

MODELO DE APRENDIZAGEM DE MÁQUINA PARA PREDIZER TRECHOS DE MÚSICAS DE CHOPIN

Gabriel Ariel Degenhardt

Prof. Aurélio Faustino Hoppe – Orientador

1 INTRODUÇÃO

A música é uma forma de arte complexa, abstrata e inerentemente subjetiva. Pode-se afirmar, que tal subjetividade está intrínseca nas várias formas como compositores são capazes de se expressar através de composições musicais. Ao longo dos séculos novos conceitos e convenções foram estabelecidas a fim de formar os sistemas que atuam até hoje como pilares da composição de novas peças musicais (MIRANDA, 2001).

Segundo Chen e Miikkulainen (2001), com o passar dos anos foram pesquisados novos algoritmos para aprimorar o processo de análise e composição musical. Porém, nas últimas décadas houve um grande avanço na geração de músicas de forma automática. Utilizando Com as Redes Neurais Artificiais (RNA) foi possível desenvolver um sistema que gera novas melodias no estilo das músicas do compositor Béla Bartok, confirmando a possibilidade de computadores comporem novas músicas, porém havendo dificuldades e limitações neste processo.

Comentado [AS1]: Deve-se evitar iniciar a frase com gerúndio. Gerúndio complementa alguma ideia.

Dentre as principais dificuldades encontradas na composição de músicas, Eck e Schmidhuber (2002) ressaltam a interdependência entre notas musicais numa peça. Notas de um determinado ponto da música sempre dependem das notas que vieram anteriormente e das notas que ainda tocarão. Por conta disso é necessário desenvolver um sistema capaz de se lembrar de certas notas durante toda a duração da música enquanto esquece de outras notas que serão menos relevantes. Contudo, o processo de composição não pode copiar músicas já existentes, mas deve criar obras originais que aparentem terem sido desenvolvidas por um processo criativo, se preocupando com a geração de melodias e harmonias que combinem com o contexto da música (PEARCE; WIGGINS, 2002).

Um dos primeiros estudos de geração de composições musicais, de Chen e Miikkulainen (2001), não utilizou uma base de dados para o treino da RNA, mas sim de algoritmos genéticos com regras pré-definidas para se aproximar do estilo de composição desejada, sendo uma forma pouco maleável para gerar músicas de estilos distintos. No entanto, o uso de Redes Neurais Recorrentes (RNR) se demonstrou eficiente nos estudos posteriores, pois esse modelo de RNA é capaz de memorizar informações pertinentes por mais tempo, identificando quais notas musicais tocaram durante toda a música para haver maior consistência na composição.

Comentado [AS2]: Qual a Fonte desta afirmação?

A partir deste contexto, esse trabalho propõe o desenvolvimento de um modelo de aprendizagem de máquina que seja capaz de analisar o estilo de músicas fornecidas como entrada, criando composições originais que sejam indistinguíveis para um leigo ou prolongando músicas existentes adicionando novos trechos. Para tanto, será utilizado uma base de dados contendo composições de piano de Frédéric Chopin para se-criar gerar músicas que se assemelhem similares ao seu estilo, tornando possível comparar as músicas geradas pelo modelo com as músicas do compositor original.

1.1 OBJETIVOS

O objetivo deste trabalho é disponibilizar um modelo computacional capaz de gerar novos trechos a partir de músicas já existentes de Chopin.

Os objetivos específicos são:

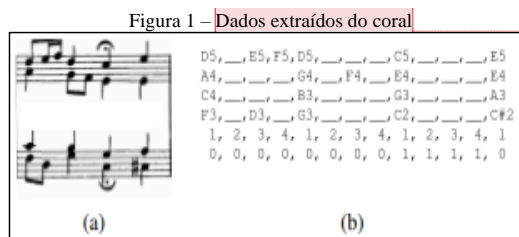
- avaliar a eficiência e a adaptabilidade do modelo de Redes Neurais Recorrentes;
- analisar automaticamente se as músicas geradas não copiam trechos de músicas existentes da base de dados;
- avaliar as músicas geradas através de um questionário, verificando se é possível identificar quais músicas foram geradas pela RNA dentre as músicas originais.

2 TRABALHOS CORRELATOS

Neste capítulo são apresentados trabalhos com objetivos semelhantes ao do estudo proposto. A seção 2.1 descreve um *plugin* de geração de trechos de corais de Bach para o programa MuseScore (HADJERES *et al.*, 2017). A seção 2.2 discorre sobre uma biblioteca Magenta para gerar músicas ao estilo de Bach e avaliar as músicas geradas (SÁ JÚNIOR *et al.*, 2019). Por fim, a seção 2.3 aborda a utilização de algoritmos genéticos para geração de melodias ao estilo de Bela Bartok (CHEN; MIKKULAINEN, 2001).

2.1 DEEPBACH: A STEERABLE MODEL FOR BACH CHORALES GENERATION

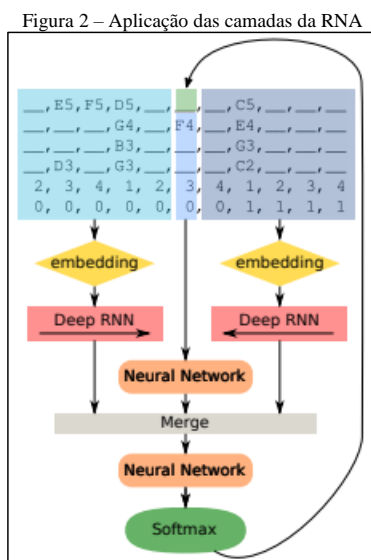
Hadjeres *et al.* (2017) desenvolveram um modelo probabilístico gráfico para representar músicas polifônicas e, em específico, harmonizações de corais de Johann Sebastian Bach. Diante do modelo descrito, foi possível gerar corais no estilo da música de Bach, de modo que o autor da música fosse imperceptível quanto a sua origem, gerada automaticamente ou pelo próprio compositor. No modelo, o usuário podia mudar a forma como a música era gerada definindo por conta própria algumas limitações como notas, ritmos e cadências na peça gerada. A Figura 1 demonstra como os dados do coral são extraídos. Eles foram divididos em quatro vozes (soprano, alto, tenor e baixo), sendo representados numa tabela em função do tempo da música, indicando se uma voz está parada ou mantendo uma nota.



Comentado [AS3]: Colocar estilo da legenda. Faça isso em todas as figuras.

Fonte: Hadjeres *et al.* (2017).

A Figura 2 representa a arquitetura das RNAs do algoritmo. Nela, foram utilizadas quatro RNAs: uma RNR que analisa os dados anteriores ao tempo atual (o passado); uma RNR para os dados posteriores ao tempo atual (o futuro); há uma rede neural não recorrente para processar os dados do tempo atual; e, então o resultado dessas três redes é passado para uma última rede neural não recorrente que é capaz de processar as interdependências entre as demais redes neurais. As RNRs da estrutura existem por conta da característica sequencial das notas, sendo responsáveis por manterem consistência nas notas utilizadas durante a música do início ao fim. Neste caso, as redes neurais não recorrentes são utilizadas para processar acordes e sequências de notas que se repetem na melodia. Segundo Hadjeres *et al.* (2017), essa divisão de camadas serve para contornar o fato de que a música é escrita de forma sequencial, com cada nota dependendo do contexto inserido e de notas que ainda virão.



Fonte: Hadjeres *et al.* (2017).

Hadjeres *et al.* (2017) utilizaram a biblioteca Keras com TensorFlow para a linguagem de programação Python. Já a base de dados utilizada para gerar o modelo consiste em 389 corais de Bach com duração aproximada

de um minuto em formato Musical Instrument Digital Interface (MIDI)¹, o qual permite a separação de cada voz do coral. Também foram removidas as partes instrumentais para aumentar a base de dados, realizando a transposição dos corais para se chegar a 2503 amostras, sendo 80% para treinamento e 20% para teste.

De acordo com Hadjeres *et al.* (2016), foram extraídos trechos de corais gerados tanto pelo DeepBach assim como por outros dois modelos: modelo de máxima entropia e um modelo *perceptron* de múltiplas camadas. Dentre os modelos selecionados, o mais eficiente para reproduzir músicas de Bach foi o DeepBach, estando logo na frente do modelo *perceptron* de múltiplas camadas, sendo que o modelo de máxima entropia obteve os piores resultados.

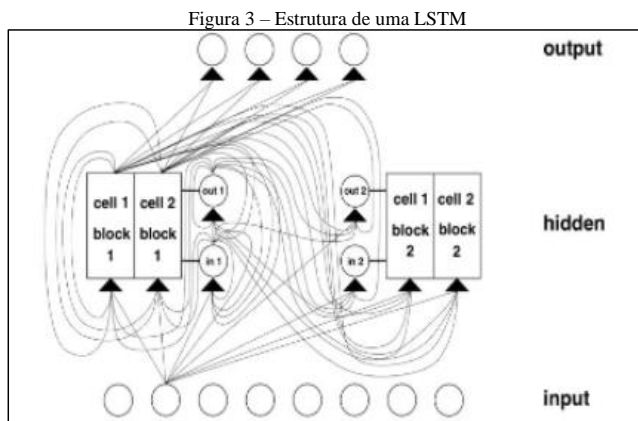
Hadjeres *et al.* (2017) realizaram um experimento com 1272 indivíduos, no qual em cerca de 50% dos casos os trechos gerados pela máquina passaram no teste, recebendo votos como se tivessem sido compostos por Bach. Os autores ressaltam que os resultados dos testes foram satisfatórios e demonstram que é possível criar um modelo flexível capaz de criar músicas que possam convencer até profissionais da área. No entanto, Hadjeres *et al.* (2017) também apontam possíveis melhorias no modelo, como refinar a interface com o usuário para possibilitar maior interação humana e expressão artística, além de aumentar a velocidade da geração da música ou trabalhar com uma base de dados menor.

2.2 GERAÇÃO DE MÚSICA COM APRENDIZADO DE MÁQUINA

Sá Júnior *et al.* (2019) utilizaram a biblioteca Magenta para gerar músicas ao estilo de Johann Sebastian Bach tendo como base composições de Bach no formato MIDI. Ainda segundo os autores, o modelo para treinamento e geração das músicas foi o Polyphony RNR, um modelo do Projeto Magenta utilizado em conjunto com a biblioteca do TensorFlow.

Dentre as opções de RNA disponíveis no Magenta, Sá Júnior *et al.* (2019) escolheram o Polyphony RNR, se tratando de um modelo *Long Short-Term Memory* (LSTM) que permite identificar notas pressionadas por múltiplos compassos ou ao mesmo tempo. Os autores destacam que RNR é o modelo predominante de ~~redes-Redes neurais-Neurais artificiais-Artificiais~~ para a geração de músicas, pois cada nota musical tem um grande impacto nas notas futuras que serão tocadas, porém a RNR memoriza apenas as informações recentes e por conta disso, o modelo LSTM (demonstrado na Figura 3) foi adotado, sendo uma variação mais recente do RNR que não se esquece das informações anteriores da música.

