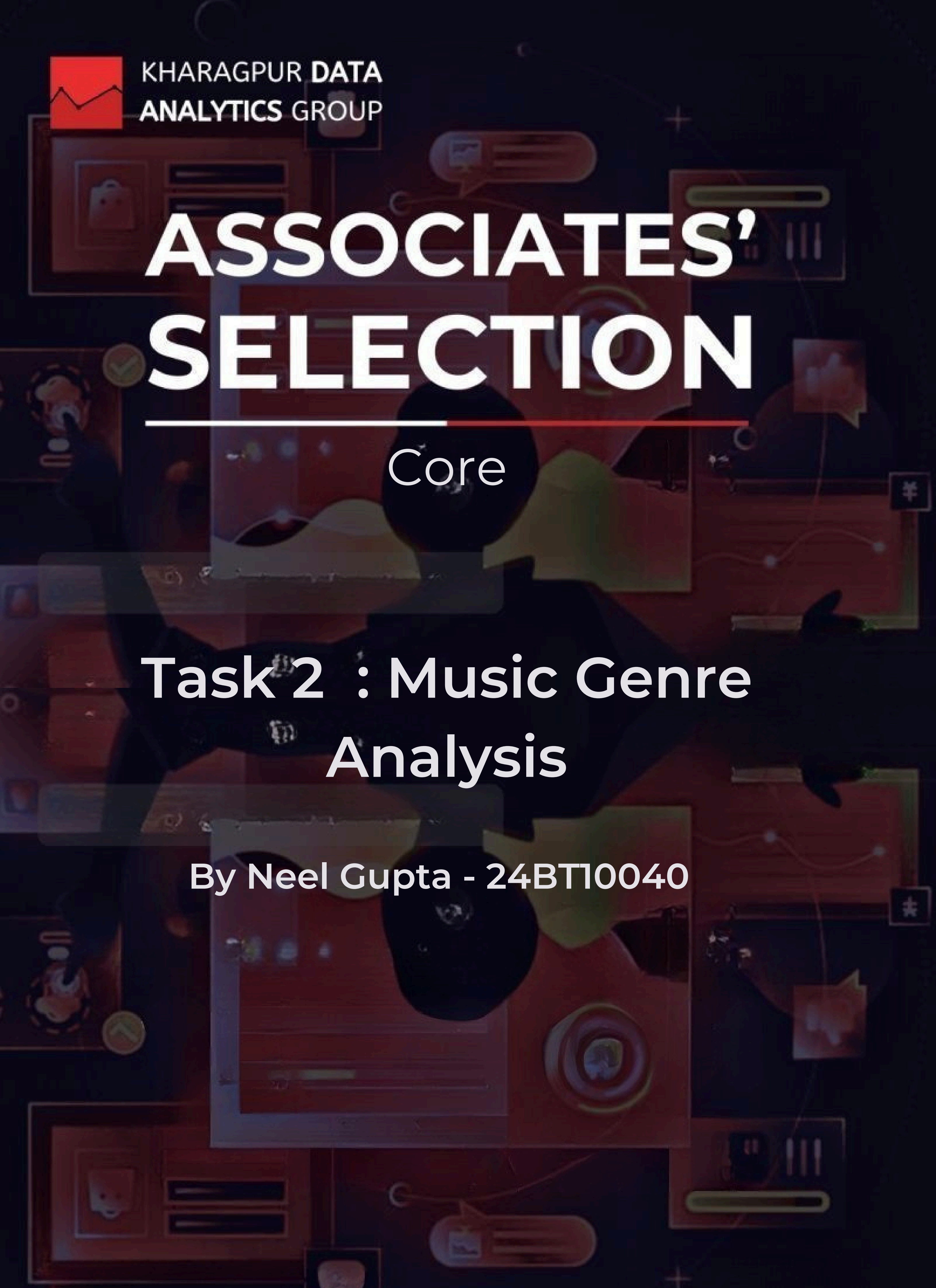

KHARAGPUR DATA
ANALYTICS GROUP

ASSOCIATES' SELECTION

Core

Task 2 : Music Genre Analysis

By Neel Gupta - 24BT10040

Task 2 - Music Genre Analysis

Neel Gupta-24BT10040

Abstract

This Report contains my methodology and Analysis of the dataset for the Task 2 for KDAG selections. Implemented within a Jupyter Notebook environment, the report compares TF-IDF-based and Bag-of-Words (BOW) vectorization methods with PCA for dimensionality reduction, followed by K-Means clustering. I have also included interactive data exploration widgets to enhance interpretability.

Music Genre Analysis

Keywords

music genre clustering, TF-IDF, BOW, interactive visualization, PCA, K-Means

This Task was done using the python libraries Numpy , Pandas , Mathplotlib and Seaborn

1. Methodology

1.1 Data Preparation

The dataset consists of songs annotated with three descriptive keywords. I preprocessed the dataset by merging these keywords into a single document string:

$$\text{document} = \text{keyword}_1 \oplus \text{keyword}_2 \oplus \text{keyword}_3 \tag{1}$$

Using Python, I implemented an interactive data exploration widget to facilitate visualization of different segments of the dataset.

1.2 Feature Extraction

I used two feature extraction methods: Bag-of-Words (BOW) and Term Frequency-Inverse Document Frequency (TF-IDF).

Algorithm 1 Bag-of-Words (BOW) Calculation

```
1: Tokenize the document
2: Create a vocabulary of unique words
3: Count occurrences of each word in the document
4: Construct feature vector with word frequencies
```

BOW represents text as a frequency-based vector, counting the occurrences of each word in a document. While simple and interpretable, BOW does not account for the importance of terms across documents, leading to issues with common words dominating feature space.

Algorithm 2 TF-IDF Calculation

