

Análise de Soluções para Busca por Similaridade (*Matching*) de Dados Musicais

Gisele Bernardes da Silva¹

¹Departamento de Informática e Estatística – Universidade Federal de Santa Catarina (UFSC)
Caixa Postal 476 – 88.045-108 – Florianópolis – SC – Brazil

`giselebernardes@grad.ufsc.br`

Resumo. *Este artigo visa apresentar e comparar soluções para recuperação de informação musical. A intenção é analisar soluções que não necessariamente buscam dados musicais apenas através do casamento direto de parâmetros de entrada para a busca, como título da música, palavras-chave ou um áudio como parte da música, mas também através do casamento aproximado (ou similar) destes parâmetros.*

Abstract. *This article aims to present and compare solutions for music information retrieval. The intention is to analyze solutions that do not necessarily search for musical data only through direct marriage of input parameters to the search, such as song title, keywords or an audio with part of the song, but also through approximate (or similar) these parameters.*

1. Introdução

O som é a variação da pressão do ar [Muller 2007]. Representar o som numericamente, é representar uma onda sonora (áudio analógico) em código binário (áudio digital) [Zuben 2004]. Portanto, um arquivo de áudio digital é composto por metadados e som digitalizado, sendo assim, um dado musical [Pacheco and Lopes 2014].

O dado é a representação física de um evento no tempo e espaço que não agrega fundamento, não podendo ser possível entender o que ele representa ou para que ele existe, porém, ao incluir um “significado” no dado e gerar sentido para quem o ouve e ficando claro ou não a que se refere, é gerada a informação [Semidão 2013].

Uma informação musical apresenta determinadas especificidades de comportamento na sua produção, objetivação e uso, pois a manifestação da música apresenta-se carregada de características próprias. Portanto, a compreensão completa da música está diretamente ligada com o reconhecimento do contexto histórico e social de sua origem, com a interpretação pessoal e individual do ouvinte, e com os aspectos sonoros que a constituem. Dessa forma, a música tem diferentes significações para cada indivíduo [Michels 1992].

Até o surgimento dos inventos tecnológicos, a música era um meio de comunicação exclusivamente presencial. Com o decorrer do tempo, as técnicas e invenções aplicadas ao processo de gravação do som foram surgindo e se aperfeiçoando, resultando em aparelhos reprodutores e suportes cada vez mais versáteis e manipuláveis. A música se tornou um objeto de consumo universal e extremamente acessível [Daquino 2012, Gomes 2015].

Com a Internet, a música ultrapassa os limites físicos da mídia, mergulhando no universo digital. Desta forma, a organização da informação, que inclui a sua representação, tem a principal finalidade de possibilitar a recuperação dessa informação, além da sua guarda para a posteridade. A busca por similaridade musical está inserida dentro de um tema de estudos denominado *Music Information Retrieval*. Os pesquisadores de MIR observam que a motivação maior para essa área de pesquisa é o grande volume de música digital disponível na Internet que, quanto mais cresce, menos possibilita sua recuperação eficiente, visto que estão disponíveis em grande volume, mas sem o tratamento adequado [McLane 1996, Gomes 2015, Santini and de Souza 2007].

Este trabalho busca reunir informações com o propósito de contribuir com futuros trabalhos, que desejam desenvolver soluções para a busca por similaridade de dados musicais. A intenção é analisar soluções que não necessariamente buscam dados musicais apenas através do casamento direto de parâmetros de entrada para a busca, como título da música, palavras-chave ou um áudio como parte da música, mas também através do casamento aproximado (ou similar) destes parâmetros.

2. Métodos e Algoritmos para Recuperação da Informação Musical

A análise da informação musical para sua representação apresenta complexidades, pois exige diferentes técnicas de extração de informações para distintas formas de apresentação [Downie 2003].

Um dado musical geralmente é representado por um conjunto de características extraídas do conteúdo da música. Isto porque dados musicais são basicamente arranjos multidimensionais de valores derivados de vários sensores, que é uma representação limitada para definir a semântica do dado. Neste sentido, dados complexos raramente são comparados diretamente. Em vez disso, o conteúdo do dado (ou uma faceta do conteúdo) é analisado por meio de algoritmos especializados de análise, extraindo um conjunto de características que descrevem numericamente o dado [dos Santos Kaster 2012]. Desta forma, as aplicações que lidam com dados complexos, como dados musicais, requerem a realização de consultas por similaridade, ou seja, consultas que realizam busca por objetos da base que sejam similares a um objeto de consulta, de acordo com uma certa medida de similaridade [Barioni 2006].

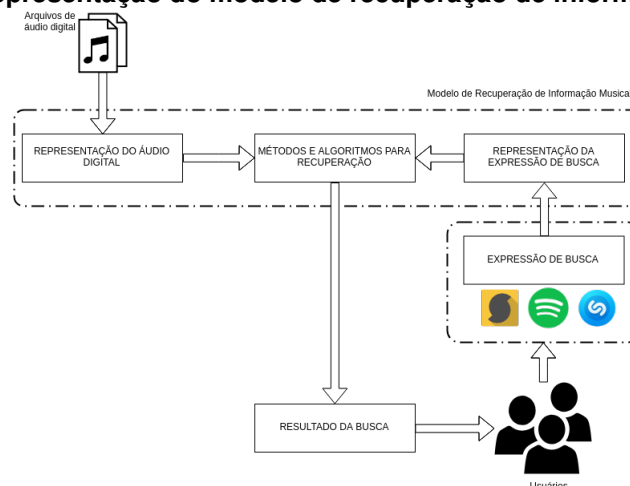
A Sociedade Internacional de Recuperação da Informação Musical (ISMIR¹, na sigla em inglês) abriga, durante sua conferência anual, o MIREX², com o objetivo de estabelecer métodos para a avaliação e comparação das aplicações atuais de recuperação musical. Nesta espécie de competição, pesquisadores inscrevem algoritmos que realizam diferentes tarefas da área de recuperação da informação musical, como classificação automática de gênero musical, identificação automática da pulsação, extração de melodia a partir de arquivos áudio, entre outros.

Um sistema de recuperação de informação musical atua, portanto, como um ambiente mediador da comunicação entre as necessidades informacionais dos usuários e o conjunto de arquivos de áudio. Na Figura 1 é possível compreender melhor o processo de recuperação de informação. Dadas as características de um sinal de áudio, pode-se definir

¹<https://transactions.ismir.net/>

²http://www.music-ir.org/mirex/wiki/MIREX_HOME

Figura 1. Representação do modelo de recuperação de informação musical



Fonte: [Ferreira 2015], adaptado pela autora

métricas de similaridade para comparar músicas. Embora exista uma grande variedade de funções de distância disponível na literatura, não existe um método que determine, de um modo geral, qual deve ser a melhor função de distância a ser utilizada em cada caso. A escolha ou definição de uma função de distância é uma tarefa que depende muito da análise das características específicas do domínio dos dados a serem manipulados [Barioni 2006].

O foco deste trabalho não é aprofundar o conhecimento nos métodos e algoritmos utilizados para recuperação da informação musical. Assim sendo, os principais métodos e algoritmos disponíveis na literatura são descritos aqui de forma breve.

2.1. Audio Fingerprint

Audio Fingerprint é como uma assinatura única de uma música, contendo um sumário de suas características que resume uma gravação de áudio. [Cano et al. 2005] que definem que uma *fingerprint* de áudio é um resumo digital, que pode ser utilizado para identificar uma amostra ou localizar rapidamente itens semelhantes em uma base de dados de áudio, independente do nível de compressão, distorção ou interferência no canal de transmissão. A questão fundamental é como fazer a comparação eficiente do áudio desconhecido contra os possíveis milhões de *fingerprints* [Cano et al. 2005].

2.2. Classificação

Classificação é a tarefa de aprender uma função alvo f que mapeie cada conjunto de atributos x para um dos rótulos de classes y pré-determinados [Tan et al. 2009]. A função alvo também é conhecida como modelo de classificação e, segundo [Tan et al. 2009], pode ser classificada de duas formas: Modelagem Descritiva e Modelagem Preditiva; E os métodos de classificação podem ser divididos de duas formas: classificadores *eager* (espertos) e classificadores *lazy* (preguiçosos). Técnicas de classificação são mais apropriadas para prever ou descrever conjuntos de dados com categorias nominais ou binárias, sendo menos efetivas para categorias ordinais.

