CAEd UFJF
PROVA – EDITAL 015/2025 - DESENVOLVIMENTO DE TECNOLOGIA
ESTAGIÁRIO EM PESQUISA DE AVALIAÇÃO

Relatório de Desenvolvimento - Análise de Letras de Músicas

Luan Henrique da Silva Barbosa

30 de junho de 2025

Sumário

1 Introdução

Este relatório descreve o processo de análise realizado sobre uma base de dados (dataset) composta por letras de música e outros dados relacionados como título, artista e gênero. O trabalho foi dividido em quatro tarefas principais, cada uma com objetivos bem definidos: análise de polaridade, predição de gênero musical, geração de letras e construção de redes de colaboração entre artistas.

Em vista de realizar um trabalho preciso e de qualidade, antes da execução dessas tarefas, foram realizadas etapas preliminares fundamentais. Essas etapas compreendem: a leitura da base, uma amostragem dos dados, o pré-processamento textual e uma breve análise exploratória.

Dessa forma, este relatório está organizado da seguinte maneira. A seção 2 descreve as ferramentas utilizadas, enquanto a seção 3 comenta sobre a leitura da base de dados. A seção 4 apresenta uma breve análise exploratória e a seção 5 detalha o procedimento de pré-processamento. Já as seções de 6 a 9 tratam das tarefas principais mencionadas. Por fim, a seção 10 conclui o trabalho.

2 Ferramentas Utilizadas

As implementações foram realizadas em *Python*, linguagem escolhida por oferecer uma ampla gama de bibliotecas voltadas para Processamento de Linguagem Natural (NLP) e análise de dados. O ambiente utilizado foi o Google Colab[1], visando evitar limitações de memória RAM e CPU do computador local, além de possibilitar testes mais rápidos e práticos. As principais bibliotecas utilizadas estão listadas a seguir, e o código está acessível em repositório público no GitHub[2]. As principais bibliotecas utilizadas foram:

- **Pandas e Numpy:** manipulação e estruturação de dados.
- **Matplotlib e Seaborn:** visualizações gráficas e análise exploratória.
- **NLTK (VADER):** análise de polaridade emocional.
- **Scikit-learn:** modelos de classificação e avaliação.
- **Imbalanced-learn (SMOTE):** balanceamento do conjunto de treino.
- **Tensorflow/Keras:** construção de modelos sequenciais.
- **NetworkX e Louvain:** modelagem e análise de redes de colaboração entre artistas.
- **Pyvis:** visualizações interativas de grafos.

[1] https://colab.google/

[2] https://github.com/LuanBarbs/caed-song-lyrics-nlp

3 Leitura e Amostragem

O dataset original possui aproximadamente 8,45 GB, o que pode causar travamentos ou consumo excessivo de memória RAM durante as etapas de análise, especialmente em ambientes com recursos limitados. Embora a biblioteca *pandas* não seja a mais eficiente para trabalhar com grandes volumes de dados, ela é amplamente utilizada em tarefas NLP. Dessa forma, adotaram-se estratégias para viabilizar a execução do projeto em um ambiente com 12,7 GB de RAM.

Uma estratégia aplicada foi carregar apenas as colunas relevantes (LABEX, 2025), além de aplicar a leitura por blocos (*chunksize*) (ICHI.PRO, 2025), reduzindo o consumo de memória. Ainda, considerando que muitas tarefas propostas envolvem diretamente o campo *lyrics* (as letras das músicas), foi realizado um filtro para manter apenas os textos em inglês — idioma predominante na base. Essa decisão buscou evitar interferências linguísticas indesejadas nos modelos e análises, além de contribuir para a redução do volume de dados.

Embora o uso de leitura em blocos (chunksize), filtro de colunas e apenas a língua inglesa tenham reduzido o consumo de memória, os requisitos de carga total em RAM para tarefas como vetorização textual ainda inviabilizaram o uso pleno da base (5,33 GB após filtragem). Por isso, adotou-se a amostragem de 10%, conforme sugestão da prova, o que permitiu fluidez no desenvolvimento. A escolha foi a de manter apenas textos em inglês nas tarefas (a), (b) e (c), visando reduzir viés linguístico e diminuir a complexidade. A tarefa (d) foi realizada com a amostra completa, independentemente do idioma.

4 Análise Exploratória

A análise exploratória do conjunto de dados foi feita com base em tutoriais técnicos, a saber: Cavalcante (2024) e Lima (2024). Foram realizadas análises básicas no conjunto, como tamanho e número de colunas. Além disso, foi feita uma busca acerca do ano de produção, número de visualizações, quantidade de músicas por ano, gêneros musicais mais frequentes e distribuição da quantidade de palavras.

A base amostrada utilizada nesse trabalho possui 513.486 registros e 8 colunas e a Tabela 1 apresenta um resumo da quantidade de valores ausentes por coluna. A maioria das informações está completa, mas a coluna *language* possui aproximadamente 4,4% de valores ausentes. Apesar disso, esses valores não são exatamente importantes nas tarefas, então a ausência pode ser ignorada.

Também foi gerado um gráfico (Figura 1) com os gêneros mais frequentes na base. O gênero Pop representa uma parte significativa da base, com mais de 200 mil registros. Gêneros como *R&B*, *Misc* e *Country* aparecem com menos de 25 mil. Isso evidencia um forte desbalanceamento, que pode afetar os modelos de predição de gênero na tarefa (b).

Adicionalmente, como as letras são extensas e podem conter textos não musicais

Tabela 1: Quantidade de valores ausentes por coluna.

