Melodify: Music Similarity through Embedding and Metadata

Gautham Satyanarayana, Yukta Salvi, Saumya Chaudhury, Pradeep Raj Prabhu Raj, Parikha Goyanka

1 Introduction

Identifying similar songs is a complex problem as songs can vary along many dimensions that are not always predefined or easily quantifiable. Traditional music recommender systems typically rely on user listening history or explicit feedback using algorithms like collaborative filtering to suggest songs. However, such methods often overlook the intrinsic characteristics of the music itself, instead focusing primarily on user behaviour patterns. This limitation is particularly evident in cold start scenarios where a lack of user history makes recommendations unreliable. While some systems incorporate basic audio features such as tempo, key, and rhythm, these approaches are often too simplistic to capture a song's full semantic and artistic value. Music is inherently multifaceted, involving a combination of lyrical meaning, vocal delivery, instrumentation, and other auditory elements that contribute to its emotional and aesthetic impact on listeners. As a result, a more holistic method is required to effectively understand and compute song similarity in a meaningful way.

2 Relevant Work

In this section, we will explore the current state-of-the-art models in this domain, the methods that Spotify uses, how they work, what are their pitfalls and advantages, and how are they fair/unfair to unpopular artists and tracks.

2.1 Modern Recommender Systems

Modern recommender systems, such as those employed by Spotify, leverage advanced machine-learning techniques to provide personalized content suggestions. These systems integrate methods like collaborative filtering, content-based filtering, and deep learning models, often augmented by contextual and behavioural data to improve user experiences. Collaborative filtering identifies patterns based on user interactions; for instance, user-based collaborative filtering recommends items liked by users with similar preferences, while item-based approaches suggest items similar to those already enjoyed. Content-based filtering complements this by analyzing the metadata and features of items, such as genre or tempo in the case of songs, to propose similar content. Modern systems frequently combine these approaches into hybrid models to harness their complementary strengths.

2.2 Evaluation Methods

Evaluating recommender systems involves both offline and online metrics. Offline evaluations, using historical data, employ metrics like precision, recall, and normalised discounted cumulative gain (NDCG) to assess accuracy and ranking quality. Online assessments, such as click-through rate (CTR) and engagement metrics, measure real-time user interactions with recommendations. A/B testing is also widely used to compare different recommendation strategies in a controlled environment. These metrics collectively ensure that recommendations not only align with user preferences but also drive meaningful engagement.

2.3 Advantages & Disadvantages

The advantages of modern recommender systems are significant. They enable a high degree of personalisation, tailoring suggestions to individual user preferences, thereby increasing satisfaction and engagement. Scalability is another strength, allowing these systems to serve millions of users effectively. Furthermore, by encouraging the exploration of new content, they help users discover items they might not have encountered otherwise. Context-aware recommendations further enhance user experience by adapting to factors like time of day or mood.

However, these systems face notable challenges. The cold start problem, for instance, limits its effectiveness for new users or items due to a lack of data. Over-personalization can also lead to filter bubbles, restricting users' exposure to diverse content. Additionally, recommender systems are heavily data-dependent, and their performance is contingent on the availability of high-quality data. Bias in algorithms or data can skew recommendations, raising fairness concerns. Moreover, privacy issues arise from the extensive data collection required for personalisation, presenting ethical and legal challenges.

3 Methodology and Models

The models we trained and compared can be categorised into 2 - Content-based and metadata-based embedding models. The content-based models would embed the inherent audio features of the songs and the metadata would embed the song's metadata such as tags, artists, and albums.

3.1 Content-Based Embedding Models

3.1.1 Audio and Tag Embedding

The audio data often varies in size. To standardize this data for machine learning models, the following process is applied: Numerical columns in the dataset are padded to ensure uniform dimensions. This step ensures that all samples have the same shape, allowing for batch processing and compatibility with the CNN

architecture. After padding, the data is represented as an N×D matrix, where N represents the number of audio samples or segments, and *D* represents the feature dimensions.