Comentado [AS4]: Frase confusa e muito longa. Rever.



Fonte: Horcheiter e Schmidhuber (1997).

De acordo com Sá Júnior *et al.* (2019), das 20 músicas geradas no projeto, foi averiguado que nenhuma possuía uma sequência de mais de três notas que repetiam nas músicas da base de dados, demonstrando que a rede não estava copiando trechos das músicas do treino, sendo considerado um resultado satisfatório. Por fim, os autores realizaram uma pesquisa para averiguar a eficiência do Magenta para a geração de músicas. Obteve-se 154

¹ MIDI é um padrão utilizado para a comunicação entre instrumentos musicais e equipamentos eletrônicos, como por exemplo, teclado e computadores. Uma partitura MIDI contém instruções que determinam os instrumentos, notas e timbres. Para tanto, cada é nota atribuída a um valor MIDI (CORRÊA, 2008, p. 20).

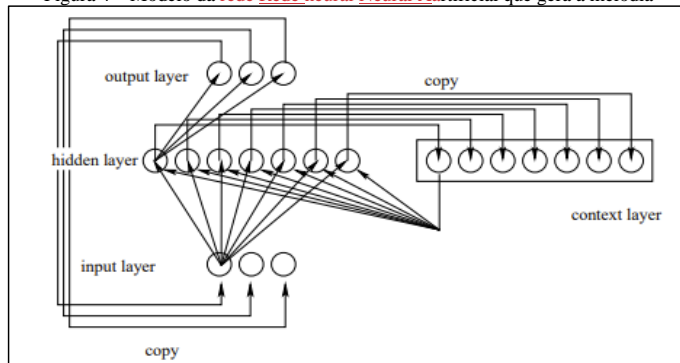
respostas a partir de um questionário online, sendo que 52,4% dos participantes acharam que a música composta pelo Magenta era de Bach.

Sá Júnior *et al.* (2019) concluíram que os resultados dos testes foram satisfatórios quanto aos usuários leigos que realizaram a pesquisa. Porém os entrevistados com maior conhecimento de teoria musical consideraram as músicas geradas como caóticas, possuindo certas inconsistências ao longo da música. O modelo de LSTM do Magenta não permite a criação de melodias com dinâmica de intensidade e tempo.

2.3 CREATING MELODIES WITH EVOLVING RECURRENT NEURAL NETWORKS

Chen e Miikkulainen (2001) desenvolveram um sistema para gerar novas melodias de Bartok. Segundo os autores, optou-se por esse compositor pois profissionais de música já conhecem suas estruturas, facilitando a verificação se as músicas geradas poderiam ser consideradas como estilo de Bartok. Para isso, os autores destacam que é necessário utilizar um modelo de RNA que seja capaz de armazenar informações para manter os padrões da música constantes durante o processo. Contudo, Chen e Miikkulainen (2001) utilizaram a arquitetura de Redes Recorrentes Simples (RNS), uma variação da RNR. A Figura 4 demonstra a arquitetura da RNS utilizada, sendo que os dados da saída de um tempo T são enviados para as entradas do tempo $T + 1$, mantendo-se apenas a camada de contexto que influencia a saída de cada iteração.

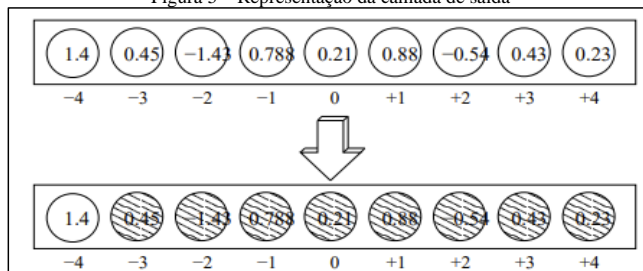
Figura 4 – Modelo da rede-Rede neural-Neural Artificial que gera a melodia



Fonte: Chen e Miikkulainen (2001).

A Figura 5 representa a camada de saída da rede-RNA neural-artificial, sendo que os números de -4 até +4 representam a distância da última nota tocada para a próxima nota que deverá tocar. O valor dentro dos nós representam os pesos baseados no contexto atual da RNS, e o nó com maior valor define qual é a próxima nota a ser tocada. No exemplo ilustrado o valor 1.4 é o maior, portanto a próxima nota tocará 4 casas abaixo da nota atual. Chen e Miikkulainen (2001) mencionam que o processo para definir a duração de cada nota é o mesmo, na qual a duração que possuir o nó com maior valor ganha e é atribuída à nota atual.

Figura 5 – Representação da camada de saída



Fonte: Chen e Miikkulainen (2001).

Chen e Miikkulainen (2001) também utilizaram um algoritmo genético para evoluir a rede-neural-artificial-RNA. A partir dele, foram definidas sete regras para a criação das funções de fitness, assim como regras sobre teoria musical em geral e regras que condizem com os estudos das composições de Bartok. Segundo os

autores, para definir se uma composição é de seu estilo ou não, tais regras recompensam os cromossomos que resultarem em composições mais semelhantes à de Bartok.

A partir dos resultados alcançados, Chen e Miikkulainen (2001) constataram que, após 195 gerações do algoritmo genético, as músicas geradas eram variadas e se assemelhavam às obras de Bartok, porém a música se perdia com o decorrer do tempo, isso acontecia porque os compassos mais ao final da música não carregavam semelhanças o suficiente com os compassos anteriores. A função de *fitness* definida funcionou para gerar músicas variadas, porém em certos casos as músicas geradas não possuíam diversidade o suficiente de notas. Além disso, também não foi realizada nenhuma pesquisa para avaliar ou comparar as músicas geradas. Chen e Miikkulainen (2001) consideram como possíveis melhorias a possibilidade ~~de~~ o algoritmo gerar acordes para formar uma harmonia e a inclusão de novas regras de teoria musical para criar músicas mais organizadas.

3 PROPOSTA DA BIBLIOTECA

Neste capítulo será apresentada a justificativa para o desenvolvimento do estudo proposto. Será demonstrado um quadro comparativo entre os trabalhos correlatos. Na sequência serão descritos os Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF), finalizando com a metodologia e o cronograma planejado para o desenvolvimento do projeto.

3.1 JUSTIFICATIVA

O Quadro 1 sumariza os trabalhos correlatos que foram descritos. É possível perceber que a maioria dos estudos sobre geração de música utilizam variações do modelo de redes neurais RNR e como as bases de dados utilizam as composições de Bach.

Quadro 1 – Comparativo dos trabalhos correlatos			
Trabalhos	Hadjeres <i>et al.</i> (2017)	Sá Júnior <i>et al.</i> (2019)	Chen e Miikkulainen (2001)
Compositor do estilo	Bach	Bach	Bartok
Tipo de música	Corais com quatro vozes	Piano	Piano
Gera acordes	Sim	Sim	Não
RNA utilizada	RNR	Magenta Polyphony RNR	RNS
Base de dados utilizada	Corais a 4 vozes de Bach	Músicas de Bach	Nenhum Não utilizou
Formato de mídia	MIDI	MIDI	Formato próprio

Fonte: elaborado pelo autor.

A partir do Quadro 1 pode-se observar que o compositor Bach foi de grande importância para os estudos pois, segundo Hadjeres *et al.* (2017), ele possui muitas composições de coral e piano, bem como utiliza princípios semelhantes para compor cada uma de suas obras. Já Chen e Miikkulainen (2001) estudaram Bartok, pois suas músicas eram compostas de forma metódica, criando um modelo de ~~rede neural artificial~~RNA para gerar apenas melodias, não harmonias.

A maioria dos trabalhos utiliza uma arquitetura baseada na RNR para realizar o treinamento com a base de dados, sendo LSTM o principal método utilizado. Chen e Miikkulainen (2001) se diferenciaram ao utilizar um algoritmo genético. Sá Júnior *et al.* (2019) utilizaram a ferramenta Magenta desenvolvida pela Google, em conjunto com o TensorFlow, ao qual disponibilizam uma variedade de algoritmos de RNA para geração de músicas, sendo escolhido o Magenta Polyphony RNR como foco do trabalho, tendo internamente uma arquitetura LSTM permite o uso de acordes.

O trabalho de Chen e Miikkulainen (2001) gerava apenas melodias, no qual duas ou mais notas não podem ser tocadas simultaneamente, enquanto o trabalho de Sá Júnior *et al.* (2019) e o de Hadjeres *et al.* (2017) estão preparados para gerar músicas com acordes e harmonização. Chen e Miikkulainen (2001) optaram por retornar como saída as informações relevantes da música, como as notas geradas, a duração e o ritmo da música. Todavia, Sá Júnior *et al.* (2019) e Hadjeres *et al.* (2017) optaram por gerar arquivos MIDI como saída bem como utilizaram MIDI na base de dados. Os arquivos em formato MIDI possuem informações de quais notas musicais estão tocando a cada momento da música, sendo mais maleáveis para manipulação e leves quando comparado a outros formatos de áudio, como por exemplo, o MP3.

Segundo Sá Júnior *et al.* (2019), os leigos em músicas não foram capazes de identificar quais músicas foram geradas pela ~~rede neural artificial~~RNA e quais foram compostas por Bach, porém os mais experientes em música clássica consideraram as músicas como “bagunçadas” e “aleatórias”, isso também foi observado no estudo de Hadjeres *et al.* (2017). O modelo validado por Sá Júnior *et al.* (2019) se provou eficiente para gerar pequenos trechos de música que possuem notas que combinam bem umas com as outras, porém ao se expandir o modelo

para a geração de músicas com maior duração, a música passa a impressão de ser composta por partes desconexas que não harmonizam umas com as outras.

Chen e Miikkulainen (2001) utilizaram algoritmos genéticos com funções *fitness* criadas para delimitar qual tipo de música deve ser gerada, limitando fatores como duração das notas musicais, intervalo entre as notas e quais notas combinam umas com as outras. Segundo os autores, as músicas geradas pela RNS possuíam partes individualmente boas, mas que não se juntavam de forma coesa para formar uma música com um bom fluxo. Dessa forma, Chen e Miikkulainen (2001) sugeriram a adição de restrições para que a música se tornasse menos dinâmica.

