

The background of the slide features a complex network graph. It consists of numerous nodes, represented by small circles in various colors including blue, green, orange, purple, and teal. These nodes are interconnected by a dense web of thin, dark grey lines, creating a complex, interconnected pattern that fills the upper and middle portions of the slide. The overall aesthetic is technical and data-driven.

# Sistema de Recomendação de Músicas

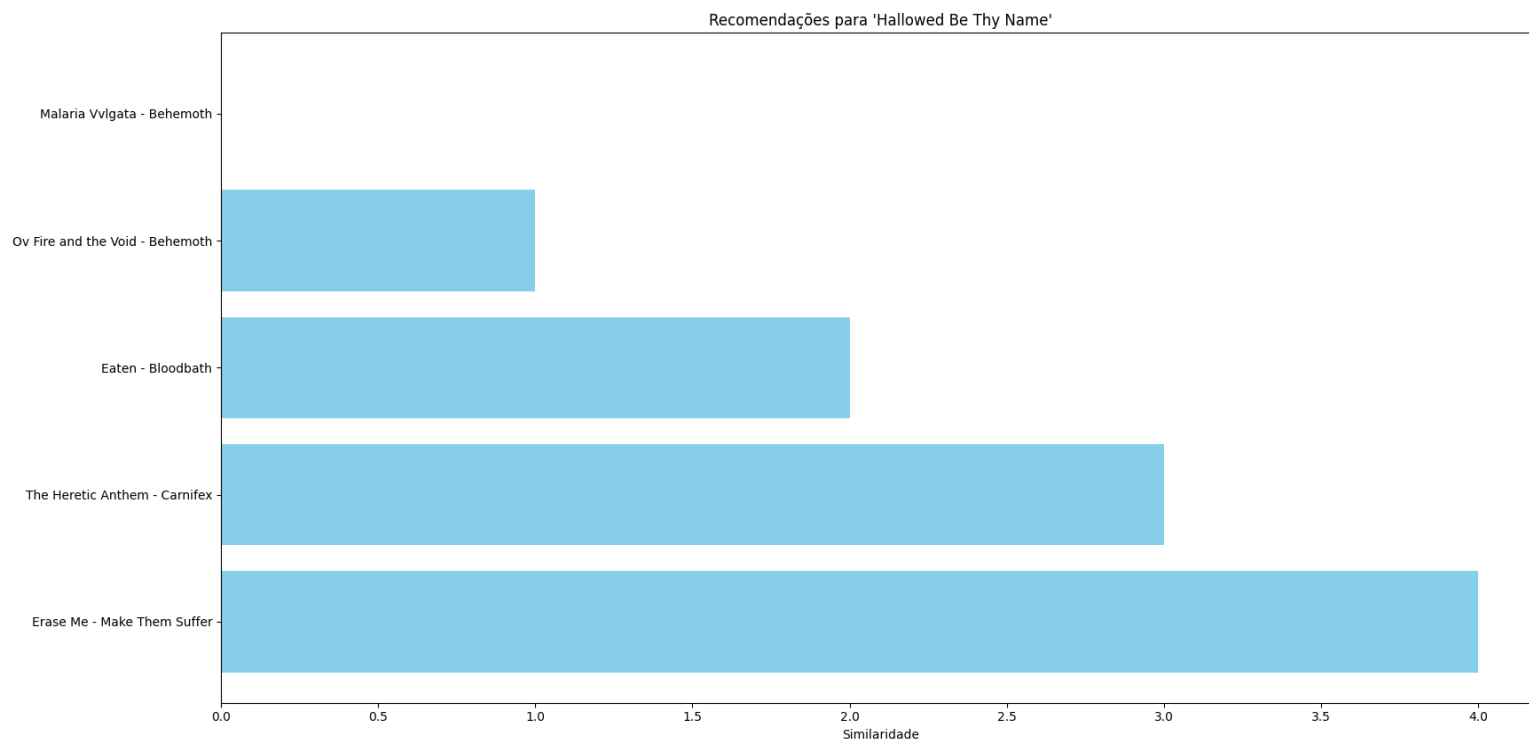
---

Desenvolvimento e Funcionamento

# INTRODUÇÃO

---

- Este projeto desenvolve um sistema de recomendação de músicas baseado em atributos textuais e numéricos.
- O modelo sugere músicas relevantes utilizando machine learning e análise de dados.



