Impacto da Variedade Instrumental na Performance de Modelos de Alinhamento Áudio-Partitura

Luiz Henrique Oliveira Martins

Centro de Informática
UNIVERSIDADE FEDERAL DA PARAÍBA

João Pessoa, 2024

Luiz Henrique Oliveira Martins

Impacto da Variedade Instrumental na Performance de Modelos de Alinhamento Áudio-Partitura

Monografia apresentada ao curso Ciência da Computação
do Centro de Informática, da Universidade Federal da Paraíba,
como requisito para a obtenção do grau de Bacharel em
Ciência da Computação.

Orientador: Thaís Gaudencio do Rêgo

M386i Martins, Luiz Henrique Oliveira.
Impacto da variedade instrumental na performance de modelos de alinhamento áudio-partitura / Luiz Henrique Oliveira Martins. - João Pessoa, 2024.
47 f. : il.

Orientação: Thaís Rêgo.
Coorientação: Yuri Barbosa.
TCC (Graduação) - UFPB/CI.

1. Alinhamento áudio-partitura. 2. Redes neurais siamesas. 3. Dynamic time warping. 4. Variedade instrumental. 5. Qualidade de dados. I. Rêgo, Thaís. II. Barbosa, Yuri. III. Título.

UFPB/CI CDU 004.6

Julho de 2024

Centro de Informática
UNIVERSIDADE FEDERAL DA PARAÍBA

Trabalho de Conclusão de Curso de Ciência da Computação intitulado **Impacto da Variedade Instrumental na Performance de Modelos de Alinhamento Áudio-Partitura** de autoria de **Luiz Henrique Oliveira Martins**, aprovada pela banca examinadora constituída pelos seguintes professores:

Profa. Dra. Thaís Gaudencio do Rêgo
Universidade Federal da Paraíba (UFPB)

Prof. Dr. Yuri de Almeida Malheiros Barbosa
Universidade Federal da Paraíba (UFPB)

Prof. Dr. Carlos Eduardo Coelho Freire Batista
Universidade Federal da Paraíba (UFPB)

João Pessoa, 02 de Julho de 2024

Centro de Informática, Universidade Federal da Paraíba
Rua dos Escoteiros, Mangabeira VII, João Pessoa, Paraíba, Brasil CEP: 58058-600
Fone: +55 (83) 3216 7093 / Fax: +55 (83) 3216 7117

Não exija que as coisas aconteçam como você deseja, mas deseje que elas aconteçam como acontecem.

- Epictetus

AGRADECIMENTOS

Agradeço primeiramente aos meus orientadores, a professora Thaís Gaudencio e o professor Yuri Barbosa. Sua orientação, paciência, e conhecimento foram fundamentais para a realização deste trabalho.

Agradeço ao meu pai, Luiz Martins, e à minha mãe, Patrícia Janaína, por todo o apoio e incentivo durante minha jornada de vida. Vocês me ensinaram valores além do que uma instituição educacional é capaz de ensinar.

Aos meus amigos Pedro e Yago, pela amizade e momentos de distração que me ajudaram a manter a mente clara, meus sinceros agradecimentos.

Agradeço à minha futura esposa Vanessa, por estar ao meu lado, pelo apoio, e pela confiança durante o processo de produção deste trabalho. Te amo.

Agradeço a todos os colegas que conheci na faculdade, que contribuíram de forma direta ou indireta para minha formação pessoal, acadêmica, e profissional.

Por fim, agradeço aos gigantes nos ombros dos quais estou apoiado.

RESUMO

O alinhamento de áudio e partitura é uma área desafiadora do processamento de áudio, especialmente quando envolve uma ampla variedade de instrumentos e estilos. Este estudo investiga o impacto da variedade instrumental no treinamento de modelos de alinhamento áudio-partitura utilizando Redes Neurais Siamesas (SNN) e *Dynamic Time Warping* (DTW). A hipótese central é que a diversidade de instrumentos no conjunto de dados de treinamento melhora a precisão e eficiência do alinhamento. Para testar essa hipótese, foram utilizadas bases de dados de áudio de diferentes instrumentos, incluindo bateria, violão, piano e flauta, e foram criados conjuntos separados para cada tipo de instrumento, além de um conjunto misto. Os modelos foram treinados e avaliados em cenários intra-instrumental, inter-instrumental e misto. Os resultados indicaram que as características intra-instrumentais são mais relevantes para a generalização dos modelos, e que há degradação de performance em bases inter-instrumentais diluídas, destacando a importância da qualidade de dados dentro do mesmo escopo instrumental. Modelos treinados puramente em bases de dados individuais apresentaram melhor desempenho em generalização para outros instrumentos e em cenários mistos.

Palavras-chave: Alinhamento Áudio-Partitura, Redes Neurais Siamesas, Dynamic Time Warping, Variedade Instrumental, Qualidade de Dados.

ABSTRACT

The alignment of audio with its score is a challenging area of audio processing, especially when it involves a wide variety of instruments and styles. This study investigates the impact of instrumental variety on the training of audio-to-score alignment models using Siamese Neural Networks (SNN) and Dynamic Time Warping (DTW). The central hypothesis is that the diversity of instruments in the training dataset improves the accuracy and efficiency of the alignment. To test this hypothesis, audio databases of different instruments, including drums, guitar, piano, and flute, were used, and separate datasets were created for each type of instrument, as well as a mixed dataset. The models were trained and evaluated in intra-instrumental, inter-instrumental, and mixed scenarios. The results indicated that intra-instrumental characteristics are more relevant for model generalization and that there is a performance degradation in diluted inter-instrumental databases, highlighting the importance of data quality within the same instrumental scope. Models trained purely on individual databases showed better performance in generalizing to other instruments and in mixed scenarios.

Keywords: Audio-to-Score Alignment, Siamese Neural Networks, Dynamic Time Warping, Instrument Variety, Data Quality.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

AI	*Artificial Intelligence*	Inteligência Artificial
ASAP	*Aligned Scores and Performances*	Alinhamento de Partituras e Performances
CNN	*Convolutional Neural Network*	Rede Neural Convolucional
DPP	*Determinantal Point Processes*	Processos de Pontos Determinantes
DTW	*Dynamic Time Warping*	Alinhamento Dinâmico de Tempo
GPU	*Graphics Processing Unit*	Unidade de Processamento Gráfico
MIDI	*Musical Instrument Digital Interface*	Interface de Instrumento Musical Digital
MIR	*Music Information Retrieval*	Recuperação de Informaçes Musicais
PCA	*Principal Component Analysis*	Análise de Componentes Principais
PDS	*Poisson Disk Sampling*	Amostragem de Disco de Poisson
ReLU	*Rectified Linear Unit*	Unidade Linear Rectificada
SNN	*Siamese Neural Network*	Rede Neural Siamesa
STFT	*Short-Time Fourier Transform*	Transformada de Fourier de Tempo Curto

1 INTRODUÇÃO

À medida que a inteligência artificial continua a expandir suas fronteiras em campos como reconhecimento de imagem e processamento de linguagem natural, também surgem oportunidades significativas para sua aplicação na música. Enquanto avanços notáveis têm sido feitos com modelos de linguagem de grande escala e modelos de difusão, que facilitam consideravelmente a geração de texto e imagens, a identificação de objetos em imagens tem visto progressos substanciais. Contudo, comparativamente, os desafios no processamento de áudio ainda oferecem um vasto campo a ser explorado.

É reconhecido que a qualidade dos dados influencia diretamente o desempenho dos modelos de inteligência artificial (CORTES; JACKEL; CHIANG, 1994; GOSWAMI, 2020). Novas descobertas sugerem que a variedade dos dados de treinamento também pode aprimorar a generalização dos modelos (LINDSAY et al., 2023). No contexto de alinhamento áudio-partitura, explorar uma ampla variedade de instrumentos pode oferecer *insights* sobre como os modelos percebem e processam nuances acústicas distintas, contribuindo assim para uma aplicação mais robusta e precisa.

Seguindo a linha de modelos utilizada por AGRAWAL; DIXON (2020) que apresentaram performance superior à modelos tradicionais, este estudo analisa a performance de *Siamese Neural Networks* (SNN) com *Dynamic Time Warping* (DWT) na tarefa de alinhamento de faixas musicais, com foco na variedade de instrumentos utilizados durante o treinamento. Através deste trabalho, buscamos aprimorar a tecnologia de alinhamento áudio-partitura e explorar em que medida a diversidade de dados pode enriquecer o aprendizado de máquina, no contexto de análise de áudio.

1.1 Problema

A área de processamendo de áudio com redes neurais enfrenta desafios significativos, especialmente quando envolve uma ampla variedade de instrumentos e estilos (HERNANDEZ-OLIVAN; BELTRÁN, 2021). Apesar dos avanços em inteligência artificial e em processamento de áudio, ainda existem barreiras técnicas significativas, que limitam a eficácia das ferramentas de alinhamento. Dificuldades na captura precisa de características temporais e harmônicas de diferentes instrumentos podem comprometer a precisão e a eficiência dos sistemas atuais. O problema se intensifica com a introdução de variações como com-

plexidade musical, gênero, e variedade instrumental. Cada aspecto introduz nuances que os modelos podem ter dificuldades em capturar. Logo, é desejável isolar e explorar os efeitos da variedade de dados na performance destes modelos.

1.2 Hipótese

A hipótese central deste estudo é que a variedade de instrumentos utilizados no treinamento de SSNs, combinada com o DWT, tem um impacto significativo na precisão e eficiência do alinhamento. Prevemos que modelos treinados com um espectro mais amplo de características acústicas decorrentes de diferentes instrumentos serão capazes de capturar com maior fidelidade as nuances essenciais para um alinhamento preciso entre faixas musicais. Esta hipótese se baseia na premissa de que uma maior diversidade nos dados de treinamento leva a uma melhor generalização do modelo (RAHIMI et al., 2023; YU; KHADIVI; XU, 2022), permitindo assim uma aplicação mais eficaz em variados contextos musicais.

1.3 Objetivo Geral

Este trabalho propõe o desenvolvimento e avaliação de modelos de alinhamento áudio-partitura usando *Siamese Neural Networks* e *Data Time Warping*, que incorpore uma variedade de instrumentos no treinamento. O objetivo é verificar se essa diversidade instrumental melhora a precisão do alinhamento de faixas musicais.

1.4 Objetivos Específicos

1. **Coleta e Normalização de Dados**: Agrupar dados de áudio de diferentes instrumentos, normalizando-os para uma forma espectral similar para garantir consistência no treinamento.
2. **Desenvolvimento de Conjuntos de Dados**: Criar conjuntos de dados separados para cada tipo de instrumento e conjuntos misturados, para analisar a influência da variabilidade dos dados no treinamento do modelo.
3. **Treinamento do Modelo**: Implementar e treinar modelos SNN usando dados de instrumentos individuais e combinados, aplicando o algoritmo DWT para alinhamento musical.

