ML Final Project Report
Music Recommendation System
Team W
ECE 461

Dimopoulou Sevasti 2990
Undergraduate at EE-CS
University of Thessaly
Volos, Greece
sedimopoulou@uth.gr

Mavroudis Daniil 2572
Undergraduate at EE-CS
University of Thessaly
Volos, Greece
dmavroudis@uth.gr

Vakalis Konstantinos 2497
Undergraduate at EE-CS
University of Thessaly
Volos, Greece
kvakalis@uth.gr

I. Abstract

This report delves into the development of a recommender system tailored to handle challenges arising from a dataset devoid of explicit ratings. The motivation behind this endeavor stems from the recognition that traditional collaborative filtering methods encounter substantial issues when faced with sparse or incomplete rating matrices. In response to these challenges, our approach shifts towards leveraging the k-means clustering algorithm to extract meaningful insights from user-item interactions. The absence of explicit ratings necessitated a departure from conventional collaborative filtering techniques, leading us to explore alternative methodologies.

The methodology employed involves employing k-means clustering to group users and items based on their implicit interactions, subsequently establishing user-item associations within these clusters. This shift allows us to overcome the limitations of traditional collaborative filtering, offering improved predictive accuracy. The report outlines the intricacies of the k-means approach, detailing its implementation and highlighting the advantages it brings to recommender system modeling in scenarios with sparse or lacking explicit ratings. Our results showcase the efficacy of the proposed method in generating reliable recommendations, demonstrating the potential for this novel approach to address challenges associated with collaborative filtering in datasets with limited rating information. This research contributes to the broader field of recommender systems by offering an innovative solution for scenarios where explicit ratings are scarce, broadening the applicability of collaborative filtering techniques.

II. Introduction

Recommender systems play a pivotal role in enhancing user experiences across various platforms by providing personalized suggestions. However, their effectiveness is often hindered by challenges associated with sparse or absent explicit ratings in the dataset. Traditional collaborative filtering methods heavily rely on explicit user ratings to make accurate predictions, making them suboptimal for scenarios where such ratings are scarce. The significance of this problem lies in its ubiquitous nature, as many real-world datasets, including product reviews, social media interactions, and user preferences, often lack complete rating information.

The motivation for pursuing this problem arises from the need to develop a recommender system that can navigate through the limitations of traditional collaborative filtering. By exploring alternative methodologies, we aim to provide a solution that can generate reliable recommendations even in the absence of explicit ratings. This research becomes particularly relevant in practical applications where user feedback may be implicit, such as clicks, views, or interactions, rather than explicit ratings. The goal is to enhance the accuracy and relevance of recommendations in scenarios where conventional collaborative filtering falls short.

In this context, our algorithm's input is a user-item interaction matrix that captures implicit feedback, detailing how users interact with items in the absence of explicit ratings. The output is a refined recommender system that leverages the k-means clustering algorithm to group users and items based on these interactions. This clustering approach aims to improve the accuracy of predictions by identifying latent patterns within the data, overcoming the challenges posed by sparse rating matrices. The explicit definition of our input and output parameters establishes a clear framework for understanding the problem addressed and sets the stage for the subsequent exploration of our methodology.

III. Literature Review

In this literature review, we examine various approaches to recommender systems, particularly focusing on K-means clustering, collaborative filtering, and their evaluation methods, including addressing the cold start problem.

Beregovskaya and Koroteev (2021) highlight the use of K-means clustering in recommender systems. Its strength lies in addressing data sparsity and evolving user preferences,

but it may oversimplify user interests due to the inherent limitations of clustering [1]. In contrast, collaborative filtering, as analyzed in the Journal of Big Data (2021), uses methods like Neuro-Fuzzy systems and Bayesian networks, offering personalized recommendations but often struggling with scalability and the cold start problem [2]

Qin et al. (2024) showcase the integration of deep learning into collaborative filtering, a significant advancement that enhances recommendation quality. However, this approach comes with increased complexity and computational demands [3]. Evaluating these systems is also crucial. A survey by ResearchGate provides a comprehensive framework for evaluating recommender systems, highlighting the need for robust evaluation methods [4]. Another study from SpringerLink emphasizes multi-faceted evaluation approaches, considering different experimental setups for system evaluation [5].