Diante deste cenário, este trabalho propõe disponibilizar um modelo próprio de ~~Rede Neural Artificial~~ baseado em RNR e LSTM ~~que seja~~ capaz de estender músicas fornecidas como entrada ou gerar novas músicas de forma aleatória. Inicialmente, optou-se por uma base de dados de Frederic Chopin para realizar os testes de eficiência do modelo, mas, também será avaliado a sua adaptabilidade em outras bases de dados. Chopin é um compositor caracterizado por suas peças solo de piano, que possuem um alto nível de complexidade, porém são mais simples que corais e orquestras que possuem diversos instrumentos. A partir disso, espera-se que as músicas geradas não copiem trechos de músicas da base de dados. Para avaliar esta condição, será realizada uma pesquisa com usuário leigos e compositores, a fim de validar a eficiência do modelo. Também acredita-se que o trabalho proposto trará os seguintes benefícios sociais e/ou práticos: (i) permitir que pessoas sem conhecimento de teoria musical possam criar novas melodias para projetos próprios; (ii) servir como ferramenta de estudo para se perceber características de certos estilos e compositores, ao utilizar uma base de dados específica; (iii) servir como ferramenta para preservar e resgatar o estilo de composição de compositores que não estejam mais ativos, permitindo a criação de novas composições que resgatem características que foram perdidas ao longo do tempo.

3.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

O modelo a ser desenvolvido deverá:

- permitir ao usuário informar a duração, em segundos, de novos trechos musicais a serem gerados/preditos pelo modelo (Requisito Funcional – RF);
- treinar a ~~rede Rede neural Neural artificial Artificial~~ a partir de músicas fornecidas como entrada (RF);
- prever sequências de notas musicais utilizando o modelo LSTM (RF);
- gerar novas músicas ou trechos a partir de diferentes bases de dados (RF);
- estabelecer a similaridade melódica segundo as distribuições de durações² e de *pitch class*³ (RF);
- disponibilizar estatísticas de quantos trechos da música copiaram uma sequência de notas (RF);
- calcular as taxas de desempenho do modelo: sensibilidade, especificidade e precisão (RF);
- desenvolver o modelo na linguagem de programação Python (Requisito Não Funcional – RNF);
- utilizar a biblioteca Keras para desenvolvimento e treinamento do modelo (RNF);
- aceitar arquivos do formato MIDI como entrada (RNF);
- gerar arquivos do formato MIDI como saída (RNF).

3.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- levantamento bibliográfico: realizar o levantamento de fontes bibliográficas quanto ao uso de RNR e LSTM para geração de músicas, conceitos de estrutura musical e trabalhos correlatos;
- elaboração de requisitos: redefinir os requisitos funcionais e não funcionais de forma que atendam ao escopo do trabalho;
- especificação: elaborar o diagrama de atividades do modelo a ser desenvolvido de acordo com os requisitos definidos e criar a documentação para o uso das rotinas;
- aquisição da base de dados: buscar e organizar músicas de piano de Chopin e de outros compositores de música clássica no formato MIDI para realizar o treinamento e testes da RNA;
- definição dos algoritmos: definir possíveis técnicas e uma arquitetura para as camadas de RNA de acordo com a documentação realizada na etapa (c) e o levantamento bibliográfico da etapa (a);
- desenvolvimento do modelo: a partir do que foi formalizado nas etapas (c) e (e) realizar a implementação da RNA, realizando ajustes na arquitetura caso necessário, utilizando a linguagem Python e a biblioteca Keras;

² Distribuição de durações: mensura a similaridade da frequência de ocorrência das durações que compõem a música (SALAZAR, 2015, p. 49)

³ Distribuição de *pitch class*: mensura a similaridade entre as notas que compõem o conjunto de alturas e o número de vezes que cada nota acontece nesse conjunto (SALAZAR, 2015, p. 49)

- g) testes: disponibilizar uma pesquisa online com trechos de músicas originais de Chopin e músicas geradas pelo modelo desenvolvido, de modo a avaliar a eficiência da RNA a partir da percepção dos usuários. Este processo será repetido utilizando outras bases de dados a fim de verificar a adaptabilidade do modelo. Além disso, também será avaliado se os novos trechos possuem notas repetidas, alternâncias abruptas de altura, notas fora de escala e o grau similaridade com relação a outros trechos existentes na composição.

As etapas serão realizadas nos períodos relacionados no Quadro 2.

Quadro 2 – Cronograma de atividades a serem realizadas

etapas / quinzenas	2021									
	fev.		mar.		abr.		maio		jun.	
	1	2	1	2	1	2	1	2	1	2
levantamento bibliográfico										
elicitación de requisitos										
especificação										
aquisição da base de dados										
definição dos algoritmos										
desenvolvimento da modelo										
testes										

Fonte: elaborado pelo autor.

4 REVISÃO BIBLIOGRÁFICA

Este capítulo está dividido em duas seções tendo como objetivo explorar assuntos de importância para o entendimento do estudo a ser realizado. A seção 4.1 apresenta os principais elementos das estruturas de teoria musical. Por fim, a seção 4.2 apresenta os conceitos fundamentais das ~~redes-Redes neurais-Neurais~~ Recorrentes e o funcionamento das ~~redes-Redes neurais-Neurais artificiais-Artificiais~~ LSTM.

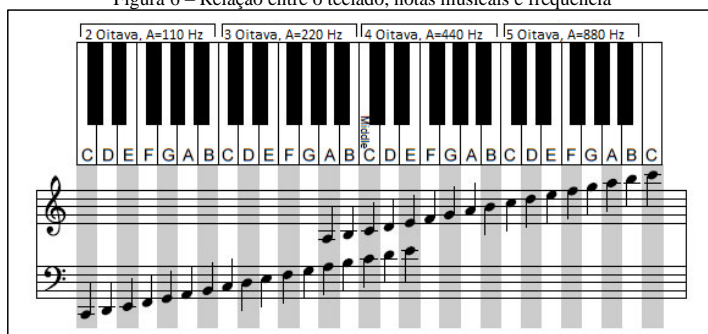
4.1 ESTRUTURA MUSICAL

As músicas são compostas por uma mistura de timbres, melodias, harmonias e ritmo. Timbres variam de acordo com os instrumentos musicais. As melodias variam em escalas e modos, harmonias variam em qualidade e os ritmos variam pela quantidade de batidas por minuto (SCHOENBERG, 1999).

De acordo com Nobre (2008), o sistema ocidental de música define as seguintes notas musicais, de acordo com sua frequência sonora: C (dó), C# (dó sustenido), D (ré), D# (ré sustenido), E (mi), F (fá), F# (fá sustenido), G (sol), G# (sol sustenido), A (lá), A# (lá sustenido) e B (si). Além disso, algumas notas possuem um semitom de intervalo uma das outras (como de E para F) enquanto outras notas possuem um tom de intervalo (como de C para D).

Corrêa (2008) afirma que as notas musicais atuam em ciclos proporcionais, chamados de oitavas. Na Figura 6 pode-se perceber que a frequência da nota A2 (A na segunda oitava) é a metade da nota A3 (A na terceira oitava), ao qual é possível identificar uma nota e sua oitava pela sua frequência. O autor também ressalta que por causa da diferença de frequência entre as notas musicais, as notas de maior frequência são mais agudas enquanto as notas de menor frequência são mais graves.

Figura 6 – Relação entre o teclado, notas musicais e frequência



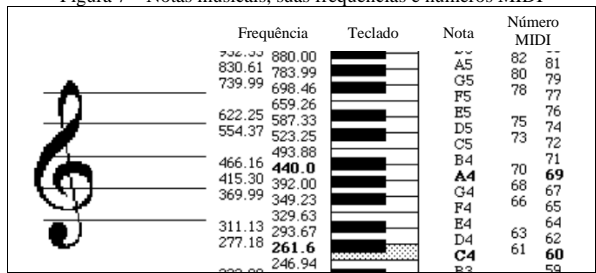
Fonte: adaptado de Corrêa (2008).

De acordo com Procopio (2016), a melodia se trata da linha principal da música, no qual, a harmonia é criada em volta. Segundo o autor, a melodia é composta por uma sequência de notas musicais que são tocadas de forma sequencial. Além disso, a melodia deve utilizar notas musicais para criar e liberar tensão na música. Portanto, utilizando-se dos tons é possível fazer subir ou descer a linha principal da música para poder expressar as emoções da peça.

Segundo Lacerda (1966), a harmonia é a parte da música que expande e aplica novas ideias na melodia. De forma simplificada, a harmonia é a montagem de acordes, tratando-se de notas musicais distintas tocadas simultaneamente que ressoam como um plano de fundo para a melodia. O autor também ressalta que a harmonia pode tornar uma melodia simples em uma peça orquestral, dando emoções à música, sem fugir da linha principal da melodia.

Para Corrêa (2008), o timbre é o principal fator que difere a origem do som, uma mesma nota musical pode ser tocada em diversos instrumentos, porém cada instrumento se diferencia de acordo com as ondas sonoras geradas. O ser humano é capaz de distinguir facilmente se uma determinada nota musical está sendo tocada por um violão ou piano, mesmo que ambos estejam tocando notas de-com a mesma frequência. Além disso, segundo o autor, como cada instrumento gera ondas sonoras com uma assinatura específica, existem variações quanto a amplitude e complexidade das ondas. Tais variações são perceptíveis aos nossos ouvidos. Portanto, cada nota musical está associada a uma frequência. A Figura 7 ilustra as frequências das notas e seus respectivos números no formato MIDI, correspondentes a um teclado de piano.

Figura 7 – Notas musicais, suas frequências e números MIDI



Fonte: adaptado de Corrêa (2008).

Por fim, Miranda (2001) destaca que a duração em composições musicais pode se referir ao tempo que uma nota é tocada ou ao tempo entre duas notas (pausas). Ainda segundo o autor, o ritmo da música é determinado pela duração de suas notas, sendo medido em batidas por minuto. Além disso, o ritmo pode ser o mesmo do início ao fim da música ou pode mudar em cada seção para expressar uma nova emoção na composição.

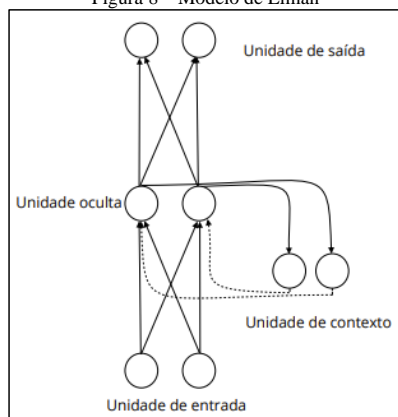
4.2 REDES NEURAIS RECORRENTES

Segundo Deng e Yu (2014), as redes neurais artificiais recorrentes são uma classe de redes neurais artificiais capaz de prever os dados do futuro a partir de exemplos anteriores. De acordo com os autores, ela se diferencia dos demais modelos devido a sua capacidade de memorização, sendo utilizada no processamento de dados sequenciais, como processamento de textos ou de linguagem falada.

Para D’arbo Junior (1998), é possível dividir os modelos de RNR em completamente ou parcialmente recorrentes. As redes completamente recorrentes se caracterizam pela interconexão entre suas unidades de processamento, onde todas as unidades são interconectadas com todas as outras unidades da arquitetura. Enquanto, as redes parcialmente recorrentes são caracterizadas pelas suas estruturas de alimentação direta, que possuem conexões cuidadosamente escolhidas umas com as outras.