2.3. Clustering

A formação de agrupamentos (ou *clustering*) divide os dados em grupos (*clusters*) que tenham significado, sejam úteis, ou ambas as coisas. Se *clusters* com significados forem o objetivo, então os *clusters* devem capturar a estrutura natural dos dados. O objetivo é que os objetos dentro de um *cluster* sejam semelhantes (ou relacionados) entre si e diferentes de (ou não relacionados aos) objetos de outros *clusters*. Quanto maior a semelhança (ou homogeneidade) dentro de um *cluster* e maior a diferença entre *clusters*, melhor ou mais distinto será o agrupamento [Tan et al. 2009].

2.4. Dynamic Time Warping

O *Dynamic Time Warping* (DTW) [Keogh and Ratanamahatana 2004] é uma técnica que permite definir uma métrica entre séries temporais, como por exemplo, características de áudio. Seu uso é inviável em grandes conjuntos de dados. Por este motivo, diversas técnicas de indexação específicas para esse algoritmo foram propostas para a tarefa de busca por similaridade, tais como *lower bound* e *early abandoning*. Com essas técnicas, é possível encontrar os vizinhos mais próximos de uma dada subsequência em uma quantidade massiva de séries temporais. Informações mais detalhadas podem ser consultadas em [Mizutani 2006], [Kruskal and Liberman 1983] e [Juang and Rabiner 1991].

2.5. Indexação

Estruturas de indexação (uma descrição sobre essas estruturas pode ser encontrada em [Garcia-Molina et al. 2002]) são normalmente fornecidas pelos SGBDs. A idéia básica dessas estruturas consiste na escolha de um objeto arbitrário central e na aplicação de uma função de distância para dividir os demais objetos em vários subconjuntos. Dessa maneira, uma estrutura de indexação é construída executando-se esse mesmo procedimento, recursivamente, para cada subconjunto não vazio [Barioni 2006]. Assim como a *Slim-tree*, na literatura podem ser encontradas outros métodos aplicados à indexação de dados musicais, são a *VP-tree* (*Vantage Point tree*) [Yianilos 1993], a *MVP-tree* (*Multi-Vantage Point tree*) [Bozkaya and Özsoyoglu 1997], a *GNAT* (*Geometric Near-neighbor Access Tree*) [Brin 1995], a *M-tree* [Ciaccia et al. 1997], *Família-Omni* e a *DBM-tree* [Filho et al. 2001, Vieira et al. 2004], a *n-grams* [Downie 1999] e o *Vantage Indexing Method* [Typke et al. 2003].

2.6. Medidas de Similaridade

A semelhança entre dois objetos é uma medida numérica do grau no qual os dois objetos se parecem. A diferença, ou dissimilaridade, entre dois objetos é uma medida numérica do grau no qual os dois objetos são diferentes. Frequentemente, o termo distância é usado como sinônimo de diferença, embora, muitas vezes seja usado para se referir a uma classe especial de diferenças. Similaridade e distância são importantes pois são utilizadas por inúmeras técnicas de mineração de dados, como *clustering* (ver subseção 2.3) e classificação (ver subseção 2.2). Quanto menor o valor desta distância, mais semelhantes serão os objetos e eles tenderão a ficar no mesmo cluster. Quanto maior a distância, menos similares serão os objetos e, em consequência, eles deverão estar em grupos distintos.

2.7. Query by Humming

A tarefa de busca musical a partir de um trecho de música cantada ou cantarolada pelo usuário passou a ser conhecida na literatura como *query by humming*, ou QBH. Trata-se de um sistema capaz de reconhecer música pelo *casamento aproximado* de cadeias de caracteres. [Ghias et al. 1995], autores do projeto, aplicaram o conceito de *contorno melódico*, ou seja, a forma natural como nós percebemos a música. Segundo [de Carvalho Araújo Santos 2011], existem dois principais interesses em QBH: Primeiramente, permitir ao usuário identificar uma música da qual não conheça (ou lembre de) qualquer metadado associado (autor, álbum, título da música, entre outros). Em segundo, é dispor de uma maneira mais natural para consultar coleções de músicas digitais armazenadas principalmente em dispositivos portáteis.

2.8. Recuperação por Conteúdo

A recuperação por conteúdo aplica-se a vários tipos de dados, inclusive música, e com o aumento da disponibilização de coleções de áudio digitais, se tornou importante permitir o gerenciamento automático desses tipos de dados por meio da utilização de metadados (como título, álbum e gênero da música). Assim, várias técnicas de extração de metadados foram desenvolvidas, sendo algumas delas especialmente adequadas à recuperação de música por conteúdo. As técnicas baseadas em busca por conteúdo utilizam metadados extraídos automaticamente dos dados musicais para representar e indexar as informações embutidas nos áudios digitais. O conjunto de metadados extraídos é chamado de *vetor de características* [Tzanetakis et al. 2002]. E as operações de comparação entre dados musicais utilizam os vetores de características para medir a similaridade do conteúdo presente nos dados que eles representam.

2.9. Spectral Modeling Synthesis

Spectral Modeling Synthesis (SMS), ou na sua tradução, Síntese de Modelagem Espectral, é uma abordagem de modelagem acústica para voz e outros sinais musicais. Ele considera os sons como uma combinação de conteúdo harmônico e conteúdo de ruído. Os componentes harmônicos são identificados com base em picos no espectro de frequência do sinal, normalmente encontrados pela transformada de Fourier de curta duração (FFT). O sinal que permanece após a remoção dos componentes espectrais, por vezes referido como residual, é modelado como ruído branco passado através de um filtro que varia no tempo. A saída do modelo, então, são as frequências e níveis dos componentes harmônicos detectados e os coeficientes do filtro que variam no tempo. Essas técnicas podem ser usadas para aplicações de síntese, processamento e codificação, enquanto alguns dos resultados intermediários também podem ser aplicados a outros problemas relacionados à música, como separação de fontes sonoras, acústica musical, percepção musical ou análise de desempenho. Informações mais detalhadas sobre a técnica pode ser encontradas em [Serra and Smith 1990].

2.10. Visualização

Visualização de dados musicais é a exibição na forma de um gráfico ou tabela. Uma visualização bem sucedida requer que os dados sejam convertidos em um formato visual de modo que as características dos mesmos e dos relacionamentos entre itens de dados possam ser analisados ou reportados. O objetivo da visualização é a interpretação

da informação visualizada por uma pessoa e a formação de um modelo mental das informações [Tan et al. 2009].

3. Soluções Existentes

Esta seção apresenta de forma resumida as principais soluções comerciais e da academia, para busca de dados musicais.

3.1. Soluções Comerciais

3.1.1. MusicID

Gracenote Inc., fundada em 1998, é uma empresa que fornece metadados de música, vídeo, esportes e tecnologias de reconhecimento automático de conteúdo para empresas e serviços de entretenimento em todo o mundo. A solução da empresa destinada à busca de dados musicais se chama *Gracenote MusicID®* disponível para *smartphones*.

O *Gracenote MusicID®*, faz o reconhecimento de músicas que são tocadas ao seu redor, combinado ao uso de *fingerprints* (ver subseção 2.1) e correspondência de texto para identificar arquivos de música digital em um banco de dados mundial de informações musicais. Uma vez reconhecidos, os arquivos são organizados por nome de faixa, nome do álbum e caminhos de pastas e, então, apresentados ao usuário. Ele é um aplicativo para *smartphones* que identifica músicas ouvindo você cantar, cantarolar ou de músicas que são tocadas ao seu redor.

3.1.2. Shazam

Shazam Entertainment Ltd. foi fundada em 2000 com a idéia de prover um serviço que pudesse conectar as pessoas à música, permitindo a identificação da música através de *smartphones*. A aplicação usa o microfone do *smartphone* ou do computador para capturar uma pequena amostra de música e, então, realiza a identificação da música em um grande banco de dados com mais de 12 bilhões de músicas, com uma alta taxa de acertos.