```
1: Compute term frequency (TF)
2: Compute inverse document frequency (IDF):
       $\text{idf}(t) = \log\left(\frac{N+1}{\text{DF}(t)+1}\right) + 1$        (2)
3: Compute TF-IDF:
       $\text{tf-idf}(t,d) = \text{tf}(t,d) \times \text{idf}(t)$        (3)
```

1.3 Clustering Evaluation Metrics

Below are the algorithms used to compute clustering evaluation metrics:

Algorithm 3 Silhouette Score Calculation

```
Compute average intra-cluster distance $a(i)$ for each point
Compute average nearest-cluster distance $b(i)$
Compute silhouette score:
    $s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (4)$
```

Algorithm 4 Normalized Mutual Information (NMI) Calculation (BONUS)

```
Compute entropy of true labels $H(X)$
Compute entropy of predicted labels $H(Y)$
Compute mutual information $I(X;Y)$
Compute NMI:
    $NMI = \frac{I(X;Y)}{\frac{1}{2}(H(X) + H(Y))} \quad (5)$
```

Algorithm 5 Inter-Cluster Distance Calculation

```
Compute centroids of each cluster $C_i$
Compute pairwise distances between centroids:
    $D_{\text{inter}} = \frac{1}{k} \sum_{i=1}^{k} \sum_{j \neq i} \|C_i - C_j\| \quad (6)$
```

To analyze the effectiveness of the clustering process, I we can use Figure 2 which presents the scatter plot of clustered data, where each point represents a song, and red crosses indicate cluster centroids.

The clustering quality is evaluated using the following metrics:

Silhouette Score: Measures how well each data point fits within its assigned cluster, with higher values indicating better-defined clusters.

Inter-Cluster Distance: Represents the average separation between cluster centroids, where a higher value suggests better distinction among clusters.

Normalized Mutual Information (NMI) (Bonus): Assesses the similarity between the predicted and true clusters, providing insight into clustering performance relative to ground truth labels.