Coluna	Valores Ausentes
title	20
tag	0
artist	0
year	0
views	0
features	0
lyrics	0
language	22.766

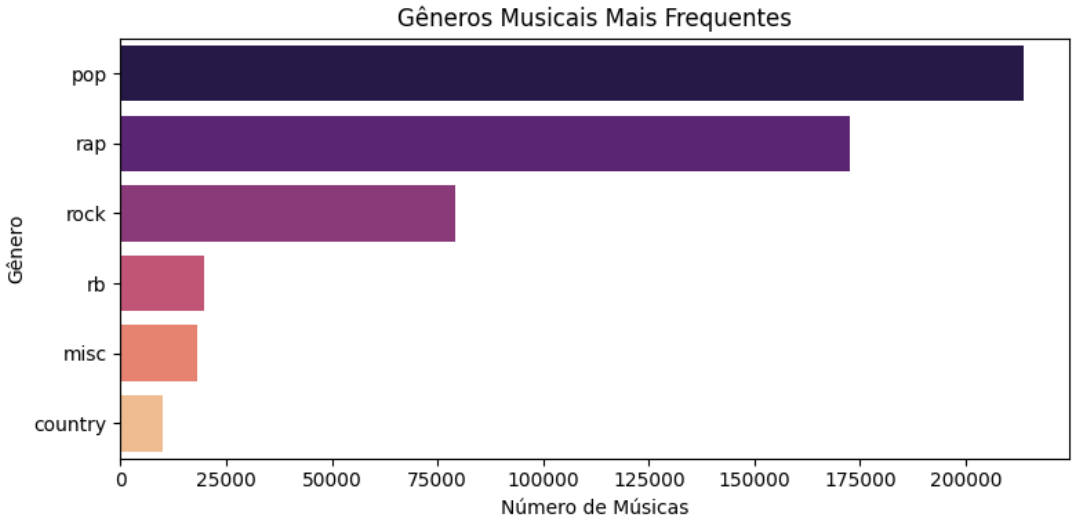

Figura 1: Gráfico dos gêneros musicais mais frequentes.

(como poemas e livros), foi realizada uma análise da quantidade de palavras por música. A média de palavras por letra foi de 307,66. Um histograma (Figura 2) foi plotado para visualização. A maioria das letras possui até 750 palavras e há um pequeno grupo entre 750 e 1000 palavras. Algumas entradas parecem ter um número de palavras muito maior que o normal, sugerindo que pode não se tratar de uma música comum, mas sim um livro.

Para investigar, foi localizado o registro com maior contagem de palavras. A entrada possui 17803 palavras, de título *On Duties De Officiis - Book III*. O título indica que realmente o exemplo se trata de um livro. Portanto, o número de palavras por letra pode ser um bom indicativo para remoção dos livros do conjunto de dados.

Para analisar a presença de poemas na base, adotou-se uma estratégia baseada em três critérios principais: (i) proporção elevada de linhas curtas (menos de 7 palavras), comum em poesias; (ii) baixa repetição de versos entre as linhas; e (iii) diversidade lexical mínima de 30%, para garantir riqueza no vocabulário. Além disso, foram aplicados filtros para garantir qualidade estrutural: exigência de pelo menos 8 linhas não vazias e uma média de palavras por linha não muito baixa.

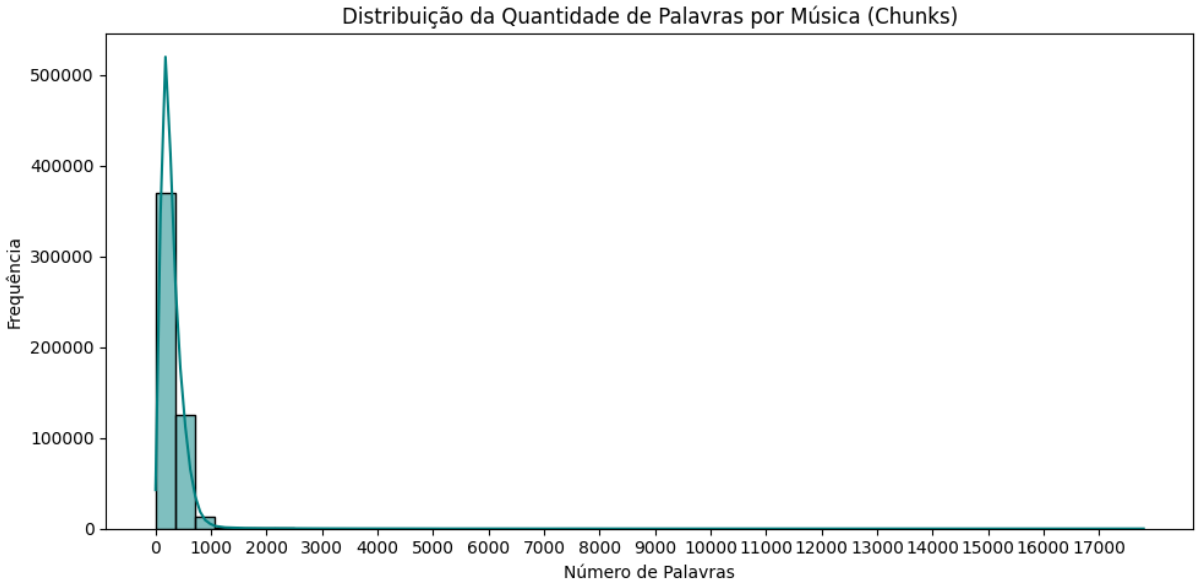

Figura 2: Histograma de palavras por música.

