Pós-Graduação em Ciência da Computação

“Incorporando conhecimento de contexto via pós-processamento de algoritmos de transcrição automática de acordes musicais”

Por

URAQUITAN SIDNEY GOUVEIA CARNEIRO DA CUNHA

Tese de Doutorado

Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE
2015

URAQUITAN SIDNEY GOUVEIA CARNEIRO DA CUNHA

INCORPORANDO CONHECIMENTO DE CONTEXTO VIA PÓS-PROCESSAMENTO DE ALGORITMOS DE TRANSCRIÇÃO AUTOMÁTICA DE ACORDES MUSICAIS

ESTE TRABALHO FOI APRESENTADO À PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO DO CENTRO DE INFORMÁTICA DA UNIVERSIDADE FEDERAL DE PERNAMBUCO COMO REQUISITO PARCIAL PARA OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIA DA COMPUTAÇÃO

ORIENTADOR: PROF. DR. GEBER LISBOA RAMALHO
COORIENTADOR: PROF. DR. GIORDANO CABRAL

RECIFE
2015

Catalogação na fonte
Bibliotecária Joana D'Arc Leão Salvador CRB4-532

C972i Cunha, Uraquitan Sidney Gouveia Carneiro da.
Incorporando conhecimento de contexto via pós-processamento de algoritmos de transcrição automática de acordes musicais / Uraquitan Sidney Gouveia Carneiro da Cunha. – Recife: O Autor, 2015.
153 f.: fig., tab., quadro, graf.

Orientador: Geber Lisboa Ramalho.
Tese (Doutorado) – Universidade Federal de Pernambuco. CIN, Ciência da Computação, 2015.
Inclui referências e anexos.

1. Redes neurais. 2. Reconhecimento de padrões. 3. Teoria musical. 4. Análise harmônica. 5. Fourier, Séries de. I. Ramalho, Geber Lisboa (Orientador). II. Titulo.

006.3 CDD (22. ed.) UFPE-MEI 2015-106

Tese de Doutorado apresentada por URAQUITAN SIDNEY GOUVEIA CARNEIRO DA CUNHA à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título "Incorporando conhecimento de contexto via pós-processamento de algoritmos de transcrição automática de acordes musicais" orientada pelo Prof. GEBER LISBOA RAMALHO e aprovada pela Banca Examinadora formada pelos professores:

_____

Prof. Paulo Jorge Leitão Adeodato
Centro de Informática / UFPE

_____

Profa. Patricia Cabral de Azevedo Restelli Tedesco
Centro de Informática / UFPE

_____

Prof. Adriano Lorena Inácio de Oliveira
Centro de Informática / UFPE

_____

Prof. Edilson Ferneda
Pró-Reitoria de Pós-Graduação e Pesquisa /UCB

_____

Prof. Tiago Alessandro Espinola Ferreira
Departamento de Estatística e Informática / UFRPE

Visto e permitida a impressão.
Recife, 25 de fevereiro de 2015.

_____

Profa. EDNA NATIVIDADE DA SILVA BARROS
Coordenadora da Pós-Graduação em Ciência da Computação do
Centro de Informática da Universidade Federal de Pernambuco.

Aos meus pais, que por serem quem são, me inspiram a cada dia. À minha esposa sempre torcedora e incentivadora.

Amo vocês!

Agradecimentos

Primeiramente agradeço à banca examinadora da tese, que foi formada por Dr. Adriano Lorena Inácio de Oliveira, Dr. Paulo Jorge Leitão Adeodato (CIN-UFPE), Dra. Patrícia Cabral de Azevedo Restelli Tedesco (CIN-UFPE), Dr. Edilson Ferneda (UCB-DF) e Tiago Alessandro Espinola Ferreira (UFRPE-PE), pela disponibilidade em analisar e avaliar o meu trabalho. Aos professores Edilson Ferneda, Tiago Alessandro e Adriano Lorena, agradeço pelas contribuições e recomendações enriquecedoras feitas em minha qualificação.

Aos meus, orientador e co-orientador, trago um agradecimento especial pela competência e conhecimento ao me guiarem neste mar revolto, sempre com muita objetividade e segurança. Se não fosse por eles, não chegaria até aqui.

Agradeço também em especial aos meus pais, sempre presentes na minha formação e que nunca duvidaram do que seus filhos são capazes. Também não posso deixar de agradecer a força sempre presente da minha esposa. Se ela não podia contribuir tecnicamente, sempre soube dar-me o tempo, paciência, espaço e até a lucidez de que muitas vezes precisei.

Por fim, aos colegas de profissão, pesquisadores parceiros, membros do Mustik e amigos que de uma forma direta ou indireta ajudaram-me neste trabalho.

Resumo

Dentro da área de pesquisa chamada de *Music Information Retrieval* (MIR, ou Recuperação de Informações Musicais), uma tarefa que vem recebendo bastante atenção é a que tenta realizar a transcrição automática dos acordes musicais. Na prática, esta tarefa se traduz no desenvolvimento de softwares capazes de analisar arquivos de canções (MP3, WAV, etc.) e extrair deles as suas grades de acordes. A tarefa é complexa e envolve várias subtarefas, algumas delas com propostas de soluções bem fundamentadas e relativamente bem sucedidas no estado da arte.

Para atuar na execução desta tarefa, as propostas de soluções em estado da arte não têm considerado algumas informações musicais relevantes. Entre elas, podemos citar as sequências recorrentes ou típicas de acordes comumente encontradas na música ocidental (IIm-V-I, I-IVm-IIm-V7, etc.), e a presença de estruturas cíclicas como refrães e estrofes. O conhecimento destas informações de caráter preditivo poderia facilitar a execução da tarefa de transcrição de acordes na medida em que ajudaria a prevê-los. No entanto, a utilização de informação preditiva nesta tarefa de classificação envolve diversas incertezas.

Nesta tese, a questão de pesquisa central a ser respondida é se o conhecimento e uso deste tipo de informação musical preditiva pode melhorar, de fato, o desempenho de um processo de transcrição de acordes. Para tanto, também seria preciso propor um meio de como adquirir este tipo de conhecimento e de como incorporá-lo na transcrição. A resposta a estes últimos pontos baseou-se em um caminho agregador capaz de aproveitar os resultados das melhores propostas de soluções, pós-processando seus resultados para melhorá-los com o uso de informação preditiva.

Nos testes realizados, através do uso de uma rede neural do tipo MLP devidamente treinada, foi demonstrado que é possível incorporar o conhecimento relacionado com este tipo de informação musical preditiva para melhorar o desempenho dos sistemas de transcrição de acordes. Nossa proposta indica que estas informações musicais, de fato, são relevantes e podem melhorar a transcrição de acordes.

Palavras Chaves: MIR. Aprendizagem de Máquina. Previsão de Acordes. Contexto Musical. Transcrição de Acordes. Reconhecimento de Acordes. Redes Neurais. HMM.

Abstract

In the domain of Music Information Retrieval (MIR), one of the tasks which have been receiving attention is the automatic transcription of musical chords. In practice, this task consists in developing software capable of analyzing song files (MP3, WAV , etc.) and drawing from them their chord grids. The task is complex and it involves several subtasks. Some well-founded and relatively well succeeded solutions have been proposed in the state of the art.

To act in this task, the proposed solutions in state of the art have not considered some relevant music information. Among them, we can mention the typical chord sequences commonly found in western song (IIm-VI, I-IVm-IIm-V7, etc.), and the presence of cyclic structures like choruses and verses. In both cases, knowledge of this information usually could facilite the execution of the chord transcription task. However, the use of this predictive information on this classification task involves several uncertainties.

In this thesis, the central question to be assessed is whether the knowledge and use of such information can improve the performance of a chord transcription process. Therefore, it would also be necessary to propose a way of how a system might acquire this knowledge. The answer to these questions, instead of relying on a new method that treats the process of transcription in all its complexities, was based on an aggregator way able to take advantage of the best results of proposed solutions, post-processing their transcripts of chords and trying to improve them with the use of this kind of information.

In the tests performed, through the use of a properly trained neural network MLP type, it was demonstrated that it is possible to incorporate knowledge related to this type of musical information to improve the performance of the chord transcription systems. Our proposal indicates that these musical information, in fact, are relevant and may improve the transcription of chords.

Key Words: MIR. Machine Learning. Chord Prediction. Musical Context. Chords Transcription. Chords Recognition. Neural Networks. SVM. ELM. HMM.

Lista de Ilustrações

Lista de Tabelas, Gráficos e Quadros

Lista de Siglas

CRF	-	Condition Random Fields
DFT	-	Discrete Fourier Transform
FFT	-	Fast Fourier Transform
HMM	-	Hidden Markov Model
ISMIR	-	International Society for Music Information Retrieval
MIR	-	Music Information Retrieval
MIREX	-	Music Information Retrieval Evaluation eXchange
MSE	-	Mean Square Error
MLP	-	Multilayer Perceptron
NNLS	-	Non-Negative Least Squares
PCP	-	Pitch Class Profile
SOM	-	Self Organized Maps
ST	-	Sequence Tracker
SVM	-	Support Vector Machines
TC	-	Tonal Centroid
WAV	-	Waveform Audio File Format

Sumário

1. Introdução

Com o advento das Tecnologias de Informação e Comunicação - TIC e da internet, há um número cada vez maior de dados disponíveis, incluindo dados multimídia como imagens, vídeos e música. A extração de informação destes dados envolve interesses científicos, comerciais, educacionais, entre outros, e para realização deste tipo de tarefa, são cada vez mais necessários algoritmos e métodos especializados.

1.1. Motivação e Contexto

No caso dos dados musicais, uma quantidade cada vez mais relevante de pesquisas vem sendo desenvolvidas na área chamada de *Music Information Retrieval* (MIR, ou Recuperação de Informações Musicais). Pesquisadores de várias partes do mundo têm trabalhado na criação de algoritmos para, por exemplo, reconhecer o estilo musical de uma composição (PIKRAKIS, 2013; WU, 2013), reconhecer acordes de uma canção (CHO; BELLO, 2013; STEENBERGEN; BURGOYNE, 2013), identificar o andamento de uma canção (CANNAM *et al.*, 2013; ELOWSSON; FRIBERG, 2013) extrair a melodia de uma canção (SONG; LI, 2013), entre outras tarefas.

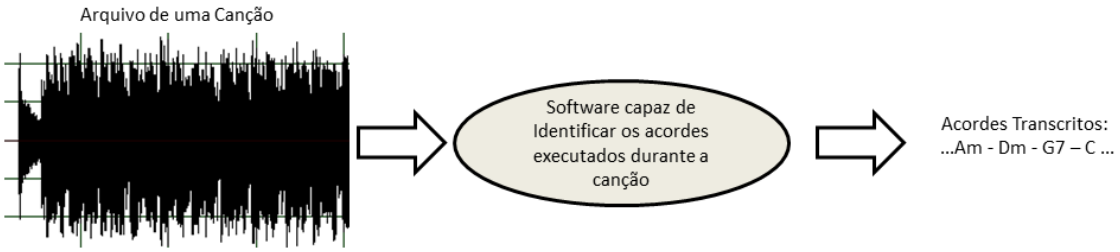


Figura 1.1 - Processo automatizado de transcrição de acordes

Neste cenário de MIR, uma das tarefas que vêm recebendo bastante atenção é a que tenta realizar a transcrição automática dos acordes musicais de uma canção qualquer. Na prática, esta tarefa se traduz no desenvolvimento de softwares

capazes de analisar arquivos de canções (MP3, WAV, etc.) e extrair dos mesmos as suas grades de acordes (Figura 1.1).

A tarefa de transcrição dos acordes de uma canção por um computador é de grande importância e interesse para a comunidade de pesquisa em computação musical. Tornar uma máquina habilitada a realizar esta tarefa poderia trazer benefícios nas áreas de composição musical, arranjos de canções, auto-acompanhamento, improvisação, educação musical, além de poder ajudar a se obter uma melhor compreensão da estrutura lógica das harmonias das canções. Investimentos nesta área estão presentes inclusive no universo privado, onde empresas buscam o desenvolvimento de soluções para o problema que nos propomos a lidar, gerando novas propostas e alternativas de navegação por canções (VIRO, 2011), formatos diferenciados de sistemas de recomendação (BETH, 2004) e até aplicações de cunho educacional, como por exemplo, o iChords (DACCORD, 2014).

Muitos algoritmos e modelos vêm sendo propostos para tentar executar esta tarefa de transcrição de acordes de forma bem sucedida. Todos os anos eles têm tido a oportunidade de serem testados e comparados na conferência anual da *ISMIR (The International Society for Music Information Retrieval)*, onde sempre se realiza o *MIREX - Music Information Retrieval Evaluation eXchange* (MIREX, 2013), uma competição entre vários algoritmos inscritos com o objetivo de realizar tarefas específicas de MIR. No caso da tarefa de transcrição de acordes, os algoritmos e propostas são avaliados no grupo do MIREX chamado de *Audio Chord Estimation* (ou *Audio Chord Detection*).

Neste contexto, avanços relevantes têm sido alcançados, sobretudo a partir da última década. Porém, além do problema de transcrição de acordes ainda não ter uma solução definitiva (BURGOYNE; WILD; FUJINAGA, 2011), após a realização de extensa análise sobre as propostas de solução mais recentes, percebemos que elas ainda apresentam margem para avanços. Esta percepção veio da observação de que algumas informações importantes relacionadas com o contexto musical, e que são normalmente úteis para um processo de transcrição de acordes, não vem sendo plenamente utilizadas.

De fato, na música ocidental existem sequências comuns ou típicas de três a quatro acordes que se repetem com frequência em uma canção (CHEDIAK, 1986). É o caso, por exemplo, da sequência denominada genericamente de IIm-V7- I

(equivalente a Dm-G7-C na tonalidade de Dó maior), muito comum no jazz, na MPB, pop, etc. Sequências desta natureza podem ajudar num processo de transcrição de um acorde a partir da análise local do seu sinal de áudio. Por exemplo, supondo que a análise local do sinal de áudio não permita saber com certeza se um acorde é um G7 ou um Bm, saber que o acorde anterior foi um Dm e que o posterior é um C pode ser muito útil, pois aumenta a probabilidade de que o acorde seja um G7 (CHEDIAK, 1986).

Por outro lado, além das sequencias típicas de acordes, na música ocidental também é muito comum a presença de estruturas cíclicas como refrães, estrofes ou seções que se repetem ao longo da execução de uma canção. A identificação da presença destas estruturas facilita o processo de definição dos acordes de uma canção porque a maior parte delas possui grades de acordes similares.

Estes tipos de sequências típicas e estruturas cíclicas podem ser tratadas como informações de potencial caráter preditivo e capazes de ajudar num processo de transcrição de acordes. Esta crença tem origem no fato de que no passado construímos um sistema híbrido de previsão de acordes que, utilizando tais informações, teve um excelente desempenho. Neste sistema, uma rede neural devidamente treinada recebia como entrada uma sequência de três acordes e previa qual deveria ser o próximo acorde de uma dada canção. Paralelamente e em parceria com a rede neural, um módulo de rastreamento de estruturas cíclicas também foi desenvolvido na definição deste sistema híbrido (CUNHA, 1999). Quando este módulo percebia que havia uma repetição de algum trecho da canção em curso, ele assumia o controle do processo de previsão; quando não, o controle da previsão era assumido pela rede neural. Com este modelo de sistema alcançamos percentuais de sucesso nos testes de previsões que giraram em torno de 87%, resultado que era melhorado paulatinamente na medida em que o módulo rastreador identificava estruturas cíclicas (ou trechos repetidos) da canção. Nesta pesquisa, uma das grandes limitações estava no fato de que todo o treinamento e previsão realizados pelo sistema aconteceram tomando como base informações retiradas de partituras de canções, e não a partir da análise direta dos sinais de áudio das mesmas, que é foco dos trabalhos em MIR. Como discutiremos mais tarde, passar de um cenário de informações simbólicas (partituras) para um de dados numéricos (sinais de áudio) traz muitos e novos desafios.

1.2. Questões de Pesquisa, Objetivos e Escopo Negativo

Diante deste contexto, a primeira e principal pergunta de pesquisa que esta tese almeja responder é:

- RQ1: A incorporação do conhecimento relacionado com informações musicais preditivas, tais como sequências típicas de acordes e estruturas musicais cíclicas, pode melhorar o desempenho de algoritmos de transcrição de acordes?

Este tipo de conhecimento não vem sendo, ou vem sendo muito marginalmente utilizado nos algoritmos atuais, que focam, sobretudo, em analisar localmente o sinal de áudio para determinar a que acorde correspondem, com uma ou outra variação. Diante disto e de nossa experiência anterior no uso destas informações musicais contextuais, definimos o objetivo principal deste trabalho:

- **Objetivo Principal**: Verificar a hipótese de que o conhecimento de informações musicais de caráter preditivo (sequências típicas de acordes e estruturas musicais cíclicas) pode melhorar o desempenho dos algoritmos de transcrição de acordes.

Em caso de uma verificação positiva, nosso trabalho representaria uma contribuição importante na área de MIR, abrindo novas perspectivas de melhora para os algoritmos atuais.

Porém, conseguir utilizar esta natureza de informações musicais preditivas para tentar melhorar um processo de transcrição de acordes não é uma tarefa simples. Sem o conhecimento dos acordes corretos de uma canção em transcrição, uma previsão baseada na identificação de sequências típicas de acordes e na presença de estruturas cíclicas teria que realizar suas análises a partir do uso de classificações de acordes previamente realizadas por um sistema qualquer de transcrição, cuja atuação, por si só já não seria confiável. Além disso, mesmo estando diante do aparente início da execução de uma sequência típica de acordes, por exemplo, extrapolá-la propondo a previsão do restante de sua execução com os acordes que comporiam a sequência típica, não seria garantia de certeza de acertos

devido a natural dúvida inerente a qualquer processo de previsão. Por isso, o uso deste tipo de informações musicais preditivas não fornece nenhuma garantia de melhora de desempenho de eventuais transcrições.

Por tudo isso, da primeira pergunta desta pesquisa (RQ1) desencadearam-se dois outros questionamentos:

- RQ2: Existe uma maneira de capturar adequadamente tais informações musicais contextuais de caráter preditivo?
- RQ3: Existe uma maneira de incorporar tais informações em algoritmos de transcrição de acordes?

Estes novos questionamentos definiram os objetivos específicos desta tese:

- **Primeiro objetivo específico**: Propor uma forma de capturar e representar as informações musicais contextuais de caráter preditivo;
- **Segundo objetivo específico**: Sugerir uma maneira de incorporá-las nos algoritmos de transcrição de acordes.

É conveniente destacar que não temos a pretensão de propor nem a “melhor” maneira de capturar/representar tais informações musicais preditivas, nem a melhor maneira de incorporá-las nos algoritmos atuais. Nosso foco permanece na primeira pergunta de pesquisa e objetivo principal, ou seja, em verificar se estas informações, pouco ou não levadas em consideração nos algoritmos atuais, poderiam melhorar o desempenho de um processo de transcrição de acordes. Porém, para fazer tal verificação será preciso que mostremos que existe pelo menos uma forma de capturar tais informações e pelo menos uma forma de incorporá-las nas técnicas de transcrição atuais (objetivos específicos). Caso a hipótese de que a utilização destas informações contextuais musicais para melhorar o desempenho de transcrições de acordes se confirme como verdadeira, entendemos que pesquisas futuras na área poderão melhorar a forma de capturar e incorporar estas informações, incluindo formas específicas para cada técnica de transcrição.

Esta redução de escopo tem motivação dupla. Primeiro, dada a complexidade do tema “extração de informação musical de sinais de áudio”, pelo tempo limitado de um trabalho de tese, existiria o risco de não alcançarmos os objetivos almejados.

Segundo, a fim de maximizar a nossa principal contribuição, que é demostrar que as informações musicais contextuais de caráter preditivo contribuem para processos de transcrição de acordes, buscamos e encontramos uma forma de incorporação de tais informações que fosse independente da técnica de transcrição de acordes. Ela é, talvez, não otimizada para cada algoritmo, mas estimulará mais amplamente novas pesquisas na linha de nossa contribuição.

1.3. Abordagem

O desenvolvimento de um sistema de transcrição de acordes implica na resolução de vários problemas tratados em várias etapas, algumas delas imprescindíveis para o processo. Entre estas etapas, destacamos aquela responsável pela análise espectral do sinal de áudio, onde costuma-se tentar extrair informações que possam facilitar a identificação de frequências que se aproximem das frequências dos acordes musicais (FUJISHIMA, 1999). Outras etapas do processo de transcrição de acordes podem envolver a escolha e aplicação de filtros para os sinais de áudio (NI *et al.*, 2011), a identificação de tonalidades das canções (UEDA *et al.*, 2010), a detecção do andamento (beats) (HAAS, 2012), a segmentação do áudio (SHEH; ELLIS, 2003), entre outras.

Sabendo que os principais algoritmos em estado da arte executam algumas destas etapas com bastante competência, percebemos que, apesar de estarmos cientes de que o conhecimento de informações musicais de caráter preditivo está praticamente inexplorado por estes mesmos algoritmos, não seria sensato criar um novo algoritmo do zero, ou seja, criar um algoritmo que reimplementasse todas as várias tarefas necessárias para a realização de um processo de transcrição de acordes, ignorando os avanços já alcançados pelos trabalhos em estado da arte relacionados com o tema.

Por este motivo, optamos por tentar responder RQ1 através do desenvolvimento de um algoritmo capaz de pós-processar as saídas de sistemas genéricos de transcrição de acordes, acrescentando o conhecimento de contexto relacionado com informações musicais preditivas, sem perder os avanços já alcançados pelos sistemas atuais de transcrição. Este algoritmo de pós-processamento receberia como entrada as saídas dos sistemas de transcrição, que são: (1) os acordes já transcritos e que ocorrem antes do acorde-alvo a ser

transcrito, e (2) a sugestão de qual é o candidato à transcrição do acorde-alvo correspondente ao trecho de áudio em questão. Em seguida, utilizando as informações musicais preditivas, este sistema de pós-processamento indicaria eventuais correções a serem realizadas na sugestão de acorde-alvo feita pelo algoritmo de transcrição no qual o pós-processamento estivesse sendo aplicado. Estas correções seriam indicadas pelo sistema de pós-processamento apenas quando o mesmo tivesse um percentual de certeza relevante a cerca das mesmas.

Este modelo de pós-processamento nos conduziria mais facilmente à resposta para RQ1 porque permitiria a sua validação através de um teste simples similar a um teste A/B. Neste teste seria possível a realização da comparação dos resultados de um processo de transcrição de acordes realizado por um sistema genérico sem a utilização, e com a utilização das informações musicais preditivas. Além disso, num modelo como este seria possível garantir a independência do sistema de pós-processamento em relação ao sistema de transcrição de acordes, o que eliminaria a necessidade de acesso aos códigos fontes deste último.

Com esta proposta, utilizando o corpus público de canções dos Beatles comumente utilizado no MIREX e baseando-nos em nossas pesquisas anteriores (CUNHA; RAMALHO, 1999), foi desenvolvido para o sistema de pós-processamento um módulo de identificação de sequências típicas de acordes baseado em uma rede neural do tipo MLP-*backpropagation*. Esta rede foi devidamente treinada e preparada para receber como entrada as transcrições de acordes realizadas por um sistema genérico de transcrição e prever ou sugerir qual deveria ser o próximo acorde a ser executado. Além disso, no mesmo sistema de pós-processamento, a fim de tentar identificar a presença de estruturas cíclicas, foi feita a adaptação do analisador de sequências repetidas (*Sequence Tracker*) desenvolvido em nosso trabalho anterior (CUNHA, 1999).

Para avaliar o sistema de pós-processamento foram feitos testes com a utilização dos resultados das transcrições realizadas por três sistemas diferentes: o primeiro deles, capaz de realizar transcrições em alto nível, o segundo, desenvolvido por nós, e capaz de realizar transcrições com resultados medianos, e o terceiro deles, com baixo desempenho em suas transcrições.

Com os experimentos e simulações realizadas, conseguimos verificar que é possível incorporar o conhecimento relativo às informações musicais de caráter preditivo a um processo de transcrição de acordes, e assim melhorar o seu

desempenho. Este objetivo foi alcançado com o uso de nosso sistema de pós-processamento utilizando a rede neural como algoritmo de previsão de acordes, e sem utilizar o módulo de identificação de estruturas cíclicas, que em sua adaptação para o cenário atual, não se mostrou efetivo na tarefa de melhorar o processo de transcrição de acordes. Com o pós-processador configurado desta forma, foram verificadas melhoras nos desempenhos dos processos de transcrição com os três algoritmos de transcrição testados.

Após todas as avaliações e testes, podemos afirmar que nossas perguntas de pesquisa puderam ser respondidas e que todos os objetivos definidos foram alcançados. Com relação a RQ2, verificamos que com o uso de uma rede neural do tipo MLP foi possível adquirir o conhecimento relacionado com as informações de sequências típicas de acordes.

Já com relação a RQ3, foi possível incorporar o conhecimento relacionado com as sequências típicas de acordes devido à forma de atuação definida para o nosso modelo de pós-processamento. Na sua atuação, o sistema de pós-processamento considerou que as previsões da rede neural só deveriam prevalecer sobre as transcrições originais caso houvesse um percentual relevante de certeza a cerca da corretude das mesmas.

Por fim, o resultado final obtido em nossas avaliações respondeu também satisfatoriamente à principal pergunta de pesquisa RQ1, já que conseguimos demonstrar que as informações musicais de caráter preditivo, especificamente aquelas relacionadas com as sequências típicas de acordes, são capazes de melhorar o desempenho de sistemas de transcrição em geral.

1.4. Estrutura do Documento

Nos próximos capítulos abordaremos:

No capítulo 2, os fundamentos teóricos que embasam a linguagem em torno do contexto do problema;

No capítulo 3 serão analisados os vários aspectos do problema de transcrição de acordes;

No capítulo 4 serão detalhadas as relações existentes entre as informações musicais de caráter preditivo e um processo de transcrição de acordes.

No capítulo 5 será abordada grande parte dos trabalhos científicos ligados ao problema ao longo dos últimos 40 anos e serão indicados os trabalhos em estado da arte;

No capítulo 6 será dada uma visão geral de nosso modelo integrado de pós-processamento envolvendo previsão e transcrição de acordes.

No capítulo 7 será abordado o módulo de previsão de acordes de nosso modelo de pós-processamento.

No capítulo 8 descreveremos o transcritor de acorde que desenvolvemos para a realização dos testes com o pós-processador.

No capítulo 9 será abordado o módulo decisor de nosso modelo de pós-processamento.

No Capítulo 10 serão detalhadas as avaliações e os resultados alcançados pelo nosso modelo

No Capítulo 11 faremos nossas últimas reflexões, conclusões e indicaremos os trabalhos futuros.

2. Fundamentação Teórica

Neste capítulo serão abordados alguns fundamentos importantes para a melhor compreensão desta tese.

2.1. Fundamentos Musicais

Nesta seção serão detalhados os conceitos musicais necessários à compreensão deste trabalho.

2.1.1. Notas Musicais

No nosso contexto, embora esta informação não seja estritamente verdade, vamos assumir que as notas musicais são as menores unidades sonoras que compõem uma canção. Existem em número de doze e são representadas pelos seguintes nomes:

- Dó (também representado pela letra C)
- Dó sustenido ou Ré bemol (Dó# ou Réb – notas enarmônicas, pois têm sons iguais na escala temperada1)
- Ré (também representado pela letra D)
- Ré sustenido ou Mi bemol (Ré# ou Mib – notas enarmônicas, pois têm sons iguais na escala temperada)
- Mi ou Fá bemol (Mi ou Fáb – notas enarmônicas, pois têm sons iguais na escala temperada – O Mi é também representado pela letra E)
- Fá (também representado pela letra F)
- Fá sustenido ou Solb (Fá# ou Solb – notas enarmônicas, pois têm sons iguais na escala temperada)
- Sol (também representado pela letra G)
- Sol sustenido ou Lá bemol (Sol# ou Láb – notas enarmônicas, pois têm sons iguais na escala temperada)

[1] Escala musical ocidental contendo oito notas com cinco intervalos de tons e dois intervalos de semitons entre as mesmas. Ao todo ela compreende doze semitons que correspondem às doze notas musicais

- Lá (também representado pela letra A)
- Lá sustenido ou Si bemol (Lá# ou Sib – notas enarmônicas, pois têm sons iguais na escala temperada)
- Si ou Dó bemol (Si ou Dób – notas enarmônicas, pois têm sons iguais na escala temperada – O Si é também representado pela letra B)

Os símbolos “#” (sustenido) e “b” (bemol) elevam ou diminuem, respectivamente, em meio-tom (ver próximo item) o som de uma nota musical. Um Ré bemol pode então ser representado como “Ré b” ou “Db”, assim como um Sol sustenido pode ser representado por “Sol #” ou “G#”.

2.1.2. Tons e semiton

Em termos de teoria musical, um semitom é o menor intervalo sonoro possível de ser analisado entre duas notas musicais. Um tom corresponde ao intervalo de dois semitons.

2.1.3. Melodia

A *Melodia* é uma sequência “qualquer” de notas musicais.

2.1.4. Intervalos

Um intervalo é a distância entre duas notas musicais e ele tem sua medida normalmente calculada em quantidade de tons e semitons. Em geral, os intervalos são classificados pelo seu *nome* e pelo seu *tipo*, sendo o nome identificado pelos termos “segunda”, “terça”, “quarta”, “quinta”, “sexta”, “sétima”, “nona”, “décima”, “décima primeira”, “décima terceira”, e o tipo identificado por “maior”, “menor”, “aumentado”, “diminuto” e “justo”.

Intervalo	Número de Tons e semitons	Nome	Tipo	Exemplo de Nomenclatura
Dó – Réb	1 semitom	Segunda	Menor	2ª Menor
Dó – Ré	1 Tom	Segunda	Maior	2ª Maior
Dó – Ré#	1 Tom e 1 semitom	Segunda	Aumentada	2ª Aumentada
Dó – Mib	1 Tom e 1 semitom	Terça	Menor	3ª Menor
Dó – Mi	2 Tons	Terça	Maior	3ª Maior
Dó – Fáb	2 Tons	Quarta	Diminuta	4ª Diminuta
Dó – Fá	2 Tons e 1 semitom	Quarta	Justa	4ª Justa
Dó – Fá#	3 Tons	Quarta	Aumentada	4ª Aumentada
Dó – Solb	3 Tons	Quinta	Diminuta	5ª Diminuta
Dó – Sol	3 Tons e 1 semitom	Quinta	Justa	5ª Justa
Dó – Sol#	4 Tons	Quinta	Aumentada	5ª Aumentada
Dó – Láb	4 Tons	Sexta	Menor	6ª Menor
Dó – Lá	4 Tons e 1 semitom	Sexta	Maior	6ª Maior
Dó – Lá#	5 Tons	Sexta	Aumentada	6ª Aumentada
Dó – Sib	5 Tons	Sétima	Menor	7ª Menor
Dó – Si	5 Tons e 1 semitom	Sétima	Maior	7ª Maior

Quadro 2.1 - Representação dos intervalos mais usuais tomando como base a nota de Dó

O significado exato destes termos não precisa ser detalhado no contexto deste trabalho. Caso o leitor deseje maiores detalhes existe extensa bibliografia sobre teoria musical que pode explicar melhor este conceito (PRIOLLI, 1977). No Quadro 2.1 mostramos todas as formas mais usuais de intervalos existentes entre notas musicais instanciados para o tom de “Dó maior”.

2.1.5. Acordes Musicais

Em termos absolutos, um *acorde musical* é definido pela junção superposta de mais de uma nota musical. Entretanto, sob um ponto de vista de análise harmônica voltada para o contexto em que atuaremos neste trabalho, podemos visualizar um acorde como a junção de dois elementos: a sua *tônica* e a sua *classe* ou *tipo*. A tônica dá nome ao acorde e representa a nota fundamental do mesmo. Por exemplo, no acorde de “Sol maior” a tônica é a nota “Sol”, no acorde de “Si bemol menor” a tônica é a nota “Si bemol”.

Já a Classe ou Tipo de um acorde tem sua definição baseada nas notas que, em conjunto com a tônica, vão compor o acorde. Como exemplo de classes podemos citar “m7”, “m6”, “7M”, “m7(b9)”. Sem precisar entrar em detalhes sobre o significado de cada uma destas expressões que representam classes de acordes, é importante que fique claro apenas que as mesmas vão especificar diferentes combinações de notas musicais que terão que ser executadas em conjunto com a

nota que representa a tônica a fim de que um determinado acorde musical possa ser considerado desta ou daquela classe. No Quadro 2.2 visualizamos alguns exemplos de notações de representações de acordes e as notas que os compõem. Estas notações gráficas que representam cada acorde são chamadas de cifras.

Cifras do Acorde	Notas do Acorde
Cm7	Dó, Mib, Sol, Sib
Dm7(b9)	Ré, Mib, Fá, Lá, Dó
G7(13)	Sol, Si, Ré, Mi, Fá
A7M	Lá, Dó#, Mi, Sol#

Quadro 2.2 - Exemplos de Cifras de acordes com as suas respectivas notas musicais

Na canção ocidental, é comum encontrarmos situações em que uma mesma canção seja representada com grades de acordes diferentes, porém não necessariamente erradas. Isto é possível porque em várias situações dentro de uma harmonização de uma canção, mais de um tipo de acorde pode ser utilizado sem que a canção perca a sua identidade. Esta é a ideia por trás do conceito de acordes substitutos, que, como o próprio nome indica, podem substituir os acordes considerados “originais” de uma canção sem que a mesma passe a ser considerada como outra canção. Em linhas gerais, o que permite que um acorde seja considerado como substituto de outro acorde é a proximidade sonora gerada pela execução simultânea das suas notas e a manutenção da função do acorde original na canção quando o mesmo é trocado pelo seu substituto (CHEDIAK, 1986).

Um exemplo de acorde substituto é o Am (Lá menor) em relação ao C (Dó Maior). O primeiro deles é formado pelas notas musicais Lá, Dó e Mi, e o segundo é formado pelas notas Dó, Mi e Sol. O que permite que o acorde Am seja considerado como substituto de acorde C é a proximidade sonora gerada pela execução simultânea das três notas de ambos (apenas uma nota de diferença), e o fato de que a sonoridade do Am, também chamado de acorde relativo menor em relação ao C, é capaz de soar funcionalmente dentro do contexto das canções de uma forma muito similar ao acorde original de C.

Em alguns casos, porém, acordes simplesmente similares não podem ser considerados como substitutos entre si por terem sonoridades funcionalmente diferentes, podendo induzir, inclusive, a melodias e harmonias distintas. Isto ocorre, por exemplo, com os acordes C e Em (Mi menor), que são compostos pelas notas Dó, Mi e Sol, e Mi, Sol e Si, respectivamente. Eles possuem sonoridades similares,

porém soando funcionalmente de uma forma bem diferente dentro do contexto da harmonia das canções, e por isso não são considerados substitutos entre si.

A análise da diferenciação entre os acordes é importante para o contexto do problema que estamos lidando neste trabalho porque no mundo real, uma transcrição de acordes fazendo uso, por exemplo, de acordes substitutos através da troca de um acorde maior por seu relativo menor, pode ser considerada muito bem sucedida, já que estes “erros” são bastante aceitáveis. Em algumas situações, inclusive, estes “erros” podem ser o resultado da intenção propositada do músico, que pode ter o intuito de variar ou alterar deliberadamente a harmonia de uma canção, objetivando, por exemplo, a criação de um novo arranjo musical para a mesma.

Diante deste cenário, espera-se que num sistema automatizado de transcrição de acordes, estes aspectos devam ser considerados para que uma transcrição aceitável não seja erradamente considerada como uma transcrição pobre. Erros grosseiros devem ser diferenciados dos erros aceitáveis.

2.1.6. Compassos

Ao representar simbolicamente a estrutura harmônica de uma canção, todos os seus acordes são agrupados em blocos que têm uma duração específica. Cada um destes blocos é chamado de *Compasso* (Figura 2.1).

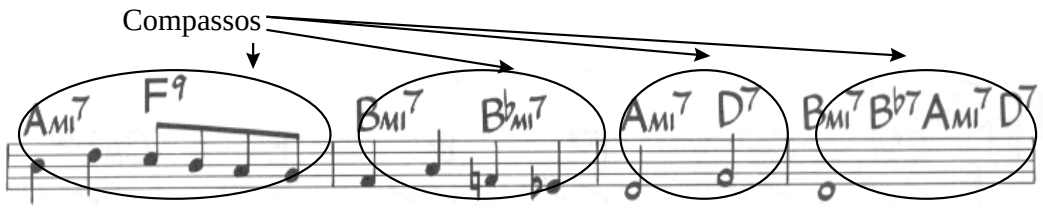


Figura 2.1 - Compassos (Trecho extraído da canção Darn That Dream – Fonte: (Bauer, 1988)

Todas as canções têm as suas estruturas harmônicas divididas entre melodia e acordes. A melodia é a parte “cantada” da canção, e os acordes são os complementos harmônicos que acompanham a melodia dando-lhe suporte e salientando as suas características (Figura 2.2).

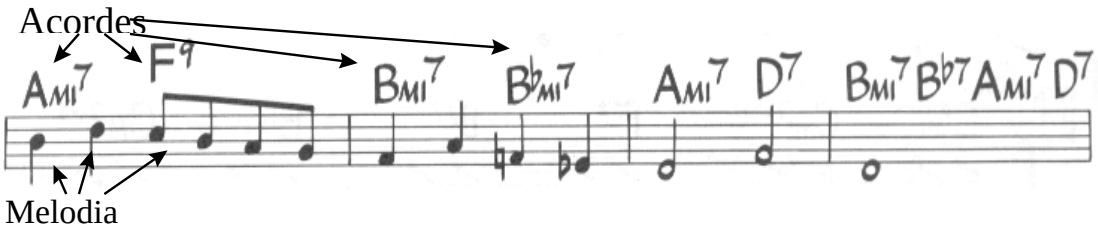


Figura 2.2 - Trecho extraído da canção Darn That Dream – Fonte: (Bauer, 1988)

2.1.7. Grade de Acordes

A *grade de acordes* é a representação gráfica dos acordes de uma canção e existe basicamente em três formatos. O primeiro deles é através da própria partitura da canção, onde encontramos, além dos acordes, toda a melodia da canção (Figura 2.3).

