Hit Predictor - Previsão de Sucessos Musicais

Marcello Gonzatto Birkan[1], **Daniela Brazolin Flauto**[1], **Amanda Gois Smanioto**[1]

[1]Ciência da Computação
Faculdade de Computação e Informática
Universidade Presbiteriana Mackenzie
São Paulo – SP – Brasil

{marcello.birkan, daniela.flauto, amanda.smanioto}@mackenzista.com.br

1. Introdução

O mercado musical tem enfrentado um desafio crescente: prever o sucesso de uma música antes de seu lançamento. Diversos fatores, como estratégias de marketing, tendências culturais e o comportamento do público, desempenham papéis cruciais nesse processo. No entanto, a ausência de ferramentas robustas que analisem as características musicais de uma faixa para prever sua probabilidade de se tornar um hit cria incertezas para artistas, gravadoras e produtores. Este projeto busca solucionar essa lacuna, desenvolvendo um modelo de aprendizado de máquina capaz de prever a probabilidade de sucesso de uma música, utilizando dados tabulares extraídos da API do Spotify.

A crescente aplicação de inteligência artificial e aprendizado de máquina em diversas áreas tem revolucionado também o campo da música. Embora a composição e produção musical ainda sejam processos predominantemente criativos e humanos, essas tecnologias têm se mostrado ferramentas valiosas para auxiliar na criação e na análise de faixas musicais. No contexto atual, experimentos estão cada vez mais sendo realizados para entender e aplicar métodos geracionais que possam criar músicas com características semelhantes às criadas por seres humanos, ou, pelo menos, sonoramente agradáveis e potencialmente bem-sucedidas.

2. Referencial teórico

O sucesso musical é frequentemente associado a características sonoras específicas, como energia, dançabilidade, valência e tempo. O estudo da neurociência tem proporcionado importantes contribuições para diversas áreas, incluindo a música. Áreas como percepção auditiva, a relação entre música e movimento, a interação entre música e memória, além de investigações sobre o impacto emocional da música, são alguns dos campos de estudo que mais se destacam. Um aspecto fundamental na música, tanto em sua percepção quanto em sua produção, é a capacidade de gerar interações auditivo-motoras no cérebro, tanto do ouvinte quanto do executor. [Rocha and Boggio 2024]

Em relação ao uso de algoritmos para análise de música, estudos recentes indicam que técnicas de Machine Learning(Aprendizado de Máquina) podem identificar padrões em grandes volumes de dados e fornecer previsões úteis para cenários complexos. O aprendizado de máquina envolve a criação de algoritmos capazes de melhorar seu desempenho por meio de exemplos, permitindo que os computadores aprendam a partir de dados sem necessidade de programação explícita para cada tarefa.[Paulo 2022]

Dentre os algoritmos mais utilizados em problemas de classificação, destacam-se Random Forest, Gradient Boosting e Regressão Logística, cada um com características próprias que os tornam eficazes em diferentes contextos. Random Forest é uma técnica robusta para tarefas de classificação e regressão, construída a partir de múltiplas árvores de decisão. Durante

o treinamento, o modelo cria várias árvores baseadas em subconjuntos aleatórios dos dados, o que ajuda a reduzir a variância e a evitar o overfitting. Essa abordagem permite ao Random Forest lidar com grandes volumes de dados e múltiplas variáveis, sendo amplamente utilizado em áreas como análise de dados médicos e previsão de tendências de mercado. [IBM 2024]

Já o Gradient Boosting é uma técnica iterativa que constrói modelos aditivos, ajustando árvores de decisão ao gradiente negativo da função de perda em cada estágio. Esse processo permite a otimização de funções de perda diferenciáveis arbitrárias, proporcionando resultados eficazes em problemas de classificação e regressão.[Gra 2024]

A Regressão Logística, por sua vez, é um modelo estatístico utilizado para estimar a probabilidade de ocorrência de um evento, categorizando-o em uma das classes possíveis. Amplamente utilizado em classificação binária e multiclasse, este modelo aplica uma função logística para calcular a probabilidade de uma variável dependente binária, ajustando os dados de entrada por meio de uma combinação linear das variáveis preditoras.[Log]

Por fim, as métricas AUC-ROC e F1 Score são essenciais para avaliar o desempenho de modelos de classificação. A AUC-ROC (Área Sob a Curva da Característica de Operação do Receptor) mede a capacidade do modelo de distinguir entre classes positivas e negativas em diferentes limiares. Quanto mais próxima de 1 for a AUC, melhor será a capacidade de diferenciação do modelo, enquanto um valor próximo de 0,5 sugere um desempenho aleatório. Já o F1 Score é a média harmônica entre a precisão e o recall, oferecendo uma medida balanceada de desempenho, especialmente útil em situações onde os dados são desbalanceados.[Keylabs 2024]

3. Métodos

O projeto utilizou a API do Spotify como fonte principal de dados, extraindo características musicais relevantes como danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence e tempo. Esses dados foram coletados a partir de playlists populares, como Top 50 Global, Today's Top Hits, Viral 50 Hits, Global Top Ever e Hot Hits Global.

