Classificação e análise de músicas para medição de popularidade

Antônio N. Rea, Douglas A. C. da Silva, Mateus M. Vaz Pinto, Leandro A. da Silva

Faculdade de Computação e Informática – Universidade Presbiteriana Mackenzie (Mackenzie)
01302-907 - São Paulo – SP – Brasil

{antonio.rea,douglasandrade.silva,mateusmosquera.pinto}@mackenzista.com.br, leandroaugusto.silva@mackenzie.br

Abstract. *The use of streaming in the music area is a reality that is increasingly needed by artists and users. For the artist in the aspect of dissemination of works and even as a source of income and for the user for entertainment purposes. It is known that some songs end up being more successful than others and understanding the reasons for this effect is interesting for artists to plan for future work. In this work, the use of a machine learning algorithm known as decision tree to classify songs in popularity is proposed. It is assumed as a premise for this that information extracted from the lyrics of the songs using natural language processing techniques together with sound metric attributes made available by the Spotify platform can characterize the songs in their popularity. The experiments with 1247 samples allowed to reach a model with predictive accuracy of 65.5%.*

Resumo. *O uso de streaming na área musical é uma realidade que se faz cada vez mais necessária aos artistas e usuários. Do artista no aspecto de difusão dos trabalhos e até mesmo como fonte de renda e do usuário para uso no entretenimento. Sabe-se que algumas músicas acabam fazendo mais sucesso que outras e entender as razões desse efeito é interessante para que os artistas se planejem em trabalhos futuros. Neste trabalho é proposto o uso de um algoritmo de aprendizagem de máquina conhecida como árvore de decisão para classificar músicas em popularidade. Assume-se como premissa para isso que informações extraídas das letras das músicas com uso de técnicas de processamento de linguagem natural em conjunto com atributos de métricas sonoras disponibilizadas pela plataforma Spotify possam caracterizar as músicas em sua popularidade. Os experimentos com 1247 amostras permitiram chegar em um modelo preditivo coma acurácia de 65.5%.*

1. Introdução

Com o aumento acelerado da tecnologia nos últimos anos, diversas mudanças surgiram no modo em que a população consome conteúdo, e dentre essas mudanças, está a popularização das plataformas de *streaming*. A pouco tempo atrás, o meio principal de consumo de músicas era através de fonogramas, como vinis e CDs que eram distribuídos em lojas. A popularização da internet revolucionou o consumo de fonogramas por apresentar alternativas de consumo através do meio digital. A indústria fonográfica se adaptou a este novo meio de comercialização, tornando a mídia digital seu principal meio de geração de receitas. (Neto 2021).

De acordo com a Federação Internacional da Indústria Fonográfica (*IFPI*), foi apontado que os serviços de *streaming* (incluindo assinaturas pagas e gratuitas com anúncios) foram o principal meio de crescimento das receitas globais de músicas gravadas em 2020, sendo responsável por 62.1% do crescimento no ano.

Através de *API's (Application Programming Interface)* disponibilizadas por serviços de *streaming* (principalmente o *Spotify*), é possível observar que uma música é composta por diversas métricas (*features*) sonoras, como 'Popularidade', 'Capacidade de dança', 'Volume', 'Acústica' etc. Através de algoritmos de Aprendizado de Máquina (*Machine Learning),* é possível analisar essas métricas, viabilizando uma coleta de informações detalhadas sobre uma determinada música. (Panda, Gonçalves, Paiva, Redinho e Malheiro 2021)

Outro fator a ser analisado para obter informações relevantes sobre uma música é a sua letra, por ser um fator presente na maioria das músicas populares e por serem fáceis de se obter através de sites *online*. Para analisar as letras das músicas, é possível utilizar ferramentas de Processamento de Linguagem Natural (PLN) ou, em inglês, *Natural Language Processing.* (Cano, Mahedero, Koppenberger, Martínes e Gouyon 2005).

Diversas músicas alcançam altas taxas de popularidade, e com o crescimento da quantidade de músicas e artistas disponíveis nos *streamings*, se torna um desafio entender quais são as métricas que mais contribuem para uma música alcançar sucesso.

1.1. Contextualização e Relevância do Tema

O contexto de letras musicais entra em uma área do universo de Inteligência Artificial chamada de Processamento de Linguagem Natural, essa área de pesquisa está por trás do conceito de que o ser humano se comunica de maneira diferente de uma máquina, enquanto os computadores utilizam linguagens formais e estruturadas com foco em sintaxe rígida para a compilação e processamento de informações (Python, Java etc.), os seres humanos se comunicam através da linguagem natural, que é a forma mais comum de comunicação. Com essas diferenças entre a forma que nós seres humanos nos comunicamos em relação a instrução de uma máquina, realizar o processamento dessa linguagem natural no escopo de Inteligência Artificial torna-se um desafio de propiciar que uma máquina entenda de fato o que uma frase significa. (Barr, 1980).

Essa dificuldade de comunicação entre humano e máquina criam barreiras que possuem um grande potencial para serem exploradas, e o foco dessa pesquisa será explorar o conjunto de dados sonoros e a análise de letras musicais com algoritmos de aprendizado de máquina, para classificação de músicas em português por popularidade.

