2024. An LSTM-Based Chord Generation System Using Chroma Histogram Representations. *arXiv preprint arXiv:2405.05240* n.d., n.d. (2024), n.d. https://doi.org/10.48550/arXiv.2405.05240 Informações de volume, número e páginas não disponíveis online.

[13] Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M Dai, Matthew D Hoffman, Monica Dinculescu, and Douglas Eck. 2018. Music transformer. *arXiv preprint arXiv:1809.04281* n.d., n.d. (2018), n.d. https://doi.org/10.48550/arXiv.1809.04281 Informações de volume, número e páginas não disponíveis online.

[14] Yu-Siang Huang, Szu-Yu Chou, and Yi-Hsuan Yang. 2018. Pop music highlighter: Marking the emotion keypoints. *arXiv preprint arXiv:1802.10495* n.d., n.d. (2018), n.d. https://doi.org/10.48550/arXiv.1802.10495 Informações de volume, número e páginas não disponíveis online.

[15] Hyungui Lim, Seungyeon Rhyu, and Kyogu Lee. 2017. Chord generation from symbolic melody using BLSTM networks. *arXiv preprint arXiv:1712.01011* n.d., n.d. (2017), n.d. https://doi.org/10.48550/arXiv.1712.01011 Informações de volume, número e páginas não disponíveis online.

[16] Nolan Monnier, Darien Ghali, and Sophie X. Liu. 2021. FFT and Machine Learning Application on Major Chord Recognition. In *2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE, Jeju Island, Korea, 426–429. https://doi.org/10.1109/ICUFN49451.2021.9528762

[17] Amin Nayebi, Sindhu Tipirneni, Chandan K Reddy, Brandon Foreman, and Vignesh Subbian. 2023. WindowSHAP: An efficient framework for explaining time-series classifiers based on Shapley values. *Journal of biomedical informatics* 144 (2023), 104438. https://doi.org/10.1016/j.jbi.2023.104438

[18] Pariwat Ongsulee. 2017. Artificial intelligence, machine learning and deep learning. In *2017 15th International Conference on ICT and Knowledge Engineering (ICT&KE)*. IEEE, Bangkok · Thailand, 1–6. https://doi.org/10.1109/ICTKE.2017.8259629

[19] Marcelo Brosowicz de Paulo et al. 2022. Geração de Música com Machine Learning. *Florianópolis, SC* n.d., n.d. (2022), n.d. Trabalho de Conclusão de Curso – Universidade Federal de Santa Catarina (UFSC). Disponível em: https://repositorio.ufsc.br/handle/123456789/237814.

[20] Rodolfo Miranda Pereira and Carlos N. Silla. 2017. Using simplified chords sequences to classify songs genres. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, Hong Kong, China, 1446–1451. https://doi.org/10.1109/ICME.2017.8019531

[21] Felix Reuß, Isabella Greimeister-Pfeil, Mariette Vreugdenhil, and Wolfgang Wagner. 2021. Comparison of long short-term memory networks and random forest for sentinel-1 time series based large scale crop classification. *Remote Sensing* 13, 24 (2021), 5000. https://doi.org/10.3390/rs13245000

[22] Curtis Roads. 1996. *The computer music tutorial.* MIT press, Massachusetts, EUA.

[23] Adam Roberts, Jesse Engel, Colin Raffel, Curtis Hawthorne, and Douglas Eck. 2018. A Hierarchical Latent Vector Model for Learning Long-Term Structure in Music. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, Cambridge MA, 4364–4373. https://proceedings.mlr.press/v80/roberts18a.html

[24] Nipun Sharma, Swati Sharma, C Bhanuprakash Reddy, Manisha A, Bahuguna P, and Ankita T. 2025. Analysis of Isotonic Calibration on Gaussian Naïve Bayes Performance for Guitar Chords Classification. In *2025 International Conference on Automation and Computation (AUTOCOM)*. IEEE, Dehradun, India, 15–19. https://doi.org/10.1109/AUTOCOM64127.2025.10956982

[25] Jáderson Aguiar Teixeira. 2015. O ensino musical interdisciplinar de harmonia, contraponto, solfejo e arranjo como estratégia de produção de conhecimento. *www. teses. ufc. br* n.d., n.d. (2015), n.d. Informações de volume, número e páginas não disponíveis online. Disponível em https://repositorio.ufc.br/handle/riufc/14365.

[26] Ziyu Wang, Ke Chen, Junyan Jiang, Yiyi Zhang, Maoran Xu, Shuqi Dai, Xianbin Gu, and Gus Xia. 2020. Pop909: A pop-song dataset for music arrangement generation. *arXiv preprint arXiv:2008.07142* n.d., n.d. (2020), n.d. https://doi.org/10.48550/arXiv.2008.07142 Informações de volume, número e páginas não disponíveis online.

[27] Bruna D Wundervald and Walmes M Zeviani. 2019. Machine learning and chord based feature engineering for genre prediction in popular Brazilian music. *arXiv preprint arXiv:1902.03283* n.d., n.d. (2019), n.d. https://doi.org/10.48550/arXiv.1902.03283 Informações de volume, número e páginas não disponíveis online.