Essa abordagem foi aplicada apenas às entradas com gênero (*tag*) igual a *misc*, que concentravam obras potencialmente poéticas. A estratégia capturou aproximadamente 3 mil letras, com predominância de textos reconhecidamente poéticos — como Emily Dickinson, Langston Hughes e Catullus. Embora algumas músicas curtas também tenham sido incluídas, o impacto da filtragem é pequeno e não compromete a integridade do conjunto.

A próxima etapa consistiu na limpeza do conjunto de dados, com a remoção sistemática dessas entradas identificadas como poemas, além da exclusão de livros com base na contagem de palavras.

5 Pré-processamento

Visando a qualidade dos dados utilizados nas análises, foi realizada uma etapa de pré-processamento composta por dois principais procedimentos: remoção de entradas não musicais e limpeza textual do campo *lyrics*. Esses procedimentos são essenciais para a execução das próximas tarefas.

Primeiramente, as entradas correspondentes a livros (identificados por uma contagem elevada de palavras — mais de 2.000) e poemas (identificados por características estruturais) foram excluídas. Essa filtragem resultou na remoção de 5.448 entradas, restando uma base final de 508.038 letras de música. Em seguida, foi aplicada uma função de limpeza nas letras, que envolveu: padronização para letras minúsculas, remoção de URLs, substituição de quebras de linha por espaços, remoção de pontuação e caracteres especiais, exclusão de *stopwords* e palavras muito curtas e remoção de múltiplos espaços.

Com isso, a base encontra-se pronta para ser utilizada nas tarefas propostas, com a

intenção de se ter uma maior relevância nos resultados e uma redução da quantidade de livros e poemas, que poderiam representar ruídos para os modelos de análise.

6 Tarefa (a): Análise de Polaridade

6.1 Objetivo

O objetivo geral desta tarefa é determinar a polaridade (positiva, negativa ou neutra) das letras de músicas. Os objetivos específicos são:

- Analisar como a polaridade das músicas de um artista evoluiu ao longo do tempo;
- Identificar artistas que tiveram uma variação mais relevante de polaridade ao longo do tempo;
- Verificar se existe uma relação entre a polaridade da música e a quantidade de visualizações recebidas.

6.2 Processo

Inicialmente, o conjunto de dados foi filtrado para incluir apenas letras em língua inglesa, totalizando 332.731 músicas.

Com base no tutorial de Thuo (2024), a ferramenta escolhida para a análise de polaridade foi o modelo pré-treinado *VADER* (Valence Aware Dictionary and sEntiment Reasoner), amplamente utilizado para classificação de sentimentos em textos curtos.

O processo de análise de polaridade seguiu as seguintes etapas:

1. Aplicação de modelo de sentimento VADER.
2. Cálculo da polaridade de cada letra com base em uma pontuação composta por três escores (pos., neu., neg.) e uma *compound* (composição) geral. A classificação final (positiva, neutra ou negativa) foi atribuída com base em limiares do *compound*.
3. Avaliação da polaridade por artista e ano, permitindo observar sua evolução ao longo do tempo.
4. Cálculos do desvios padrão por artista para identificar aqueles com maior variação emocional ao longo dos anos.
5. Aplicação da regressão linear e polinomial para investigar a relação entre polaridade e número de visualizações das músicas.

6.3 Resultados

A distribuição geral da polaridade das músicas pode ser visualizada na Figura 3. A maioria das músicas apresenta polaridade negativa ou positiva, o que é coerente com a natureza emocional frequentemente presente em composições musicais.

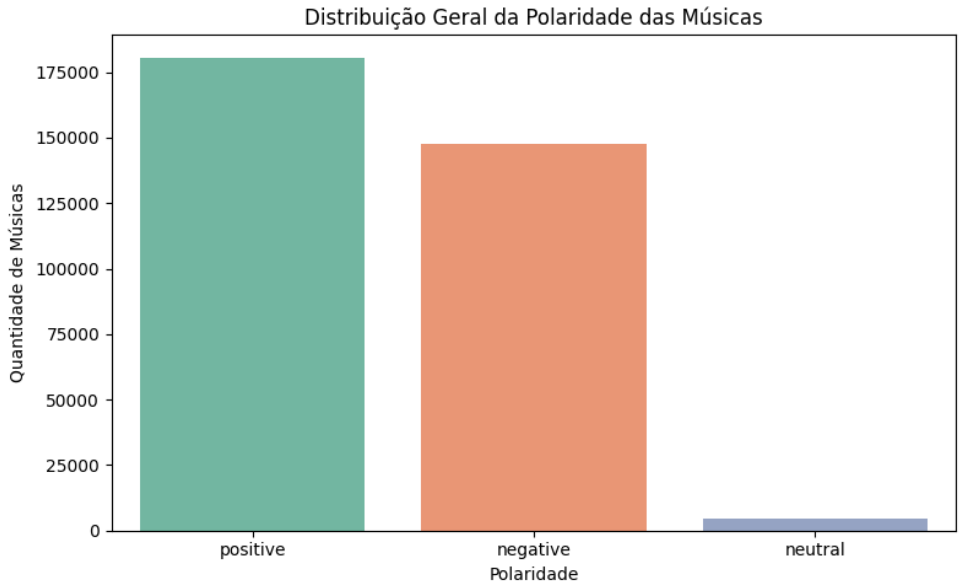

Figura 3: Distribuição geral da polaridade das músicas.

Para análise temporal, escolheu-se o artista *Lil B*. A Figura 4 apresenta a evolução de polaridade de suas músicas ao longo dos anos. Observa-se uma predominância de polaridade neutra, com algumas ocorrências de letras mais negativas.