# BIBLIOTECAS

---

- Pandas: Manipulação de grandes volumes de dados, essencial para organizar e processar informações musicais.
- Scikit-learn: Modelagem de dados, vetorização de texto e normalização numérica para análise eficiente.
- Matplotlib e Numpy: Visualização gráfica e operações matemáticas avançadas para representar tendências musicais.

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.preprocessing import MinMaxScaler
from sklearn.decomposition import PCA
from sklearn.metrics.pairwise import cosine_similarity
from scipy.sparse import hstack, csr_matrix
import matplotlib.pyplot as plt
```

# PROCESSO DE LIMPEZA

---

- Padronização de Categorias: Uniformização de gêneros musicais e artistas para garantir consistência.

```
music_data_sample.fillna({'track_genre': 'unknown', 'artists': 'unknown'}, inplace=True)
music_data_sample['combined_features'] = (
    music_data_sample['track_genre'] + " " + music_data_sample['artists']
)
```

# PROCESSO DE LIMPEZA

---

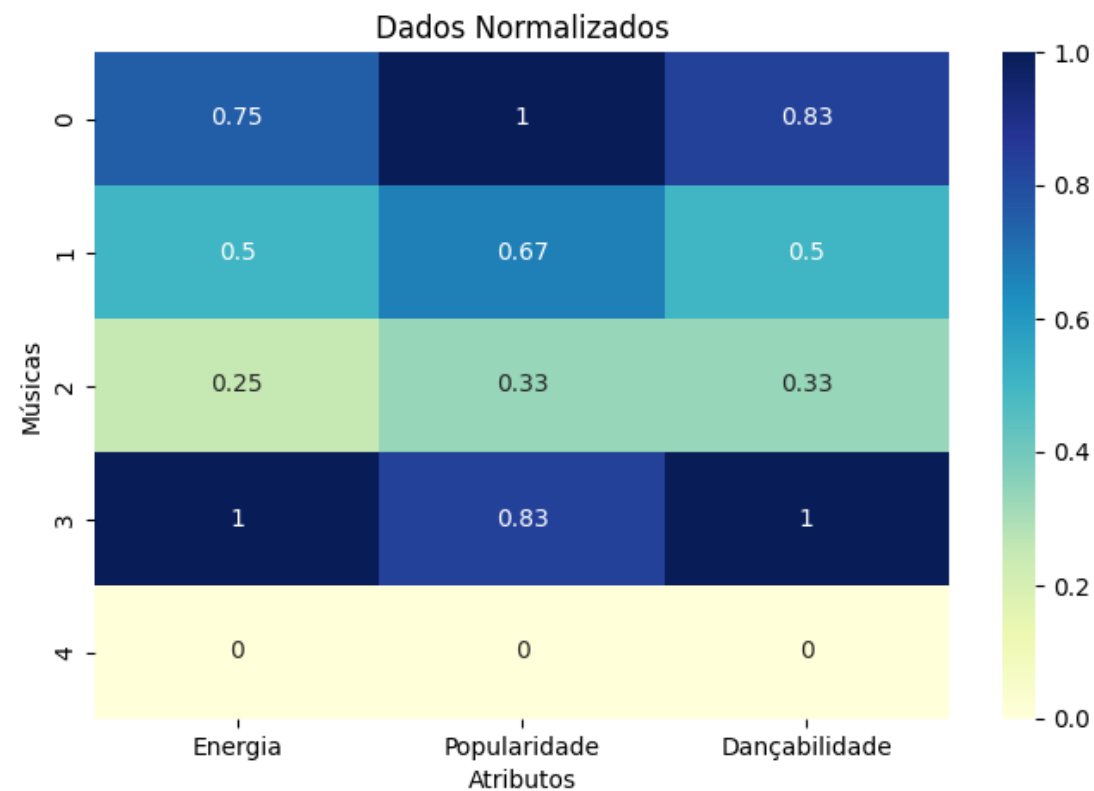
- Tratamento de Valores Ausentes: Substituição de valores numéricos por medianas
- Normalização: Escalonamento de dados numéricos para o intervalo  $[0, 1]$