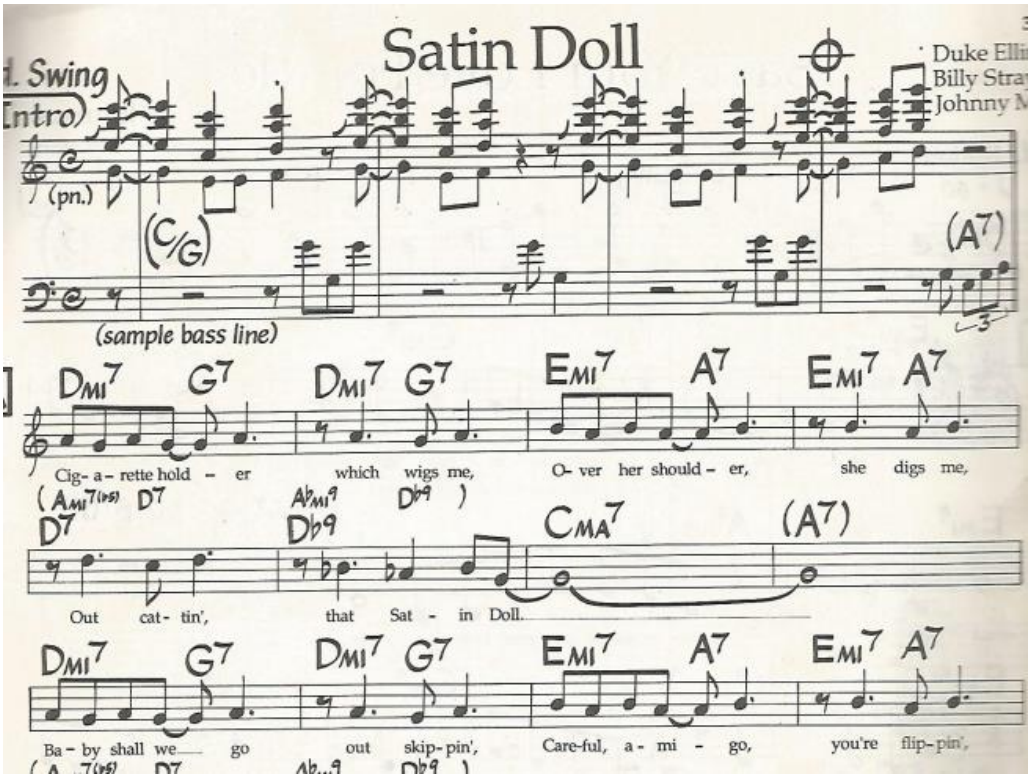


Figura 2.3 - Primeira representação de parte da grade de acordes - Canção "Satin Doll" – Fonte: (Bauer, 1988)

A outra forma é mais compacta. Nela, toda a estrutura dos acordes da canção é apresentada como em uma tabela onde cada compasso é representado por uma

célula. Neste formato, as informações sobre melodia não são mostradas (Figura 2.4).

Fa7M	Fa7M	G7b5	G7b5
G min 7	C 7	Am7b5	D 7b9
Gm7	A7b9	D 7	D7b9

Figura 2.4 - Segunda Representação de acordes da canção - Fonte: (Bauer, 1988)

A terceira forma é a mais simples, e nela os acordes são apresentados separados por barras delimitadoras dos compassos da canção (Figura 2.5).

F7M | F7M | G7b5 | G7b5 |
Gm7 | C7 | Am7b5 | D7b9 |
Gm7 | A7b9 | D7 | D7b9 |

Figura 2.5 - Terceira Representação da grade de acordes uma canção - Fonte: (Bauer, 1988)

2.1.8. Tonalidade

Simplificadamente, e para atender às necessidades de compreensão de nosso trabalho, podemos afirmar que Tonalidade é o conceito que define que notas musicais deverão ser prioritariamente utilizadas numa canção, tanto na sua melodia como na constituição de seus acordes. Excluindo o universo das músicas atonais (sem tonalidade definida), a tonalidade é normalmente uma propriedade de cada canção, que é identificada por dois termos: uma das doze notas musicais, e a palavra “maior” ou “menor”.

A tonalidade é uma informação importantíssima para a compreensão da estrutura harmônica e lógica de cada canção, podendo contribuir muito para um processo de transcrição de acordes.

2.1.9. Campo Harmônico

Mais uma vez de uma forma simplificada e reduzida, podemos afirmar que o campo harmônico de uma canção é definido pelo conjunto de acordes que têm em

sua constituição as notas musicais que devem ser executadas prioritariamente em uma dada tonalidade. Isso significa que para definirmos o campo harmônico de uma canção, precisamos definir a sua tonalidade. Apenas para melhorar ilustrar, no Quadro 2.3 identificamos de forma genérica como Graus os tipos de acordes que podem acontecer em uma tonalidade maior. Estes tipos são definidos exatamente a partir das notas musicais que fazem parte do campo harmônico de uma tonalidade maior. No mesmo quadro exemplificamos nas tonalidades de Dó e Ré maior os respectivos graus instanciados nos acordes que fazem parte do campo harmônico de cada uma destas tonalidades.

Graus	I	IIm	IIIm	IV	V	VIm	VIIm5-
Dó maior	C	Dm	Em	F	G	Am	Bm5-
Ré maior	D	Em	F#m	G	A	Bm	C#m5-

Quadro 2.3 – Tipos de acordes esperados em uma tonalidade maior, com exemplos e Dó maior e Ré Maior

2.1.10. Informações Musicais de Caráter Preditivo: Sequências Típicas ou Comuns de Acordes e Estruturas Cíclicas

No universo da música, é muito comum encontrarmos sequências de acordes que se repetem em diferentes canções, e até mesmo dentro de uma mesma canção. São muitas as canções que, quando executadas na tonalidade de “C” (Dó maior), por exemplo, apresentam a sequência de acordes “Dm - G7 - C” (Ré menor – Sol com sétima – Dó maior), ou a sequência “C – Am – Dm7 – G7” (Dó maior – Lá menor - Ré menor com sétima – Sol com sétima), entre outras. Sequências como estas são recorrentes e formam um conhecimento que é adquirido por músicos experientes através da prática. Elas são de tal maneira comuns que, nos livros de harmonia funcional (CHEDIAK, 1986), (FOX; WEISSMAN, 2013), recebem nomes genéricos independentes da tonalidade, como IIm-V-I, IIm-SubV-I, IV-V7-I, VI-IIm-V7-I, etc. Nas Figuras 2.6, 2.7 e 2.8 são indicados trechos de algumas canções com as respectivas análises funcionais indicando a presença de algumas destas sequências de acordes.

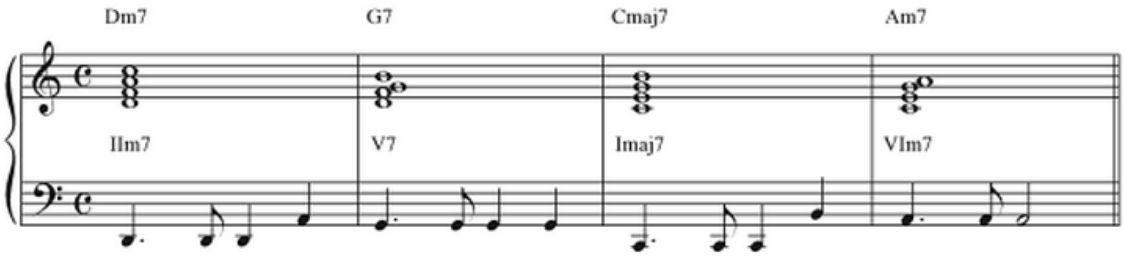


Figura 2.6 - IIm-V7-I - "Even the Nights are Bettes" – Air Supply (Fonte: Fox, et al., 2013)

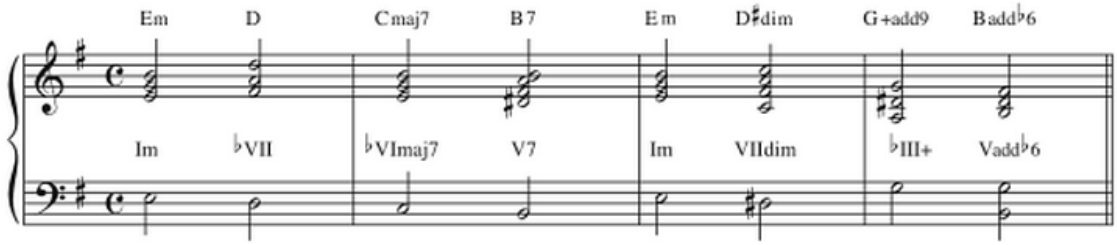


Figure 2.7 –Im-bVII-bVI-V7 – "Fifty ways To Leave Your Lover" – Paul Simon (Fonte: Fox, et al., 2013)

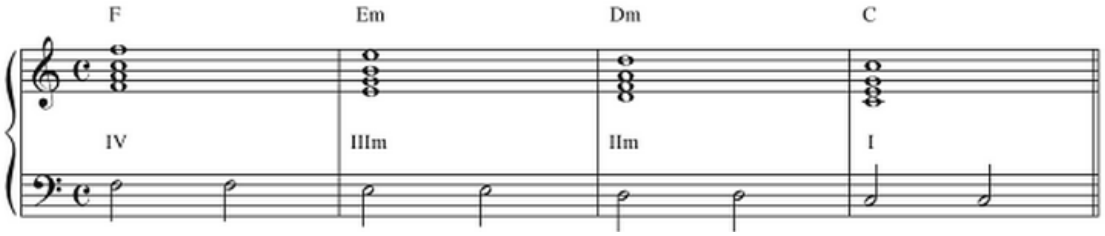


Figure 2.8 - IV-IIIm-IIm-I – Canção "You´re My Heart" – Rod Stwart (Fonte Fox, et al., 2013)

Iremos aprofundar mais tarde o papel destas sequencias na predição de acordes, mas é importante entender desde já que, embora não haja um modelo que defina quando uma ou outra destas sequências deverá ser utilizada em uma canção, músicos experientes conhecem estas sequências e as usam, entre outras, para antecipar o que vai acontecer em uma canção. De fato, uma das grandes vantagens que um músico tem em conhecer tais sequências é a de que, ao reconhecer o início de uma delas, ele sabe quais são os acordes mais prováveis que vão se seguir. Por exemplo, ao identificar um Dm7 seguido de um G7, o músico sabe que provavelmente está diante de uma sequência IIm-V7-I, e que portanto, provavelmente o próximo acorde é um C. Esta antecipação ajuda no

acompanhamento e na improvisação, especialmente de canções não conhecidas pelo músico (CHEDIAK, 1986).

Além de sequências recorrentes de acordes, bons músicos, e até não músicos, também conseguem identificar outro tipo de estrutura presente dentro de grande parte das canções populares: refrães e estrofes, que aqui referenciamos como estruturas cíclicas. Estas estruturas definem repetições de blocos de acordes que compõem a canção como um todo. Identificar o início e término destas estruturas diminui o esforço de um trabalho de transcrição de acordes de toda a canção, já que grande parte destes blocos de acordes se repete quase que de forma integral. Na Figura 2.9, temos um exemplo da canção *Satin Doll* composta por seções do tipo A e B que se repetem ao longo da execução da canção.

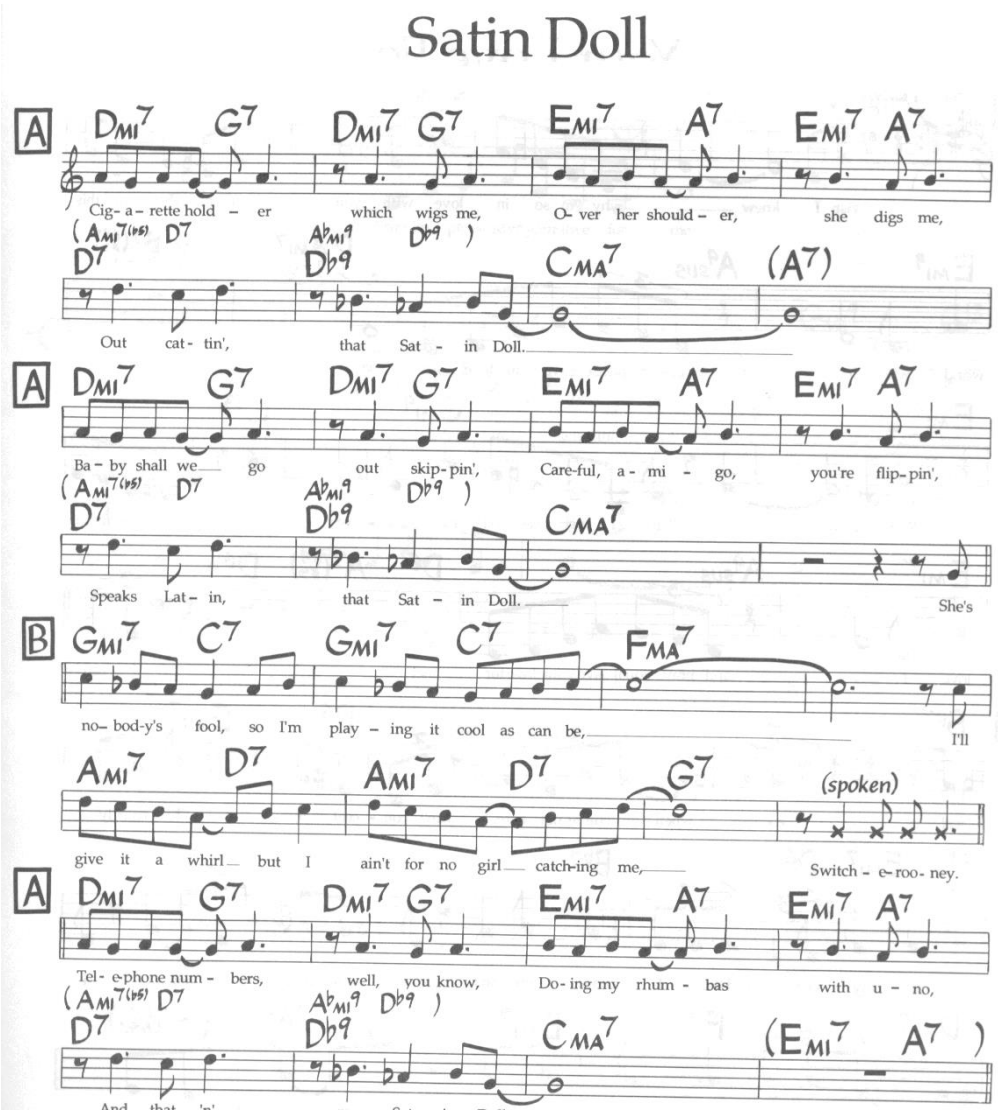


Figura 2.9 - Seções na canção Satin Doll (Fonte: Bauer, 1988)

Tanto as sequências típicas de acordes como as estruturas cíclicas das canções poderão ser referenciadas nesta tese como informações musicais preditivas ou de caráter preditivo.

2.1.11. Andamento (Beat)

Também de uma forma simplificada podemos definir um *beat* como a “batida” de uma canção. É a marcação rítmica que costuma ser utilizada para a identificação do ritmo de cada canção, normalmente referenciada pelo termo Andamento.

No contexto do nosso trabalho, a correta identificação do andamento de uma canção enriquece bastante um processo de transcrição de acordes porque dificilmente uma canção possui mais de um acorde entre duas “batidas” ou *beats*. Em geral, como consequência de uma boa identificação do andamento de uma canção, pode-se garantir um melhor processo de segmentação do áudio, o que significa uma mais eficiente separação entre os limites de cada acorde em execução numa dada canção.

2.2. Rede Neural MLP - *Backpropagation*

A Rede Neural de Perceptrons Multicamadas (MLP-Multilayer Perceptron) é de fundamental importância para os resultados alcançados neste trabalho de tese. Baseadas nos perceptrons (MINSKY; PAPERT, 1969), modelos de neurônios criados à imagem do modelo de neurônio biológico, as redes MLP (*MultiLayer-Perceptron*) (RUMELHART; HINTON; WILLIAMS, 1986) surgiram renovando as pesquisas que envolviam redes neurais e que se encontravam, até o final dos anos oitenta, bastante desprezadas devido a pouca utilidade dos modelos iniciais baseados em simples perceptrons. As redes MLP fazem parte do grupo de redes mais utilizadas para as mais diversas aplicações (HAYKIN, 1994).

O MLP é uma rede *feedforward*, ou seja, que recebe uma entrada e a passa adiante camada a camada até a camada de saída. No contexto em que será utilizada neste trabalho, ela passará por processos de aprendizagem supervisionada, utilizando o algoritmo clássico de aprendizagem ou de adaptação dos pesos de seus neurônios, conhecido como *backpropagation* (RUMELHART; HINTON; WILLIAMS, 1986).

O princípio básico deste algoritmo gira em torno da ideia de que os neurônios devem aprender a partir dos seus próprios erros. Erros estes, que devem ser minimizados ao máximo para que a rede tenha a melhor resposta e mais próxima da realidade. Este processo é realizado através da apresentação de conjuntos de padrões de entrada bastante representativos das classes a serem aprendidas. O objetivo é que a rede adapte os seus pesos assimilando as características dos padrões que lhes são apresentados. Nesta fase, cada entrada é conduzida pela rede, camada a camada até a camada de saída, quando, então, a saída atual da rede é comparada com a saída desejada para aquela determinada classe (aprendizado supervisionado). Se houver alguma diferença entre as duas, o erro gerado por esta diferença é retro propagado camada a camada a partir da camada de saída até a primeira camada intermediária da rede (já que a primeira camada é o próprio padrão de entrada). Este erro é usado para realizar correções nos pesos de cada neurônio proporcionais aos erros cometidos por eles. O processo é repetido até que a saída real da rede seja a própria saída desejada, ou algo próximo da mesma, para todos, ou quase todos os padrões usados no processo de treinamento. O processo de treinamento também pode ser interrompido após um determinado número de ciclos de correções dos pesos, o que vai ser determinado pela política de aprendizagem definida antes do início do processo. A Figura 2.10 representa um esquema genérico de uma rede MLP.

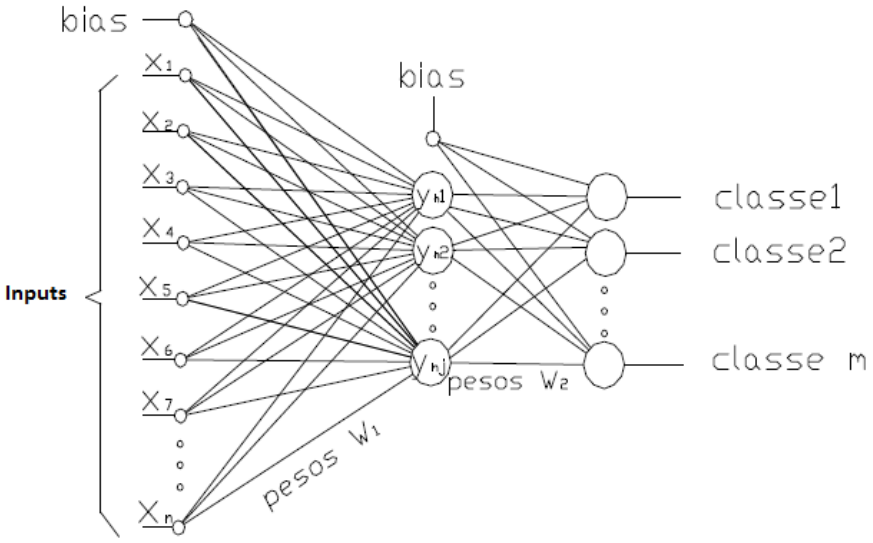


Figura 2.10 - Esquema de uma Rede Neural MLP

3. O Problema

O problema que este trabalho se propõe a ajudar a resolver é o de como fazer com que uma máquina consiga transcrever a estrutura harmônica ou a sequência de acordes de uma determinada canção a partir dos seus sinais de áudio. As dificuldades deste processo são inúmeras.

Desde os primeiros trabalhos relacionados com as tarefas de reconhecimento de notas musicais e transcrição de acordes a partir de sinais de áudio (PISZCZALSKI; GALLER, 1977; KASHINO, 1993; MARTIN, 1995; MOORER, 1975; FUJISHIMA, 1999), a base do processo de busca para a solução do problema sempre envolveu o cálculo de transformadas de Fourier e algumas de suas derivações (PAPADOPOULOS; PEETERS, 2008; CHO; WEISS; BELLO, 2010; HUMPHREY; BELLO, 2012; CHAFE; JAFFE, 1986) com o intuito de se obter os espectros das frequências dos sinais de áudio das canções (espectrogramas), processo normalmente chamado de Análise Espectral. Com esta análise, para cada instante de execução de cada canção, a ideia sempre passa pela tentativa de utilizar as informações das frequências de áudio obtidas para compor o cenário de busca pela identificação de um ou outro acorde.

Nas pesquisas iniciais, inclusive, a análise espectral seguida de uma análise de frequências executadas em cada instante de um determinado áudio foi utilizada como único meio de se identificar as notas e acordes musicais em cada um destes instantes. Os resultados práticos obtidos por este tipo de tentativa de solução se mostraram promissores com as primeiras implementações que realizavam testes com áudios contendo um único instrumento (piano e flauta, em geral) e com polifonia de até três sons em execução simultaneamente (PISZCZALSKI; GALLER, 1977; KASHINO, 1993; FUJISHIMA, 1999). Porém, quando este tipo de método passou a ser utilizado em sinais de áudio de canções reais, o desempenho das transcrições apresentou resultados pífios. O principal problema é que um sinal de áudio de uma canção real contém, além das informações harmônicas relacionadas com cada acorde em execução, muitas outras frequências que não têm relação direta com a harmonia das canções. Entre elas podemos destacar os sons tocados por outros instrumentos, as frequências percussivas e, em algumas situações, até aquelas frequências relacionadas com as melodias cantadas que não necessariamente

precisam estar em total comunhão harmônica com as frequências do acorde em execução em cada momento.

Este tipo de cenário empobrece muito os resultados obtidos nas tentativas de identificação de qualquer acorde em um dado instante de uma canção pela simples análise e comparação das frequências executadas neste instante em relação às frequências esperadas para acordes específicos, tornando a pura e direta análise espectral insuficiente para garantir a solução do problema de transcrição.

Porém, embora não seja suficiente, a realização da análise espectral em um sinal de áudio ainda é o melhor passo inicial para a extração de algumas características ou atributos (comumente chamados de descritores acústicos) relacionados com os acordes em execução. Estas características costumam fornecer os subsídios iniciais que vão alimentar um processo de transcrição mais rico, sendo utilizadas como parte importante e decisiva em vários passos do mesmo.

Embora na extensa lista de propostas de soluções para o problema de transcrição de acordes encontremos variadas alternativas de buscas pelo sucesso no processo, a grande maioria das soluções mais promissoras se propõe a atuar no problema resolvendo um conjunto de subproblemas mais ou menos recorrente. Na prática, a busca pela solução torna-se um trabalho que envolve a atuação em algumas etapas que misturam alguns subproblemas. Nas próximas seções, descreveremos cada um destes subproblemas recorrentes identificados na Figura 3.1.

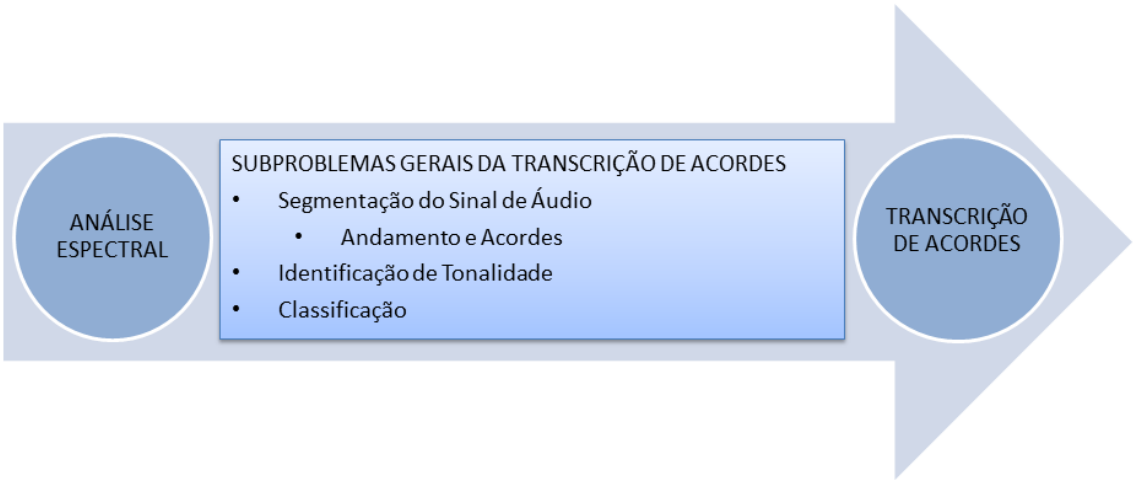


Figura 3.1 - Principais passos e subproblemas que colaboram para o sucesso da tarefa de transcrição de acordes

3.1. Segmentação de áudio e Detecção de Andamento

Um importante problema relacionado com um processo de transcrição de acordes é o da segmentação do áudio de uma canção por cada acorde executado (SHEH; ELLIS, 2003; CANNAM *et al.*, 2013). Ao analisar um sinal de áudio de uma dada canção a partir da extração de seu espectrograma, é muito difícil definir de forma exata onde começa e onde termina cada acorde (Figura 3.2).

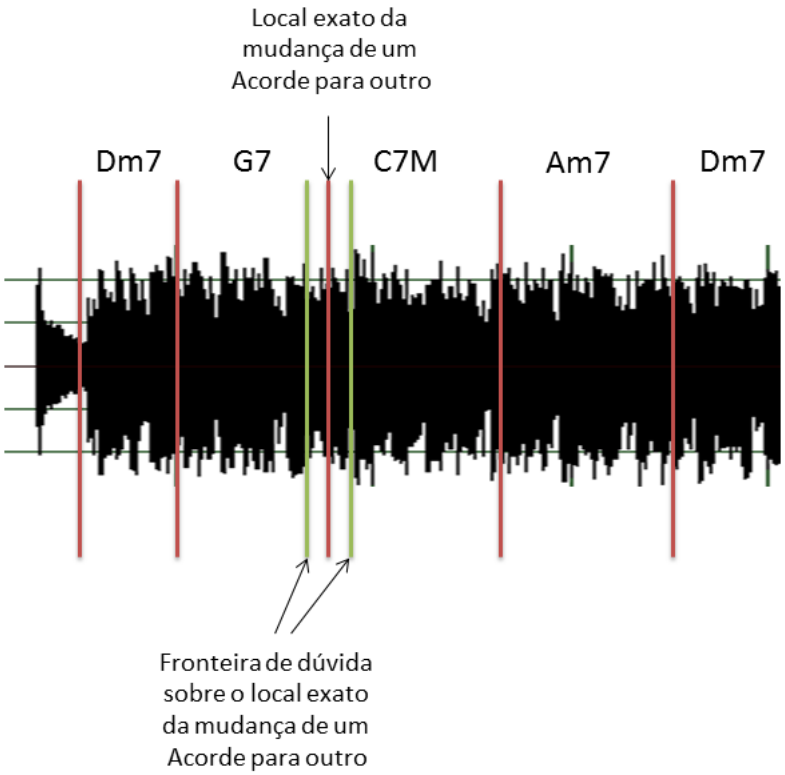


Figura 3.2 - Segmentação de uma canção por seus acordes

Em algumas situações comuns, inclusive, os acordes de uma canção não são necessariamente executados com todas as suas notas ao mesmo tempo, que é o que ocorre quando os mesmos são tocados de forma arpejada, ou seja, com cada uma de suas notas executadas em sequência, mas não ao mesmo tempo. Isso não só dificulta a identificação do acorde em execução, como dificulta muito a identificação automática do momento exato da transição de um acorde para outro. Na prática, mesmo com acordes não arpejados, o ruído proveniente da execução, normalmente ao mesmo tempo de muitos instrumentos musicais e percussivos,

inclusive de vozes humanas, tornam este um problema de difícil solução, mas relevante para o sucesso de uma transcrição de acordes.

Apesar da dificuldade, vários algoritmos alcançaram resultados expressivos nesta tarefa de segmentação, chegando até a 86% de sucesso (ELOWSSON; FRIBERG, 2013) na competição do MIREX do ano de 2013. Em geral, as melhores técnicas de segmentação tentam identificar os picos harmônicos que definem os momentos de mudanças de acordes.

Outra forma de lidar com o problema de segmentação é tentar detectar o ritmo de canções através da identificação do andamento (detecção da marcação rítmica da canção, mais conhecida como detecção dos *beats*) (PEETERS, 2007). Como é pouquíssimo provável que na canção ocidental ocorram acordes entre dois *beats* (entre duas marcações rítmicas), com a identificação dos momentos em que os mesmos ocorrem, torna-se mais fácil a detecção dos possíveis momentos de transição entre dois acordes e, consequentemente, a identificação dos limites dos segmentos que contém cada acorde (Figura 3.3).

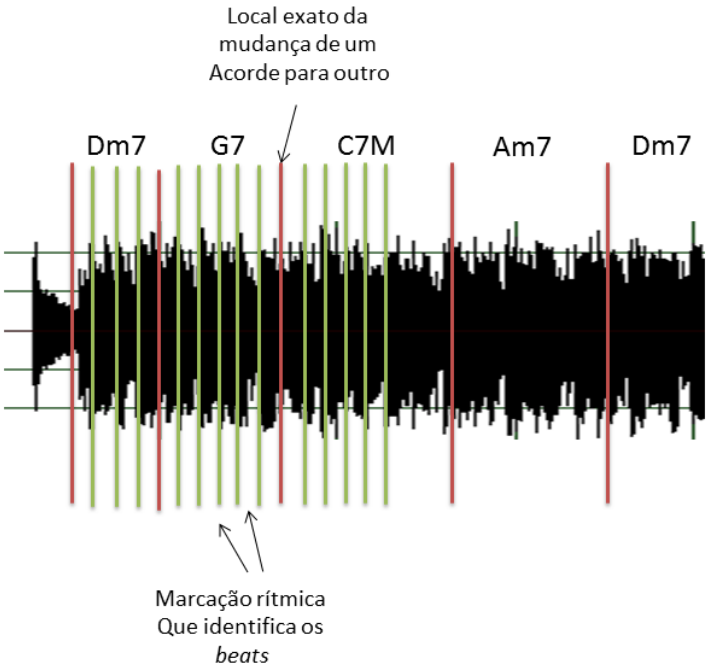


Figura 3.3 - Identificação de beats

Porém, a identificação correta do andamento, que também parte normalmente da análise do espectrograma do sinal de áudio, também exige a superação de dificuldades relacionadas, sobretudo, com a identificação de padrões repetidos de

sinais periódicos obscurecidos pelos ruídos presentes no áudio da canção em análise, da mesma forma que acontece num processo de segmentação. No MIREX, os melhores resultados alcançados nas últimas edições alcançaram em média 60% de sucesso nesta tarefa (PEETERS; CORNU, 2013). Este trabalho fez uso de técnicas de identificação de padrões repetidos através da formulação inversa de Viterbi

3.2. Identificação da Tonalidade

Como já descrito na seção 2.9, o conhecimento da tonalidade de uma canção enriquece e facilita um processo de transcrição de acordes, pelo menos quando realizado por músicos. Com esta informação, define-se um campo harmônico (ver seção 2.10) de possibilidades de acordes com probabilidades muito maiores de ocorrer do que aqueles que não pertencem ao campo. A eliminação de acordes candidatos à transcrição obviamente torna o processo de transcrição menos complexo. Porém, conseguir identificar a tonalidade de uma canção não é uma operação simples e exige o uso de algoritmos que também costumam partir da análise espectral do sinal de áudio, e como já foi discutido, este tipo de análise é muito dependente da complexidade do sinal e do nível de ruído presente na canção em análise, apresentando várias limitações.

Mesmo partindo do pressuposto de que a tonalidade de uma canção possa ser identificada, esta informação não trará nenhum benefício para um processo de transcrição de acordes se não for utilizada adequadamente. O maior problema a ser resolvido é o de como fazer uso dos acordes “viáveis” estabelecidos a partir do campo harmônico definido pela tonalidade identificada. Na prática, para que este campo harmônico tivesse utilidade, o processo de transcrição teria que definir listas de acordes candidatos e, pelo campo harmônico encontrado, este mesmo processo teria que ser capaz de excluir aqueles acordes que não pertencessem ao mesmo.

Neste sentido, alguns casos podem ser considerados simples, pois a “distância” de um eventual acorde candidato em relação a qualquer acorde do campo harmônico da tonalidade identificada pode torná-lo facilmente descartável. Em outros casos, porém, este descarte pode não estar tão claro. Entretanto, mesmo nos casos simples, onde teríamos um descarte de um acorde que estivesse claramente fora do campo harmônico da tonalidade encontrada, surgem outros

problemas. Qual seria a estratégia, por exemplo, para a escolha do acorde substituto daquele que seria descartado, caso a lista de candidatos não possuísse um opção óbvia? Nos casos onde não houvesse tanta certeza sobre o descarte, que estratégia teria que ser utilizada para que um acorde fosse considerado realmente descartável? Estas seriam perguntas cujas respostas precisariam ser encontradas para que um algoritmo que aplicasse o conhecimento teórico-musical em torno da tonalidade de uma canção conseguisse enriquecer o processo de transcrição de acordes como um todo.

3.3. Classificação

Seja qual for o rumo ou o conjunto de técnicas e algoritmos utilizados para atuar em um ou outro subproblema ligado ao processo de transcrição de acordes, a classificação correta dos mesmos é sempre o principal objetivo a ser alcançado, sendo esta normalmente a última etapa do processo como um todo. Ela é normalmente executada num contexto já enriquecido pela busca de solução de subproblemas já detalhados, como segmentação, detecção de andamento, identificação de tonalidade, entre outros menos comuns.

Reunindo e aliando todo o conhecimento já extraído previamente, o processo básico de classificação sempre passa pela divisão do sinal de áudio das canções em intervalos ou janelas sobre as quais um classificador buscará o enquadramento dos padrões de frequências presentes nas mesmas em alguma classe de acorde. Este enquadramento não é simples, em primeiro lugar porque não existem regras universais que definam padrões de acordes suficientemente representativos de tal forma que os mesmos possam ser considerados como padrões canônicos a serem utilizados como objetivos a serem alcançados numa classificação. Além disso, mesmo de posse de modelos de acordes, muitas vezes obtidos pela aplicação de processos probabilísticos, o processo de classificação ainda assim não é simples, dada, como já mencionada, a grande possibilidade de ocorrências de eventos sonoros e percussivos durante a execução de qualquer acorde em qualquer canção.

Com este cenário, a classificação num processo de transcrição tem sido muitas vezes tratada com o uso de aprendizagem de máquina, sobretudo com técnicas envolvendo Cadeias Escondidas de Markov ou HMM (RABINER, 1989), *Support Vector Machines* ou SVM (CORTES; VAPNIK, 1995) e até Redes Neurais

(BELLO; PICKENS, 2005), (CHENG; DIXON; MAUCH, 2013), (BURGOYNE; WILD; FUJINAGA, 2011). Com este tipo de abordagem, além de tudo, no processo de classificação terão que ser resolvidas todas as dificuldades inerentes à natureza da tarefa de aprendizagem: decisão e escolha da técnica ou algoritmo ideal; configuração dos parâmetros dos algoritmos selecionados; definição do corpus de treinamento e testes; identificação dos atributos ou descritores acústicos ideais para alimentar os algoritmos de aprendizagem e realização dos experimentos com análises de resultados.

Por fim, em qualquer situação, seja pelo uso de técnicas de casamento de padrões ou de aprendizagem, ao processo de classificação são aliadas as informações obtidas pelas tentativas de resolução dos subproblemas relacionados com o processo de transcrição. O contexto já conhecido e enriquecido através de informações de tonalidade, segmentações de acordes, identificação de andamento, participa e interfere muitas vezes decisivamente para o resultado final do processo de classificação e transcrição. Como e quando decidir pela interferência do conhecimento destas informações sobre o processo puro de classificação define outro nível de dificuldade essencial para o sucesso de nossa tarefa.

3.4. Conclusão

A busca pela solução do problema de transcrição de acordes a partir de sinais de áudio é a busca pela solução de uma série de subproblemas intrinsecamente relacionados. Desde a análise espectral, com suas muitas formas, passando por todas as dificuldades inerentes aos processos de segmentação e detecção de andamento, identificação de tonalidade e escolha da melhor técnica de classificação, a tarefa de transcrição de acordes se apresenta como um problema complexo e sem solução definida.

4. Informações Musicais Preditivas e Transcrição de Acordes

No capítulo 2, na seção 2.1.10 apresentamos dois tipos de informações musicais de caráter preditivo, a saber, as sequências típicas de acordes e as estruturas cíclicas. Mostramos que na prática musical estas informações desempenham um papel importante ajudando o músico a prever que novo acorde será executado em uma canção, conhecimento este que pode ser útil em várias situações para o mesmo.

No entanto, apesar da prática musical dar pistas, seria importante descobrir evidências empíricas sólidas de que, de fato estas informações poderiam ser úteis. Em outras palavras, seria preciso saber se é possível escrever algoritmos capazes de:

- Capturar estas informações musicais contextuais de caráter preditivo
- Utilizar estas informações em uma tarefa de previsão de acordes

Este capítulo, baseado principalmente em trabalhos que fizemos no passado, fornecerá as evidências de que, tanto é possível capturar, quanto utilizar estas informações na transcrição de acordes.

4.1. Previsão de acordes

Tentar prever os acordes de uma canção significa tentar identificar qual será o próximo acorde de uma canção em andamento, dado o conhecimento dos acordes que já foram executados até então. Esta não é uma tarefa simples, e sem regras claras que a definam, já que mesmo sabendo da existência das sequências típicas de acordes (ver seção 2.1.10), não existem regras que possam ser aplicadas de forma generalizada a ponto de que seja possível identificar sempre e com certeza que uma sequência de acordes em curso numa canção, mesmo que tenha o seu início igual ao de uma sequência típica, que ela seja de fato esta sequência. Músicos experientes têm bons desempenhos nesta tarefa, mas podem falhar em canções harmonicamente menos convencionais. Na Figura 4.1, temos um exemplo

do que significa a tarefa de prever acordes: dados os acordes Dm7, G7 e Dm7, qual deverá ser o próximo a ser executado?

Em outras palavras, a tarefa de tentar prever este tipo de sequência de acordes, mesmo para um músico, requer que alguns dos acordes da mesma sejam previamente executados para que, de posse desta suposta introdução da sequência, ele possa supô-la de forma completa. Da mesma forma, para um sistema se tornar capaz de identificar a presença de uma destas sequências de acordes, parte da mesma já teria que ter sido previamente identificada (janela de acordes), para que o mesmo pudesse tentar "casá-la" com o padrão de alguma das sequências recorrentes de acordes.

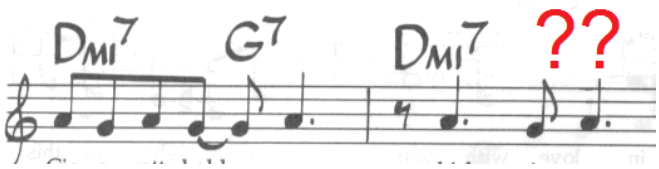


Figura 4.1 – Dada uma sequência de acordes, qual deverá se no próximo a ser executado?

4.2. Sequências Típicas e Previsão de Acordes

Assumindo que a constante presença de sequências típicas de acordes nas canções ocidentais poderia constituir um conhecimento a ser aprendido por uma máquina, em pesquisas anteriores nossas (CUNHA; RAMALHO, 1999) conseguimos demostrar que através de aprendizagem de máquina é possível, de fato, identificar suas ocorrências com relativo sucesso.

Nossa proposta foi a de utilizar um algoritmo de aprendizagem que recebesse como entrada janelas de acordes já executados, e que fosse capaz de aprender de forma supervisionada a prever qual deveria ser o próximo acorde a ser executado. Vários algoritmos e técnicas foram comparadas e a de melhor desempenho foi uma rede neural do tipo MLP com o algoritmo *backpropagation* (RUMELHART; HINTON; WILLIAMS, 1986). Utilizando um corpus de 58 canções de jazz extraídas do *New Royal Book* (BAUER, 1988) - das quais 40 foram usadas como conjunto de treinamento da rede neural – foi possível definir uma rede MLP com desempenho de acertos de 87% na tarefa de previsão de acordes sobre o conjunto de canções de testes (18 canções).