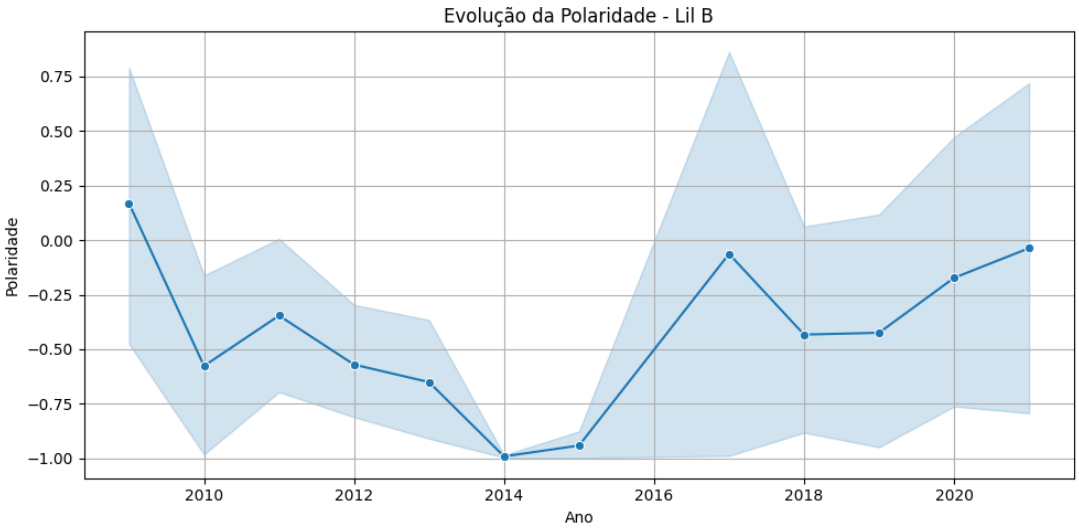

Figura 4: Gráfico da evolução de polaridade do artista Lil B.

Adicionalmente, sobre os artistas com maior variação de polaridade ao longo do tempo, foram analisadas as médias anuais da polaridade de cada artista com pelo menos cinco composi-

ções. Em seguida, calculou-se o desvio padrão dessas médias, permitindo mensurar a flutuação de sentimentos expressos nas letras ano a ano. Os 20 artistas com maior variação temporal foram destacados na Figura 5.

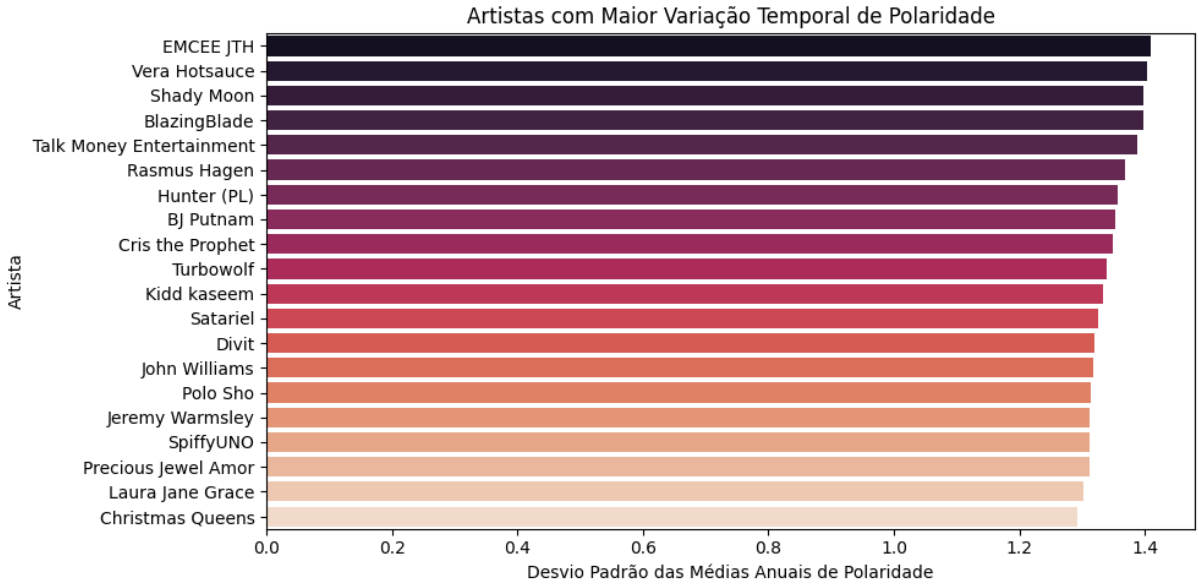

Figura 5: Gráfico de barras dos artistas com maior variação de polaridade ao longo do tempo.

Por fim, para investigar a relação entre polaridade emocional e popularidade das músicas, foi utilizada a métrica de visualizações (em escala logarítmica). A Figura 6 mostra um gráfico de dispersão com regressão linear. A correlação linear entre polaridade e log(views) foi baixa ($r = 0{,}024$), indicando ausência de associação linear.

Entretanto, ao ajustar um modelo de regressão polinomial de segundo grau, observou-se uma curva com formato de parábola. Isso sugere que músicas com polaridade mais extrema (positiva ou negativa) tendem a apresentar médias mais altas de visualizações, quando comparadas a músicas neutras. O coeficiente do termo quadrático foi $0{,}4937$, indicando efeito relevante nesse padrão. Apesar disso, o valor de R^2 obtido (0,005) indica que apenas 0,5% da variação no número de visualizações das músicas e explicada pela polaridade emocional. Isso sugere que a polaridade é apenas um dos muitos fatores que podem influenciar a popularidade.