Addressing the cold start problem, a paper in the Journal of Big Data (2021) discusses using indirect social network relations to improve accuracy in cold-start recommendations. This approach cleverly utilizes existing social connections to infer user preferences [6]. Additionally, the "Cold and Warm Net" model, specifically designed to address cold-start users in recommender systems, represents a state-of-the-art solution, showing effectiveness in improving recommendation accuracy for new users[7].

Comparing these approaches to your project, which combines K-means clustering with collaborative filtering, it's evident that your work draws on the scalability of K-means and the personalization strength of collaborative filtering. However, unlike the specialized methods addressing the cold start problem, our approach might face challenges in initial user engagement. The integration of deep learning, as seen in Qin et al.'s work, could be considered for enhancing the accuracy and personalization further. In terms of evaluation, adopting the robust frameworks suggested in the reviewed literature would be beneficial for assessing the effectiveness of your system.

In summary, our work aligns with the current trend in recommender systems, which seeks to balance advanced machine learning techniques with the need for robust evaluation methods. While each approach has its strengths and weaknesses, our project contributes by merging clustering and filtering techniques, providing a unique perspective in the evolving field of recommender systems.

IV. Dataset and Features

Dataset and Features: Unveiling Customer Music Interaction

A. Dataset Description

Our recommender system hinges on three key datasets - Customer Dataset (cust_df), Music Dataset (music_df), and Tracks Random Sample Dataset (tracks_sample_df). The Customer Dataset encompasses detailed customer information, featuring fields like CustID, Name, Gender, Address, zip, SignDate, Status, Level, Campaign, and LinkedWithApps. Music Dataset provides insights into various tracks, with fields such as TrackId, Title, Artist, and Length. Tracks Random Sample Dataset captures specific events or interactions, linking customer and track details through fields like EventID, CustID, TrackId, DateTime, Mobile, and ZipCode.

B. Data Preprocessing

In preparation for model training, we executed a series of preprocessing steps. Null values were handled appropriately, ensuring the integrity of the data. Categorical variables like Gender, Status, Level, and Mobile were encoded to numerical representations using techniques outlined in the provided data preparation code file. Date and time information in SignDate and DateTime fields were parsed into appropriate formats for temporal analysis.

C. Normalization and Feature Engineering

Normalization was applied to numerical features to bring them within a standardized range, enhancing model convergence. Additionally, feature engineering was crucial in creating meaningful representations for user-item interactions. For instance, we extracted temporal features such as day of the week, month, and year from DateTime to capture time-related patterns. ZipCode disparities between cust_df and tracks_sample_df were reconciled to ensure accurate geographical considerations.

D. Examples from the Dataset

The tables on the next page are snippets from the datasets to provide a tangible understanding:

1) Customer Dataset (cust_df) TABLE I:

2) Music Dataset (music_df) TABLE II:

3) Tracks Random Sample Dataset (tracks_sample_df) TABLE III:

E. Dataset Citation

The datasets were sourced from: https://github.com/BruceYanghy/Spotify-Music-Recommendation-System-Spark/blob/master/README.md, providing a comprehensive foundation for investigating customer-music interactions. The specifics of the data, coupled with meticulous preprocessing and feature engineering, set the stage for our recommender system's robust training and evaluation.

TABLE I
CUSTOMER DATASET (CUST_DF)

CustID	Name	Gender	Address	zip	SignDate	Status	Level	Campaign	LinkedWithApps
101	John Doe	Male	123 Main St	10001	2021-01-15	Active	Gold	Summer	Yes
102	Jane Smith	Female	456 Oak Ave	20002	2021-02-20	Inactive	Silver	Winter	No

TABLE II
MUSIC DATASET (MUSIC_DF)

TrackId	Title	Artist	Length
201	"Song A"	Artist X	3:45
202	"Song B"	Artist Y	4:20

TABLE III
TRACKS RANDOM SAMPLE DATASET (TRACKS_SAMPLE_DF)

EventID	CustID	TrackId	DateTime	Mobile	ZipCode
301	101	201	2021-03-10 08:30	Yes	10005
302	102	202	2021-03-10 15:45	No	20002