Para alcançar os resultados foram feitas várias simulações com janelas de acordes de tamanhos diferentes, e com o fornecimento como entrada da rede de vários tipos de atributos de cada acorde com o objetivo de encontrar aqueles mais influentes e importantes para os resultados do processo de previsão. Foram feitas simulações apenas com os atributos tônica e tipo ou categoria de cada acorde, e com informações obtidas facilmente pela observação das mesmas nas partituras do corpus de canções de treinamento. Isso ocorreu com os testes realizados com atributos como o intervalo do acorde em relação ao acorde anterior ou posterior, a posição do acorde dentro da canção, a posição do mesmo dentro de cada compasso da canção, além da sua duração.

Após todas as simulações, o melhor resultado foi obtido com o fornecimento como entrada para a rede neural de uma janela de três acordes para que ela pudesse prever o acorde seguinte, sendo cada um deles composto pelos seguintes atributos: tônica, tipo do acorde, posição do acorde no compasso e duração do acorde. Com relação ao domínio de valores possíveis para cada um destes atributos, para as tônicas foram consideradas as doze notas musicais, e para o tipo do acorde, foram considerados seis tipos ou categorias: “maior”, “menor”, “dominante”, “menorTonico”, “Meio-Diminuto”, “diminuto”. No caso da posição de um acorde no compasso, como indicado na Figura 4.2, foram considerados quatro possíveis valores.

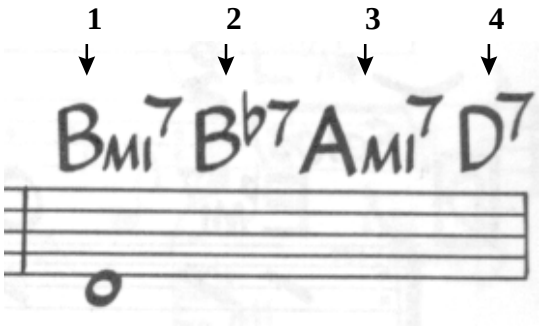


Figura 4.2 - Possíveis posições de um acorde no compasso – Fonte: (Bauer, 1988)

Já o atributo de duração de cada acorde foi valorado de acordo com a “quantidade de tempos” que cada acorde poderia durar. Como medida de simplificação, e também porque o nosso corpus de canções permitia esta aproximação, foi utilizada a ideia de que cada unidade de tempo seria representada

por um dos quatro tempos do compasso (no corpus de canções, cada compasso sempre possuía no máximo quatro tempos). Desta forma, um acorde com duração 16, por exemplo, ocuparia quatro compassos (Figura 4.3).

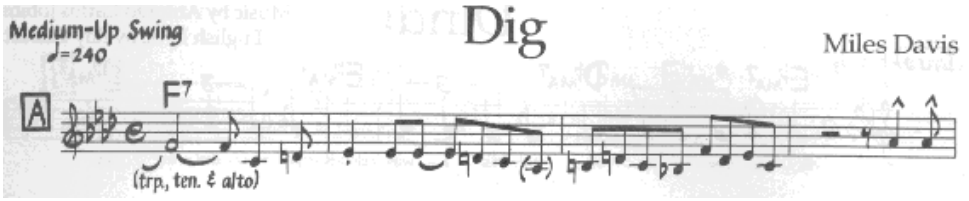


Figura 4.3 - Acorde "F7" com duração de 16 unidades de tempo (4 compassos) – Fonte: (Bauer, 1988)

4.3. Estruturas Cíclicas e Previsão de Acordes

Como já mencionado na seção 2.1.10, estruturas cíclicas como refrães, estrofes e seções são comuns na música ocidental e a identificação de suas ocorrências é importante para um processo de previsão da estrutura harmônica de acordes de uma canção. As principais propostas de soluções em estado da arte que tentam identificar estes tipos de estruturas atuam num nível de análise espectral do áudio pela busca de padrões similares dentro dos sinais das canções (CANNAM *et al.*, 2006). Este tipo de análise traz as mesmas dificuldades inerentes a todos os processos que partem do princípio da análise espectral (ver capítulo 3) e, além disso, as propostas desta natureza atuam na busca dos blocos dos sinais de áudio, ou melhor, das partes da canção potencialmente em repetição, detectando os seus limites, sem necessariamente identificar quais acordes tais partes contém.

Em pesquisas anteriores nossas (CUNHA; RAMALHO, 1999), por outro lado e partindo de uma análise simbólica, desenvolvemos um algoritmo capaz de identificar a ocorrência destas estruturas cíclicas. Neste trabalho foi proposto um algoritmo chamado de *Sequence Tracker*, que tinha o objetivo de observar as sequências de acordes em execução e tentar casá-las com sequências candidatas já executadas.

Na construção do algoritmo do *Sequence Tracker* foram consideradas várias premissas estabelecidas como regras, obtidas a partir da análise direta do corpus de canções. Entre estas regras destacamos a que exige o uso da informação da melodia da canção no processo de detecção de uma eventual repetição de uma sequência de acordes; a que indica que apenas após três compassos iguais é que o sistema pode começar a considerar a ocorrência de um bloco em repetição; e a que

define que blocos em repetição não são, necessariamente, completamente iguais, podendo apresentar diferenciações nos compassos de número 8n e 8n+1, onde n=1,2,3,....

A proposta de desenvolvimento do algoritmo do *Sequence Tracker* surgiu da ideia de um modelo maior que previa a montagem de uma estrutura híbrida de previsão de acordes envolvendo uma rede neural (a mesma descrita na seção 4.2) treinada para identificar sequências típicas de acordes, e o *Sequence Tracker*, responsável pela detecção de estruturas cíclicas maiores, tais como refrães e estrofes. Neste modelo, o módulo do *Sequence Tracker* teria sempre prioridade na tentativa de identificar uma possível estrutura cíclica em repetição. Quando isto não fosse possível, o controle da previsão seria repassado para a rede neural de previsão de acordes que tentaria prever qual o próximo acorde. No caso em que fosse identificada a presença de uma estrutura cíclica, o *Sequence Tracker* assumiria o controle do processo, e as previsões realizadas pela rede neural seriam suspensas enquanto o bloco em repetição identificado persistisse.

Com este modelo híbrido, num corpus de 30 canções, foram verificadas melhorias nos resultados finais de previsão quando comparados com os resultados obtidos com as previsões realizadas puramente pela rede neural. Os ganhos em rendimento variaram de 5% a 30%, a depender da complexidade da canção.

Tanto a identificação de sequências típicas de acordes, como a identificação de estruturas cíclicas definem o contexto da área de atuação de nossa proposta de solução a ser descrita nos próximos capítulos.

4.4. Previsão na Transcrição de Acordes

Um processo de transcrição de acordes, como descrito no Capítulo 3, envolve uma série de passos básicos e preliminares, cada qual com suas complexidades e dificuldades próprias. A busca pela solução deste problema tem passado pela tentativa de atuar bem em cada um destes passos e, eventualmente, em um ou outro passo que possa ser considerado relevante ou que contenha informações que indiquem a melhora no desempenho do processo de transcrição final.

Em nosso caso, este “outro passo” quer dizer “previsão de acordes”, seja pela identificação de sequências típicas, seja pela identificação de estruturas cíclicas de acordes. A extensa maioria dos trabalhos em estado da arte focam seus esforços, e

têm conseguido relevantes avanços, na análise espectral do sinal de áudio sem que informações musicais de caráter preditivo como as que temos nos referido sejam levadas em consideração em suas análises. Em cerca de trinta trabalhos publicados no ISMIR ou testados no MIREX nos últimos dez anos, nenhum tenta fazer uso deste tipo de informação. Informações estas que são notadamente importantes para um processo convencional de identificação da estrutura harmônica de uma canção.

A transcrição pura a partir da análise espectral é importante, mas dependendo de como seja feita, pode deixar de considerar o conhecimento por trás da lógica do encadeamento dos acordes de uma canção em execução. Por isso, parece-nos relevante que uma transcrição de acordes, além de todos os passos básicos trabalhados a partir da análise do sinal de áudio, também envolva um módulo que tente identificar a lógica por trás destas sequências de acordes, sejam as sequências típicas, sejam as estruturas cíclicas. Com o conhecimento das mesmas, o cenário da transcrição pode ser enriquecido e seu desempenho melhorado.

4.5. Conclusões e Limitações

Baseando-nos em resultados de pesquisas anteriores, percebe-se que com um cenário adequado, de posse das informações necessárias e suficientes, é possível capturar o conhecimento relacionado com informações musicais como sequências típicas e estruturas cíclicas com o objetivo de utilizá-lo para o desenvolvimento de um algoritmo de previsão de acordes.

Por outro lado, é preciso que sejam consideradas algumas limitações, principalmente se imaginarmos o uso do modelo híbrido descrito neste capítulo para atuar ou auxiliar a tarefa de transcrição de acordes executada diretamente a partir de sinais de áudio.

Primeiro, saindo do domínio simbólico original para o numérico (sinais de áudio), será difícil identificar com precisão informações como a posição do acorde dentro de cada compasso, ou até mesmo a duração dos mesmos acordes, informações esperadas para a atuação da rede neural de previsão de acordes.

Além disso, todos os nossos trabalhos anteriores partem da premissa de que os acordes já executados em uma canção, que constituem a entrada para o modelo de previsão proposto, por serem extraídos diretamente de partituras de canções, são sempre compostos por dados corretos e precisos, o que também não seria garantido

num processo onde estes acordes tivessem sido previamente extraídos a partir de sinais de áudio, e por um sistema de transcrição genérico. Neste sentido, num ambiente menos favorável, os resultados alcançados por este modelo híbrido não teriam garantia de mesmo nível de desempenho.

5. Estado da Arte

Ao longo dos anos a produção científica sobre a tarefa de transcrição de acordes de uma canção tem crescido bastante. Os primeiros trabalhos buscaram a transcrição das notas musicais simples, e evoluíram naturalmente para a transcrição da grade harmônica de acordes de canções. Durante esta evolução, percebe-se que a cada nova tentativa de solução para o problema de transcrição em si, seja de notas musicais, seja de acordes, quase sempre existe a necessidade da agregação de ideias já testadas e validadas em trabalhos anteriores, e por isso consideradas relevantes. O trabalho tem sido de constante construção de conhecimento, sem descarte do que já é bem sucedido. A fim de tentar deixar mais clara esta evolução nos modelos de solução propostos, neste capítulo vamos fazer uma análise cronológica dos trabalhos que influenciaram direta ou indiretamente os modelos atuais, até chegarmos ao momento atual de estado da arte da produção científica nesta área. No Anexo I deste trabalho, agrupamos em uma tabela todos estes trabalhos propostos nos últimos quarenta anos.

5.1. Período até os anos 90

O primeiro trabalho na área de transcrição de notas musicais em áudios polifônicos que encontramos registros (MOORER, 1977) tentou transcrever as notas musicais de fontes de áudio contendo duetos, ou composições a duas vozes executadas com instrumentos como guitarra e violinos. Foi um modelo proposto com grandes restrições, como o fato de que ele lidava com sinais de áudios polifônicos com a execução simultânea de apenas duas notas musicais (polifonia = 2). Além disso, havia uma limitação do intervalo de 24 semitons entre as alturas das notas que poderiam ser executadas.

Ainda na década de 70, foi proposto um trabalho para a detecção de notas musicais (PISZCZALSKI; GALLER, 1977) que realizava análise espectral através da transformada de Fourier – FFT (BRACEWELL, 2000) sobre fontes de áudios monofônicos (um único som por vez). Para garantir o bom funcionamento do projeto, as fontes de áudio deveriam ser de canções executadas por instrumentos com frequências fundamentais mais agudas, já que a FFT atua melhor nas mesmas. Nos experimentos foram utilizadas flautas como fontes sonoras.

Na década de 80, alguns pesquisadores já propunham modelos que tentavam separar notas musicais em canções polifônicas. Um destes modelos (CHAFE; JAFFE, 1986) propunha a realização da análise acústica considerando transformações espectrais com a transformada da constante Q (MONT-REYNAUD, 1985), a detecção de segmentos de canções através da análise do envelope dos sinais, e até a estimação de periodicidade e identificação de notas pela comparação das frequências das notas musicais fundamentais. O trabalho também tinha a restrição de utilizar polifonia máxima de duas notas musicais, e utilizou o piano como instrumento base para as fontes de áudio.

Em 1993 um grupo de pesquisadores da Universidade de Tóquio demonstrou um sistema de transcrição que empregou várias técnicas estudadas até então (KASHINO, 1993). Este trabalho foi o primeiro a se basear nos princípios utilizados pelos ouvidos dos seres humanos para a separação dos sinais sonoros, como fazem os músicos experientes que conseguem identificar os timbres e notas apenas ouvindo canções. Para tanto, eles utilizaram técnicas de síntese e decomposição de frequências de sinais simultâneos. Além disso, no trabalho também foram desenvolvidos algumas técnicas para identificação de tonalidade das canções, informação inegavelmente importante para uma tarefa de transcrição de acordes. Em 1995, o trabalho foi melhorado com o emprego de uma arquitetura *blackboard* [2] (HAYES-ROTH, 1985), particularmente modelada para realizar transcrição de notas musicais em áudios polifônicos com até três sons simultâneos (polifonia = 3) (KASHINO; KINOSHITA, 1995).

No final da década de 90, um importante trabalho foi publicado sobre transcrição específica de acordes a partir de sinais de áudio (FUJISHIMA, 1999). No algoritmo proposto, o ponto mais importante é a descrição de como se realiza a extração do *Pitch Class Profile (PCP)* de um instante de um áudio. O PCP é um vetor de características de dimensão doze (doze notas musicais) que, uma vez extraído segundo o algoritmo proposto, irá conter em cada uma de suas posições as intensidades das frequências de cada um dos doze semitons ou notas musicais de um determinado instante de um áudio, independentemente da altura ou oitava em que se encontrar a nota musical. O vetor PCP é também referenciado como vetor *chroma* (WAKEFIELD, 1999), embora este último termo costume ser utilizado de

[2] **Blackboard**: Aplicação de inteligência artificial onde um conhecimento comum, o "blackboard", é interativamente atualizado por um grupo diversificado de fontes de especialistas no conhecimento tratado.

uma forma mais genérica, já que engloba também outras formas de extração deste vetor de características que não seguem necessariamente a abordagem proposta no algoritmo de extração do PCP do trabalho do Fujishima. Existem, inclusive, trabalhos que comparam várias formas de cálculo e extração deste vetor testando-as em várias tarefas de MIR (CABRAL; BRIOT; PACHET, 2005). De uma forma mais genérica, o processo de cálculo e extração dos vetores *chroma* de toda uma canção costuma ser chamado de *Cromagrama*.

A ideia proposta por Fujishima foi a de utilizar pequenos intervalos de sinais de áudio extraídos de canções digitalizadas e calcular a Transformada Discreta de Fourier (DFT) sobre cada um destes intervalos. A cada DFT calculada, para chegar ao PCP de cada fragmento do áudio, o algoritmo utilizou o princípio logarítmico (BACKUS, 1977) que traduz o fato de que os saltos entre os valores absolutos das frequências fundamentais de notas musicais iguais, mas em alturas diferentes (oitavas diferentes), ocorrem sempre pela dobra no valor da frequência de uma oitava para a seguinte. Utilizando este principio, o algoritmo agrupa as frequências que se aproximam das frequências fundamentais de cada nota musical, classificando-as como pertencentes a uma ou outra nota musical. Aliando esta extração do PCP com a aplicação de algumas heurísticas, o sistema proposto por Fujishima tenta enquadrar as notas musicais mais frequentes no vetor PCP com as notas que representam padrões de acordes.

O trabalho se mostrou muito bem sucedido na tarefa de transcrição para sinais de áudio obtidos diretamente de um instrumento musical do tipo teclado executando acordes em tons puros com timbres como pianos, flautas e cordas. Na Figura 5.1 está indicado o esquema de funcionamento do sistema.

Apesar do trabalho de Fujishima ter realizado transcrições de acordes em ambientes bem controlados, o seu valor é inegável, inclusive sendo reconhecido até hoje, haja vista a quantidade de trabalhos que ainda fazem uso dos seus princípios para tentar alcançar sucesso na tarefa de transcrição de acordes.

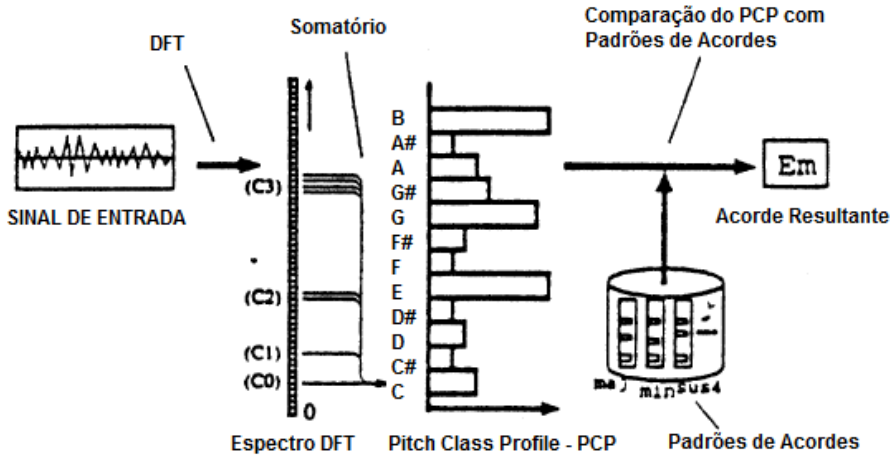


Figura 5.1 - Esquema do Sistema de Fujishima - Fonte: (Fujishima, 1999)

5.2. O uso de HMM para a tarefa de transcrição de acordes

Os trabalhos sobre transcrição de acordes em sinais de áudio de canções completas começaram a alcançar resultados mais relevantes durante a última década. Entretanto, embora alguns tenham alcançado esta relevância em resultados, temos sempre que considerar as condições em que eles foram alcançados. Como veremos adiante, questões como o universo de classes de acordes considerados, restrições a estilos musicais, corpus de canções usadas em testes, podem mudar consideravelmente os níveis de sucesso de muitas propostas promissoras.

Além disso, na ampla maioria dos trabalhos mais promissores e com melhores resultados, o modelo predominantemente usado é o de Cadeias Escondidas de Markov (HMM), conhecido por ser muito eficiente na tarefa de reconhecimento da linguagem falada (MANNING, 1999; RABINER, 1989), sendo por isso uma escolha indicada para a tarefa de transcrição de acordes, que segue, em linhas gerais, o mesmo paradigma. Pela sua importância na tarefa de transcrição de acordes, daremos um breve detalhamento sobre como funciona este método.

Na Figura 5.2 está indicado o esqueleto fundamental de um HMM do tipo utilizado na tarefa de transcrição.

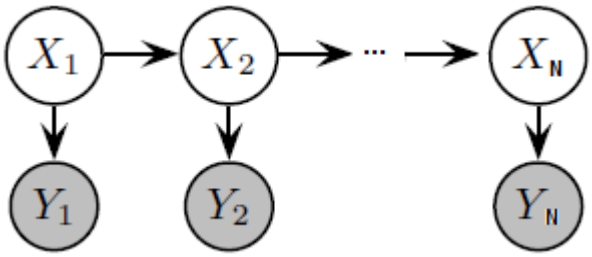


Figura 5.2 - Representação de um HMM. No contexto de transcrição de acordes, as variáveis de estado escondidas Xt representam uma sequência desconhecida de acordes, enquanto que as variáveis Yt representam as observações identificadas pelos vetores chroma – Fonte: (Murphy, 2002)

Nesta figura, as variáveis X_t, t ϵ { 1, ..., K } representam os estados ou acordes, sendo K portanto, o número de acordes considerados. A premissa básica dos modelos markovianos aplicados ao problema de transcrição, é a de que seus estados estão submetidos a seguinte propriedade:

$$P(X_t \mid X_{t-1}, X_{t-2},\ldots) = P(X_t \mid X_{t-1}) \qquad (5.1)$$

Esta expressão quer dizer que o estado ou acorde corrente depende apenas do estado ou acorde imediatamente anterior. Como discutiremos em nossa proposta, esta premissa do modelo do HMM vai servir de motivação para algumas de nossas hipóteses.

A ideia básica do HMM é a de propor um modelo probabilístico que, baseado em um processo de aprendizagem, consiga calcular as probabilidades de ocorrências de todos os estados ou acordes tomando como base uma dada observação, normalmente representada pelo vetor *chroma* extraído de um instante do áudio, e ao mesmo tempo calcular as probabilidades de transição entre todos os pares de estados ou acordes possíveis e considerados. Para se chegar a estas probabilidades, costuma-se utilizar um algoritmo interativo como o Baum-Welch, também conhecido como método EM-Expectation-Maximization, ou alguma técnica de gradiente (RABINER, 1989).

De posse destas probabilidades de ocorrência de um acorde, dada certa observação ou vetor *chroma*, e das probabilidades de transições entre todos os pares de acordes, faz-se uso do algoritmo Viterbi (FORNEY, 1973) capaz de, dada uma sequência de observações (vetores *chroma*), encontrar a sequência de estados ou acordes mais provável, ou melhor, com mais alta probabilidade de ocorrer. Sendo

assim, com o uso deste par HMM-Viterbi consegue-se definir modelos probabilísticos capazes de, dada uma sequência de vetores chroma, identificar a mais provável sequência de acordes, ou seja, a mais provável transcrição de acordes.

A principal limitação deste modelo, sobretudo se o imaginarmos aplicado ao próprio problema de transcrição de acordes, está no fato de que nas observações ou acordes sucessivos não é considerada a interdependência que pode existir em uma sequência de acordes, que sabemos que no âmbito da harmonia musical realmente existe devido às conhecidas sequências comuns de acordes (ver seção 2.1.10).

Porém, esta limitação pode ser suplantada pela riqueza do modelo que dá uma distribuição de probabilidades tomando como base as possíveis ocorrências de acordes, dadas as observações de vetores *chroma*, e uma distribuição de probabilidades de transições entre cada par de acordes. Além disso, o par HMM-Viterbi possui um forte embasamento matemático, devido principalmente à garantia de convergência dada pelo uso de um eficiente algoritmo de aprendizagem que automaticamente otimiza os seus parâmetros.

Como já adiantamos, muitos trabalhos fazem uso de HMM-Viterbi e na próxima seção vamos analisar aqueles propostos na década iniciada no ano 2000.

5.3. Década iniciada no ano 2000

Neste período, um dos primeiros trabalhos a ter destaque foi o que propôs um método de transcrição de acordes baseado no modelo humano de percepção (SU; JENG, 2001). Este modelo fez uso da transformada *Wavelet*[3] (PATHAK, 2009) em conjunto com uma rede neural do tipo SOM (self-organized map)[4] (KOHONEN, 1982), com o objetivo de simular, respectivamente, o ouvido e o cérebro humanos. Para a época, o trabalho se mostrou promissor, obtendo bons resultados mesmo em áudios com ruídos. O sistema foi treinado com 480 amostras de sons de 48 tipos diferentes de acordes. Os testes foram realizados com o quarto movimento da quinta sinfonia de Beethoven, alcançando resultados de 95% de sucesso na transcrição.

[3] **Wavelet**: Função capaz de decompor e descrever ou representar outra função (ou uma série de dados) originalmente descrita no domínio do tempo (ou outra ou outras várias variáveis independentes, como o espaço), de forma a permitir a análise desta outra função em diferentes escalas de frequência e tempo.

[4] **SOM**: Rede neural considerada um mapa auto-organizável e que é capaz de diminuir a dimensão de um grupo de dados mantendo a representação real em relação às propriedades relevantes dos vetores de entrada.

Em 2003, um trabalho se baseou no princípio de que vetores *chroma* teriam informação suficiente para garantir o sucesso de uma transcrição de acordes (SHEH; ELLIS, 2003). Foi então proposto um modelo para a execução desta tarefa que fez uso de HMM e Viterbi[5] para estimar acordes e os momentos de mudanças dos mesmos. Assumindo que os acordes em diferentes alturas deveriam ter perfis de *chroma* muito próximos, também foi proposta uma representação geral de cada acorde através de distribuições gaussianas. Utilizando sete tipos de acordes diferentes, o sistema proposto, em seu corpus de canções, conseguiu taxas de acerto de 76% na tarefa de segmentação, e 23% na taxa de reconhecimento de acordes. Talvez a baixa taxa de reconhecimento tenha se devido à pequena quantidade de canções no corpus de treinamento (apenas 20 canções dos Beatles), em comparação à grande quantidade de classes utilizadas na análise.

Em 2004, outro trabalho propôs um sistema que em seu corpus de testes (sete canções) chegou a obter 77% de acerto na tarefa de transcrição de acordes (YOSHIOKA *et al.*, 2004). O trabalho proposto tentou reconhecer os acordes detectando, ao mesmo tempo, o momento em que cada um deles é finalizado ou iniciado (segmentação). Para realizar esta tarefa foi utilizado um algoritmo detector de andamento ou beats (GOTO, 2003). Além disso, foram definidos padrões de vetores *chroma* para cada acorde através de distribuições gaussianas, que eram comparados com os vetores *chroma* das canções em teste para verificação de similaridades. Nos testes, sete canções foram utilizadas e todas elas estavam na tonalidade de dó maior. O sistema foi capaz de identificar até 48 tipos de acordes.

Em 2005, um trabalho merece um especial registro por ter proposto um modelo de representação simbólica para acordes (HARTE *et al.*, 2005) que mais tarde se tornou um padrão na comunidade de MIR (*Music Information Retrieval*), inclusive passando a ser utilizado também nos algoritmos submetidos para o MIREX - *Music Information Retrieval Evaluation eXchange* a partir do ano de 2008. Na competição, estes padrões e tipos de acordes passaram a ser aqueles esperados como saídas dos algoritmos de transcrição de acordes submetidos ao MIREX. No modelo de acordes proposto no trabalho, foi prevista uma gama de possibilidades de representações e combinações de acordes e seus tipos com um nível de riqueza e

[5] **Viterbi**: Algoritmo capaz de encontrar o melhor caminho de estados escondidos – chamado de Viterbi´s path – que resulta em uma sequência de eventos observados, especialmente no contexto de HMM´s.

detalhamento bastante expressivo. Este padrão de representação de acordes foi utilizado nesta tese e no Anexo II explanamos o mesmo em detalhes.

Continuando a nossa análise, no ano de 2006, foi proposto um sistema de transcrição de acordes utilizando HMM, mas que optou por uma forma diferente nos meios utilizados para a geração das informações dos acordes das canções que fariam parte do corpus de treinamento e testes (LEE; SLANEY, 2006). Neste caso, os autores do projeto do sistema fizeram uso de arquivos MIDI que já continham as informações dos acordes das canções. Com estes arquivos MIDI, eles geraram arquivos de áudio a partir de sintetizadores e utilizaram os mesmos para a extração dos vetores PCP para a alimentação do algoritmo de aprendizagem. A proposta deste modelo facilitou principalmente a resolução do problema, até então manual, de etiquetar o corpus das canções de treinamento e testes com os seus respectivos acordes. O sistema foi treinado utilizando 175 arquivos de quartetos de cordas de Haydn e foi testado com uma gravação do Prelúdio em Dó Maior de Bach. A transcrição foi feita tomando como base 36 classes de acordes e os testes alcançaram 93,35% de acertos na transcrição.

Neste mesmo ano, alguns outros trabalhos foram importantes para as pesquisas na tarefa de transcrição de acordes como um todo. Um deles tinha como objetivo identificar a tonalidade de uma canção através da análise do seu sinal de áudio. O trabalho em questão (PEETERS, 2006) se baseou no cálculo da DFT do sinal de áudio de uma dada canção, seguido da aplicação de uma função chamada de HPS – *Harmonic Peak Subtraction*, capaz de enfatizar a energia da frequência da tônica de um determinado instante de um sinal de áudio e enfraquecer a energia dos picos dos harmônicos da mesma. Após a aplicação desta função, seriam extraídos os vetores PCP para serem comparados com perfis de vetores PCP de todas as tonalidades. Aquele perfil que tivesse maior correlação média ao longo da análise do sinal (apenas os 20 primeiros segundos de cada canção eram analisados) indicaria a tonalidade. Este trabalho obteve taxa de acertos de 88,4% na identificação da tonalidade num corpus de 302 canções.

Em 2007, uma pesquisa utilizou uma abordagem diferente dos já tradicionais algoritmos baseados em HMM´s e Viterbi. Nesta época ainda não existiam corpus de canções oficiais, públicas e etiquetadas com acordes que pudessem ser utilizados pela comunidade de MIR, e neste trabalho (BURGOYNE *et al.*, 2007) foi utilizado um corpus de 20 canções dos Beatles num modelo baseado em *Conditional*

Random Fields (CRF)[7] (SUTTON; MCCALLUM, 2006) e em vetores *chroma* modelados como distribuições de gaussianas[8] e distribuições de Dirichlet[9] (BURGOYNE; SAUL, 2005). Uma das grandes contribuições deste trabalho foi a comparação dos resultados alcançados por este modelo com os resultados de modelos baseados em HMM-Viterbi. O trabalho demonstrou que as técnicas baseadas em CRF alcançaram em torno de 45% de sucesso em suas transcrições, e as técnicas baseadas em HMM alcançaram, no mesmo corpus de canções, em torno de 49% de sucesso de transcrição. Foram utilizadas 48 classes de acordes nos testes.

Outra pesquisa deste mesmo ano de 2007 fez uso de uma variação do vetor *chroma*, chamado de Centróide Tonal, em conjunto com HMM e Viterbi. Este modelo de vetor *chroma* foi proposto num trabalho de 2006, e baseou-se na teoria da Rede Harmônica ou *Tonnetz* [10] (HARTE; SANDLER; GASSER, 2006). O uso deste modelo de *chroma* permite a definição de uma função de detecção de mudança harmônica que tem o objetivo de identificar os momentos em que ocorrem mudanças de um acorde para outro de uma canção (em outras palavras, esta função executa a segmentação da canção com a separação entre os acordes da mesma). Além do uso do vetor *chroma* Centróide Tonal, este trabalho de 2007 também propôs um modelo que tenta identificar a tonalidade da canção aliando esta informação ao processo de detecção da sequência de acordes a ser transcrita (LEE; SLANEY, 2007). Utilizando um corpus de canções de dois discos dos Beatles (28 canções), 24 classes de acordes, o sistema implementado foi testado e alcançou uma média de acertos de 72.75% quando a informação da tonalidade não estava definida, e de 74.36% quando a tonalidade estava definida.

Neste mesmo período foi desenvolvido um sistema de transcrição que fez uso de um algoritmo de detecção de *beats* (tempo da canção), o *BeatRoot* (DIXON, 2006), para melhor identificar os momentos de mudanças de acordes (ZENZ; RAUBER, 2007). Além disso, para melhorar o desempenho da transcrição, o sistema

---

[7] **Conditional Random Fields (CRFs)**: Representam uma classe de métodos de modelagem estatistica frequentemente aplicada a reconhecimento de padrões e aprendizagem de máquina.

[8] **Distribuição Gaussiana:** é uma das mais importantes distribuições da estatística, conhecida também como Distribuição de Gauss ou Distribuição Normal.

[9] **Distribuição de Dirichlet**: Nome em homenagem à Johann Peter Gustav Lejeune Dirichlet, frequentemente representada por Dir(α), é uma distribuição discreta multivaridada com um parâmetro (vetorial) α não-negativo e real.

[10] **Tonnetz**: É um diagrama conceitual que representa o espaço tonal na toria musical.

também tentava detectar a tonalidade da canção, que permitiria a exclusão dos acordes candidatos, daqueles que não pertencessem à tonalidade. O sistema realizou transcrições de 36 classes de acordes e atingiu o percentual de 65% de sucesso em seu corpus de testes de 35 canções de vários estilos (canção popular, rock e canção clássica).

Até o ano de 2007, as análises dos desempenhos de algoritmos que se propunham a resolver o problema de transcrição de acordes apresentam um grande problema entre si: todos eles utilizavam corpus de canções e métricas de avaliação próprios, o que impedia a realização de qualquer análise comparativa com real embasamento científico. Analisando cada um deles, evidencia-se a diversidade de ambientes.

Alguns algoritmos, por exemplo, são testados com fontes de áudio com apenas um instrumento em execução, outros dentro de um estilo musical específico, e outros tratam, em suas transcrições de uma quantidade de classes, ou melhor, de uma quantidade de tipos de acordes menor do que a quantidade tratada por outros algoritmos. Questões como estas podem definir o que pode ser considerada uma proposta de solução relevante, ou ainda, se os resultados de uma solução podem realmente ser considerados superiores ou inferiores aos de outra.

A percepção destes aspectos é importante porque, em tese, algoritmos que, por exemplo, lidam com fontes de áudio tratadas e sem ruídos ou que tentam transcrever uma menor quantidade de classes, tendem a ter melhores resultados do que os que lidam com sinais de áudio sem tratamento ou do que aqueles que lidam com uma maior quantidade de classes de acordes. Eles tendem a ter melhores resultados, mas ao mesmo tempo, se aplicados em condições mais próximas de cenários reais, tendem a ter grandes quedas em seus desempenhos finais.

Apenas para demonstrar este fato, no Gráfico 5.1 montamos uma análise comparativa dos percentuais de sucesso na tarefa de transcrição obtidos por quatro algoritmos que foram testados sobre a mesma base de dados (em torno de 20 canções dos Beatles), em função do número de classes ou tipos de acordes tratados pelos mesmos. É claro que devido à falta de uma amostra comparativa maior, não podemos concluir nada de absoluto após analisarmos este gráfico. Porém, o mesmo parece indicar uma tendência que confirmaria aquilo que intuitivamente parece claro, que é a relação de sucesso na transcrição de acordes em função do menor número de classes a serem testadas.

Objetivando a minimização deste tipo de problema, a partir do ano de 2008, no congresso da ISMIR começaram a ocorrer competições científicas do MIREX na tarefa de transcrição de acordes. Os resultados dos principais sistemas de transcrição passaram a ter a oportunidade de ser avaliados lidando com a mesma quantidade de classes de acordes, sob mesmas métricas e sobre um mesmo corpus de canções, de forma que a análise comparativa entre seus resultados passou a ter um peso científico real.

Um dos primeiros trabalhos avaliados sob esta nova ótica fez uso de HMM e Viterbi com vetores PCP para detectar os acordes de canções, assim como vários trabalhos similares (UCHIYAMA *et al.*, 2008). O grande diferencial deste trabalho foi o uso de técnicas que suprimiam as frequências dos instrumentos percussivos através de algoritmos de separações harmônicas (MIYAMOTO; AL, 2008). O trabalho mostrou que os PCP´s extraídos de áudios com as informações harmônicas enfatizadas pela filtragem das frequências percussivas, resultam em melhor precisão na tarefa de transcrição de acordes. O sistema proposto foi submetido ao MIREX e alcançou a melhor taxa de acerto na tarefa de transcrição e acordes na competição daquele ano, com um percentual de 72% de sucesso na transcrição, num corpus de 176 canções dos Beatles e 180 classes de acordes.

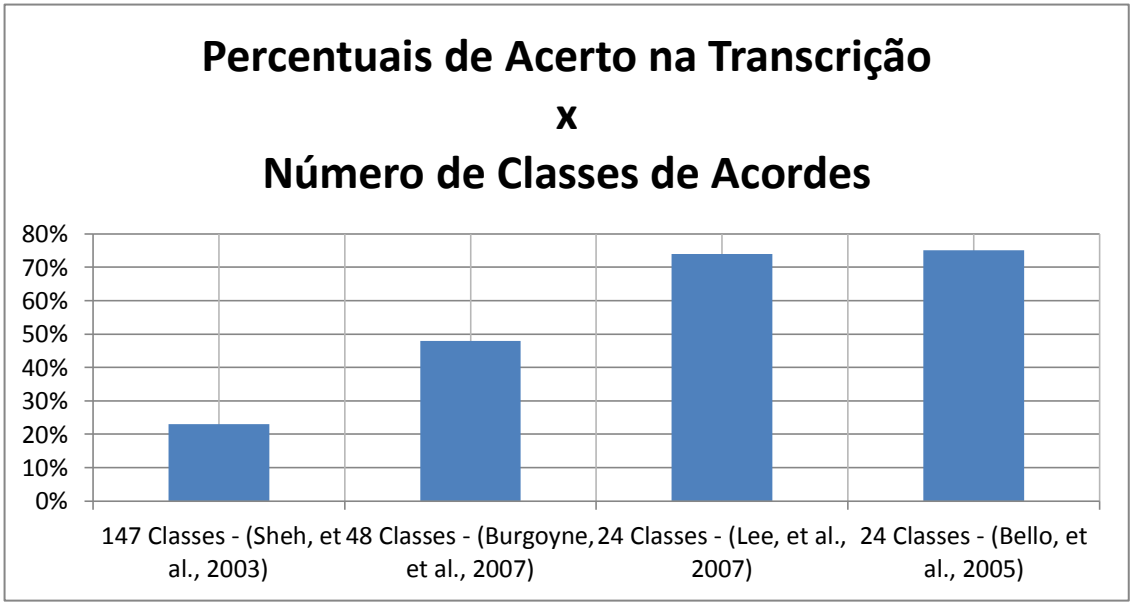


Gráfico 5.1 – Percentual de Acertos na Transcrição x Número de Classes de Acordes

No mesmo ano, outro trabalho fez uso de um algoritmo de detecção de *beats* (ELLIS; POLINER, 2007) e HMM-Viterbi (ELLIS, 2008) com o diferencial de propor a extração de dois vetores PCP, calculados após a aplicação de filtros que enfatizavam, no primeiro, frequências em torno de 400hz, realçando as frequências dos acordes, e no segundo, frequências em torno de 100hz com o objetivo de realçar as frequências da tônica do acorde, normalmente executada pelo instrumento contrabaixo. O sistema implementado também foi submetido ao MIREX e obteve percentuais de acerto de 66% no mesmo corpus de 176 canções dos Beatles e lidando com 180 classes de acordes.

Outro trabalho também submetido ao MIREX deste mesmo ano baseou-se nos mesmos princípios de alguns trabalhos anteriores (BELLO; PICKENS, 2005), utilizando HMM-Viterbi e vetores PCP modelados por gaussianas (WEIL; DURRIEU, 2008). O grande diferencial estava na ideia de que as frequências relativas à melodia da canção foram atenuadas com o objetivo de melhorar o processo de detecção de acordes. O trabalho alcançou o percentual de 62% de sucesso na tarefa de transcrição no MIREX no mesmo corpus de 176 canções dos Beatles e lidando com 180 classes de acordes.

Vários outros trabalhos foram publicados durante este ano, mas a grande maioria seguiu os mesmos princípios de utilização de HMM e Viterbi com a extração dos vetores PCP que, em alguns casos, sofreu algumas variações na sua dimensão e na sua forma de extração. Entre aqueles que foram publicados no MIREX tivemos dois que alcançaram 63% de acertos (PAPADOPOULOS; PEETERS, 2008), (KHADKEVICH; OMOLOGO, 2008), mais um com 59% de acertos (PAUWELS; VAREWYCK; MARTENS, 2008) e o menos expressivo alcançando 36% de acertos (ZHANG; LASH, 2008), todos atuando no mesmo corpus de 176 canções dos Beatles e lidando com as mesmas 180 classes de acordes.