1.5 Estrutura da Monografia

1.6 Estrutura da Monografia

No Capítulo 2, a fundamentação teórica é detalhada. Conceitos essenciais como SNN e DTW são discutidos. Este capítulo inclui explicações sobre a função de Perda Contrastiva, algoritmos de alinhamento como DTW, e métodos de extração de características de áudio, como a Transformada de Fourier de Tempo Curto (do inglês, *Short-Time Fourier Transform* - STFT).

No Capítulo 3, a revisão de literatura é apresentada, abordando trabalhos relevantes na área de alinhamento áudio-partitura e diversidade de dados em aprendizado de máquina. Estudos que utilizam arquiteturas de Redes Siamesas e técnicas de alinhamento de áudio com partitura, além de pesquisas sobre a importância da diversidade de dados no treinamento de modelos, são discutidos.

No Capítulo 4, a metodologia utilizada no estudo é descrita em detalhes. Os conjuntos de dados utilizados, o processo de extração e pré-processamento dos dados, e as arquiteturas de modelos implementadas são apresentados. Este capítulo também aborda os procedimentos de treinamento e avaliação dos modelos.

No Capítulo 5, os resultados obtidos nos testes de alinhamento são apresentados. Os resultados são analisados e comparados em cenários intra-instrumental, inter-instrumental e misto, utilizando as métricas de avaliação definidas.

No Capítulo 6, a discussão dos resultados é realizada, interpretando os achados do estudo e suas implicações para a área de alinhamento áudio-partitura e aprendizado de máquina. Este capítulo destaca as observações mais relevantes sobre o desempenho dos modelos.

Finalmente, no Capítulo 7, as conclusões do trabalho são apresentadas. As principais contribuições do estudo são sintetizadas, limitações são apresentadas, e direções para futuras pesquisas na área de alinhamento áudio-partitura e na aplicação de técnicas de aprendizado de máquina ao processamento de áudio são sugeridas.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, apresentaremos os conceitos teóricos e as bases metodológicas que sustentam este estudo. Inicialmente, abordaremos as técnicas de alinhamento áudio-partitura, uma área crucial na recuperação de informação musical. Em seguida, discutiremos a extração de características de áudio, destacando o método STFT. Posteriormente, exploraremos as SNNs e sua aplicação no alinhamento áudio-partitura. Finalmente, discutiremos o algoritmo DTW e as métricas de avaliação utilizadas para medir a precisão do alinhamento.

Primeiramente, vamos explorar o conceito de alinhamento áudio-partitura. Entender como sincronizar gravações de áudio com partituras musicais é essencial para contextualizar os métodos subsequentes discutidos neste capítulo.

2.1 Alinhamento Áudio-Partitura

Alinhamento áudio-partitura é uma técnica fundamental na área de Recuperação de Informação Musical (do inglês, *Musical Information Retrieval* - MIR), que sincroniza gravações de áudio com suas respectivas partituras musicais. Essa tecnologia pode ser dividida em duas categorias principais: alinhamento *online*, que ocorre em tempo real durante performances ao vivo, e alinhamento *offline*, realizado em gravações previamente feitas (ARZT, 2016).

Os usos do alinhamento áudio-partitura incluem a capacidade de acompanhar automaticamente músicos durante performances ao vivo, especialmente útil em peças longas, e fornecer *feedback* imediato durante prática instrumental. O processo enfrenta uma variedade de desafios técnicos. Diferentes afinações entre a partitura e a performance, variações de velocidade, saltos para frente e para trás – comuns em ensaios – e improvisações introduzidas pelo músico (MORSI; SERRA, 2022). Todas estas nuancias podem complicar significativamente o alinhamento. Tais variações exigem que os sistemas sejam altamente adaptáveis e sensíveis às variações da performance musical.

Para realizar o alinhamento, métodos como DTW e Cadeias Ocultas de Markov são frequentemente utilizados (AGRAWAL, 2022; MÜLLER, 2015). Na área de redes neurais, redes recorrentes, convolucionais, e siamesas são métodos de alinhamento comuns na literatura (AGRAWAL; WOLFF; DIXON, 2021; KWON; JEONG; NAM, 2017).

Uma etapa crítica no processo de muitos métodos de alinhamento, com redes neurais, envolve a síntese de áudio usando programas de sintetização como *FluidSynth* (AGRAWAL; DIXON, 2020; KWON; JEONG; NAM, 2017; MORSI; SERRA, 2022). Este áudio é utilizado para criar uma representação padronizada do objetivo, facilitando o alinhamento inicial com a gravação ao vivo ou pré-gravada.

Neste contexto, muitos sistemas de alinhamento áudio-partitura primeiro alinham o áudio da performance, ao vivo, com o áudio sintetizado pela partitura, que é gerada a partir das notas e ritmos especificados. Esta abordagem de alinhamento áudio-áudio serve como um passo intermediário para a realização do alinhamento final com a partitura (HENKEL; WIDMER, 2021).

O primeiro passo para a realização do alinhamento é a extração de características relevantes do áudio. A precisão do alinhamento depende da qualidade das características extraídas, que servem como dados de entrada para os modelos de aprendizado de máquina.

2.2 Extração de Características de Áudio

A extração de características de áudio é uma etapa importante no processamento e análise de sinais musicais. Ela envolve transformar um sinal bruto em representações que podem ser usadas em algoritmos de aprendizado de máquina e outras técnicas analíticas. O objetivo é capturar as informações essenciais do sinal que descrevem aspectos como tonalidade, ritmo, timbre, e intensidade.

A extração de características geralmente começa com a divisão do áudio em pequenas janelas temporais, ou *frames*, para que cada janela possa ser analisada independentemente. As técnicas mais comuns de análise de áudio incluem (SHARMA; UMAPATHY; KRISHNAN, 2020):

- Espectrogramas: Representam a distribuição de frequências ao longo do tempo. Ao aplicar a Transformada de Fourier a cada janela, obtemos um mapa visual de frequências, que revela padrões fundamentais de intensidade e timbre no áudio.

- Coeficientes Cepstrais de Freqüência Mel: Esses coeficientes representam a forma da curva espectral e são uma característica padrão em reconhecimento de fala e outras tarefas de processamento de áudio.

- Características de Croma: Características que mapeiam as frequências para suas classes cromáticas correspondentes (as 12 notas da escala ocidental), permitindo a identificação de acordes e outros aspectos harmônicos.

Além dessas, outras características como *zero-crossing rate*, *spectral centroid*, e *spectral roll-off* fornecem *insights* sobre o timbre, a articulação e outros aspectos rítmicos do áudio (SHARMA; UMAPATHY; KRISHNAN, 2020; TZANETAKIS; COOK, 2002).

A extração dessas características permite que modelos de aprendizado de máquina processem padrões musicais em um formato numérico útil, e muitas vezes mais simples. Esses modelos podem ser alimentados a redes neurais ou processados manualmente para o uso em técnicas de análise de sequência, como DTW e Cadeias Ocultas de Markov, facilitando a compreensão e a comparação de diferentes sinais de áudio.

Compreendendo algumas das técnicas de extração de características de áudio, vamos nos aprofundar na STFT. Esta abordagem é utilizada para obter representações espectrais do áudio, essenciais para o processamento e análise subsequentes.

2.2.1 *Transformada de Fourier de Tempo Curto (*Short Time Fourier Transform*)*

A STFT, é uma técnica de extração de características de áudio comum, se enquadrando na categoria de espectrograma. Ela transforma o sinal de áudio de tempo discreto em um domínio de frequência, permitindo a análise de padrões e características musicais (TZANETAKIS; COOK, 2002).

Para aplicar a STFT, o sinal de áudio é primeiro segmentado em janelas ou *frames*, tipicamente utilizando uma função de janela, como Hamming, para minimizar os efeitos do vazamento espectral. Cada segmento é então transformado de seu domínio de tempo original para um domínio de frequência usando, a Transformada de Fourier (GRÖCHENIG, 2013). O resultado é um espectrograma que mostra a intensidade de várias frequências ao longo do tempo, apresentado na Figura 1.

A resolução fornecida pela STFT é afetada pelo tamanho da janela utilizada. Janelas maiores proporcionam uma melhor resolução de frequência, mas pior resolução de tempo e vice-versa. Esta relação de compromisso entre resolução de tempo e frequência é um desafio inerente ao uso da STFT (GRÖCHENIG, 2013) .

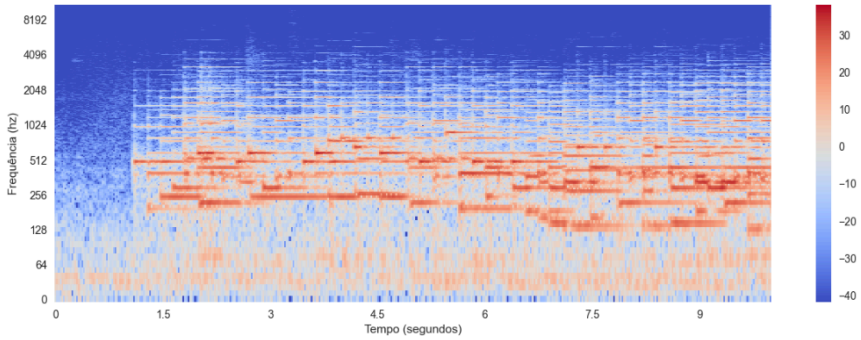

Figura 1: O espectrograma mostra a distribuição de frequências ao longo do tempo de uma gravação de piano, obtido através da STFT. As cores representam a intensidade das frequências, com tons mais quentes indicando maior intensidade.

Fonte: Autoria Própria.

A principal vantagem da STFT é a sua capacidade de fornecer uma análise localizada de frequência e tempo simultaneamente. Isso é especialmente útil em cenários onde o sinal tem padrões de frequência complexos. Este método permite mais facilidade e precisão na decomposição e identificação de padrões no áudio por algoritmos de processamento (GHORAANI; KRISHNAN, 2011).

Após explorarmos a extração de características de áudio, discutiremos as SNNs. Esses modelos são especializados em aprender a comparar entradas, sendo utilizados para analisar a similaridade entre *frames* de áudio extraídos, um passo crítico no alinhamento.

2.3 Redes Neurais Siamesas (*Siamese Neural Networks*)

As SNNs são redes neurais especializadas em aprender a comparar duas entradas e determinar a semelhança entre elas. Consistem em duas sub-redes idênticas, cada um processando uma entrada, como apresentado na Figura 2. Cada ramo opera como uma rede neural convencional, e ambos compartilham os mesmos pesos, assegurando a extração consistente de características semelhantes de ambas as entradas (CHICCO, 2021).