Na etapa de modelagem, três algoritmos de machine learning foram treinados: Random Forest, conhecido por sua capacidade de capturar relações não lineares e sua robustez contra overfitting; Gradient Boosting, eficiente no aprendizado sequencial e com boa performance em dados numéricos; e Regressão Logística, escolhida como modelo base pela sua simplicidade e boa interpretabilidade. Todo o desenvolvimento foi realizado em Python 3.9+, utilizando bibliotecas como Scikit-learn, Pandas e NumPy, além de ferramentas para visualização e interface, como Streamlit, e para gestão do ambiente, como Docker.

model_training.py

O código principal tem como objetivo treinar três modelos diferentes: o Random Forest Classifier, que é robusto e capaz de capturar relações complexas nos dados; o Gradient Boosting Classifier, otimizado para dados numéricos e com alta capacidade preditiva; e a Logistic Regression, utilizada como modelo base devido à sua simplicidade e alta interpretabilidade. Após o treinamento, o código avalia o desempenho de cada modelo utilizando métricas como AUC-ROC, Precisão e F1-Score, armazenando os resultados para futuras análises. Além disso, os modelos treinados são salvos para serem reutilizados posteriormente.

O código inicia com o carregamento dos dados, provenientes de um arquivo `.csv` chamado `data/processed_data.csv`, que contém as características musicais das músicas,

como danceability, valence, tempo, entre outras. A classe DataPreprocessor é responsável por dividir os dados em conjuntos de treino e teste, garantindo uma separação adequada para o treinamento e avaliação do modelo. Esse processo é feito com o seguinte código:

```
df = pd.read_csv('data/processed_data.csv')
preprocessor = DataPreprocessor()
X_train, X_test, y_train, y_test = preprocessor.split_data(df)
```

Em seguida, a classe ModelTrainer gerencia o treinamento dos modelos. Para cada algoritmo, o método `fit()` é utilizado para ajustar o modelo aos dados de treino, e o método `predict()` é utilizado para fazer predições sobre o conjunto de teste. Além disso, o método `predict_proba()` é empregado para calcular as probabilidades de cada classe. As métricas de avaliação são então calculadas para cada modelo, sendo a AUC-ROC a principal métrica para avaliar a capacidade de distinção entre as classes. O código para calcular essa métrica é o seguinte:

```
results['auc_roc'] = roc_auc_score(y_test, y_prob)
```

Além disso, para os modelos que permitem, como o Random Forest e o Gradient Boosting, o código coleta e armazena a importância de cada característica musical para as predições, permitindo analisar quais fatores são mais relevantes para a decisão do modelo. Isso é feito com o seguinte trecho:

```
results['feature_importance'] = dict(zip(
    self.preprocessor.features,
    model.feature_importances_
))
```

Em seguida, a classe ModelTrainer gerencia o treinamento dos modelos. Para cada algoritmo, o método `fit()` é utilizado para ajustar o modelo aos dados de treino, e o método `predict()` é utilizado para fazer predições sobre o conjunto de teste. Além disso, o método `predict_proba()` é empregado para calcular as probabilidades de cada classe. As métricas de avaliação são então calculadas para cada modelo, sendo a AUC-ROC a principal métrica para avaliar a capacidade de distinção entre as classes. O código para calcular essa métrica é o seguinte:

```
joblib.dump(model, f'models/trained_models/{name}.joblib')
results_df.to_csv('models/model_results.csv', index=False)
```

4. Resultados

Os resultados de desempenho dos modelos treinados foram avaliados com base em três métricas principais: AUC-ROC, Precisão e F1-Score. A AUC-ROC (Área Sob a Curva de Operação do Receptor) é uma métrica crucial que indica a capacidade do modelo de distinguir entre as classes positivas e negativas. Um valor de AUC-ROC próximo de 1 sugere uma excelente separação entre as classes. O F1-Score, por sua vez, é uma métrica que combina precisão e recall, sendo útil especialmente quando há desequilíbrio entre as classes.