A utilização dos algoritmos propostos no projeto permite a obtenção e análise das canções, e com a letra sendo uma parte importante da semântica de uma música, os dados obtidos através da análise das letras e métricas sonoras possibilitam uma análise detalhada a respeito de informações contidas dentro de uma música. (Cano, Mahedero, Koppenberger, Martínes e Gouyon 2005)

1.2. Objeto de Pesquisa

Essa pesquisa propõe o estudo de métodos de classificação em relação a popularidade de músicas com a utilização de dados sonoros como fonte a plataforma de streaming *Spotify* e bibliotecas de PLN em letras das músicas coletadas, buscando a maneira mais eficaz de analisar as principais variáveis presentes na música, assim coletando informações relevantes a respeito das tendências que uma música possui para se tornar popular, com o foco de melhorar o entendimento de como os usuários consomem conteúdo musical atualmente.

1.2.1. Contextualização do Problema de Pesquisa

Com a ascensão das plataformas de streaming, a indústria musical encontrou novas maneiras de analisar os dados das músicas. De acordo com Maaso e Hagen (2020), métricas de músicas são essenciais para gerentes na indústria musical e seus respectivos artistas, e o *Spotify* é atualmente o principal *streaming* que providencia essas métricas e ainda regularmente adicionando dados com informações relevantes disponibilizando-as para as partes interessadas.

Dados mais recentes do Spotify, lançados através de um comunicado de imprensa em outubro de 2021, mostram seu último resultado trimestral, obtendo uma alta de 26,6% sobre o mesmo período de 2020, e com uma base de 381 milhões de usuários mensais. Portanto demonstrando que esse nicho do mercado é expressivo e em expansão.

Neste contexto, a pergunta que pretende ser respondida nesta pesquisa é: Um modelo tem potencial de prever a classificação de popularidade de uma música através de seus dados?

1.2.2. Hipótese

Aplicando o processamento de linguagem natural para a análise de letras de músicas e extração de métricas sonoras somados, possibilita a criação de um modelo capaz de classificar com desempenho adequado músicas populares sem interação com dados prévios de usuários, obtendo uma acurácia preditiva do caso.

1.3. Objetivos do Estudo

1.3.1 Objetivo Geral

Como objetivo geral, tem-se a implementação de uma estrutura arquitetural para suportar a coleta de dados na *API* de *streaming* de músicas disponibilizada pelo *Spotify*, utilização de bibliotecas de PLN em letras musicais para geração de métricas quantitativas e a geração de modelos preditivos com foco no desempenho de acurácia e precisão de resultados.

1.3.2 Objetivos Específicos

Como objetivos específicos, tem-se a coleta de dados de *API's* que a plataforma de *streaming* disponibiliza, além da letra das músicas que através de análise dos algoritmos serão extraídas métricas relevantes para a pesquisa, persistindo em banco de dados. Para análise as letras das músicas, será utilizado bibliotecas de PLN como o NLTK (*Natural Language Toolkit*), análise de dados, algoritmos de classificação como: árvore de decisão, matriz de confusão e validação cruzadam, implementadas na biblioteca *scikit-learn*.

Com a utilização dessas ferramentas, é possível analisar a letra de uma música e seus dados de sonoridade (acústica, volume, valência etc.), portanto tornando possível obter um modelo de classificação com teste e validação de predição. As músicas extraídas na *API* do *Spotify* já são classificadas com um grau de popularidade (de 0 até 100) o que possibilita utilizar as técnicas devidamente selecionadas e obter uma acurácia nos grupos de classificação propostos.

1.4 Justificativa

A crescente quantidade de usuários e de músicas presentes nas plataformas de *streaming* torna relevante a utilização de novas métricas para medir a popularidade de uma música. A análise de Maaso e Hagen (2020) concluem que as partes interessadas (*stakeholders)* precisam de um número de dados cada vez maior para tomar decisões sobre o que promover. No entanto, a maioria das partes interessadas se concentram em métricas simples, como picos de visualização perceptíveis "de relance".

Com isso, surge a possibilidade de usar os dados obtidos através de uma análise mais detalhada para entender quais são as métricas das músicas que as pessoas mais gostam e que mais contribuem para o sucesso de uma música no contexto da sociedade atual.

Este projeto foi criado pensando em resolver esse problema com ferramentas modernas utilizando a linguagem natural para a análise das músicas do meio digital, com essas tecnologias é possível elaborar um algoritmo totalmente funcional para suprir a tendência de análises automatizadas.

1.5 Delimitação do Estudo

O projeto foi desenvolvido em torno das músicas brasileiras, independente do gênero musical, a ideia é fazer um algoritmo que contemple todos os gêneros para saber quais parâmetros as pessoas gostam. Delimitando a área de atuação apenas para as músicas na língua portuguesa, outras línguas estarão fora da nossa linha de análise, para ter uma análise mais assertiva usando as ferramentas tecnológicas.

O projeto está delimitado aos tempos atuais pois como proposto é preciso utilizar ferramentas tecnológicas com acesso à internet para o desenvolvimento, projeto possa ser usado também no futuro com os avanços das plataformas de streaming, sendo base para futuros trabalhos que aprimora a análise das músicas.