7 Tarefa (b): Predição de Gênero Musical

7.1 Objetivo

O objetivo desta tarefa é construir um modelo de classificação capaz de predizer o gênero musical (*rap*, *pop*, entre outros) a partir do conteúdo textual das letras de músicas. Além disso, busca-se avaliar a qualidade do processo preditivo adotado.

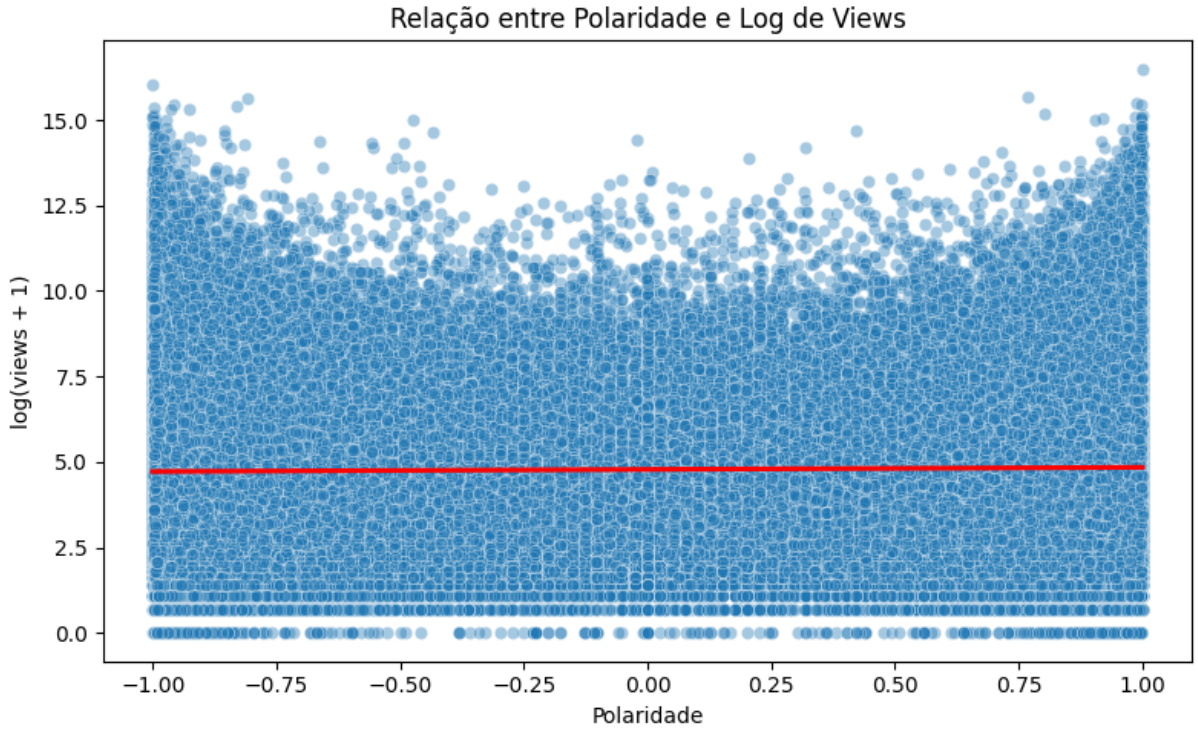

Figura 6: Gráfico da relação entre a polaridade e número de visualizações.

7.2 Processo

Para esta tarefa, a base de dados foi novamente restrita às músicas em inglês, resultando em um total de 332.731 letras para análise. O processo de predição seguiu as seguintes etapas:

1. Aplicação de pré-processamento textual, com vetorização baseada em **TF-IDF** (*Term Frequency-Inverse Document Frequency*).

2. Treinamento de dois modelos clássicos de classificação: **Logistic Regression** (LR) e **SVM** com kernel linear (*LinearSVC*).

3. Avaliação dos modelos com as métricas: **acurácia**, **precisão**, **sensibilidade**, **F1-score** e **F1-score macro** (de forma a considerar o impacto do desbalanceamento entre classes).

Um dos principais desafios enfrentados nesta tarefa foi o desbalanceamento entre os gêneros musicais. Isso impactou negativamente o desempenho, especialmente na capacidade de prever classes minoritárias. Para mitigar esse problema, foi empregada a técnica de balanceamento **SMOTE** (*Synthetic Minority Over-sampling Technique*), com o intuito de gerar exemplos sintéticos das classes menos representadas no conjunto de treino.

7.3 Resultados

Os resultados dos modelos treinados, considerando o conjunto original (desbalanceado) e com *oversampling* (balanceado), estão apresentados nas Tabelas 2 e 3, respectivamente.

Tabela 2: Resultados da predição de gênero no conjunto desbalanceado.

Modelo	Acurácia	Precisão	Sensibilidade	F1-score	F1-score macro
LR	**0,68**	**0,67**	**0,68**	**0,65**	**0,48**
LinearSVC	0,67	0,66	0,67	0,64	0,47

Tabela 3: Resultados da predição de gênero no conjunto balanceado.

Modelo	Acurácia	Precisão	Sensibilidade	F1-score	F1-score macro
LR	0,74	0,73	0,74	0,73	0,73
LinearSVC	**0,80**	**0,79**	**0,80**	**0,79**	**0,79**

Antes da aplicação do balanceamento, a distribuição dos gêneros era bastante desigual: *pop* e *rap* dominavam a base de dados, enquanto gêneros como *country* e *misc* apresentavam representatividade muito menor. Esse desequilíbrio dificultava a aprendizagem de padrões para as classes menos frequentes e impactava diretamente as métricas mais sensíveis ao desbalanceamento, como o *F1-score macro*, que foi inferior a 0,50 em ambos os modelos no cenário inicial.