Para o trecho de música capturado pela aplicação é criado uma *fingerprint* (ver subseção 2.1), que é comparada com todas as outras *fingerprints* derivadas das músicas no banco de dados. Se houver uma correspondência, são enviadas informações da música para o usuário, como artista, álbum e título da música.

3.1.3. SoundHound

SoundHound Inc., fundada em 2005, é uma empresa pioneira em desenvolvimento de aplicações para reconhecimento de voz, compreensão da linguagem natural, reconhecimento de som e tecnologias de busca.

A plataforma independente de Inteligência Artificial *Houndify*, combinada ao *Automatic Speech Recognition* (ASR) e o *Natural Language Understanding* (NLU), permite ao *SoundHound* a identificação de músicas de forma rápida e eficiente. Seus dois produtos conhecidos no meio musical são:

1. *SoundHound Music Search & Play*³: aplicativo para *smartphones* onde é possível descobrir, pesquisar e reproduzir qualquer música com controle de voz. Ele é um aplicativo para *smartphones* que identifica músicas ouvindo você cantar, cantarolar ou de músicas que são tocadas ao seu redor.
2. *Midomi*⁴: aplicação com as mesmas características do item anterior, porém possui versão para *web*. Sua versão *mobile* é destinada a modelos mais antigos de *smartphones*.

3.1.4. Deezer

A Deezer funciona através da correspondência de texto para identificar arquivos de música digital, além do recurso de identificação de músicas que são tocadas ao seu redor combinado ao uso de *fingerprints*, o *SongCatcher*, desenvolvido pela *ACRCloud*.

Recentemente, no final do ano de 2017, além da correspondência de texto para identificar arquivos de música digital, a Deezer lançou o seu próprio recurso de identificação de músicas que são tocadas ao seu redor combinado ao uso de *fingerprints*, o *SongCatcher*, desenvolvido pela *ACRCloud* (ver subseção 3.1.8).

3.1.5. Spotify

Spotify Ltd., fundada em 2006, é um serviço de *streaming* de música, *podcast* e vídeo, além de ser o mais usado no mundo. A plataforma fornece conteúdo protegido provido de restrição pela gestão de direitos digitais de gravadoras e empresas de mídia. O Spotify é um serviço *freemium*: ele possui recursos gratuitos com propagandas ou limitações, e recursos adicionais, como qualidade de transmissão aprimorada e *downloads* de música, que são oferecidos para assinaturas pagas.

A plataforma emprega um modelo de distribuição de dados híbrido com uma combinação de compartilhamento de dados peer-to-peer (P2P) e uma infraestrutura de servidor. A busca da música é feita pela correspondência de texto para identificar arquivos de música digital através da técnica de recuperação por conteúdo (ver subseção 2.8).

3.1.6. SoundCloud

SoundCloud, criada em 2007, é uma plataforma on-line de publicação de áudio utilizada por profissionais de música. Nela os músicos podem colaborar, compartilhar, promover e distribuir suas composições. Originalmente, seu objetivo era permitir que profissionais da música trocassem ideias sobre as composições nas quais estão trabalhando, permitindo uma fácil colaboração e comunicação antes de um lançamento público. Hoje, o site também é utilizado por ouvintes e usuários da *web* em geral.

O SoundCloud descreve as faixas de música graficamente como formas de onda e permite aos usuários comentar partes específicas do áudio (conhecido como comentários

³<https://soundhound.com/soundhound>

⁴<https://www.midomi.com/>

cronometrados). Estes comentários são exibidos ao escutar a parte do áudio que estão se referindo. Outras características incluem respostas, listas de reprodução, seguidores e *downloads* digitais de cortesia. A busca da música é feita pela correspondência de texto para identificar arquivos de música digital através da técnica de recuperação por conteúdo (ver subseção 2.8). Não foi encontrada documentação informando como funciona detalhadamente a recuperação da música.

3.1.7. Musixmatch

A *Musixmatch* foi criada em 2010 com o objetivo de mudar a forma como as pessoas experimentam música e letras. A plataforma pode ser acessada através do site ou via aplicativo para *smartphones*. O Musixmatch digitaliza todas as músicas da biblioteca de música do usuário e encontra letras para todas elas, identificando a letra da música e mantendo sincronizada enquanto a música é tocada. Além da correspondência de texto para identificar arquivos de música digital através da técnica de recuperação por conteúdo (ver subseção 2.8), ela possui também a capacidade para capturar uma pequena amostra de música através de *fingerprints* (mesma função encontrada em soluções como o *Shazam*), desenvolvido pela ACRCLOUD (ver subseção 3.1.8).

3.1.8. ACRCLOUD

A *ACRCLOUD* foi criada em 2015, sendo a solução vitoriosa no campeonato de *Audio Fingerprinting* do MIREX2015, organizado pelo Laboratório Internacional de Avaliação de Sistemas de Recuperação de Informação Musical (IMIRSEL, sigla em inglês).

ACRCLOUD é uma plataforma de microserviços na nuvem que possui reconhecimento de música através de *fingerprints*, onde identifica músicas ouvindo você cantar ou de músicas que são tocadas ao seu redor, além do monitoramento de transmissão com identificação e apresentação de conteúdo, entre outros. Ele possui integração com serviços de música como o Spotify, Deezer, entre outros, que permite desenvolvedores acessarem diretamente esses serviços e oferecer links diretos para seus usuários.

3.1.9. Musipedia

Musipedia é uma enciclopédia aberta de música, criação inspirada no Wikipedia⁵, para localização, edição e expansão de coleções de tons, melodias e temas musicais. A enciclopédia utiliza o mecanismo de pesquisa de melodias, do qual chamam de *melodyhound*, onde é possível encontrar e identificar uma música, mesmo que a melodia seja tudo o que você saiba no momento. A busca também pode ser feita através da pesquisa de contorno melódico (Código de Pearson) ou com base no ritmo. Ainda, os conteúdos podem ser alterados por qualquer usuário, podendo conter um pedaço de música, um arquivo MIDI, informações textuais sobre o trabalho e o compositor.

A recuperação da informação, segundo o site da enciclopédia⁶, pode ser feita da

⁵<https://www.wikipedia.org/>

⁶<https://www.musipedia.org/about.html>

seguinte forma, tradução nossa: tocá-lo em um teclado de piano, assobiar para o computador, simplesmente tocar o ritmo no teclado do computador ou usar o código Pearsons [Development 2018].

3.2. Soluções Acadêmicas

3.2.1. AMUSE

AMUSE (*Advanced Music Explorer*)⁷ é um *framework* desenvolvido pela TU Dortmund, na Alemanha, licenciado sob a GPL e implementado em JAVA.

Segundo [Vatolkin et al. 2010], o *framework* fornece diferentes funcionalidades, como:

- Processamento de som, convertendo arquivos de áudio MP3 em ondas sonoras;
- *Downsampling* e estéreo para a conversão de arquivos de áudio mono;
- Divisão automática de arquivos wave;
- Escalabilidade usando multi-threading em uma máquina ou fornecendo as tarefas para sistemas de grade como Sun Grid Engine ou LSF Batch;
- Gerenciamento eficiente do conjunto de dados que suporta diretamente o formato WEKA ARFF;
- Componente logger integrado.

O projeto é oferecido gratuitamente à comunidade de pesquisa. Informações mais detalhadas sobre o projeto podem ser consultadas em [Vatolkin et al. 2010] e [Dortmund 2018].

3.2.2. CLAM

CLAM (*C++ Library for Audio and Music*)⁸ é um *framework* desenvolvido em C++ no *Music Technology Group* (MTG) da Universidade Pompeu Fabra em Barcelona, Espanha. Ele oferece uma plataforma completa de desenvolvimento e pesquisa para o domínio de áudio e música baseado na técnica SMS (ver subseção 2.9).