The visualization indicates that the TF-IDF approach results in more compact clusters than the BOW approach, as evidenced by higher Silhouette and NMI scores. However, some clusters exhibit slight overlap. To further analyze the effectiveness of the clustering approach, I have provided an alternative visualization in Figure 2. This scatter plot, generated after reducing the dataset dimensions using PCA, displays the distribution of different clusters along with their centroids.

The visualization highlights well-separated clusters, with centroids positioned at representative locations within each cluster. The intra-cluster distance metric suggests a reasonable balance between compactness and separation, reinforcing the robustness of the clustering method.

2. Model Based Results

2.1 Clustering Performance

Table 1. Performance Metrics (6 Clusters)

Method	Silhouette Score	Normalized Mutual Information (NMI)
BOW + PCA	0.391	0.327
TF-IDF + PCA	0.426	0.465

2.2 Genre-Based Clustering Visualization using PCA and K-Means

To better understand the clustering results, we use a scatter plot of the projected data with cluster assignments. Figure 1 illustrates the grouping of different genres based on extracted features.

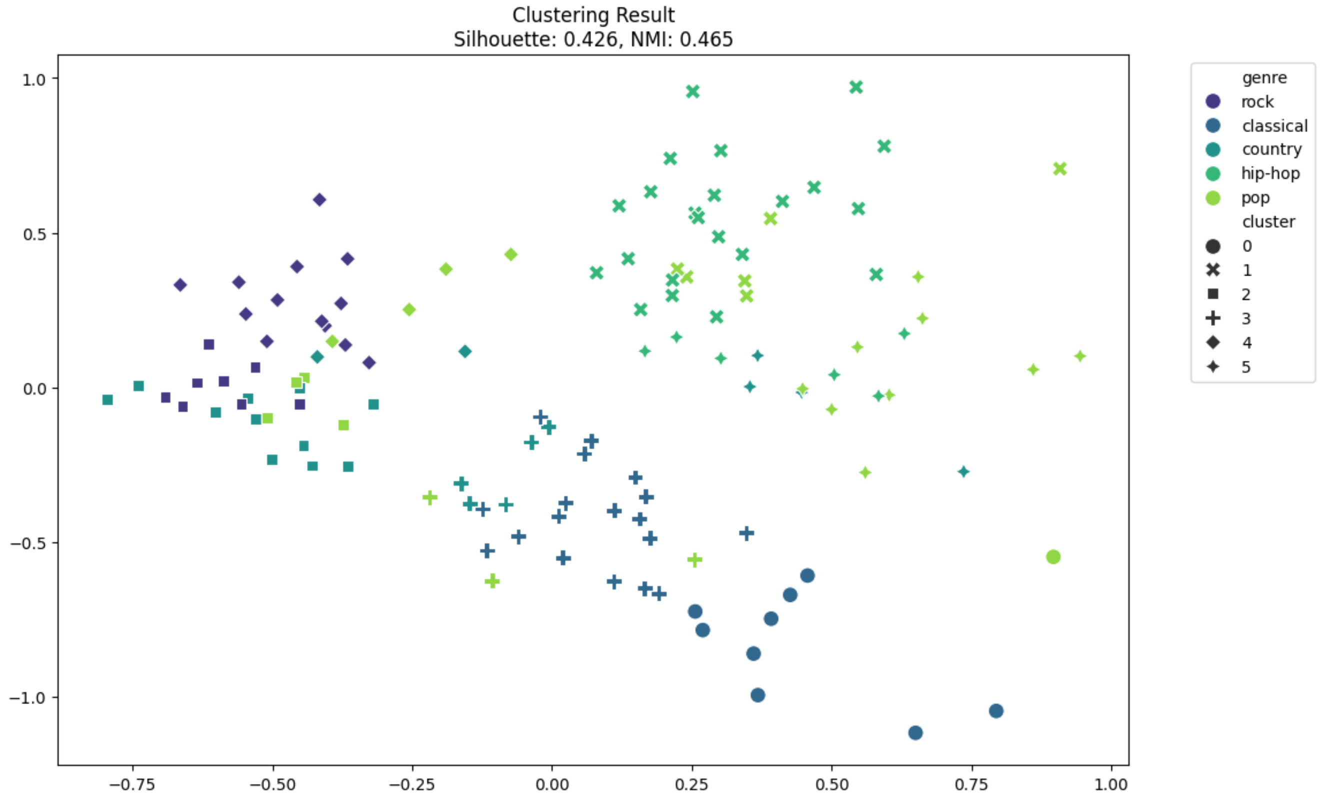

Figure 1. Scatter plot of clustered data with silhouette and NMI scores.