Em 2009, também foi submetido ao MIREX uma pesquisa que utilizou um algoritmo que não se baseou em aprendizagem e que teve como grande diferencial a capacidade de detectar segmentos ou estruturas cíclicas como seções, versos ou estrofes de canções a fim de facilitar o seu processo de transcrição de acordes (MAUCH; NOLAND; DIXON, 2009). O método de detecção automática destes segmentos baseava-se em dois passos principais: encontrar sequências de vetores *chroma* aproximadamente repetidas e num algoritmo capaz de decidir qual das sequências de vetores *chroma* são de fato segmentos. Eram calculados os

coeficientes de correlação de Pearson[11] entre cada par de vetores *chroma* que juntos determinavam uma matriz de auto similaridade R de toda a canção. Nesta matriz de similaridade, linhas paralelas diagonais indicavam seções repetidas da canção e alguns filtros foram aplicados para eliminar pequenos desvios na matriz. Todo o processo também foi auxiliado pelo uso de um algoritmo de detecção de beats, de tal forma que na busca por segmentos repetidos nas diagonais da matriz de similaridades, foi assumido que cada possível segmento poderia ter no mínimo 12 beats e no máximo 128 beats. As análises indicaram que cada segmento era normalmente múltiplo de 4 beats. Utilizando um procedimento para a extração dos acordes da canção a partir de cromagramas calculados com o uso da identificação de beats, e facilitado pela aplicação de filtros que enfatizavam frequências graves e agudas, e ainda usando uma rede bayesiana, foram identificadas as mais prováveis sequências de acordes de toda a canção. O conhecimento relacionado com os segmentos repetidos ajudou no processo de transcrição através da junção de todos aqueles que coincidissem entre si, seguido do cálculo de um cromagrama médio entre os mesmos. A transcrição final passou a ser feita então a partir da extração dos acordes baseada neste cromagrama médio. O sistema gerado com esta proposta alcançou acertos de 71,2% na tarefa de transcrição e acordes, num corpus de 206 canções (Beatles, Queen e Zweieck) e 180 classes de acordes.

Outro trabalho que se destacou no MIREX de 2009 (OUDRE; GRENIER; FÉVOTTE, 2009) utilizou a ideia simples de tentar identificar os acordes em execução pela medida de proximidade com padrões de vetores PCP definidos previamente para cada acorde. Um dos diferenciais do trabalho foi o fato de que nestes padrões de vetores PCP dos acordes foram considerados alguns harmônicos de cada acorde. O sistema testado no corpus de 206 canções do MIREX e 180 classes de acordes alcançou resultados de 71,1% de acertos.

Weller e sua equipe (WELLER; ELLIS; JEBARA, 2009) utilizaram alguns princípios comuns à extração dos vetores PCP, só que ao invés de utilizar HMM e Viterbi, eles fizeram uso do SVMstruct (TSOCHANTARIDIS *et al.*, 2004), uma variação do algoritmo de *Support Vector Machine* (CORTES; VAPNIK, 1995). A performance do algoritmo em relação a mesma implementação utilizando HMM

[11] **Coeficiente de Correlação de Pearson: T**ambém chamado de "coeficiente de correlação produto-momento" ou simplesmente de "ρ de Pearson", este coeficiente mede o grau da correlação (e a direção dessa correlação - se positiva ou negativa) entre duas variáveis de escala intervalar ou de razão.

apresenta um considerável ganho, levando o mesmo a alcançar o melhor resultado em toda a competição do MIREX, com 74,2% de acertos no ano de 2009, no mesmo corpus de 206 canções (Beatles, Queen e Zweieck) e 180 classes de acordes.

Neste mesmo ano, também no MIREX, alguns outros trabalhos foram publicados, também com boas performances. Todos eles atuaram sobre as mesmas 206 canções e tentaram identificar as mesmas 180 classes de acordes. Uma das pesquisas utilizou HMM e Viterbi com algumas técnicas de separação de harmonia e frequências percussivas e alcançou 70,1% de acertos no grupo dos algoritmos que realizam treinamentos (REED *et al.*, 2009). Outro trabalho criou um modelo probabilístico que fez uso de uma forma mais robusta de extração do vetor PCP, partindo das frequências dos harmônicos e não das frequências fundamentais dos acordes (PAUWELS; VAREWYCK; MARTENS, 2009). O sistema desenvolvido alcançou 68,2% de sucesso na transcrição de acordes no corpus do MIREX.

Um dos trabalhos mais bem sucedidos do período de 2010 propôs uma nova forma de extração dos vetores PCP através da utilização de uma técnica para a resolução de problemas de mínimos quadrados não negativos (NNLS)[12] em conjunto com uma rede bayesiana dinâmica[13] (MAUCH; DIXON, 2010). Com esta técnica, os vetores PCP passaram a ser chamados de vetores NNLS e permitiram que o trabalho alcançasse 79% de sucesso na tarefa de transcrição de acordes quando avaliada no MIREX. Este foi o melhor resultado da competição do ano de 2010. Este trabalho também aliou à tarefa de transcrição de acordes a um processo de identificação dos *beats* para cada canção em avaliação. Neste ano, o MIREX passou a considerar um conjunto de 217 classes de acordes com o mesmo corpus de 206 canções (Beatles, Queen e Zweieck) utilizado no ano de 2009.

Outro trabalho que obteve bons resultados na edição do MIREX de 2010 envolveu uma sistemática avaliação das principais técnicas de transcrição de acordes testando diversas configurações de parâmetros (CHO; WEISS; BELLO, 2010). Na pesquisa foram realizados testes com vetores PCP extraídos através da transformada Q, com filtragem com ênfase nas frequências graves e com a

[12] **Método mínimos quadrados não negativos (NNLS - non-negative leas squares)**: Também referenciado como Quadrados Mínimos Ordinários (MQO) ou OLS (do inglês *Ordinary Least Squares*) é uma técnica de otimização matemática que procura encontrar o melhor ajuste para um conjunto de dados tentando minimizar a soma dos quadrados das diferenças entre o valor estimado e os dados observados (tais diferenças são chamadas resíduos).

[13] **Redes Bayesianas (RB)**: São modelos que codificam os relacionamentos probabilísticos entre as variáveis que definem um determinado domínio e que são utilizadas para representar processos probabilísticos e causais.

utilização de técnicas de casamento de padrões com padrões de vetores PCP binários e baseados em gaussianas. Por fim, no trabalho ainda foi utilizado HMM e Viterbi para finalização das avaliações. O algoritmo final submetido ao MIREX obteve 78% de sucesso na tarefa de transcrição de acordes, utilizando o mesmo corpus de 206 canções (Beatles, Queen e Zweieck) com as mesmas 217 classes de acordes.

Na competição do MIREX deste ano também participou um sistema que se baseou em SVMstruct (ELLIS; WELLER, 2010). Neste trabalho, foi proposto um algoritmo que fazia a extração de dois vetores *chroma*, o primeiro pré-tratado com um filtro que enfatizava as frequências entre 100hz e 1600hz, e o segundo pré-tratado com um filtro que enfatizava as frequências entre 25hz e 400hz. Para realizar o reconhecimento dos acordes propriamente ditos, o algoritmo fez uso de um classificador do tipo SVM e, quando avaliado no MIREX no mesmo corpus de canções daquele ano, o sistema alcançou o percentual de 77% de sucesso na transcrição de acordes.

Neste ano, outros trabalhos também foram submetidos ao MIREX e aplicados no mesmo corpus de canções e para o mesmo número de classes de acordes, todos utilizando técnicas sem grandes inovações, e alcançando resultados pouco menos expressivos. (UEDA *et al.*, 2010), (OUDRE; ; GRENIER, 2011), (ROCHER *et al.*, 2009), e (PAPADOPOULOS; PEETERS, 2010) alcançaram, respectivamente, 76%, 74%, 71% e 68% de acertos no processo de transcrição.

5.4. Período da década iniciada em 2011

Na década atual, tivemos os maiores avanços relativos aos desempenhos dos algoritmos de transcrição de acordes. Em 2011 o algoritmo que teve o melhor resultado no MIREX propôs um modelo que estimava simultaneamente o tom da canção, os acordes e as notas executadas pelo contrabaixo das canções em teste (NI *et al.*, 2011). O algoritmo, batizado de *Harmony Progression Analyser (HP*), fazia a extração dos vetores PCP após uma separação das informações harmônicas e percussivas feita com o algoritmo *Harmonic/Percussive Signal Separation-HPSS* (ONO *et al.*, 2008). Além disso, a ideia aplicada foi a de calcular dois vetores PCP, o primeiro calculado após a enfatização das frequências mais graves (55 Hz - 207.65 Hz) e o segundo com ênfase nas frequências mais agudas (220Hz - 1661.2Hz). A

proposta também fez uso de um algoritmo padrão para identificação de *beats* (ELLIS; POLINER, 2007) e os vetores PCP´s gerados foram tratados com HMM-Viterbi. Quando avaliado no MIREX em seu corpus composto por 206 canções e considerando a análise de até 217 classes de acordes, este sistema alcançou o impressionante percentual de 97% de acertos na transcrição de acordes.

Devido a este resultado quase perfeito, surgiram algumas reflexões sobre a forma como os testes no MIREX vinham sendo realizados. Até então, todo o corpus de canções utilizado para a validação dos algoritmos submetidos à competição era público e todos os proponentes poderiam testar seus sistemas livremente com estas canções, o que poderia gerar algoritmos com bons resultados, mas em alguns casos, algoritmos “viciados” neste corpus. Para evitar esta situação, as edições do MIREX a partir de 2012 passaram a utilizar um corpus para testes desconhecido dos proponentes de trabalhos. Provavelmente por isso, como veremos adiante, em 2012 este mesmo algoritmo sofreu uma queda significativa em seu desempenho no MIREX.

Outro trabalho de destaque do ano de 2011 propôs um método baseado na repetição de padrões de sequências de acordes (CHO; BELLO, 2011). Para analisar as sequências de acordes que poderiam estar em repetição foi utilizado um algoritmo de plotagem recorrente (MARWAN *et al.*, 2007) já utilizado em trabalhos que tentavam identificar canções cover (SERRA; SERRA; ANDRZEJAK, 2009) e similaridades estruturais em canções (BELLO, 2011). Similarmente ao método utilizado em (MAUCH; NOLAND; DIXON, 2009), a ideia foi a de encontrar padrões repetidos a partir de comparações de cromagramas. Porém, com a utilização de técnicas de suavização de desvios e ruídos, e de reforço de informações harmônicas nos frames de *chromas*, com a utilização do algoritmo de plotagem recorrente o sistema decidia que partes da canção deveriam ser consideradas como trechos em repetição. Neste sentido, não havia limites quanto ao tamanho destes trechos, de forma que eles podiam corresponder a apenas um acorde ou a segmentos maiores como seções, estrofes ou refrães. Desta forma, o sistema conseguia gerar muito mais informações repetidas para cada acorde, enriquecendo o conhecimento a cerca de suas detecções. Sobre os trechos considerados repetidos eram calculadas as médias dos cromagramas e sobre os mesmos eram feitas as transcrições de acordes. O algoritmo alcançou resultados de 80% de sucesso na transcrição de

acordes no MIREX no mesmo corpus e analisando o mesmo grupo de classes de acordes.

Em 2012, o mesmo grupo de pesquisadores que propôs o trabalho de melhor resultado no MIREX de 2011 (NI *et al.*, 2011) apresentou uma evolução do seu modelo, que passou a considerar aspectos relativos aos diferentes estilos musicais (NI *et al.*, 2012). Com esta estrutura, o trabalho propôs o uso de um HMM independente para cada gênero musical e, no momento da transcrição, alguns parâmetros globais decidiam pelo uso de um ou outro HMM. Como já relatado, neste ano o MIREX passou a utilizar dois corpus de canções para a validação dos algoritmos. Um primeiro deles, composto por 406 canções (Beatles, Queen, Zweieck e uma lista da Billboard de Ashley Burgoyne (BURGOYNE; WILD; FUJINAGA, 2011)) conhecidas e disponibilizadas para os autores dos trabalhos, e o segundo deles contendo 200 canções, também do conjunto da Billboard de Ashley Burgoyne, só que desconhecidas dos autores. Este trabalho alcançou 83% de sucesso na tarefa de transcrição no primeiro conjunto de testes, e 72% no segundo conjunto. Assim como nos anos anteriores, o número de classes de acordes analisadas pelos algoritmos submetidos ao MIREX foi de 217.

Outro trabalho que atingiu bons resultados no MIREX deste ano propôs uma mudança na forma como os vetores PCP deveriam ser gerados (KHADKEVICH; OMOLOGO, 2011), pela utilização de uma técnica chamada de ré-associação de Tempo e Frequência (KODERA; GENDRIN; VILLEDARY, 1978) sobre os espectrogramas resultantes dos cálculos das transformadas. Com a aplicação desta técnica, e com o uso de vetores PCP extraídos em duas faixas de frequências, aliados a um HMM para o processo de aprendizagem, o algoritmo alcançou 82% de sucesso no primeiro conjunto de testes do MIREX, e 70% no segundo conjunto.

Neste mesmo ano foi proposto um trabalho que utilizou HMM-Viterbi para a tarefa de transcrição de acordes, aliando a isso a tentativa de detecção da duração dos mesmos (CHEN *et al.*, 2012). No trabalho também foi definida outra abordagem para a definição dos vetores *chroma*, que passaram a ter dimensão 24, sendo as 12 primeiras utilizadas para representar as notas dos baixos de acordes invertidos, e as seguintes utilizadas para representar as notas dos acordes. Com este modelo o algoritmo alcançou 78% de acerto no primeiro conjunto de testes do MIREX e 66% no segundo conjunto.

Ainda em 2012, outra pesquisa (HAAS; MAGALHÃES; WIERING, 2012) propôs um modelo baseado no algoritmo do NNLS *Chroma* (MAUCH; DIXON, 2010), originalmente desenvolvido a partir do cálculo da transformada Q para extrair um vetor *chroma* após a aplicação de um filtro de enfatização de frequências agudas, e outro vetor após a aplicação de um filtro que enfatizava frequências graves do sinal de áudio. Além disso, o modelo utilizava técnicas de detecção de andamento (*beats*) (MUSIC; COMPUTING, 2013), detecção de tonalidades globais e possíveis modulações (mudanças de tonalidades) e identificação de campos harmônicos. Com o acréscimo e uso de algumas regras de harmonia tonal (HAAS, 2012), o algoritmo alcançou 72% de sucesso no primeiro, e 62% no segundo grupo de canções.

Ainda no ano de 2012, um trabalho não submetido ao MIREX fez uso de uma rede neural convolucional (NGIAM, 2010) como algoritmo de aprendizagem supervisionada com o intuito de realizar reconhecimento de acordes. No seu treinamento foi utilizado um corpus de 475 canções com acordes devidamente etiquetados. Neste trabalho, foram consideradas 25 classes de acordes e o mesmo alcançou em torno de 77% de sucesso em suas transcrições de acordes.

Na competição do MIREX do ano de 2013 aconteceram algumas mudanças nos critérios de avaliação e comparação dos algoritmos. Devido à constatação de que existem, sobretudo nos corpus de canções utilizadas para a realização de testes no MIREX, uma predominância muito maior de certos tipos de acordes, como os tipos maiores e menores básicos em relação a outros tipos, como os aumentados e diminutos, por exemplo, foi proposta uma validação dos resultados dos algoritmos levando em consideração a capacidade de transcrição dos mesmos por tipos de acordes. Foram então feitas avaliações dos algoritmos tomando como base cinco vocabulários de acordes:

- Apenas a tônica do acorde;
- Acordes maiores e menores: maj, min e N = ausência de acorde;
- Acordes maiores, menores e com sétima: maj, min, maj7, min7 e N = ausência de acorde;
- Acordes maiores e menores com e sem inversões: maj, min, maj/3, min/b3, maj/5, min/5 e N = ausência de acorde;

- Acordes maiores, menores, com sétima com e sem inversões: maj, min, maj7, min7, 7, maj/3, min/b3, maj7/3, min7/b3, 7/3, maj/5, min/5, maj7/5, min7/5, 7/5, maj7/7, min7/b7, 7/b7 e N = ausência de acorde;

Na análise do corpus de canções utilizadas para a realização de testes no MIREX constatou-se que menos de 1% dos acordes eram dos tipos diminutos, alterados ou aumentados e, por este motivo, como podemos observar nos vocabulários de acordes descritos acima, estes tipos de acordes não foram incluídos nos mesmos. Sendo assim, assumindo estes vocabulários para validar os algoritmos de transcrição, nos casos de ocorrências destes tipos de acordes em uma canção (diminutos, alterados ou aumentados), a métrica de análise e comparação dos algoritmos submetidos ao MIREX ignorou as partes das canções onde os mesmos ocorressem.

Sob esta nova forma de avaliação, um dos trabalhos de melhor resultado no MIREX deste ano de 2013 propôs mais uma alteração na forma de extração dos vetores PCP (CHO; BELLO, 2013). Esta alteração seria feita através da divisão em sub-bandas de toda a banda de frequências de cada instante do áudio, seguida da geração de vetores PCP individuais para cada uma destas sub-bandas. Cada vetor PCP de cada sub-banda foi utilizado para treinar HMM´s independentes, num processo que acabou por definir o nome do sistema resultante do trabalho como *Multistream HMM*. Com este modelo, quando submetido ao novo método de avaliação do MIREX, os autores alcançaram percentuais de sucesso na transcrição de acordes que chegaram a uma média geral de acertos que alcançou 71,43%.

Outro trabalho deste mesmo ano, se baseou em uma pesquisa do mesmo autor realizada no ano anterior, e que utilizava técnicas de re-associação de tempo e frequência (KODERA; GENDRIN; VILLEDARY, 1978) em conjunto com HMM e Viterbi (KHADKEVICH; OMOLOGO, 2013). Com este modelo, quando submetido ao novo método de avaliação do MIREX, o trabalho alcançou média geral de acertos de 75,80%.

Uma abordagem diferente proposta no MIREX de 2013 considerou um sistema de transcrição de acordes baseado na combinação de redes neurais e HMM (STEENBERGEN; BURGOYNE, 2013). A ideia se baseou no sistema de Bengio (BENGIO *et al.*, 1992) que foi proposto para resolver problemas de reconhecimento de fala. Entretanto, quando utilizado para a tarefa de reconhecimento de acordes, o

modelo não teve bons resultados em nenhum dos corpus de testes do MIREX. O trabalho alcançou média geral de acertos de 6,65%.

Ainda em 2013, um trabalho também indicou que as redes neurais ainda podem ser uma alternativa para a resolução do nosso problema de transcrição de acordes (BOULANGER-LEWANDOWSKI; BENGIO; VINCENT, 2013). Utilizando um modelo de Rede Neural Recorrente (MANDIC; CHAMBERS, 2002) capaz de aprender sobre propriedades musicais básicas e de escolher eficientemente entre várias sequências de acordes, aquela que é mais plausível harmonicamente, o sistema desenvolvido passou por testes que indicaram resultados próximos aos dos trabalhos em estado da arte. Entretanto, como este trabalho não foi submetido ao MIREX deste ano de 2013, ele não pode ser avaliado e comparado com outros trabalhos sob as mesmas condições.

Em 2014, o trabalho que obteve o melhor resultado no MIREX foi desenvolvido pelos mesmos autores que obtiveram o melhor resultado de 2013 (KHADKEVICH; OMOLOGO, 2014). Mais uma vez eles utilizaram técnicas de re-associação de tempo e frequência para a obtenção dos vetores *chroma*, e neste novo trabalho utilizaram múltiplos HMM´s treinados para cada tipo de acorde. A média de sucesso de seus resultados no processo de transcrição de acordes alcançou o percentual de 66,13%.

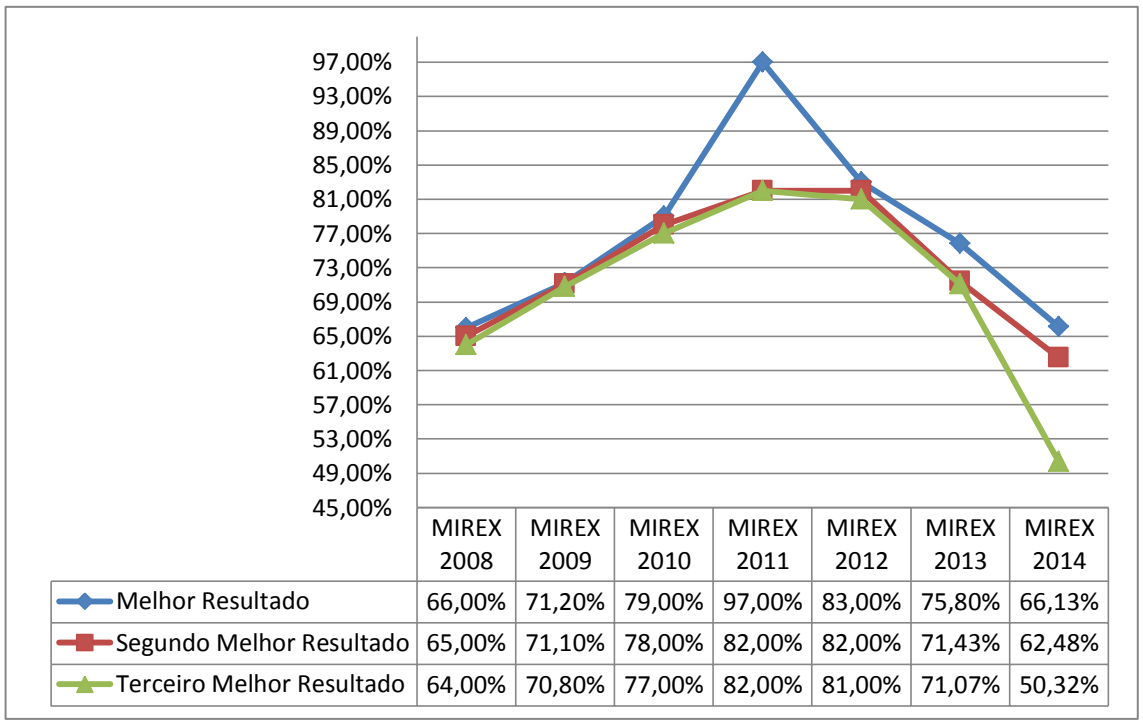
Outro trabalho que também teve resultados relevantes no MIREX de 2014 utilizou um conjunto de técnicas para alcançar os seus objetivos (ROLLAND, 2014). Partindo de uma extração dos vetores *chroma* convencional, através do uso da transformada Fourier, testando vários tamanhos de janelas para a transformada, sendo auxiliado por uma rede neural treinada no reconhecimento de acordes, de um HMM para identificar o melhor encadeamento entre acordes e utilizando algoritmos clássicos de detecção de beats e de macro estruturas cíclicas, o trabalho alcançou no MIREX percentuais de sucesso em média de 50,32%.

Por fim, um último trabalho submetido ao MIREX no ano de 2014 foi o da equipe do Queen Mary que submeteu os seus algoritmos e plug-ins para serem validados nos corpus de canções do MIREX. Com um sistema baseado em extração de vetores *chroma* utilizando o algoritmo do NNLS *chroma*, e também utilizando HMM e Viterbi, com identificação de tonalidade, o sistema batizado de *Chordino* alcançou 62,48% em média de sucesso na tarefa de transcrição de acordes (CANNAM *et al.*, 2014)

5.5. Resumo

Após esta extensa análise de uma grande quantidade de pesquisas e trabalhos relacionados com a tarefa de transcrição de acordes, e propostos ao longo das últimas décadas, alguns pontos podem ser concluídos e, como veremos, eles servirão de motivação para a proposta de nosso trabalho.

Em primeiro lugar, considerando o excelente trabalho comparativo realizado pelos organizadores do MIREX nos últimos anos e eliminando alguns pontos fora da curva (NI *et al.*, 2011), a grande maioria dos trabalhos em nível de estado da arte tem alcançado resultados bem próximos. No Gráfico 5.2, mostramos como evoluíram os resultados dos principais algoritmos de transcrição de acordes testados no MIREX desde 2008, ano em que esta tarefa passou a ser avaliada. No Gráfico 5.3, as referências a estes trabalhos são indicadas.



	MIREX 2008	MIREX 2009	MIREX 2010	MIREX 2011	MIREX 2012	MIREX 2013	MIREX 2014
Melhor Resultado	66,00%	71,20%	79,00%	97,00%	83,00%	75,80%	66,13%
Segundo Melhor Resultado	65,00%	71,10%	78,00%	82,00%	82,00%	71,43%	62,48%
Terceiro Melhor Resultado	64,00%	70,80%	77,00%	82,00%	81,00%	71,07%	50,32%

Gráfico 5.2 - Evolução dos resultados dos melhores algoritmos submetidos ao MIREX

Sobre estes resultados (referências dos últimos três anos identificada no Gráfico 5.3), vale ressaltar que a quase totalidade destes algoritmos foi treinada, testada e validada nos gêneros musicais Pop ou Pop Rock, o que leva-nos a supor que os seus resultados podem ter uma variação de rendimento se os mesmos forem aplicados em estilos musicais que utilizem padrões de sequências de acordes diferentes ou elementos percussivos e harmônicos de outra natureza. Além disso,

com a mudança, a partir de 2013, da metodologia utilizada no MIREX para a análise das transcrições de acordes realizadas por cada algoritmo, com a adoção de um processo mais rigoroso e criterioso, percebemos que os resultados alcançados pelos melhores algoritmos neste ano sofreram uma queda de desempenho em relação aos melhores resultados alcançados em 2012. No Quadro 5.1 listamos as formas de avaliação ao longo dos anos na competição do MIREX para a tarefa de transcrição de acordes, bem como o tipo de corpus de testes utilizado nas avaliações.

Ano do MIREX	Métricas e Corpus de Testes
2008	Métrica: Número de unidades de tempo onde os acordes têm identificação correta, dividido pelo número de unidades de tempo que contém acordes detectáveis na canção original Corpus de Testes: Público
2009	Métrica: Distância de *Hamming* Corpus de Testes: Público
2010	Métrica: Distância de *Hamming* Corpus de Testes: Público
2011	Métrica: Distância de *Hamming* Corpus de Testes: Público
2012	Métrica: Distância de *Hamming* Corpus de Testes: Parte Público e Parte Privado
2013	Métrica: Distância de *Hamming* com avaliações independentes por tipos de acordes Corpus de Testes: Parte Público e Parte Privado
2014	Métrica: Distância de *Hamming* com avaliações independentes por tipos de acordes Corpus de Testes: Parte Público e Parte Privado

Quadro 5.1 – Evolução das métricas de avaliação utilizadas no MIREX e os tipos de corpus utilizado nos testes

Outro aspecto interessante que podemos analisar diz respeito aos algoritmos submetidos ao MIREX e validados sobre corpus públicos e corpus desconhecidos de canções. No Gráfico 5.3, indicamos como os melhores algoritmos apresentaram resultados diferentes em cada um destes corpus de canções, em avaliações realizadas nas edições do MIREX de 2012, 2013 e 2014. Como já supúnhamos, as validações em corpus desconhecidos de canções tendem a apresentar resultados inferiores.

Além destas conclusões, pelo levantamento que fizemos, observamos que os trabalhos e modelos que tentam realizar transcrições de acordes diretamente de sinais de áudio de canções, em geral começam as suas análises com a aplicação da transformada de Fourier ou da transformada da constante Q a fim de obter o vetor PCP ou vetor *chroma* do sinal de áudio a cada instante. Com a extração do vetor *chroma*, independentemente do método escolhido para tal, alguns algoritmos

trabalham no caminho de realizar o reconhecimento dos acordes por meio de um casamento de padrões com modelos de acordes, e outros partem para o uso de algoritmos de aprendizagem que, também em sua imensa maioria, envolvem o uso de HMM e Viterbi, ou ainda, como vimos nos trabalhos mais recentes, envolvem o uso de aprendizagem com redes neurais ou SVM (*Support Vector Machines*), embora com menor expressividade.

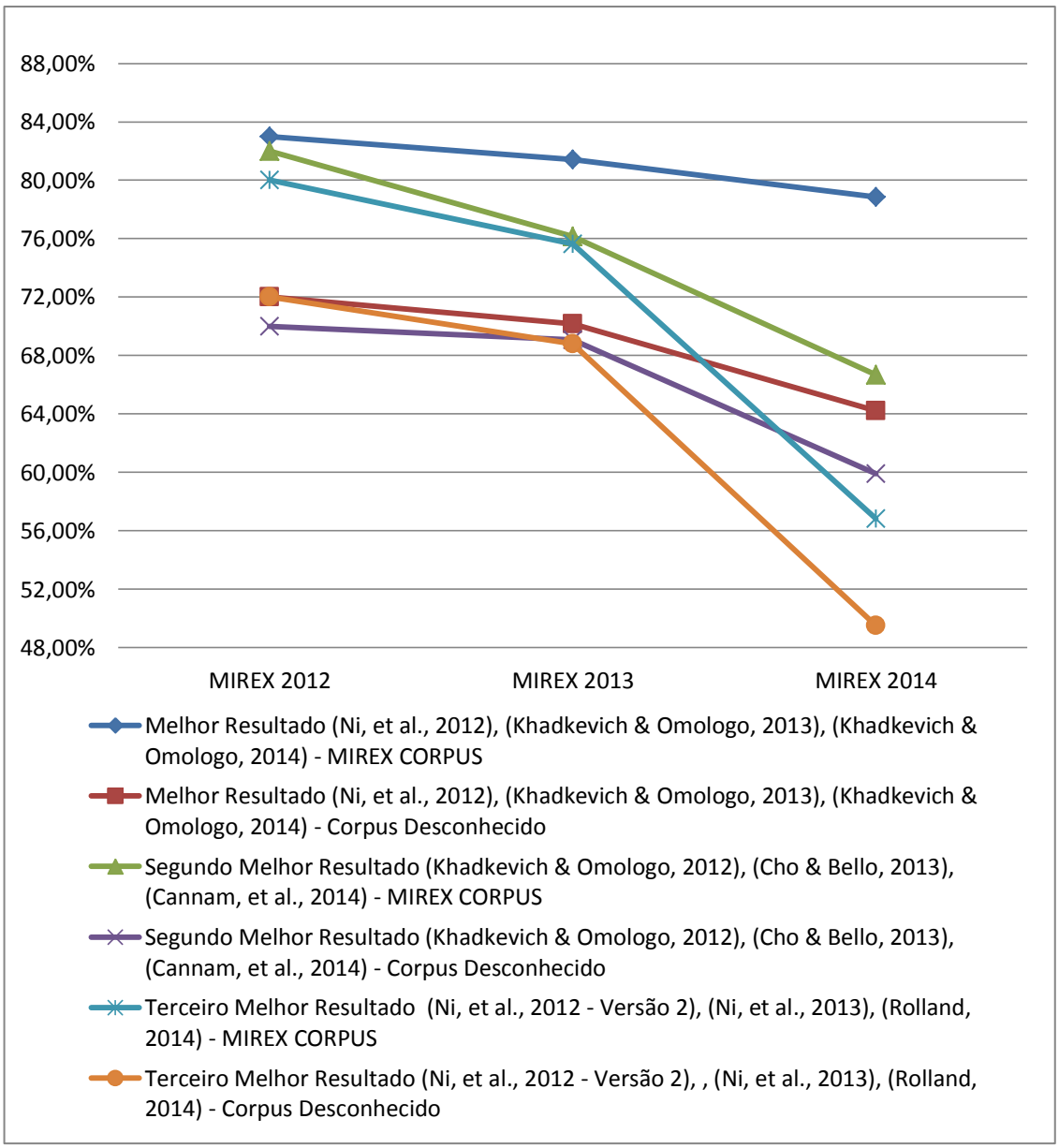


Gráfico 5.3 – Resultados dos algoritmos nos corpus públicos (MIREX) e corpus desconhecidos de canções (McGILL) a partir do MIREX 2012

Em geral, os algoritmos de melhores resultados aliam à extração dos vetores *chroma* e à técnica ou algoritmo escolhido para a realização do reconhecimento de

acordes (classificação), outros algoritmos que tentam compor a resolução do problema de transcrição atuando em alguns dos seus subproblemas (detalhados no capítulo 3), como a segmentação pela detecção de andamentos (beats) e a detecção de tonalidade geral ou local de canções.

Em bem menor escala, encontramos alguns trabalhos que tentam atuar num outro nível, detectando estruturas cíclicas, como refrães e estrofes e sequências comuns de acordes com intuito de enriquecer o contexto harmônico-musical do processo de transcrição e assim diminuir o universo de possibilidades de acordes a serem transcritos. No Gráfico 5.4 indicamos as relações entre estas implementações dentro do universo de propostas de soluções analisadas.

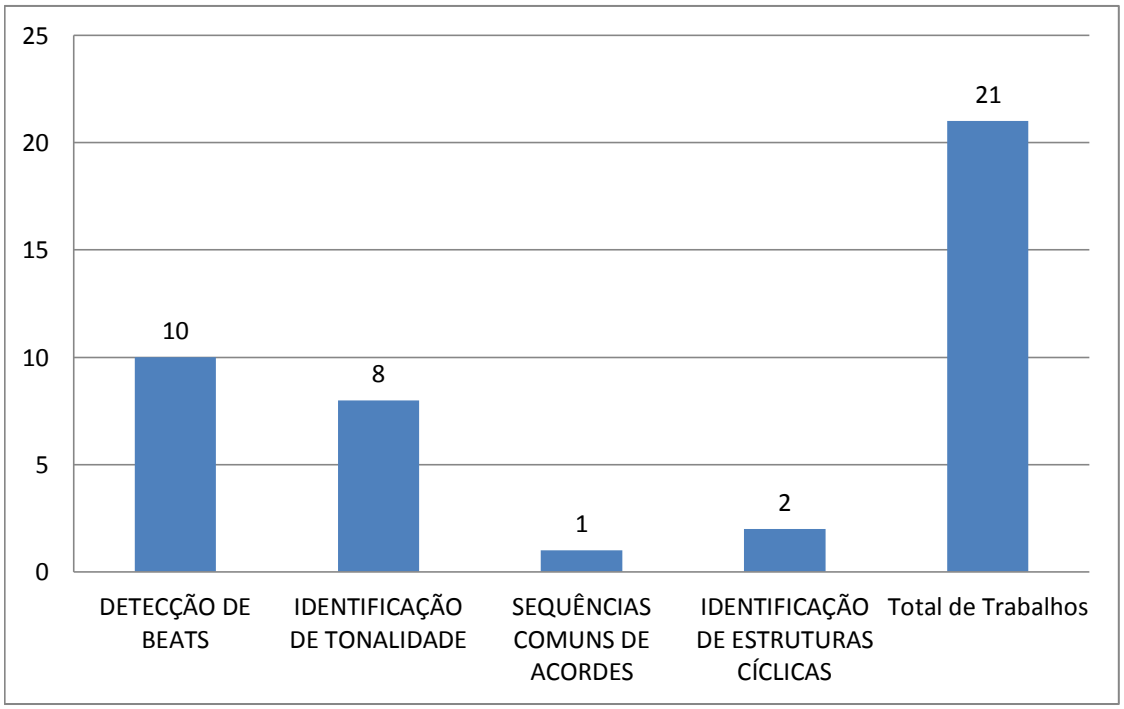


Gráfico 5.4 - Quantidade de Artigos Analisados por Tipos de Informações de Contexto Musical

No Quadro 5.2 detalhamos os artigos utilizados para compor o Gráfico 5.4 separados pelos tipos de informações de contexto musical com as quais os mesmos lidam.

É importante também frisar que os melhores algoritmos submetidos ao MIREX nos últimos quatro anos sempre fizeram uso de uma ou mais destas informações de contexto, como indicamos a seguir:

- Melhor algoritmo submetido ao MIREX em 2010: Utilizou detecção de andamento (beats) (MAUCH; DIXON, 2010)
- Melhor algoritmo submetido ao MIREX em 2011: Utilizou detecção de andamento (beats) e detecção de tonalidade (NI *et al.*, 2011)
- Melhor algoritmo submetido ao MIREX em 2012: Utilizou Detecção de andamento (beats) e detecção de tonalidade (NI *et al.*, 2012)
- O melhor algoritmo submetido ao MIREX em 2013: Utilizou, assim como o segundo melhor algoritmo, detecção de andamento (beats) e detecção de tonalidade (KHADKEVICH; OMOLOGO, 2013).
- O segundo melhor algoritmo submetido ao MIREX em 2014: Utilizou detecção de tonalidade (CANNAM *et al.*, 2014).

Detecção de Beats	Identificação de Tonalidade	Sequências Comuns de Acordes	Identificação de Estruturas Cíclicas
(Peeters, 2007)	(Yoshioka, et al., 2004)	(Cho & Bello, 2011)	(Cho & Bello, 2011)
(Yoshioka, et al., 2004)	(Lee & Slaney, 2007)		(Mauch, et al., 2009)
(Zenz & Rauber, 2007)	(Zenz & Rauber, 2007)		
(Ellis & Poliner, 2007)	(Weil & Durrieu, 2008)		
(Papadopoulos & Peeters, 2009)	(Ueda, et al., 2010)		
(Mauch & Dixon, 2010)	(Ni, et al., 2011)		
(Ni, et al., 2011)	(Haas, et al., 2012)		
(Cho & Bello, 2011)	(Cannam, et al., 2014)		
(Ni, et al., 2012)			
(Haas, et al., 2012)			
(Rolland, 2014)			

Quadro 5.2 – Detalhamento dos artigos que tratam de trabalhos que fazem uso de informações do contexto musical

De posse de todas estas observações, fica-nos a clara indicação de que a resolução do problema de transcrição de acordes a partir de sinais de áudio é uma tarefa que é quase sempre melhor resolvida quando se faz uso de várias técnicas que atuam em problemas que fazem parte do contexto geral do problema de transcrição (análise de sinal de áudio por espectrogramas, segmentação e detecção de andamento, detecção de tonalidade, classificação, entre outras). Este contexto

não pode ser desprezado e, como já analisamos, as principais soluções em estado da arte dos últimos anos sempre fazem uso de uma ou outra destas informações.

Todas estas percepções, além da proximidade da curva de percentuais de sucesso encontrada nos melhores algoritmos de transcrição, nos levam ao questionamento sobre o quanto realmente os modelos e propostas de soluções atuais estariam próximos dos seus limites e, principalmente, se não haveria espaço para propostas alternativas que pudessem contribuir para a melhora de desempenho dos sistemas em estado da arte em geral.

Por outro lado, os poucos algoritmos e propostas que tentam lidar com informações musicais contextuais e de caráter preditivo, como sequências típicas de acordes e estruturas cíclicas, sempre o fizeram a partir de técnicas de análise dos sinais de áudio das canções que, como já frisamos no capítulo 3, possuem algumas limitações. Além disso, entre os membros da comunidade científica que atuam nesta área de MIR, não existe um consenso sobre a real importância deste tipo de informação para um processo de transcrição de acordes, e as propostas que tentaram utilizá-las, quando o fizeram não se preocuparam em avaliar o real impacto das mesmas no processo de transcrição de acordes. Seu uso foi motivado pela possibilidade de que o conhecimento relacionado com elas poderia melhorar suas transcrições. Tentar uma abordagem diferente tanto na forma de identificar estas informações preditivas, como na forma de utilizar este conhecimento, poderia ser, portanto, uma alternativa interessante.