2. Referencial Teórico

2.1. Processamento de Linguagem Natural

A fundamentação teórica do trabalho se refere à literatura básica do assunto de Processamento de Linguagem Natural. Conforme descrito por SL Pereira (2011), o PLN

é constituído por modelos computacionais que realizam tarefas dependentes da linguagem natural, como a tradução e interpretação de textos. O PLN está voltado para três aspectos da comunicação: som, estrutura e significado. Entender sobre o que se consiste no PLN é fundamental para compreender suas possibilidades de utilização, proporcionando uma base de pesquisa sólida para o desenvolvimento de novos frameworks e para o aperfeiçoamento dos já existentes.

Há várias formas de estudar uma língua para facilitar o seu entendimento, a fonologia é relacionada ao reconhecimento dos sons que contém cada palavra, já a morfologia usa unidades primitivas, isto é, partes das palavras, para reconhecer como um todo. A sintaxe determina a estrutura de uma frase, classificando a frase entre sujeito e predicado e depois mais especificamente como artigo, substantivo e verbo. Outro estudo importante é a semântica que associa os significados das palavras a uma estrutura sintática, usando os sentidos das palavras para transmitir o significado da frase. Por último a pragmática verifica o significado da frase levando em conta o contexto, assim escolhendo o significado mais apropriado para aquele contexto. (SL Pereira 2011)

De acordo com Dale (2010), os métodos de PLN são desenvolvidos através dos seguintes estágios de abordagem linguística: Pré-processamento, Análise léxica, Análise sintática, Análise semântica e Análise pragmática.

Algumas técnicas de pré-processamento de dados junto com os estudos da língua serão utilizadas como base para o projeto, como a tokenização ou análise léxica, técnica que quebra as frases usando a análise sintática em tokens, ou pedaços menores, estes que podem ser palavras, números ou sinais de pontuação. Outra técnica de processamento de dados que será utilizada como base é a remoção das *stopwords*, que são palavras frequentemente usadas em texto de linguagem natural, porém seu valor para o significado da frase é pouco em relação ao significado geral da frase, estas palavras são classificadas como *stopwords* e são removidas. (Jain, Kulkarni e Shah 2018)

Etiquetamento (*tagging*) é outro pré-processamento de texto que também será utilizada no algoritmo, onde é feito um reconhecimento e rotulação dos elementos do texto para a síntese da fala, isso é útil para a extração de termos, eliminar ambiguidades, compor novas frases e saber os significados das palavras (Miasato, Gonçalves, Costa e Silva 2014), outra técnica que será utilizada é o *stemming* que é um procedimento que reduz todas as palavras de uma frase para a mesmo forma em comum, normalmente removendo a derivação e sufixos flexionais. (Lovins 1968)

2.2 Algoritmos de aprendizagem de máquina

2.2.1. Support Vector Machine

O Support Vector Machine ou SVM é um algoritmo que aprende através de exemplos a rotular objetos, tal como aprender a reconhecer um cartão de crédito fraudulento ao examinar centenas de milhares de cartões de créditos fraudulentos e não fraudulentos. (Noble 2006) A abordagem de SVM que será usada nesse projeto é a abordagem de separação de classes, neste conceito usasse um hiperplano que é dividido em duas classes maximizando a margem dos pontos que estão mais pertos das classes, estes pontos que estão situados nos limites são chamados de *support vectors*(vetores de suporte), o meio

da margem entres esses sup*port vectors* é a separação otimizada do hiperplano. (Meyer 2017)
O objetivo de implementar esse algoritmo é ajudar nas classificações das músicas usando os dados coletados das letras e as métricas disponíveis do Spotify, para obter a acurácia de acerto do projeto. Segundo Lorena e Carvalho (2007), diferentes aplicações obtiveram sucesso na utilização do modelo (SVM), incluindo categorização de textos.

2.2.2. Árvore de Decisão

Árvore de decisão é uma das técnicas de mineração de dados geralmente responsável pela tarefa de classificar dados. De acordo com Silva (2016), é um modelo capaz de guiar a tomada de decisão sobre a determinação da classe esperada, ela consiste em uma coleção de nós internos e de nós folhas em hierarquia, que cada nó representa uma regra de "SE-ENTÃO", que representa um conjunto de condições para uma tomada de decisão, que representam os nós folhas da árvore, para cada profundidade da árvore é gerada uma regra de "SE-ENTÃO".

Para a classificação dos dados, é relevante entendo o conceito de impureza, que tem relação a variabilidade de classes presentes através de uma partição dos dados. Quanto mais classes diferentes existirem, quer dizer que mais impura a partição é, sendo apenas pura quando contém apenas uma classe. O algoritmo determinara as partições levando em conta sua impureza, a forma de medir a impureza pode ser pelo índice Gini ou pela medida de entropia, que mede a desordem do sistema, quanto maior o grau de entropia, mais desorganizado será o sistema, assim maior esforço para determinar quais dados pertencem a quais classes. (Silva 2016)

Nos modelos de árvores de decisão um parâmetro importante a ser observado seria a profundidade máxima da árvore, onde são realizados testes com sua profundidade afim evitar um *Overfitting*, que são procedimentos que incluem mais termos do que o necessário ou usam abordagens mais complicadas do que o necessário (Hawkins 2004). A Árvore de decisão é um algoritmo que foi usada neste projeto para classificar os dados, resultando na acurácia de acerto.