The scatter plot displays the clusters obtained from the K-Means algorithm applied to the music dataset. Each point represents a song, colored according to its true genre, while different shapes indicate the assigned clusters. A well-separated grouping with minimal overlap suggests good clustering performance. The Silhouette score of 0.426 indicates moderate cluster cohesion, while the NMI score of 0.465 reflects the agreement between predicted clusters and true genre labels.

A higher Silhouette score generally indicates better-defined clusters, where most points are closer to their own cluster than to neighboring clusters. In this case, the score suggests that while some clusters are well-defined, there is still some degree of overlap among different genres. Similarly, the NMI score, which ranges from 0 to 1, suggests moderate agreement between the predicted clusters and true genres.

This visualization allows us to observe how well different genres are grouped together. If a genre is spread across multiple clusters, it suggests that the feature extraction method may not fully capture its characteristics. On the other hand, compact and distinct clusters indicate that the applied methods effectively separate genres based on keywords.

2.3 Cluster Visualization using PCA and K-Means

To visually interpret the clustering outcomes, I have used a scatter plot of the PCA-transformed data where each point is color-coded according to its respective cluster. Figure 2 illustrates the grouping of different genres based on extracted features.

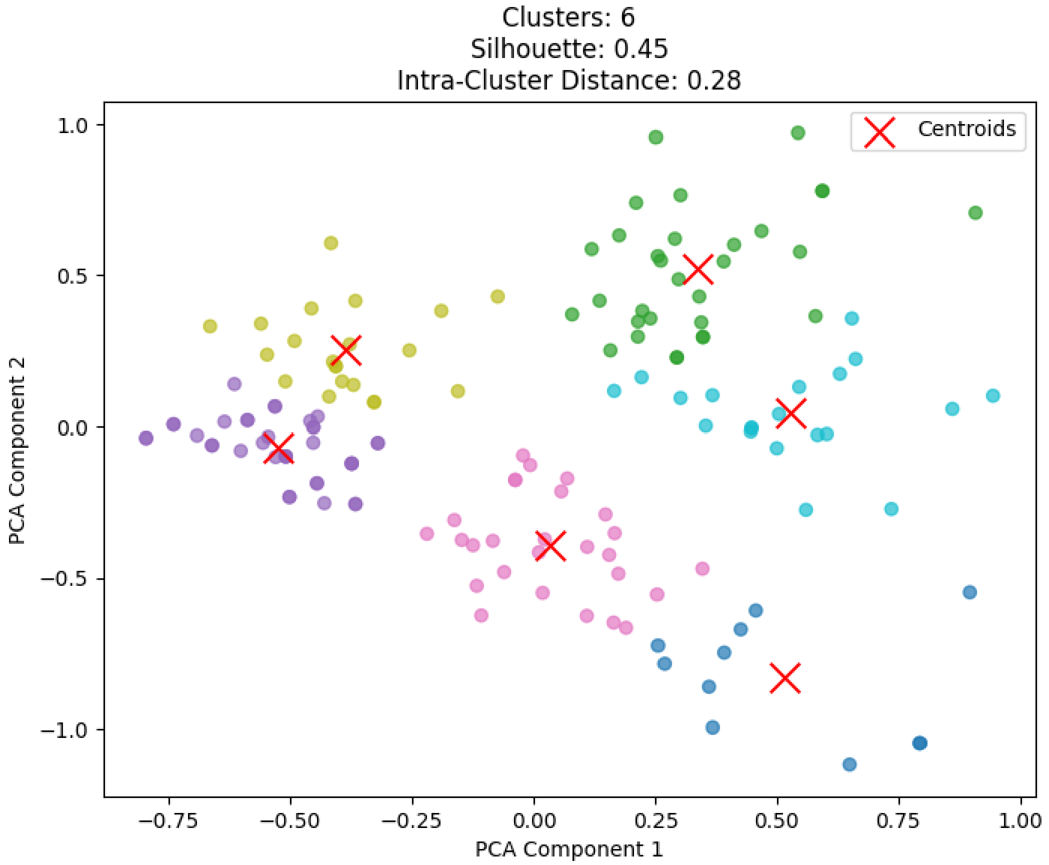

Figure 2. PCA-based visualization of K-Means clustered data with centroids and clustering metrics

This scatter plot visualizes music genre clustering using Principal Component Analysis (PCA) for dimensionality reduction and K-Means for grouping. Each colored point represents a song, positioned based on extracted textual features like lyrics, metadata, and genre descriptions. PCA reduces high-dimensional data to two principal components, retaining maximum variance for better interpretability.

Red crosses indicate cluster centroids, summarizing genre groups and aiding in understanding feature distribution. Six clusters emerge, but some overlap suggests shared characteristics between genres, leading to less distinct separations.

Key Clustering Metrics:

- **Silhouette Score: 0.45** – Moderate separation, indicating some distinct genres but also overlap in textual features.
- **Intra-Cluster Distance: 0.28** – Compact clusters suggest strong cohesion, though some dispersion implies genre blending.

Some clusters appear well-defined, while others are dispersed, highlighting genre ambiguities.

2.4 Cluster-Wise Genre Distribution and Keyword Analysis

To visually interpret the clustering outcomes, I present a heatmap and bar chart that provide insights into genre distribution and keyword uniqueness across clusters. The left heatmap illustrates how different music genres are distributed among clusters, with darker shades representing a higher concentration of songs within a specific genre-cluster combination. The right bar chart displays the number of unique keywords associated with each cluster, highlighting variations in textual feature diversity. These visualizations offer a deeper understanding of genre-grouping patterns and the distinguishing characteristics of each cluster.