Segundo [Amatriain 2004], as características mais importantes do *framework* são:

- Verdadeiramente orientado a objetos. Extensas técnicas de engenharia de software foram aplicadas para projetar uma estrutura que seja altamente (re)utilizável e compreensível;
- É abrangente, uma vez que não só inclui classes para processamento, mas também para entrada e saída de áudio e MIDI, serviços de serialização XML, algoritmos e visualização e interação de dados, e manipulação multi-threading;
- Lida com uma ampla variedade de tipos de dados extensíveis que vão desde sinais de baixo nível (como áudio ou espectro) até estruturas semânticas de nível superior (como frase musical ou segmento);
- É multiplataforma. Todo o código é ANSI C ++ e é regularmente compilado no Linux, Windows e Mac OSX usando os compiladores mais usados. Até mesmo o código para entrada/saída, visualização e multithreading é de plataforma cruzada até a camada mais baixa possível;

⁷<https://sourceforge.net/projects/amuse-framework/>

⁸<http://clam-project.org/>

- O projeto está licenciado sob os termos e condições GPL (Licença Pública GNU). Apesar de possuir a opção de licenciamento duplo da estrutura (ou seja, oferecer uma licença comercial alternativa), tudo o que é oferecido na versão pública é GPL e o projeto é, portanto, Software Livre, código aberto e colaborativo;
- Base para todos os desenvolvimentos futuros no MTG e, portanto, mantido e atualizado regularmente;
- O framework pode ser usado como uma biblioteca C++ regular ou como uma ferramenta de prototipagem. No primeiro modo, o usuário pode estender, adaptar ou otimizar a funcionalidade da estrutura para implementar um aplicativo específico. No segundo modo, o usuário pode facilmente construir um protótipo para testar um novo algoritmo ou aplicativo de processamento de sinais.

Informações mais detalhadas sobre este *framework* podem ser consultadas em [Amatriain 2007, Amatriain 2004].

3.2.3. Java MIR

jMIR (Java MIR)⁹ é um *software* que possui um conjunto de componentes desenvolvido na CIRMMT e Marianopolis College, ambos localizados no Canadá. Cada um dos componentes pode ser utilizado separadamente ou como um todo.

O software é de código livre implementado em Java para uso nas pesquisas de Recuperação de Informação Musical (MIR) e se baseia em uma técnica de mineração de dados, a classificação (ver subseção 2.2).

O objetivo principal do *software* é auxiliar nas pesquisas em classificação automática de música e a análise de similaridade, proporcionando as seguintes características:

- Tornar tecnologias sofisticadas de reconhecimento de padrões acessíveis a pesquisadores de música com históricos técnicos e não técnicos;
- Eliminar duplicação redundante de esforço;
- Aumentar a cooperação e a comunicação entre os grupos de pesquisa;
 - Facilitar o desenvolvimento iterativo e o compartilhamento de novas tecnologias MIR;
 - Facilitar comparações objetivas de algoritmos.
- Facilitar a pesquisa combinando características musicais de alto nível, baixo nível e culturais (ou seja, características simbólicas, áudio e web-minadas).

Informações mais detalhadas sobre o projeto estão disponíveis nas publicações acadêmicas¹⁰. Manuais e documentação para cada componente também podem ser consultados em [McKay 2018] e [McKay 2010].

3.2.4. MIRtoolbox

*MIRtoolbox*¹¹ é um pacote de ferramentas escritas em Matlab para a extração de recursos musicais, como tonalidade e ritmo, tanto para especialistas quanto para não especialistas

⁹<http://jmir.sourceforge.net/>

¹⁰<http://jmir.sourceforge.net/publications.html>

¹¹<https://www.jyu.fi/hytk/fi/laitokset/mutku/en/research/materials/mirtoolbox>

do Matlab.

Segundo [Lartillot 2013], foi elaborado um manual onde são descritas as seguintes especificações da solução:

- Quadro modular: É baseado em um conjunto de blocos de construção que podem ser parametrizados, reutilizados, reordenados, etc.;
- Sintaxe simples e adaptativa: Os usuários podem se concentrar no design geral e o MIRtoolbox que cuida das tarefas laboriosas subjacentes;
- Software livre e código-fonte aberto: A ideia é propor a capitalização da expertise da comunidade de pesquisa e oferecê-la de volta à comunidade;
- Recursos: O MIRtoolbox inclui cerca de 50 extratores de recursos de áudio e música e descritores estatísticos.

O MIRtoolbox é baseado em técnicas de mineração de dados e, por ser um pacote de ferramentas, são variadas as técnicas para recuperação da informação musical, como a classificação (ver subseção 2.2). Mais informações sobre o projeto podem ser consultadas em [Lartillot and Toivainen 2007, Lartillot 2013] e [of Jyväskylä 2018]

3.2.5. MusicMiner

O *Databionic MusicMiner*, desenvolvido como parte de um projeto de pesquisa do Grupo de Pesquisa em Databionics da Universidade de Marburg, na Alemanha, é um navegador para dados musicais baseado em técnicas de mineração de dados, como clusterização (ver subseção 2.3) e visualização (ver subseção 2.10) com base no paradigma dos mapas geográficos ESOM.

O site do projeto¹² apresenta as seguintes características da solução:

- Análise automática de uma árvore de pastas com arquivos de música (MP3, OGG, WMA, M4A, MP2, WAV);
- Descrição automática de arquivos de áudio digital por som;
- Criação de *MusicMaps* para navegar pelo espaço sonoro com base no paradigma dos mapas geográficos ESOM;
- Criação visual de *playlists*;
- Pesquisa por similaridade na coleção de músicas com base no som;
- Navegação hierárquica personalizável da base de dados, como por exemplo, por gênero/artista/álbum ou ano/artista;
- Base de dados flexível, incluindo o armazenamento separado de vários artistas por música, álbuns e listas de reprodução como parte de uma lista de reprodução;
- Importação e exportação de meta informações baseadas em XML.

O *MusicMiner* é escrito em Java para máxima portabilidade e publicado sob os termos da GPL (General Public License). Seu foco principal é a pesquisa e o ensino. Informações mais detalhadas sobre o projeto podem ser encontradas em [Morchén et al. 2005] e [of Marburg 2005].

¹²<http://musicminer.sourceforge.net/>

3.2.6. Tunebot

*Tunebot*¹³ é um projeto criado em 2015, desenvolvido e mantido por *Interactive Audio Lab*¹⁴ na Universidade de Northwestern, nos Estados Unidos. Segundo [Huq et al. 2010b], o Tunebot está disponível como um serviço web e está atualmente em teste beta como um aplicativo do iPhone.

O *Tunebot* compara as músicas com as músicas cantadas pelos usuários utilizando o algoritmo *query by humming* (ver subseção 2.7).

Outro objetivo do projeto é ajudar pesquisadores na área de reconhecimento de músicas que utilizam o algoritmo *query by humming*, facilitando uma pesquisa mais precisa do desempenho do mundo real do que seria possível com conjuntos de dados existentes.

Informações mais detalhadas sobre o projeto podem ser consultadas em [Huq et al. 2010b, Cartwright and Pardo 2012] e [Huq et al. 2010a].

4. Análise Comparativa

Esta seção apresenta os critérios definidos para a análise das soluções existentes, a análise comparativa propriamente dita e os resultados obtidos com a análise. A metodologia de comparação entre as as soluções leva em conta documentações públicas disponíveis que possuam informação pertinente.

4.1. Critérios de Análise

Com base nos conceitos apresentados por [Wazlawick 2012, pg.305], que define os atributos de qualidade internos, externos e de uso de produtos de software; e pela autora [rer. nat. Christiane Gresse von Wangenheim 2017], que apresenta heurísticas definidas por [Nielsen 1995] para assegurar que os produtos são usáveis, esta seção tem por objetivo apresentar os critérios que devem ser considerados na avaliação das soluções para busca de dados musicais, de modo a compará-las e facilitar a escolha pela mais apropriada para uma determinada situação.

Neste trabalho foi realizada uma adaptação de heurísticas para permitir uma boa análise da eficiência e adequação funcional das soluções apresentadas, sendo utilizado informações disponibilizadas nas documentações próprias de cada solução comercial e acadêmica. Já para a análise da usabilidade foi realizada uma adaptação do *MATcH Checklist* disponibilizado pelo Grupo de Qualidade de Software (GQS) da UFSC, onde o conjunto de perguntas possui uma escala de resposta com 2 opções: Sim (a solução atende o objetivo) e Não (a solução não atende o objetivo).