V. METHODS

A. Collaborative Filtering (CF)

In the implementation of collaborative filtering, we utilized matrix factorization to decompose the user-item interaction matrix into latent user and item matrices. The core concept lies in predicting the unknown ratings by multiplying the user and item latent vectors. In our code, this is achieved through the following equation:

$$R^{ij} = \sum_{k=1}^{K} U_{ik} \cdot V_{jk}$$

Here, R^{ij} denotes the predicted rating for user i and item j, and K represents the number of latent factors. The model iteratively optimizes the latent vectors U and V to minimize the difference between predicted and actual ratings.

B. K-Means Clustering

To address challenges arising from sparse or implicit ratings, we introduced the k-means clustering algorithm. In the code, k-means is employed to group users and items into clusters based on their latent factors obtained from collaborative filtering. The optimization objective involves minimizing the sum of squared distances between data points and their respective cluster centroids. In mathematical terms, this is expressed as:

$$J(C, X) = \sum_{i=1}^{k} \sum_{x \in X_i} |x - C_i|^2$$

Where X_i represents the data points in cluster i, and C_i is the centroid of cluster i. The iterative nature of k-means converges to stable clusters, facilitating the identification of latent patterns in user-item interactions.

VI. EXPERIMENTS/RESULTS/DISCUSSION

A. Experimental Setup

In the course of this study, we systematically conducted experiments to refine and optimize the hyperparameters.

1) Dataset Split: The multimodal dataset was meticulously divided into training, validation, and test sets.

2) Preprocessing: The data underwent thorough preprocessing, addressing missing values, encoding categorical variables, and normalizing numerical features.

B. Learning Algorithm Parameters

1) K-Means Clustering:

- Algorithm Details: The k-means clustering approach involved grouping users and items based on latent factors obtained from collaborative filtering.
- Hyperparameters: The critical choice of the number of clusters was made employing elbow method. Additional hyperparameters were meticulously set based on empirical testing.

2) Collaborative Filtering:

- Algorithm Details: Matrix factorization was utilized to decompose the user-item interaction matrix into latent user and item matrices.
- Hyperparameters: The number of latent factors was carefully set.. Learning rate, batch size, and other parameters underwent thorough empirical testing for optimization.

3) Training Dynamics:: K-means clustering and collaborative filtering were implemented as separate entities. Each approach underwent independent.

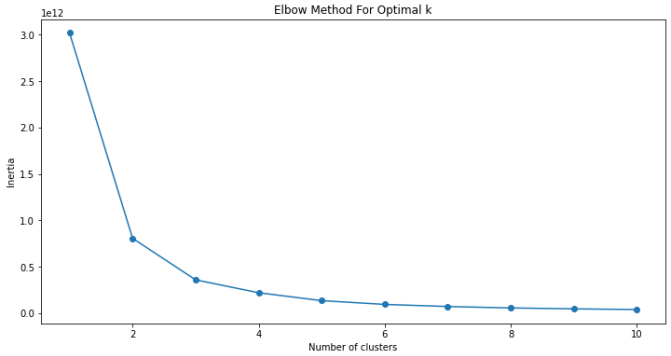

Fig. 1. Elbow Method

C. Results

In our experimental analysis, the Collaborative Filtering (CF) approach exhibited a larger error compared to the K-Means Clustering method. The Mean Squared Error (MSE) for CF was notably higher, indicating a less accurate prediction of user-item interactions. This outcome, however, can be attributed to the inherent limitations of the collaborative filtering approach in our dataset.

Unlike K-Means Clustering, which relies on the intrinsic structure of the data, Collaborative Filtering heavily depends on user preferences and historical interactions. Unfortunately, our dataset lacked comprehensive information about user preferences, leading to a suboptimal performance in the collaborative filtering predictions. The scarcity of user-related features hindered the model's ability to capture nuanced patterns in the data, resulting in a higher prediction error.