Gomes (2005) classifica a RNR como uma rede *feedforward*, ou seja, uma RNA de arquitetura semelhante aos dos neurônios biológicos. A rede possui pesos sinápticos correspondente para cada ligação entre neurônios, de forma que o treinamento da rede altera os pesos sinápticos das ligações para influenciar o processamento de dados posteriores. Dentre os modelos desenvolvidos de RNR, o mais conhecido se trata do modelo de Elman, ilustrado na Figura 8.

Figura 8 – Modelo de Elman



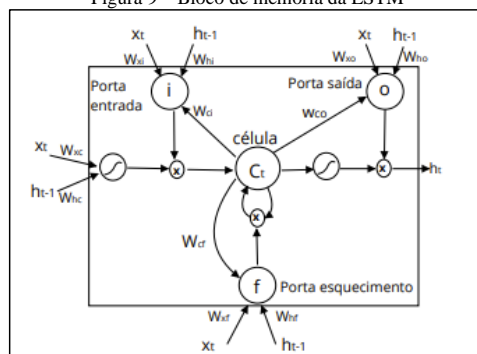
Fonte: Santana (2017).

No modelo de Elman é possível observar como as saídas de um neurônio podem ser reintroduzidas em qualquer neurônio através da unidade de contexto, mantendo o resultado de todos os elementos passados na sequência. Porém, há a defasagem para cada realimentação dos dados. O processo de realimentação das saídas de um neurônio como entrada de outros neurônios é o que caracteriza a RNA como *feedforward*. Tal modelo possui problemas de desaparecimento e explosão do gradiente. O problema da explosão do gradiente refere-se ao grande aumento na norma do gradiente durante o treino. Estes eventos ocorrem devido à explosão dos componentes de longo prazo, que podem crescer exponencialmente com a retroalimentação. Entretanto, o desaparecimento do gradiente é inerente ao comportamento oposto, quando a diferença na atualização dos pesos a longo prazo decai rapidamente à norma 0, tornando impossível o aprendizado do modelo em relação os eventos temporalmente distantes (SANTANA, 2017).

D'arbo Junior (1998) afirma que o algoritmo de retropropagação através do tempo é comumente utilizado no treinamento de RNR. No qual, na fase de treinamento a rede é replicada T vezes, sendo T o número de passos de tempo da sequência. Os pesos iniciais dos neurônios são escolhidos aleatoriamente e, da camada de saída até a inicial, são recalculados os pesos dos neurônios de acordo com os erros obtidos pelos pesos aleatórios. Ainda segundo o autor, existem variações do algoritmo para minimizar seus principais problemas, como necessidade de armazenamento, tempo de cálculo por simulação e número de exemplos de treinamento.

Segundo Horcheiter e Schmidhuber (1997), a rede LSTM foi proposta para resolver os problemas de lembrança de informações a longo prazo. Neste intuito, surgiu o modelo LSTM que é capaz de memorizar informações de forma semelhante ao cérebro humano. Para Salazar (2015), a rede LSTM se destaca em casos nos quais as informações mais antigas do contexto são importantes para o aprendizado, devido a sua estrutura mais complexa com blocos de memória. A LSTM tem o poder de retirar ou adicionar informações às células do bloco de memória, permitindo que certas informações permaneçam relevantes por mais tempo desde que a rede não a tenha esquecido. Na Figura 9, pode-se observar um modelo que representa a arquitetura dos blocos de memória da LSTM. ~~No qual, e~~Cada bloco possui uma ou mais células de memória conectadas com três portas: esquecimento, entrada e saída. As portas permitem redefinir, escrever e ler as operações do bloco de memória. Todas as interações externas são realizadas através dessas portas, a operação é definida através de matrizes de pesos conectadas a cada porta (COSTA *et al.*, 2018).

Figura 9 – Bloco de memória da LSTM



Fonte: Santana (2017).

Segundo Corrêa (2008), a rede LSTM possui um fluxo de erro constante nas células de memória, permitindo que os erros não decresçam quando retropropagados, melhorando a aprendizagem da rede. O processo de treinamento da LSTM é semelhante ao das RNR, geralmente é utilizado a retropropagação através do tempo, porém com a atualização dos pesos das portas de entrada, saída e esquecimento (SALAZAR, 2015).

Para com Deng e Yu (2014), tanto a LSTM como a RNR podem usufruir de métodos de treinamento com aprendizado supervisionado e não supervisionado. O aprendizado supervisionado consiste em providenciar tanto os dados de um padrão a ser reconhecido como a classificação desses dados para um algoritmo da RNA. A partir dessas informações, a rede é capaz de encontrar relações entre os dados e sua classificação. O aprendizado não supervisionado se difere do supervisionado pois os dados não são previamente classificados, de forma que a rede busque relações entre os dados sem haver informações prévias definidas. Segundo os autores, essa forma de aprendizado trouxe resultados positivos para reconhecimento de linguagem falada utilizando RNR, porém para se identificar os padrões de fala de forma eficiente é necessária a introdução de novas camadas escondidas para a arquitetura da rede.

REFERÊNCIAS

- CHEN, Chun-Chi. J.; MIIKKULAINEN, Risto. Creating Melodies with Evolving Recurrent Neural Network. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS, 3, 2001, Piscataway. **Proceedings...** Piscataway: University of Texas at Austin, 2001. p. 2241-2246.
- CORRÊA, Débora Cristina. **Sistema baseado em redes neurais para composição musical assistida por computador**. 2008. 165 f. Dissertação (Mestrado em Ciências Exatas e da Terra) - Universidade Federal de São Carlos, São Carlos.
- D'ARBO JUNIOR, Hélio. **Redes neurais recorrentes para produção de sequências temporais**. 1998. 139 f. Dissertação (Mestrado em Engenharia Elétrica) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos.
- DENG, Li; YU, Dong. Deep Learning Methods and Applications. **Foundations and Trends in Signal Processing**, Hanover, v. 7, n. 3-4, p. 197-387, jun. 2014.
- ECK, Douglas; SCHMIDHUBER, Jergen. **A First Look at Music Composition using LSTM Recurrent Neural Networks**. Manno: Istituto Dalle Molle di studi sull'intelligenza artificiale, 2002. 11 p.
- GOMES, Daniel Takata. **Modelos de redes neurais recorrentes para previsão de series temporais de memórias curta e longa**. 2005. 137 f. Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica, Campinas.
- HADJERES, Gaetan; PACHET, François; NIELSEN, Frank. DeepBach: a Steerable Model for Bach Chorales Generation. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 34, 2017, Sydney. **Proceedings...** Sydney: PMLR, 2017. p. 1362-1371.
- HORCHEITER, Sepp; SCHMIDHUBER, Jergen. Long Short-Term Memory. **Neural computation**, Cambridge, v. 9, n. 8, p. 1735-1780, nov. 1997.
- LACERDA, Osvaldo Costa. **Compendio de Teoria Elementar da Música**. São Paulo: Ricordi, 1966.
- MIRANDA, Eduardo Reck. **Composing Music with Computers**. Burlington: Focal Press, 2001.

NOBRE, Jorge. **Projeto Fortalecimento Musical**. [S.l.], 2008. Disponível em: <http://www2.secult.ce.gov.br/Recursos/PublicWebBanco/Partituraacervo/Apt000002.pdf>. Acesso em: 22 nov. 2020.

PEARCE, Marcus; WIGGINS, Geraint A. Aspects of a Cognitive Theory of Creativity in Musical Composition. In: ECAI02 WORKSHOP ON CREATIVE SYSTEMS, 2, 2002, Lyon. **Proceedings...** Lyon: University Claude Bernard. 2002. p. 17-24.

PROCOPIO, Joe. **Basic Music Theory**. New Jersey: Bookbaby, 2016.

SÁ JÚNIOR, Joel Alexandre *et al.* Geração de música com aprendizado de máquina. **Colloquium Exactarum**, Presidente Prudente, v. 11, n.2, p. 56-65, jul. 2019.

SALAZAR, Andrés Eduardo Coca. **Mineração de estruturas musicais e composição automática utilizando redes complexas**. 2015. 197 f. Tese (Doutorado em Ciências de Computação e Matemática Computacional) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos.

SANTANA, Luciana Maiara Queiroz de. **Aplicação de redes neurais recorrentes no reconhecimento automático da fala em ambientes com ruídos**. 2017. 68 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Sergipe, São Cristóvão.

SCHOENBERG, Arnold. **Fundamentals of Musical Composition**. Londres: Faber & Faber, 1999.

ASSINATURAS

(Atenção: todas as folhas devem estar rubricadas)

Assinatura do(a) Aluno(a): _____

Assinatura do(a) Orientador(a): _____

Assinatura do(a) Coorientador(a) (se houver): _____

Observações do orientador em relação a itens não atendidos do pré-projeto (se houver):

FORMULÁRIO DE avaliação – PROFESSOR AVALIADOR

Acadêmico(a): Gabriel Ariel Degenhardt _____

Avaliador(a): Andreza Sartori _____

ASPECTOS AVALIADOS ¹		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?	X		
	O problema está claramente formulado?	X		
	2. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?	X		
	Os objetivos específicos são coerentes com o objetivo principal?	X		
	3. TRABALHOS CORRELATOS São apresentados trabalhos correlatos, bem como descritas as principais funcionalidades e os pontos fortes e fracos?	X		
	4. JUSTIFICATIVA Foi apresentado e discutido um quadro relacionando os trabalhos correlatos e suas principais funcionalidades com a proposta apresentada?	X		
	São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?	X		
	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?	X		
	5. REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO Os requisitos funcionais e não funcionais foram claramente descritos?	X		
	6. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?	X		
	Os métodos, recursos e o cronograma estão devidamente apresentados e são compatíveis com a metodologia proposta?	X		
	7. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?	X		
ASPECTOS METODOLÓGICOS	As referências contemplam adequadamente os assuntos abordados (são indicadas obras atualizadas e as mais importantes da área)?	X		
	8. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?	X		
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?	X		

PARECER – PROFESSOR AVALIADOR: (PREENCHER APENAS NO PROJETO)

O projeto de TCC ser deverá ser revisado, isto é, necessita de complementação, se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos 5 (cinco) tiverem resposta ATENDE PARCIALMENTE.

PARECER: (x) APROVADO () REPROVADO

Assinatura: _____ Data: 07/12/2020 _____

PROJETO TCC - BCC	ANO/SEMESTRE:	2020.2
-------------------	---------------	--------

MODELO DE APRENDIZAGEM DE MÁQUINA PARA PREDIZER TRECHOS DE MÚSICAS DE CHOPIN

Gabriel Ariel Degenhardt

Prof. Aurélio Faustino Hoppe – Orientador

1 INTRODUÇÃO

A música é uma forma de arte complexa, abstrata e inerentemente subjetiva. Pode-se afirmar, que tal subjetividade está intrínseca nas várias formas como compositores são capazes de se expressar através de composições musicais. Ao longo dos séculos novos conceitos e convenções foram estabelecidas a fim de formar os sistemas que atuam até hoje como pilares da composição de novas peças musicais (MIRANDA, 2001).