Após aplicar a técnica de oversampling com SMOTE, o número de instâncias em cada classe foi igualado para 139.223 exemplos. Esse balanceamento proporcionou uma melhoria significativa em todas as métricas, sobretudo no *F1-score macro*, que passou a refletir uma performance mais justa entre as classes. O modelo *LinearSVC* se destacou nesse novo cenário, atingindo 80% de acurácia e *F1-score macro* de 0,79 — valores considerados satisfatórios para a tarefa.

Esses resultados evidenciam que técnicas de balanceamento de dados são fundamentais em problemas com distribuição desigual. Além disso, mostram que modelos lineares simples, aliados a uma vetorização eficaz (como TF-IDF), podem ser bastante competitivos em tarefas de classificação textual.

8 Tarefa (c): Geração de Letras

8.1 Objetivo

O objetivo desta tarefa é desenvolver um modelo sequencial capaz de gerar novas letras de músicas a partir de um texto inicial. A ideia é que o modelo aprenda os padrões estruturais presentes no conjunto original, sendo capaz de continuar uma sequência textual de forma coerente e similar às músicas da base.

8.2 Processo

Os principais desafios enfrentados nesta tarefa foram as restrições computacionais do ambiente (memória RAM limitada e tempo de execução), o que exigiu adaptações no pipeline de treinamento, como a utilização de geradores otimizados e redução do número de amostras por época.

Foram exploradas arquiteturas de apenas uma camada oculta e duas camadas empilhadas de LSTM, com aumento no número de épocas e tamanho do dataset de treino, buscando maximizar a capacidade do modelo sem exceder os limites computacionais disponíveis. O tempo total de treino foi de aproximadamente 1 (uma) hora, no ambiente Colab. Nesse sentido, o principal desafio foi o tempo total de treino e o uso excessivo dos recursos de GPU do ambiente, que impossibilitou gerar testes mais demorados.

Dessa forma, o processo de geração textual foi composto pelas seguintes etapas:

1. As letras foram concatenadas em um único texto, transformado em uma sequência de caracteres únicos.

2. A partir do texto completo, foram geradas amostras de entrada com tamanho fixo de 60 caracteres, juntamente com o caractere subsequente como rótulo, criando assim o conjunto de dados para treinamento supervisionado. Utilizou-se um passo de 5 posições para reduzir o tamanho do corpus e mitigar o consumo de memória.

3. Foi construído um modelo de rede recorrente do tipo LSTM com duas camadas ocultas de 128 unidades e regularização com *dropout*. O modelo foi treinado utilizando batches gerados dinamicamente, visando reduzir o uso de memória RAM. O tempo de treino foi de 100 épocas, com 3000 passos por época.

4. Para a etapa de geração, foi utilizada a técnica de *temperature sampling*, que permite controlar a aleatoriedade da saída com base na distribuição de probabilidade prevista pela rede. Temperaturas mais baixas geram textos mais previsíveis, enquanto temperaturas mais altas promovem maior diversidade e criatividade.

8.3 Resultados

A Tabela 4 apresenta exemplos de letras geradas a partir do modelo treinado (*loss* de 1,6659), com diferentes temperaturas, começando com o texto "just stay with".

Com base na análise de sensibilidade da temperatura, obervam-se alguns pontos. Temperaturas baixas (0,1 a 0,3) geram mais repetições, como "*think think*", sugerindo que o modelo aprendeu padrões de repetição, mas que não consegue ser criativo. Já com temperaturas médias (0,4 a 0,6), o modelo mostra um pouco de criatividade e maior vocabulário, mas ainda com frases sem coesão, como "*oroan hook*". Por fim, temperaturas altas (0,7 a 0,9) geram letras com palavras sem sentido, como "*bep*", além de erros ortográfios e uma falta de estrutura. Esses

Tabela 4: Exemplo de geração de letra com diferentes temperaturas.

Temperatura	Tempo Gerado
0,1	just stay without start start think think show show shit shit show shit shit shouldnt see like shit see shouldnt s
0,2	just stay without see see pressure bitch better soul think think think come cant see start show shouldnt think say
0,3	just stay without dont always think like call like make something cant let go see start chorus time try see sing c
0,4	just stay without think take cant see around chorus think oroan hook could book like heart think tomight love life
0,5	just stay without let see like dont give pote money like got come something got really think got feel real come fi
0,6	just stay without cenl im one love shit bust thing say never im make dance boom verse burn somemne like got head b
0,7	just stay without money too gotta past find curse get dont fild time mind wait come back come pray lie show damp l
0,8	just stay without got ill give tarter take revercer im beat forde believe line youre everybody dont want imtro ill
0,9	just stay without something calming money baby bitch line demasse shadows em means bep four come parevl could ei m

erros podem acontecer devido à aleatoriedade mais intensa, mas mostram que o modelo não lidou bem com a gramática e fluidez.

Embora o modelo tenha gerado letras básicas e com vocabulário relacionado a letras musicais, os exemplos evidenciam limitações em termos de coesão e criatividade real. A limitação do modelo LSTM de duas camadas e do tempo de treino reduzido podem ter contribuído para esses resultados. Ainda assim, o experimento cumpre um papel exploratório, sugerindo que melhorias, como uso de *Transformers* ou de mais camadas LSTM podem ter maior capacidade de generalização e criatividade.

