Documentação Técnica

Módulo preprocessing

4.1, 4.2

Escrito por: *Everton Daniel de Lima Romualdo*

1. Introdução

O objetivo deste sistema é gerar playlists musicais com transições suaves entre faixas. Para isso foi utilizade como base o dataset do Kaggles - Spotify Tracks Dataset com aproximadamente 120 mil amostras.

Em resumo, para gerar o grafo, primeiro tratamanos o dataset bruto coletando apenas uma amostra. Em seguida usamos algumas tecnicas para reduzir o custo computacional da criação e manipulação do grafo para viabilizar sua criação e permitir a execução de um algoritmo de recomendação usando os pesos calculados para a aresta.

2. Estratégia de Seleção de Dados (Amostragem)

2.1. Seleção de Gêneros (Estratégia de Continentes)

Para a construção do grafo, optamos por não utilizar a totalidade dos gêneros disponíveis no dataset bruto (mais de 100). A utilização de todos os generos poderia gerar um grafo esparso com clusters isolados

prejudincando a implementação do algoritmo de recomendação.

Foram selecionados 12 gêneros estratégicos que cobrem o espectro sonoro principal e garantem "pontes" naturais de transição:

- **Gêneros Base:** pop, rock, metal, classical, acoustic, piano, dance, brazil.
- **Gêneros de Conexão:** jazz, hip-hop, electronic, reggae.

Essa seleção garante que o grafo possua alta densidade, permitindo caminhos viáveis de extremos opostos (ex: de *Metal* para *Clássica*) passando por nós intermediários.

2.2. Amostragem Balanceada (Sampling)

Foi definido um parâmetro de amostragem (samples_per_genre) fixado por padrão em 800 faixas por gênero por ser um número que mais se aproxima do equilibrio.

O dataset original possui desbalanceamento de classes. Sem essa limitação, gêneros populares dominariam o grafo, enviesando as recomendações. O balanceamento garante que todos os estilos tenham representatividade topológica igualitária na construção das arestas.

---

3. Modelagem Matemática

3.1. Definição do Espaço Vetorial

Cada música é representada como um vetor de características composto por 6 dimensões extraídas dos atributos do dataset:

1. **Danceability:** Adequação rítmica para dança.
2. **Energy:** Intensidade perceptiva e atividade.
3. **Valence:** Positividade musical (humor).
4. **Tempo (BPM):** Velocidade da faixa.
5. **Acousticness:** Confiança de que a faixa é acústica.
6. **Instrumentalness:** Probabilidade de ausência de vocal.

3.2. Normalização de Dados (Min-Max Scaling)

Um passo crítico implementado na classe GraphBuilder é a padronização da escala dos dados. Observamos que o atributo tempo (BPM) varia de 60-200 muito superior em comparação com os outros atributos que variam entre 0-1

Da forma que estava tempo dominaria o cálculo da distância, tornando as outras características irrelevantes. Aplicou-se a técnica de **Min-Max Scaling (**Artigo Mediun**)** para colocar todos os atributos na escala de 0-1

3.3. Métrica de Distância: Euclidiana.

Para calcular a similaridade entre as duas músicas, optamos pela **Distância Euclidiana.** Em transições musicais, a magnitude dos vetores importa. Uma música com Energy = 0.2 é perceptivelmente diferente de uma com Energy = 0.8.

O algoritmo de Dijkstra opera minimizando custos acumulados. A distância Euclidiana fornece um valor não-negativo onde 0 representa identidade e valores maiores representam maior dissimilaridade, funcionando perfeitamente como "peso" da aresta.

---

4. Topologia do Grafo

4.1. Estrutura de Dados

O sistema utiliza um **Grafo Direcionado (DiGraph)** implementado via biblioteca NetworkX.

- **Nós (V):** Representam as músicas (identificadas pelo track_id).
- **Arestas (E):** Representam a transição possível de uma música para outra.
- **Pesos (W):** O valor da Distância Euclidiana calculada.

4.2. Construção de Arestas (K - Vizinhos)

Devido ao alto custo de um grafo completo O(N^2) , implementou-se a estratégia de **K-Vizinhos Mais Próximos (K-NN)**.

Para cada nó de origem, calculam-se as distâncias para todos os outros nós, mas criam-se arestas apenas para os **K** nós com menor distância onde definos k = 50 como padrão do projeto.