Segundo Chen e Miikkulainen (2001), com o passar dos anos foram pesquisados novos algoritmos para aprimorar o processo de análise e composição musical. Porém, nas últimas décadas houve um grande avanço na geração de músicas de forma automática. Utilizando Redes Neurais Artificiais (RNA) foi possível desenvolver um sistema que gera novas melodias no estilo das músicas do compositor Béla Bartok, confirmando a possibilidade de computadores comporem novas músicas, porém havendo dificuldades e limitações neste processo.

Dentre as principais dificuldades encontradas na composição de músicas, Eck e Schmidhuber (2002) ressaltam a interdependência entre notas musicais numa peça. Notas de um determinado ponto da música sempre dependem das notas que vieram anteriormente e das notas que ainda tocarão. Por conta disso é necessário desenvolver um sistema capaz de se lembrar de certas notas durante toda a duração da música enquanto esquece de outras notas que serão menos relevantes. Contudo, o processo de composição não pode copiar músicas já existentes, mas deve criar obras originais que aparentem terem sido desenvolvidas por um processo criativo, se preocupando com a geração de melodias e harmonias que combinem com o contexto da música (PEARCE; WIGGINS, 2002).

Um dos primeiros estudos de geração de composições musicais, de Chen e Miikkulainen (2001), não utilizou uma base de dados para o treino da RNA, mas sim de algoritmos genéticos com regras pré-definidas para se aproximar do estilo de composição desejada, sendo uma forma pouco maleável para gerar músicas de estilos distintos. No entanto, o uso de Redes Neurais Recorrentes (RNR) se demonstrou eficiente nos estudos posteriores, pois esse modelo de RNA é capaz de memorizar informações pertinentes por mais tempo, identificando quais notas musicais tocaram durante toda a música para haver maior consistência na composição.

A partir deste contexto, esse trabalho propõe o desenvolvimento de um modelo de aprendizagem de máquina que seja capaz de analisar o estilo de músicas fornecidas como entrada, criando composições originais que sejam indistinguíveis para um leigo ou prolongando músicas existentes adicionando novos trechos. Para tanto, será utilizado uma base de dados contendo composições de piano de Frédéric Chopin para se criar músicas que se assemelhem ao seu estilo, tornando possível comparar as músicas geradas pelo modelo com as músicas do compositor original.

1.1 OBJETIVOS

O objetivo deste trabalho é disponibilizar um modelo computacional capaz de gerar novos trechos a partir de músicas já existentes de Chopin.

Os objetivos específicos são:

- avaliar a eficiência e a adaptabilidade do modelo de Redes Neurais Recorrentes;
- analisar automaticamente se as músicas geradas não copiam trechos de músicas existentes da base de dados;
- avaliar as músicas geradas através de um questionário, verificando se é possível identificar quais músicas foram geradas pela RNA dentre as músicas originais.

2 TRABALHOS CORRELATOS

Neste capítulo são apresentados trabalhos com objetivos semelhantes ao do estudo proposto. A seção 2.1 descreve um *plugin* de geração de trechos de corais de Bach para o programa MuseScore (HADJERES *et al.*, 2017). A seção 2.2 discute sobre uma biblioteca Magenta para gerar músicas ao estilo de Bach e avaliar as músicas geradas (SÁ JÚNIOR *et al.*, 2019). Por fim, a seção 2.3 aborda a utilização de algoritmos genéticos para geração de melodias ao estilo de Bela Bartok (CHEN; MIKKULAINEN, 2001).

Comentado [MCL1]: "demonstrou ser"

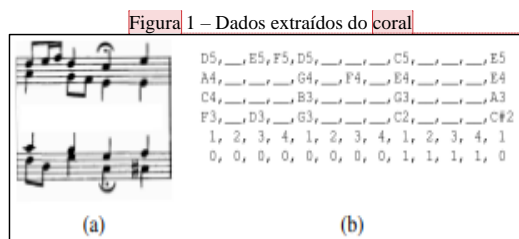
Comentado [MCL2]: utilizada

Comentado [MCL3]: Ainda não sei qual base é. Isso é relevante? Sem esse fim o objetivo perde o sentido?

Comentado [MCL4]: É bom especificar para quem será aplicado.

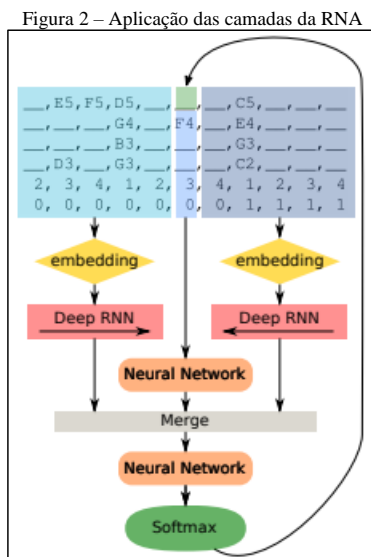
2.1 DEEPBACH: A STEERABLE MODEL FOR BACH CHORALES GENERATION

Hadjeres *et al.* (2017) desenvolveram um modelo probabilístico gráfico para representar músicas polifônicas e, em específico, harmonizações de corais de Johann Sebastian Bach. Diante do modelo descrito, foi possível gerar corais no estilo da música de Bach, de modo que o autor da música fosse imperceptível quanto a sua origem, gerada automaticamente ou pelo próprio compositor. No modelo, o usuário podia mudar a forma como a música era gerada definindo por conta própria algumas limitações como notas, ritmos e cadências na peça gerada. A Figura 1 demonstra como os dados do coral são extraídos. Eles foram divididos em quatro vozes (soprano, alto, tenor e baixo), sendo representados numa tabela em função do tempo da música, indicando se uma voz está parada ou mantendo uma nota.



Fonte: Hadjeres *et al.* (2017).

A Figura 2 representa a arquitetura das RNAs do algoritmo. Nela, foram utilizadas quatro RNAs: uma RNR que analisa os dados anteriores ao tempo atual (o passado), uma RNR para os dados posteriores ao tempo atual (o futuro), há uma rede neural não recorrente para processar os dados do tempo atual e então o resultado dessas três redes é passado para uma última rede neural não recorrente que é capaz de processar as interdependências entre as demais redes neurais. As RNRs da estrutura existem por conta da característica sequencial das notas, sendo responsáveis por manterem consistência nas notas utilizadas durante a música do início ao fim. Neste caso, as redes neurais não recorrentes são utilizadas para processar acordes e sequências de notas que se repetem na melodia. Segundo Hadjeres *et al.* (2017), essa divisão de camadas serve para contornar o fato de que a música é escrita de forma sequencial, com cada nota dependendo do contexto inserido e de notas que ainda virão.



Fonte: Hadjeres *et al.* (2017).

Comentado [MCL5]: Não usou o estilo recomendado.

Comentado [MCL6]: Não usou o estilo recomendado.

Hadjeres *et al.* (2017) utilizaram a biblioteca Keras com TensorFlow para a linguagem de programação Python. Já a base de dados utilizada para gerar o modelo consiste em 389 corais de Bach com duração aproximada de um minuto em formato Musical Instrument Digital Interface (MIDI)¹, o qual permite a separação de cada voz do coral. Também foram removidas as partes instrumentais para aumentar a base de dados, realizando a transposição dos corais para se chegar a 2503 amostras, sendo 80% para treinamento e 20% para teste.

De acordo com Hadjeres *et al.* (2016), foram extraídos trechos de corais gerados tanto pelo DeepBach assim como por outros dois modelos: modelo de máxima entropia e um modelo *perceptron* de múltiplas camadas. Dentre os modelos selecionados, o mais eficiente para reproduzir músicas de Bach foi o DeepBach, estando logo na frente do modelo *perceptron* de múltiplas camadas, sendo que o modelo de máxima entropia obteve os piores resultados.

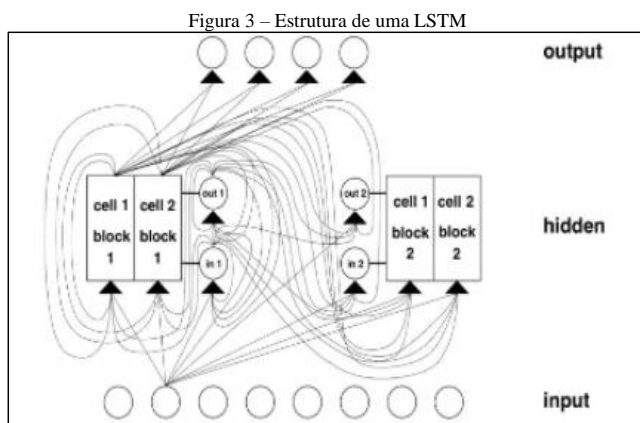
Hadjeres *et al.* (2017) realizaram um experimento com 1272 indivíduos, no qual em cerca de 50% dos casos os trechos gerados pela máquina passaram no teste, recebendo votos como se tivessem sido compostos por Bach. Os autores ressaltam que os resultados dos testes foram satisfatórios e demonstram que é possível criar um modelo flexível capaz de criar músicas que possam convencer até profissionais da área. No entanto, Hadjeres *et al.* (2017) também apontam possíveis melhorias no modelo, como refinar a interface com o usuário para possibilitar maior interação humana e expressão artística, além de aumentar a velocidade da geração da música ou trabalhar com uma base de dados menor.

2.2 GERAÇÃO DE MÚSICA COM APRENDIZADO DE MÁQUINA

Sá Júnior *et al.* (2019) utilizaram a biblioteca Magenta para gerar músicas ao estilo de Johann Sebastian Bach tendo como base composições de Bach no formato MIDI. Ainda segundo os autores, o modelo para treinamento e geração das músicas foi o Polyphony RNR, um modelo do Projeto Magenta utilizado em conjunto com a biblioteca do TensorFlow.

Dentre as opções de RNA disponíveis no Magenta, Sá Júnior *et al.* (2019) escolheram o Polyphony RNR, se tratando de um modelo *Long Short-Term Memory* (LSTM) que permite identificar notas pressionadas por múltiplos compassos ou ao mesmo tempo. Os autores destacam que RNR é o modelo predominante de redes neurais artificiais para a geração de músicas, pois cada nota musical tem um grande impacto nas notas futuras que serão tocadas, porém a RNR memoriza apenas as informações recentes e por conta disso, o modelo LSTM (demonstrado na Figura 3) foi adotado, sendo uma variação mais recente do RNR que não se esquece das informações anteriores da música.