```
# 4. Normalizar atributos numéricos
numerical_features = ['danceability', 'energy', 'valence', 'tempo', 'popularity']
scaler = MinMaxScaler()
numerical_data = scaler.fit_transform(music_data_sample[numerical_features].fillna(0))
```

# FUNCIONAMENTO DO CÓDIGO

---

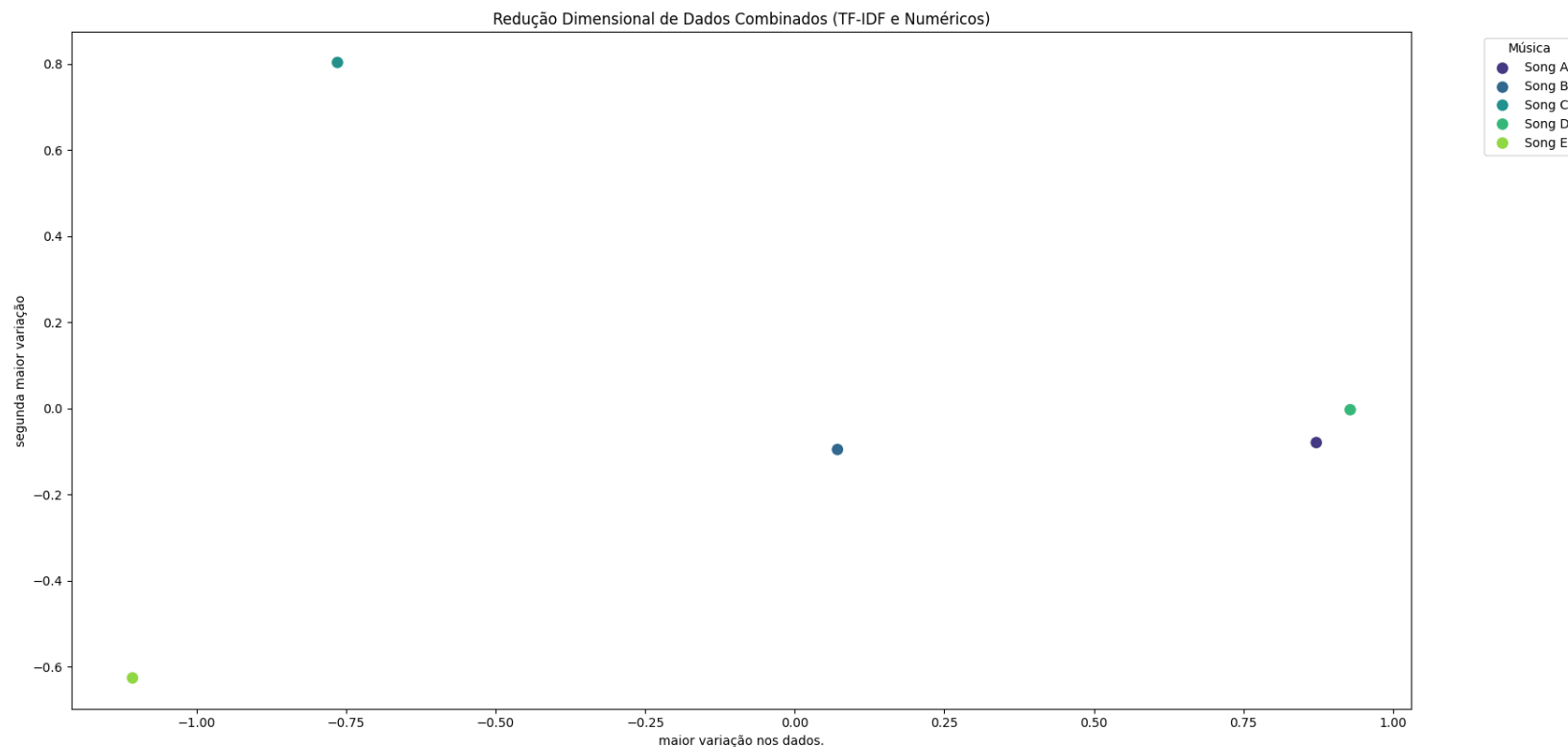
- Combinação de Dados: Transformação de atributos textuais com TF-IDF(Frequency - Inverse Document Frequency) e