The N×D matrix is passed through a Convolutional Neural Network (CNN), which learns hierarchical features from the audio data. This allows the model to capture intricate patterns and relationships within the audio input. The output of the CNN is a fixed-size embedding vector of 50 dimensions. This embedding serves as a compact representation of the audio data, encapsulating the essential information needed for downstream tasks such as classification, clustering, or retrieval.

Tags, which are textual labels or annotations (e.g., "rock," "pop") from the Last.fm Dataset, are processed to capture their semantic relationships using the Word2Vec technique. The tags are passed through the Word2Vec model, which generates continuous-valued vector embeddings for each tag. Word2Vec uses a neural network-based approach to map tags into a high-dimensional space where the semantic relationships between words are preserved. The embeddings capture both syntactic and semantic nuances of the tags, making them valuable for tasks like similarity computation, clustering, and recommendation. By providing a semantic context for the audio data, these embeddings enhance the system's ability to link tags with corresponding audio embeddings, enabling a more comprehensive analysis.

3.2 Metadata-Based Embedding Models

3.2.1 Graph Embedding

A heterogeneous graph is constructed using data from the Last.fm Dataset, with entities such as artists, albums, tracks, and tags represented as nodes. The connections between these nodes are defined by edges that capture their relationships. For Example, **track** nodes are linked to **tag** nodes to associate tracks with descriptive labels. This heterogeneous graph captures the complex relationships between these entities, allowing for advanced data analysis and embedding generation. To generate embeddings for the tracks, the Node2Vec algorithm is applied to this constructed graph. Node2Vec is a feature learning method designed for graphs, leveraging random walks to extract both local and global structures of the nodes. Using the Node2Vec algorithm, embeddings are generated for each node in the graph. These embeddings are high-dimensional continuous-valued vectors that represent the position and relationships of nodes in the graph. For tracks, embeddings are predicted based on their connections to artists, albums, and tags, encapsulating contextual information such as the artist's style, the album's context, and the descriptive tags associated with the track.

3.2.2 XgBoost

The textual data, including summaries, song titles, and album names, was converted into vector representations using the Word2Vec technique. This transformation captures the semantic relationships within the text, enabling effective use of this data in downstream tasks. Additionally, an inverse transformation was applied to listener count and play count metrics to adjust their

weightage. This approach ensures that artists with lower listener and play counts are given more significance, providing a balanced representation in the dataset.

Tags associated with artists were processed using a pre-trained Sentence Transformer (SBERT) to generate high-quality embeddings. These embeddings capture the semantic meaning of the tags, allowing for their integration into the model. Agglomerative clustering was performed to organise the tags, grouping them into 10 distinct clusters based on their embeddings. A precedence order was then assigned to each cluster according to its frequency, with the least frequent clusters given higher priority. This structured representation of tags enhances their interpretability and relevance for subsequent analysis. The model is then trained using XGBoost on the prepared data, as outlined in Section 4. It aims to recommend artists to users based on the tags they provide.

3.2.3 RNN

The audio embedding process involved assigning greater importance to key features by applying weights, ensuring that critical aspects of the audio data were emphasized. Dimensionality reduction and normalization techniques were applied to numerical features to streamline the data while maintaining its integrity and comparability. An RNN (Recurrent Neural Network) was then trained on the processed embeddings to recommend five similar songs for any given input song, leveraging the sequential patterns and relationships inherent in the audio data.

3.2.4 Random Forest

The Million Song Subset includes nodes representing artists, albums, tracks, and tags from the Last.fm Dataset. The Node2Vec algorithm is utilized to predict embeddings for songs by leveraging the relationships between these nodes, such as the associations between artists, albums, and tags. This approach captures the structural and contextual relationships within the dataset to generate meaningful song embeddings.

4 Data

4.1 Million Song Dataset

The Million Song Dataset is a comprehensive collection containing audio features for over 1 million songs, offering a rich resource for music-related analysis and research. For a more focused and detailed analysis, a subset of 17,636 songs was selected, ensuring the dataset remains manageable while retaining its diversity and representativeness. This subset serves as a basis for extracting meaningful insights and developing robust models.