Assim sendo, os seguintes critérios e subcritérios são considerados:

- Eficiência de desempenho: Trata da otimização do uso de recursos de tempo e espaço. Espera-se que o sistema seja o mais eficiente possível de acordo com o tipo de problema que ele soluciona:

¹³<http://music.cs.northwestern.edu/data/tunebot/>

¹⁴<http://music.eecs.northwestern.edu/>

- *Comportamento em relação ao tempo*: Mede o tempo que o sistema leva para processar suas funções, ou seja, o tempo de reconhecimento/busca de uma música;
- *Utilização de recursos*: Avalia a complexidade das estratégias e algoritmos utilizados na recuperação de informação musical;
- *Bitrate*: Mede a qualidade do áudio. Essa qualidade consiste no número médio de bits que será comprimido em um segundo de dados. A unidade utilizada é o KBPS ou 1000 BITS por segundo;
- **Adequação Funcional**: Mede o grau no qual o produto oferece funções que satisfazem necessidades estabelecidas e implicadas quando o produto é usado sob condições especificadas:
 - *Disponibilidade*: Avalia a disponibilidade da aplicação em diferentes plataformas;
 - *Modelo de desenvolvimento*: Avalia se a solução é de código aberto, dando a possibilidade para que qualquer um consulte, examine ou modifique o produto;
 - *Integrações*: Avalia se a solução permite extensões e/ou integrações com outras aplicações;
 - *Acessibilidade*: Avalia se a solução possui acesso ao acervo de músicas on-line e/ou off-line;
 - *Busca de dados*: Avalia se a solução foi projetada para *matching* exato ou por similaridade (aproximado);
 - *Inclusão da dados*: Avalia se a solução permite o envio de músicas feito pelo usuário;
 - *Modelo de pagamento*: Avalia o custo da solução, como por exemplo: Gratuito, Pago ou Freemium.
- **Usabilidade**: Avalia o grau no qual o produto tem atributos que permitem que seja entendido e que seja atraente ao usuário, quando usado sob condições especificadas:
 - *Visibilidade do status do sistema*: O sistema deve sempre manter o usuário informado sobre o que está acontecendo. Por exemplo, os componentes interativos selecionados são claramente distintos dos demais?
 - *Prevenção de erros*: Mensagens de erros devem ser claras e objetivas, devem indicar o problema com precisão e sugerir uma solução;
 - *Flexibilidade e eficiência de uso*: A solução deve permitir configuração de ações frequentes. Por exemplo, as funções mais utilizadas são facilmente acessadas?
 - *Estética e Design minimalista*: Mensagens de diálogos não devem conter informações irrelevantes. Por exemplo, o menu é esteticamente simples e claro, com opções fáceis de encontrar, dispostas em uma ordem lógica e com títulos curtos?
 - *Pouca interação homem/dispositivo*: Qualquer informação deve ser fácil de pesquisar e deve ser focada na tarefa do usuário. Por exemplo, a navegação da solução é intuitiva, é fácil chegar à tela desejada?

4.2. Análise

Uma comparação é primeiramente realizada entre as soluções comerciais, seguida das soluções acadêmicas.

4.2.1. Soluções Comerciais

Inicialmente, pode-se verificar que as soluções comerciais, em sua grande maioria, possuem um modelo de desenvolvimento fechado (F), ou seja, seu código não pode ser alterado, mas disponibilizam APIs Web para a comunidade de desenvolvedores, para que a solução possa ser incorporada a seus próprios sites e aplicações.

Das 9 soluções comerciais analisadas, 4 possuem reconhecimento de músicas de forma aproximada (A), ou seja, é possível realizar buscas de músicas por similaridade através da voz, sendo que 2 delas permite cantarolar. Já as 5 demais soluções realizam o reconhecimento de músicas de forma exata (E), sendo necessária uma parte de áudio "original" e/ou uma busca por conteúdo através de metadados.

Apenas 3 soluções permitem a inclusão de músicas criadas por usuários, enquanto as outras 6 permitem a inclusão de novas músicas somente através de contatos com gravadoras e/ou artistas. Destas 3 primeiras, duas reconhecem músicas de forma aproximada: ACRCLOUD e Musipedia. A primeira é um serviço na nuvem, sendo, até o momento, o maior banco de dados de músicas e um serviço utilizado pela maioria das outras soluções aqui analisadas, como Musixmatch, que permite o reconhecimento de músicas por similaridade, e o Deezer, que possui o recurso de identificação de músicas que são tocadas ao seu redor. Ambas as soluções utilizam o método *Fingerprint*.

Musipedia é uma wikipedia de músicas que aceita contribuições musicais de diversas formas: através da voz, partes de músicas, ou até em formato MIDI (quando o som é criado digitalmente). Não foi encontrada documentação que especificasse o método ou algoritmo utilizado para o reconhecimento das músicas.

Com isso, é possível verificar que tanto soluções para o reconhecimento de músicas de forma aproximada quanto soluções para o reconhecimento de forma exata podem utilizar o mesmo método de *Fingerprint*. Apesar de utilizarem o mesmo método, a forma como pode ter sido desenvolvido é que gerou a vantagem competitiva dentre os concorrentes do mesmo ramo.

O SoundHound utiliza a tecnologia de Inteligência Artificial (IA) para o reconhecimento de músicas. O desenvolvimento de uma solução baseada em IA se dá através de métodos e algoritmos de aprendizagem de máquina, como Classificação e Clustering, porém não foi encontrada documentação que especificasse o método ou algoritmo utilizado pelo SoundHound para o reconhecimento das músicas. Com essa solução, é possível recuperar músicas através da voz, cantarolando ou até mesmo informando uma parte do áudio "original". Conforme os usuários cantam e conforme a escolha da música do resultado amostrado, a IA associa a cantoria àquela música. Então, quanto mais cantar e buscar as músicas "certas" para a cantoria, a taxa de acerto aumenta, formando uma rede totalmente interligada para o reconhecimento correto de músicas.

Há também as soluções que utilizam o método de Recuperação por Conteúdo (RPC), que seriam as cadeias de caracteres, em buscas, por exemplo, por título, álbum ou gênero da música - o que chamamos de metadados -, conforme já explicitado em seções anteriores deste trabalho. Destas soluções, pode-se citar o Spotify e o SoundCloud. A busca das músicas no Spotify, por exemplo, é feita exclusivamente através de texto, não sendo possível adicionar músicas criadas pelo usuário. Entretanto, é possível criar

playlists com as suas músicas preferidas.

Em relação ao tempo, as diferenças foram constatadas verificando o tipo de método utilizado para o reconhecimento/busca de músicas. Os testes foram realizados usando conexão de Internet Wi-Fi e Dados Móveis (4G). Para as soluções utilizando RPC, foi testada a busca de músicas pela correspondência de texto através de metadados (ou seja, através da informação do título, álbum ou gênero da música). Neste caso, a busca demorou até 30s para retornar uma amostra de resultados. Para as soluções que utilizam *Fingerprint* e fazem reconhecimento de música de forma exata (E), foi testada a busca através da informação de uma parte de áudio "original". Já para as soluções de reconhecimento de músicas de forma aproximada (A), foi testada a busca através da informação de uma parte de áudio "original", do canto ou do cantarolar. Ambas as soluções podem variar de 5s até 13s para retornar uma amostra de resultados.

Cabe observar aqui que uma análise feita apenas sobre o comportamento geral das soluções em relação ao tempo não seria relevante, já que tempos entre 5s e 30s são relativos à situação em que o usuário está no momento. Por exemplo, o usuário pode estar em lugares onde a conexão de Internet não é boa ou quase nula, ou pode estar em lugares onde a conexão de Internet é ótima, já que o reconhecimento de uma música depende de uma conexão de Internet. Quanto melhor a conexão, mais rápido será o reconhecimento da música.

Com exceção do ACRCLOUD, todas as demais soluções possuem versões gratuitas para uso, com a possibilidade de pagar uma mensalidade e não ter interrupções e propagandas entre as músicas. Ainda, poucas soluções possuem integrações com outros serviços e/ou aplicações. Com o Shazam, por exemplo, é possível integrar-se ao Spotify e, então, ao encontrar uma música, poder ouvi-la por completo. Da mesma forma para o SoundHound, além de ser possível o compartilhamento da sua pesquisa com o Twitter. Com o Musixmatch integrado ao Spotify, por exemplo, ao encontrar uma música, você acompanha a música com a letra em tempo real. O ACRCLOUD e o Musixmatch são as soluções que possuem o maior número de integrações.

