KHARAGPUR **DATA ANALYTICS** GROUP

TASK TWO:

Music Genre Analysis

WHITE PAPER REPORT

BY – SHAURYA SHREYAS

24IM10023

KHARAGPUR **DATA**
ANALYTICS GROUP

Table of Contents

KHARAGPUR **DATA**
ANALYTICS GROUP

1. Introduction

The project aims to implement unsupervised learning techniques to analyze musical patterns through keyword clustering. The task involves processing song keywords using Bag-of-Words vectorization and reducing dimensionality with Principal Component Analysis (PCA). K-means clustering is then applied to discover natural groupings in the reduced feature space, with cluster quality evaluated using silhouette scores.

This report analyzes music data using unsupervised learning to group similar songs based on descriptive keywords. Using Bag-of-Words and TF-IDF techniques, we converted keywords into numerical vectors. After visualizing token frequencies, Principal Component Analysis was applied for dimensionality reduction. K-Means clustering with optimal cluster count determined by the Elbow Method revealed distinct song groupings. The results demonstrate effective pattern identification in musical characteristics, establishing groundwork for future music classification and recommendation systems.

KHARAGPUR **DATA**
ANALYTICS GROUP

2. Technical Implementation: What and How

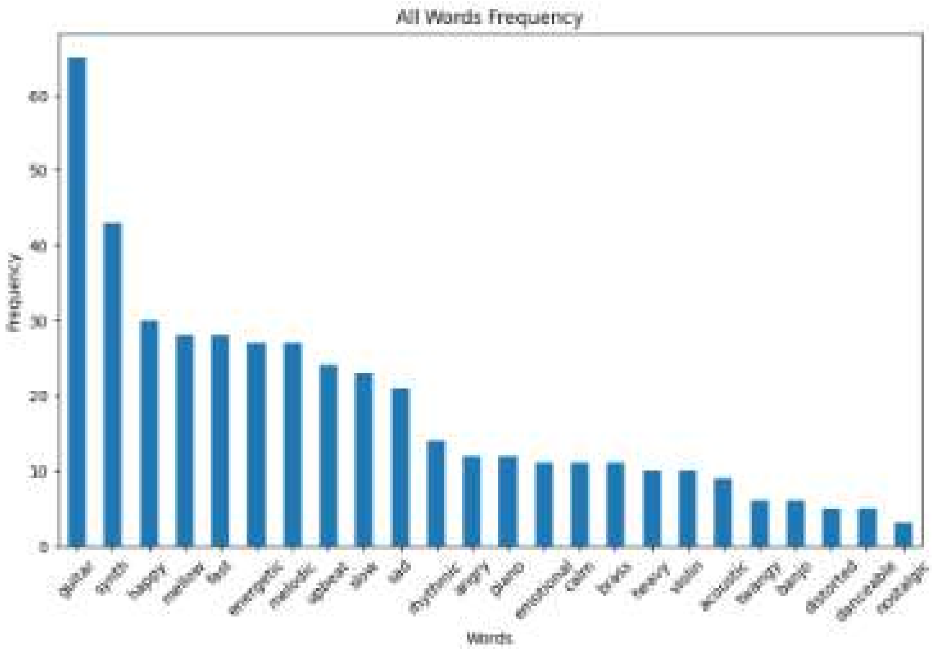
2.1 Data Preprocessing

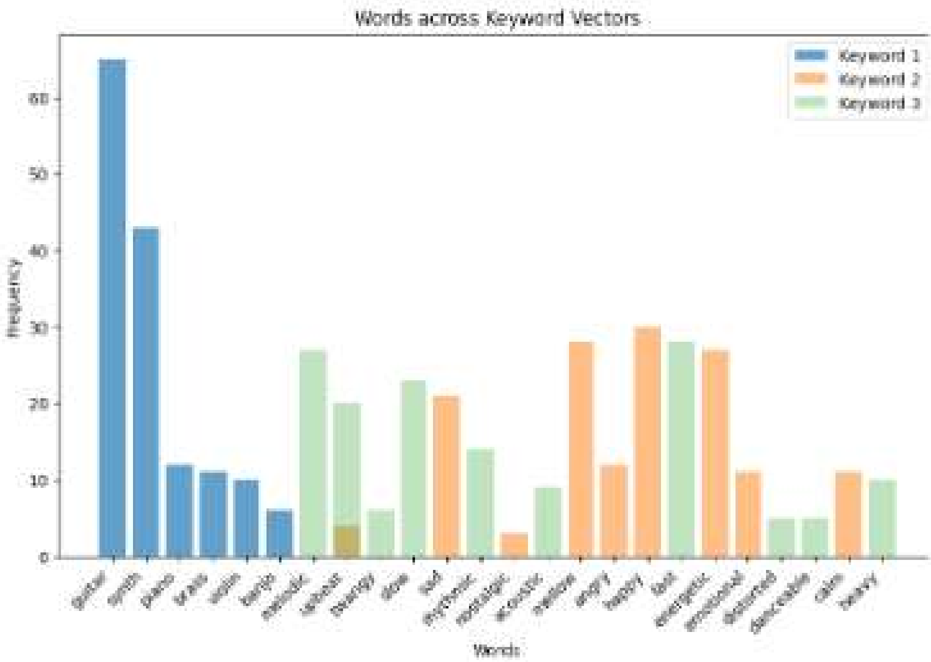
The data preprocessing begins by tokenizing text data and creating a vocabulary from unique words. The code implements BOW vectorization by creating a sparse matrix where each row represents a song and columns represent vocabulary words, containing counts of word appearances in keywords. A helper function create_bow_vector() converts individual text inputs into BOW vectors