To enhance the relevance of features within the dataset, non-numeric columns and columns with low variance were removed. This preprocessing step ensures that only the most informative features are retained, eliminating redundant or insignificant data that could negatively impact the analysis. By refining

the dataset in this way, the feature set becomes more suitable for advanced analytical tasks, improving the overall efficiency and performance of the models applied.

4.2 Last FM Dataset

The Last.fm dataset offers an extensive repository of rich metadata, including detailed information about artists, albums, and user interactions. This comprehensive dataset is instrumental in understanding user preferences, music trends, and relationships within the music ecosystem. For the purpose of training machine learning models, a subset of 28,743 songs was selected, ensuring the data remains focused and manageable while retaining the diversity needed for robust analysis. This subset allows for the exploration of key patterns and insights in music metadata while supporting efficient model development.

To enhance the relevance of the dataset for genre-based analysis, the subset was further refined to focus on the top 40 most popular genre tags. This approach streamlined the dataset by prioritizing widely recognized and frequently used genre tags, ensuring the inclusion of critical artist, album, and user data. By narrowing the scope to these genres, the subset not only reflects popular music trends but also maintains the richness and diversity of the original data, making it highly suitable for advanced machine-learning applications.

5 Experiments

We employ Cosine Similarity measurements and clustering techniques to identify and analyze similar songs based on known examples. This analysis enables us to investigate the comparability of their view or play counts, as well as to evaluate potential biases inherent in these metrics. Given that our content-based models are trained on audio-derived features rather than raw audio signals, we aim to assess how this methodological choice influences the quality and accuracy of the recommendations. Furthermore, we seek to identify the specific audio features that contribute most significantly to the similarity and clustering outcomes.

6 Results

This section presents the outcomes of our analysis, focusing on the performance and implications of content-based and metadata-based embedding models in generating song recommendations. By evaluating the independence of play/view counts, the presence of popularity bias, and the alignment with baseline datasets such as LastFM's similar song dataset, we assess the strengths and limitations of each approach. The findings provide insights into the efficacy of mitigating biases and highlight the potential benefits for emerging artists in the music recommendation landscape.

6.1 Qualitative Analysis

The table showcases songs paired with their closest matches based on cosine similarity for each model, alongside their respective play/view counts. This comparison highlights the models' ability to identify similar

songs while accounting for popularity metrics. The results provide insights into how embedding techniques influence recommendations.

Model	**Songs / Artist**	**Play / View count**
RNN	1. Neva Play - Megan Thee Stallion 2. Superconfidential	1. 28 Million 2. 628
Tag Embedding	1. Yes I'm a mess - AJR 2. Cut your hair - Pavement	1. 8 Million 2. 528,000
Audio Embedding	1. Poppin' them thangs - G Unit 2. Money Blues - SAM	1. 161 Million 2. 156,000
XGBoost	1. The Offspring 2. Sufjan Stevens	1. 3 Million 2. 828k