Quanto à acessibilidade, todas precisam de conexão com a Internet para o uso de suas funcionalidades e apenas 2 das 9 soluções analisadas possuem acesso off-line, que se dá através do download de músicas para a memória interna do dispositivo. As 2 soluções são Spotify e Deezer.

Com relação à usabilidade, testes foram realizados considerando os critérios e subcritérios descritos na seção 4.1. Foi realizada uma adaptação do *MATCH Checklist* disponibilizado pelo Grupo de Qualidade de Software (GQS) da UFSC, onde o conjunto de perguntas possui uma escala de resposta com 2 opções: Sim (a solução atende o objetivo) e Não (a solução não atende o objetivo). Cada solução possui particularidades que podem atrair ou afastar o usuário. Os resultados dos testes são discriminados abaixo, feitos em um smartphone, modelo Asus Zenfone 4, com sistema operacional Android.

Todas as soluções são simples, possuem navegação intuitiva e são de fácil execução. As funções mais utilizadas são facilmente acessadas. Os componentes interativos são claramente distintos uns dos outros, com ícones intuitivos. Com exceção do Musipedia, as soluções possuem uma linguagem clara, concisa e funcionam corretamente, não apresentando problemas. Com relação à Musipedia, ela é uma enciclopédia de

músicas e sua navegação não é muito intuitiva, além de possuir textos longos, não apresentando uma linguagem clara e concisa. Não funciona corretamente, não apresentando mensagens de erros claras e objetivas.

No caso do SoundHound, Deezer e Spotify, o menu é esteticamente simples, mas suas versões gratuitas possuem propagandas, o que polui a tela. Ao se adquirir a versão paga, as propagandas são retiradas. Já o MusicID e o Musixmatch não possuem muitas funcionalidades e o menu é esteticamente simples, claro e sem propagandas.

Deezer, Spotify e SoundCloud executam buscas de músicas pela correspondência de texto (metadados) para identificar arquivos de música digital, ou seja, através da informação do título, álbum ou gênero da música.

O MusicID é de uso exclusivo para o reconhecimento de músicas ouvindo você cantar, cantarolar ou que são tocadas ao seu redor. O mesmo vale para o SoundHound, com o diferencial de possuir mais funcionalidades além do reconhecimento de músicas, como a possibilidade de criar playlists e o compartilhamento com o Twitter. Além das informações básicas sobre a música, o aplicativo sugere alguns vídeos que podem ser assistidos diretamente no SoundHound.

O Shazam é similar ao MusicID e ao SoundHound. A diferença é que faz o reconhecimento de músicas apenas tocadas ao seu redor. Ele também possui a funcionalidade de tirar fotos de QR Codes para realizar a busca da música. O Deezer é similar ao Shazam, pois também possui a funcionalidade de reconhecimento de músicas que são tocadas ao seu redor e a possibilidade do download das músicas para uso off-line, além da criação de playlists.

O Spotify é similar ao Deezer em termos de funcionalidades. A diferença é que o Spotify não possui reconhecimento das músicas pelo som (trecho de áudio ou voz). Para usufruir de todas as possibilidades do aplicativo, como o download de músicas para uso off-line, é necessário obter a versão premium. Por sua vez, o SoundCloud é similar ao Spotify, com a diferença de ser o único aplicativo comercial analisado que permite a inclusão de músicas criadas pelos usuários.

O Musixmatch é um aplicativo que sincroniza letras de músicas e também permite o reconhecimento de músicas que são tocadas ao seu redor. Ainda, o ACRCLOUD é um serviço na nuvem, com um grande banco de dados musical. Além de oferecer o serviço de reconhecimento de músicas, ele também permite o reconhecimento de músicas pela web, como um teste do seu serviço.

Quanto ao critério de Bitrate, das 9 soluções analisadas, foi encontrado essa informação em apenas 4. Diferentes bitrates podem produzir qualidade de som variável. Assim, quanto maior o bitrate, mais vezes por segundo o som original é amostrado, produzindo assim uma reprodução mais autêntica e melhor do som [SanDisk 2018].

O Deezer e Spotify apresentam bitrate de até 320kbps, possuindo 3MB por minuto de áudio e uma qualidade aproximada à de um CD. Já o MusicID e o SoundCloud apresentam bitrate de até 128kbps, possuindo 1MB por minuto de áudio e uma qualidade mínima aceitável em termos de alta fidelidade do áudio [F 2012].

Por fim, das 9 soluções analisadas, 7 são multiplataformas, sendo em sua maioria voltadas para uso em smartphones e 2 de uso exclusivo via Web.

4.2.2. Soluções Acadêmicas

As soluções acadêmicas apresentam um modelo de desenvolvimento aberto, ou seja, seu código pode ser alterado por outros usuários, contribuindo para a resolução de problemas existentes nas aplicações. Das 6 soluções analisadas, todas possuem reconhecimento de músicas de forma aproximada (A), ou seja, permitem realizar buscas de músicas através de trechos de áudio e/ou voz e/ou codificação MIDI e/ou por conteúdo (metadados).

Todas as soluções permitem a inclusão de músicas criadas por usuários. Quanto à disponibilidade, com exceção do Tunebot, todas as soluções estão disponíveis para Linux, Mac OSX e Windows e seu acesso é off-line, ou seja, é necessário o download e instalação do projeto no dispositivo. O download e instalação das soluções é feito de forma gratuita. O Tunebot está disponível na Web e também possui uma versão beta para smartphones iOS, tendo o seu acesso on-line.

Cada solução possui ferramentas variadas para uso distinto no reconhecimento de músicas, o que gera diversas formas de utilização dos recursos. Por exemplo, o Music-Miner possui ferramentas que utilizam os métodos de Classificação (ver subseção 2.2) e Visualização (ver subseção 2.10) através de Mapas Musicais Topográficos.

O AMUSE e o Java MIR utilizam o método de Classificação (ver subseção 2.2). Já o Tunebot utiliza Query by Humming (QBH) (ver subseção 2.7). O CLAM utiliza Spectral Modeling Synthesis (SMS) (ver subseção 2.9), e o MIRtoolbox é um acoplado de ferramentas baseado em técnicas de mineração de dados. Por ser um pacote de ferramentas, são variadas técnicas para recuperação da informação musical que ele oferece, como por exemplo, clusterização e classificação.

O AMUSE e o Tunebot são as duas soluções que possuem integração com outras aplicações. O AMUSE faz parte do rol de ferramentas do MIRtoolbox. Já o Tunebot tem integração com o *Karaoke Callout*¹⁵, que ajuda a construir a base de conhecimento de músicas do Tunebot.

Em relação ao tempo, foi encontrada documentação apenas para o Tunebot. A solução utiliza QBH e, segundo [Lab 2008], o comprimento mediano de uma música cantada por um usuário é de cerca de 18 segundos de áudio, levando cerca de 5s para retornar resultados a partir do momento em que a aplicação recebe a música. Para comparação, a música mais longa recebida até o momento tem cerca de 48 segundos e a aplicação demora cerca de 13s para retornar uma resposta. O tempo de retorno é uma função de vários fatores, incluindo o tamanho do banco de dados e o comprimento da consulta, tanto em termos da duração total do áudio quanto do número de notas que o usuário cantou.

Quanto ao critério de bitrate, das 6 soluções analisadas, não foi encontrada documentação que especificasse o bitrate utilizado para o fluxo de transferência das músicas.

Como explicitado anteriormente, as soluções analisadas tem seu acesso off-line. Diante disso, para o funcionamento correto de cada aplicação é necessário a instalação de projetos de terceiros. Desta forma, com a impossibilidade de funcionamento das

¹⁵*Karaoke Callout* é um jogo de karaokê para a plataforma iOS da Apple que permite aos usuários "tunebot"desafiarem uns aos outros para um duelo de canto [Shamma and Pardo 2006].

aplicações, não foi possível analisar os critérios de usabilidade das soluções acadêmicas.