Comentado [MCL7]: Não tirou o itálico de todos os nomes próprios.



Fonte: Horcheiter e Schmidhuber (1997).

De acordo com Sá Júnior *et al.* (2019), das 20 músicas geradas no projeto, foi averiguado que nenhuma possuía uma sequência de mais de três notas que repetiam nas músicas da base de dados, demonstrando que a rede não estava copiando trechos das músicas do treino, sendo considerado um resultado satisfatório. Por fim, os autores

¹ MIDI é um padrão utilizado para a comunicação entre instrumentos musicais e equipamentos eletrônicos, como por exemplo, teclado e computadores. Uma partitura MIDI contém instruções que determinam os instrumentos, notas e timbres. Para tanto, cada é nota atribuída a um valor MIDI (CORRÊA, 2008, p. 20).

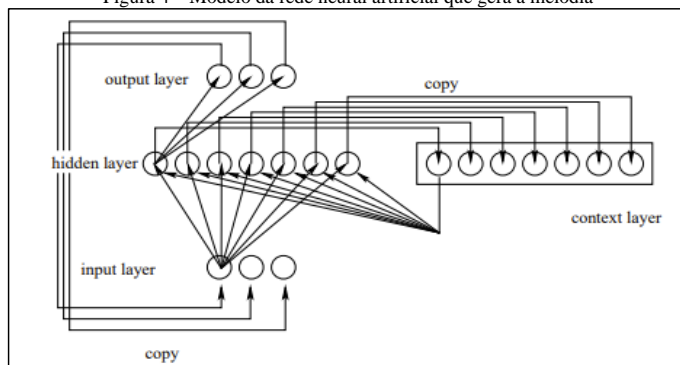
realizaram uma pesquisa para averiguar a eficiência do Magenta para a geração de músicas. Obteve-se 154 respostas a partir de um questionário online, sendo que 52,4% dos participantes acharam que a música composta pelo Magenta era de Bach.

Sá Júnior *et al.* (2019) concluíram que os resultados dos testes foram satisfatórios quanto aos usuários leigos que realizaram a pesquisa. Porém os entrevistados com maior conhecimento de teoria musical consideraram as músicas geradas como caóticas, possuindo certas inconsistências ao longo da música. O modelo de LSTM do Magenta não permite a criação de melodias com dinâmica de intensidade e tempo.

2.3 CREATING MELODIES WITH EVOLVING RECURRENT NEURAL NETWORKS

Chen e Miikkulainen (2001) desenvolveram um sistema para gerar novas melodias de Bartok. Segundo os autores, optou-se por esse compositor pois profissionais de música já conhecem suas estruturas, facilitando a verificação se as músicas geradas poderiam ser consideradas como estilo de Bartok. Para isso, os autores destacam que é necessário utilizar um modelo de RNA que seja capaz de armazenar informações para manter os padrões da música constantes durante o processo. Contudo, Chen e Miikkulainen (2001) utilizaram a arquitetura de Redes Recorrentes Simples (RNS), uma variação da RNR. A Figura 4 demonstra a arquitetura da RNS utilizada, sendo que os dados da saída de um tempo T são enviados para as entradas do tempo $T + 1$, mantendo-se apenas a camada de contexto que influencia a saída de cada iteração.

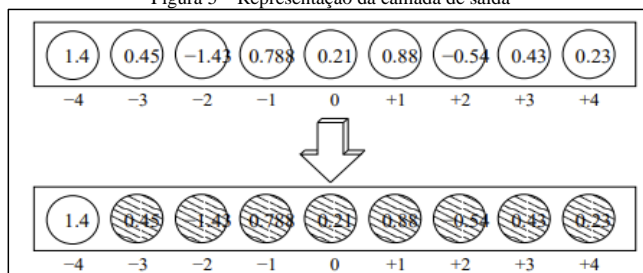
Figura 4 – Modelo da rede neural artificial que gera a melodia



Fonte: Chen e Miikkulainen (2001).

A Figura 5 representa a camada de saída da rede neural artificial, sendo que os números de -4 até +4 representam a distância da última nota tocada para a próxima nota que deverá tocar. O valor dentro dos nós representam os pesos baseados no contexto atual da RNS, e o nó com maior valor define qual é a próxima nota a ser tocada. No exemplo ilustrado o valor 1.4 é o maior, portanto a próxima nota tocará 4 casas abaixo da nota atual. Chen e Miikkulainen (2001) mencionam que o processo para definir a duração de cada nota é o mesmo, na qual a duração que possui o nó com maior valor ganha e é atribuída à nota atual.

Figura 5 – Representação da camada de saída



Fonte: Chen e Miikkulainen (2001).

Chen e Miikkulainen (2001) também utilizaram um algoritmo genético para evoluir a rede neural artificial. A partir dele, foram definidas sete regras para a criação das funções de *fitness*, assim como regras sobre teoria

musical em geral e regras que condizem com os estudos das composições de Bartok. Segundo os autores, para definir se uma composição é de seu estilo ou não, tais regras recompensam os cromossomos que resultarem em composições mais semelhantes à de Bartok.

A partir dos resultados alcançados, Chen e Miikkulainen (2001) constataram que, após 195 gerações do algoritmo genético, as músicas geradas eram variadas e se assemelhavam às obras de Bartok, porém a música se perdia com o decorrer do tempo, isso acontecia porque os compassos mais ao final da música não carregavam semelhanças o suficiente com os compassos anteriores. A função de *fitness* definida funcionou para gerar músicas variadas, porém em certos casos as músicas geradas não possuíam diversidade o suficiente de notas. Além disso, também não foi realizada nenhuma pesquisa para avaliar ou comparar as músicas geradas. Chen e Miikkulainen (2001) consideram como possíveis melhorias a possibilidade do algoritmo gerar acordes para formar uma harmonia e a inclusão de novas regras de teoria musical para criar músicas mais organizadas.

3 PROPOSTA DA BIBLIOTECA

Neste capítulo será apresentada a justificativa para o desenvolvimento do estudo proposto. Será demonstrado um quadro comparativo entre os trabalhos correlatos. Na sequência serão descritos os Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF), finalizando com a metodologia e o cronograma planejado para o desenvolvimento do projeto.

3.1 JUSTIFICATIVA

O Quadro 1 sumariza os trabalhos correlatos que foram descritos. É possível perceber que a maioria dos estudos sobre geração de música utilizam variações do modelo de redes neurais RNR e como as bases de dados utilizam composições de Bach.

Quadro 1 – Comparativo dos trabalhos correlatos

Trabalhos Características	Hadjeres <i>et al.</i> (2017)	Sá Júnior <i>et al.</i> (2019)	Chen e Miikkulainen (2001)
Compositor do estilo	Bach	Bach	Bartok
Tipo de música	Corais com quatro vozes	Piano	Piano
Gera acordes	Sim	Sim	Não
RNA utilizada	RNR	Magenta Polyphony RNR	RNS
Base de dados utilizada	Corais a 4 vozes de Bach	Músicas de Bach	Nenhum
Formato de mídia	MIDI	MIDI	Formato próprio

Fonte: elaborado pelo autor.

A partir do Quadro 1 pode-se observar que o compositor Bach foi de grande importância para os estudos pois, segundo Hadjeres *et al.* (2017), ele possui muitas composições de coral e piano, bem como utiliza princípios semelhantes para compor cada uma de suas obras. Já Chen e Miikkulainen (2001) estudaram Bartok, pois suas músicas eram compostas de forma metódica, criando um modelo de rede neural artificial para gerar apenas melodias, não harmonias.

A maioria dos trabalhos utiliza uma arquitetura baseada na RNR para realizar o treinamento com a base de dados, sendo LSTM o principal método utilizado. Chen e Miikkulainen (2001) se diferenciaram ao utilizar um algoritmo genético. Sá Júnior *et al.* (2019) utilizaram a ferramenta Magenta desenvolvida pela Google, em conjunto com o TensorFlow, ao qual disponibilizam uma variedade de algoritmos de RNA para geração de músicas, sendo escolhido o Magenta Polyphony RNR como foco do trabalho, tendo internamente uma arquitetura LSTM permite o uso de acordes.

O trabalho de Chen e Miikkulainen (2001) gerava apenas melodias, no qual duas ou mais notas não podem ser tocadas simultaneamente, enquanto o trabalho de Sá Júnior *et al.* (2019) e o de Hadjeres *et al.* (2017) estão preparados para gerar músicas com acordes e harmonização. Chen e Miikkulainen (2001) optaram por retornar como saída as informações relevantes da música, como as notas geradas, a duração e o ritmo da música. Todavia, Sá Júnior *et al.* (2019) e Hadjeres *et al.* (2017) optaram por gerar arquivos MIDI como saída bem como utilizaram MIDI na base de dados. Os arquivos em formato MIDI possuem informações de quais notas musicais estão tocando a cada momento da música, sendo mais maleáveis para manipulação e leves quando comparado a outros formatos de áudio, como por exemplo, o MP3.

Segundo Sá Júnior *et al.* (2019), os leigos em músicas não foram capazes de identificar quais músicas foram geradas pela rede neural artificial e quais foram compostas por Bach, porém os mais experientes em música clássica consideraram as músicas como “bagunçadas” e “aleatórias”, isso também foi observado no estudo de

Comentado [MCL8]: Melhorar redação.

Hadjeres *et al.* (2017). O modelo validado por Sá Júnior *et al.* (2019) se provou eficiente para gerar pequenos trechos de música que possuem notas que combinam bem umas com as outras, porém ao se expandir o modelo para a geração de músicas com maior duração, a música passa a impressão de ser composta por partes desconexas que não harmonizam umas com as outras.

Chen e Miikkulainen (2001) utilizaram algoritmos genéticos com funções *fitness* criadas para delimitar qual tipo de música deve ser gerada, limitando fatores como duração das notas musicais, intervalo entre as notas e quais notas combinam umas com as outras. Segundo os autores, as músicas geradas pela RNS possuíam partes individualmente boas, mas que não se juntavam de forma coesa para formar uma música com um bom fluxo. Dessa forma, Chen e Miikkulainen (2001) sugeriram a adição de restrições para que a música se tornasse menos dinâmica.

Diante deste cenário, este trabalho propõe disponibilizar um modelo próprio de rede neural baseado em RNR e LSTM que seja capaz de estender músicas fornecidas como entrada ou gerar novas músicas de forma aleatória. Inicialmente, optou-se por uma base de dados de Frederic Chopin para realizar os testes de eficiência do modelo, mas, também será avaliado a sua adaptabilidade em outras bases de dados. Chopin é um compositor caracterizado por suas peças solo de piano, que possuem um alto nível de complexidade, porém são mais simples que corais e orquestras que possuem diversos instrumentos. A partir disso, espera-se que as músicas geradas não copiem trechos de músicas da base de dados. Para avaliar esta condição, será realizada uma pesquisa com usuário leigos e compositores a fim de validar a eficiência do modelo. Também acredita-se que o trabalho proposto trará os seguintes benefícios sociais e/ou práticos: (i) permitir que pessoas sem conhecimento de teoria musical possam criar novas melodias para projetos próprios; (ii) servir como ferramenta de estudo para se perceber características de certos estilos e compositores, ao utilizar uma base de dados específica; (iii) servir como ferramenta para preservar e resgatar o estilo de composição de compositores que não estejam mais ativos, permitindo a criação de novas composições que resgatem características que foram perdidas ao longo do tempo.