The alternative TF-IDF implementation includes Term Frequency calculation for word frequency within documents, Inverse Document Frequency for word uniqueness across documents, and their product for balanced metrics. BOW was chosen over TF-IDF due to better silhouette scores, simpler implementation, numerical stability, and the fact that the dataset contains pre-curated keywords rather than natural text.
Finally, vectors for all three keywords are combined using np.maximum(). I am using this method as all the keywords are independent to each other.

2.2 Token Frequency Visualization

Here are the plots to analyze word frequencies in the dataset. The first visualization creates a bar chart displaying the overall frequency of all words in the corpus using the Bag-of-Words matrix, with words on the x-axis and their frequencies on the y-axis. The second visualization provides a more detailed analysis by creating a stacked bar chart that compares the frequency distribution of words across all three keyword positions (Keyword 1, 2, and 3). This is achieved by summing the vectors for each keyword position separately and using different transparency levels (alpha values) to distinguish between them. Both plots use matplotlib with customized figure sizes, rotated x-axis labels for better readability, and proper labeling of axes and titles. The word 'upbeat' appears in both Keyword 1 and Keyword 2 positions, indicating its versatility in describing different aspects of songs.

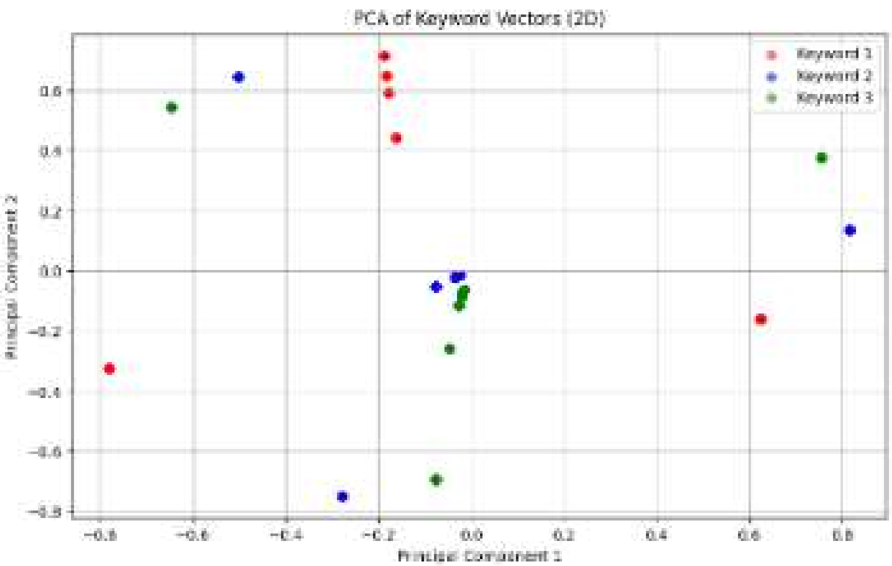

All Words Frequency
Frequency
Words
0
10
20
30
40
50
60
guitar
synth
happy
mellow
fast
energetic
melodic
upbeat
slow
sad
rhythmic
angry
piano
emotional
calm
brass
heavy
violin
acoustic
twangy
banjo
distorted
danceable
nostalgic


Words across Keyword Vectors
Keyword 1
Keyword 2
Keyword 3
Frequency
Words
0
10
20
30
40
50
60
guitar
synth
piano
brass
violin
banjo
melodic
upbeat
twangy
slow
sad
rhythmic
nostalgic
acoustic
mellow
angry
happy
fast
energetic
emotional
distorted
danceable
calm
heavy