3. Método proposto

Os métodos utilizados para atingir os objetivos propostos, foram organizados seguindo uma arquitetura conjunta de diferentes tecnologias, cada componente especializado em um passo específico do projeto total, como é visto na representação da Figura 1. Os passos utilizados: 1 - Fontes de Dados; 2 – População dos Dados; 3 – Modelo de Persistência; 4 – Processamento de Linguagem Natural; 5 – Algoritmos de Classificação.

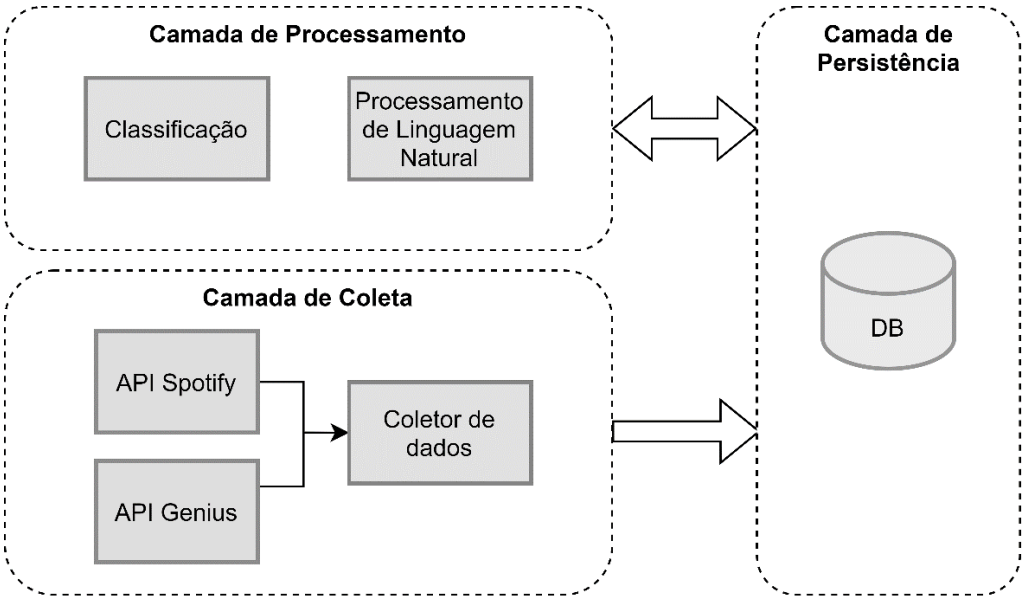

Figura 1. Representação dos componentes em nível arquitetural.

3.1. Fontes de Dados

Com a proposta e contexto deste projeto estabelecidos, o primeiro passo para entendimento das métricas de uma música é a coleta de dados. Assim como nós seres humanos tomamos decisões, é necessário gerar conhecimento através de informações oriundas de dados previamente coletados (Silva 2016), uma das maiores plataformas de *streaming* de músicas o *Spotify* disponibiliza dados de artistas, álbuns e músicas em sua *Web API,* para aplicações cadastradas no *dashboards* de aplicativos, vinculadas a uma conta unificada de usuário.

Para coleta de letras de músicas foi escolhido o portal *Genius*, o qual já possui parceria com o *Spotify* no tocante a letras musicais, além de também disponibilizar uma *Web API* para busca de música, entretanto mesmo realizando a busca do identificador da música a letra não é disponibilizada diretamente, mas sim a localização da página da letra no site oficial do Genius, o que possibilita a criação de um *web crawler* (método de busca e indexação em páginas web) que executará expressões regulares com o objetivo de obter somente a letra. A consulta de ambas as interfaces de aplicação é padronizada pelo formato *REST (Representational State Transfer)* o qual define entre outras regras a comunicação de dados semiestruturados no formato *JSON (JavaScript Object Notation),* o acesso dessas interfaces é realizado por solicitação de autenticação e autorização via credencias de acesso seguindo o padrão de mercado *OAuth 2.0* (*Open Authorization - 2.0*).

3.2. População dos Dados

A população dos dados será realizada pelo consumo de recursos expostos nas interfaces de aplicação (*Spotify* e *Genius*), esse consumo foi realizado via protocolo *HTTP* (*Hyper Text Transfer Protocol*) e seus respectivos verbos (*GET, POST, PUT* etc.) definidos no modelo *REST*. Os dados semiestruturados obtidos nessa comunicação viabiliza a criação de uma coleção de objetos em estruturas de persistência, os sistemas modularizados são interligados e se complementam através dessa unificação com a base de dados.

A busca dos artistas é realizada pelo nome em ambas as interfaces, ação essa que possibilita asserção de consistência de dados entre as plataformas de sonoridade e letras consultadas. Após a obtenção dos dados dos artistas é realizada um lote de chamadas assíncronas para consulta de cada música por artista, as músicas na estrutura de exposição *Spotify* possuem uma análise de sonoridade denominada "*tracks's audio features*" que são o foco de persistência entre os dados sonoros. Neste mesmo lote de cadastro de variáveis sonoras é realizado a execução do *web crawler* dentro do site *Genius* para obtenção das respectivas letras necessárias. Para realização de comunicação genérica de comandos em base de dados foi utilizado o padrão de abstração de comandos *Repository*, uma classe que possibilita o encapsulamento de lógicas com a fontes de dados.