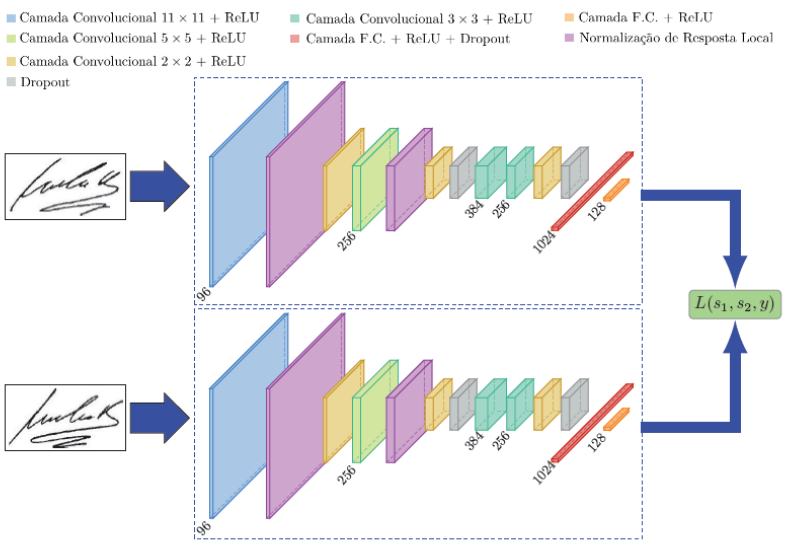

Figura 2: A Rede Neural Siamesa é composta por duas sub-redes idênticas que compartilham pesos. Cada sub-rede inclui camadas convolucionais (11x11, 5x5, 3x3, 2x2) com funções de ativação ReLU, normalização de resposta local e dropout, seguidas por camadas pooling e densas. A função de perda contrastiva $L(s_1, s_2, y)$ compara as saídas das sub-redes para determinar a similaridade entre as entradas. Esta estrutura é usada para tarefas de comparação de pares, como verificação de assinaturas.

Fonte: (DEY et al., 2017)

Essas redes são comumente utilizadas em aplicações que requerem a verificação de semelhança ou compatibilidade, como na verificação de assinaturas (SHARMA et al., 2022), reconhecimento facial (CUI et al., 2019), e biometria de voz (KAAVYA SRISKANDARAJA, 2018; MARLON ALMSTRÖM, THI THU HÒA TRÂN, 2022), muitas apresentando performance comparável ou melhor que o estado da arte em suas respectivas áreas.

A principal métrica de aprendizado em SNNs é a função de Perda Contrastiva (do inglês, *Contrastive Loss*), que mede a distância entre as representaçes das entradas. Essa função é projetada para diminuir a distância entre entradas similares e aumentar entre aquelas que são diferentes (CHOPRA; HADSELL; LECUN, 2005).

A função de Perda Contrastiva é fundamental nas SNNs, onde o principal objetivo é aprender distinções claras entre pares de entradas baseadas em sua semelhança. Essa função considera a distância euclidiana entre as saídas dos dois ramos da rede, que processam separadamente cada uma das entradas do par (CHOPRA; HADSELL; LECUN, 2005).

Formalmente, se x_1 e x_2 são duas entradas com um rótulo binário y, onde $y = 1$ indica que as entradas são semelhantes e $y = 0$ indica que são diferentes, a função de Perda Contrastiva L pode ser expressa da seguinte forma (UTKIN; KOVALEV; KASIMOV, 2019):

$$L(x_1, x_2, y) = y(D_W)^2 + (1 - y)\ \max(0, m - D_W)^2 \tag{1}$$

Onde D_W é a distância euclidiana entre as representações aprendidas pela sub-rede G_W de x_1 e x_2:

$$D_W = \sqrt{(G_W(x_1) - G_W(x_2))^2} \tag{2}$$

O hiperparâmetro m é uma margem de dissimilaridade definida que ajuda a modelar quão separadas as representações de pares de classes diferentes devem estar. O primeiro termo da função penaliza pares similares ($y = 1$), que estão distantes um do outro, enquanto o segundo termo penaliza pares diferentes ($y = 0$), que estão mais próximos do que a margem m (CHOPRA; HADSELL; LECUN, 2005).

Este mecanismo permite que a rede aprenda a embutir exemplos semelhantes próximos um do outro no espaço de características, enquanto exemplos de diferentes classes são empurrados para ficar mais distantes, ao menos pela distância da margem m (AGRAWAL; DIXON, 2020). A escolha de m pode variar, com o valor $m = 1$ sendo comumente utilizado, e a formulação exata da função de perda podem variar dependendo das especificidades do problema e do domínio de aplicação (KAAVYA SRISKANDARAJA, 2018).

No processamento de áudio, as SNNs têm aplicações como a verificação de locutor e, de particular interesse para este estudo, o alinhamento de áudio e partitura. Elas são empregadas para identificar alinhamentos harmônicos entre diferentes faixas de áudio, uma tarefa crucial de alinhamento áudio-partitura. Esta aplicação tira proveito da capacidade das redes de comparar uma ampla variedade de características acústicas, o que é fundamental para tratar a complexidade dos sinais de áudio.

Após a implementação das SNNs, precisamos de um método que utiliza as comparações de similaridade para alinhar sequências temporais. O método discutido a seguir é o DTW. Este algoritmo é vital para ajustar variações de velocidade entre gravações de áudio e gerar o alinhamento.

2.4 *Dynamic Time Warping*

O DTW é uma técnica usada para medir a similaridade entre duas sequências temporais que podem variar em velocidade. Por exemplo, DTW pode ser usado para comparar

dois sinais de áudio onde as velocidades de fala ou as durações podem diferir (MÜLLER, 2007).

Essa técnica identifica a melhor correspondência entre duas sequências, minimizando as diferenças locais entre elas (MÜLLER, 2007). O algoritmo alinha as sequências em pontos específicos, onde a distância euclidiana entre os pontos das sequências é minimizada, permitindo que múltiplos pontos de uma sequência sejam alinhados com um único ponto da outra, como apresentado na Figura 3.

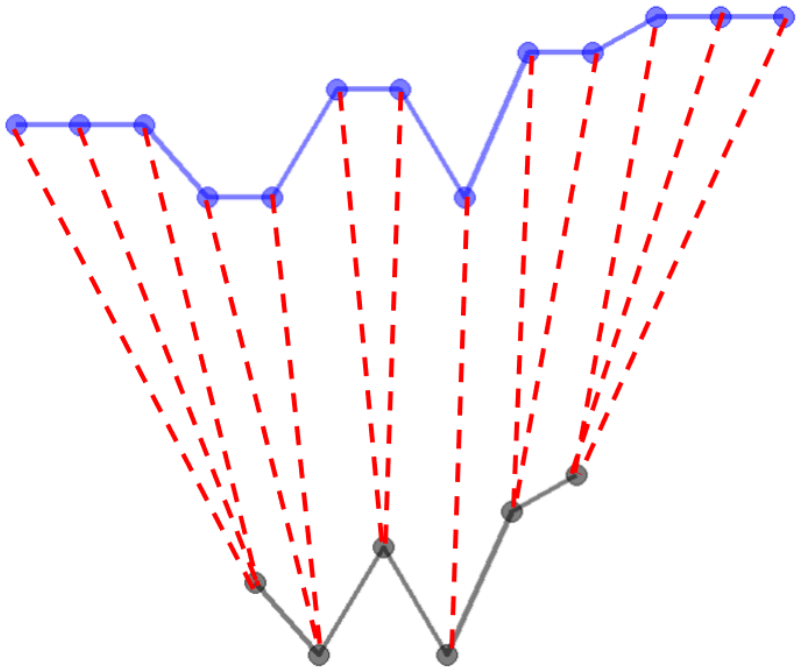

Figura 3: O gráfico ilustra o alinhamento de duas sequências usando o algoritmo DTW. As linhas azuis e pretas representam as duas sequências de dados, enquanto as linhas tracejadas vermelhas mostram a correspondência entre os pontos das sequências, minimizando a distância local.

Fonte: d3view

Formalmente, se temos duas sequências, $X = x_1, x_2, ..., x_n$ e $Y = y_1, y_2, ..., y_m$, o DTW constrói uma matriz de distâncias, onde cada elemento $D(i, j)$ representa o custo mínimo para alinhar os subconjuntos $x_1, ..., x_i$ e $y_1, ..., y_j$. O valor de $D(i, j)$ é calculado como (MÜLLER, 2007):

$$D(i,j) = d\big(x_i, y_j\big) + m \in (D(i-1,j), D(i,j-1), D(i-1,j-1)) \tag{3}$$

Onde $d\big(x_i, y_j\big)$ é a distância entre os elementos x_i e y_j das sequências, e os termos $D(i-1,j)$, $D(i,j-1)$ e $D(i-1,j-1)$ representam os custos mínimos para alinhar as

sequências até os elementos $i-1$ e j, i e $j-1$, $i-1$ e $j-1$, respectivamente (MÜLLER, 2007).

A matriz de distâncias gerada pelo DTW armazena os custos de alinhamento. Cada célula $D(i,j)$ na matriz representa o custo acumulado para alinhar os elementos até i, da primeira sequência, com os elementos até j, da segunda sequência. O caminho através desta matriz que minimiza o custo total de $D(1,1)$ até $D(n,m)$ é conhecido como o caminho ótimo de alinhamento. Este caminho pode ser traçado retroativamente, a partir de $D(n,m)$ até $D(1,1)$, escolhendo em cada passo o predecessor que contribuiu com o menor custo acumulado. A Figura 4 ilustra este caminho dentro da matriz de distâncias (MÜLLER, 2007).