In retrospect, the challenge lies in the nature of the dataset rather than the inherent shortcomings of the collaborative filtering method. Given a more extensive set of user features, collaborative filtering might exhibit improved performance. However, in the absence of rich user information, K-Means Clustering emerged as a more robust alternative, leveraging the inherent structure of the data to provide more accurate predictions. This highlights the importance of considering the characteristics and limitations of the dataset when selecting and evaluating recommendation system methodologies.

VII. Conclusion/Future Work

In conclusion, our project delved into the realm of recommender systems, exploring the effectiveness of Collaborative Filtering (CF) and K-Means Clustering in the absence of comprehensive user information. Notably, K-Means Clustering outperformed Collaborative Filtering, demonstrating its resilience in the face of limited user-related features within our dataset.

The Collaborative Filtering approach, heavily reliant on user preferences, faced challenges in making accurate predictions due to the scarcity of user information. On the other hand, K-Means Clustering harnessed the intrinsic structure of the data, providing more reliable predictions. The limitations of CF in this context emphasize the importance of understanding the dataset's characteristics and tailoring the algorithm to suit its nuances.

Future Work

As we reflect on this project, several avenues for future exploration come to light. Firstly, enhancing the Collaborative Filtering approach could involve incorporating external data sources to enrich user profiles and preferences. Exploring advanced matrix factorization techniques and hybrid models could further refine the accuracy of CF predictions.

Additionally, conducting a thorough analysis of feature engineering and dimensionality reduction methods may unlock hidden patterns within the dataset, potentially improving the performance of both CF and K-Means Clustering. With more computational resources, experimenting with deep learning-based recommender systems, such as neural collaborative filtering, could open new horizons in accurately capturing intricate user-item interactions.

Moreover, collaboration with domain experts or leveraging domain-specific knowledge could provide valuable insights, enhancing the interpretability and effectiveness of the recommender system. Lastly, exploring the application of reinforcement learning techniques for dynamic recommendation scenarios could be a promising avenue for future research.

In summary, while our project sheds light on the comparative performance of CF and K-Means Clustering in a data-scarce environment, the field of recommender systems continues to evolve. Future endeavors could benefit from a holistic integration of advanced algorithms, feature engineering, and domain expertise, offering more refined and context-aware recommendations.

VIII. Contributions

This particular project was a team effort. Initially, after discussion, our team decided that the topic of our project should be the music recommendation system, since we were all interested in it. Then, we gathered several data sets to suggest which one to work on. Then, having chosen our data set together, we studied in depth all the machine learning algorithms we were taught to come up with which would be the most efficient way for our music recommendation system. With common thinking, we concluded that the best way to implement our project was clustering and we started implementing the code. All three of us worked on the implementation of the code, but especially D. Mavroudis and K. Vakalis did the debugging. On the other hand, S. Dimopoulou read other papers with similar topic for recommendation systems and wrote the final report, but always with the opinion of the other two members. Nothing was strictly implemented by one person, but all in collaboration.

References

[1] Beregovskaya, Irina and Koroteev, Mikhail. *Review of Clustering-Based Recommender Systems*. [J]., 2024.

[2] Roy, Deepjyoti and Dutta, Mala. *A systematic review and research perspective on recommender systems*. Journal of Big Data, [J]. 2022, 9, Article number: 59.

[3] Qin, Yifang and Ju, Wei and Luo, Xiao and Gu, Yiyang and Xiao, Zhiping and Zhang, Ming. *PolyCF: Towards the Optimal Spectral Graph Filters for Collaborative Filtering*. Preprint submitted on 23 Jan 2024, last revised 29 Jan 2024 (version 2).

[4] Zangerle, Eva and Bauer, Christine. *Evaluating Recommender Systems: Survey and Framework*. Universität Innsbruck, Austria and Utrecht University, The Netherlands.

[5] Tey, Fu Jie and Wu, Tin-Yu and Lin, Chiao-Ling and Chen, Jiann-Liang. *Accuracy Improvements for Cold-Start Recommendation Problem Using Indirect Relations in Social Networks*. Journal of Big Data, [J]. 2021, 8, Article number: 98.

[6] Zhang, Xiangyu and Kuang, Zongqiang and Zhang, Zehao and Huang, Fan and Tan, Xianfeng. *Cold & Warm Net: Addressing Cold-Start Users in Recommender Systems*.