2.3 Dimensionality Reduction (PCA)

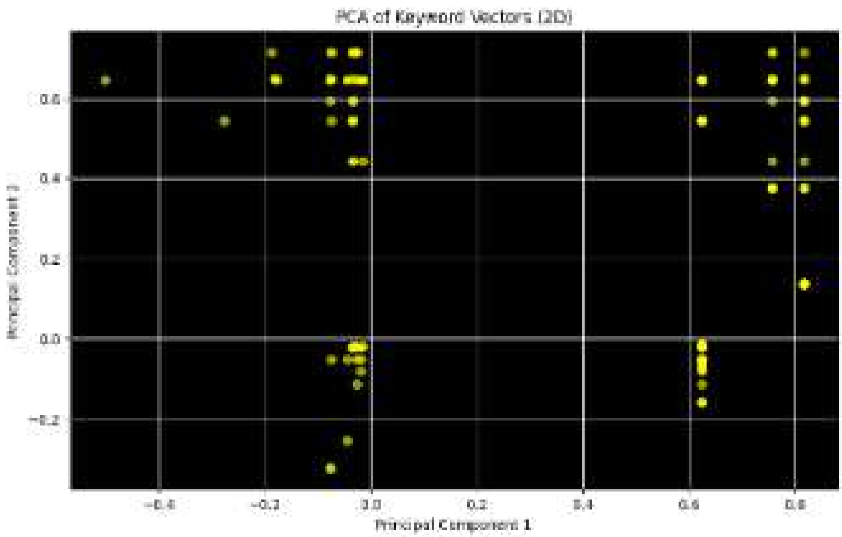
The PCA implementation performs dimensionality reduction through several key steps. First, it centers the data by subtracting the mean from each feature. Then, it computes the covariance matrix to measure relationships between features. Using eigendecomposition, it calculates and sorts eigenvalues and eigenvectors in descending order to identify principal components. Finally, it projects the data onto the selected eigenvectors to reduce dimensionality to 2D. The code applies this PCA function to reduce the three keyword vectors (keyword_1_vectors, keyword_2_vectors, keyword_3_vectors) from their high-dimensional space to 2D representations for visualization and further analysis.

Here's the plot visualizing the distribution of keywords in 2D space after PCA dimensionality reduction. The scatter plot displays three distinct keyword groups using different colors: Keyword 1 in red, Keyword 2 in blue, and Keyword 3 in green, each with 50% transparency (alpha=0.5). The x and y axes represent the first and second principal components respectively, showing how the keywords are distributed in the reduced dimensional space. A grid is added for better readability, and a legend helps identify each keyword group. The figure size is set to 10x6 for optimal visualization of the data points.



2.4 Combining the embedding

A code combines the three keyword embeddings into a single representation using two methods. The final combination method uses the element-wise maximum value across all three vectors using np.maximum(), which preserves the strongest features from each keyword position. The visualization then displays these combined embeddings in a 2D scatter plot with a black background and white grid for contrast. Each point, represented in yellow with 60% opacity (alpha=0.6), shows the final position of a song in the reduced dimensional space, where the x and y axes represent the first and second principal components respectively. This visualization helps in understanding how songs are distributed in the feature space after combining information from all three keywords.



2.5 Clustering Implementation (K-Means)

The initialize_centroids function starts the process by randomly selecting k data points as initial cluster centers. The assign_clusters function calculates the Euclidean distance between each data point and all centroids, then assigns each point to its nearest centroid. The update_centroids function recalculates the cluster centers by computing the mean of all points assigned to each cluster. Finally, the kmeans function combines these steps iteratively until convergence or until reaching the maximum iterations limit.