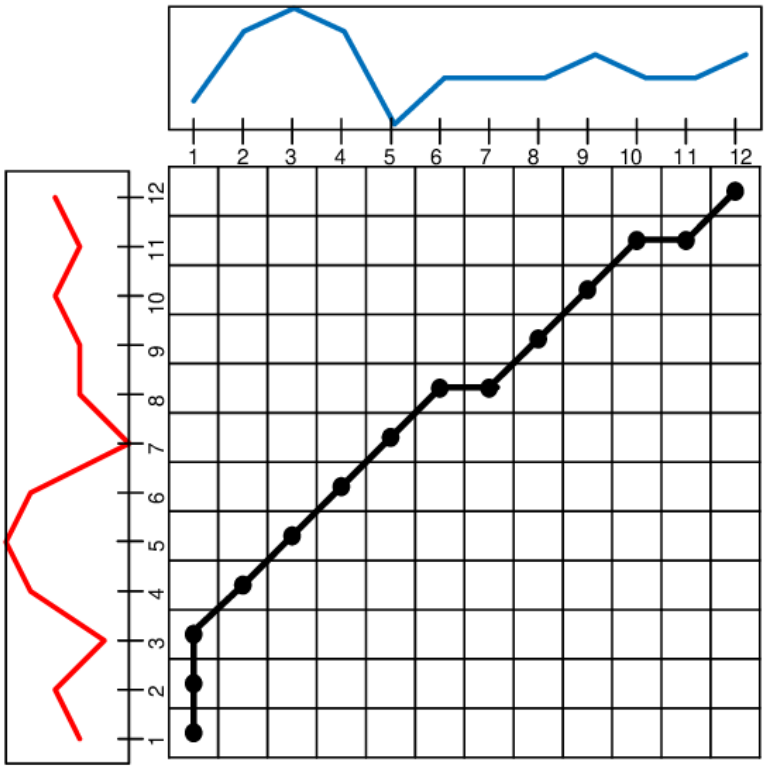

Figura 4: O gráfico apresenta o caminho ótimo de alinhamento entre duas sequências, utilizando uma matriz de distâncias. As sequências são mostradas nas margens superior (em azul) e esquerda (em vermelho). Os pontos pretos no interior da matriz representam o alinhamento ótimo calculado pelo algoritmo DTW, minimizando a distância local entre as duas sequências.

Fonte: (GOLDER et al., 2019)

DTW é amplamente utilizado em várias aplicações de áudio, incluindo o reconhecimento de fala (PERMANASARI; HARAHAP; ALI, 2019), onde é essencial alinhar e comparar sinais vocais, que podem ter sido pronunciados em ritmos diferentes. No contexto

de alinhamento áudio-partitura, DTW permite alinhar e sincronizar faixas musicais com variações de tempo, facilitando a comparação precisa de características musicais, mesmo quando apresentam variações de tempo e estilo.

Este método é particularmente poderoso em situações onde os métodos tradicionais de comparação de sequências falham em capturar a essência das diferenças temporais, como é frequentemente o caso em aplicações musicais. A capacidade de DTW de adaptar-se a variações dinâmicas faz dele uma ferramenta importante para análises precisas em processamento de áudio.

Finalmente, com o método de alinhamento estabelecido, precisamos de métricas para avaliar a eficácia dos modelos. A Seção 2.5 tratará das métricas utilizadas para medir a precisão do alinhamento.

2.5 Métricas de Avaliação

No contexto de alinhamento áudio-partitura, a métrica de distância temporal é fundamental para avaliar o quão precisamente os modelos conseguem sincronizar eventos musicais. As métricas mais relevantes incluem (CONT et al., 2007):

- Erro Médio Absoluto (do inglês, *Mean absolute error* - MAE): Mede a diferença média entre as previsões e os tempos reais dos eventos. Esta métrica é essencial para determinar se os modelos conseguem identificar corretamente os pontos de marcação na linha do tempo musical.

- Distribuição Percentual do Erro Temporal: Esta métrica divide o erro temporal em faixas, como <25ms, <50ms, <100ms, e <200ms. Cada faixa mostra a porcentagem de marcações que se encontram dentro dessa janela temporal, permitindo comparar a precisão do modelo em diferentes margens de erro.

- Taxa de Desalinhamento: Mede a proporção de eventos que estão significativamente desalinhados entre a marcação prevista e o tempo real. A faixa de aceitação da métrica varia de acordo com a aplicação.

- Desvio Padrão do Erro Temporal: Mede a dispersão dos erros temporais em relação à média. Esta métrica é útil para avaliar a consistência do modelo em diferentes partes do áudio.

As métricas tradicionais de aprendizado de máquina, como Acurácia, Precisão, Sensibilidade e Medida F1, também fornecem *insights* importantes sobre o desempenho geral do modelo. No entanto, métricas específicas de tempo são especialmente importantes para entender a eficácia de modelos em tarefas de alinhamento musical, onde precisão temporal é um ponto crítico (CONT et al., 2007).

3 TRABALHOS RELACIONADOS

Este capítulo está organizado em duas seções, cada uma abordando aspectos importantes para este trabalho na área de alinhamento áudio-partitura.

Primeiramente, apresentamos os avanços e métodos no alinhamento áudio-partitura, focando em técnicas de alinhamento utilizando redes neurais. Nesta seção, revisamos trabalhos que utilizam diferentes abordagens de redes neurais.

Em seguida, exploramos a diversidade de dados e sua importância no treinamento de modelos de aprendizado de máquina. Analisamos estudos que discutem a influência da variedade de dados na generalização e precisão dos modelos, destacando metodologias que promovem a diversidade de dados para melhorar o desempenho.

3.1 Alinhamento Áudio-Partitura

No artigo AGRAWAL (2022), o autor desenvolve uma abordagem inovadora para o alinhamento entre performances musicais e partituras, avançando progressivamente de modelos de Redes Neurais Convolucionais (do inglês, *Convolutional Neural Networks* - CNNs) tradicionais para Redes Siamesas e modelos ponta-a-ponta. O autor detalha como a arquitetura neural pode ser refinada para capturar relações temporais complexas e melhorar a precisão do alinhamento. O trabalho explora vários métodos, culminando em uma arquitetura que utiliza CNNs dilatadas e modelos com auto-atenção para construir uma solução ponta-a-ponta, que aprende diretamente a alinhar áudio e partitura.

AGRAWAL (2022) apresenta uma abordagem de aprendizado de métricas usando SNNs para capturar e representar a similaridade espectral entre gravações de áudio e partituras. A arquitetura é composta por duas redes CNN idênticas, que processam separadamente as janelas de áudio das partituras e performances. As redes são treinadas para maximizar a similaridade entre *frames* alinhados e minimizar entre *frames* desalinhados. Estes dados são processados usando o algoritmo DTW durante a etapa de teste. O modelo é testado em vários conjuntos de dados, como 'MAPS', 'RWC', 'SCREAM-MAC-EMT', e 'Traditional Flute Dataset', cobrindo diferentes instrumentos e cenários acústicos. A entrada do modelo é baseada em espectrogramas obtidos através de STFT, com áudio sintetizado por MIDI via o programa *FluidSynth*.

Os resultados mostram que as representações aprendidas melhoram a precisão de alinhamento em comparação com abordagens tradicionais. O modelo adapta-se bem a diferentes instrumentos e ambientes acústicos, mesmo em condições de escassez de dados. A abordagem adotada é particularmente relevante para este estudo, pois aborda o alinhamento áudio-partitura com redes siamesas, apesar de não realizar uma análise profunda da variabilidade instrumental. Nosso trabalho busca expandir esses conceitos e explorar mais detalhadamente o impacto de diversidade instrumental na performance do modelo.

Os trabalhos AGRAWAL; WOLFF; DIXON (2021) e LEPISTÖ (2023) também servem como referência, oferecendo variações na técnica de alinhamento. O primeiro apresenta uma arquitetura neural baseada em convolução e *self-attention*, que aplica um modelo *encoder-decoder* para prever caminhos de alinhamento entre performances e partituras. Eles usam uma função de perda customizada baseada em *soft-DTW* e auto-atenção para melhorar a precisão do alinhamento em condições com variações estruturais. Já o segundo, de Lepistö, investiga diferentes métodos siameses de medição de similaridade de áudio, utilizando arquiteturas que empregam similaridade de coseno, concatenação de vetores, e distância L1 ponderada. Os resultados demonstram o potencial desses modelos para generalizar sobre classes de áudio distintas e melhorar a precisão do alinhamento.

3.2 Diversidade de Dados

A importância da diversidade de dados e modelos é fundamental no aprendizado de máquina. No artigo GONG; ZHONG; HU (2019) apresentam uma análise teórica abrangente, explorando a fundo técnicas de diversificação, como *Determinantal Point Processes* (DPPs) e *Poisson Disk Sampling* (PDS), além de métodos de aprendizado ativo. Eles também discutem os possíveis efeitos da diversidade no desempenho dos modelos, oferecendo insights sobre a sua relevância e como ela pode ser maximizada.

Em contraste, AQUINO et al. (2017), oferece uma análise prática dos impactos das estratégias de aumento de dados e volume em CNNs. O estudo evidencia como o aumento equilibrado das classes pode melhorar significativamente a precisão na classificação de atributos biométricos, especialmente para conjuntos de dados pequenos e desequilibrados.

Já no campo do alinhamento áudio-partitura, MORSI; SERRA (2022), apresenta uma representação direta ao consolidar conjuntos de dados como *Aligned Scores and Per-*

formances (ASAP). Eles propõem metodologias como interpolação entre anotações de batida para melhorar a precisão do alinhamento, e destacam a importância de diversidade nos dados, principalmente no cenário atual em onde são limitadas as databases de alta qualidade com variedade interna.

Apesar das pesquisas sólidas sobre a importância da diversidade de dados e a ênfase em suas aplicações, há pouca exploração prática do impacto da falta de diversidade nos dados na área de alinhamento de áudio-partitura. Esta lacuna é a que este estudo busca preencher, examinando especificamente o efeito da variabilidade de instrumentos na performance de modelos para alinhamento musical.

4 METODOLOGIA

Este capítulo descreve a metodologia utilizada para conduzir esta pesquisa, organizada em quatro seções principais.

Inicialmente, apresentamos as bases de dados utilizadas no estudo, explicando os critérios de seleção e as características específicas de cada conjunto de dados instrumental.

Em seguida, detalhamos o processo de extração de dados, que envolve a conversão e normalização das anotações para um formato padronizado, garantindo consistência no treinamento dos modelos. Discutimos também o uso de ferramentas e bibliotecas específicas para a síntese de áudio.

A terceira seção aborda o pré-processamento, onde descrevemos as etapas de transformação dos arquivos de áudio, segmentação, redimensionamento, e pareamento de *frames*. Este processo é importante para assegurar que os dados de entrada dos modelos sejam padronizados e comparáveis.

Finalmente, apresentamos o modelo utilizado na pesquisa, uma SNN, e o procedimento de treinamento. Discutimos as diferentes variantes do modelo, os parâmetros de treinamento, e o ambiente computacional utilizado. Esta seção inclui também a descrição das métricas de avaliação empregadas para medir a precisão dos modelos treinados.

4.1 Bases de Dados

Para a realização deste estudo, foi essencial selecionar bases de dados que oferecessem um volume significativo de dados, uma composição instrumental isolada, e anotações alinhadas aos áudios. A escolha de conjuntos de dados que representam instrumentos solo permite a análise da influência da variedade instrumental no desempenho dos modelos de alinhamento áudio-partitura. A seguir, descrevemos cada um dos conjuntos de dados utilizados.