6.1 Audio Feature Embedding Model

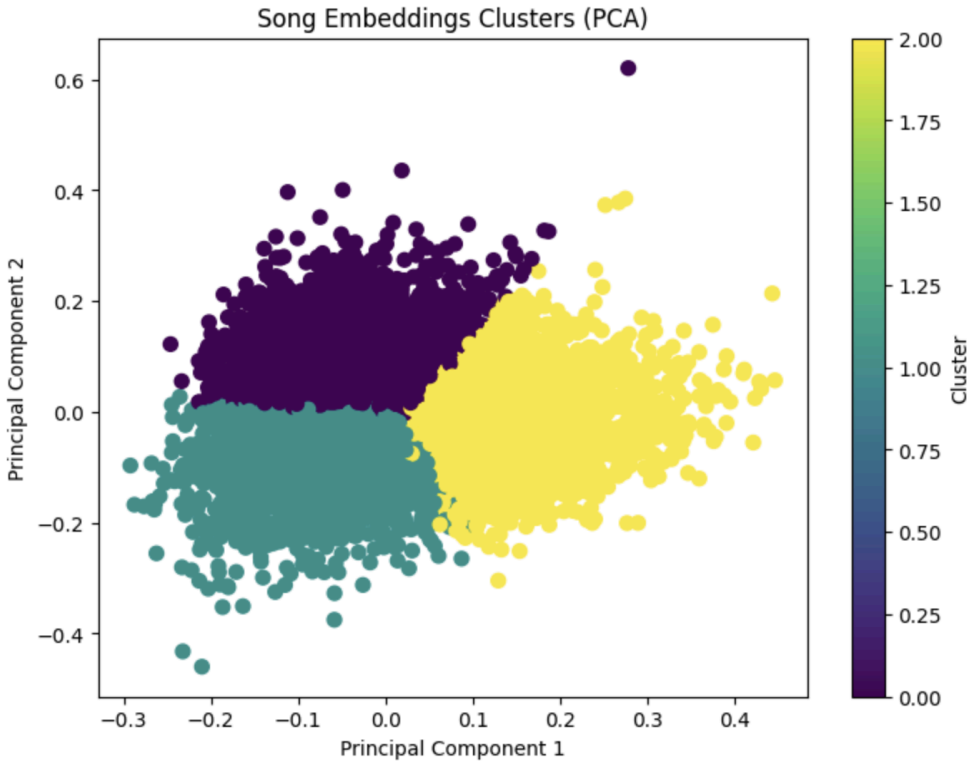

Fig 1 Clustering On Audio Feature Embeddings - 3 clusters

Cluster	**Songs**
0	Loud And Clear, Boss-Eyed Whelk, Summer Summer, Separate Ways, Body, Freiwild,
1	Baby My Love, Focus, Chim Chim Cheree, Girls Against Boys (LP Version), Saturday Night Special
2	I've Won (Introduction – Speaking), The Big Stall, Container, Money Blues, Style, High On the Mountain

Fig 2 Examples from each cluster

6.2 Tag Word2Vec Embedding Model

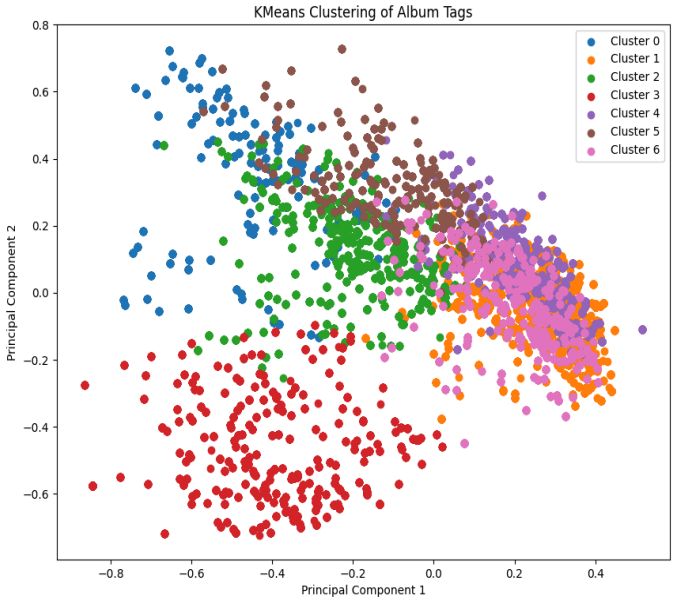

Fig 3 Clustering on Tag Word2Vec Embeddings - 7 clusters

Cluster	**Songs**
4 (roughly corresponds to classical, and jazz)	*13 Pieces, Op. 76: No. 11 Linnaea (Twinflower o, Next Year, Baby,* *Get Some Rest, Nocturne, Swan Lake,* *Op.20, Act III: 20. Hungarian Dance,* *You Love Jazz Now, Sirénes de la Féte,* *These Foolish Things (Remind Me Of You),* *Si tu n'étais pas là, Kiss Me On My Neck (Hesi)*
5 (roughly corresponds to musicals)	I would have followed you, Out Of My Dreams (bonus track), I Can Play The Piano, The Big House (feat. Frances Quinlan)
6 (roughly corresponds to house and techno)	Break & Enter (2005 Live Edit), Regeneration (2009 Remaster), I Love My Harbour, Not Giving In (feat. John Newman & Alex Clare), Tour De France (2009 Remaster), Delilah (Pull Me Out Of This), Tyranny, What If I Go? J'adore Hardcore [Explicit], is to create