9 Tarefa (d): Rede de Colaboração

9.1 Objetivo

Esta tarefa foi feita visando construir um grafo de colaboração entre artistas. A partir desse grafo, buscou-se identificar padrões de colaboração e detectar comunidades de artistas frequentemente conectados entre si.

9.2 Processo

O processo de construção da rede de colaboração foi realizado com os seguintes passos:

1. Extração dos pares colaborativos, utilizando o campo *artist* como artista principal e o campo *features* para identificar colaborações.

2. Construção do grafo, adicionando uma aresta entre cada par de artistas colaborativos, ponderada pela quantidade de músicas em que colaboraram.

3. Cálculo de métricas de estrutura da rede: grau médio, densidade, componentes conectados, artistas com maior grau e centralidade.

4. Detecção de comunidades com o algoritmo de *Louvain*, uma técnica amplamente utilizada para encontrar agrupamentos densamente conectados em grafos.

5. Geração de visualizações estáticas e interativas, focando em subconjuntos relevantes do grafo.

9.3 Resultados

A Tabela 5 resume as principais métricas extraídas do grafo completo de colaboração entre artistas. A rede resultante apresenta densidade praticamente nula (0,0000), indicando uma estrutura extremamente esparsa. Observou-se a existência de 36.909 componentes conectados, sendo que a maior delas reúne 39.523 artistas, revelando um núcleo colaborativo considerável. A aplicação do algoritmo de *Louvain* permitiu identificar 37.115 comunidades distintas, sugerindo que colaborações tendem a ocorrer em grupos específicos e relativamente isolados.

Tabela 5: Resumo estrutural da rede de colaboração entre artistas.

Total de nós (artistas)	132.799
Total de arestas (colaborações)	107.849
Grau médio	1,62
Densidade da rede	0,0000
Número de componentes conectados	36.909
Maior componente conectada	39.523 artistas
Número de comunidades detectadas (Louvain)	37.115

A análise dos nós mais conectados revelou que os artistas com maior número de colaborações foram *Walt Disney Records* (com 131 conexões), *Gucci Mane* (com 114), *Various Artists* (108), *Lil Wayne* (106) e *Snoop Dogg* (95). Além disso, a inspeção das parcerias mais frequentes revelou que muitas delas não correspondem a colaborações artísticas tradicionais, mas sim a grupos de tradução de músicas, como, por exemplo: *Genius Brasil Traduções com*

Genius Brasil Tradues (889 músicas), *Genius Traducciones Al Español com Genius Traducciones Al Espaol* (680), e *Genius Traductions Françaises com Genius Traductions Franaises* (463). Esse achado evidencia um comportamento particular da base, que incorpora metadados linguísticos como nós na rede.

No que diz respeito à centralidade de grau, que indica a importância de um nó em termos de conexões diretas, os artistas mais centrais foram *Walt Disney Records* (0,0010), *Gucci Mane* (0,0009) e *Lil Wayne* (0,0008). Esses valores, embora baixos em termos absolutos devido ao tamanho da rede, destacam artistas que possuem uma quantidade significativa de colaborações.

A Figura 7 apresenta uma visualização do grafo completo, com coloração por comunidades detectadas via *Louvain*. Para facilitar a interpretação, uma versão interativa da rede contendo apenas artistas com grau maior ou igual a 50 foi gerada. Essa versão mais enxuta permite uma visualização mais clara das principais conexões e estruturas colaborativas.

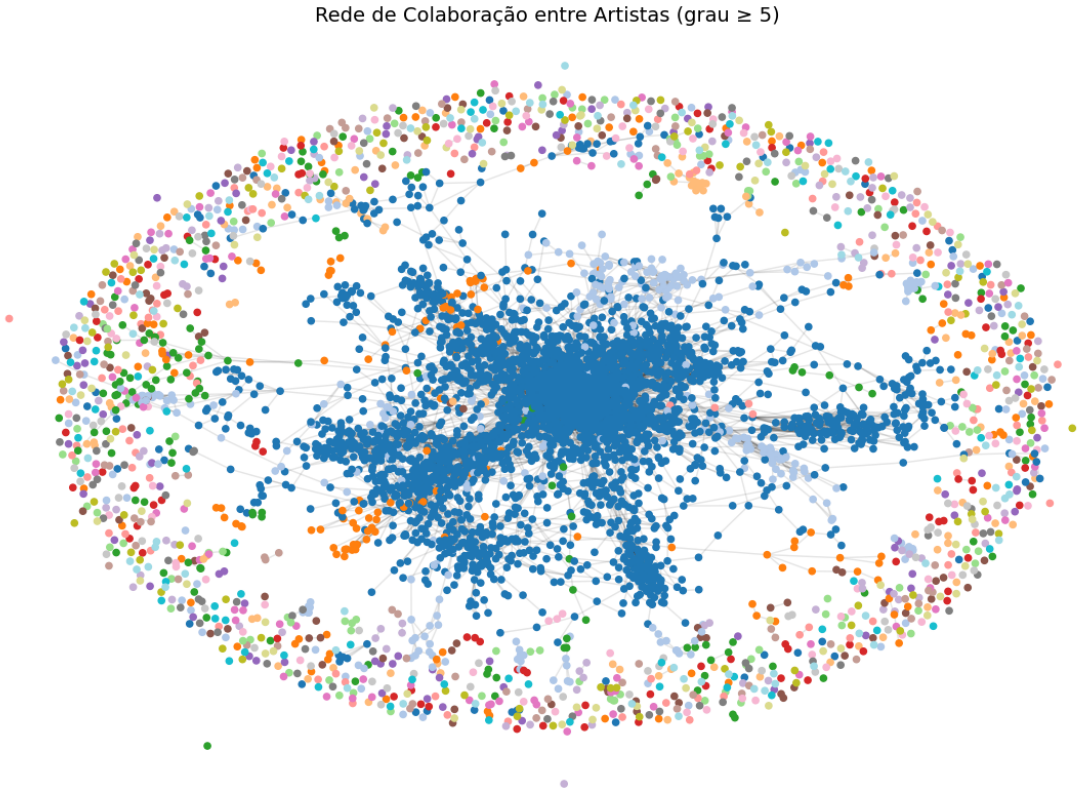

Figura 7: Grafo de colaboração entre artistas. Cores representam diferentes comunidades detectadas.