3.2.1. Limpeza de Dados

Em um fluxo de exceção nas delimitações propostas pode ocorrer de artistas brasileiros possuírem a postagem de músicas em outros idiomas ou somente instrumentais, nesse contexto é realizado um processo regressivo de remoção desses tipos de *outliers.*

3.3. Modelo de Persistência

Para realização da persistência de dados foi escolhido o banco de dados *MongoDB*, um tipo de *NoSQL (Not only Structured Query Language)* organizado em documentos. Seus documentos são organizados em uma estrutura *JSON*. O principal motivo da sua escolha foi em sua não necessidade de declaração de campos para persistência como é observado nos modelos *SQL (Structured Query Language)*, graças a possibilidade de adicionar campos dinamicamente foi possível evoluir a capacidade e alcance de novas variáveis sonoras e de letras sem muito retrabalho.

Dentro da coleção de documentos a estrutura criada para artistas e músicas seguiu:

Artistas → Lista [Músicas].

O total de artistas armazenados é de 188 com 1247 músicas cadastradas em um espaço de cerca de 3 *MB.*

Figura 2. Estrutura de dados de um Artista e uma Música.

3.3.1. Glossário "*Audio Features*"

As métricas que foram adicionadas como foco na análise preditiva são os seguintes campos qualitativos:

Tabela 1. Glossário de dados referente aos atributos do *Spotify*.

Nome do Atributo	**Significado do atributo**
Popularidade	Popularidade definida pelos algoritmos internos do *Spotify* (0 - 100).
Capacidade de dança	Descreve o quão adequada uma faixa é para dançar.
Energia	Representa uma medida percentual de intensidade e atividade da música.
Volume	O volume geral de uma faixa em decibéis (dB).
Discurso	Percentual de presença de palavras faladas durante uma faixa.
Acústica	Uma medida de confiança se a música é acústica.

Instrumentalidade	Prevê se uma música não contém vocais, apenas instrumentos.
Vivacidade	Detecta se a presença de público nas gravações da faixa.
Valência	Medida que descreve a positividade musical da faixa.
Duração em ms	A duração da música em milissegundo (ms).
Término do *fade-in*	*Fade-in* é uma transição de som em volume baixo para o volume original, o atributo mede em segundos a duração do *fade-in*.
Começo do *fade-out*	*Fade-out* é o momento em que a música está em seu volume original gradualmente diminui o volume até o silêncio, o atributo mede a duração em segundo do *fade-out*.
Tempo	O tempo estimado geral de uma batida por minuto. Na terminologia musical, o tempo é a velocidade ou ritmo de uma determinada peça.
Confiança do tempo	Percentual de confiança do atributo tempo.
Fórmula de compasso	Uma fórmula de compasso estimada. A fórmula de compasso (medidor) é uma convenção notacional para especificar quantas batidas existem em cada barra.
Confiança da fórmula de compasso	Percentual de confiança do atributo fórmula de compasso.
Chave	A chave em que está a faixa. Os inteiros mapeiam os tons usando a notação padrão de classe de tom.
Confiança da chave	Percentual de confiança do atributo chave.
Modo	Modo indica a modalidade (maior ou menor) de uma faixa, o tipo de escala da qual seu conteúdo melódico é derivado.
Confiança do modo	Percentual de confiança do atributo modo.
Letra	Atributo que armazena a letra da música.

3.4. Processamento de Linguagem Natural

O passo de coleta de dados refletiu as métricas sonoras em uma base unificada, em meio aos dados coletados ainda restam um tipo de métrica que não pode ser considerada qualitativa, a letra musical, este campo obtido de maneira separada da maioria das "*Audio*

Features" é um exemplo de um processo de comunicação humana assim como descrito no referencial deste trabalho, a linguagem natural. O objetivo deste segmento da arquitetura é de utilizar técnicas de PLN para compilação de valores quantitativos provenientes da letra da música.

Para implementação dos padrões e algoritmos já propostos pela comunidade de PLN indicadas no referencial teórico, será utilizado a biblioteca NLTK (*Natural Language Toolkit*) foi desenvolvido com o foco em quatro objetivos: Simplicidade, consistência, extensibilidade e modularidade (Bird, Klein e Loper 2009), essa ferramenta já possui implementação de diversas técnicas com suporte a língua portuguesa como por exemplo: Tokenização, *Stemmer, Tagging* e remoção de *Stopwords*.

Antes da realização da extração de métricas da letra foi executado o processo de limpeza do texto, onde é feito um filtro de palavras que estejam nesse catálogo de *stopwords* resultando em um texto para análise mais limpo para análise.

Com a técnica de Tokenização foi possível segmentar as palavras e obter as seguintes métricas: Total de palavras, quantidade de palavras únicas e frequência de repetição de palavras.

A técnica de *Tagging* utilizada para classificação gramatical da letra foi o padrão de treinamento Trigrama + Bigrama utilizando o *corpus* (módulo da biblioteca NLTK) Mac-Morpho possibilitando a obtenção dos seguintes dados: Quantidade de Verbos, Quantidade de substantivos e Quantidade de Advérbios. Com a inclusão de mais etiquetas foi possível observar no momento do processamento de dados que muitas etiquetas não foram encontradas na maioria das letras, portanto foi decidido nesse momento utilizar somente esses três tipos de etiqueta que seriam as mais frequentes após a remoção das *stopwords* (Verbos, Substantivos e Advérbios).