Todas estas questões se traduziram em motivações para a abordagem que estamos propondo neste trabalho e que será detalhada ao longo dos próximos capítulos.

6. Modelo Integrado de Predição-Transcrição

Para poder oferecer respostas às nossas perguntas de pesquisa, em particular à RQ1, foi preciso propor um método para captura das informações musicais preditivas (o que responderia a RQ2), assim como um modo de integrar tais informações a um processo de transcrição de acordes (o que responderia a RQ3). Neste capítulo apresentamos uma visão geral do nosso modelo que integra três módulos principais: um Transcritor Genérico de Acordes, o Preditor de Acordes e o Módulo Decisor.

6.1. Visão geral do Modelo

A proposta de nosso modelo para tentar responder a todas às perguntas de pesquisa formuladas se baseará na ideia de uma camada ou sistema de pós-processamento composta por um módulo de previsão de acordes a ser acoplado em algoritmos de transcrição de acordes genéricos com o intuito de, não descartando seus avanços e desempenhos, tentar melhorar suas transcrições corrigindo eventuais erros dos mesmos através do uso de informações musicais preditivas (sequências típicas de acordes e de estruturas cíclicas).

Na prática, porém, este sistema de pós-processamento só atuaria nos momentos em que suas sugestões de correção fossem por ele consideradas suficientemente relevantes. Sendo assim, uma parte importante de todo o modelo seria a definição de um Módulo Decisor capaz de definir quando as sugestões do pós-processamento ou previsões de acordes deveriam prevalecer em relação ao que tivesse sido transcrito originalmente. Na Figura 6.1 indicamos a visão geral do modelo em proposta.

A escolha por um modelo de pós-processamento teve várias motivações: (a) é metodologicamente mais fácil realizar um teste A/B que permita a comparação entre sistemas de transcrição de acordes existentes utilizando e sem utilizar o passo do pós-processamento; (b) desenvolver todo o processo de transcrição de acordes tentando atuar em todos os seus subproblemas desde o início, para então acrescentar a ideia de previsão de acordes, poderia trazer novos problemas inerentes aos passos preliminares da transcrição, alguns deles já tratados em sistemas em estado da arte, e que poderiam inviabilizar as respostas às nossas

perguntas de pesquisa; (c) com esta proposta de pós-processamento, o impacto de uma resposta positiva a nossa principal pergunta de pesquisa, RQ1, seria mais relevante, já que estaríamos propondo um modelo independente do sistema de transcrição de acordes.

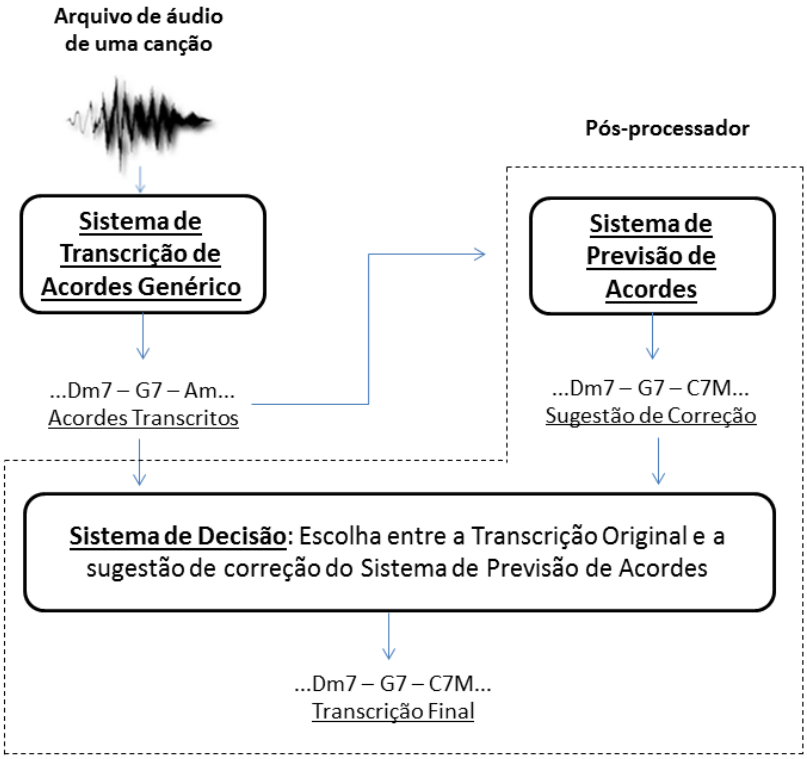


Figura 6.1 - Visão Geral do Modelo Integrando as Partes do Modelo

6.2. Integrando as Partes do Modelo

Com a ideia geral do modelo detalhada e justificada, o próximo passo seria conseguir integrar cada uma de suas partes e definir como elas interagiriam entre si, Em primeiro lugar, em uma proposta desta natureza, embora seu objetivo seja claro, para que ela seja possível de ser posta em prática, será preciso esclarecer, em primeiro lugar, como acontecerão os acoplamentos do pós-processamento (módulo de previsão de acordes), com os sistemas de transcrição genéricos. Se a ideia é de propor um modelo que possa ser utilizado por sistemas de transcrição em geral, é preciso que exista algum protocolo claro e viável para que este acoplamento possa acontecer.

Além disso, ainda existe outra questão importante a ser analisada e respondida. Como será possível realizar as correções sobre as transcrições dos

sistemas aos quais o pós-processamento irá se acoplar, sem que existam parâmetros claros que permitam uma comparação entre o que pode ser considerado mais ou menos certo num processo de transcrição de acordes? Em outras palavras, o que vai garantir que uma sugestão de correção feita pelo Módulo Decisor do pós-processamento é mais confiável do que uma transcrição feita por um sistema ao qual ele esteja acoplado?

Nas próximas subseções apresentaremos propostas de solução para estes questionamentos relacionados com a integração do nosso modelo.

6.2.1. Como acoplar o módulo de previsão de acordes do sistema de pós-processamento a sistemas de transcrição de acordes em geral?

Em primeiro lugar é preciso estar claro que não seria possível, nem é nosso objetivo, definir um processo de acoplamento universal que funcione para todo e qualquer sistema de transcrição de acordes. Porém, para tentar responder a esta pergunta, é essencial propor um método de acoplamento que possa ser aplicado em uma escala minimamente aceitável de algoritmos, preferencialmente aqueles com relevância na área. Em tese, a agregação de informações do contexto musical de caráter preditivo nos processos de transcrição dos sistemas em geral poderia ser melhor trabalhada se estes sistemas fornecessem acesso às execuções de seus processos de transcrições ou até mesmo aos seus códigos fontes. Poderia ser, por exemplo, enriquecedor, se fosse possível acessar a execução do processo de transcrição destes algoritmos e, agregando este tipo de conhecimento musical, pudéssemos interferir nas regras que definem a seleção entre os acordes candidatos à transcrição, e até na escolha daquele que deveria ser, de fato, o acorde a ser transcrito.

Porém, para uma interferência neste nível, além da necessidade de acesso ao código fonte de cada sistema, nem sempre disponível, teríamos que lidar com a óbvia dificuldade imposta pela necessidade de darmos tratamentos personalizados para cada sistema que precisássemos intervir. Neste sentido, a ideia mais simples é realmente a de acoplar-se a estes sistemas lidando com as suas saídas (acordes transcritos) como se estes algoritmos fossem “caixas pretas”, cada qual com suas próprias regras e processos de transcrições.

Entretanto, mesmo para este modelo de acoplamento, é preciso estabelecer um protocolo padrão de integração que permita a máxima universalização do mesmo. Embora seja claro que este modelo não funcionará para todos os algoritmos de transcrição de acordes, a forma como será feita a proposta possibilitará a utilização do nosso pós-processamento por uma parte importante dos sistemas de transcrição de acordes, entre eles, alguns dos principais em estado da arte.

Para tanto, esta integração será feita através do uso do modelo de arquivo de saída de acordes transcritos definido pelos organizadores do MIREX (HARTE *et al.*, 2005) (ver Anexo III). Como a grande maioria dos sistemas em estado da arte é submetida à competição do MIREX, e como para isso suas saídas de acordes transcritos têm que obedecer a um formato padrão de armazenamento de dados num arquivo em disco, tudo exigido pela organização da competição, o desenvolvimento em nosso sistema de pós-processamento de um processo de importação dos dados de transcrições destes arquivos definirá um procedimento que compatibilizará nossa proposta de solução com a maioria dos algoritmos em estado da arte. Seguindo esta abordagem, nosso pós-processamento poderá acoplar-se sobre todos os sistemas que foram submetidos ao MIREX na tarefa de transcrição de acordes.

6.2.2. Pós-processamento e Sistemas de Transcrição de Acordes: Quais transcrições são mais confiáveis?

Após a definição de como irá funcionar a integração do sistema de pós-processamento com os sistemas de transcrição de acordes, ainda é preciso responder a uma importante pergunta: Como e quando o Módulo Decisor de nosso sistema de pós-processamento deverá decidir que uma correção, seja pela identificação de sequências comuns de acordes, seja pela identificação de alguma estrutura cíclica, deve ser imposta sobre os resultados obtidos por um sistema de transcrição de acordes? Na prática, a dificuldade estaria em conseguir mensurar o quanto as análises realizadas pelo Módulo Decisor seriam, de fato, confiáveis.

Neste sentido, como não existem parâmetros que permitam que façamos uma comparação do grau de confiança daquilo que o Módulo Decisor sugere em relação às saídas de qualquer algoritmo de transcrição de acordes, para assim podermos

sempre saber a quem priorizar, a forma encontrada para que fosse tomada uma decisão se baseou em uma análise experimental.

Após várias simulações, e utilizando o fato de que o Módulo de Previsão de Acordes sugere um acorde em determinado ponto da canção conseguindo sempre informar o seu percentual de certeza quanto àquela sugestão, foi analisado o desempenho das correções do Módulo Decisor supondo o aceite das sugestões do Módulo de Previsão de acordo com um maior ou menor “percentual de sua certeza” em relação às mesmas. Com os experimentos, foi verificado que existe um percentual médio de certeza do Módulo de Previsão a partir do qual suas sugestões de correções podem ser consideradas pelo Módulo Decisor mais confiáveis do que as transcrições realizadas por um sistema de transcrição de acordes genérico. Nossas avaliações indicaram que, neste patamar, a atuação do sistema pós-processamento melhora o desempenho geral das transcrições realizadas por um sistema genérico de transcrição de acordes. Todos os testes e experimentos serão detalhados nos próximos capítulos.

6.2.3. O Algoritmo geral do Sistema de Pós-Processamento

Com a definição de como o sistema de pós-processamento se integrará com os sistemas de transcrição de acordes e de como será mensurada a confiabilidade de suas sugestões de correções de transcrições, na Figura 6.2 propomos um esquema mais detalhado de nossa proposta de solução. Nos passos a seguir, descrevemos de uma forma geral o funcionamento do modelo.

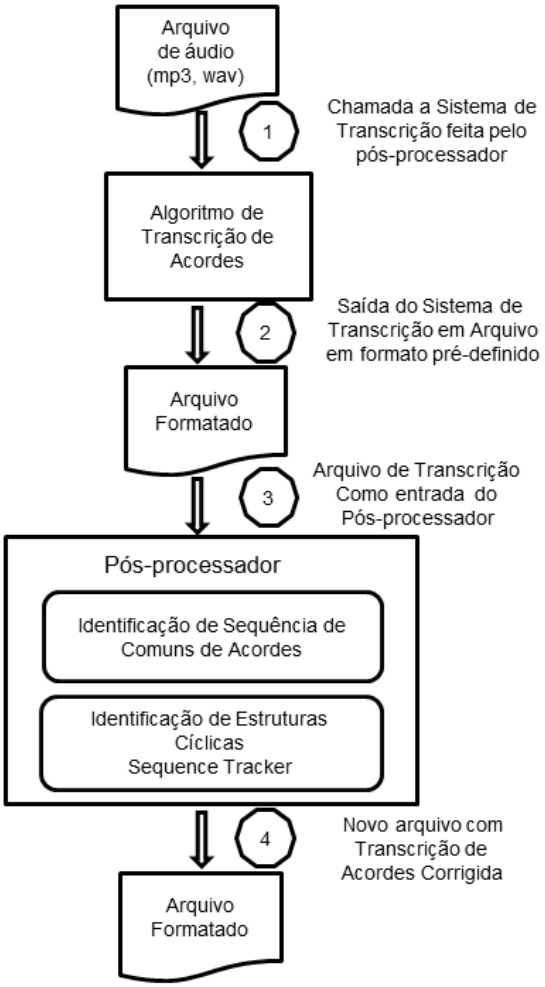


Figura 6.2 - Modelo da Camada ou Sistema de Pós-Processamento

1. Fornecimento de arquivo de áudio como entrada de um sistema de transcrição de acordes que obedece ao padrão MIREX. (Passo 1 da Figura 6.2);
2. O resultado desta transcrição, ou melhor, os acordes transcritos pelo algoritmo de transcrição são exportados para arquivo no padrão MIREX (HARTE *et al.*, 2005) (Passo 2 da Figura 6.2);
3. O arquivo com os acordes transcritos é fornecido como entrada para o pós-processamento (Passo 3 da Figura 6.2);
4. Atuação do pós-processamento neste arquivo de acordes transcritos tentando realizar correções nos eventuais erros ocorridos na transcrição de acordes realizada pelo sistema de transcrição, finalizando com a consequente saída de novo arquivo de acordes transcritos (Passo 4 da Figura 6.2).

A descrição detalhada do funcionamento deste modelo será dada a partir do próximo capítulo.

7. O Módulo de Previsão de Acordes

O Módulo de Previsão de Acordes do sistema de pós-processamento será responsável pela tentativa de identificação de sequências típicas de acordes e de estruturas cíclicas, como refrães, estrofes e seções. Nas próximas seções, detalharemos o desenvolvimento do mesmo.

7.1. Identificando Sequências Típicas de Acordes

Como já mencionado, em trabalhos anteriores nós obtivemos relevantes resultados no uso de uma rede neural MLP com o algoritmo *backpropagation* para a realização da tarefa de identificação de sequências típicas de acordes. Neste trabalho original, esta tarefa foi realizada com a aplicação da rede neural como um algoritmo preditor de acordes, que recebia como entrada uma janela de três acordes (compostos pela sua tônica, tipo ou categoria, posição dentro do compasso e duração, informações extraídas diretamente das partituras de cada canção). Neste cenário, a rede neural foi capaz de identificar qual seria o próximo acorde com um percentual de sucesso que chegou ao nível 87% de acertos. Foi utilizado um corpus de 58 canções de jazz, das quais extraímos um conjunto de testes independente formado por 18 canções e sobre o qual o percentual de sucesso de previsão foi encontrado. Durante os treinamentos a prevenção de *overfitting* foi realizada através de validação cruzada.

A ideia para a nossa proposta atual seria utilizar o mesmo tipo de rede neural MLP a ser treinada com o algoritmo *backpropgation* para a execução da tarefa de previsão de acordes. Porém, no contexto atual, onde o ambiente é bastante diferente, já que a análise parte de arquivos de canções de áudio e não de dados retirados diretamente de partituras de canções, como ocorreu em nosso trabalho original, tentar utilizar a mesma ideia exigiu, antes de tudo, uma remodelagem dos atributos que deveriam compor cada acorde. Sem poder definir exatamente onde começa e onde termina cada compasso em uma canção analisada a partir de seu arquivo de áudio, não foi possível o uso dos atributos de Posição Dentro do Compasso e Duração que ajudaram a definir a configuração de rede neural que alcançou os melhores resultados na tarefa de previsão de acordes de nosso trabalho anterior. Sem a possibilidade de usar estes atributos, cada acorde passou a ser

codificado apenas com os atributos restantes: a tônica e o seu tipo. Com o redimensionamento dos atributos que definiriam cada acorde, partimos para a sua codificação.

7.1.1. Codificando os Atributos de Entrada da Rede Neural

Com a definição dos atributos de tônica e tipo de cada acorde como únicos disponíveis para a representação de cada acorde, o passo seguinte seria definir como codificá-los adequadamente para que os mesmos pudessem ser utilizados em um processo de treinamento de uma rede neural do tipo MLP. Para tanto, optamos pela utilização da mesma codificação binária utilizada em nossa pesquisa anterior e que levou a rede neural treinada aos seus melhores resultados (CUNHA, 1999).

Para a definição exata da codificação de cada valor possível de cada atributo, antes de tudo, foi feita uma análise para a identificação de todos os acordes com seus respectivos tipos encontrados no corpus de 180 canções que seriam utilizadas para treinamento e testes. Neste universo foram encontrados 16 tipos de acordes aliados às 12 tônicas existentes (Dó, Dó#, Ré, Re#, Mi,...,Si). Com a definição do domínio dos valores de cada atributo, partimos para a codificação dos mesmos no formato binário utilizado em nossa pesquisa anterior.

Tônica	Codificação da Tônica
C ou B#	000000000001
C# ou Db	000000000010
D	000000000100
D# ou Eb	000000001000
E ou Fb	000000010000
E# ou F	000000100000
F# ou Gb	000001000000
G	000010000000
G# ou Ab	000100000000
A	001000000000
A# ou Bb	010000000000
B ou Cb	100000000000

Tabela 7.1 – Tônicas de acordes e suas Codificações

Tipo do Acorde	Codificação do Tipo do Acorde
Maj	0000000000000001
Min	0000000000000010
Dim	0000000000000100
Aug	0000000000001000
maj7	0000000000010000
min7	0000000000100000
7	0000000001000000
dim7	0000000010000000
hdim7	0000000100000000
minmaj7	0000001000000000
maj6	0000010000000000
min6	0000100000000000
9	0001000000000000
maj9	0010000000000000
min9	0100000000000000
sus4	1000000000000000

Tabela 7.2 - Tipos de Acordes e suas Codificações

Nesta codificação, cada um dos possíveis valores de cada atributo seria definido como um vetor binário com a mesma dimensão do número de valores existentes no domínio do atributo (tônicas = 12, tipos de acordes = 16), e o que identificaria cada possível valor do atributo individualmente seria a posição em que ocorreria no vetor de codificação binária um bit com o valor 1. Todos os demais valores do vetor seriam iguais a zero. Nas Tabelas 7.1 e 7.2 são identificados os valores utilizados para codificar cada tipo de acorde com sua respectiva tônica.

Segundo esta codificação, o acorde “Amin7”, seria traduzido para a seguinte representação “00100000000000000000000100000”, o que significa na prática a definição de que cada acorde seria codificado como um vetor de 28 bits (Figura 7.1).

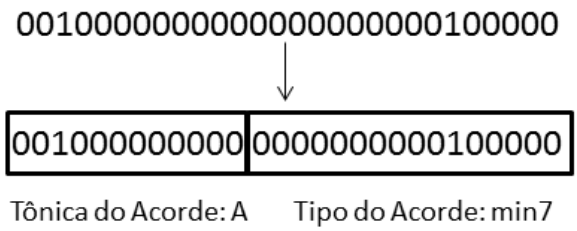


Figura 7.1 - Codificação com matriz esparsa binária do acorde “Amin7”

Vale ressaltar que para esta pesquisa atual utilizamos uma quantidade de tipos de acordes maior do que em nosso trabalho anterior, onde apenas seis tipos de acordes foram utilizados. Na nossa implementação atual utilizamos dezesseis tipos de acordes para cada uma das doze tônicas possíveis. Este novo cenário também revela uma diferença expressiva entre o ambiente de nossa pesquisa anterior e o ambiente atual.

Com a definição dos atributos que iriam compor cada acorde, bem como de suas codificações, o próximo passo seria a realização de processos de treinamento e testes com a rede neural MLP com o objetivo de verificar o desempenho da mesma no processo de identificação de sequências típicas de acordes, através da realização da previsão de ocorrência das mesmas.

7.1.2. Procedimento Experimental de Treinamento e Testes da Rede Neural

O primeiro passo para a realização do procedimento experimental foi a definição dos conjuntos de treinamento e testes. O corpus de canções utilizado em

todos os experimentos foi composto por 180 canções dos Beatles. Como todo o procedimento experimental iria exigir duas fases bem distintas, sendo a fase inicial a de treinamento, validação e testes da rede neural para a tarefa de previsão de acordes, e a fase final a de uso desta rede neural já treinada como parte de nosso sistema de pós-processamento, e sendo utilizada para sugerir correções de transcrições, optamos pela seguinte divisão do nosso corpus de canções (Figura 7.2):

- Conjunto A: Composto por 70% das canções, 126 ao todo, escolhidas aleatoriamente a partir do corpus total. Este conjunto seria depois separado em novos subconjuntos de treinamento, validação e testes para a definição da configuração da rede neural ideal;
- Conjunto B; Composto por 30% das canções, 54 ao todo, correspondente ao restante do conjunto total de canções do corpus. Este conjunto seria isolado de todos os experimentos de treinamento, validação e testes da rede neural, e seria utilizado na fase final do procedimento experimental quando a rede neural seria incluída no modelo de pós-processamento para a validação de nossa proposta.

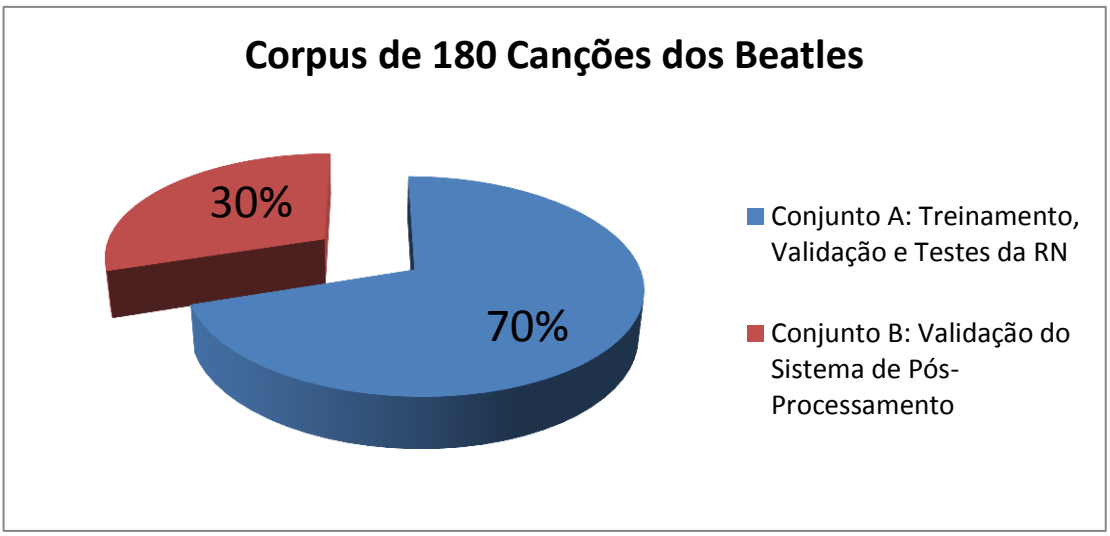


Figura 7.2 – Separação do Corpus de 180 canções dos Beatles para o procedimento experimental

Optamos por este percentual de separação entre estes conjuntos por ser uma prática comum que o conjunto de treinamento fique com um tamanho entre 50% e 80% do conjunto total de dados (KOHAVI, 1995).

Em nossas pesquisas anteriores, quando desenvolvemos um sistema de previsão de acordes baseado no mesmo tipo de rede neural MLP, fizemos testes de treinamento considerando o fornecimento como entrada para a rede neural de uma janela de acordes já executados compostos por dois, três e quatro acordes, e com a rede neural sendo treinada para prever qual deveria ser o próximo acorde. Nos experimentos que foram realizados nesta pesquisa anterior, foi constatado que o tamanho de janela ideal seria de três acordes.

No cenário atual, com as mudanças dos atributos que definiriam cada acorde, e com a alteração da dimensão do domínio de valores do atributo do tipo de cada acorde, tomamos a decisão de novamente testar a quantidade de acordes ideal para a definição do tamanho da janela que deveria ser fornecida como entrada para o treinamento da rede. Mais uma vez validamos as janelas de tamanhos de dois, três e quatro acordes.

Com esta definição, e sabendo que com o padrão de codificação de acordes definido, cada bit da codificação exigiria uma entrada na rede neural, caso a mesma estivesse sendo treinada com uma janela de três acordes, cada um deles representado por 28 bits, seriam necessárias 84 entradas na rede. Como a ideia no processo de aprendizagem supervisionado com uma rede MLP com o algoritmo *backpropagation* é a de fornecer uma entrada, que é nossa janela de acordes e, na camada de saída, “mostrar” à rede a saída desejada, que neste caso é o acorde a ser previsto, na nossa camada de saída foi necessário que criássemos 28 neurônios, cada um responsável pela identificação de um dos 28 bits de representação do acorde de saída.

Para efetivamente realizar o treinamento, como iríamos realizar avaliações com janelas de acordes de tamanhos de dois, três e quatro acordes, para cada canção do corpus, a partir de seus arquivos de acordes corretos, foram criados três novos arquivos, sendo um para cada tamanho de janela. Dependendo do tamanho da janela utilizada, cada arquivo seria composto por linhas contendo os acordes devidamente codificados e na quantidade definida pelo tamanho da janela de entrada da rede. Além disso, no final de cada linha do arquivo seria informado o acorde que representaria a saída desejada da rede para cada janela de entrada. Na prática, supondo uma janela de três acordes, se no arquivo original de acordes do corpus de uma dada canção tivéssemos a sequência de acordes C, Am, Dm e G7, os três primeiros acordes seriam codificados, cada um no formato já definido de 28

bits, e comporiam a janela de entrada da rede. Esta janela seria posta em uma linha do arquivo que seria complementada com o quarto acorde da sequência, o acorde de saída, também devidamente codificado. O processo de construção de cada arquivo deveria se repetir para cada nova sequência de acordes até o final de cada canção. Na Figura 7.3 ilustramos o exemplo dado para um arquivo com janela de três acordes.

Após a criação dos três arquivos para cada uma das canções do conjunto “A” do corpus, cada um deles organizado em linhas contendo os pares de entrada e saída da rede e compostos pelas janelas de 2, 3 e 4 acordes, respectivamente, juntamente com os acordes que identificariam as saídas desejadas, foram criados arquivos únicos onde foram postos os conteúdos de cada arquivo de cada canção por cada tamanho de janela de entrada. Desta forma, passamos a ter apenas três grandes arquivos, devidamente organizados em pares de entrada e saída, respectivamente, de janelas de dois, três e quatro acordes, e compostos, cada um deles, por todas as canções do conjunto “A”.

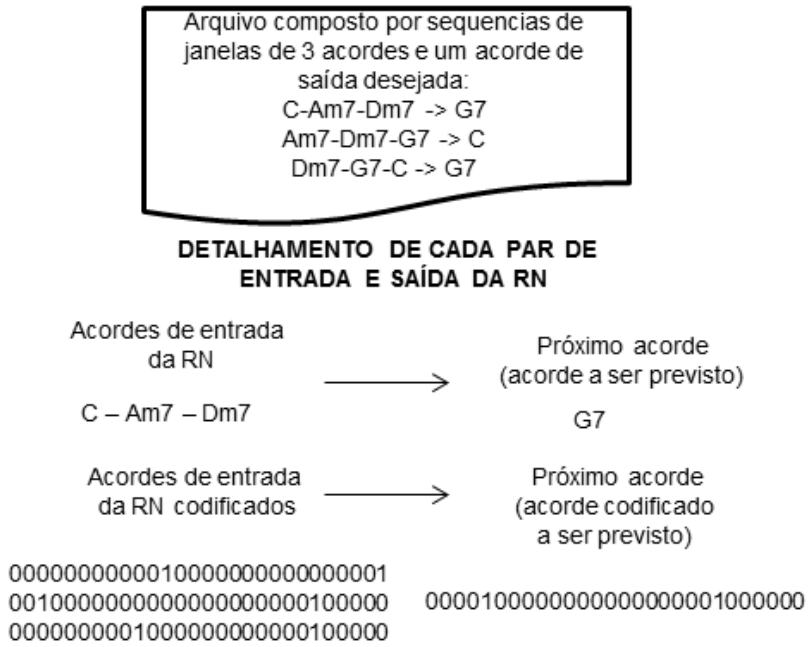


Figura 7.3 - Arquivo de entrada de três acordes codificados com a respectiva saída desejada da rede neural

Com a preparação dos dados para os treinamentos, validações e testes, para a execução destes procedimentos foi utilizado o pacote de redes neurais do Matlab,

com treinamentos envolvendo as variações do algoritmo de aprendizagem *Resilient Backpropagation* (Riedmiller, 1993) e *Scaled Conjugate Gradient Backpropagation* (Moller, 1993). No Matlab, o procedimento experimental realizado seguiu os seguintes passos:

1. Leitura e carregamento do arquivo do conjunto "A" de canções devidamente codificadas com janelas de dois acordes de entrada;
2. Configuração no Matlab da divisão do arquivo de entrada entre, aproximadamente, 70% das canções como conjunto de treinamento, 15% das canções como conjunto de testes e os 15% restantes como conjunto de validação, a fim de aplicar *Early Stopping*[14] (KIRAN; DEVI; LAKSHMI, 2010).
3. Número máximo de épocas definido para cada treinamento foi de 3.000. Este número foi definido experimentalmente a partir de pré-testes de treinamentos que indicaram um valor aproximado para o mesmo. Ao término dos treinamentos, menos de 3% foram finalizados porque atingiram o número máximo de épocas;
4. A fim de prevenir *overfitting,* o treinamento seria finalizado após 8 validações consecutivas (valor padrão sugerido no processo de treinamento no Matlab) do conjunto de validação com resultados indicando queda gradativa de desempenho. Neste caso o treinamento seria finalizado e seria selecionada a melhor configuração da rede antes da detecção do início do *overfitting* (KIRAN; DEVI; LAKSHMI, 2010);
5. Manutenção dos valores *default* dos demais parâmetros e recursos disponibilizados para configurar a rede;
6. Sabendo que o conjunto de treinamento seria composto por cerca de 9.500 exemplos, foram realizadas simulações com redes com uma única camada intermediária e com quantidades de neurônios variando nos intervalos descritos a seguir, para cada tamanho da janela de entrada de acordes:

[14] Técnica em que o conjunto de treinamento é dividido em duas partes, sendo uma utilizada para o treinamento propriamente dito, e outra ustilizada para a realização de validações durante o treinamento com o objetivo de evitar overfitting.

a) Janela de dois acordes, com a rede neural constituída de 56 entradas e 28 neurônios de saída: foram realizados experimentos com quantidades de neurônios na camada intermediária variando entre 40 e 115, com saltos de 5 unidades;
b) Janela de três acordes, com rede neural constituída de 84 entradas e 28 neurônios de saída: foram realizados experimentos com quantidades de neurônios na camada intermediária variando entre 30 e 85, com saltos de 5 unidades;
c) Janela de quatro acordes, com rede neural constituída de 112 entradas e 28 neurônios de saída: foram realizados experimentos com quantidades de neurônios na camada intermediária variando entre 25 e 70, com saltos de 5 unidades;

Como não há um consenso sobre qual deve ser a quantidade ideal de neurônios na camada intermediária de uma rede neural, definimos os procedimentos experimentais acima a fim de encontrar este valor, e para isso, baseamo-nos em indicações da literatura da área (BARTLETT, 1998), (BAUM; HAUSSLER, 1989), (BARTLETT; MAASS, 2003), (KOIRAN; SONTA, 1997), (GUYON; BOSER; VAPNIK, 1993), (VAPNIK, 2000), (BOGER; GUTERMAN, 1997), (BERRY; LINOFF, 2004).

7. Realização de 10 re-treinamentos (número indicado pelo manual técnico do Matlab em sua seção *Improve Neural Network Generalization and Overfitting*) para cada rede com o mesmo número de neurônios na camada intermediária a fim de tentar garantir que uma rede com melhor capacidade de generalização fosse encontrada;
8. Ao término do processo, cálculo para cada tamanho de rede do MSE médio sobre os conjuntos de testes, bem como do desvio padrão do mesmo;
9. Repetição de todo este processo a partir do Passo 1 para o arquivo de treinamento “A” definido com janelas de três e depois de quatro acordes.

Após a realização dos treinamentos[15], foram selecionadas as estruturas de rede de melhor desempenho médio de acordo com a variação do algoritmo de aprendizagem utilizado, tamanho da janela de acordes de entrada e número de neurônios na camada intermediária de cada rede. Os melhores resultados foram escolhidos tomando como base o cálculo do menor MSE – *Mean Square Error* médio. Nas Tabelas 7.3 e 7.4 resumimos os melhores resultados para cada janela de acordes em cada variação de algoritmo de aprendizagem.

Número de Neurônios na Camada Intermediária	MSE médio sobre Conjunto de Testes (Resilient BP)	Desvio Padrão (Resilient BP)	Número médio de Épocas	Desvio Padrão Em relação ao número de épocas
Janela de 2 Acordes				
110	0,05985	0,00373	792,50	183,084
Janela de 3 Acordes				
85	**0,05012**	**0,00301**	**1.292,50**	**265,06**
Janela de 4 Acordes				
65	0,05621	0,00389	1.965,00	437,06

Tabela 7.3 - Melhores Resultados dos Processos de Previsão de Acordes - Resilient Backpropagation

Número de Neurônios na Camada Intermediária	MSE médio sobre Conjunto de Testes (Scaled Conjugate Gradient BP)	Desvio Padrão (Scaled Conjugate Gradient BP)	Número médio de Épocas	Desvio Padrão Em relação ao número de épocas
Janela de 2 Acordes				
105	0,06382	0,00484	972,50	133,91
Janela de 3 Acordes				
80	**0,06281**	**0,00232**	**1.422,50**	**239,64**
Janela de 4 Acordes				
70	0,06499	0,00125	2.097,50	487,65

Tabela 7.4 - Melhores Resultados dos Processos de Previsão de Acordes - Scaled Conjugate Gradient BP

Para definir a rede ideal a ser utilizada como parte do pós-precessamento, escolhemos aquela de melhor desempenho nos treinamentos realizados no Passo 7 de nosso plano, que foi uma rede treinada com o algoritmo *Resilient Backpropagation* e que possuía 85 neurônios na camada intermediária (pela Tabela 7.3, esta foi a configuração de melhor desempenho médio).

[15] Todas as avaliações e passos do processo de treinamento foram realizadas em uma máquina Dual Xeon 3.4Ghz, 34GbRAM e duraram cerca de 200horas

7.2. Identificando Estruturas Cíclicas

A segunda parte do módulo de previsão de acordes do pós-processamento é a responsável pela identificação de estruturas cíclicas como refrães, estrofes ou seções. Assim como a identificação de sequências típicas de acordes, o desenvolvimento desta parte do modelo também vai se basear em nossos esforços e trabalhos anteriores (CUNHA; RAMALHO, 1999), quando desenvolvemos um analisador de sequências de acordes batizado com o nome de *Sequence Tracker (ST)*.

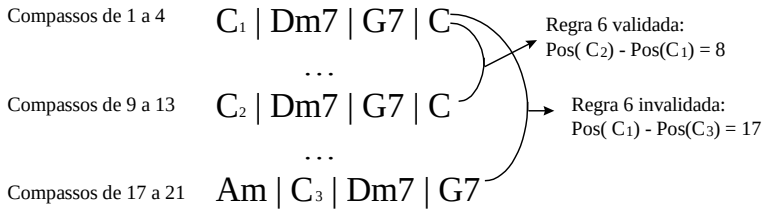
Neste trabalho anterior, o ST foi desenvolvido da análise de informações de um corpus de canções disponibilizadas a partir de suas partituras. Isso permitiu o uso de vários dados não disponíveis no contexto atual de análise de sinais de áudio. Na prática, informações fundamentais para o funcionamento do ST como o momento de início de um determinado compasso, a quantidade de compassos já executados e, sobretudo, a descrição das notas executadas na melodia da canção em cada compasso, não puderam ser mais utilizadas, o que inviabilizou o uso do algoritmo original do ST.

Porém, como sabíamos que a criação de um algoritmo similar no contexto atual poderia ser bastante útil para o pós-processamento, partimos para o uso de outra abordagem que permitiu a viabilização do desenvolvimento de um novo ST. Optamos por utilizar um algoritmo de detecção de compassos (DAVIES; PLUMBLEY, 2006) em uso no software *Sonic Visualizer* (CANNAM *et al.*, 2006) como forma de enriquecer o contexto em que o ST atual iria atuar, aproximando-o de sua ideia original. Embora os cálculos das localidades de cada compasso fossem gerados sempre com certa margem de erros, a ideia seria a de tentar verificar a viabilidade da abordagem.

Neste cenário, foi desenvolvido um protótipo em Matlab que definiu o funcionamento do novo ST. Com a possibilidade de identificar os limites de cada compasso, mesmo que de uma forma aproximada, o algoritmo deste novo analisador de sequências pôde ser definido com quase todas as regras do algoritmo original. A exceção estaria na ausência das informações da melodia executada em cada compasso da canção, que tinham forte peso nas decisões do ST original. No Quadro 7.1 listamos as regras originais do ST, e as novas regras adaptadas ao novo

cenário. No Anexo IV disponibilizamos o algoritmo do novo *Sequence Tracker*, modificado a partir do algoritmo disponibilizado em nosso trabalho original.

Regra do ST Original	**Regra adaptada para o novo ST**
1. Toda a canção é considerada como uma *repetição* de uma sequência**.** Isto é feito para que o algoritmo seja capaz de identificar a repetição da canção;	**Regra Mantida**
2. Quando existem duas possíveis *repetições* com inícios iguais concorrendo para serem usadas, nenhuma pode ser escolhida até que a sequência de acordes e melodia que venha sendo executada permita a diferenciação entre ambas.	**Regra Alterada:** Quando existem duas possíveis *repetições* com inícios iguais concorrendo para serem usadas, nenhuma pode ser escolhida até que a sequência de acordes que venha sendo executada permita a diferenciação entre ambas. **JUSTIFICATIVA:** Ausência da informação da melodia
3. Nós só podemos concluir que uma *repetição* que está se iniciando é uma repetição de outra sequência já executada após a comparação de, pelo menos, três compassos, sendo considerados nesta comparação não só os acordes como também a melodia.	**Regra Alterada:** Nós só podemos concluir que uma *repetição* que está se iniciando é uma repetição de outra sequência já executada após a comparação de, pelo menos, três compassos, sendo considerados nesta comparação apenas os acordes. **JUSTIFICATIVA:** Ausência da informação da melodia
4. Cada parte de uma canção que ainda não aconteceu em nenhuma outra parte da mesma canção é considerada uma *repetição candidata*. Como não temos condições de definir que partes da canção vão ser repetidas, consideramos todo bloco inédito de acordes desta forma;	**Regra Mantida**
5. Em muitos casos, *repetições* dentro de uma	**Regra Mantida**

canção não são *repetições exatas*, apresentando algumas diferenças nos seus compassos de números 8n ou 8n+1, onde n=1,2,3 ...;	
6. A posição absoluta dos acordes na grade influencia na detecção de uma *repetição*. Um acorde só pode ser considerado como pertencente a uma possível sequência se a diferença entre a posição do compasso que ele ocupa e a posição do compasso do acorde que a ele esteja sendo comparado seja igual a um múltiplo de quatro. Por exemplo, temos uma possível *repetição* se iniciando no compasso número 6 da canção, e a sequência com a qual estamos comparando esta possível *repetição* começa no compasso 2. A diferença entre 6 e 2 é igual a quatro, o que faz valer esta regra e permitir que a comparação continue até a constatação ou não da repetição da sequência seguindo os outros princípios 	**Regra Mantida**
7. A melodia é fundamental para a detecção de repetições de sequências de acordes. Portanto, o teste de identificação de *repetições* deve ser feito através da comparação de toda a estrutura harmônica da canção incluindo a sua melodia, que deve ser estritamente a mesma da encontrada no trecho em teste.	**Regra EXCLUÍDA** **JUSTIFICATIVA:** Ausência da informação da melodia
8. Após a exata coincidência de pelo menos 17 compassos, podemos afirmar que a canção está se repetindo.	**Regra Alterada**: Após a exata coincidência, considerando apenas a coincidência entre acordes, de pelo menos 25 compassos, podemos afirmar que a canção está se repetindo; **JUSTIFICATIVA:** Ausência da informação da melodia

Quadro 7.1 – Regras do ST original x Regras do ST atual

Para exemplificar a ideia do funcionamento ideal do ST, vamos propor um exemplo hipotético. No Quadro 7.2 visualizamos uma suposta harmonia de uma canção transcrita por um sistema de transcrição de acordes, também hipotético.