Random Forest

AUC-ROC de 0.98, Precisão de 0.73 e F1-Score de 0.84. O modelo apresentou a melhor combinação de precisão e F1-Score, destacando-se como o mais eficaz para a tarefa de previsão de sucesso musical.

Gradient Boosting

AUC-ROC de 0.96, Precisão de 0.87 e F1-Score de 0.73. Embora tenha obtido uma precisão superior, o modelo teve um desempenho ligeiramente inferior no equilíbrio entre precisão e recall, o que afetou o F1-Score.

Regressão Logística

AUC-ROC de 0.93, Precisão de 0.48 e F1-Score de 0.65. Como esperado, devido à sua simplicidade, a Regressão Logística obteve resultados inferiores, especialmente na precisão, mas ainda assim se mostrou eficaz como modelo base para comparações.

O Random Forest se destacou principalmente pela sua robustez na captura de relações não lineares, o que resultou em uma maior capacidade de separar as classes. Seu F1-Score elevado sugere que ele tem um bom equilíbrio entre os falsos positivos e falsos negativos. Já o Gradient Boosting, embora muito eficiente em termos de precisão, mostrou uma leve queda no F1-Score, indicando que o modelo pode ter gerado mais falsos positivos, o que pode ser um aspecto a ser ajustado em futuras iterações.

Além dos resultados quantitativos, a interface interativa desenvolvida com Streamlit também desempenhou um papel importante na avaliação dos modelos, permitindo simulações em tempo real de possíveis hits antes mesmo de sua criação. Isso representa um avanço significativo para os profissionais da música, oferecendo uma ferramenta inovadora no processo de previsão de sucesso musical. Gráficos produzidos com Matplotlib e Seaborn ajudaram a visualizar as diferenças de desempenho entre os modelos, facilitando a interpretação dos resultados.

Esses resultados mostram que o Random Forest é o modelo mais promissor para a tarefa de prever o sucesso de músicas, com a combinação de robustez e precisão. No entanto, cada modelo tem seus pontos fortes, e a escolha do modelo ideal pode depender de fatores como a necessidade de interpretabilidade ou a priorização de precisão em relação ao equilíbrio entre precisão e recall.

5. Conclusão

Este projeto teve como objetivo explorar a viabilidade de prever o sucesso de músicas utilizando algoritmos de aprendizado de máquina, com base em dados musicais extraídos da API do Spotify. Através da análise de características como dançabilidade, energia, valência, entre outras, conseguimos construir e avaliar três modelos distintos: Random Forest, Gradient Boosting e Regressão Logística, para classificar músicas com base em seu potencial de sucesso. Os resultados demonstraram que o modelo Random Forest se destacou em termos de desempenho, apresentando a melhor combinação entre AUC-ROC, Precisão e F1-Score. Sua capacidade de capturar relações não lineares entre as variáveis e sua robustez contra overfitting tornaram-no o modelo mais eficaz para prever o sucesso musical.

Além dos resultados quantitativos, a interface interativa desenvolvida com o Streamlit mostrou-se util, permitindo que os usuários simulem o potencial de sucesso de músicas antes mesmo de seu lançamento. Essa funcionalidade pode ser uma ferramenta para profissionais da indústria musical, fornecendo feedbacks rápidos e dados empíricos que podem auxiliar em decisões de marketing e produção.

Em conclusão, este estudo mostrou que, com o uso adequado de dados musicais e técnicas de aprendizado de máquina, é possível prever com alta precisão o sucesso de músicas, proporcionando uma ferramenta poderosa para a indústria musical. No entanto, como todo modelo preditivo, há limitações que devem ser consideradas, como a necessidade de dados mais

robustos e a evolução das tendências musicais, que podem afetar a eficácia do modelo ao longo do tempo.

Referências Bibliográficas

Logistic regression (aka logit, maxent) classifier. https://www.example.com/logistic-regression. Acessado em: 20 de novembro de 2024.

(2024). Gradient boosting for classification. Acessado em: 20 de novembro de 2024.

IBM (2024). Random forest: Algoritmo de aprendizado de máquina. Acessado em 20 de novembro de 2024.

Keylabs (2024). Understanding the f1 score and auc-roc curve. Accessed: 2024-11-20.

Paulo, M. B. d. (2022). Geração de música com machine learning. In *Trabalhos de Conclusão de Curso de Graduação, Universidade Federal de Santa Catarina*.

Rocha, V. C. d. and Boggio, P. S. (2024). A música por uma óptica neurocientífica. *Universidade Presbiteriana Mackenzie*.