O método de *Stemmer Snowball* foi utilizado para busca e separação de prefixos e sufixos, escolhido para esse processamento por possuir um suporte a língua portuguesa dentro do NLTK, as métricas obtidas: dados analíticos referente a repetição de radicais.

3.4.1. Dificuldades no Processamento

A plataforma *Genius API* não disponibiliza a letra da música com sentenças separadas como os artistas compõem ou interpretam incialmente, isto é, o texto é obtido como uma linha única sem caracteres de controle (sinalização dentro do texto como *\n que indica a quebra de linha*) para identificação da divisão dentro do texto, inviabilizando o processo de tokenização de sentenças. Sem esse tipo de separação a possibilidade de obter métricas referentes as sentenças separadas da música se tornaram uma barreira difícil de ser superada sem a mudança da fonte de dados de letras. Métricas não viáveis com o tratamento da letra atual: quantidade de sentenças, similaridade entre sentenças e características da repetição de sentenças.

Uma música que possui os caracteres de controle pode ser segregada da seguinte forma:

1ª Sentença – Palavra 1,

2ª Sentença – Palavra 2 Palavra 3

3ª Sentença - Palavra 4 Palavra 5 Palavra 6.

Uma letra de música sem esse tipo de divisão é tratada como uma sentença única por exemplo:

1ª Sentença - Palavra 1, Palavra 2 Palavra 3 Palavra 4 Palavra 5 Palavra 6.

3.5. Algoritmos de Classificação

Algoritmos de Classificação são o último passo do método proposto, consistem em absorver todo o levantamento e complemento das fases anteriores para gerar os modelos preditivos. Os modelos de classificação SVM e Árvore de Decisão são algoritmos de *Machine Learning* do tipo supervisionado amplamente estudados e suas definições e pesquisas relacionadas na comunidade científica possibilitou a adição dessas técnicas em bibliotecas focadas em Inteligência Artificial, no caso destes dois modelos será utilizado a biblioteca *scikit-learn,* seguindo a modularização e padronização desenvolvidas nesta ferramenta é obtido uma implementação comum para os modelos de classificação, que podem ser representada resumidamente como:

Configuração → Treinamento → Teste → Avaliação de Resultados.

Cada estágio de implementação possui uma ou mais técnicas específicas para sua realização, é importante realizar um processo assertivo para obter um modelo adequado para o contexto proposto.

3.5.1. Validação Cruzada e Matriz de Confusão

A separação de treinamento e teste entre as músicas coletadas foram selecionadas com a utilização do método de Validação Cruzada, além de ser também método que facilita a execução de teste dos modelos preditivos, de acordo com Stone (1974) Validação Cruzada (*Cross-Validation*) consiste na divisão controlada ou não controlada de uma amostra de dados em duas subamostras, a escolha de um preditor estatístico, incluindo qualquer estimativa necessária em uma subamostra e em seguida, a avaliação de seu desempenho, medindo suas previsões em relação a outra subamostra.

Os modelos de classificação configurados precisam de alguma validação após treinamento e teste, e esse modelo foi utilizado o método de Matriz de Confusão para identificação da predição pelas classes esperadas. De acordo com Prina e Trentin (2015), a matriz de confusão pode ser definida como a forma de representar a qualidade obtida de uma classificação digital de imagem, sendo expressa através da correlação de informações dos dados de referência (compreendido como verdadeiro) com os dados classificados, utilizando essa técnica é possível verificar a taxa de acertos na classificação das classes propostas pelo modelo.

3.5.2. Configuração Árvore de decisão

Nos modelos disponíveis na biblioteca *Scikit-learn* já existe a parametrização padrão e muitas vezes genéricas para necessidades específicas de quem as utiliza, para o modelo de Árvore de Decisão foram utilizadas as seguintes variáveis visando as necessidades do projeto: critério e profundidade máxima.

A variável "critério" tem como possíveis entradas os valores "*Gini"* e "*Entropy"* que definem o método escolha de qualidade da divisão dos nós, se será por impureza coeficiente Gini ou por entropia. Após execução de teste realizando validação cruzada foi escolhido o critério Gini, o qual possuiu um menor número de desvio padrão.

Tabela 2. Diferença de critérios

Critério	Desvio Padrão
Gini	0.0875
Entropia	0.0938

A variável "profundidade máxima", foi modificada para evitar a geração de modelos com *overfitting* ou até mesmo de *underfitting*, este parâmetro está relacionado com o conceito de poda da árvore de decisão que seria uma das formas de corrigir problemas de ajuste de classificação. Utilizando o método de validação cruzada K-fold, foi obtido uma profundidade máxima de 5, o que possibilitou encontrar um modelo com uma acurácia estável.