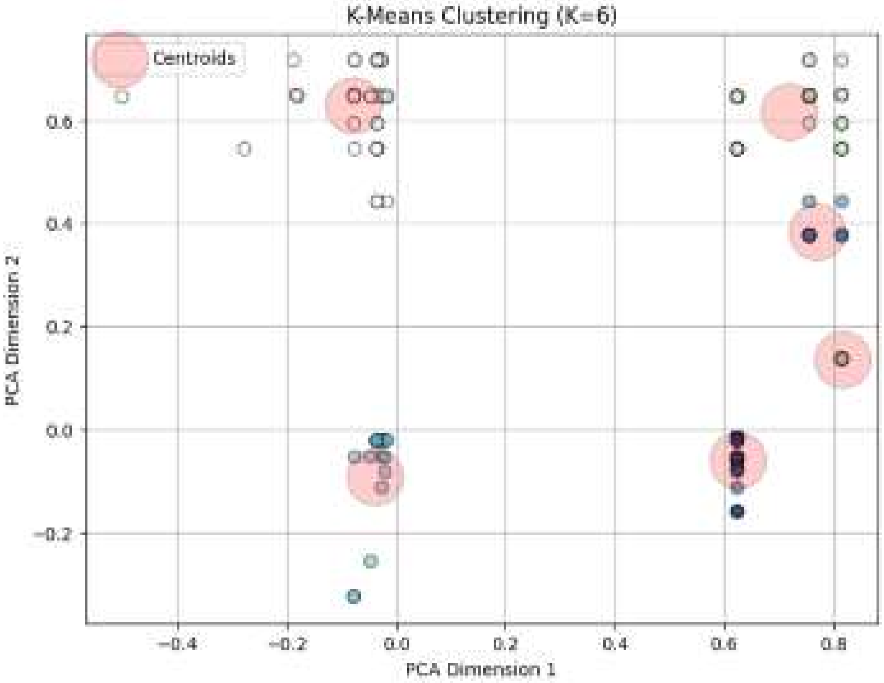
The algorithm continues updating cluster assignments and centroid positions until either the centroids stop moving (indicating convergence) or the maximum iteration count is reached. This implementation ensures stable clustering of the PCA-reduced song keyword data into k distinct groups.

The optimal K value for clustering is determined through an analytical process known as the Elbow Method. This method works by calculating the Within-Cluster Sum of Squares (WCSS) for different values of K (from 1 to 10). WCSS measures how similar points are within each cluster by summing the squared distances between points and their assigned cluster centroids.

As K increases, WCSS naturally decreases since points are closer to their centroids in smaller clusters. However, the rate of decrease typically slows down at a certain point, creating an "elbow" in the curve. Through this analysis, the optimal value of K was determined to be 6, representing the best balance between cluster granularity and model complexity. This point indicates where adding more clusters doesn't significantly improve the clustering quality.

KHARAGPUR **DATA ANALYTICS** GROUP

The visualization represents the clustering results where each point in the 2D space represents a song, colored based on its assigned cluster. The six large red circles mark the cluster centroids, showing the center points around which similar songs are grouped. This plot effectively displays how the songs are naturally grouped based on their keyword similarities after dimensionality reduction through PCA



3. RESULTS AND ANALYSIS

A genre prediction function was implemented that processes three input keywords, converts them to vectors, applies PCA transformation, combines them using maximum values, and predicts genre based on nearest cluster centroid.

The K-means clustering algorithm achieved a moderate accuracy of 40.14% in grouping similar songs based on their keyword features.

The clustering implementation achieved a high silhouette score of 0.7655, indicating well-defined and distinctly separated clusters with strong cohesion within groups and good separation between clusters.

The low accuracy of 40.14% is due to overlapping keyword patterns across genres, subjective musical descriptions, and the inherent limitations of K-means clustering with high-dimensional musical data.

KHARAGPUR **DATA**
ANALYTICS GROUP

4. Discussion

The implementation of unsupervised learning techniques for music classification revealed several interesting insights and challenges. The project successfully demonstrated the potential of clustering algorithms in identifying patterns within musical descriptions, while also highlighting limitations in genre classification accuracy.

The data preprocessing phase employed both Bag-of-Words (BOW) and TF-IDF vectorization techniques, with BOW ultimately proving more effective for our specific use case. This was evidenced by better silhouette scores and more stable clustering results. The visualization of word frequencies across different keyword positions provided valuable insights into how musical descriptions overlap across genres, with words like 'upbeat' appearing prominently in multiple positions.

Dimensionality reduction through PCA proved crucial in making the data more manageable while preserving essential patterns. The reduction to two dimensions enabled effective visualization of the data distribution and cluster formations. The elbow method determined an optimal cluster count of 6, suggesting natural groupings within the musical descriptions.

Despite achieving a strong silhouette score of 0.7655, indicating well-defined and separated clusters, the overall clustering accuracy of 40.14% reveals the inherent challenges in music genre classification. This relatively low accuracy can be attributed to the subjective nature of musical descriptions, overlapping characteristics between genres, and the limitations of unsupervised learning in capturing genre-specific nuances.

The project demonstrates that while unsupervised learning can effectively identify patterns in musical descriptions, the complex and subjective nature of music genre classification presents significant challenges. Future work could explore hybrid approaches combining supervised and unsupervised techniques, or incorporate additional musical features to improve classification accuracy.

KHARAGPUR **DATA**
ANALYTICS GROUP

5. Acknowledgement

I would like to express my sincere gratitude to KDAG Associates for providing this opportunity to work on an engaging machine learning project. The task was particularly enriching as it challenged me to implement algorithms from scratch without relying on inbuilt library functions, enhancing my understanding of core concepts.