4.1.1 *Groove (Bateria)*

O Groove MIDI Dataset[1] compreende gravações em MIDI de performances de bateria, executadas por profissionais em um kit de bateria eletrônica Roland TD-11. Este

[1] https://magenta.tensorflow.org/datasets/groove

conjunto inclui aproximadamente 13,6 horas de gravações, abrangendo 1.150 arquivos MIDI que representam mais de 22.000 compassos. Por conter um volume substancial de dados, anotações de metrônomo, gênero musical, e identificação do baterista, esta base é comumente utilizada na área de análise musical, facilitando estudos detalhados sobre variação de estilo e técnica.

4.1.2 *GuitarSet (Violão)*

GuitarSet[2] fornece gravações de violão acústico capturados com captação hexafônica, permitindo a separação e análise de dados de cada corda, individualmente. Composto por 360 trechos musicais, cada um com anotações alinhadas de contornos de pitch, posições de cordas, acordes, entre outros. Esta base de dados é ideal para explorar a precisão em técnicas de alinhamento, análise musical, e a presença de acordes e notas o torna particularmente útil para análise de partituras.

4.1.3 *MAESTRO (Piano)*

MAESTRO[3] consiste em cerca de 200 horas de gravações de áudio e MIDI, capturadas durante competições internacionais de piano. As gravações foram realizadas em pianos Disklavier da Yamaha, que integram um sistema de captura MIDI de alta precisão. Este conjunto é destacado pela sua qualidade excepcional de áudio e dados MIDI[4], alto volume de dados[5], e anotações detalhadas sobre cada peça musical.

4.1.4 *Traditional Flute (Flauta)*

O conjunto de dados de Traditional Flute[6] inclui gravações de flauta solo com anotações correspondentes em notação simbólica. Composto por 30 fragmentos de áudio de obras clássicas, e anotações detalhadas sobre as técnicas tradicionais de execução, cobrindo um total de 2.245 eventos musicais.

[2]https://guitarset.weebly.com/

[3]https://magenta.tensorflow.org/datasets/maestro

[4]Os dados MIDI gravados têm fidelidade suficiente para permitir que a etapa de audição da competição seja julgada remotamente, ouvindo as performances dos concorrentes reproduzidas à distância em outro instrumento Disklavier.

[5]A base possui mais de 100GB de arquivos de áudio alinhados.

[6]https://www.kaggle.com/datasets/jbraga/traditional-flute-dataset

Embora o número de fragmentos e o volume de dados deste conjunto seja consideravelmente menor em comparação com as demais bases de dados utilizadas neste estudo, sua inclusão foi uma decisão metodológica para proporcionar maior variedade instrumental aos experimentos. Apesar de o alinhamento manual das anotações potencialmente introduzir inconsistências, esse trade-off foi considerado necessário para investigar a influência da diversidade instrumental no desempenho dos modelos

4.2 Extração de Dados

As bases descritas possuem arquivos de áudio com performances gravadas por artistas, e anotações alinhadas para cada arquivo de áudio. Porém, estas anotações não estão em formato padronizado, variando entre formatos MIDI, JAMS, e GT.

O processo de extração de dados envolve a conversão das anotações de cada conjunto de dados para um formato padronizado MIDI. Então, estes dados são sintetizados para produzir arquivos de áudio alinhados aos áudios de performance já presentes nas bases de dados.

4.2.1 *Groove e MAESTRO*

Os conjuntos de dados Groove e *MAESTRO* incluem arquivos MIDI e áudio de performances. Utilizamos o FluidSynth para gerar áudio sintetizado, a partir dos arquivos MIDI fornecidos.

4.2.2 *GuitarSet*

O *GuitarSet* possui anotações em formato JAMS, incluindo uma diversidade de dados, incluindo notas MIDI, contorno de tom, acorde, e *tempo*. Os arquivos JAMS foram convertidos para MIDI usando scripts disponíveis no repositório do projeto[7], utilizando a biblioteca `pretty_midi`. Após a conversão, os arquivos MIDI são processados com FluidSynth para produzir áudio sintetizado correspondente.

4.2.3 *Traditional Flute*

[7]https://github.com/marl/GuitarSet/blob/master/visualize/interpreter.py

Na base de dados *Traditional Flute*, as anotações são salvas no formato customizado '.gt'. Este formato expõe, em CSV, colunas descrevendo o início da nota, a frequência, e a duração. Linhas no arquivo representam notas individuais presentes no áudio da performance.

Extraímos as anotações dos arquivos em formato '.gt', e as convertemos MIDI usando a biblioteca `pretty_midi`. Em seguida, usamos FluidSynth para sintetizar os arquivos de áudio a partir dos MIDI criados.

4.3 Pré-Processamento

Os áudios foram pré-processados para garantir consistência entre os dados na análise subsequente. Este processo é crucial para assegurar que os modelos de aprendizado de máquina recebam dados de entrada padronizados e comparáveis.

4.3.1 *Transformação*

Inicialmente, cada arquivo de áudio foi transformado do domínio do tempo para o domínio da frequência, utilizando a STFT com a biblioteca `librosa`. Aplicamos uma função de janela de Hamming sobre segmentos de 1.024 amostras com *hop length* de 512 amostras.

4.3.2 *Segmentação e Redimensionamento*

Após a obtenção dos espectrogramas através da STFT, estes foram divididos em janelas menores, referidas como *frames*. Cada *frame* do espectrograma foi cortado no comprimento para um tamanho uniforme de 128 pixels e redimensionado na altura para 128 pixels, utilizando interpolação bilinear. Este processo garante que cada *frame* tenha dimensões consistentes de 128x128 pixels, normalizando a entrada dos modelos.

Os áudios ao vivo e sintetizados possuem leves diferenças de comprimento. Logo, mantemos o número menor de *frames* entre os dois para garantir a consistência no alinhamento, equalizando a quantidade de *frames* em ambos.

4.3.3 *Pareamento de* Frames

Frames dos áudios são pareados por índice. Formalmente, se L é o *array* de *frames* do áudio gravado, e S o *array* de *frames* do áudio sintetizado, para cada índice $i \in \#L$, um índice $j \neq i$ é selecionado uniformemente em $[0, \#L)$. A partir dos índices i e j, são gerados um par similar $\langle L_i, S_i \rangle$ com rótulo 1, e um par dissimilar $\langle L_i, S_j \rangle$ com rótulo 0.

Para cada base de dados, pares são acumulados até atingir 40.000 exemplos[8] em sua respectiva base de dados, formando 4 bases. Adicionalmente, uma base de dados mista contendo 10.000 pares de *frames* de cada base é gerada. Ao final do pré-processamento, possuímos 5 bancos de dados com 40.000 exemplos cada.

4.4 Modelo

O modelo utilizado nesta pesquisa é uma SNN, baseado nas arquiteturas descritas por AGRAWAL; DIXON (2020) e AGRAWAL (2022). O modelo foi implementado utilizando a biblioteca *PyTorch*.

A rede consiste em quatro camadas convolucionais, cada uma utilizando a função de ativação ReLU (Unidade Linear Retificada). As três primeiras camadas são seguidas por camadas de *max pooling* para agregação de características relevantes, enquanto a última camada convolucional é seguida por uma camada *flatten*, que lineariza os dados para a mensuração de distâncias na etapa final do processamento.

Para investigar o impacto do tamanho da rede nas capacidades de aprendizado, foram exploradas três variantes do modelo, diferenciadas pelo número de kernels em cada camada convolucional, escalonado por um fator N:

- Modelo Pequeno: $N = 1$
- Modelo Médio: $N = 2$
- Modelo Grande: $N = 4$

O menor é uma réplica do modelo usado em AGRAWAL; DIXON (2020), enquando os outros dois são escalonamentos dele. Cada variante visa avaliar como a capacidade de modelagem e generalização é influenciada pelo aumento da complexidade da rede. A Tabela 1 detalha a configuração do modelo generalizado.

[8] A quantidade de *frames* em cada base de dados são, aproximadamente: 280.000 Groove, 900.000 MAESTRO, 700.000 GuitarSet, 70.000 Traditional Flute. O valor de selecionado de 40.000 é o mais próximo da base de dados com a menor quantidade de *frames* (Traditional Flute), para garantir que todas as bases tenham a mesma quantidade de pares.

Tipo de Camada	Tam. Entrada	Kernels	Tam. Kernel
Convolução	$128 \times 128 \times 1$	$64 \times N$	5×5
Max-Pooling	$128 \times 128 \times 64 \times N$	1	2×2
Convolução	$64 \times 64 \times 64 \times N$	$128 \times N$	5×5
Max-Pooling	$64 \times 64 \times 128 \times N$	1	2×2
Convolução	$32 \times 32 \times 128 \times N$	$256 \times N$	3×3
Max-Pooling	$32 \times 32 \times 256 \times N$	1	2×2
Convolução	$16 \times 16 \times 256 \times N$	$512 \times N$	3×3
Flatten	$16 \times 16 \times 512 \times N$	-	-
Saída	$131072 \times N$	-	-

Tabela 1: A tabela descreve a arquitetura do modelo generalizado utilizado no estudo. As camadas incluem convolução, max-pooling, flatten e saída, com seus respectivos tamanhos de entrada, número de kernels, e tamanhos de kernels. O parâmetro N representa o fator de escalonamento, variando o número de kernels para avaliar o impacto da complexidade do modelo. Esta configuração permite analisar a performance de modelos de diferentes tamanhos (Pequeno, Médio, e Grande) nas tarefas de alinhamento.

Fonte: Autoria Própria.

4.5 Treinamento

O treinamento dos modelos foi realizado utilizando a biblioteca `pytorch`. Foram treinados 15 modelos no total, um para cada combinação de tamanho (3 tamanhos) e banco de dados (5 conjuntos).

Os dados de cada banco foram divididos em conjuntos de treino e validação, de proporções 90% e 10%, respectivamente. A configuração do treinamento incluiu uma taxa de aprendizagem de 1×10^{-6}, tamanho de lote (*batch size*) de 32, e margem contrastiva de 1.

Foi utilizado o critério de parada antecipada para prevenir sobreajuste, encerrando o treinamento caso a perda de validação não apresente melhora após três épocas consecutivas.

Os modelos foram treinados em uma GPU NVIDIA RTX 3060Ti com 8GB de memória VRAM.