1 C Am	2 Dm	3 G7	4 C
5 Dm	6 Em F	7 G7	8 C
9 C Am	10 Dm	11 G7	12 C
13 Dm	14 F	15 B7	16 C

Quadro 7.2 – Exemplo de trecho de harmonia de uma canção hipotética. Cada célula da tabela representa um compasso numerado de 1 a 16.

No Quadro 7.3 visualizamos os acordes originais e corretos da canção:

1 C Am	2 Dm	3 G7	4 C
5 Dm	6 Em F	7 G7	8 C
9 C Am	10 Dm	11 G7	12 C
13 Dm	14 Em F	15 G7	16 C

Quadro 7.3 – Acordes corretos (*ground truth*) da mesma canção hipotética do Quadro 7.3

Considerando cada célula de cada quadro como um compasso, percebe-se claramente que os compassos 14 e 15 dos Quadros 7.2 e 7.3 são diferentes. Isto significa que, sendo o Quadro 7.2 o resultado da atuação de um transcritor de acordes sobre o arquivo de áudio da canção, existem alguns erros de transcrição nestes dois compassos.

Porém, caso aplicássemos o ST no modelo atual sobre o arquivo de acordes transcritos, pelas suas regras, uma sequência de acordes em repetição seria identificada a partir do terceiro compasso em igualdade de acordes executados (Regra 3). Para os acordes transcritos e identificados no Quadro 7.2, a primeira identificação de uma sequência em repetição pelo ST aconteceria ao término do compasso 11, com as comparações feitas entre os acordes encontrados a partir do compasso 9 (até o compasso 11), com os acordes executados do compasso 1 até o

compasso 3. Pelas suas regras, o ST já teria armazenada a sequência de acordes executados que teria se iniciado no compasso 1 e iria até o compasso 7 (compassos múltiplos de 8 não são considerados – Regras 4 e 5), como indicado na área em destaque no Quadro 7.4.

1 C Am	2 Dm	3 G7	4 C
5 Dm	6 Em F	7 G7	8 C
9 C Am	10 Dm	11 G7	12 C
13 Dm	14 F	15 B7	16 C

Quadro 7.4 –Em destaque a sequência de acordes já armazenada pelo ST no momento da identificação da suporta repetição

Com a identificação de uma repetição, o ST substitui os próximos compassos e acordes da canção pelos compassos e acordes da sequência repetida e já armazenada pelo ST. No Quadro 7.5 identificamos a sequência de acordes que será substituída pelo ST.

1 C Am	2 Dm	3 G7	4 C
5 Dm	6 Em F	7 G7	8 C
9 C Am	10 Dm	11 G7	12 C
13 Dm	14 F	15 B7	16 C

Quadro 7.5 – Em destaque a sequência de acordes que será substituída pela sequência que foi destacada no Quadro 7.4

Após a substituição da sequência em repetição, o produto final das transcrições de acordes aplicadas ao ST seria formado pelos acordes identificados no Quadro 7.6.

1 C Am	2 Dm	3 G7	4 C
5 Dm	6 Em F	7 G7	8 C
9	10	11	12

C Am	Dm	G7	C
13 Dm	14 Em F	15 G7	16 C

Quadro 7.6 – Resultado final da aplicação do ST sobre as transcrições de acordes realizadas pelo hipotético algoritmo de transcrição.

Se melhor observarmos, o Quadro 7.6 é idêntico ao Quadro 7.3 que contém os acordes corretos da canção (*ground truth*). Isso nos leva a conclusão de que após a atuação do ST sobre os acordes transcritos da hipotética canção, houve uma melhora de desempenho no resultado final da transcrição.

Este modelo funcional adaptado do ST, capaz de identificar sequências de acordes em repetição, foi definido como o modelo final a ser utilizado em nosso módulo de previsão de acordes em conjunto com a rede neural de previsão, a fim de que o nosso sistema de pós-processamento pudesse atuar de forma efetiva.

8. Desenvolvimento de um Transcritor de Acordes

O modelo de pós-processamento de nossa proposta pressupõe a existência de um algoritmo de transcrição de acordes que ele possa pós-processar. Com o intuito de mensurar melhor o desempenho de nossa proposta, optamos por tentar avaliar seus pós-processamentos sobre as saídas de algoritmos de transcrição de diferentes capacidades. Como a base do modelo é a tentativa de correção de eventuais erros de transcrição, a ideia seria a de verificar o seu comportamento quando aplicado sobre as transcrições de sistemas que apresentassem diferentes níveis de erros. Porém, embora tenhamos buscado inúmeros trabalhos e sistemas de transcrição, a disponibilização dos mesmos por seus autores nem sempre foi possível, por vários motivos.

Diante deste cenário, e buscando dentre os sistemas submetidos ao MIREX aqueles que poderiam se enquadrar às nossas necessidades, encontramos, com disponibilidade para avaliações, um algoritmo com resultados considerados muito bons no MIREX (NI *et al.*, 2011), e um algoritmo com resultados considerados fracos no MIREX (STEENBERGEN; BURGOYNE, 2013).

Para o sistema com bons resultados no MIREX utilizamos o algoritmo batizado de *Harmony Progression (HP)* (NI *et al.*, 2011), que obteve o melhor resultado na competição do MIREX nos anos de 2011 e 2012 na tarefa de transcrição de acordes.

Baseado em aprendizagem com o uso de HMM e Viterbi, o processo funcional do algoritmo HP se inicia com a aplicação de filtros com o objetivo de enfatizar as frequências do sinal de áudio relevantes para a identificação da harmonia da canção. A partir dos sinais de áudio filtrados, são calculados vetores *chroma* que alimentam um algoritmo de aprendizagem baseado em HMM e Viterbi. Aliando algumas informações de contexto musical e detecção de andamento (beats), o algoritmo, quando posto em execução com o corpus de testes do MIREX do ano de 2012, alcançou o desempenho de 80% na tarefa de transcrição.

Por outro lado, para o sistema submetido ao MIREX que apresentou resultados considerados fracos, utilizamos a proposta baseada em um algoritmo que aliou o uso de um HMM com uma rede neural (STEENBERGEN; BURGOYNE, 2013) e que foi submetido à competição no ano de 2013.

O sistema se baseava em uma rede neural com ativação do tipo *softmax* treinada com os vetores *chroma* extraídos dos sinais de áudio das canções a partir da aplicação da transformada Q. A saída da rede neural alimentava a entrada de um HMM que tinha o objetivo de estimar a correlação temporal entre progressões de acordes e realizar suas transcrições finais. Ambos os algoritmos eram treinados separadamente. Quando submetido ao MIREX, este sistema alcançou níveis de desempenho com cerca 10% de sucesso na tarefa de transcrição de acordes.

Para complementar o escopo de nossas avaliações, o caminho ideal indicava a utilização de pelo menos um sistema de transcrição com desempenhos intermediários entre os que conseguimos acesso para testes. Diante do cenário de dificuldade de acesso a um algoritmo com estas características, e também com o objetivo de buscar uma melhor compreensão do problema de transcrição no mais baixo nível, além da possibilidade de nos depararmos com situações que exigissem uma mais livre manipulação de um algoritmo de transcrição de acordes, optamos pelo desenvolvimento de um sistema de transcrição próprio. Sem o intuito de encontrar uma solução ótima, a ideia seria buscar um transcritor capaz de alcançar resultados convenientes às nossas necessidades de avaliação. O que buscamos, minimamente, foi um transcritor capaz de gerar resultados de sucesso em transcrições que pudessem ser enquadrados, por exemplo, entre os resultados do algoritmo top-MIREX e do algoritmo bottom-MIREX selecionados para as avaliações.

Neste desenvolvimento, partimos do mesmo principio utilizado pela ampla maioria dos algoritmos de transcrição de acordes atuais. Utilizando a teoria da extração do vetor *chroma* em sua fundamentação original, quando o mesmo foi referenciado como vetor PCP (FUJISHIMA, 1999), com a aplicação prévia de filtros que enfatizaram as frequências mais graves do sinal de áudio, entre 50hz e 400hz, implementamos um sistema que realizava a extração deste vetor de características a cada dois décimos de segundo a partir de um arquivo de áudio de uma canção, para depois aplicá-lo em um algoritmo de aprendizagem de máquina que pudesse aprender a reconhecer estes padrões como pertencentes a um ou outro tipo de acorde.

Como o intuito era ter em mãos, e de uma forma manipulável, um algoritmo de transcrição de acordes que não precisaria ter desempenho ótimo, não houve uma preocupação maior com sua otimização. Optamos por desenvolvê-lo baseando-nos em uma rede neural do tipo MLP (*Multilayer Perceptron*), com o uso do algoritmo

clássico de aprendizagem *backpropagation* (RUMELHART; HINTON; WILLIAMS, 1986), que por não ter sido ainda avaliado, ao menos pelo que tivemos conhecimento, na tarefa de transcrição de acordes, poderia trazer alguma contribuição para a literatura da área.

Para executar os treinamentos e testes com a rede neural, utilizamos o pacote do Matlab (THE MATHWORKS, 2015), com o qual fizemos vários treinamentos com duas variações do algoritmo de aprendizagem *backpropagation* disponibilizadas no mesmo: o *Resilient Backpropagation* (RIEDMILLER, 1993) e o *Scaled Conjugate Gradient Backpropagation* (MOLLER, 1993). Além disso, foram utilizadas diferentes configurações para a estrutura da rede neural. O corpus de dados para treinamento, avaliação e testes foi formado pelas canções contidas no conjunto “A”, definido na seção 7.1.2. Na prática, para cada uma das canções do corpus foi disponibilizado um arquivo WAV, amostrado em 16.000hz, com 16bits e mono, e um arquivo em formato texto com a identificação dos acordes executados em cada canção por intervalo de sua duração no áudio.

De posse desta massa de informações, foi desenvolvido um protótipo no Matlab que executou o seguinte procedimento:

1. Leitura e recuperação das informações de áudio de um arquivo WAV seguida da aplicação de filtros que enfatizaram as frequências graves no intervalo entre 50Hz e 400hz;
2. Separação do sinal de áudio da canção em intervalos de 0,2 segundos, seguido do cálculo, para cada um destes intervalos, dos vetores *chroma* (PCP´s) de acordo com o padrão básico definido no trabalho original de Fujishima (FUJISHIMA, 1999);
3. Associação de cada vetor *chroma* calculado com cada acorde em execução no momento em que os vetores foram extraídos, criando assim, um mapeamento por intervalo de 0,2 segundos, de cada acorde em execução com o seu respectivo vetor *chroma*;
4. Criação de novos arquivos para cada canção, agora contendo para cada intervalo de 0,2 segundos de sua execução, seus vetores *chroma* e os respectivos acordes (Figura 8.1). Estes arquivos serão referenciados como Arquivos Pré-formatados da rede neural.

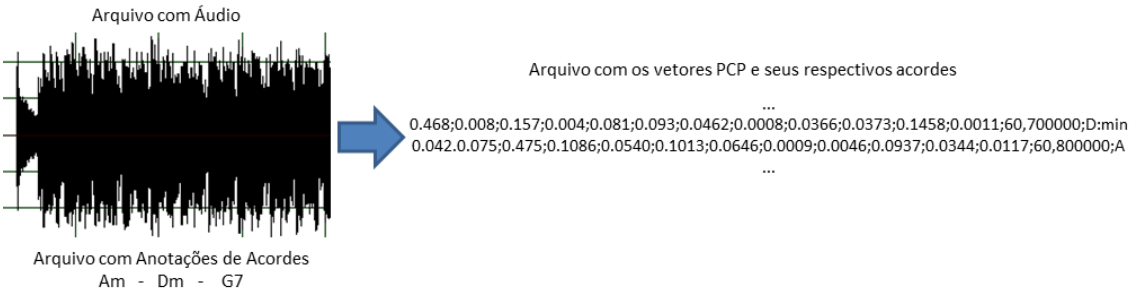


Figura 8.1 - Arquivo Pré-Formatado da Rede Neural: PCP´s x Acordes

Após a criação destes arquivos, a ideia seria tentar fazer com que a rede neural aprendesse a reconhecer acordes recebendo como entrada os vetores *chroma*, e como saída desejada os acordes relacionados com os mesmos. Para tanto, ainda foram executados os seguintes procedimentos de preparação destes dados que seriam utilizados para o treinamento, validação e teste da rede neural:

1. Leitura e carregamento dos dados de cada um dos arquivos Pré-Formatados da rede neural (ver passo 4 dos procedimentos descritos anteriormente neste mesmo capítulo), e execução da codificação de cada acorde vinculado ao vetor *chroma* (saída desejada da rede) segundo o padrão descrito e identificado nas Tabelas 7.1 e 7.2;
2. Recriação de cada um destes arquivos de cada canção, agora contendo em cada linha um vetor *chroma* e o seu respectivo acorde devidamente codificado;
3. Junção dos registros de todos os arquivos de vetores *chroma* x acordes codificados do conjunto de canções “A” em um novo único arquivo. Este processo criou uma fonte de dados única para o processo de treinamento, validação e teses da rede neural a ser avaliada nesta seção.

Após a definição da fonte de dados, o próximo passo foi a configuração das redes neurais que iriam ser treinadas para realizar o reconhecimento de acordes tomando como base esta massa de dados.

Utilizando o Matlab, a partir deste ponto foi criado um procedimento experimental que obedeceu aos seguintes passos:

1. Divisão do corpus de canções “A” (que neste momento já estava todo contido em um arquivo único com cada canção separada e devidamente organizada pelos seus respectivos vetores *chroma* calculados e respectivos acordes vinculados e codificados) no conjunto de treinamento, com aproximadamente 70% das canções, conjunto de testes, formado por aproximadamente 15% das canções, e conjunto de validação, formado pelo restante das canções, a fim de aplicar *Early Stopping* (KIRAN; DEVI; LAKSHMI, 2010);
2. Foi definido que o número máximo de épocas de treinamento seria de 800. Este número foi definido a partir de pré-testes de treinamentos que indicaram um valor aproximado para o mesmo. Ao término dos processos de aprendizagem, menos de 3% dos treinamentos atingiram este valor;
3. A fim de prevenir o *overfitting*, o treinamento seria finalizado após 8 validações consecutivas do conjunto de validação (valor padrão do processo de treinamento no Matlab) com sua taxa de erro em crescimento. Neste caso, seria assumida como válida a última configuração da rede antes do início do aumento da curva de erros nos testes sobre o conjunto de validação (KIRAN; DEVI; LAKSHMI, 2010);
4. Manutenção dos valores *default* dos demais parâmetros e recursos disponibilizados para configurar a rede;
5. Sabendo que o conjunto de treinamento seria composto por cerca de 18.000 exemplos, com uma rede com 12 entradas e 28 saídas, foram realizadas simulações com redes com uma única camada intermediária com quantidade de neurônios variando entre 50 e 450 neurônios, com saltos de 10 unidades. Os limites de variação do número de neurônios da camada intermediária foram escolhidos a partir de indicações da literatura da área (Bartlett, 1998), (Baum, et al., 1989), (Bartlett, et al., 2003), (Koiran, et al., 1997), (Guyon, et al., 1993), (Vapnik, 2000), (Boger, et al., 1997), (Berry, et al., 2004);
6. Seguindo a orientação do manual técnico do Matlab em sua seção “*Improve Neural Network Generalization and Overfitting*”, foram realizados 10 re-treinamentos para cada configuração de rede com um

determinado número de neurônios na camada intermediária a fim de garantir que uma rede com boa generalização fosse encontrada;

7. Ao término do processo, realização do cálculo do MSE médio sobre o conjunto de testes e do seu desvio padrão para cada tamanho de rede, além da seleção da rede com o melhor desempenho.

Após a execução do plano de treinamento e testes[16], os melhores resultados foram alcançados com o algoritmo *Resilient Backpropagation*, com a utilização de 430 neurônios na camada intermediária. O MSE médio calculado para esta configuração foi de 0,06435 com um desvio padrão de 0,002313, e o número de épocas médio foi de 752,50 com um desvio padrão de 49,23.

Após a identificação da rede de melhor desempenho, partimos para a realização dos testes na prática, quando a rede treinada seria utilizada como parte do processo de transcrição de acordes. Para avaliar a sua capacidade de contribuir com o processo de transcrição, a rede treinada foi utilizada para reconhecer os acordes de todas as canções do conjunto de testes referenciado como “B”. Além disso, para auxiliar o processo de transcrição de acordes, sobre este mesmo conjunto também foi aplicado um algoritmo de detecção de andamento (beats) (DAVIES; PLUMBLEY, 2007), implementado no software *Sonic Visualiser* (CANNAM *et al.*, 2006). Como já descrito, é pouco provável que haja uma mudança de acordes dentro de um intervalo de dois *beats*, e por isso, identificar o momento em que a marcação de um *beat* ocorre pode garantir uma melhora nas transcrições porque pode ajudar a eliminar acordes erráticos que eventualmente aparecem dentro de um intervalo de *beats*. Quando este tipo de situação ocorreu, a ideia foi a de selecionar o acorde com maior frequência de ocorrência entre cada par de *beats* e assumir que este seria o único acorde presente no intervalo. Em suma, os reconhecimentos dos acordes realizados sobre o conjunto de testes “B”, aliados aos pré-processamentos executados com os filtros aplicados, e aos processamentos realizados com a identificação dos *beats*, definiram o procedimento experimental realizado nos testes de nosso algoritmo de transcrição de acordes.

[16] Todas as avaliações e passos do processo de treinamento foram realizados em uma máquina Dual Xeon 3.4Ghz, 34GbRAM e duraram cerca de 150horas

Com as transcrições feitas sobre todos os arquivos do conjunto “B”, foi aplicada a mesma métrica do MIREX, a distância de *hamming* (ABDALLAH *et al.*, 2005), com o objetivo de avaliar o desempenho final da transcrição.

No cálculo da distância de *hamming* são considerados, inicialmente, todos os segmentos que particionaram a canção em dois décimos de segundo. Para cada acorde real da canção é calculada a quantidade de segmentos que os mesmos ocupam. Este conjunto de segmentos é o que chamamos de grupo S_G. Em seguida, para cada acorde transcrito da canção é novamente calculada a quantidade de segmentos que os mesmos ocupam. Este conjunto é referenciado como S_M. A distância de *hamming* é calculada pela contagem do número de segmentos que fazem parte do conjunto de intersecção pela igualdade de acordes entre estes dois conjuntos S_G e S_M. Em outras palavras, a distância de *hamming* é calculada pela contagem do número de segmentos em que o conjunto S_M (acordes transcritos) possui o mesmo acorde que o conjunto S_G (acordes corretos). Caso a intersecção seja total, ou seja, caso todos os segmentos se apresentem com o mesmo acorde, tanto para o conjunto S_G quanto para o conjunto S_M, teremos uma transcrição sem erros, e a distância calculada corresponderá ao tamanho, em número de segmentos, de toda a canção. Para calcular o percentual de acertos total, a distância de *hamming* é normalizada através da divisão da mesma pelo número total de segmentos de dois décimos de segundo da canção inteira em análise.

Ao término dos testes, com a utilização da rede neural e aplicação do algoritmo de detecção de *beats*, a distância de *hamming* foi calculada para todos os arquivos de testes do conjunto “B” e o nosso transcritor de acordes alcançou o percentual médio de sucesso de 52,18% na tarefa de transcrição sobre todo este corpus de testes.

Este resultado enquadra o nosso algoritmo de transcrição num nível que pode ser considerado intermediário, sobretudo quando comparado com aqueles submetidos ao MIREX nos últimos anos. Entretanto, como já mencionado, como nosso objetivo era o de utilizar as saídas deste algoritmo para testar e “calibrar” o nosso verdadeiro modelo, nem sua importância, nem sua utilidade foi diminuída dentro do contexto do objetivo geral do nosso trabalho, que seria o de usá-lo na avaliação do modelo de pós-processamento. Além disso, este resultado intermediário, como será visto no Capítulo 10, onde acontecerão as avaliações do modelo de pós-processamento, terá um papel importante para as análises que serão

feitas. Neste sentido, não houve maiores preocupações com a busca por maiores otimizações da rede em busca do melhor desempenho da mesma no reconhecimento de acordes.

9. O Módulo Decisor

Após o desenvolvimento do Módulo de Previsão de Acordes do sistema de pós-processamento, composto pela rede neural treinada para a realização de previsão de acordes e pelo *Sequence Tracker – (ST*), o próximo passo seria o desenvolvimento do Módulo Decisor que iria ser responsável pela tomada de decisão de quando as sugestões de correção de transcrição feitas pelo módulo de previsão de acordes deveriam ser consideradas em detrimento às transcrições realizadas originalmente pelo sistema de transcrição genérico aplicado ao pós-processamento.

Com a composição do módulo de previsão de acordes a partir de uma rede neural de previsão e do ST, o módulo de decisão precisaria estar preparado para tratar as sugestões de correções feitas por estes dois algoritmos, e para tanto, precisaria também estar preparado para lidar com eventuais conflitos entre as sugestões de ambos. Na prática, como a proposta do ST seria indicar a presença de uma sequência de acordes em repetição, e fazer isso com relevante grau de certeza, a ideia seria a de que a atuação do ST deveria prevalecer sobre qualquer sugestão feita pela rede neural de previsão de acordes. Neste sentido, a atuação da rede neural e do ST ocorreria de forma concorrente, porém em casos de sugestões simultâneas e diferentes de ambos, o ST teria sempre prioridade a qualquer processo de previsão realizado pela rede neural.

Além desta definição, havia um ponto mais importante a ser discutido e que está relacionado com a relevância da sugestão de correção de transcrição feita pelo módulo de previsão de acordes, seja através da atuação da rede neural de previsão, seja pela atuação do ST. Na verdade, a questão a ser abordada seria a de quando esta sugestão deveria ser considerada mais relevante do que a transcrição original. A resposta a esta questão foi encontrada através de procedimentos experimentais que serão discutidos nas próximas seções.

9.1. O Módulo Decisor com Sugestões da Rede Neural

O funcionamento do Módulo Decisor foi inicialmente modelado a partir da definição de como seria o seu comportamento em relação às sugestões de correções feitas pela rede neural de previsão de acordes. Como já havíamos

sugerido previamente, a ideia seria a de que o Módulo Decisor deveria optar pelas sugestões da rede neural apenas se a mesma gerasse um relevante “percentual de certeza” sobre suas saídas. Neste sentido, em primeiro lugar seria preciso definir como calcular este percentual de certeza, para depois definir qual o limiar ideal do mesmo para que ele devesse ser considerado mais relevante do que qualquer sugestão feita pelo algoritmo de transcrição.

9.1.1. Cálculo do Percentual de Certeza (PC) da Rede Neural

Assumindo que $X = \{x_1, x_2, ..., x_{84}\}$ e $Y = \{y_1, y_2, ..., y_{28}\}$, são respectivamente, os vetores de entrada e saída da rede neural configurada para receber como entrada uma janela de três acordes, e considerando ainda que a matriz “A” a seguir corresponde às codificações binárias de cada tônica de acordes (cada linha é uma codificação de uma tônica):

Matriz A:

$$\begin{vmatrix} a_{1-1} & a_{1-2} & a_{1-3} & \dots & a_{1-12} \\ a_{2-1} & a_{2-2} & a_{2-3} & \dots & a_{2-12} \\ . & & & & \\ . & & & & \\ . & & & & \\ a_{12-1} & a_{12-2} & a_{12-3} & \dots & a_{12-12} \end{vmatrix}$$

E que a matriz T a seguir corresponde às codificações binárias de cada tipo de acorde (cada linha é uma codificação de um tipo de acorde):

Matriz T:

$$\begin{vmatrix} t_{1-1} & t_{1-2} & t_{1-3} & \dots & t_{1-16} \\ t_{2-1} & t_{2-2} & t_{2-3} & \dots & t_{2-16} \\ . & & & & \\ . & & & & \\ . & & & & \\ t_{16-1} & t_{16-2} & t_{16-3} & \dots & t_{16-16} \end{vmatrix}$$

Para definirmos uma forma de cálculo que indique a relevância da previsão realizada pela rede neural, sugerimos inicialmente o cálculo de um Indicador de Certeza (IC) de acordo com a Equação 9.1 a seguir.

$$IC_{jk} = \frac{\sum_{i=1}^{12} \mathrm{abs}(y_i - a_{ji}) + \sum_{i=13}^{28} \mathrm{abs}(y_i - t_{ki-12})}{28} \quad \textbf{(9.1)}$$

O Indicador de Certeza calculado pela Equação 9.1 determina para uma hipotética saída da rede Y o quanto uma determinada combinação de tônica (a_j) e tipo de acorde (t_k) podem estar aproximados numericamente desta mesma saída. Para encontrar a tônica e tipo de acordes que mais se aproximam da saída rede, deverão ser aplicadas nesta fórmula todas as combinações de valores existentes de *j* (tônicas), ou seja, os valores 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 e 12, cruzando as mesmas com todas as combinações de valores de *k* (tipos dos acordes), ou seja, os valores 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 e 16. O acorde com sua tônica e tipo ideais, ou seja, que mais se aproximam da saída Y da rede, será aquele que gerar o menor valor de IC segundo esta fórmula. A ideia, portanto, é encontrar a j-ésima linha da matriz A (a tônica do acorde da linha j de A) e a k-ésima linha da matriz T (o tipo do acorde da linha k de T) que gerem o menor valor do IC. Este IC calculado poderia, por sua vez, ser transformado num Percentual de Certeza (PC) da rede em relação ao acorde e tipo previstos. A Equação 9.2 define como calcular este percentual.

$$\textbf{PC = 100 x (1 – IC)} \qquad \textbf{(9.2)}$$

Um exemplo de como o cálculo do PC é realizado, é ilustrado na Figura 9.1. Os passos detalhados são descritos a seguir:

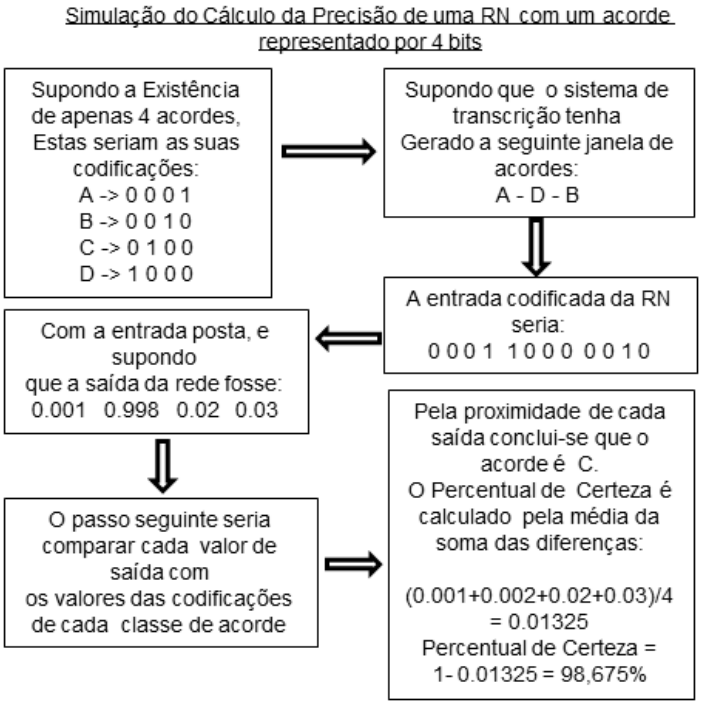


Figura 9.1 - Simulação do cálculo da precisão e percentual de certeza na previsão da rede neural

Detalhamento do procedimento descrito na Figura 9.1:

1. Após a previsão realizada pela rede neural do pós-processador, calculamos os módulos das diferenças das saídas geradas por cada neurônio da camada de saída da rede com as posições respectivas e correspondentes de cada bit de codificação de cada tônica e tipo de acorde existente, como indica a Equação 9.1;
2. Para todas as tônicas e tipos de acordes, os módulos de todas as diferenças são somados e o resultado é dividido pela quantidade de neurônios de saída (dimensão de cada acorde). Este procedimento define o cálculo dos Indicadores de Certeza de todos os possíveis acordes (representados por suas tônicas e tipos) (Equação 9.1);
3. Será selecionado como tônica e tipo de acorde a serem previstos aqueles que gerarem o menor Indicador de Certeza;
4. O cálculo do Percentual de Certeza da previsão é encontrado subtraindo-se de 1 o Indicador de Certeza calculado no Passo 3 e multiplicando-se o resultado por 100, como indica a Equação 9.2.

O algoritmo completo do cálculo do PC poderia ser escrito em pseudocódigo da seguinte forma:

```
// A: Matriz de codificação das tônicas (12 bits/tônica)
// T: Matriz de codificação dos tipos dos acordes (16 bits/tipo)
// Y: Vetor de Saída da Rede (28 bits)
// X: Vetor de Entrada da Rede (84 bits)

IC <- 0
Y = RN(X)
Para j de 1 ate 12 faca
     Para k de 1 ate 16 faca
          //Subtraindo cada vetor binário

          Somatorio <- 0

          //Tônicas

          Para i de 1 ate 12 faca
             Somatorio <- Somatorio + abs(Y[i] - A[j][i])
          FimPara

          //Tipos dos Acordes

          Para i de 13 ate 28 faca
             Somatorio <- Somatorio + abs(Y[i] - T[k][i-12])
          FimPara

          Se (j = 1 e k = 1) OU IC > Somatorio Entao
             IC = Somatorio
             Tonica = j
             Tipo = k
          FimSe
     FimPara
FimPara
IC <- IC / 28
PC <- 100 x (1 - IC)
```

Partindo deste padrão de cálculo do Percentual de Certeza (PC) da previsão de cada acorde realizada pela rede neural do pós-processador, o próximo passo seria definir qual o limiar ideal de valor para este percentual para que uma previsão da rede neural pudesse ser considerada em detrimento ao trabalho de um transcritor de acordes. Este limiar será referenciado como Limiar de Percentual de Certeza, ou LPC.

9.1.2. Calculando o Limiar de Percentual de Certeza - LPC

O cálculo do valor ideal para o percentual de certeza da rede neural, o LPC, foi realizado a partir de um procedimento experimental. A ideia seria a de buscar este valor ideal calibrando o mesmo através da realização de diversos testes de experimentação. Estas experimentações, por sua vez, só poderiam ser realizadas se tivéssemos canções transcritas por um transcritor hipotético, a fim de que a rede neural pudesse atuar sobre as mesmas, calcular percentuais de certeza, e assim pudéssemos avaliar qual deveria ser o percentual ideal para que o que fosse previsto pela rede pudesse prevalecer sobre as transcrições originais. Como o

transcritor que desenvolvemos apresentou um desempenho intermediário em suas transcrições, optamos pela sua utilização para a realização dos experimentos que definiriam o LPC supondo que este fator poderia ajudar na busca por um valor próximo do ideal para o mesmo. Estes experimentos seguiram os seguintes passos iniciais:

1. Seleção do corpus de canções “B” definido na seção 7.1.2;
2. Aplicação do sistema de transcrição de acordes que desenvolvemos sobre todas as canções deste corpus a fim de que fossem gerados arquivos contendo as transcrições de acordes para cada uma das canções;
3. Carregamento dos dados de acordes transcritos para cada canção e agrupamento dos mesmos em janelas de três acordes (melhor tamanho previamente identificado para a rede neural de previsão)
4. Codificação dos acordes segundo o padrão definido nas tabelas 7.1 e 7.2;
5. Fornecimento dos acordes codificados de cada canção, agrupados por janelas de três acordes, como entrada para a rede neural. A Figura 9.2 descreve estes passos iniciais do procedimento experimental.

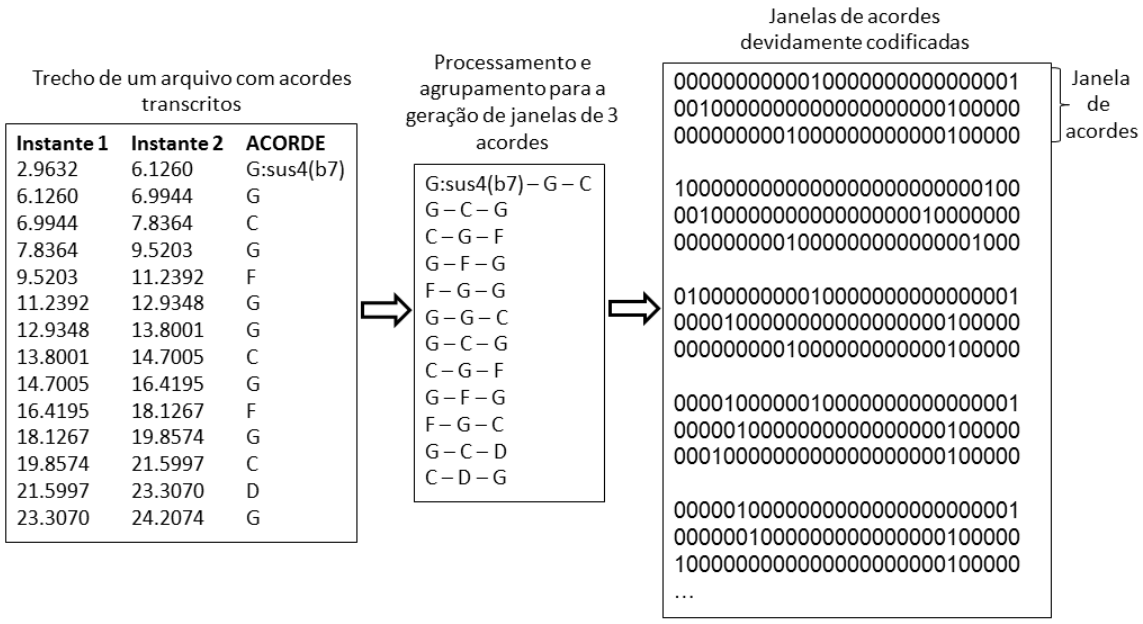


Figura 9.2 - Parte de arquivo de transcrições formatado x Agrupamento dos acordes em janelas de tamanho

Com a aplicação da rede neural para a realização das previsões a partir da recepção como entrada de sequências de três acordes já executados e originados das transcrições realizadas pelo nosso transcritor, o passo seguinte seria encontrar o percentual de certeza limite, o LPC, para que as sugestões da rede neural fossem consideradas mais relevantes do que as transcrições.

Como já sugerimos, a ideia foi a de defini-lo experimentalmente. Para tanto, começamos nossas avaliações e experimentos assumindo um LPC alto. Na prática, começamos assumindo que para que uma previsão realizada pela rede neural se sobrepusesse à transcrição do acorde original, o PC do acorde previsto teria que ser igual ou superior, inicialmente, a um LPC de 90%. Escolhemos este limiar de forma arbitrária e apenas como balizador de nossos experimentos.

Após a realização das primeiras simulações em todo o corpus de testes “B”, foi identificado que o LPC no patamar de 90% se mostrou muito exigente e com ele não conseguimos alcançar sucesso, ou melhor, não conseguimos que a nossa rede neural melhorasse o percentual de sucesso final das transcrições em relação ao que o transcritor realizou (sempre utilizando a métrica da distância de *Hamming* para comparar o desempenho da transcrição pura e da transcrição “corrigida” pelo preditor de acordes).

Com o insucesso dos testes com o LPC igual a 90%, demos continuidade no nosso procedimento experimental realizando simulações variando os valores do LPC com saltos decrescentes de uma unidade a partir de 90%. Até o patamar de 78% nenhuma melhora foi alcançada. Apenas a partir do valor de 77% para o LPC é que começamos a verificar algum crescimento no desempenho final da transcrição realizada pelo nosso modelo.

Ao final dos testes, o LPC ideal para ser adotado como medida de opção do que é previsto pela rede neural em detrimento do que é transcrito pelo algoritmo de transcrição de acordes que desenvolvemos, alcançou o patamar de 75%. Durante os experimentos realizados sobre o corpus de testes “B”, observamos que, com este LPC, a rede neural treinada para realizar previsão de acordes gerou sugestões de correções nas transcrições originais que levaram a uma diminuição de erros de em torno de 16% em relação à transcrição realizada puramente pelo transcritor de acordes. Este melhora foi verificada através da aplicação da mesma métrica baseada na distância de *Hamming*. Através dela, a transcrição original que teve um desempenho de 52,18%, com a aplicação do pós-processamento, passou a ter um

desempenho de 59,84%. A Tabela 9.1 resume os resultados detalhados alcançados pelos nossos experimentos.

LPC da Rede Neural	Diminuição do percentual de erros da transcrição de acordes em relação ao percentual original
Entre 90% e 78%	0,00%
77%	4,89%
76%	11,67%
75%	**16,01%**
74%	9,05%
73%	3,49%

Tabela 9.1 -Resultados de experimentos em busca da melhor precisão para a rede neural de previsão de acordes

Estes resultados indicaram, ao menos preliminarmente, que o nosso modelo de pós-processamento baseado apenas na rede neural de previsão de acordes, e na atuação do módulo decisor configurado experimentalmente pela definição do valor do LPC, se mostrou viável. Porém, para corroborar esta conclusão, seriam realizados outros experimentos com outros sistemas de transcrição de acordes.

9.2. O Módulo Decisor com Sequence Tracker (ST)

Na seção 7.2 foi descrito como o algoritmo do Sequence Tracker – ST, desenvolvido em nossas pesquisas anteriores (CUNHA, 1999)), foi adaptado para o cenário atual. Como já descrito no início deste capítulo, no funcionamento de nosso sistema de pós-processamento as eventuais sugestões de sequências de acordes em repetição feitas pelo ST teriam prioridade sobre qualquer sugestão feita pela rede neural de previsão de acordes, independentemente do Percentual de Certeza gerado para as suas saídas.