Fig 4 - Examples from each cluster

6.3 Graph Embedding Model

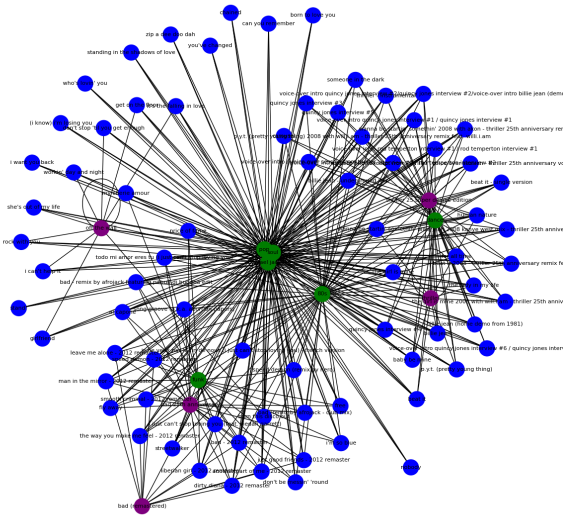

Fig 5 - Michael Jackson Connected Subgraph

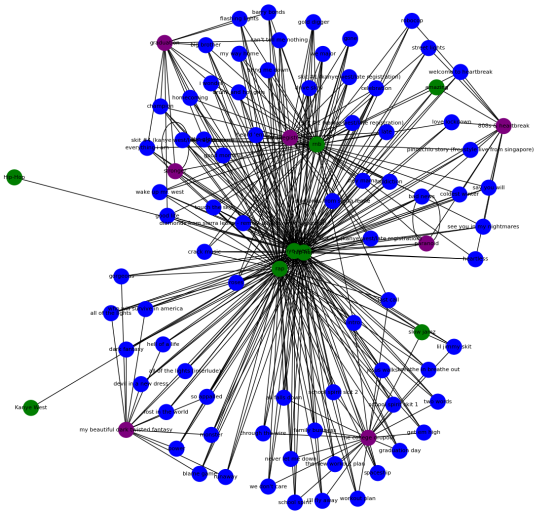

Fig 6 - Kanye West Connected Subgraph

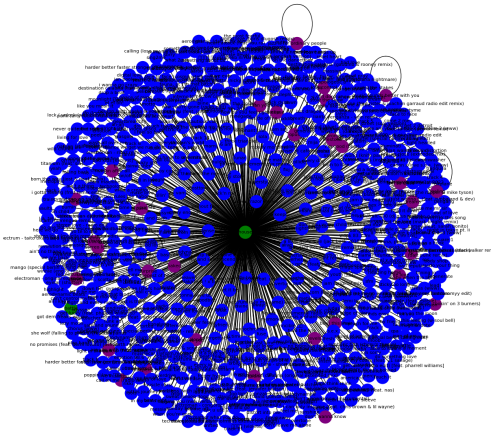

Fig 7 - Connected Subgraph of Tag "House"

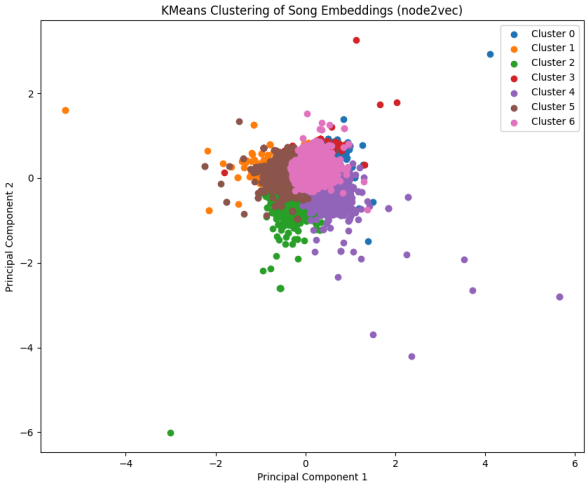

Fig 8 - Clustering on Node2Vec embeddings

Cluster	Songs
0	Chamber the Cartridge The Handshake no pressure
1	Get It Together (feat. Bread & Water) Under My Sensi Your Love
2	Waiting for Never Superman (feat. Dina Rae) Numb/Encore