Portanto, a análise da rede de colaboração revelou uma estrutura altamente esparsa, com poucas regiões densamente conectadas. Apesar de a maioria das colaborações ocorrer em pequenos grupos isolados, foi possível identificar um núcleo expressivo de interações artísticas. A detecção de comunidades e o estudo de centralidade se mostraram ferramentas úteis para

entender a dinâmica de colaboração na base, ao mesmo tempo, em que permitiram identificar padrões atípicos, como agrupamentos relacionados a traduções.

10 Considerações Finais

Este trabalho teve o objetivo de apresentar um relatório sobre as tarefas desenvolvidas sobre uma base de dados disponibilizada pelo **CAEd** para fins de avaliação. A manipulação do grande conjunto de dados foi um desafio significativo em termos de consumo de memória RAM. Para mitigar essas limitações e viabilizar o desenvolvimento, estratégias cruciais foram tomadas, como leitura por bloco e amostragem. Adicionalmente, o pré-processamento dos dados foi fundamental ao identificar e remover entradas não musicais que poderiam introduzir ruídos, bem como para realizar a limpeza textual das letras.

As quatro tarefas propostas foram executadas, gerando informações e percepções relevantes:

Na primeira tarefa foi realizada uma análise de polaridade, possível através do modelo VADER e embora a relação entre polaridade e visualizações seja pequena, observou-se que músicas com polaridade mais extrema tendem a ter mais visualizações.

A segunda tarefa tratou da predição do gênero musical. O desbalanceamento entre os gêneros apresentou-se como um desafio para os modelos se adequarem aos dados. Isso foi superado com a aplicação da técnica SMOTE, resultando em uma melhoria significativa no desempenho ao alcançar 80% de acurácia e 79% de F1-score macro. Logo, demonstra-se a capacidade e competitividade dos modelos lineares, junto da vetorização, em tarefas de classificação textual.

A terceira tarefa utilizou um modelo sequencial baseado em LSTM para gerar novas letras de música a partir de um texto inicial. Apesar das restrições computacionais e de tempo, o experimento demonstrou capacidade em aprender os padrões e vocabulários musicais. Observou-se que temperaturas de amostragem influenciam a criatividade e coesão, com temperaturas mais baixas gerando repetição e mais altas gerando textos sem sentido.

Na quarta tarefa, um grafo foi construído para analisar uma rede de colaboração entre artistas. O grafo gerado é altamente esparso, indicando poucas relações. Entretanto, um excesso de comunidades distintas foi identificado, sugerindo que colaborações tendem a ocorrer em grupos específicos e isolados. Além disso, uma observação importante foi a identificação de grupos de tradução de músicas entre as parcerias mais frequentes, evidenciando que a base incorpora dados de traduções das músicas.

Em geral, o trabalho demonstra a importância de estratégias de pré-processamento e amostragem para lidar com complexidades inerentes ao conjunto de dados. As estratégias adotadas fornecem uma base para futuras investigações, como exploração de modelos mais avançados (como *Transformers* para geração de texto) ou aprofundamento na análise de fatores que influenciam a popularidade das músicas e artistas.

Referências

CAVALCANTE, Nadinne. **Análise Exploratória de Dados com Python: Um Guia Básico**. [S.l.: s.n.], 2024. Acesso em: 29 jun. 2025. Disponível em: <https://medium.com/@nadinne.cavalcante94/an%C3%A1lise-explorat%C3%B3ria-de-dados-com-python-um-guia-b%C3%A1sico-055e3e7e8c6c>.

ICHI.PRO. **Carregando grandes conjuntos de dados no Pandas**. [S.l.: s.n.], 2025. Acesso em: 28 jun. 2025. Disponível em: <https://ichi.pro/pt/carregando-grandes-conjuntos-de-dados-no-pandas-19507168112507>.

LABEX. **Escalando Grandes Conjuntos de Dados com Pandas - Guia Prático**. [S.l.: s.n.], 2025. Acesso em: 28 jun. 2025. Disponível em: <https://labex.io/pt/tutorials/pandas-scaling-large-datasets-65453>.

LIMA, Vinícius Rocha. **Turbinando as suas Análises Exploratórias - Vinícius Rocha Lima - Medium**. [S.l.: s.n.], 2024. Acesso em: 29 jun. 2025. Disponível em: <https://viniciusrochalima.medium.com/turbinando-as-suas-an%C3%A1lises-explorat%C3%B3rias-cbe788993106>.

THUO, Elizabeth. **Sentiment Analysis 101: Technical Foundations - Elizabeth Thuo - Medium**. [S.l.: s.n.], 2024. Acesso em: 29 jun. 2025. Disponível em: <https://medium.com/@elizabeththuo15/sentiment-analysis-101-technical-foundations-53b61fb485f6>.