Com esta decisão tomada, partimos para realizar simulações com o objetivo de validar o desempenho do ST na identificação de eventuais sequências de acordes em repetição. Para os testes, utilizamos mais uma vez o conjunto de canções “B”, bem como as transcrições de acordes realizadas pelo nosso transcritor (seção 9.1) sobre cada uma das canções que compunham este conjunto. Para os testes, foi desenvolvido um protótipo em Matlab capaz de executar o seguinte plano:

1. Leitura e carregamento das informações de áudio de uma dada canção;
2. Aplicação do algoritmo de identificação de compassos e definição dos limites de cada provável compasso da canção;
3. Utilizando as transcrições realizadas pelo nosso transcritor, foi aplicado o algoritmo do ST na canção para que o mesmo tentasse identificar sequências de acordes em repetição. A ideia seria a de que o ST, no momento em que identificasse uma eventual sequência de acordes em repetição, substituísse cada próximo acorde oriundo da transcrição realizada pelo nosso transcritor, por cada acorde presente na sequência de acordes previamente armazenada pelo ST, e identificada como a sequência em repetição;
4. Após o término da atuação do ST, foi aplicada a métrica da distância de *hamming* sobre as suas transcrições corrigidas a fim de determinar o seu desempenho;
5. Repetição de todos os passos para cada uma das canções do corpus de testes “B”.

Após a realização de todos os testes e validações, chegamos à conclusão de que não possuímos informações suficientes para utilizar o ST. Os resultados indicaram que a ausência da informação da melodia e a falta de precisão dos limites dos compassos não permitiu a identificação de muitas sequências em repetição. E mesmo quando as sequências foram identificadas, as regras do ST nem sempre puderam ser utilizadas corretamente porque os limites de cada sequência em repetição dificilmente correspondiam exatamente com os limites da sequência original.

Além disso, como a identificação de sequências repetidas parte do principio de que sequências de acordes já executadas e armazenadas pelo ST tiveram que ser construídas a partir de acordes transcritos pelo nosso transcritor, e por isso não livre de erros, não há garantia de que, mesmo que uma sequência de acordes em repetição seja identificada, a substituição da mesma por uma sequência já executada leve a uma melhora do desempenho da transcrição, pois a própria sequência original já pode estar com erros oriundos da transcrição inicial.

Nas análises realizadas com o uso do ST, apenas em cerca de 3,70% das canções conseguimos melhorar o desempenho da transcrição de acordes final. Em 81,48% o ST simplesmente não conseguiu identificar sequências em repetição que obedecessem as suas regras, sobretudo devido às limitações impostas pela falta de precisão na identificação dos limites dos compassos. Por fim, em cerca de 14,82% das canções houve piora nos desempenhos das transcrições finais após o uso do ST.

De posse destes resultados, optamos por alterar a estrutura proposta inicialmente para o nosso modelo, com a retirada do módulo do ST, já que se o mesmo fosse utilizado, o desempenho do sistema pós-processamento teria fortes chances de ser negativamente influenciado. Na Figura 9.3 indicamos o novo modelo do pós-processamento.

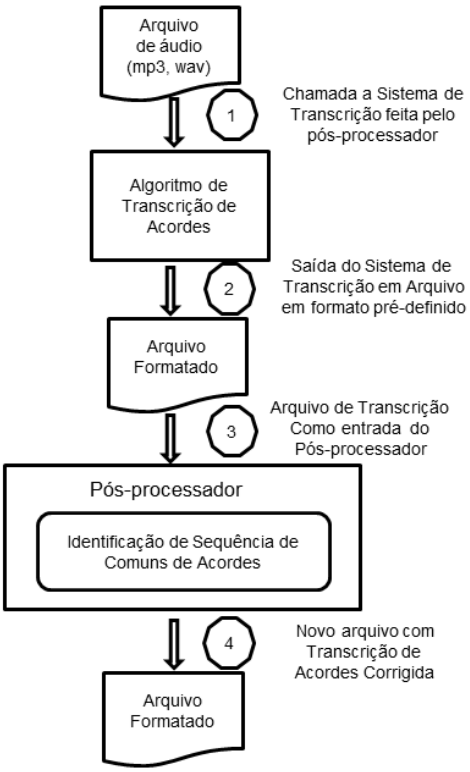


Figura 9.3 – Novo Modelo do Pós-Processador

9.3. Conclusões sobre o Módulo Decisor

Após todos os experimentos realizados, e com a conclusão de que o *Sequence Tracker*, ao menos com o modelo em que ele foi proposto, não conseguiria trazer melhoras de desempenho para o processo de transcrição de acordes, ficou definido que o nosso Módulo Decisor atuaria apenas tomando decisões baseadas na comparação dos Percentuais de Certeza das sugestões de correções de transcrição feitas pela rede neural de previsão de acordes em relação ao Limiar de Percentual de Certeza encontrado experimentalmente e assumido como ideal. Em outras palavras, apenas as sugestões de correções feitas pela rede neural e que tivessem Percentuais de Certeza iguais ou superiores ao Limiar de Percentual de Certeza, seriam consideradas em detrimento às transcrições originais.

10. Avaliando o Pós-Processamento da Transcrição de Acordes Usando Informações Musicais Preditivas

Neste capítulo vamos descrever os experimentos e resultados das avaliações realizadas após a atuação do nosso sistema de pós-processamento sobre as saídas de alguns sistemas de transcrição que selecionamos para testes. O objetivo destes experimentos era responder a questão de pesquisa RQ1, ou seja, analisar o impacto de nosso pós-processamento. Para tanto, e para ter uma ideia mais precisa de tal impacto, decidimos, como já discutido anteriormente o capítulo 8, fazer o pós-processamento na saída de três sistemas de transcrição de acordes: um com resultados muito bons no MIREX, outro com resultados fracos no MIREX, e um terceiro, que nós mesmos desenvolvemos para tornar a análise mais completa, e com resultados intermediários entre os dois iniciais.

10.1. Procedimento Experimental para os Testes

A natureza dos testes que foram realizados para avaliar o desempenho e capacidade de pós-processamento de nossa proposta foi similar a de testes A/B. Nestes testes foram comparados os desempenhos dos processos de transcrições de acordes sem e com a intervenção do sistema de pós-processamento. A execução dos procedimentos experimentais foi precedida pela definição dos seguintes pontos:

- Dados de Teste: Conjunto de canções “B”, definido na seção 7.1.2, e composto por 54 canções dos Beatles;
- Configuração do Sistema de Pós-Processamento:
    - Módulo de Previsão de Acordes: Rede Neural MLP;
    - Módulo Decisor: Percentual de Certeza estabelecido experimentalmente em 75% para todas as avaliações com os três sistemas de transcrição a serem utilizados;
- Sistemas de Transcrição: Três sistemas selecionados (ver Capítulo 8) com diferentes níveis de desempenho.

Após a definição destes pontos básicos, o procedimento experimental para a realização dos testes foi dividido entre aqueles do tipo “X”, sem a atuação de pós-

processamento, e do tipo “Y”, com a atuação do pós-processamento. O procedimento experimental dos testes do tipo “X” envolveu os seguintes passos:

1. Aplicação dos três sistemas de transcrição sobre os arquivos digitais de todas as canções do corpus “B”, com a geração como saída de arquivos em formato específico (HARTE *et al.*, 2005) contendo as transcrições dos acordes de cada canção avaliada.
2. Aplicação da métrica da distância de *hamming* (em relação ao *ground truth* dos acordes de cada canção) para cada arquivo gerado por cada sistema de transcrição de acordes, para as canções do corpus “B”;
3. O desempenho final da transcrição de cada um dos sistemas avaliados foi calculado pela média de sucesso de suas transcrições em todas as canções do corpus “B”. A média foi calculada entre as distâncias de *hamming* encontradas para cada canção transcrita;

Finalizado o procedimento experimental para os testes do tipo “X”, o procedimento experimental para os testes do tipo "Y" obedeceu aos seguintes passos:

1. Fornecimento como entrada do sistema de pós-processamento do arquivo de acordes transcritos da primeira canção do corpus e com transcrições geradas pelo primeiro sistema de transcrição de acordes a ser avaliado;
2. Pré-processamento de cada um destes arquivos de transcrições de forma que seus conteúdos fossem organizados no formato de janelas de três acordes;
3. Codificação de cada acorde para o formato esperado pelo Módulo de Previsão de Acordes definido pela rede neural MLP (ver Tabelas 7.1 e 7.2);
4. Fornecimento como entrada da rede neural de previsão de acordes de cada janela de três acordes (devidamente codificados), a fim de que a rede realizasse a previsão de qual deveria ser o próximo acorde;
5. De posse da saída da rede, a sua previsão, realização da classificação da mesma para a identificação do tipo e tônica do acorde previsto;

6. Cálculo do Percentual de Certeza da classe de acorde prevista e fornecimento deste percentual para o Módulo Decisor do sistema de pós-processamento;
7. Comparação do Percentual de Certeza encontrado com o LPC de 75% definido experimentalmente (ver seção 9.2.2);
8. Sugestão de correção do acorde previsto pela rede neural prevalecendo sobre a transcrição original para os casos em que o Percentual de Certeza calculado fosse igual ou superior a 75%.
9. Repetição dos passos 4 a 8 para todas as janelas de acordes da canção. A cada sugestão de correção aceita, a próxima janela de acordes a ser fornecida como entrada da rede passou a considerar o acorde corrigido em lugar do acorde transcrito originalmente. A ideia seria a de diminuir a propagação dos erros de transcrições;
10. Após a atuação do sistema de pós-processamento sobre todos os acordes transcritos e realização de eventuais correções pelas sugestões da rede neural, foi aplicada a métrica da distância de *hamming* a fim de avaliar os novos resultados de desempenho de transcrição final com a atuação do sistema de pós-processamento;
11. Todo o procedimento foi repetido para todas as canções transcritas (corpus “B”) para cada um dos três sistemas de transcrição;
12. O desempenho final da transcrição com o pós-processamento para cada um dos sistemas de transcrição foi calculado pela média de sucesso das transcrições finais para todas as canções do corpus “B” (sempre utilizando a métrica da distância de *hamming*);
13. Em busca de mais robustez nas avaliações, separamos o corpus de testes “B” em 6 grupos de 9 canções e calculamos os novos desempenhos alcançados pelo pós-processamento acoplado a cada um dos sistemas de transcrição e em cada um destes grupos individualmente. Com o término dos testes, foi calculado o desvio padrão para o desempenho de cada sistema acoplado ao pós-processamento;
14. O passo final das avaliações foi a realização de um teste de hipótese com o objetivo de avaliar a significância estatística dos resultados

encontrados quando o sistema de pós-processamento foi aplicado sobre o sistema de melhor desempenho no MIREX.

10.2. Resultados

Após a execução dos procedimentos experimentais descritos na seção 10.1, na Tabela 10.1 são indicados os resultados. Na coluna de Testes “X” são indicados os percentuais de erros médios de transcrição para cada um dos três sistemas de transcrição sem a atuação do sistema de pós-processamento. Na coluna de Testes “Y” são indicadas as diminuições percentuais de erros médios de transcrição para cada um dos três sistemas de transcrição com a atuação do sistema de pós-processamento.

Sistema de Transcrição de Acordes	**Nível de Desempenho de Transcrição**	**Testes “X” Percentual médio de erro de transcrição SEM o pós-processamento**	**Testes “Y” Redução percentual dos erros COM o pós-processamento**
Sistema HP (NI *et al.*, 2011)	ALTO DESEMPENHO	26,59%	2,93%
Nosso Transcritor	DESEMPENHO INTERMEDIÁRIO	52,18%	16,01%
Sistema baseada em RN e HMM (STEENBERGEN; BURGOYNE, 2013)	BAIXO DESEMPENHO	90,11%	7,45%

Tabela 10.1 – Desempenho do Pós-Processador atuando sobre as saídas de três algoritmos de transcrição de acordes

Nas tabelas 10.2 (sistema HP), 10.3 (nosso transcritor) e 10.4 (sistema baseado em RN e HMM) são indicados os percentuais de diminuição de erros com a atuação do pós-processador sobre as transcrições dos três sistemas avaliados sobre os 6 grupos de canções definidos no passo 13 do procedimento experimental.

Grupos	**Redução do percentual de erros da transcrição de acordes em relação ao percentual original**
1	1,48%
2	2,17%
3	3,79%
4	4,45%
5	1,89%
6	3,77%
Média	**2,93% - Desvio Padrão: 1,22**

Tabela 10.2 – Desempenho do Pós-Processador com a RN aplicada sobre as saídas do transcritor HP em 6 grupos de 9 canções tomados do conjunto de testes "B"

Grupos	Redução do percentual de erros da transcrição de acordes em relação ao percentual original
1	13,34%
2	17,38%
3	18,89%
4	17,91%
5	15,75%
6	12,79%
Média	**16,01% - Desvio Padrão: 2,50**

Tabela 10.3 – Desempenho do Pós-Processador com a RN aplicada sobre as saídas do nosso transcritor em 6 grupos de 9 canções tomados do conjunto de testes "B"

Grupos	Redução do percentual de erros da transcrição de acordes em relação ao percentual original
1	4,78%
2	8,77%
3	9,34%
4	8,91%
5	6,98%
6	5,89%
Média	**7,45% - Desvio Padrão: 1,85**

Tabela 10.4 – Desempenho do Pós-Processador com a RN aplicada sobre as saídas do transcritor baseado em RN e HMM em 6 grupos de 9 canções tomados do conjunto de testes "B

Para finalizar as nossas análises estatísticas, realizamos um teste de hipótese para avaliar a significância estatística dos resultados do pós-processamento sobre as transcrições do melhor sistema de transcrição avaliado, o sistema HP.

Sabendo que a média de sucesso na transcrição alcançada por este algoritmo atuando isoladamente foi de $\mu = 73,41\%$, com um desvio padrão de $\sigma = 3,245$, e que o erro nas transcrições foi diminuído em 2,93% com o uso do sistema de pós-processamento, o que elevou o percentual de desempenho da transcrição para $x = 74,19\%$, e agora com desvio padrão $\sigma_2 = 3,281$ (muito próximo do original), precisamos determinar se este acréscimo é estatisticamente significativo. Para tanto, assumiremos como H_0 a hipótese de que a média de transcrições de acordes seja de 73,41%, e como hipótese alternativa H_1 que a média seja diferente deste valor. O percentual de 74,19% foi alcançado em uma amostra de 56 canções. Sendo assim:

H_0: $\mu = 73,41\%$

H_1: μ ≠ 73,41%

Assumindo uma amostra de n = 54 canções, e um nível de significância α = 0,05, o que significa um Z_α tabelado em aproximadamente 1,64, a estatística Z do problema é dada pela seguinte expressão:

$$Z = \frac{\mathrm{x} - \mu}{\frac{\sigma}{\sqrt{n}}}$$

$$Z = \frac{74{,}19 - 73{,}41}{\frac{3{,}245}{\sqrt{54}}}$$

$$Z = 1{,}76$$

Com o valor de $Z > Z_\alpha$, e portanto dentro da região crítica, podemos assumir que a hipótese H_0 está rejeitada, ou melhor, que a um nível de significância de 5% existem evidências de que o percentual médio de sucesso da transcrição não permaneça em 73,41%, mas sim, num patamar superior. Assumindo outra interpretação, e chegando ao percentual correspondente a Z através de consulta a tabela normal, encontraremos o valor aproximado de 4%, o que significa que se a média de transcrições fosse realmente sempre de 73,41%, a probabilidade de encontramos uma amostra de 54 canções com média de transcrições de 74,19% ou maior, seria de 4%. Isto indica que, ou estamos diante de uma amostra muito incomum (4 em cada 100), ou então que a hipótese formulada não deve ser aceita. Diante deste cenário, somos levados a considerar a segunda opção, ou seja, a que indica que os dados da amostra de 54 canções sugerem que a hipótese H_0 seja rejeitada. Por ambas as análises, podemos considerar nosso resultado estatisticamente aceitável.

10.3. Análise dos Resultados

Os resultados alcançados indicam que, embora em níveis diferentes, nosso sistema de pós-processamento conseguiu melhorar o desempenho dos três sistemas de transcrição avaliados (CUNHA; RAMALHO; CABRAL, 2014). O melhor

nível de melhora foi verificado com a aplicação do mesmo sobre as transcrições realizadas pelo transcritor que desenvolvemos, e que apresentou, originalmente, resultados de transcrições enquadrados como intermediários.

Uma hipótese que poderia justificar este cenário seria o fato de que o desempenho do sistema de pós-processamento depende diretamente da qualidade das transcrições originais, já que são elas que alimentam o seu Módulo de Previsão. As transcrições pobres do sistema de baixo desempenho levarão a previsões pobres, pois os seus erros serão propagados. Por outro lado, as transcrições muito boas do sistema de alto desempenho, por si só, já trazem a dificuldade inerente que existe em qualquer tentativa de melhora acentuada em resultados que já estão em alto nível.

Neste sentido, a melhora alcançada pelo sistema de transcrição de nível intermediário pode ser justificada pela junção dos fatores que supomos terem levado à diminuição do desempenho do pós-processamento quando este foi aplicado aos sistemas de alto e baixo nível. Em primeiro lugar, o nível das transcrições do sistema intermediário apresenta uma coerência maior do que as do sistema de baixo desempenho. Por isso, ao alimentarem o Módulo de Previsão do pós-processamento, suas transcrições são capazes de extrair do mesmo previsões mais precisas e ricas, definindo um melhor comportamento do pós-processamento. Em segundo lugar, o desempenho original de transcrição do sistema que desenvolvemos ainda apresenta uma boa margem de melhora. Este fato permite que mais sugestões de correções possam ser feitas pelo pós-processamento, levando a uma melhora mais acentuada do desempenho final do mesmo quando aplicado a este sistema de transcrição. Transcrições mais coerentes induzem a pós-processamentos mais precisos. Esta precisão, aliada à boa margem ainda existente para melhora no desempenho das transcrições originais, induzem a uma melhor atuação do sistema de pós-processamento.

Além destes aspectos, também é importante ressaltar que as configurações do Módulo Decisor do sistema de pós-processamento foram feitas a partir de avaliações realizadas unicamente com o sistema de transcrição que desenvolvemos. Este aspecto também ajuda a justificar o melhor desempenho do pós-processamento quando acoplado a este sistema. Embora tenhamos conseguido alcançar nossos objetivos utilizando este formato de configuração do Módulo Decisor, ele também é uma indicação de que os resultados poderiam ser ainda

melhores se fossem feitas configurações do Módulo Decisor para LPC´s adequados para cada sistema de transcrição. Esta análise não foi feita, mas será objeto de pesquisas futuras.

Outra análise que poderia ser feita em relação a estes resultados é a de que eles foram alcançados considerando que o modelo de pós-processamento atua de forma independente do sistema de transcrição, que é visto como uma caixa preta. O seu propósito é avaliar o impacto real das informações musicais preditivas sobre o desempenho de sistemas de transcrições em geral, e atingiu o seu objetivo. Porém, o uso deste modelo elimina algumas possibilidades que poderiam enriquecer o cenário. Caso, por exemplo, o nosso modelo também pudesse acessar os processos de transcrição realizados por cada sistema de transcrição nele acoplado, poderia ser possível a realização de interferências nas suas decisões, o que poderia definir um novo nível de desempenho do pós-processamento. Este é um tipo de análise que também será objeto de pesquisas futuras e que tem um inconveniente muitas vezes impeditivo que é a necessidade de acesso aos códigos fontes dos sistemas de transcrição avaliados.

Por fim, após todas as análises de nossos resultados, e mesmo havendo indícios de que eles podem ainda ser melhorados, podemos concluir que o principal objetivo desta tese foi alcançado, pois conseguimos responder a nossa principal pergunta de pesquisa RQ1. Com o sistema de pós-processamento desenvolvido com o Módulo de Previsão constituído por uma rede neural treinada com o algoritmo *backpropagation*, e um Módulo Decisor configurado a partir de procedimentos experimentais, foi possível alcançar resultados bem sucedidos na tarefa de incorporar o conhecimento relativo às informações musicais de caráter preditivo (especificamente falando das sequências típicas de acordes) a um processo de transcrição de acordes com o intuito de enriquecê-lo e melhorar o seu desempenho.

11. Conclusão e Trabalhos Futuros

O trabalho que desenvolvemos se propôs a utilizar informações contextuais musicais de caráter preditivo, que incluíam as sequências típicas de acordes e as estruturas cíclicas, para com elas tentar melhorar o desempenho de processos de transcrição de acordes musicais em geral. Nas proposições iniciais feitas nesta tese foram formuladas três perguntas de pesquisa que definiram nossos objetivos, e cujas respostas levaram às conclusões de que o modelo proposto no trabalho obteve êxito em seus intentos. Eis as perguntas de pesquisa e suas respostas:

- RQ1: A incorporação do conhecimento relacionado com informações musicais contextuais preditivas, tais como sequências típicas de acordes e estruturas cíclicas, pode melhorar o desempenho de algoritmos de transcrição de acordes genéricos?

    - R-Nosso modelo de pós-processamento demonstrou que a resposta a esta pergunta é afirmativa para sequências típicas de acordes.

- RQ2: Existe uma maneira de capturar adequadamente as informações musicais de caráter preditivo?

    - Para o caso das sequências típicas de acordes a resposta é SIM. Foi possível identificar a presença das mesmas através de aprendizagem de máquina, mais especificamente, através de uma rede neural devidamente treinada.

- RQ3: Existe uma maneira de incorporar as informações musicais de caráter preditivo em algoritmos de transcrição de acordes?

    - Também para o caso das sequências típicas de acordes a resposta é SIM. No desenvolvimento de nosso Módulo Decisor, através do conceito de Limiar de Percentual de Certeza – LPC, foi definido um

meio de se incorporar o conhecimento relacionado com estas informações a um processo de transcrição de acordes.

Com relação às estruturas cíclicas, não conseguimos demonstrar, ao menos pelo método que utilizamos, se seria possível utilizar o conhecimento sobre as mesmas como forma de enriquecer um processo de transcrição de acordes.

Além destes resultados, é importante ressaltar que para alcançá-los foi proposto um modelo inovador capaz de incorporar as informações musicais contextuais relacionadas com as sequências típicas de acordes de uma forma complementar aos algoritmos já desenvolvidos para transcrição de acordes, e melhorando seus desempenhos. Utilizando técnicas da inteligência artificial clássica, aliadas a uma metodologia experimental, desenvolvemos um modelo que pode ser aplicado em soluções de alto nível e diretamente disponibilizadas para o usuário final. Nas próximas subseções detalharemos as contribuições, reflexões, conclusões e trabalhos futuros.

11.1. Contribuição

A principal contribuição para a área está relacionada com a confirmação de que, embora esta não seja uma tarefa fácil, é possível utilizar informações do contexto musical para enriquecer e melhorar o desempenho de um processo de transcrição de acordes. Numa maior ou menor escala, o modelo proposto em nosso trabalho demonstrou robustez em variadas formas de avaliação e testes, melhorando o resultado de uma transcrição quando lhe era possível, ou não atuando quando não havia total certeza para isso, não interferindo assim, negativamente nos resultados das transcrições. Esta é uma contribuição importante na medida em que aponta para novas possibilidades de algoritmos de transcrição que podem tirar proveito das informações preditivas citadas, assim como novos algoritmos para a captura e incorporação de tais informações.

11.2. Reflexões e Conclusões

Do trabalho desenvolvido, podemos refletir sobre muitos aspectos e tirar muitas conclusões. Corroborando os resultados alcançados em trabalhos anteriores

nossos, quando demonstramos que sequências típicas de acordes existentes em praticamente todas as canções podiam ser vistas como padrões de repetições generalizáveis para outras canções, mais uma vez conseguimos demonstrar, agora num contexto mais complexo, com acordes representados puramente por suas tônicas e tipos, que ainda assim, estes acordes retêm informações suficientes para a identificação da presença destas sequências.

Esta confirmação também indica que a tarefa de previsão de acordes realmente não é tão não determinística quanto parece, pois a capacidade de generalização demonstrada pela rede neural testada indica que existem regras "gerais" escondidas dentro das canções, sobretudo nas populares.

Por outro lado, fica claro que a falta de maiores informações do próprio contexto musical ainda não torna possível a viabilização do uso de um algoritmo como o *Sequence Tracker*, proposto em nossos trabalhos anteriores. A falta de informações de melodia e de estruturas básicas das canções, como compassos, por exemplo, e mesmo na ausência destas, a falta de algoritmos que consigam detectá-las com a precisão esperada, tornam o uso do *Sequence Tracker* praticamente inviável, ao menos da forma como o mesmo foi proposto originalmente. Na verdade, mesmo tentando viabilizá-lo de forma mais simplificada, eliminando as informações de melodia, e utilizando o auxílio de algoritmos de detecção de compassos, os resultados de suas atuações não se mostraram animadores, causando relevante perda de desempenho nas correções realizadas nas saídas do transcritor testado.

Outra questão relevante a ser ressaltada é que o desempenho do pós-processamento apresenta certa proporcionalidade em relação à qualidade do transcritor. Embora não tenhamos chegado a um fator exato desta proporcionalidade, a nossa experiência indica que, após a atuação do pós-processamento, algoritmos de transcrição com desempenho mediano tendem a ter suas transcrições melhoradas em um percentual mais alto do que algoritmos de transcrição com desempenhos em estado da arte, que já têm bons desempenhos. Como as previsões realizadas pelo nosso Módulo de Previsão de Acordes são alcançadas a partir de janelas de acordes já transcritas por um sistema de transcrição previamente executado, em muitos casos os erros gerados por estas transcrições vão interferir negativamente no desempenho da previsão e, consequentemente, no desempenho do pós-processamento. Por isso que também não podemos assumir ou até mesmo supor que o ganho de desempenho final, após

a atuação do sistema pós-processamento sobre transcrições realizadas por um algoritmo de baixo rendimento, deva ser num fator percentual tão relevante como o ganho de rendimento obtido sobre as transcrições de um algoritmo intermediário, cujas transcrições apresentam, normalmente, maior coerência. A nossa percepção indica que a aplicação de nosso sistema de pós-processamento sobre algoritmos de baixo rendimento não deve gerar grandes melhoras no desempenho final das transcrições. Por outro lado, a aplicação do pós-processamento sobre algoritmos intermediários, deve gerar melhoras mais relevantes, e sobre algoritmos em estado da arte, deve gerar pequenas melhoras.

Por fim, após todos os experimentos e resultados alcançados, parece-nos claro que em investigações futuras ainda será possível evoluir os resultados finais alcançados. Para tanto, na próxima seção detalhamos nossas próximas metas.

11.3. Trabalhos Futuros

Como trabalhos futuros para incrementar os resultados alcançados, podemos propor os seguintes:

- Realização de novos testes com codificações de acordes em formato binário e denso (CUNHA, 1999);
- Realização de testes com saídas da rede neural compostas por acordes codificados como números reais normalizados no intervalo entre zero e um;
- Avaliação do desempenho de redes neurais que tenham MSE´s ou Acurácias próximas daquelas alcançadas pelas redes de melhor desempenho, mas que possuam menor, ou até maior dimensão;
- Realização de simulações com outros algoritmos de aprendizagem, em especial, o HMM, no lugar da rede neural;
- Remodelagem do pós-processamento para que o mesmo possa funcionar utilizando mais de um algoritmo de previsão de acordes ao mesmo tempo a fim de que, numa interface interativa, ele possa fazer mais de uma sugestão de acordes em execução, considerando inclusive aqueles transcritos pelo sistema de transcrição nele acoplado;

- Realização de testes com corpus de canções de outros estilos musicais;
- Reavaliação do algoritmo do *Sequence Tracker* com o acréscimo de recursos como algoritmos de detecção de *beats*, detecção de melodia e ritmo das canções, além do já utilizado algoritmo de detecção de compassos;
- Criação de uma aplicação completa que possa utilizar todos os recursos do pós-processamento de transcritores de acordes;
- Alteração da forma de atuação do pós-processamento através de acesso aos processos de escolha de acordes a serem transcritos pelo sistema de transcrição nele acoplado;
- Alteração da forma de cálculo do LPC para que o mesmo seja gerado com valores específicos para cada sistema de transcrição que for acoplado ao sistema de pós-processamento.

REFERÊNCIAS

ABDALLAH, S. et al. **Theory and Evaluation of a Bayesian Music Structure Extractor**, 2005. 420-425 p.

BACKUS, J. **The Acoustical Foundations of Music**. 2nd edition. ed. New York: W. W. Norton and Company, Inc, 1977.

BARTLETT, P. L. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. **IEEE Trans. Inf. Theory**, 1998. 525–536.

BARTLETT, P. L. **The sample complexity of pattern classification with neural networks:** the size of the weights is more important than the size of the network. Information Theory, IEEE Transactions. p. 525-536.

BARTLETT, P. L.; MAASS, W. Vapnik Chervonenkis Dimension of Neural Nets, 2003. 1188-1192.

BAUER, B. In: **The New Royal Book**. 2nd Edition. ed. Petaluma: Sher Music CO., v. 1, 1988. p. 12, 61, 65-66, 307-308.

BAUM, E. B.; HAUSSLER, D. What size net gives valid generalization? **Neural computation**, 1, n. 1, 1989. 151-160.

BELLO, J. P. **Measuring structural similarity in music**. Audio, Speech, and Language Processing, IEEE Transactions. 2011. p. 2013 – 2025.

BELLO, J. P.; PICKENS, J. **A Robust Mid-level Representation for Harmonic Content in Music Signals**. 6th International Conference on Music Information Retrieval - ISMIR. London, UK. 2005. p. 304-311.

BENGIO, Y. et al. **Global optimization of a neural network-hidden markov model hybrid**. IEEE Transactions. 1992. p. 252–259.

BERRY, M. J.; LINOFF, G. S. **Data mining techniques:** for marketing, sales, and customer relationship management. 3nd Edition. ed. NY: John Wiley & Sons, 2004.

BETH, L. Music Recommendation from Song Sets. **5th International Conference on Music Information Retrieval - ISMIR**, Barcelona, Spain, 2004. 425-428.

BOGER, Z.; GUTERMAN, H. Knowledge extraction from artificial neural network models. **Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on.**, Orlando, FL, USA, 4, 1997. 3030-3035.

BOULANGER-LEWANDOWSKI, N.; BENGIO, Y.; VINCENT, P. **Audio Chord Recognition with Recurrent Neural Networks**. 14th International Society for Music Information Retrieval Conference - ISMIR. Curitiba, Brasil: 2013. p. 335-340.

BRACEWELL, R. The Fourier Transform and IIS Applications. In: **The Fourier Transform and its Applications**. 3rd. Ed. ed. New York: McGraw-Hill, 2000. Cap. Cap. 2, p. pp. 5 e 6.

BURGOYNE, J. A. et al. **A Cross-Validated Study of Modelling Strategies for Automatic Chord Recognition in Audio**. 8th International Conference on Music Information Retrieval - ISMIR. Vienna, Austria: 2007. p. 251-254.

BURGOYNE, J. A.; SAUL, L. K. **Learning harmonic relationships in digital audio with Dirichlet-based hidden Markov models**. 6th International Conference on Music Information Retrieval - ISMIR. London, UK: 2005. p. 438–443.

BURGOYNE, J. A.; WILD, J.; FUJINAGA, I. An Expert Ground Truth Set for Audio Chord Recognition and Music Analysis. **12th International Society for Music Information Retrieval Conference**, Miami, Florida, 2011. 633–38.

BURGOYNE, J. A.; WILD, J.; FUJINAGA, I. **An expert ground truth set for audio chord recognition and music analysis**. 12th International Society for Music Information Retrieval Conference - ISMIR. Miami, Florida: 2011. p. 633-638.

CABRAL, G.; BRIOT, J.-P.; PACHET, F. **Impact of Distance in Pitch Class Profile Computation**. Proceedings of the Brazilian Symposium on Computer Music. Sao Paulo, Brasil: 2005. p. 319-324.

CANNAM, C. et al. **The Sonic Visualiser:** A Visualisation Platform for Semantic Descriptors from Musical Signals. 14th International Society for Music Information Retrieval - ISMIR. 2006. p. 324-327.

CANNAM, C. et al. **Vamp Plugins from the Centre for Digital Music**. MIREX - Music Information Retrieval Evaluation eXchange. Curitiba, BR: 2013.

CANNAM, C. et al. **Vamp Plugins from the Centre for Digital Music**. MIREX. Curitiba, BR: 2013.

CANNAM, C. et al. Vamp Plugins from the Centre for Digital music. **MIREX - Music Information Retrieval Evaluation eXchange**, Taipei, Taiwan, 27-31 October 2014.

CHAFE, C.; JAFFE, D. **Source Separation and Note Identification in Polyphonic Music**. Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'86. California, USA: 1986. p. 1289-1292.

CHANG, C.-C.; LIN, C.-J. **LIBSVM:** a library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST). 2011. p. 27.

CHEDIAK, A. **Harmonia & Improvisaçao**. São Paulo: Irmãos Vitale, v. 1 e 2, 1986. 290 p.

CHEN, R. et al. **Chord Recognition Using Duration-explicit Hidden Markov Models**. 13th International Society for Music Information Retrieval Conference - ISMIR. Porto, Portugal: 2012. p. 445-450.

CHENG, T.; DIXON, S.; MAUCH, M. **A Deterministic Annealing EM Algorithm for Automatic Music Transcription**. 14th International Society for Music Information Retrieval Conference - ISMIR. Curitiba, Brasil: 2013. p. 475-480.

CHO, T.; BELLO, J. P. **A Feature Smoothing Method for Chord Recognition Using Recurrent Plots**. 12th International Society for Music Information Retrieval - ISMIR. Miami, USA: 2011. p. 651-656.

CHO, T.; BELLO, J. P. **Large Vocabulary Chord Recognition System Using Multi-Band Features and a Multi-Stream HMM**. MIREX. Curitiba, Brasil: 2013.

CHO, T.; WEISS, R. J.; BELLO, J. P. **Exploring Common Variations in State of the Art Chord Recognition Systems**. Proceedings of the Sound and Music Computing Conference (SMC). Utrecht, Netherlands: 2010. p. 1-8.

CORTES, C.; VAPNIK, V. Support-vector networks. **Machine Learning**, 20, n. 3, 1995. 273-297.

CUNHA, U. S. G. C. D. **Um Ambiente Híbrido Inteligente para Previsão de Acordes Musicais em Tempo Real, Dissertação de Mestrado**. Recife: Universidade Federal de Pernambuco, 1999. Dissertação de Mestrado.

CUNHA, U. S. G. C. D. **Um Ambiente Híbrido Inteligente para Previsão de Acordes Musicais em Tempo Real, Dissertação de Mestrado**. Recife: 1999. Dissertação de Mestrado.

CUNHA, U. S.; RAMALHO, G. An Intelligent Hybrid Model for Chord Prediction. **Organised Sound**, 4, n. 02, 1999. 115-119.

CUNHA, U. S.; RAMALHO, G. L.; CABRAL, G. **Algoritmo de Pós-Processamento para Sistemas de Transcrição de Acordes**. 12o Congresso de Engenharia de Áudio na 18a Convenção Nacional da AES Brasil. São Paulo: 2014. p. 67-74.

CUNHA, U. S.; RAMALHO, G. L.; CABRAL, G. **Algoritmo de Pós-Processamento para Sistemas de Transcrição de Acordes**. 12o Congresso de Engenharia de Áudio na 18a Convenção Nacional da AES Brasil. São Paulo: 2014.

DACCORD. Daccord Music, 2014. Disponivel em: <http://www.daccord.com.br/>.

DAVIES, M. E. P.; PLUMBLEY, M. D. A spectral difference approach to downbeat extraction in musical audio. **14th European Signal Processing Conference (EUSIPCO)**, Italy, 2006. 245-263.

DAVIES, M. E. P.; PLUMBLEY, M. D. **Context-dependent beat tracking of musical audio**. Audio, Speech, and Language Processing, IEEE Transactions. 2007. p. 1009-1020.

DIXON, S. **Audio Beat Tracking Evaluation:** BeatRoot. MIREX - Music Information Retrieval Evaluation eXchange. Victoria, Canada: 2006. p. 27.

ELLIS, D. P. W. **Simple Trained Audio Chord Recognition**. MIREX - Music Information Retrieval Evaluation eXchange. Philadelphia, USA: 2008. p. 65-69.

ELLIS, D. P. W.; POLINER, G. E. **Identifying cover songs with chroma features and dynamic-programming beat tracking**. Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference. Hawaii, USA: 2007. p. 1429-1432.

ELLIS, D. P. W.; WELLER, A. **Labrosa Chord Recognition System**. MIREX - Music Information Retrieval Evaluation eXchange. Utrecht, Netherlands: 2010. p. 45-48.

ELOWSSON, A.; FRIBERG, A. **Tempo Estimation by Modelling Perceptual Speed**. MIREX - Music Information Retrieval Evaluation eXchange. Curitiba, Brasil: 2013. p. 45-46.

FORNEY, G. D. **The Viterbi Algorithm**. Proceedings of the IEEE. 1973. p. 268-278.

FOX, D.; WEISSMAN, D. **Chord Progressions:** Theory And Practice: : Everything You Need to Create and Use Chords in Every Key. Alfred Music, 2013. 96 p. ISBN ISBN 978-0739070567.

FUJISHIMA, T. **Realtime Chord Recognition of Musical Sound:** a System Using Commom Lisp Music. ICMC - International Computer Music Conference. Stanford, USA: CCRMA. 1999. p. 464-467.

FUJISHIMA, T. **Realtime Chord Recognition of Musical Sound:** a System Using Commom Lisp Music. ICMC - International Computer Music Conference. Stanford, USA: CCRMA. 1999.

GOLD, B.; MORGAN, N. **Speech and Audio Signal Processing:** Processing and Perception of Speech and Music. 2nd Edition. ed. John Wiley & Sons, Inc., 2001.

GOMEZ, E. Tonal description of polyphonic audio for music content processings. **INFORMS Journal on Computing**, 18, n. 3, 2006. 294-304.

GOTO, M. **A Chorus-Section Detecting Method for Musical Audio Signals**. Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference. 2003. p. 437–440.

GUYON, I.; BOSER, B.; VAPNIK, V. Automatic capacity tuning of very large VC-dimension classifiers. **Advances in neural information processing systems**, 1993. 147-147.

HAAS, W. B. **Music information retrieval based on tonal harmony**. Utrecht University. 2012. PhD thesis.

HAAS, W. B. D.; MAGALHÃES, J. P.; WIERING, F. **Improving Audio Chord Trasncription by Exploiting Harmonic and Metric Knowledge**. 13th International Society for Music Information Retrieval Conference - ISMIR. Porto, Portugal: 2012. p. 295-300.

HARTE, C. A.; SANDLER, M. B.; GASSER, M. **Detecting harmonic change in musical audio**. Proceedings of the 1st ACM workshop on Audio and music computing multimedia. ACM. Santa Barbara, CA: 2006. p. 21-26.

HARTE, C. et al. **Symbolic Representation of Musical Chords:** A Proposed Syntax for Text Annotations. 6th International Conference on Music Information Retrieval. London, UK: 2005. p. 66-71.

HAYES-ROTH, B. A blackboard architecture for control. In: HAYES-ROTH, B. **Artificial Intelligence**. v. 26, n. 3, 1985. p. 251–321.

HAYKIN, S. **Neural Networks:** A Comprehensive Foundation. New York: Macmillian College Publishing Company, 1994.

HAYKIN, S. **Neural Networks:** A Comprehensive Foundation. 3nd Edition. ed. New York: Prentice Hall, 2008. 936 p. ISBN 978-0131471399.

HSU, C. W.; LIN, C. J. **A comparison of methods for multi-class support vector machines**. Neural Networks, IEEE Transactions. March 2002. p. 415–425.