6.4 XGBoost Model

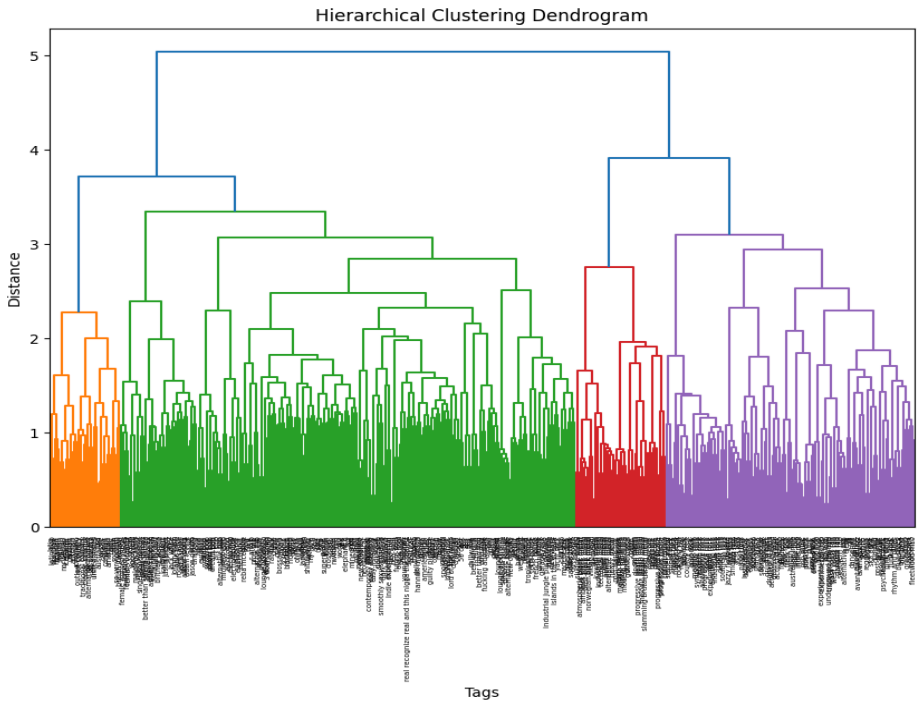

Fig 8 - Agglomerative Clustering on XGBoost Classification pruned at 9 clusters

```
Accuracy: 0.465
F1 Score: 0.37969599232937407
              precision    recall  f1-score   support

           0       0.00      0.00      0.00         8
           1       0.50      0.30      0.37        27
           2       1.00      0.06      0.12        16
           3       0.00      0.00      0.00         0
           4       0.00      0.00      0.00         2
           5       0.00      0.00      0.00         4
           6       0.52      0.89      0.65        73
           7       0.14      0.05      0.07        43
           8       0.40      0.63      0.49        27

    accuracy                           0.47       200
   macro avg       0.28      0.21      0.19       200
weighted avg       0.42      0.47      0.38       200
```