3.5.3. Configuração *SVM*

Para o modelo SVM o parâmetro modificado foi o "*kernel*", valor responsável por definir a estratégia e o algoritmo utilizado. Os valores válidos para esse campo são: *linear, rbf, sigmoid, poly e precomputed.* Para a escolhe entre esses valores foi realizado o processo de teste de acurácia com matriz de confusão, os valores que possuírem a maior consistência de teste Verdade – Falso seria escolhido, neste caso foi utilizado o *kernel linear.*

3.5.4. Popularidade, Treinamento e Teste

Os modelos de classificação utilizam métricas de objetos de entrada para posicioná-los em determinadas categorias, no caso deste projeto quem define tais categorias ou classes é a métrica obtida pela *API Spotify* "Popularidade", um indicador de 0 - 100 que combina diversas variáveis dos ouvintes da música para ser contabilizada, essa métrica vai ser utilizada para indicar a predição de treinamento e teste de ambos os classificadores selecionados (Árvore de Decisão e SVM). Para realização da classificação são definidas duas categorias de música 1 – Não Popular e 2 – Popular, sendo a seleção de músicas com esses rótulos definidos pela seguinte regra:

Equação 1. Representação lógica da categorização

Se "popularidade" ≤ "média de dados de popularidade":
Categoria → 1
Senão:
Categoria → 2

4. Resultados Experimentais

O resultado da execução dos testes é o passo final para avaliação de conclusão da hipótese propostas sendo o estágio de observação os resultados dos métodos implementados que viabilizando.

Os modelos *SVM* e Árvore de decisão foram trabalhados no mesmo formato, possuindo entrada X sendo as métricas quantitativas e Y o resultado de popularidade esperado 1 = Não Popular e 2 = Popular. Realizando a execução do teste preditivo foi obtido uma acurácia de 65.5% para o modelo de árvore de decisão como é visto na figura 2 e 59% para o modelo *SVM* como é possível observar na figura 3.

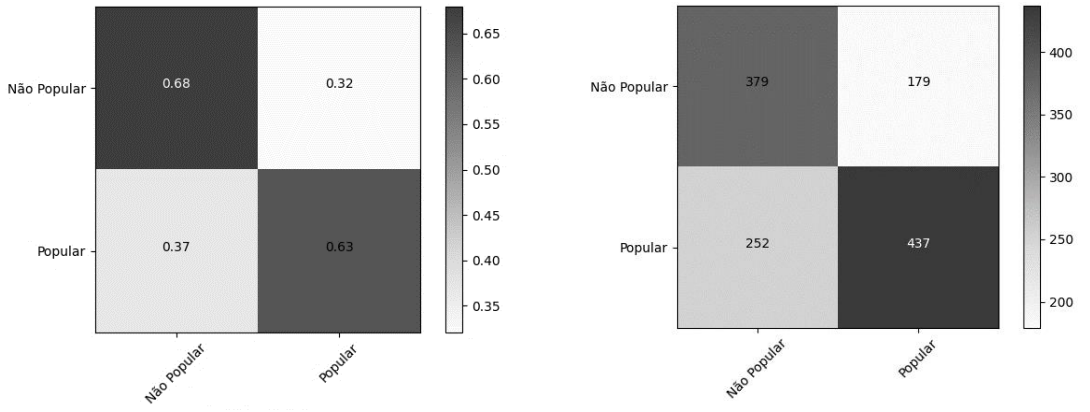

Figura 3. Matriz de Confusão – Árvore de decisão - acurácia e número absoluto

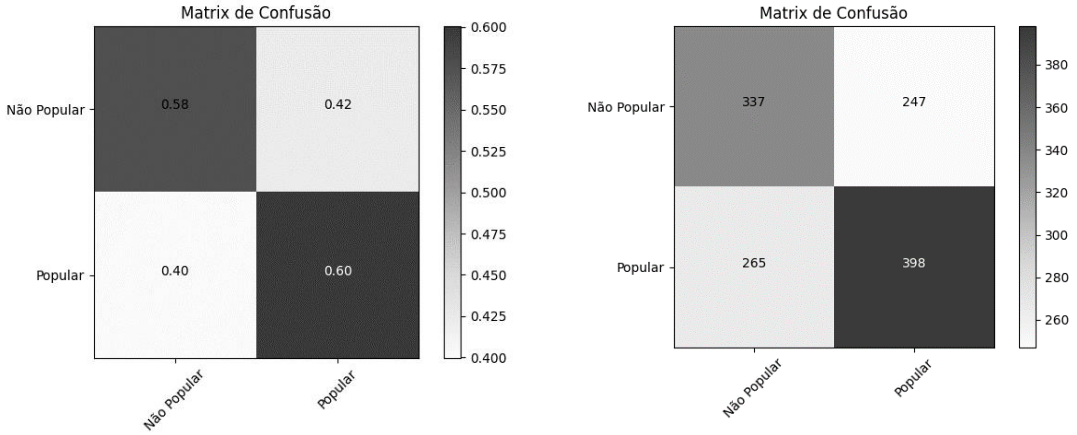

Figura 4. Matriz de Confusão – SVM - acurácia e número absoluto

No teste de predição a árvore de decisão obteve uma acurácia maior em comparação ao modelo *SVM* com *kernel linear*, também com a árvore de decisão é possível obter o peso das métricas dentro da classificação, isto é, o peso que cada variável tem nesta execução do modelo.