4.3. Discussão dos Resultados

Conforme explicado na seção anterior, a análise de soluções foi dividida em dois grupos: soluções comerciais e soluções acadêmicas. Uma análise entre todas elas seria inviável, pois cada grupo se difere no motivo de uso e é voltado a públicos-alvo diferentes. As soluções comerciais são mais utilizadas por usuários finais. Já as soluções acadêmicas são voltadas a usuários pesquisadores da comunidade de MIR.

4.3.1. Soluções Comerciais

Com relação à análise das soluções comerciais, pode-se concluir que as soluções focam na usabilidade visando facilitar o uso para o seu público-alvo, que apenas as utilizam no seu dia-a-dia. Estes usuários não possuem muito interesse em como essas soluções funcionam, desejando apenas que a busca seja rápida e satisfaça os filtros informados.

Os testes realizados de reconhecimento e busca de músicas, por metadados ou por som, demonstraram que as aplicações atenderam bem a todas as funcionalidades oferecidas. Se o objetivo for o reconhecimento de músicas quando não se conhece nenhum valor de metadado, o Shazam e o Deezer são essenciais. Porém, se o objetivo for aquela música que não sai da cabeça e precisar cantar ou cantarolar, então o ideal é usar o SoundHound ou o MusicID. Se o conhecimento da letra da música for indispensável, é possível usar o Musixmatch.

Se o objetivo for playlists e download de músicas, para não ter a necessidade do uso de conexão com a Internet, o Spotify ou o Deezer atendem a demanda. Já o Musipedia, em particular, é uma solução voltada a usuários músicos, que compõem músicas através da codificação MIDI.

Em relação ao tempo, o ideal é a conexão de Internet Wi-Fi. Nestas condições, todas as soluções, exceto o Musipedia, fazem o reconhecimento/busca de músicas no tempo mínimo de 5s. Nesse caso, saber o objetivo do usuário com relação a desempenho é imprescindível para a escolha de qual solução usar.

Pode-se concluir que o método mais utilizado para o reconhecimento de músicas é o *Fingerprint*. Inclusive, o uso do serviço do ACRCcloud tem sido incorporado em cada vez mais soluções para o reconhecimento das músicas, como é o caso do Musixmatch e do Deezer. Outro método bastante utilizado é o RPC, principalmente pelas soluções que permitem apenas buscas exatas, onde é feita a correspondência de texto para identificar arquivos de música digital, ou seja, através da informação do título, álbum ou gênero da música.

4.3.2. Soluções Acadêmicas

Com relação à análise realizada para as soluções acadêmicas, pode-se concluir que as soluções focam na otimização e desempenho no que diz respeito ao reconhecimento de músicas. O desenvolvimento de tais aplicações é voltado para os usuários pesquisadores da comunidade de MIR, com o objetivo de trazer inovação na busca de dados musicais.

Todas as soluções incluem uma estratégia para busca por similaridade (a maioria delas utilizando técnicas de Classificação), são gratuitas e permitem a inclusão de dados musicais em seus repositórios. Ainda, a maioria são soluções off-line e estão disponíveis em diversas plataformas. Por outro lado, poucas soluções oferecem integração com outras soluções ou aplicações que tocam músicas. Inclusive apresentam um modelo de desenvolvimento aberto, ou seja, seu código pode ser alterado por outros usuários.

Houve dificuldade no uso das soluções acadêmicas e por este motivo, não houve a análise dos critérios de usabilidade, além de não ter sido possível o teste prático dos critérios de eficiência de desempenho, o que poderia ter enriquecido ainda mais a presente pesquisa. A documentação existente para cada proposta sugere que o desenvolvimento das soluções acadêmicas geralmente tem o objetivo de validar uma determinada pesquisa, sendo após oferecida gratuitamente à comunidade para futuras pesquisas e o aprimoramento da solução. Desta forma, a usabilidade nem sempre é o foco principal.

Por outro lado, certos processos já foram desenvolvidos e estão sendo utilizados como parte de uma nova solução, como é o caso do projeto AMUSE, que faz parte do rol de ferramentas do MIRtoolbox. Outro caso é o Tunebot, que possui uma versão beta desenvolvida para plataformas iOS, sendo um dos seus objetivos ajudar pesquisadores na área de reconhecimento de músicas que utilizam o algoritmo *Query by Humming*, facilitando uma pesquisa mais precisa do desempenho do mundo real do que seria possível com conjuntos de dados existentes.

Por fim, a escolha de um método para a recuperação da informação musical vai depender do objetivo da solução a ser desenvolvida. Neste sentido, quanto mais aprimorados forem os métodos e algoritmos de soluções acadêmicas, melhores soluções comerciais serão desenvolvidas para a utilização pelo usuário final.

5. Considerações Finais

Este trabalho teve como objetivo analisar soluções para a busca por similaridade de dados musicais, ou seja, soluções que não necessariamente buscam dados musicais apenas através do casamento direto de parâmetros de entrada para a busca, como título da música, palavras-chave ou um áudio como parte da música, mas também através do casamento aproximado (ou similar) destes parâmetros. Além de uma pesquisa aplicada sobre o que são dados musicais, sobre a forma como os dados são tratados e armazenados, para que possam posteriormente ser recuperados e ouvidos no dia-a-dia, este trabalho buscou reunir e comparar informações com o propósito de contribuir com futuros trabalhos que desejam desenvolver soluções para a busca por similaridade de dados musicais.

Pela observação das características analisadas, as soluções comerciais enfatizam a usabilidade, visando facilitar o uso para o público-alvo, que é o usuário final no seu dia-a-dia. O objetivo destas soluções pode ser o reconhecimento de músicas através de uma parte da música ou da voz, bem como a criação de playlists e download de músicas para uso off-line. O método mais utilizado para o reconhecimento de músicas é o Fingerprint.

As soluções acadêmicas, por sua vez, enfatizam o desempenho no reconhecimento de músicas. O desenvolvimento de tais soluções é destinado principalmente a usuários pesquisadores da comunidade de MIR com o objetivo de trazer inovação na maneira como são recuperados os dados musicais. Todas as propostas incluem uma estratégia para busca

por similaridade (a maioria delas utilizando técnicas de Classificação), são gratuitas e permitem a inclusão de dados musiciais em seus repositórios. Ainda, a maioria são soluções off-line e estão disponíveis em diversas plataformas. Por outro lado, poucas soluções oferecem integração com outras soluções ou aplicações que tocam músicas.

Como trabalhos futuros, sugere-se, primeiramente, a aquisição dos protótipos das soluções acadêmicas para que uma avaliação de usabilidade destas propostas possa ser realizada. Neste trabalho, devido a restrições de tempo, não foi possível ter acesso a essas soluções. Outra sugestão é um aprofundamento no estudo dos métodos e algoritmos utilizados para a recuperação da informação musical, bem como uma comparação para verificar os métodos mais eficientes. Por fim, o desenvolvimento e avaliação de soluções para recuperação de informação musical poderia contar com a participação de mais mulheres no programa de mentoria WIMIR¹⁶, incluindo soluções para busca por similaridade.

Referências

- Amatriain, X. (2004). *An Object-Oriented Metamodel for Digital Signal Processing with a focus on Audio and Music*. Phd thesis, universitat Pompeu Fabra, Barcelona.
- Amatriain, X. (2007). Clam: A framework for audio and music application development. *IEEE Softw* 24(1), page 82–85.
- Barioni, M. C. N. (2006). *Operações de consulta por similaridade em grandes bases de dados complexos*. Ciências de computação e matemática computacional, Instituto de Ciências Matemáticas e de Computação - ICMC-USP, São Carlos.
- Bozkaya, T. and Özsoyoglu, M. (1997). Distance-based indexing for high-dimensional metric spaces. In *23rd ACM International Conference on Management of Data (SIGMOD)*, page 357–368, Tucson, USA. Anais eletrônicos.
- Brin, S. (1995). Near neighbor search in large metric spaces. In *21st International Conference on Very Large Data Bases (VLDB)*, page 574–584, Zurich, Switzerland. Anais eletrônicos.
- Cano, P., Battle, E., Kalker, T., and Haitsma, J. (2005). A review of audio fingerprinting. *Journal of VLSI Signal Processing*, 41:271–284.
- Cartwright, M. and Pardo, B. (2012). Building a music search database using human computation. *Proceedings of the 9th Sound and Music Computing Conference (SMC 2012)*, Copenhagen, Denmark, July 12-14, 2012.
- Ciaccia, P., Patella, M., and Zezula, P. (1997). M-tree: An efficient access method for similarity search in metric spaces. In *23rd International Conference on Very Large Data Bases (VLDB)*, page 426–435, Athens, Greece. Anais eletrônicos.
- Daquino, F. (2012). A evolução do armazenamento de músicas [infográfico].
- de Carvalho Araújo Santos, E. J. (2011). Ferramentas de busca musical baseada em reconhecimento de melodias: Estudo, análise e implementação. Graduação, Universidade de Pernambuco, Recife - PE.
- Development, M. (2018?). Musipedia the open music encyclopedia.