normalização de atributos numéricos como energia e popularidade.



# FUNCIONAMENTO DO CÓDIGO

---

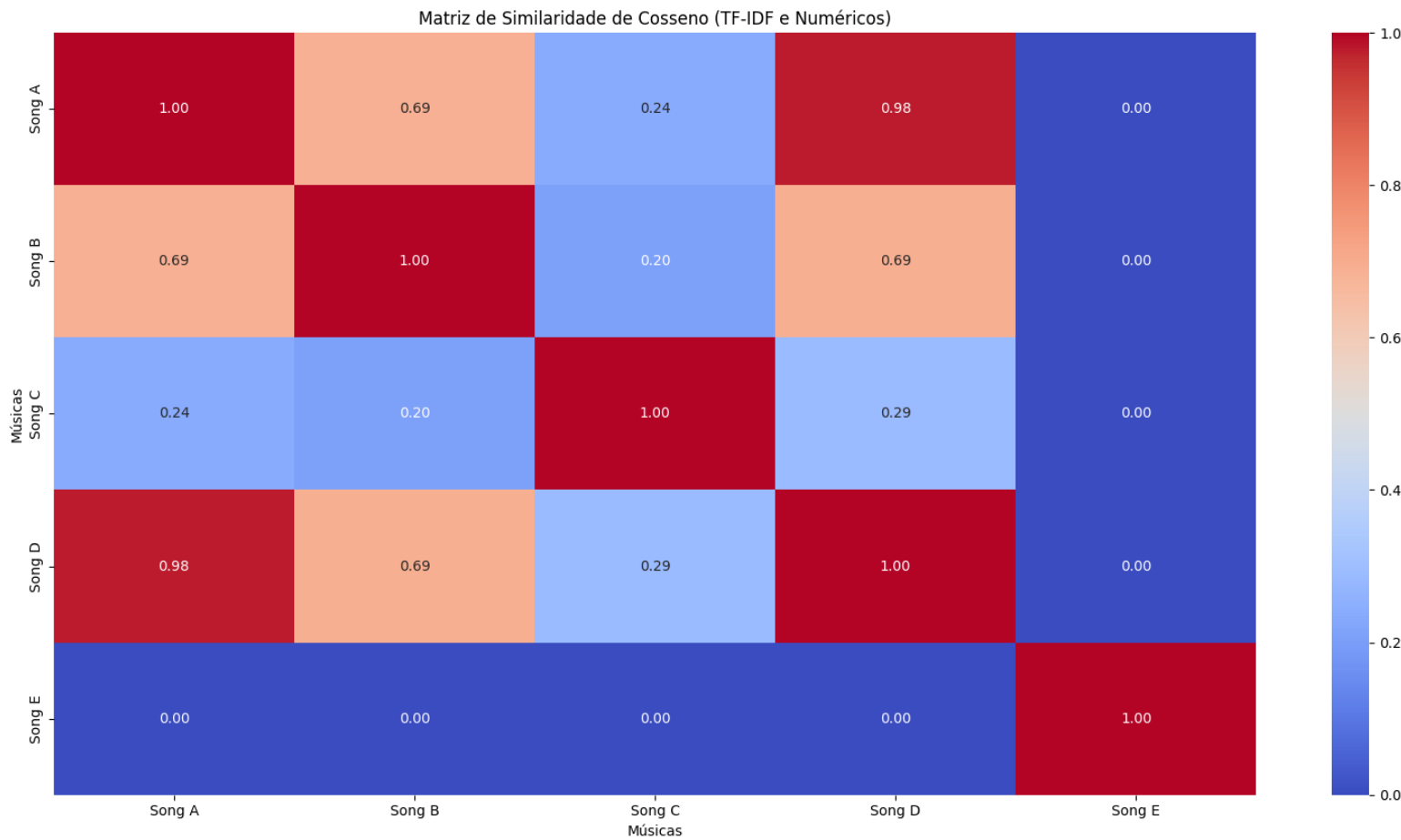
- Redução Dimensional: Utilização de PCA(Principal Component Analysis) para reduzir a complexidade dos dados combinados.



# FUNCIONAMENTO DO CÓDIGO

---

- Cálculo de Similaridade: Uso de similaridade de cosseno para identificar músicas relacionadas.





# RESULTADO

---