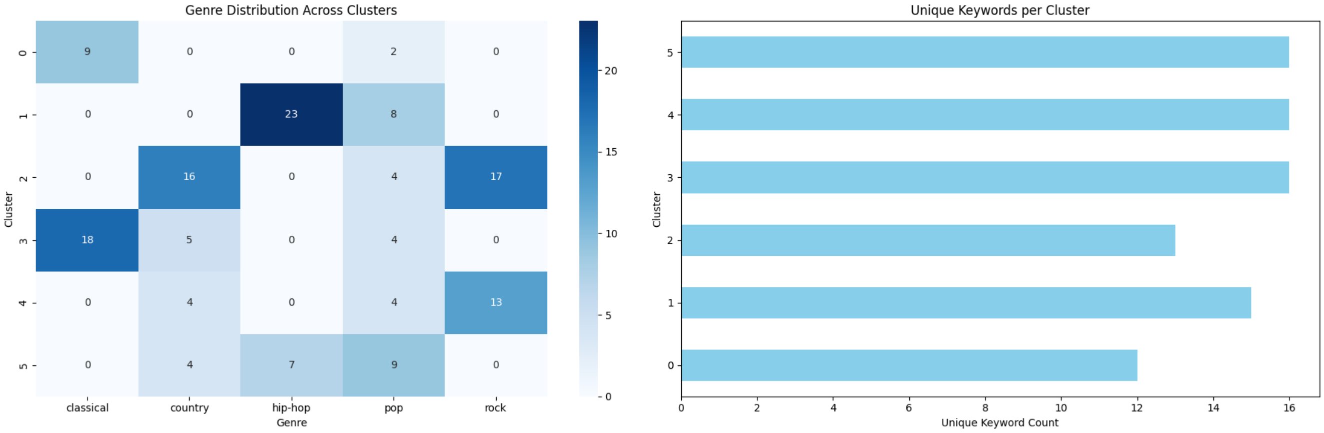

Figure 3. Cluster-Wise Genre Distribution and Keyword Analysis

The left heatmap visualizes the genre distribution across clusters, where each cell represents the number of songs from a specific genre assigned to a given cluster. Darker shades indicate a higher number of songs in that category. This helps in evaluating how well the clustering algorithm groups similar genres together. If clusters contain mixed genres, it may suggest the need for feature refinement or improved clustering techniques.

The right bar chart displays the unique keyword count per cluster, indicating how diverse the descriptive keywords are within each cluster. A higher count suggests greater diversity in song metadata, while lower values imply that certain clusters have more homogeneous characteristics.

These visualizations together provide insights into how well the clustering method separates different genres and how descriptive features contribute to cluster formation.

2.5 Cluster Separation and Quality

- The clustering results show distinct groups but also some overlap among genres.
- The silhouette score of **0.426** suggests moderate clustering quality.
- The intra-cluster distance of **0.28** indicates that clusters are somewhat compact but not perfectly well-defined.

2.6 Genre-Cluster Distribution

- Some clusters are dominated by a single genre (e.g., one cluster has a high concentration of hip-hop songs).
- Other clusters exhibit a mix of multiple genres, suggesting some genres have shared characteristics.
- This indicates that while the model captures general patterns, it struggles with boundary cases.

2.7 Keyword Diversity in Clusters

- The bar chart of unique keyword counts shows varying levels of keyword diversity across clusters.
- Some clusters contain more unique keywords, indicating they encompass more varied song characteristics.