¹⁶<https://wimir.wordpress.com/mentoring-program/>

- Dortmund, T. (2018?). Amuse - advanced music exploration.
- dos Santos Kaster, D. (2012). *Tratamento de condições especiais para busca por similaridade em bancos de dados complexos*. PhD thesis, Instituto de Ciências Matemáticas e de Computação - ICMC-USP. 177 p.
- Downie, J. S. (1999). *Evaluating a simple approach to music information retrieval: Conceiving melodic n-grams as text*. Pós-graduação, University of Western Ontario, London, Ontário, Canadá.
- Downie, J. S. (2003). Music information retrieval (chapter 7). In *Annual Review of Information Science and Technology*, number 37, pages 295 – 340. Blaise Cronin, Medford, NJ: Information Today.
- F, S. (2012). itunes's new 1080p compares well to blu-ray quality, at a fraction of file size.
- Ferreira, J. B. (2015). Recuperação de informação de música e dados id3: possíveis aplicações. Dissertação (mestrado em ciência da informação), Universidade Estadual Paulista, Faculdade de Filosofia e Ciências, Marília.
- Filho, R. F. S., Traina, A. J. M., Traina-Jr, C., and Faloutsos, C. (2001). Similarity search without tears: The omni family of all-purpose access methods. In *17th IEEE International Conference on Data Engineering (ICDE)*, page 623–630, Heidelberg, Germany.
- Garcia-Molina, H., Ullman, J. D., and Widom, J. (2002). *Database systems: The complete book*. Editora Ciência Moderna, Prentice Hall, Upper Saddle River, New Jersey.
- Ghias, A., Logan, J., Chamberlin, D., and Smith, B. C. (1995). Query by humming: Musical information retrieval in an audio database. In *Proceedings of the Third ACM International Conference on Multimedia, MULTIMEDIA '95*, pages 231–236, New York, NY, USA. ACM.
- Gomes, A. C. F. (2015). Representação da informação musical: uma análise a luz dos requisitos funcionais para dados de autoridade (frad). Graduação, Universidade de Brasília, Brasília.
- Huq, A., Cartwright, M., and Pardo, B. (2010a). Crowdsourcing a real - world on - line query by humming system. *Proceedings of the 7th Sound and Music Computing Conference (SMC 2010), Barcelona, Spain, July 21-24, 2010*.
- Huq, A., Cartwright, M., and Pardo, B. (2010b). Crowdsourcing a real-world on-line query by humming system. In *Proceedings of the SMC 2010 - 7th Sound and Music Computing Conference, Barcelona, Espanha*.
- Juang, B. H. and Rabiner, L. R. (1991). Hidden markov models for speech recognition. *Technometrics*, 33(3):251–272.
- Keogh, E. and Ratanamahatana, C. A. (2004). Exact indexing of dynamic time warping. *Knowledge and Information Systems 7.3*, 7(10.1007/s10115-004-0154-9).
- Kruskal, J. and Liberman, M. (1983). The symmetric time-warping problem: From continuous to discrete. *7th International Workshop, GREC*.
- Lab, I. A. (2008). Tunebot dataset.
- Lartillot, O. (2013). Mirtoolbox user's guide 1.7.1.

- Lartillot, O. and Toivainen, P. (2007). Mir in matlab (ii): A toolbox for musical feature extraction from audio. In *Proc. 8th International Conference on Music Information Retrieval (ISMIR 2007)*, page 127–130, Vienna.
- McKay, C. (2010). *Automatic music classification with jMIR*. Phd thesis, CIRMMT and Marianopolis College, Québec.
- McKay, C. (2018?). jmir.
- McLane, A. (1996). Music as information. In *Annual Review of Information Science and Technology (ARIST)*, volume 31, pages 225 – 262. American Society for Information Science.
- Michels, U. (1992). *Atlas de música*. Madrid: Alianza, 2 edition.
- Mizutani, E. (2006). The dynamic time warping algorithms. *Lecture Note for Mechanical Engineering Seminar, Tokyo Metropolitan University*.
- Morchen, F., Ultsch, A., Nocker, M., and Stamm, C. (2005). Databionic visualization of music collections according to perceptual distance. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, pages 396–403, London, UK.
- Muller, M. (2007). *Information Retrieval for Music and Motion*. ISBN 978-3-540-74047-6. Springer Berlin Heidelberg New York, Bonn, Germany.
- Nielsen, J. (1995). 10 usability heuristics for user interface design.
- of Jyväskylä, U. (2018?). Mirtoolbox.
- of Marburg, U. (2005). Musicminer - databionic musicminer.
- Pacheco, J. and Lopes, R. (2014). Descrição e procura de música. *Comunicação Áudio e Vídeo, Instituto Superior Técnico*.
- rer. nat. Christiane Gresse von Wangenheim, D. (2017). Critérios e recomendações ergonômicas - aula 3.
- SanDisk (2018?). Definição de bitrate.
- Santini, R. M. and de Souza, R. F. (2007). Recuperação da informação de música e a ciência da informação: Tendências e desafios de pesquisa. In *VIII ENANCIB – Encontro Nacional de Pesquisa em Ciência da Informação*, pages 1 – 14, Salvador, Bahia. Organização e Representação do Conhecimento Poster.
- Semidão, R. A. M. (2013). Dados, informação e conhecimento: elementos de análise conceitual. *DataGramaZero*, 14(4).
- Serra, X. and Smith, J. (1990). Spectral modeling synthesis: A sound analysis/synthesis based on a deterministic plus stochastic decomposition. *Computer Music Journal*, 14:12–24. SMS.
- Shamma, D. and Pardo, B. (2006). Karaoke callout: using social and collaborative cell phone networking for new entertainment modalities and data collection. In *In Proceedings of ACM Multimedia Workshop on Audio and Music Computing for Multimedia (AMCMM 2006)*, Santa Barbara, CA, USA.

- Tan, P.-N., Steinbach, M., and Kumar, V. (2009). *Introdução ao Data Mining Mineração de Dados*. Editora Ciência Moderna, Rio de Janeiro.
- Typke, R., Giannopoulos, P., Veltkamp, R. C., Wiering, F., and van Oostrum, R. (2003). Using transportation distances for measuring melodic similarity. In *ISMIR Proceedings*, page 107–114.
- Tzanetakis, G., Ermolinskiy, A., and Cook, P. R. (2002). Pitch histograms in audio and symbolic music information retrieval. *3rd International Symposium on Music Information Retrieval (ISMIR)*, pages 31–38.
- Vatolkin, I., Theimer, W., and Botteck, M. (2010). Amuse (advanced music explorer) – a multitool framework for music data analysis. In *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010)*, page 33–38, Utrecht, Netherlands.
- Vieira, M. R., Chino, C. T.-J. F., and Traina, A. J. M. (2004). Dbm-tree: A metric access method sensitive to local density data. In *19 o Simpósio Brasileiro de Banco de Dados (SBBd)*, page 163–177, Brasília, DF. Anais eletrônicos.
- Wazlawick, R. S. (2012). *Engenharia de Software para Sistemas de Informação: Conceitos e práticas que fazem sentido*. UFSC, Florianópolis.
- Yianilos, P. N. (1993). Data structures and algorithms for nearest neighbor search in general metric spaces. In *4th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, page 311–321, Austin, USA. Anais eletrônicos.
- Zuben, P. (2004). *Música e tecnologia: o som e seus novos instrumentos*, volume 1. Irmãos Vitale, second edition.