5 RESULTADOS

Neste capítulo, apresentamos os resultados dos testes realizados para avaliar a performance dos modelos em diferentes cenários. A metodologia de teste incluiu três categorias principais: Intra-Instrumental, Misto e Inter-Instrumental. Cada categoria foi avaliada com base em diferentes conjuntos de dados e modelos.

Os resultados foram obtidos através matrizes de distância de 64x64, que representam as diferenças temporais entre duas sequências de áudio ao longo do tempo, calculadas pela distância par-a-par entre duas sequências. Utilizando o módulo `dtw` da biblioteca `dtaidistance`, é possível gerar um caminho ótimo para as matrizes para identificar as áreas de maior correlação temporal entre as músicas.

O processo de alinhamento envolve a identificação dos caminhos ótimos através das matrizes de distância. Estes caminhos representam a menor diferença acumulada entre os pares das sequências, essencialmente traçando um caminho ao longo do mínimo local da matriz, da quina inferior à superior, como visto na Figura 5. A linha de menor custo na matriz, indica que os trechos em os modelos indicam que as músicas estão bem alinhadas. Alinhamento perfeito é indicado por uma linha perfeitamente diagonal, e desvios à diagonal são erros.

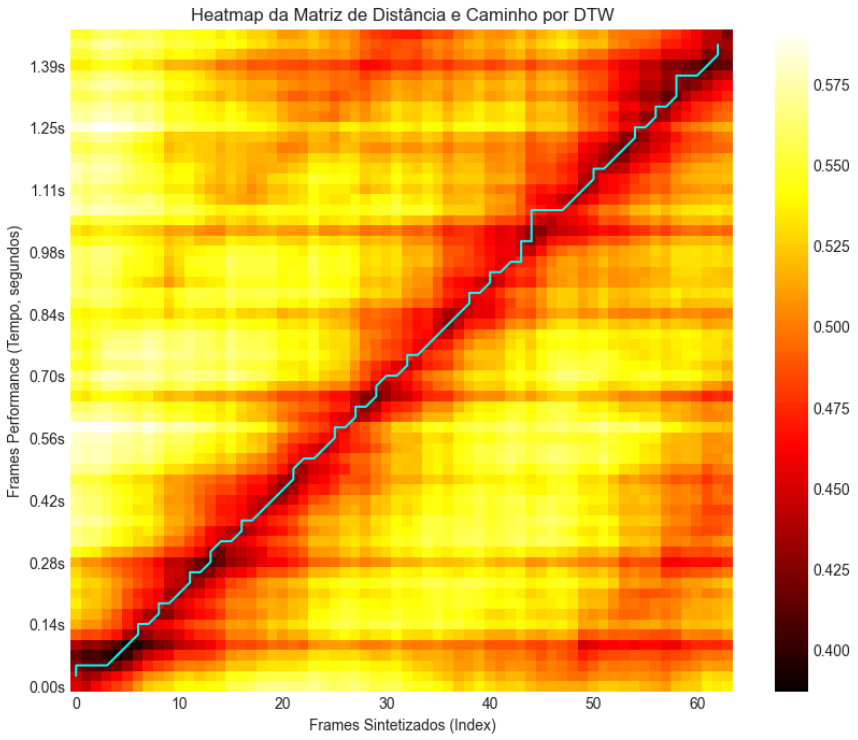

Figura 5: O mapa de calor mostra a matriz de distância calculada pelo algoritmo DTW para duas sequências de áudio. As cores indicam a magnitude das distâncias, com tons mais escuros representando menores distâncias. A linha azul sobreposta representa o caminho ótimo de alinhamento, minimizando a distância local entre as duas sequências ao longo do tempo.

Fonte: Autoria Própria.

Os resultados são apresentados em termos de distribuição percentual do erro temporal. As janelas de erro temporal são intervalos de tempo usados para categorizar a precisão do alinhamento produzido pelo modelo em diferentes margens de erro. Elas são definidas como $\leq$ 25ms, $\leq$ 50ms, $\leq$ 100ms e $\leq$ 200ms, representando os limites máximos permitidos para a diferença entre os tempos previstos e os tempos reais dos eventos musicais.

Uma janela de $\leq$ 25ms, por exemplo, indica que a previsão está a menos de 25 milissegundos de distância da anotação real, sugerindo alta precisão. Essas janelas proporcionam uma análise granular da distribuição percentual do erro, refletindo a capacidade do modelo de alinhar eventos musicais com diferentes níveis de precisão.

As Tabelas 2-20 mostram os resultados para cada combinação de tamanho (Pequeno, Médio e Grande) e banco de dados de treino do modelo (Groove, GuitarSet, MAESTRO, Traditional Flute, e Misto).

5.1 Intra-Instrumental

Nos testes Intra-Instrumental, cada modelo foi treinado e testado exclusivamente em seu próprio banco de dados, com 10.000 amostras utilizadas para o teste. Esta abordagem permite avaliar como os modelos performam ao lidar com dados de teste do mesmo instrumento.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Pequeno)	83,6	88,9	92,1	94,9
Groove (Médio)	*80,8*	*88,0*	*91,8*	*94,4*
Groove (Grande)	**89,9**	**93,3**	**95,7**	**98,1**

Tabela 2: Erro Intra-Instrumental | Groove.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
GuitarSet (Pequeno)	76,8	80,2	85,0	*89,7*
GuitarSet (Médio)	*75,6*	*79,1*	*84,4*	89,9
GuitarSet (Grande)	**79,3**	**84,4**	**89,5**	**91,1**

Tabela 3: Erro Intra-Instrumental | GuitarSet.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
MAESTRO (Pequeno)	**67,9**	**83,2**	**91,9**	**94,1**
MAESTRO (Médio)	67,1	82,6	*89,0*	*91,3*
MAESTRO (Grande)	*67,0*	*80,7*	90,8	92,9

Tabela 4: Erro Intra-Instrumental | MAESTRO.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Traditional Flute (Pequeno)	52,0	55,7	*62,8*	*73,6*
Traditional Flute (Médio)	*50,4*	*54,9*	62,9	75,5
Traditional Flute (Grande)	**56,0**	**59,3**	**66,8**	**77,9**

Tabela 5: Erro Intra-Instrumental | Traditional Flute.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Os testes intra-instrumentais demonstram que, em geral, os modelos maiores (Grande) tendem a ter um desempenho superior em termos de precisão de alinhamento, especialmente em margens de erro menores. O banco de dados Groove apresentou a melhor performance global, seguido por GuitarSet, MAESTRO, e Traditional Flute. Uma exceção notável foi observada no banco de dados MAESTRO, onde o modelo pequeno superou os modelos médio e grande.

Essa exceção pode ser atribuída a possíveis problemas de sobreajuste nos modelos maiores, que podem estar superajustando os dados de treinamento. Além disso, é possível que os dados do MAESTRO podem fazer com que modelos menores capturem melhor padrões gerais, enquanto modelos maiores se percam em detalhes menos relevantes. No

entanto, é importante notar que não há dados suficientes para chegar a uma conclusão definitiva sobre essa anomalia.

5.2 Misto

Nos testes com o banco de dados Misto, os modelos foram testados com amostras de todos os instrumentos, sendo 2.500 de cada (10.000 no total). Esta configuração visa avaliar a capacidade dos modelos de generalizar quando expostos uma variedade de instrumentos.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Pequeno)	95,8	96,1	96,4	97,0
GuitarSet (Pequeno)	*67,0*	*70,4*	*76,2*	*84,0*
MAESTRO (Pequeno)	**97,6**	**97,8**	**97,9**	**98,0**
Traditional Flute (Pequeno)	97,3	97,6	97,7	97,9
Misto (Pequeno)	84,6	85,7	87,4	89,8

Tabela 6: Erro em Base de Dados Mista | Modelo Pequeno.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Médio)	94,9	95,2	95,8	96,3
GuitarSet (Médio)	*53,8*	*57,2*	*62,6*	*70,5*
MAESTRO (Médio)	97,5	97,6	97,7	97,8
Traditional Flute (Médio)	**97,6**	**98,0**	**98,1**	**98,2**
Misto (Médio)	80,7	82,0	84,1	87,8

Tabela 7: Erro em Base de Dados Mista | Modelo Médio.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Grande)	95,1	95,6	96,1	96,8
GuitarSet (Grande)	*74,4*	*76,6*	*80,0*	*85,4*
MAESTRO (Grande)	**97,2**	**97,4**	**97,5**	**97,7**
Traditional Flute (Grande)	96,3	96,9	97,3	97,6
Misto (Grande)	85,9	87,1	88,6	91,1

Tabela 8: Erro em Base de Dados Mista / Modelo Grande.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Em geral, os modelos menores (Pequeno) e maiores (Grande) tiveram desempenho superior em termos de precisão de alinhamento. Notavelmente, os modelos treinados nos bancos de dados MAESTRO, Traditional Flute, e Groove apresentaram alta precisão em todos os modelos.

O modelo misto por outro lado apresentou um desempenho inferior em todos os casos, com erros mais altos do que os modelos individuais, com excessão do GuitarSet. Estes resultados indicam que o modelo misto não generaliza bem, indicando a possibilidade que a diluição dos dados de treinamento podem estar afetando o desempenho.

5.3 Inter-Instrumental

Nos testes Inter-Instrumental, os modelos foram testados em bancos de dados que diferem do bancos de dados de treino, cada um com 10.000 amostras. Este método permite avaliar a capacidade de generalização dos modelos, quando expostos a instrumentos específicos que não foram apresentados para a fase de treinamento.

5.3.1 *Groove*

Os modelos nesta seção foram testados no bancos de dados Groove e treinados em GuitarSet, MAESTRO, Traditional Flute, e Misto.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
GuitarSet (Pequeno)	**99,1**	**99,3**	**99,4**	99,4
MAESTRO (Pequeno)	98,7	98,9	*99,0*	99,2
Traditional Flute (Pequeno)	98,9	99,0	*99,0*	*99,0*
Misto (Pequeno)	*97,0*	*98,8*	99,3	**99,6**

Tabela 9: Erro Inter-Instrumental / Groove, Modelo Pequeno.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
GuitarSet (Médio)	**99,2**	**99,3**	99,3	99,3
MAESTRO (Médio)	98,2	*98,5*	*98,7*	*98,8*
Traditional Flute (Médio)	99,0	99,1	99,1	99,2
Misto (Médio)	*97,1*	98,7	**99,4**	**99,7**

Tabela 10: Erro Inter-Instrumental | Groove, Modelo Médio.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
GuitarSet (Grande)	**98,5**	**99,1**	**99,3**	**99,3**
MAESTRO (Grande)	98,2	98,5	98,7	98,8
Traditional Flute (Grande)	98,5	98,7	98,8	98,9
Misto (Grande)	*91,2*	*95,8*	*97,7*	*98,5*

Tabela 11: Erro Inter-Instrumental | Groove, Modelo Grande.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Os testes inter-instrumentais com o banco de dados Groove mostram que os modelos, quando treinados em outros conjuntos de dados (GuitarSet, MAESTRO, Traditional Flute e Misto), ainda conseguem manter uma alta precisão ao serem testados no Groove. A precisão dos modelos variou de 91,2% a 99,7% em diferentes margens de erro, indicando uma forte capacidade de adaptação às características rítmicas e padrões do Groove.