Comentado [MCL9]: Falta a relevância para a computação. Pense nisso para o seu TCC.

3.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

O modelo a ser desenvolvido deverá:

- permitir ao usuário informar a duração, em segundos, de novos trechos musicais a serem gerados/preditos pelo modelo (Requisito Funcional – RF);
- treinar a rede neural artificial a partir de músicas fornecidas como entrada (RF);
- predizer sequências de notas musicais utilizando o modelo LSTM (RF);
- gerar novas músicas ou trechos a partir de diferentes bases de dados (RF);
- estabelecer a similaridade melódica segundo as distribuições² e de *pitch class*³ (RF);
- disponibilizar estatísticas de quantos trechos da música copiaram uma sequência de notas (RF);
- calcular as taxas de desempenho do modelo: sensibilidade, especificidade e precisão (RF);
- desenvolver o modelo na linguagem de programação Python (Requisito Não Funcional – RNF);
- utilizar a biblioteca Keras para desenvolvimento e treinamento do modelo (RNF);
- aceitar arquivos do formato MIDI como entrada (RNF);
- gerar arquivos do formato MIDI como saída (RNF).

3.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- levantamento bibliográfico: realizar o levantamento de fontes bibliográficas quanto ao uso de RNR e LSTM para geração de músicas, conceitos de estrutura musical e trabalhos correlatos;
- elicitação de requisitos: redefinir os requisitos funcionais e não funcionais de forma que atendam ao escopo do trabalho;
- especificação: elaborar o diagrama de atividades do modelo a ser desenvolvido de acordo com os requisitos definidos e criar a documentação para o uso das rotinas;
- aquisição da base de dados: buscar e organizar músicas de piano de Chopin e de outros compositores de música clássica no formato MIDI para realizar o treinamento e testes da RNA;
- definição dos algoritmos: definir possíveis técnicas e uma arquitetura para as camadas de RNA de acordo com a documentação realizada na etapa (c) e o levantamento bibliográfico da etapa (a);
- desenvolvimento do modelo: a partir do que foi formalizado nas etapas (c) e (e) realizar a implementação da RNA, realizando ajustes na arquitetura caso necessário, utilizando a linguagem

² Distribuição de durações: mensura a similaridade da frequência de ocorrência das durações que compõem a música (SALAZAR, 2015, p. 49)

³ Distribuição de *pitch class*: mensura a similaridade entre as notas que compõem o conjunto de alturas e o número de vezes que cada nota acontece nesse conjunto (SALAZAR, 2015, p. 49)

- Python e a biblioteca Keras;
- g) testes: disponibilizar uma pesquisa online com trechos de músicas originais de Chopin e músicas geradas pelo modelo desenvolvido, de modo a avaliar a eficiência da RNA a partir da percepção dos usuários. Este processo será repetido utilizando outras bases de dados a fim de verificar a adaptabilidade do modelo. Além disso, também será avaliado se os novos trechos possuem notas repetidas, alternâncias abruptas de altura, notas fora de escala e o grau similaridade com relação a outros trechos existentes na composição.
- As etapas serão realizadas nos períodos relacionados no Quadro 2.

Quadro 2 – Cronograma de atividades a serem realizadas

etapas / quinzenas	2021									
	fev.		mar.		abr.		maio		jun.	
	1	2	1	2	1	2	1	2	1	2
levantamento bibliográfico										
elicitación de requisitos										
especificação										
aquisição da base de dados										
definição dos algoritmos										
desenvolvimento da modelo										
testes										

Fonte: elaborado pelo autor.

4 REVISÃO BIBLIOGRÁFICA

Este capítulo está dividido em duas seções tendo como objetivo explorar assuntos de importância para o entendimento do estudo a ser realizado. A seção 4.1 apresenta os principais elementos das estruturas de teoria musical. Por fim, a seção 4.2 apresenta os conceitos fundamentais das redes neurais recorrentes e o funcionamento das redes neurais artificiais LSTM.

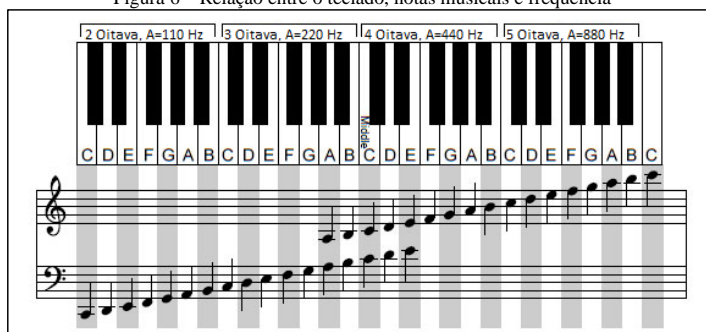
4.1 ESTRUTURA MUSICAL

As músicas são compostas por uma mistura de timbres, melodias, harmonias e ritmo. Timbres variam de acordo com os instrumentos musicais. As melodias variam em escalas e modos, harmonias variam em qualidade e os ritmos variam pela quantidade de batidas por minuto (SCHOENBERG, 1999).

De acordo com Nobre (2008), o sistema ocidental de música define as seguintes notas musicais, de acordo com sua frequência sonora: C (dó), C# (dó sustenido), D (ré), D# (ré sustenido), E (mi), F (fá), F# (fá sustenido), G (sol), G# (sol sustenido), A (lá), A# (lá sustenido) e B (si). Além disso, algumas notas possuem um semitom de intervalo uma das outras (como de E para F) enquanto outras notas possuem um tom de intervalo (como de C para D).

Corrêa (2008) afirma que as notas musicais atuam em ciclos proporcionais, chamados de oitavas. Na Figura 6 pode-se perceber que a frequência da nota A2 (A na segunda oitava) é a metade da nota A3 (A na terceira oitava), ao qual é possível identificar uma nota e sua oitava pela sua frequência. O autor também ressalta que por causa da diferença de frequência entre as notas musicais, as notas de maior frequência são mais agudas enquanto as notas de menor frequência são mais graves.

Figura 6 – Relação entre o teclado, notas musicais e frequência



Fonte: adaptado de Corrêa (2008).

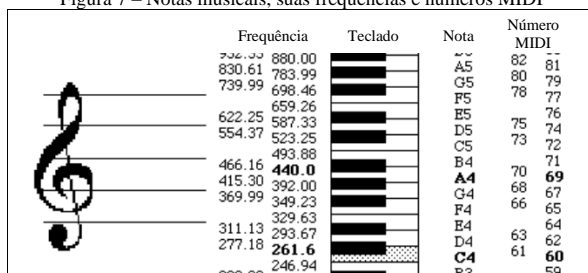
De acordo com Procopio (2016), a melodia se trata da linha principal da música, no qual, a harmonia é criada em volta. Segundo o autor, a melodia é composta por uma sequência de notas musicais que são tocadas de forma sequencial. Além disso, a melodia deve utilizar notas musicais para criar e liberar tensão na música. Portanto, utilizando-se dos tons é possível fazer subir ou descer a linha principal da música para poder expressar as emoções da peça.

Segundo Lacerda (1966), a harmonia é a parte da música que expande e aplica novas ideias na melodia. De forma simplificada, a harmonia é a montagem de acordes, tratando-se de notas musicais distintas tocadas simultaneamente que ressoam como um plano de fundo para a melodia. O autor também ressalta que a harmonia pode tornar uma melodia simples em uma peça orquestral, dando emoções à música, sem fugir da linha principal da melodia.

Para Corrêa (2008), o timbre é o principal fator que difere a origem do som, uma mesma nota musical pode ser tocada em diversos instrumentos, porém cada instrumento se diferencia de acordo com as ondas sonoras geradas. O ser humano é capaz de distinguir facilmente se uma determinada nota musical está sendo tocada por um violão ou piano, mesmo que ambos estejam tocando notas de mesma frequência. Além disso, segundo o autor, como cada instrumento gera ondas sonoras com uma assinatura específica, existem variações quanto a amplitude e complexidade das ondas, tais variações são perceptíveis aos nossos ouvidos. Portanto, cada nota musical está associada a uma frequência. A Figura 7 ilustra as frequências das notas e seus respectivos números no formato MIDI, correspondentes a um teclado de piano.

Comentado [MCL10]: Melhorar redação

Figura 7 – Notas musicais, suas frequências e números MIDI



Fonte: adaptado de Corrêa (2008).

Por fim, Miranda (2001) destaca que a duração em composições musicais pode se referir ao tempo que uma nota é tocada ou ao tempo entre duas notas (pausas). Ainda segundo o autor, o ritmo da música é determinado pela duração de suas notas, sendo medido em batidas por minuto. Além disso, o ritmo pode ser o mesmo do início ao fim da música ou pode mudar em cada seção para expressar uma nova emoção na composição.

4.2 REDES NEURAIIS RECORRENTES

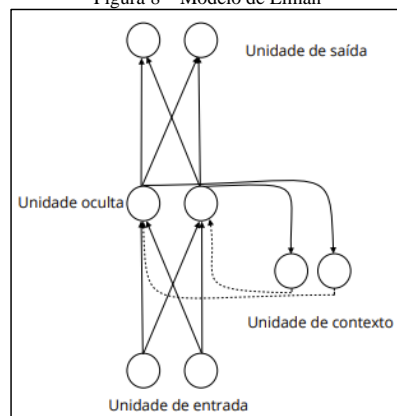
Segundo Deng e Yu (2014), as redes neurais artificiais recorrentes são uma classe de redes neurais artificiais capaz de prever os dados do futuro a partir de exemplos anteriores. De acordo com os autores, ela se

diferencia dos demais modelos devido a sua capacidade de memorização, sendo utilizada no processamento de dados sequenciais, como processamento de textos ou de linguagem falada.

Para D'arbo Junior (1998), é possível dividir os modelos de RNR em completamente ou parcialmente recorrentes. As redes completamente recorrentes se caracterizam pela interconexão entre suas unidades de processamento, onde todas as unidades são interconectadas com todas as outras unidades da arquitetura. Enquanto, as redes parcialmente recorrentes são caracterizadas pelas suas estruturas de alimentação direta, que possuem conexões cuidadosamente escolhidas umas com as outras.