HUANG, G.-B.; H. ZHOU, X. D.; ZHANG, R. Extreme Learning Machine for Regression and Multiclass Classification. **Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions**, 42, n. 2, 2012. 513-529.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. **Extreme Learning Machine:** A New Learning Scheme of Feedforward Neural Networks. Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference. 2004. p. 985-990.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: Theory and applications. **Neurocomputing 70**, 2006. 489–501.

HUMPHREY, E. J.; BELLO, J. P. **Rethinking Automatic Chord Recognition with Convolutional Neural Networks**. Machine Learning and Applications (ICMLA), 11th International Conference. Boca Raton, FL: 2012. p. 357 - 362.

ISMIR. MIREX. **MIREX**, 2011. Disponivel em: <http://www.music-ir.org/mirex/wiki/MIREX_HOME>.

KASHINO, N.; KINOSHITA, T. **Application of Bayesian probability network to music scene analysis**. Computational auditory scene analysis. 1995. p. 1-15.

KASHINO, T. **A sound source separation system with the ability of automatic tone modeling**. Proceedings of the International Computer Music Conference. INTERNATIONAL COMPUTER MUSIC ACCOCIATION: 1993. p. 248-248.

KASHINO, T. **A sound source separation system with the ability of automatic tone modeling**. International Computer Music Conference. 1993.

KATAYOSE, I.; INOKUCHI, S. The Kansei music system. **Computer Music Journal**, 1989. 72-77.

KHADKEVICH, M.; OMOLOGO, M. **MIREX Audio Chord Detection**. 9th International Conference on Music Information Retrieval. Philadelphia, USA: 2008. p. 34-38.

KHADKEVICH, M.; OMOLOGO, M. **Audio Chord Detection**. MIREX - Music Information Retrieval Evaluation eXchange. Utrecht, Netherlands: 2010. p. 10,15.

KHADKEVICH, M.; OMOLOGO, M. **Time-Frequency Reassigned Features for Automatic Chord Recognition**. Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference. Kyoto, Japan: 2011. p. 181-184.

KHADKEVICH, M.; OMOLOGO, M. **Time-Frequency Reassigned Features for Automatic Chord Recognition**. MIREX - Music Information Retrieval Evaluation eXchange. Curtiba, Brazil: 2013. p. 213-214.

KHADKEVICH, M.; OMOLOGO, M. **Time-Frequency Reassigned Features for Automatic Chord Recognition**. MIREX - Music Information Retrieval Evaluation eXchange. Taipei, Taiwan: 27-31 October 2014. p. 121-125.

KIRAN, N. V. N. I.; DEVI, M. P.; LAKSHMI, G. V. Training Multilayered Perceptions for Pattern Recognition: A Comparative Study of Three Training Algorithms. **International Journal of Information Technology and Knowledge Management**, 2, n. 2, n. 2, July-December 2010. 579-584.

KODERA, K.; GENDRIN, R.; VILLEDARY, C. **Analysis of time-varying signals with small bt values**. Acoustics, Speech and Signal Processing, IEEE Transactions. 1978. p. 64-76.

KOHAVI, R. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, p. 1137-1143, 1995.

KOHONEN, T. **Self-Organized Formation of Topologically Correct Feature Maps**. Biological cybernetics. 1982. p. 59–69.

KOIRAN, P.; SONTA, E. D. Neural networks with quadratic VC dimension. **Journal of Computer and System Sciences**, 54, n. 1, 1997. 190-198.

LEE, K.; SLANEY, M. **Automatic Chord Recognition from Audio Using an HMM with Supervised Learning**. 7th International Conference on Music Information Retrieval - ISMIR. Victoria, Canada: 2006. p. 133-137.

LEE, K.; SLANEY, M. **A Unified System for Chord Transcription and Key Extraction Using Hidden Markov Models**. 8th International Conference on Music Information Retrieval. Vienna, Austria: 2007. p. 245-250.

MANDIC, D.; CHAMBERS, J. **Recurrent Neural Networks for Prediction:** Learning Algorithms, Architectures and Stability. John Wiley & Sons, Ltd, 2002. ISBN ISBN 9780471495178.

MANNING, C. D. Foundations of statistical natural language processing, 1999.

MARTIN, K. **A Blackboard system for Automatic Transcription of Simple polyphonic Music**. Massachusetts. 1995.

MARTIN, K. **A Blackboard system for Automatic Transcription of Simple polyphonic Music**. Massachusetts Institute of Technology Media Laboratory Perceptual Computing Section Technical Report 385. Massachusetts. 1996. (n. 385).

MARWAN, N. et al. **Recurrence plots for the analysis of complex systems.** Physics Reports. p. 237–329, v. 438, n. 5. 2007.

MAUCH, M.; DIXON, S. **Approximate Note Transcription for the Improved Identification of Difficult Chords**. 11th International Society for Music Information Retrieval Conference. Utrecht, Netherlands: 2010. p. 135-140.

MAUCH, M.; DIXON, S. **Simultaneous estimation of chords and musical context from audio**. Audio, Speech, and Language Processing, IEEE Transactions. 2010. p. 1280-1289.

MAUCH, M.; NOLAND, K. C.; DIXON, S. **Using Musical Structure to Enhance Automatic Chord Transcription**. 10th International Conference on Music Information Retrieval - ISMIR. Kobe, Japan: 2009. p. 231-236.

MINSKY, M. L.; PAPERT, S. A. Perceptrons. **MIT Press**, Cambridge, Massachussets, 1969.

MIREX. MIREX. **MIREX-Music Information Retrieval Evaluation eXchange**, 2013. Disponivel em: <http://www.music-ir.org/mirex/wiki/MIREX_HOME>.

MIYAMOTO, K.; AL, E. **Separation of harmonic and non-harmonic sounds based on anisotropy in spectrogram**. IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP. 2008. p. 245-254.

MOLLER. Neural Networks. In: MOLLER **Neural Networks**. v. 6, 1993. p. 525–533.

MOLLER, M. F. A Scaled Conjugated Gradient Algorithm for Fast Supervised Learning. In: MOLLER, M. F. **Neural Networks**. v. 6, n. 4, 1990. p. 525–533.

MONT-REYNAUD, B. **The Bounded-Q Approach to Time-Varying Spectral Analysis**. Stanford University, CCRMA. California. 1985. (STAN-M-28).

MOORER, J. A. On the Transcription of Musical Sound by Computer. **Computer Music Journal**, Stanford, nov. 1975.

MOORER, J. A. On the Transcription of Musical Sound by Computer. **Computer Music Journal**, Stanford, nov. 1977. 32-38.

MURPHY, K. P. **Dynamic Bayesian Networks:** Representation, Inference and Learning - PhD thesis. University of California, Berkeley: 2002. Doctoral dissertation.

MUSIC, C. F. D.; COMPUTING, D. O. OMRAS2. **Sonic Annotator**, May 2013. Disponivel em: <http://omras2.org/SonicAnnotator>.

NGIAM, J. . C. Z. . C. D. . K. P. W. . L. Q. V. . &. N. A. Y. **Tiled convolutional neural networks**. Advances in Neural Information Processing Systems. 16 nov. 2010. p. 1279-1287.

NI, Y. et al. **Harmony progression analyzer for MIREX 2011**. MIREX - Music Information Retrieval Evaluation eXchange. Miami, Florida: 2011. p. 235-238.

NI, Y. et al. **Using Hyper-genre Training to Explore Genre Information for Automatic Chord Estimation**. 13th International Society for Music Information Retrieval Conference - ISMIR. Porto, Portugal: 2012. p. 109-114.

ONO, N. et al. **Separation of a monaural audio signal into harmonic/percussive components by complimentary diffusion on spectrogram**. Proc. EUSIPCO. 2008. p. 124-131.

OUDRE, L.; , C. F.; GRENIER, Y. **A Probabilistic Template-Based Chord Recognition Method**. Audio, Speech, and Language Processing, IEEE Transactions. Utrecht, Netherlands: 2011. p. 2249-2259.

OUDRE, L.; GRENIER, Y.; FÉVOTTE, C. **MIREX Chord Recognition System 2:** Major, Minor and Dominat Seventh Chords. MIREX - Music Information Retrieval Evaluation eXchange. Kobe, Japan: 2009. p. 312-313.

PAPADOPOULOS, H.; PEETERS, G. **Large-scale Study of Chord Estimation Algorithms Based on Chroma Representation and HMM**. Content-Based Multimedia Indexing, 2007. CBMI'07. International Workshop on. IEEE. 2007. p. 53–60.

PAPADOPOULOS, H.; PEETERS, G. **Chord Estimation Using Chord Templates and HMM**. MIREX - Music Information Retrieval Evaluation eXchange. Philadelphia, USA: 2008. p. 7-9.

PAPADOPOULOS, H.; PEETERS, G. **Audio Chord Estimation Task:** IRCAMCHORD. MIREX - Music Information Retrieval Evaluation eXchange. Utrecht, Netherlands: 2010. p. 435-436.

PAPADOPOULOS, H.; PEETERS, G. **Joint estimation of chords and downbeats from an audio signal**. Audio, Speech, and Language Processing, IEEE Transactions. Kobe, Japan: 2011. p. 138-152.

PATHAK, R. S. **The wavelet transform**. Amsterdam: Springer Science & Business Media, v. 4, 2009. ISBN ISBN 978-94-91216-24-4.

PAUWELS, J.; VAREWYCK, M.; MARTENS, J.-P. **Audio Chord Extraction Using a Probabilistic Model**. Abstract of the Music Information Retrieval Evaluation Exchange. Philadelphia, USA: 2008. p. 543-546.

PAUWELS, J.; VAREWYCK, M.; MARTENS, J.-P. **Audio Chord Extraction Using a Probabilistic Model**. Abstract of the Music Information Retrieval Evaluation Exchange. Kobe, Japan: 2009. p. 321-322.

PEETERS, G. **Chroma-based estimation of musical key from audio-signal analysis**. 7th International Conference on Music Information Retrieval - ISMIR. Victoria, Canada: 2006. p. 115-220.

PEETERS, G. Template-based estimation of time-varying tempo. **EURASIP Journal on Applied Signal Processing**, 2007, n. 1, 2007. 158-158.

PEETERS, G.; CORNU, F. **Audio Beat Tracking:** IRCAMBEAT. MIREX - Music Information Retrieval Evaluation eXchange. Curitiba: 2013. p. 89-90.

PIKRAKIS, A. **Audio Latin Music Genre Classification:** a MIREX 2013 sumission based on a Deep Learning Aproacho To Rhythm Modelling. MIREX - Music Information Retrieval Evaluation eXchange. Curitiba, Brazil: 2013. p. 23-24.

PISZCZALSKI, M.; GALLER, B. A. Automatic Music Transcription. **Computer Music Journal**, Menlo Park, CA, v. 1, p. 24-31, 1977.

PRIOLLI, M. L. Princípios Básicos da Música para Juventude. In: PRIOLLI, M. L. **Princípios Básicos da Música para Juventude**. 15a Edição. ed. Rio de Janeiro: Casa Oliveira de Músicas LTDA, v. 1o Volume, 1977.

RABINER, L. R. **A tutorial on Hidden Markov Models and selected applications in speech recognition**. Proceedings of the IEEE. 1989. p. 257-286.

RAO, C. R.; MITR, S. K. **Generalized Inverse of Matrices and its Applications**. New York: Wiley, v. 7, 1971.

REED, J. T. et al. **Minimum Classification Error Trainning to Improve Isolated Chord Recognition**. 10th International Society for Music Information Retrieval - ISMIR. Kobe, Japan: 2009. p. 609-614.

REFAEILZADEH, P.; TANG, L.; LIU, H. **Cross Validation - Encyclopedia of Database Systems (EDBS)**. Springer, v. 1, 2009. 356 p.

RIEDMILLER, M. **A direct adaptive method for faster backpropagation learning:** the RPROP algorithm. IEEE International Conference on Neural Networks (ICNN). San Francisco, CA: 1993. p. 586–591.

ROCHER, T. et al. **Dynamic chord analysis for symbolic music**. University of Michigan Library: Ann Arbor, MI: MPublishing, 2009.

ROLLAND, J.-B. **Chord Detection Using Chromagram Optimized by Extraction Additional Features**. MIREX - Music Information Retrieval Evaluation eXchange. Taipei, Taiwan: 27-31 October 2014. p. 43-44.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. **Nature**, v. 323, p. 533–536, 1986. ISSN doi:10.1038/323533a0.

SCHÖLKOPF, B.; BURGES, C.; SMOLA, A. **Advances in Kernel Methods - Support Vector Learning**. MIT Press, 1999.

SERRA, J.; SERRA, X.; ANDRZEJAK, R. G. Cross recurrence quantification for cover song identification. **New Journal of Physics**, 11, n. 9, 2009. 093017.

SHEH, A.; ELLIS, D. **Chord segmentation and recognition using EM-trained hidden Markov Models**. 4th International Conference on Music Information Retrieval, ISMIR. Baltimore, USA: 2003. p. 185-191.

SHEH, A.; ELLIS, D. **Chord segmentation and recognition using EM-trained hidden Markov Models**. 4th International Conference on Music Information Retrieval, ISMIR 2003. Baltimore, USA: 2003.

SONG, L.; LI, M. **Bayesian Framework-Based Vocal Melody Extraction**. Extended abstract submission to the Music Information Retrieval Evaluation eXchange (MIREX). Curitiba, Brazil: 2013. p. 211-212.

STEENBERGEN, N.; BURGOYNE, J. A. **Joint Optimization of an Hidden Markov Model - Neural Network Hybrid for Chord Estimation**. MIREX - Music Information Retrieval Evaluation eXchange. Curitiba, Brasil: 2013. p. 189-190.

SU, B.; JENG, S.-K. **Multi-timbre chord classification using wavelet transform and self-organized**. Acoustics, Speech, and Signal Processing, IEEE International Conference. 2001. p. 3377–3380.

SU, B.; JENG., S. **Multi-timbre chord classification using wavelet transform and self-organized**. IEEE International Conference on Acoustics. 2001. p. 3377–3380.

SUTTON, C.; MCCALLUM, A. **An introduction to conditional random fields for relational learning**. Introdution to Statiscal Relational Learning: MIT Press, 2006. 93-128 p.

THE MATHWORKS, I. MatLab, p. R2011a, 2015. Disponivel em: <http://www.mathworks.com/products/matlab/>.

TSOCHANTARIDIS, I. et al. **Support vector machine learning for interdependent and structured output spaces**. Proceedings of the twenty-first international conference on Machine learning. ACM. 2004. p. 104.

UCHIYAMA, Y. et al. **Automatic chord detection using harmonic sound emphasized chroma from musical acoustic signal**. Proc. ASJ Spring Meeting. Philadelphia, USA: 2008. p. 901-902.

UEDA, Y. et al. **MIREX 2010:** joint recognition of key and chord from music audio signals using key-modulation HMM. Proceedings of the Music Information Retrieval Evaluation Exchange (MIREX). Utrecht, Netherlands: 2010. p. 45-46.

VAPNIK, V. N. **The nature of Statistical learning theory**. New York: Springer Science & Business Media, 2000.

VIRO, V. http://www.peachnote.com. **Music Ngram Viewer**, 2011.

VIRO, V. PeachNote. **Music Ngram Viewer**, 2011. Disponivel em: <http://www.peachnote.com>.

WAKEFIELD, G. H. **Mathematical representation of joint time-chroma distributions**. SPIE's International Symposium on Optical Science, Engineering, and Instrumentation. International Society for Optics and Photonics. 1999. p. 637-645.

WEIL, J.; DURRIEU, J.-L. **An HMM-based audio chord detection system:** Attenuating the main melody. MIREX - Music Information Retrieval Evaluation eXchange. Pennsylvania, Philadelphia - USA: 2008. p. 165-166.

WELLER, A.; ELLIS, D.; JEBARA, T. **Structured Prediction Models for Chord Transcription of Music Audio**. Machine Learning and Applications, 2009. ICMLA'09. International Conference on. IEEE. Kobe, Japan: 2009. p. 590-595.

WU, M.-J. **MIREX 2013 Submissions for Train/Test Tasks (Draft)**. MIREX - Music Information Retrieval Evaluation eXchange. Curitiba, Brazil: 2013. p. 88-89.

YOSHIOKA, T. et al. **Automatic Chord Transcription with Concurrent Recognition of Chord Symbols and Boundaries**. 5th International Conference on Music Information Retrieval. Barcelona, Spain: 2004. p. 34-39.

ZENZ, V.; RAUBER, A. **Automatic chord detection incorporating beat and key detection**. Signal Processing and Communications. ICSPC 2007. IEEE International Conference on. IEEE. 2007. p. 1175-1178.

ZHANG, X.; LASH, C. **MIREX Audio Chord Detection**. 9th International Conference on Music Information Retrieval. Philadelphia, USA: 2008. p. 78-83.

ANEXO I

A seguir montamos uma tabela com toda a evolução dos trabalhos relacionados com a tarefa de transcrição de acordes propostos nos últimos quarenta anos.

Trabalho	Proposta	Numero de Acordes	Número de Classes	Tamanho do Corpus de Canções	Taxa de Acerto na Transcrição
(SU; JENG., 2001)	Wavelet + Rede Reural SOM	4 tipos de acordes	12 tônicas x 4 tipos de acordes = 48	480 amostras de sons do Quarto Movimento da Quinta Sinfonia de Bethoven	95%
(SHEH; ELLIS, 2003).	Vetor Chroma + HMM + Expatation-Maximization	7 tipos de acordes	21 tônicas x 7 tipos de acordes = 147	20 canções dos Beatles	22%
(YOSHIOKA *et al.*, 2004).	Detecção de Beats + Vetor Chroma + Heuríticas (Restrição de não mudança de tonalidade e uso de apenas tons maiores)	4 tipos de acordes	12 tônicas x 4 tipos de acordes = 48	7 canções em dó maior	77%
(LEE; SLANEY, 2006)	Vetor Chroma + HMM a partir de MIDI sintetizado	3 tipos de acordes	12 tônicas x 3 tipos de acordes = 36	140 segundos do Prelúdio em Dó Maior de Bach	93,35%
(HARTE *et al.*, 2005)	Proposta de modelo de acordes				
(Burgoyne, et al., 2007)	Conditional Random Fields + Vetor Chroma + Gaussianas/ Dirichlet	4 tipos de acordes	12 tônicas x 4 tipos de acordes = 48	20 canções dos Beatles	45% a 49%
(LEE; SLANEY, 2007)	Vetor chroma centróide tonal + HMM + Viterbi	2 tipos de acordes	12 tônicas x 2 tipos de acordes = 24	28 canções dos Beatles (dois álbuns)	72,7% a 74,36%
(PAPADOPOULOS; PEETERS, 2007)	Vetor chroma harmônico + HMM + Viterbi + teoria musical + estudos cognitivos + modelos gaussianos	2 tipos de acordes	12 tônicas x 2 tipos de acordes = 24	110 canções dos oito primeiros álbuns dos Beatles	71%
(ZENZ;	Vetores PCP +	3 tipos de	12	35 canções de	65%

RAUBER, 2007)	Identificação de Beats + Identificação de Tonalidade	acordes	tônicas x 3 tipos de acordes = 36	vários estilos (pop, rock e clássico)	
(UCHIYAMA *et al.*, 2008)	Vetores PCP + HMM + Viterbi + Supressão de frequências de instrumentos percussivos	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	176 cançãocanções dos Beatles	72%
(ELLIS; POLINER, 2007)	Vetores PCP + HMM + Viterbi + Aplicação de Filtros + detecção de Beats	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	176 cançãocanções dos Beatles	66%
(WEIL; DURRIEU, 2008)	Vetores PCP + HMM + Viterbi + vetores tonais centroides, com distribuição de acordes modelada como gaussianas + atenuação de frequências de melodia	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	176 cançãocanções dos Beatles	62%
(PAPADOPOULOS; PEETERS, 2008)	Vetores PCP (Transformada Q) + HMM + Viterbi	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	176 cançãocanções dos Beatles	63%
(KHADKEVICH; OMOLOGO, 2008)	Vetores PCP + HMM + Viterbi + Filtros entre 100hs e 2khz	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	176 cançãocanções dos Beatles	63%
(PAUWELS; VAREWYCK; MARTENS, 2008)	Vetores PCP + Modelo Probabilístico baseado em funções de densidade derivadas da métrica de distância tonal de Lerdahl	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	176 cançãocanções dos Beatles	59%
(ZHANG; LASH, 2008)	Vetores PCP + HMM + Viterbi + Gaussianas	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	176 cançãocanções dos Beatles	36%
(MAUCH; NOLAND; DIXON, 2009)	Vetores PCP + Detecção de Estruturas Cíclicas + Redes Bayesianas	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	206 cançãocanções (Beatles, Queen e Zweieck)	71,2%
(OUDRE;	Vetores PCP	15 tipos	12	206	71,1%

GRENIER; FÉVOTTE, 2009)	comparados com padrões de PCP de cada acorde considerando alguns harmônicos	de acordes	tônicas x 15 tipos de acordes = 180	cançãocanções (Beatles, Queen e Zweieck)	
(WELLER; ELLIS; JEBARA, 2009)	PCP + Support Vector Machine	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	206 cançãocanções (Beatles, Queen e Zweieck)	74,2%
(Reed, et al., 2009)	HMM + Viterbi +técnicas de separação entre harmonia e frequências percussivas	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	206 cançãocanções (Beatles, Queen e Zweieck)	70,1%
(PAUWELS; VAREWYCK; MARTENS, 2009)	Modelo probabilístico para extração do vetor PCP, a partir de frequências dos harmônicos	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	206 cançãocanções (Beatles, Queen e Zweieck	68,2%
(PAPADOPOULOS; PEETERS, 2011)	PCP + HMM com topologia que permitiu a modelagem da interdependência dos acordes com a estrutura métrica das canções + Viterbi + Detecção de beats	15 tipos de acordes	12 tônicas x 15 tipos de acordes = 180	206 cançãocanções (Beatles, Queen e Zweieck)	67,3%
(MAUCH; DIXON, 2010)	PCP baseado na resolução de problemas de mínimos quadrados não negativos (NNLS) + Detecção de Beats	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck	79%
(CHO; WEISS; BELLO, 2010)	PCP e Transformada Q + Filtragem de frequências graves + casamento de padrões + HMM + Viterbi	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck)	78%
(Khadkevich, et al., 2010)	PCP + HMM + Viterbi + Filtragem de sinal + padrões de acordes por gaussianas	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência	206 cançãocanções (Beatles, Queen e Zweieck)	78%

			de acorde) = 217		
(Ellis, et al., 2010)	2 vetores chroma + filtragem de sinal + Support Vector Machine	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck)	77%
(UEDA *et al.*, 2010)	PCP + HMM + Viterbi + Tonalidades Regionais	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck)	76%
(OUDRE; ; GRENIER, 2011)	PCP + Modelo probabilístico montado a partir do algoritmo EM (Expectation-Maximization)	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck)	74%
(OUDRE; ; GRENIER, 2011)	Grafo de encaminhamentos de acordes + Regras para determinar acordes candidatos	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck)	71%
(PAPADOPOULOS; PEETERS, 2010)	Vetores Chroma + HMM + Viterbi + conhecimento de contexto	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck	68%
(Ni, et al., 2011).	PCP + Separação de informações harmônicas percussivas+ detecção de tonalidade + Identificação de	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência	206 cançãocanções (Beatles, Queen e Zweieck)	98%

	beats + filtragem de sinal + HMM + Viterbi		de acorde) = 217		
(Cho, et al., 2011)	PCP + Plotagem recorrente	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck)	80%
(Ni, et al., 2012)	PCP + Separação de informações harmônicas percussivas +Identificação de beats + filtragem de sinal + HMM + Viterbi + Informações de contexto de gênero musical	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	72% no conjunto desconhecido de canções e 83% no conjunto conhecido de canções
(Khadkevich, et al., 2012)	PCP + Espectrogramas por ré-associação de tempo e frequência + filtragem de frequências + HMM + Viterbi	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	70% no conjunto desconhecido de canções e 82% no conjunto conhecido de canções
(Chen, et al., 2012)	PCP + HMM com identificação de acorde e duração + Viterbi	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	66% no conjunto desconhecido de canções e 78% no conjunto conhecido de canções
(Haas, et al., 2012)	NNLS Chroma + Identificação de beats + Tonalidades globais + campo harmônico + regras de harmonia tonal	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	62% no conjunto desconhecido de canções e 72% no conjunto conhecido de canções
(Humphrey, et al., 2012)	Espectros de frequências + Rede Neural	2 tipos de acordes	12 tônicas x 2 tipos de	475 cançãocanções (181 dos Beatles,	77%

	Convolucional		acordes = 24 + 1 (ausência de acorde) = 25	100 do RWC Pop dataset e 194 do US Pop dataset)	
(Khadkevich, et al., 2013	Vetor Chroma com re-associação de tempo e frequência + HMM +Viterbi	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	75,80% em média
(CHO; BELLO, 2013)	PCP por subbandas do áudio + HMM + Viterbi	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	71,43% na média
(Steenbergen, et al., 2013)	HMM + Viterbi + Redes Neurais	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	6,65%
(KHADKEVICH; OMOLOGO, 2014)	HMM + Viterbi + Técnica de reassociação de tempo e frequência	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	66,13%
(ROLLAND, 2014)	HMM + Viterbi + Detecção de beats e detecção de segmentos	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas	50,32%

				para testes	
(CANNAM *et al.*, 2014)	HMM + Viterbi + NNLS *chroma*	18 tipos de acordes	12 tônicas x 18 tipos de acordes = 216 + 1 (ausência de acorde) = 217	206 cançãocanções (Beatles, Queen e Zweieck) + Ashley Burgoyne's Billboard corpus com 200 canções para treinamento e 200 canções desconhecidas para testes	62,48%

Tabela AII.1 – Trabalhos publicados e relacionados com a área de Transcrição de Acordes até o período atual

ANEXO II

A seguir detalhamos o padrão sintático que define o modelo geral de representação de cada acorde, segundo o trabalho de (HARTE *et al.*, 2005):

Tônica do Acorde : (grau 1, grau 2, ...) / Nota do Baixo

Segundo este modelo, a tônica de cada acorde (identificada pela sua nota fundamental, como por exemplo, no acorde de Dó Maior (C) a tônica seria o Dó e no acorde de Ré Menor (Dm) a tônica seria Ré) deve ser seguida de um sinal de dois pontos, a fim de separá-la do tipo de acorde (menor, maior, aumentado, diminuto, etc.) ao qual a mesma está associada. Após o sinal de dois pontos, devem ser listados os graus das notas musicais que compõem o acorde, excluindo a sua tônica, separados por vírgulas e entre parênteses. Caso o baixo do acorde não seja a tônica, após uma barra deverá ser informada qual a nota que o representa. O acorde de Dó maior, segundo este modelo, é representado pela seguinte expressão:

C : (3,5) (Sendo o grau 3 = Mi e o grau 5 = Sol)

No exemplo acima, como o acorde de Dó Maior é representado também pela letra C (D=Ré, E=Mi, F=Fá, G=Sol, A=Lá e B=Si) e como o mesmo é composto pelas notas musicais Dó (sua tônica), Mi e Sol, e além disso, como o intervalo entre Do e Mi, segundo a teoria musical, é de grau 3, e entre Dó e Sol é de grau 5, o acorde de Dó Maior foi representado no formato **C:(3,5)**. Como não é nosso intuito, nem existe esta necessidade, não detalharemos maiores informações a cerca da teoria musical envolvida na formação dos acordes. O ponto mais importante deste trabalho foi a definição de um padrão muito completo de representação simbólica de acordes. Neste sentido, este trabalho foi de fundamental importância para toda a comunidade de MIR, em especial àqueles que trabalham com acordes em geral.

Seguindo este padrão de representação de acordes, os modificadores Sustenido e Bemol são representados pelos símbolos “**#**” e “b” e devem ser usados do lado direito da tônica do acorde, e antes do sinal de dois pontos. O acorde Dó# maior é representado segundo o seguinte padrão:

C# : (3,5) (Sendo o grau 3 = Mi# e o grau 5 = Sol#)

Para resolver uma possível ambiguidade entre a nota “B” (si) e o símbolo “b” (bemol) a análise dos acordes deve diferenciar maiúsculas de minúsculas.

Bb : (3,5) corresponde ao acorde de Si bemol maior

Modificadores de graus do acorde devem ser postos do lado esquerdo dos mesmos. Para exemplificar este caso, observemos a representação do acorde de Ré menor:

D : (b3,5) (Sendo o grau b3 = Fá e o grau 5 = Lá)

Um exemplo mais complexo como o acorde D# menor com sétima, baixo na quinta e nona adicionada teria a seguinte representação:

D# : (b3,5,b7,9) / 5

A Tabela A1.1 traz uma listagem de todos os acordes previstos no trabalho de Harte, inclusive apresentando uma notação opcional simbólica para os mesmos.

Tipo do Acorde	Notação Simbólica	Notação Detalhada
Maior	maj	(3,5)
Menor	min	(b3,5)
Diminuto	dim	(b3,b5)
Aumentado	aug	(3,#5)
Maior com Sétima	maj7	(3,5,7)
Menor com Sétima	min7	(b3,5,b7)
Sétima	7	(3,5,b7)
Diminuto com Sétima	dim7	(b3,b5,bb7)
Meio Diminuto	hdim7	(b3,b5,b7)
Menor com Sétima Maior	minmaj7	(b3,5,7)
Maior com Sexta	maj6	(3,5,6)
Menor com Sexta	min6	(b3,5,6)
Nona	9	(3,5,b7,9)
Maior com Nona	maj9	(3,5,7,9)
Menor com Nona	min9	(b3,5,b7,9)
Quarta Suspensa	sus4	(4,5)

Tabela A1.1 - Tabelas de Tipos de Acordes e suas Notações

Segundo esta tabela, o acorde de Dó menor com sétima poderá ser representado de duas formas:

C:min7 e C : (b3,5,b7)

Seguindo o mesmo raciocínio, o acorde de Lá maior com baixo em Dó# poderá ser representado das seguintes formas:

A/3 , A:maj/3 e A:(3,5)/3

ANEXO III

O formato do arquivo utilizado como padrão de saída de transcrições de acordes para sistemas submetidos ao MIREX (HARTE *et al.*, 2005) foi definido para conter, por cada uma de suas linhas, um intervalo de tempo representado por um instante inicial e um instante final, seguido do tipo de acorde que ocorre durante este mesmo intervalo dentro da canção (Figura AIII.1).

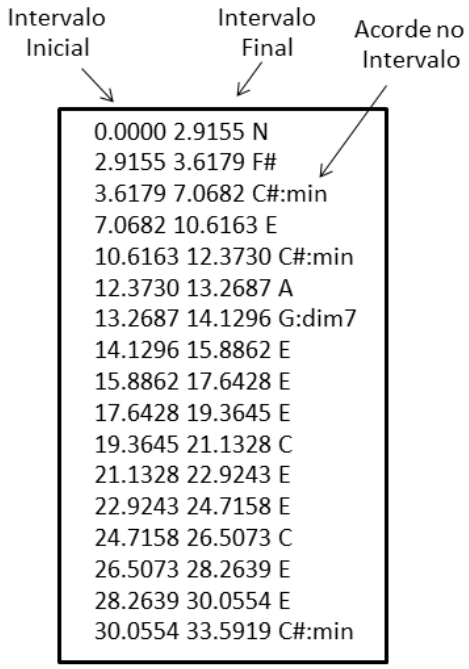


Figura AIII.1 - Exemplo de parte de arquivo formatado contendo saídas de sistemas de transcrições de acordes

ANEXO IV

Neste apêndice o nosso objetivo é a descrição do algoritmo o *Sequence Tracker* desenvolvido em trabalhos anteriores nossos (CUNHA, 1999) e adaptado para o contexto atual.

A ideia foi a de obedecer o mesmo algoritmo já definido, com a retirada apenas das verificações relativas a melodias de cada canção em teste. No algoritmo, existe uma variável central chamada **PossivelRep** que irá conter a lista de todas as sequências que existem na canção em execução e que vão sendo adicionadas a ela na medida em que vão sendo encontradas e também na mesma ordem em que vão sendo encontradas.

O algoritmo se inicia com esta variável vazia e até que sejam completados os três primeiros compassos nenhum teste é realizado (Regra 3 – Linhas 58 e 59). Com o três primeiros compassos executados, a cada novo acorde tocado um teste é realizado para verificar a sua presença numa sequência já existente na variável **PossivelRep** (Linha 61). Quando um acorde coincide com algum que inicie uma sequência presente em **PossivelRep**, os três próximos compassos de acordes tocados são verificados para se poder concluir a existência de uma sequência repetida (Regra 3 – Linhas 62 a 67). Caso haja qualquer diferença de acordes (Regra 7) entre qualquer dos três compassos presentes em **PossivelRep** e os compassos tocados, todo o teste é cancelado e todo o processo volta ao estado inicial exatamente a partir do ponto em que a diferença foi encontrada (Linhas 69 a 77). Caso os três compassos tocados sejam exatamente iguais aos presentes em **PossivelRep**, os acordes que se seguirão ao processo serão definidos pelo Rastreador que os gerará até que a sequência encontrada em **PossivelRep** termine (Linhas 29 a 51). Durante a execução da sequência encontrada, a fim de se justificar a Regra 5, são realizados testes para que quando a sequência de acordes atinja o oitavo e nono compassos e os seus múltiplos (Linha 33), o Rastreador não tenta usar os acordes que fazem parte da sequência em uso no momento. Quando estes compassos se passam, se a sequência ainda não tiver sido finalizada, é feita uma comparação para testar se os acordes que foram gerados nestes compassos em que o Rastreador não interferiu na previsão, a fim de se saber se eles

correspondem, de fato, aos acordes da sequência que o Rastreador vinha gerando (Linha 37). Caso sejam os mesmos, o processo continua até que o fim da sequência seja alcançado. Caso não sejam os mesmos, a sequência é dada como encerrada e o processo se reinicia na busca de novas sequências.

A variável **PossivelRep** está sempre armazenando não só as sequências encontradas, como também as sequências repetidas, sempre indicando as posições de início e fim de cada uma delas. Desta forma, quando a canção começa a se repetir, existe uma estrutura mais ou menos definida de como é a sequência de acordes de toda a canção. Por exemplo, se os oito primeiros compassos de uma canção formam um bloco de acordes que estão presentes em **PossivelRep,** como uma possível sequência, e se a partir do vigésimo quinto compasso, este bloco se reinicia, ele também seria adicionado em **PossivelRep**, só que com a propriedade de um bloco repetido e posicionado dentro do vetor **PossivelRep** exatamente na mesma posição em que ele ocorre na canção. A ideia é a de conseguir montar a estrutura de toda a canção de forma a identificar a sua repetição (Regra 1). A variável **PossivelRep** funciona, então, como um repositório de todas sequências da canção juntamente com todas as suas repetições em seus devidos locais.

O primeiro teste realizado pelo algoritmo diz respeito à repetição da canção (Linha 11). Após a análise de várias canções do corpus, concluímos que após a repetição de, aproximadamente, 25 compassos exatamente iguais em acordes, é muito pouco provável que a canção não esteja se repetindo. Quando o rastreador alcança esta situação, todos os testes são cancelados e a única tarefa dele passa a ser gerar os acordes presentes em **PossivelRep** na sequência em que foram armazenados neste vetor.

```
1.   Programa RastreadorSeq
2.   CompassoAtual = 0
3.   CompassoAtualMusica = 0
4.   NumCompassosIguais = 0
5.   NumCompMusica = 0
6.   bSeqEncontrada = Falso
7.   bPrimeiro = Verdade
8.   bNovaSeq = Verdade
9.   NumSeq = 1
10.  Enquanto Leia( Acorde ) faça
11.      Se PossivelRep.CompareMusica( Acorde, AcordePos ) então
12.          IndAcorde = PossivelRep.CompareMusica( Acorde, AcordePos )
13.          Se CompAtualMusica <> PossivelRep.Acorde[ IndAcorde ].Compasso então
14.              Incremente(NumCompMusica)
15.              CompAtualMusica = PossivelRep.Acorde[ IndAcorde ].Compasso
```

```
16.          Senão
17.             NumCompMusica = 1
18.          Fim se
19.          Se NumCompMusica > 24 então
20.             bMusicaEmRepetição = Verdade
21.          Senão
22.             bMusicaEmRepetição = Falso
23.          Fim se
24.       Senão
25.          NumCompMusica = 0
26.          bMusicaEmRepetição = Verdade
27.       Fim se
28.       Se não bMusicaEmRepetição então
29.          Se bSeqEncontrada e não SeqEncontrada.Fim então
30.             AcordeImp = SeqEncontrada.Acorde
31.             PossivelRep.Adicione( AcordeImp )
32.             bNovaSeq = Verdade
33.             Se Mlt8(AcordeImp.Compasso) ou Mlt8(AcordeImp.Compasso-1) então
34.                SeqAcordesMultiplo8.Adicione( AcordeImp )
35.                Imprima AcordeNull
36.             Senão
37.                Se SeqAcordesMultiplo8 não é null então
38.                   Se SeqEncontrada.CompareComp(SeqAcordesMultiplo8, 2) então
39.                      Imprima AcordeImp
40.                      SeqAcordesMultiplo8 = null
41.                   Senão
42.                      bSeqEncontrada = False; Imprima AcordeNull
43.                   Fim  se
44.                Senão
46.                   Imprima AcordeImp
50.                Fim se
51.                SeqEncontrada.Incremente
52.          Fim se
53.          Se não bSeqEncontrada or SeqEncontrada.Fim então
54.             Se bNovaSeq então
55.                PossivelRep.IncrementeNumSeq
56.                bNovaSeq = Falso
57.             Fim se
58.             Se Acorde.Compasso <= 3 então
59.                PossivelRep.Adicione( Acorde )
60.             Senão
61.                Se PossivelRep.Compare( Acorde ) então
62.                   Se Acorde.Compasso <> CompassoAtual então
63.                      NumCompassosIguais = NumCompassosIguais + 1
64.                      CompassoAtual = Acorde.Compasso
65.                   Fim se
66.                   SeqAnteriores.Adicione( Acorde )
67.                   NumSeqEnc = PossivelRep.NumSeqAtual
68.                Senão
69.                   NumCompassosIguais = 0
70.                   CompassoAtual = 0
71.                   bPrimeiro = Verdade
72.                   Se SeqAnteriores <> null então
73.                      PossivelRep.AdSeq( SeqAnteriores )
74.                      SeqAnteriores = null
75.                   Fim Se
76.                   PossivelRep.Adicione( Acorde )
77.                   Imprima AcordeNull
78.                Fim Se
79.                Se NumCompassosIguais > 3 então
```

```
80.                     bSeqEncontrada = Verdade
81.                     PossivelRep.AdSeq( SeqAnteriores )
82.                     PossivelRep.IncrementeNumSeq
83.                     PossivelRep[ NumSeqAtual ].Repetição = Verdade
84.                     SeqEncontrada = PossivelRep[ NumSeqEnc ]
85.                     SeqEncontrada.IncrementeTrêsCompassos
86.                 Senão
87.                     bSeqEncontrada = Falso
88.                     SeqEncontrada = null
89.                 Fim se
90.             Fim se
91.         Fim se
92.         Senão
93.             Se bDelete então
94.                 PossivelRep.DeleteUltimos16Comp
95.                 bDelete = Falso
96.             Fim se
97.             Imprima PossivelRep.Acorde
98.             PossivelRep.Incremente
99.         Fim se
100.    Fim faça
```