Essa alta precisão pode ser atribuída a vários fatores. Primeiro, as características rítmicas distintivas e menos complexas do Groove, que é uma base de dados de bateria, podem ser mais fáceis de generalizar pelos modelos, mesmo quando treinados em dados de outros instrumentos. Segundo, a consistência e a qualidade das anotações no Groove podem contribuir para uma melhor performance durante o teste. A similaridade entre os padrões temporais da percussão e de outros instrumentos também pode facilitar a transferência de aprendizado. No entanto, essas são apenas hipóteses, e não há dados suficientes para conclusões definitivas sobre o motivo exato dessa alta precisão nos testes inter-instrumentais com o Groove.

5.3.2 *GuitarSet*

Os modelos nesta seção foram testados no bancos de dados GuitarSet e treinados em Groove, MAESTRO, Traditional Flute, e Misto.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Pequeno)	**84,2**	**88,1**	**92,7**	95,1
MAESTRO (Pequeno)	77,7	84,3	90,3	93,3
Traditional Flute (Pequeno)	76,3	84,3	**92,7**	**95,2**
Misto (Pequeno)	*63,2*	*75,0*	*87,2*	*91,9*

Tabela 12: Erro Inter-Instrumental | GuitarSet, Modelo Pequeno.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Médio)	**80,7**	86,3	92,1	**94,4**
MAESTRO (Médio)	79,7	86,1	91,7	94,1
Traditional Flute (Médio)	80,1	**87,9**	**93,0**	94,2
Misto (Médio)	*63,4*	*73,6*	*85,3*	*89,0*

Tabela 13: Erro Inter-Instrumental | GuitarSet, Modelo Médio.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Grande)	**82,0**	85,0	88,6	90,9
MAESTRO (Grande)	79,8	**86,6**	91,3	93,0
Traditional Flute (Grande)	80,1	86,2	**92,5**	**94,1**
Misto (Grande)	*63,1*	*72,0*	*83,9*	*87,7*

Tabela 14: Erro Inter-Instrumental | GuitarSet, Modelo Grande.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Nos testes inter-instrumentais com o banco de dados GuitarSet, os modelos apresentaram uma variação de precisão considerável (entre 63,1% e 95,2%) em diferentes margens de erro, indicando uma menor capacidade de adaptação, em comparação com o Groove. Além disso, os modelos treinados com dados mistos não apresentaram a melhoria de performance esperada, tendo a pior performance em todos os casos (entre 63,1% e 91.9%).

Essa divergência pode ser atribuída à complexidade das características acústicas dos dados de violão, que incluem variações de timbre e nuances de execução, tornando mais difícil a generalização para modelos treinados com dados de múltiplos instrumentos. A diversidade presente nos dados mistos pode introduzir ruídos que dificultam a extração de padrões consistentes. A especificidade dos dados de violão pode exigir um treinamento mais focado para capturar as nuances necessárias para alta precisão. No entanto, essas são apenas hipóteses, e não há dados suficientes para conclusões definitivas.

5.3.3 *MAESTRO*

Os modelos nesta seção foram testados no bancos de dados MAESTRO e treinados em Groove, GuitarSet, Traditional Flute, e Misto.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Pequeno)	**81,5**	**87,5**	**92,1**	**95,1**
GuitarSet (Pequeno)	76,1	84,8	90,7	93,5
Traditional Flute (Pequeno)	75,3	84,6	90,6	93,6
Misto (Pequeno)	*62,2*	*74,5*	*83,5*	*88,8*

Tabela 15: Erro Inter-Instrumental | MAESTRO, Modelo Pequeno.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Médio)	**80,2**	86,2	90,0	92,8
GuitarSet (Médio)	72,2	80,3	86,5	*89,9*
Traditional Flute (Médio)	79,1	**86,6**	**90,9**	**93,2**
Misto (Médio)	*65,9*	*77,1*	*86,0*	90,7

Tabela 16: Erro Inter-Instrumental | MAESTRO, Modelo Médio.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Grande)	**84,0**	**87,3**	**90,2**	**92,8**

GuitarSet (Grande)	69,2	76,6	83,8	*88,5*
Traditional Flute (Grande)	74,0	81,3	88,1	92,5
Misto (Grande)	*61,6*	*72,7*	*82,9*	88,6

Tabela 17: Erro Inter-Instrumental | MAESTRO, Modelo Grande.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Nos testes inter-instrumentais com o banco de dados MAESTRO, os modelos também exibiram significante variância (entre 61,6% e 95,1%) em diferentes margens de erro, refletindo uma capacidade de adaptação variável. Os modelos treinados com dados mistos não mostraram a melhoria de performance antecipada, tendo a pior performance em todos os casos (entre 61,6% e 90,7%).

Essa variabilidade pode ser explicada pela complexidade e diversidade das peças de piano no MAESTRO, que é uma base de dados gravada em apresentações ao vivo, incluindo variações expressivas e dinâmicas, tornando a generalização mais desafiadora para modelos treinados em múltiplos instrumentos. A riqueza acústica dos dados do MAESTRO pode introduzir variabilidade significativa, dificultando a obtenção de padrões consistentes. As nuances específicas das execuções de piano podem exigir um treinamento mais especializado para alcançar alta precisão. No entanto, essas são apenas teorias preliminares, e não há dados suficientes para conclusões definitivas.

5.3.4 *Traditional Flute*

Os modelos nesta seção foram testados no bancos de dados Traditional Flute e treinados em Groove, GuitarSet, MAESTRO, e Misto.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Pequeno)	**62,9**	**66,8**	**72,8**	**80,2**
GuitarSet (Pequeno)	53,4	57,9	64,2	*73,5*
MAESTRO (Pequeno)	58,3	63,1	69,5	76,9
Misto (Pequeno)	*47,5*	*54,7*	*63,7*	74,0

Tabela 18: Erro Inter-Instrumental | Traditional Flute, Modelo Pequeno.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Médio)	58,0	61,7	67,2	74,2
GuitarSet (Médio)	**60,4**	**65,2**	**71,7**	**79,6**
MAESTRO (Médio)	*49,6*	*54,9*	*61,6*	*70,3*
Misto (Médio)	50,0	56,1	64,2	74,4

Tabela 19: Erro Inter-Instrumental | Traditional Flute, Modelo Médio.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Modelo	Margem de Erro			
	<25ms	**<50ms**	**<100ms**	**<200ms**
Groove (Grande)	**56,6**	**60,3**	65,8	74,0
GuitarSet (Grande)	53,9	59,1	**66,9**	**76,7**
MAESTRO (Grande)	54,0	59,1	65,1	73,4
Misto (Grande)	*44,3*	*50,4*	*58,7*	*69,3*

Tabela 20: Erro Inter-Instrumental | Traditional Flute, Modelo Grande.

Os melhores resultados para cada margem de erro estão em negrito, e os piores resultados estão em itálico.

Fonte: Autoria Própria.

Nos testes inter-instrumentais com o banco de dados Traditional Flute, os modelos demonstraram variações significativas na precisão (variando de 44,3% a 80,2%) em diferentes margens de erro, indicando a menor capacidade de adaptação entre todos os conjuntos de dados testados. Além disso, os modelos treinados com dados mistos não apresentaram a melhoria de performance esperada, tendo a pior performance em todos os casos (variando entre 44,3% e 69,3%), exceto o modelo médio, que apresenta a segunda pior performance (entre 50,0% e 74,4%), a pior performance do modelo médio sendo treinado no banco de dados MAESTRO (entre 49,6% e 70,3%).

Essa baixa performance pode ser atribuída às características acústicas específicas e às nuances dos instrumentos de sopro, que são mais difíceis de generalizar para modelos treinados com dados de múltiplos instrumentos. Ademais, as execuções de flauta tradicional, com anotações produzidas manualmente, podem ter uma precisão inferior, afetando negativamente os resultados.

5.4 Visão Geral

A análise dos resultados dos testes inter-instrumentais revela tendências claras na performance dos modelos. O gráfico apresentado na Figura 6 ilustra a média de acurácia de todos os modelos em todos os bancos de dados de teste, proporcionando uma visão abrangente do desempenho relativo entre os diferentes modelos e configuraçes de treinamento.

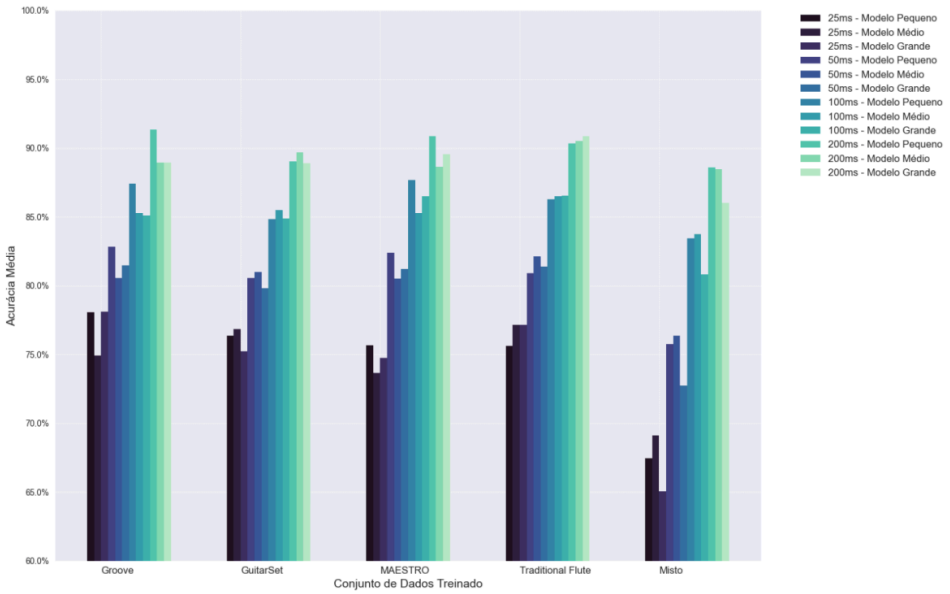

Figura 6: O gráfico apresenta a acurácia média dos modelos de diferentes tamanhos (Pequeno, Médio e Grande), em várias margens de erro (25ms, 50ms, 100ms, 200ms), treinados em diferentes bases de dados (Groove, GuitarSet, MAESTRO, Traditional Flute e Misto).