Fig 10 - Metrics for XGBoost

6.5 RNN Embedding Model

title	I Didn't Mean To	Soul Deep	Amor De Cabaret	Something Girls	Face the Ashes	The Moon And I (Ordinary Day Album Version)	Keepin It Real (Skit)	Drop of Rain	Pink World	Insatiable (Instrumental Version)	...	20.000 Seconds	Pastel
title													
I Didn't Mean To	1.000000	0.067220	0.999999	-0.917843	0.646928	0.868617	0.341102	0.988153	0.710170	0.010337	...	0.593782	0.682947
Soul Deep	0.067220	1.000000	0.068583	-0.457742	0.804313	0.552751	-0.914971	-0.086699	-0.654701	0.998380	...	-0.762892	0.774724
Amor De Cabaret	0.999999	0.068583	1.000000	-0.918385	0.647969	0.869293	0.339818	0.987943	0.709207	0.011703	...	0.592682	0.683944
Something Girls	-0.917843	-0.457742	-0.918385	1.000000	-0.896468	-0.993933	0.060058	-0.846051	-0.372365	-0.406409	...	-0.225608	-0.916792
Face the Ashes	0.646928	0.804313	0.647969	-0.896468	1.000000	0.939765	-0.496149	0.522236	-0.077430	0.769198	...	-0.229434	0.998837
The Moon And I (Ordinary Day Album Version)	0.868617	0.552751	0.869293	-0.993933	0.939765	1.000000	-0.169480	0.782285	0.268030	0.504436	...	0.117090	0.955154
Keepin It Real (Skit)	0.341102	-0.914971	0.339818	0.060058	-0.496149	-0.169480	1.000000	0.481327	0.904048	-0.936450	...	0.958910	-0.453705
Drop of Rain	0.988153	-0.086699	0.987943	-0.846051	0.522236	0.782285	0.481327	1.000000	0.809804	-0.143247	...	0.710233	0.562751
Pink World	0.710170	-0.654701	0.709207	-0.372365	-0.077430	0.268030	0.904048	0.809804	1.000000	-0.696652	...	0.988167	-0.029264

Fig 11 - Cosine Similarity Matrix

6.6 Accuracies

Model	Accuracy
Graph Node2Vec	0.025
RNN	0.05
Audio CNN Embedding	0.3
Tag Embedding	0.53
XGBoost	0.465

7 Discussion

The results indicate that the examples generated by content-based embedding models are independent of the play/view counts of the songs, demonstrating an absence of popularity bias in these models. Conversely, the metadata-based embedding models exhibit persistent popularity bias, as evidenced by the examples they produce. Baseline comparisons with LastFM's similar song dataset reveal substantial potential for improvement. This could be achieved by incorporating raw audio data instead of derived features, expanding the metadata tags in the dataset, refining the graph construction for Node2Vec to model relationships more effectively, or developing a hybrid approach that integrates these methodologies.

Notably, the content-based approach successfully mitigates popularity bias, as observed in the examples generated. This advancement holds significant promise for promoting lesser-known or emerging artists within the music industry, potentially fostering greater equity and diversity in music recommendations.

8 Conclusion

This work demonstrates the potential of leveraging unsupervised feature extraction from songs, drawing inspiration from the success of word embeddings in natural language processing. While modern systems, such as Spotify, rely heavily on predefined human-generated tags, this approach, although effective, is inherently limited by the scope and biases of human tagging capabilities. In contrast, our approach shows that unsupervised methods can uncover meaningful song representations even with relatively smaller datasets, offering a scalable and unbiased alternative for improving recommender systems.

By focusing on data-driven feature extraction, our method paves the way for equitable exposure of lesser-known or emerging artists, as it minimises biases introduced by popularity and listening history. This

has the potential to foster a more inclusive recommendation ecosystem, where diverse content is surfaced based on intrinsic song features rather than historical trends.

Future work could extend this research by exploring different embedding dimensions, experimenting with various hyperparameters, and incorporating larger and more diverse audio datasets. These advancements could further refine the effectiveness of unsupervised feature extraction, enhancing the overall quality and fairness of music recommendation systems.

References

[1] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, Minyi Guo. Knowledge Graph Convolutional Networks for Recommender Systems. arXiv:1904.12575v1 [cs.IR] 18 Mar 2019.

[2] Jongpil Lee, Kyungyun Lee, Jiyoung Park, Jangyeon Park, Juhan Nam. Deep Content-User Embedding Model for Music Recommendation. arXiv:1807.06786v1 [cs.IR] 18 Jul 2018.

[3] Seokgi Kim, Jihye Park, Kihong Seong, Namwoo Cho, Junho Min, Hwajung Hong. Can Music Be Represented With Numbers? arXiv:2102.13350 26 Feb 2021.