Tabela 3. Importância das métricas na árvore de decisão

Métrica	**Importância na Classificação (%)**

Capacidade de dança	14.06
Energia	0.00
Volume	7.37
Discurso	1.79
Acústica	5.72
Instrumentalidade	0.00
Vivacidade	0.00
Valência	12.31
Duração em ms	0.00
Término do fade-in	0.00
Começo do fade-out	0.00
Tempo	1.02
Confiança do tempo	0.00
Fórmula de compasso	0.00
Confiança da fórmula de compasso	6.24
Chave	0.00
Confiança da chave	0.00
Modo	0.00
Confiança do modo	0.00
Total de palavras	37.48
Repetição de radicais	3.43
Média de repetição por radical	0.00
Mediana de repetição por radical	0.00
Moda de repetição por radicais	0.00
Quantidade de palavras únicas	0.00
Valor máximo de repetições	0.00
Valor mínimo de repetições	2.72

Quantidade de Substantivos	0.00
Quantidade de Verbos	7.85
Quantidade de Advérbios	0.00

Com os dados representados na tabela 2, é possível observar uma mesclagem ente métricas sonoras e de PLN, os dados coletados via PLN obtiveram uma relevância importante na classificação, como por exemplo a quantidade total de palavras obtido com o uso da tokenização e a quantidade de verbos obtido com o método de tagueamento. Os demais dados referentes a PLN não foram incluídos com os parâmetros Gini e de poda da árvore de decisão.

5. Conclusões e Trabalhos Futuros

Nos resultados apresentados é possível observar que nos modelos propostos algumas funcionalidades são consideradas importantes na relação Música Vs. Popularidade levando em consideração o classificador "Árvore de Decisão", tanto nos aspectos sonoros quanto na sua relação sintática da letra, o dado mais relevante neste modelo é mostrado como o total de palavras na letra, portanto demonstrando que, existe uma relação no modo como uma música é escrita em sua popularidade. Dados relativos à sonoridade tiveram peso no que tange a classificação de popularidade enquanto outros não estiveram presentes no modelo configurado, isso não significa que são irrelevantes nesse modelo, só demonstraram que para a poda proposta com as características de coeficiente *Gini* eles não foram selecionados pelo algoritmo para geração de nós da árvore.

Como recomendação para trabalhos futuros e continuidade, recomenda-se foco em análise de sentenças na letra, agrupar os dados por gêneros músicas ou divisão de histórica de lançamento de músicas, além de sempre atualizar a base de variáveis sonoras do *Spotify*.

Referências

Barr, A. (1980) "Natural Language Understanding". "AI Magazine", Palo Alto, v.1, n.1.

Bird, S. Klein, E. Loper, E. (2009) "Natural Language Processing with Python". O'reilly Media, Sebastopol, Estados Unidos, 1 ed, p.15.

Cano, P, et. al. (2005) "Natural language processing of lyrics". "Proceedings of the 13th ACM International", Singapura, p. 475-478.

Dale, R. (2010) "Handbook of natural language processing". Londres: Taylor & Francis Group, n 1, p. 4-8.

Hawkins, D. (2004) "The Problem of Overfitting". J. Chem. Inf. Comput. Sci. 44, 1, p.1-12.

IFPI (Org). (2021) "IFPI issues Global Music Report 2021", Londres.

Jain, A. Kulkarni, G. Shah, J. (2018) "Natural Language Processing" International Journal of Computer Sciences and Engineering, p.166.

Lorena, A. Carvalho, A. (2007) “Uma Introdução às Support Vector Machines. “RITA Volume XIV, Número 2.

Lovins, J. B. (1968) “Development of a stemming algorithm. Mechanical Translation and Computational Linguistics”, Vol. 11 p. 22.

Maaso, A.; Hagen, N. (2020) “Metrics and decision-making in music streaming”. Noruega: Popular Communication, v. 18, n. 1, p. 18-31.

Meyer, D. (2017) “Support Vector Machines The Interface to libsvm in package e1071”, p. 1.

Miasato, V. Gonçalves, B. Costa, B. Silva, J. (2014) “Modelos de Predição Estruturada em Part-of-Speech Tagging para Português do Brasil”, p.1.

Neto, L. (2021) “A indústria fonográfica no século XXI: A popularização das plataformas de streaming”. Música em Foco, São Paulo, v. 3, n. 1, p. 58-75.

Noble, W. S. (2006) “What is a support vector machine? Nature Biotechnology”, Nat Biotechnol 24, p. 1565.

Panda, R; Redinho, H; Gonçalves; Malheiro, R; Paiva, R. (2021) “HOW DOES THE SPOTIFY API COMPARE TO THE MUSIC EMOTION RECOGNITION STATE-OF-THE-ART?”. “Proceedings of the 18th Sound and Music Computing Conference”, University of Coimbra, Portugal, p. 238-240.

Pereira, S.L. (2011) “Processamento de Linguagem Natural”. Universidade de São Paulo, São Paulo, p. 1.

Porto, F. Ziviani, A. (2014) “Ciência de Dados”. III Seminário Dos Grandes Desafios Da Computação - Fase 2.

Prina, B. Trentin, R. (2015) “GMC: Geração de Matriz de Confusão a partir de uma classificação digital de imagem do ArcGIS®”.

Silva, L. Peres, S. Boscarioli, C. (2016) “Introdução à mineração de dados: com aplicações em R.” 1. ed. Rio de Janeiro: GEN LTC, p. 101.

Scikit-learn: Machine Learning in Python, (2011) Pedregosa et al., JMLR 12, p. 2825-2830.

Spotify Investors (Org). (2021) “Spotify Technology S.A. Announces Financial Results for Third Quarter 2021”, Nova York.

Stone, M. (1974) “Cross-validatory Choice and Assessment of Statistical Predictions”. Journal of the Royal Statistical Society. Series B (Methodological), Vol. 36, No. 2. p. 111-147.