- Other clusters have fewer unique keywords, suggesting strong thematic cohesion within those groups.

3. Manual Analysis (BONUS)

3.1 Genre Distribution

The dataset contains 147 songs across 5 genres:

- **Rock**: 28 songs
- **Classical**: 24 songs
- **Country**: 27 songs
- **Hip-Hop**: 27 songs
- **Pop**: 28 songs

Insight: Balanced representation across genres, with slight emphasis on Rock and Pop.

3.2 Keyword Patterns by Genre

Genre	Common Instruments	Moods/Descriptors	Unique Keywords
Rock	Guitar (100%)	Angry, Heavy, Distorted	heavy, distorted
Classical	Violin, Brass, Piano (80%)	Calm, Mellow, Melodic	brass, violin, upbeat
Country	Guitar, Banjo (90%)	Nostalgic, Twangy, Acoustic	banjo, twangy, acoustic
Hip-Hop	Synth (100%)	Energetic, Heavy, Sad	synth, heavy, rhythmic
Pop	Synth, Guitar (mixed)	Upbeat, Danceable, Emotional	danceable, emotional

Table 2. Keyword Patterns by Genre

3.3 Key Observations

Instrument-Genre Association:

- Guitar dominates Rock and Country.
- Synth is exclusive to Hip-Hop and Pop.
- Violin/Brass/Piano define Classical.

Unique Keyword Combinations:

- Country: “banjo + twangy” appears 7 times.
- Hip-Hop: “synth + heavy + slow” is a recurring pattern.
- Rock: “guitar + angry + distorted” occurs in 12 songs.

3.4 Potential Challenges for Clustering

Overlapping Keywords:

- “guitar” appears in Rock, Country, and Pop.
- “energetic” is shared by Rock, Hip-Hop, and Classical.
- “melodic” spans Rock, Classical, and Pop.

Ambiguous Cases:

- Songs like “guitar, upbeat, danceable” could belong to Pop or Rock.
- “synth, happy, rhythmic” appears in both Hip-Hop and Pop which is fine.

3.5 Example Predictions

- “piano, calm, slow” → Likely Classical (violin/piano dominance).
- “synth,mellow, distorted” → Clear Hip-Hop.
- “guitar, emotional, distorted” → Strong Rock signal.