Music Recommendation System - Technical Report

Juee Jahagirdar
23B2185

1. Introduction

The objective of this project was to develop a **Music Recommendation System** that suggests songs based on their features using **Natural Language Processing (NLP), vector similarity measures, and machine learning techniques**. The system aims to provide personalized recommendations by analyzing song attributes and their relationships.

Additionally, during the project, I explored **Spotify Developer API** for real-time song data retrieval and built an interactive **Streamlit-based web application**. As part of my learning journey, I also implemented a **Movie Recommendation System** as an assignment, which helped me understand **collaborative filtering** and **content-based filtering** techniques.

2. Dataset

The project utilized the **Million Song Dataset**, a comprehensive collection of song metadata and audio features. The dataset was preprocessed to extract relevant song attributes such as **tempo, key, loudness, mode, lyrics (if available), and spectral features**.

3. Methodology

The project was executed in structured phases:

Week 1-2: Foundational Learning

- Introduction to **Python programming**, including data handling with **NumPy, Pandas, and Matplotlib**.
- Basics of **Natural Language Processing (NLP)**, covering text preprocessing (tokenization, stemming, lemmatization, stopword removal), feature representation techniques like **Bag of Words (BoW)** and **TF-IDF (Term Frequency-Inverse Document Frequency)**.

Week 3: Feature Extraction & Similarity Measures

- Used **Natural Language Toolkit (NLTK) and Scikit-learn** for text preprocessing and vectorization.
- Implemented **TF-IDF vectorization** to represent textual features numerically.
- Applied **cosine similarity** to compute pairwise similarity between song lyrics and metadata.
- Processed **movie data** from an API into a dataframe with names, descriptions, and genre mappings (for the **Movie Recommendation System Assignment**).
- Implemented text preprocessing functions: lowercasing, punctuation & special character removal, tokenization, stopword removal, and lemmatization

Week 4: End-to-End Model Development

- Studied and implemented an **end-to-end recommendation system** based on a movie recommendation model using collaborative filtering.
- Adapted the approach to work with **music data**, utilizing **content-based filtering**.
- Implemented **weighted feature fusion**, combining metadata and lyrics similarity scores.
- Conducted **hyperparameter tuning** to optimize similarity thresholds.

Week 5: Final System Implementation & Deployment

- Developed the **Music Recommendation System** using a **hybrid approach**, integrating:
  Content-based filtering (similarity-based recommendations).
  Spotify API data (to enhance recommendations with real-time song details).
- Designed an interactive **Streamlit-based web application** to facilitate song input and display recommendations.
- Deployed the system using **Flask API** (if applicable).

4. Challenges Faced

- **Data Preprocessing:** Handling missing values, text inconsistencies, and noise in song metadata.
- **High-Dimensional Data:** Managing sparse feature representations in TF-IDF vectors.
- **Cold Start Problem:** Addressing recommendation limitations for newly added or less popular songs.
- **Scalability:** Optimizing similarity computations for large datasets.
- **API Integration**: Learning to interact with **Spotify Developer API** and handling rate limits efficiently.

5. Results & Performance Evaluation

- Successfully built a system capable of recommending songs based on **feature similarity and NLP techniques**.
- Evaluated model performance using **precision, recall, and F1-score metrics**.
- Conducted **qualitative validation** by analyzing user feedback on recommendations.

6. Future Scope

- Integrating **collaborative filtering** to complement content-based recommendations.
- Leveraging **deep learning-based embeddings** such as **Word2Vec, FastText, or BERT** for enhanced text representation.
- Implementing a **graph-based recommendation system** using **Neo4j** to capture relationships between songs.
- Deploying the model as a **fully functional web application** with user interaction features.

7. Conclusion

The Music Recommendation System successfully applied **NLP techniques, similarity measures, and content-based filtering** to provide meaningful song recommendations. The project facilitated hands-on experience with **machine learning, information retrieval, and software deployment**, contributing to a deeper understanding of AI applications in the music industry.

8. Acknowledgments

I would like to express my sincere gratitude to my mentor for their guidance and support throughout the project. Their insights and feedback were invaluable in refining the system. Additionally, I extend my thanks to the **Analytics Club** for organizing and facilitating WiDS 4.0, providing a structured learning experience and the opportunity to work on this exciting project.