Fonte: Autoria Própria.

De forma geral, observa-se que, apesar das variações nos detalhes, as performances entre os modelos individuais são relativamente similares dentro de cada banco de dados treinado. Esta consistência sugere que a escolha do tamanho do modelo (pequeno, médio ou grande) tem um impacto limitado na precisão do alinhamento, quando se trata de dados específicos.

Por outro lado, uma tendência distinta é a inferioridade dos modelos treinados com dados mistos. Em todos os casos analisados, os modelos mistos apresentaram a pior performance em comparação com aqueles treinados exclusivamente em um tipo de instrumento. Esse resultado contraria a hipótese de que dados mistos melhorariam a generalização e a precisão dos modelos.

Finalmente, os resultados destacam a eficácia do treinamento focado. Modelos treinados em dados específicos, como Groove, GuitarSet, MAESTRO, e Traditional Flute, demonstraram performances superiores, sugerindo que a especialização ainda é a melhor abordagem para otimizar a precisão do alinhamento.

6 TRABALHOS FUTUROS

Uma das principais direções para pesquisas futuras é expandir a variedade instrumental dos experimentos. Embora tenhamos incluído quatro diferentes instrumentos, seria valioso incluir uma gama mais ampla de instrumentos, especialmente aqueles com características únicas, como metais e cordas friccionadas. Isso ajudaria a testar a generalização dos modelos para uma maior variedade instrumental. O mesmo vale para bases de dados que possuem múltiplos instrumentos em uma única performance gravada.

Para melhorar a precisão dos modelos, futuras pesquisas poderiam priorizar o uso de bases de dados com anotações automaticamente alinhadas, que tendem a oferecer maior consistência em comparação com anotações manuais. A qualidade das anotações é um fator crucial para o treinamento de modelos eficazes.

Além disso, a utilização de arquiteturas de rede neural mais avançadas, como aquelas baseadas em Self-Attention, exploradas em AGRAWAL (2022), pode proporcionar melhores resultados ao capturar nuances temporais e harmônicas dos sinais de áudio. Explorar diferentes arquiteturas poderia oferecer uma abordagem mais robusta para o problema de alinhamento de áudio.

Outra direção importante seria aumentar significativamente o tamanho dos conjuntos de dados de treinamento e teste, seja com a utilização completa dos dados disponíveis nas bases de dados, pela inclusão de novas bases, ou pela aplicação de técnicas de aumento de dados. Isso permitiria que futuros estudos testem hipóteses em um cenário mais amplo e com maior representatividade.

Finalmente, uma investigação mais profunda nas particularidades dos bancos de dados utilizados poderia ajudar a isolar as causas das variações observadas na performance dos modelos. Estudos futuros poderiam se concentrar em analisar detalhadamente os dados de cada banco de dados e entender como suas características específicas afetam os resultados, fornecendo uma visão direcionada para melhorias no desenvolvimento de futuros modelos de alinhamento.

REFERÊNCIAS

AGRAWAL, R. **Towards Context-Aware Neural Performance-Score Synchronisation**. arXiv preprint arXiv:2206.00454, 2022. Disponível em: <https://arxiv.org/abs/2206.00454>.

AGRAWAL, R. ; DIXON, S. **Learning Frame Similarity using Siamese networks for Audio-to-Score Alignment**. arXiv preprint arXiv:2011.07546, 2020. Disponível em: <https://arxiv.org/abs/2011.07546>.

AGRAWAL, R. ; WOLFF, D. ; DIXON, S. **Structure-Aware Audio-to-Score Alignment Using Progressively Dilated Convolutional Neural Networks**. In: IEEE International Conference on Acoustics, Speech and Signal Processing, 2021, Toronto. **Anais**... New York: IEEE, 2021. p. 571–575.

AGRAWAL, R. ; WOLFF, D. ; DIXON, S. A Convolutional-Attentional Neural Framework for Structure-Aware Performance-Score Synchronization. **IEEE Signal Processing Letters**, p. 1–2, 2021.

AQUINO, N. R. et al. The effect of data augmentation on the performance of convolutional neural networks. **Braz. Soc. Comput. Intell**, 2017.

ARZT, A. **Flexible and Robust Music Tracking**. Tese (Doutorado de Ciências Técnicas) - Department of Computational Perception, Technische Wissenschaften, Johannes Kepler University Linz, p. 151, 2016.

CHICCO, D. Siamese Neural Networks: An Overview. In: CARTWRIGHT, H. (Ed.). **Artificial Neural Networks**. New York, NY: Springer US, 2021. p. 73–94.

CHOPRA, S. ; HADSELL, R. ; LECUN, Y. **Learning a similarity metric discriminatively, with application to face verification**. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, Porto Rico. **Anais**... New York: International Committee on Computational Linguistics, 2005. p. 539–546.

CONT, A. et al. **Evaluation of real-time audio-to-score alignment**. In: International Symposium on Music Information Retrieval, 2007, Vienna. **Anais**... Vienna: International Society for Music Information Retrieval, 2007.

CORTES, C. ; JACKEL, L. D. ; CHIANG, W.-P. **Limits on Learning Machine Accuracy Imposed by Data Quality**. (G. Tesauro, D. Touretzky, T. Leen, Eds.)In: Advances in Neural Information Processing Systems, 1994, Colorado. **Anais**... Colorado, 1994. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/1994/file/1e056d2b0ebd5c878c550da6ac5d3724-Paper.pdf>

CUI, W. et al. **Face Recognition via Convolutional Neural Networks and Siamese Neural Networks**. In: International Conference on Intelligent Computing, Automation and Systems, 2019, Chongqing. **Anais**... New York: IEEE, 2019. p. 746–750.

DEY, S. et al. SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification. **CoRR:abs/1707.02131**, 2017.

GHORAANI, B. ; KRISHNAN, S. Time–Frequency Matrix Feature Extraction and Classification of Environmental Audio Signals. **IEEE Transactions on Audio, Speech, and Language Processing**, v. 19, n. 7, p. 2197–2209, 2011.

GOLDER, A. et al. Practical Approaches Toward Deep-Learning-Based Cross-Device Power Side-Channel Attack. **IEEE Transactions on Very Large Scale Integration (VLSI) Systems**, v. 27, n. 12, p. 2720–2733, 2019.

GONG, Z. ; ZHONG, P. ; HU, W. Diversity in machine learning. **IEEE Access**, v. 7, p. 64323–64350, 2019.

GOSWAMI, S. **Impact of Data Quality on Deep Neural Network Training**. arXiv preprint arXiv:2002.03732, 2020. Disponível em: <https://arxiv.org/abs/2002.03732>.

GRÖCHENIG, K. The Short-Time Fourier Transform. In: **Foundations of time-frequency analysis**. Birkhäuser, Boston: Springer Science & Business Media, 2013. p. 73–94.

HENKEL, F. ; WIDMER, G. **Multi-modal Conditional Bounding Box Regression for Music Score Following**. 29th European Signal Processing Conference, 2021, Dublin. **Anais**... New York: IEEE, 2021.

HERNANDEZ-OLIVAN, C. ; BELTRÁN, J. R. Music Composition with Deep Learning: A Review. **CoRR:abs/2108.12290**, p. 25–50, 2021.

KAAVYA SRISKANDARAJA, E. A., Vidhyasaharan Sethu. **Deep Siamese Architecture Based Replay Detection for Secure Voice Biometric**. Interspeech, 2018, Graz. **Anais**... Sydney, 2018. p. 671–675.

KWON, T. ; JEONG, D. ; NAM, J. **Audio-to-score alignment of piano music using RNN-based automatic music transcription**. arXiv preprint arXiv:1711.04480, 2017. Disponível em: <https://arxiv.org/abs/1711.04480>.

LEPISTÖ, J. **Audio Similarity with Siamese Networks**. Tese (Bacharelado de Tese de Ciências) - Faculty of Information Technology and Communication Sciences, Computing and Electrical Engineering, Tampere University, p. 32, 2023.

LINDSAY, M. B. et al. **Impact of data variety on physics-informed neural network lens design**. In: SPIE Optics + Optoelectronics, 2023, California. **Anais**... California, 2023. p. 125300–125301.

MARLON ALMSTRÖM, THI THU HÒA TRÂN. **Voice Feature Extraction Using Siamese Neural Networks for Detecting Impersonators**. Tese (Mestrado em Ciências Matemáticas) - Faculty of Science, Mathematical Statistics, Lund University, p. 62, 2022.

MORSI, A. ; SERRA, X. **Bottlenecks and solutions for audio to score alignment research**. 23nd International Society for Music Information Retrieval Conference, 2022, Bengaluru. **Anais**... Bengaluru: International Society for Music Information Retrieval, 2022. p. 272–279.

MÜLLER, M. Dynamic Time Warping. In: **Information Retrieval for Music and Motion**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. p. 69–84.

MÜLLER, M. **Fundamentals of music processing: Audio, analysis, algorithms, applications**. Cham: Springer, 2015. v. 5 p. 1–65

PERMANASARI, Y. ; HARAHAP, E. H. ; ALI, E. P. Speech recognition using Dynamic Time Warping (DTW). **Journal of Physics: Conference Series**, v. 1366, n. 1, p. 12091–12092, 2019.

RAHIMI, A. et al. **D3: Data Diversity Design for Systematic Generalization in Visual Question Answering**. arXiv preprint arXiv:2309.08798, 2023. Disponível em: <https://arxiv.org/abs/2309.08798>.

SHARMA, G. ; UMAPATHY, K. ; KRISHNAN, S. Trends in audio signal feature extraction methods. **Applied Acoustics**, v. 158, p. 107020–107021, 2020.

SHARMA, N. et al. Siamese Convolutional Neural Network-Based Twin Structure Model for Independent Offline Signature Verification. **Sustainability**, v. 14, n. 18, 2022.

TZANETAKIS, G. ; COOK, P. Musical genre classification of audio signals. **IEEE Transactions on Speech and Audio Processing**, v. 10, n. 5, p. 293–302, 2002.

UTKIN, L. V. ; KOVALEV, M. S. ; KASIMOV, E. M. **An explanation method for Siamese neural networks**. Disponível em: <https://arxiv.org/abs/1911.07702>.

YU, Y. ; KHADIVI, S. ; XU, J. **Can Data Diversity Enhance Learning Generalization?**. (N. Calzolari et al., Eds.) 29th International Conference on Computational Linguistics, 2022, Gyeongju. **Anais**... New York: International Committee on Computational Linguistics, out. 2022. p. 4933–4945.