Gomes (2005) classifica a RNR como uma rede *feedforward*, ou seja, uma RNA de arquitetura semelhante aos dos neurônios biológicos. A rede possui pesos sinápticos correspondente para cada ligação entre neurônios, de forma que o treinamento da rede altera os pesos sinápticos das ligações para influenciar o processamento de dados posteriores. Dentre os modelos desenvolvidos de RNR, o mais conhecido se trata do modelo de Elman, ilustrado na Figura 8.

Figura 8 – Modelo de Elman



Fonte: Santana (2017).

No modelo de Elman é possível observar como as saídas de um neurônio podem ser reintroduzidas em qualquer neurônio através da unidade de contexto, mantendo o resultado de todos os elementos passados na sequência. Porém, há a defasagem para cada realimentação dos dados. O processo de realimentação das saídas de um neurônio como entrada de outros neurônios é o que caracteriza a RNA como *feedforward*. Tal modelo possui problemas de desaparecimento e explosão do gradiente. O problema da explosão do gradiente refere-se ao grande aumento na norma do gradiente durante o treino. Estes eventos ocorrem devido à explosão dos componentes de longo prazo, que podem crescer exponencialmente com a retroalimentação. Entretanto, o desaparecimento do gradiente é inerente ao comportamento oposto, quando a diferença na atualização dos pesos a longo prazo decai rapidamente à norma 0, tornando impossível o aprendizado do modelo em relação os eventos temporalmente distantes (SANTANA, 2017).

D'arbo Junior (1998) afirma que o algoritmo de retropropagação através do tempo é comumente utilizado no treinamento de RNR. No qual, na fase de treinamento a rede é replicada T vezes, sendo T o número de passos de tempo da sequência. Os pesos iniciais dos neurônios são escolhidos aleatoriamente e, da camada de saída até a inicial, são recalculados os pesos dos neurônios de acordo com os erros obtidos pelos pesos aleatórios. Ainda segundo o autor, existem variações do algoritmo para minimizar seus principais problemas, como necessidade de armazenamento, tempo de cálculo por simulação e número de exemplos de treinamento.

Segundo Horcheiter e Schmidhuber (1997), a rede LSTM foi proposta para resolver os problemas de lembrança de informações a longo prazo. Neste intuito, surgiu o modelo LSTM que é capaz de memorizar informações de forma semelhante ao cérebro humano. Para Salazar (2015), a rede LSTM se destaca em casos nos quais as informações mais antigas do contexto são importantes para o aprendizado, devido a sua estrutura mais complexa com blocos de memória. A LSTM tem o poder de retirar ou adicionar informações às células do bloco de memória, permitindo que certas informações permaneçam relevantes por mais tempo desde que a rede não tenha esquecido. Na Figura 9, pode-se observar um modelo que representa a arquitetura dos blocos de memória da LSTM. No qual, cada bloco possui uma ou mais células de memória conectadas com três portas: esquecimento, entrada e saída. As portas permitem redefinir, escrever e ler as operações do bloco de memória. Todas as interações

Comentado [MCL11]: tirar

Comentado [MCL12]: Redação

Comentado [MCL13]: Pontuação

Comentado [MCL14]: Não está nas referências.

Comentado [MCL15]: Sem essas linhas em branco

O diagrama ilustra a arquitetura interna de um neurônio artificial. No topo, há duas portas: a 'Porta entrada' (à esquerda) e a 'Porta saída' (à direita). A porta de entrada recebe inputs x_t e h_{t-1} com pesos W_{xi} e W_{hi} respectivamente, e produz o output W_{ci} . A porta de saída recebe inputs x_t e h_{t-1} com pesos W_{xo} e W_{ho} respectivamente, e produz o output W_{co} . No centro, há uma 'célula' (C_t) que recebe inputs W_{ci} e W_{co} . Abaixo da célula, há uma função de esquecimento (f) que recebe inputs W_{ct} e W_{ht} e produz o output W_{ct} . A célula também recebe inputs x_t e h_{t-1} com pesos W_{xt} e W_{ht} respectivamente, e produz o output h_t .

Segundo Corrêa (2008), a rede LSTM possui um fluxo de erro constante nas células de memória, permitindo que os erros não decresçam quando retropropagados, melhorando a aprendizagem da rede. O processo de treinamento da LSTM é semelhante ao das RNR, geralmente é utilizado a retropropagação através do tempo, porém com a atualização dos pesos das portas de entrada, saída e esquecimento (SALAZAR, 2015).

Para com Deng e Yu (2014), tanto a LSTM como a RNR podem usufruir de métodos de treinamento com aprendizado supervisionado e não supervisionado. O aprendizado supervisionado consiste em providenciar tanto os dados de um padrão a ser reconhecido como a classificação desses dados para um algoritmo da RNA. A partir dessas informações, a rede é capaz de encontrar relações entre os dados e sua classificação. O aprendizado não supervisionado se difere do supervisionado pois os dados não são previamente classificados, de forma que a rede busque relações entre os dados sem haver informações prévias definidas. Segundo os autores, essa forma de aprendizado trouxe resultados positivos para reconhecimento de linguagem falada utilizando RNR, porém para se identificar os padrões de fala de forma eficiente é necessária a introdução de novas camadas escondidas para a arquitetura da rede.

CHEN, Chun-Chi. J.; MIIKKULAINEN, Risto. Creating Melodies with Evolving Recurrent Neural Network. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS, 3, 2001, Piscataway. **Proceedings...** Piscataway: University of Texas at Austin, 2001. p. 2241-2246.

CORRÊA, Débora Cristina. **Sistema baseado em redes neurais para composição musical assistida por computador**. 2008. 165 f. Dissertação (Mestrado em Ciências Exatas e da Terra) - Universidade Federal de São Carlos, São Carlos.

D'ARBO JUNIOR, Hélio. **Redes neurais recorrentes para produção de seqüências temporais**. 1998. 139 f. Dissertação (Mestrado em Engenharia Elétrica) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos.

DENG, Li; YU, Dong. Deep Learning Methods and Applications. **Foundations and Trends in Signal Processing**, Hanover, v. 7, n. 3-4, p. 197-387, jun. 2014.

ECK, Douglas; SCHMIDHUBER, Jurgen. **A First Look at Music Composition using LSTM Recurrent Neural Networks**. Manno: Istituto Dalle Molle di studi sull'intelligenza artificiale, 2002. 11 p.

GOMES, Daniel Takata. **Modelos de redes neurais recorrentes para previsão de series temporais de memórias curta e longa**. 2005. 137 f. Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica. Campinas.

HADJERES, Gaetan; PACHET, François; NIELSEN, Frank. DeepBach: a Steerable Model for Bach Chorales Generation. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 34, 2017, Sydney.
Proceedings... Sydney: PMLR, 2017. p. 1362-1371.

HORCHEITER, Sepp; SCHMIDHUBER, Jurgen. Long Short-Term Memory. **Neural computation**, Cambridge, v. 9, n. 8, p. 1735-1780, nov. 1997.

LACERDA, Osvaldo Costa. **Compendio de Teoria Elementar da Música**. São Paulo: Ricordi, 1966.

MIRANDA, Eduardo Reck. **Composing Music with Computers**. Burlington: Focal Press, 2001.

NOBRE, Jorge. **Projeto Fortalecimento Musical**. [S.l.], 2008. Disponível em: <http://www2.secult.ce.gov.br/Recursos/PublicWebBanco/Partituraacervo/Apt000002.pdf>. Acesso em: 22 nov. 2020.

PEARCE, Marcus; WIGGINS, Geraint A. Aspects of a Cognitive Theory of Creativity in Musical Composition. In: ECAI02 WORKSHOP ON CREATIVE SYSTEMS, 2, 2002, Lyon. **Proceedings...** Lyon: University Claude Bernard. 2002. p. 17-24.

PROCOPIO, Joe. **Basic Music Theory**. New Jersey: Bookbaby, 2016.

SÁ JÚNIOR, Joel Alexandre *et al.* Geração de música com aprendizado de máquina. **Colloquium Exactarum**, Presidente Prudente, v. 11, n.2, p. 56-65, jul. 2019.

SALAZAR, Andrés Eduardo Coca. **Mineração de estruturas musicais e composição automática utilizando redes complexas**. 2015. 197 f. Tese (Doutorado em Ciências de Computação e Matemática Computacional) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos.

SANTANA, Luciana Maiara Queiroz de. **Aplicação de redes neurais recorrentes no reconhecimento automático da fala em ambientes com ruídos**. 2017. 68 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Sergipe, São Cristóvão.

SCHOENBERG, Arnold. **Fundamentals of Musical Composition**. Londres: Faber & Faber, 1999.

ASSINATURAS

(Atenção: todas as folhas devem estar rubricadas)

Assinatura do(a) Aluno(a): _____

Assinatura do(a) Orientador(a): _____

Assinatura do(a) Coorientador(a) (se houver): _____

Observações do orientador em relação a itens não atendidos do pré-projeto (se houver):

FORMULÁRIO DE AVALIAÇÃO – PROFESSOR TCC I

Acadêmico(a): Gabriel Ariel Degenhardt _____

Avaliador(a): Mauricio Capobianco Lopes _____

ASPECTOS AVALIADOS ¹		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?	x		
	O problema está claramente formulado?	x		
	2. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?	x		
	Os objetivos específicos são coerentes com o objetivo principal?		x	
	3. JUSTIFICATIVA São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?		x	
	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?	x		
	4. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?	x		
ASPECTOS METODOLÓGICOS	Os métodos, recursos e o cronograma estão devidamente apresentados?	x		
	5. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?	x		
	6. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?	x		
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?	x		
	7. ORGANIZAÇÃO E APRESENTAÇÃO GRÁFICA DO TEXTO A organização e apresentação dos capítulos, seções, subseções e parágrafos estão de acordo com o modelo estabelecido?	x		
	8. ILUSTRAÇÕES (figuras, quadros, tabelas) As ilustrações são legíveis e obedecem às normas da ABNT?		x	
	9. REFERÊNCIAS E CITAÇÕES As referências obedecem às normas da ABNT?	x		
As citações obedecem às normas da ABNT?		x		
Todos os documentos citados foram referenciados e vice-versa, isto é, as citações e referências são consistentes?			x	

PARECER – PROFESSOR DE TCC I OU COORDENADOR DE TCC (PREENCHER APENAS NO PROJETO):

O projeto de TCC será reprovado se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos 4 (quatro) itens dos **ASPECTOS TÉCNICOS** tiverem resposta ATENDE PARCIALMENTE; ou
- pelo menos 4 (quatro) itens dos **ASPECTOS METODOLÓGICOS** tiverem resposta ATENDE PARCIALMENTE.

PARECER: (x) APROVADO () REPROVADO

Assinatura: Mauricio Capobianco Lopes _____ Data: 10/12/2020 _____

¹ Quando o avaliador marcar algum item como atende parcialmente ou não atende, deve obrigatoriamente indicar os motivos no texto, para que o aluno saiba o porquê da avaliação.