

A Work Project, presented as part of the requirements for the Award of a Master's degree
in Business Analytics from the Nova School of Business and Economics.

**Improving Personalized Recommendations for Cold-Start Users on the NetEase
Cloud Music Platform: User Similarity-Based Recommendation Algorithm**

Mykyta Kolisnyk

Work project carried out under the supervision of:

Yufei Shen

17-12-2024

Abstract

The rise of big data and rapid digitization in the digital media industry have made recommendation systems essential for delivering relevant, personalized content to users. In the music streaming sector, platforms like NetEase face the cold-start problem when recommending content to new users with minimal interaction data. To address this, we developed and compared various techniques, including a User Similarity-Based Recommendation Algorithm, a User Preference Elicitation Recommendation Algorithm, a DeCS-Inspired Recommendation Algorithm, and a Discriminative Frequent Itemsets model. Our findings show that the DeCS-Inspired model performs best in data-rich scenarios, while Demographic-Based methods excel in cold-start situations. To optimize performance, we propose a hybrid approach that combines Demographic-Based techniques for cold-starts and transitions to the DeCS-Inspired model as user data grows.

Key Words: Recommendation System, Information System, Cold-Users, Music Streaming Service

This work used infrastructure and resources funded by Fundação para a Ciência e a Tecnologia (UID/ECO/00124/2013, UID/ECO/00124/2019 and Social Sciences DataLab, Project 22209), POR Lisboa (LISBOA-01-0145-FEDER-007722 and Social Sciences DataLab, Project 22209) and POR Norte (Social Sciences DataLab, Project 22209).

INDEX

1. Introduction.....	4
2. Literature Review.....	6
2.1 Challenges and Solutions for Cold-Start Scenarios	6
2.2 Advanced Learning Models and Optimization Techniques	7
2.3 Emerging Trends in Cold-Start Recommendation Systems	8
2.4 Limitations and Future Perspective.....	8
2.5 Summary.....	8
3. Context and Data.....	9
3.1 NetEase Research Context.....	9
3.2 Data Description.....	9
3.3 Metadata	9
3.4 Cold-Start Problem: Definition and Challenges.....	10
4. Exploratory Data Analysis	11
4.1 Feature Explanation.....	12
4.2 Summary.....	13
5. Methods	14
5.1 Evaluating Recommender Systems.....	14
6. User Similarity-Based Recommendation Algorithm	15
6.1 Feature Engineering	16
6.2 Data Preprocessing.....	17
6.3 Similarity Index.....	18
6.4 Recommendation based on similarity	19
6.5 Recommendation Evaluation.....	20
6.6 Results and Interpretation.....	20

6.7 Summary.....	22
7. Conclusion and Discussion.....	23
7.1 Key Findings.....	23
7.2 Limitations.....	25
7.3 Research Contributions	26
7.4 Implications for Practice.....	27
7.5 Future Research Directions.....	27
8. References	29
9. Appendix	34

1. Introduction

Automation and machine learning drive digital interactions, shaping social media, e-commerce, and entertainment. Recommendation systems analyze user behavior to personalize content, enhancing engagement and retention (What Netflix's Recommendation Systems Can Teach Us About The Computing Challenges Of The Near Future, n.d.). As digital platforms expand, managing vast content efficiently requires advanced algorithms, especially in music streaming, where personalization is key to competitiveness (AI in Music Streaming: Personalization at its Best, n.d.). *Figure 1* provides an abstract overview of the key components that constitute a recommendation system.

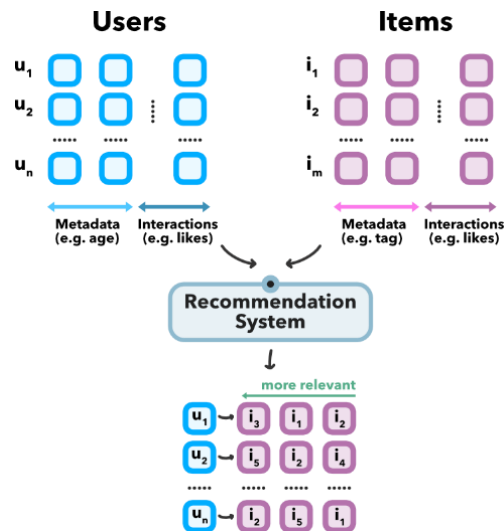


Figure 1 - Recommendation System Key Components Overview

The music streaming industry is growing rapidly, with global revenue projected to exceed \$33.97 billion by 2027 (Statista, n.d.). In China, Tencent Music and NetEase leverage AI-driven recommendations to maintain market dominance in a sector expected to grow at a

CAGR of 4.84% from 2024 to 2027 (Financial Times, 2023). AI-driven accuracy is crucial for user acquisition, retention, and satisfaction (AI-based Recommendation System Market Report, n.d.). However, the cold-start problem—where a system lacks historical interaction data—remains a key challenge, leading to suboptimal recommendations (Chadha & Jain, 2022).

The cold-start problem appears in three contexts: new-user, new-item, and new-community challenges (Pandey & Rajpoot, 2016). New users lack behavioral data, limiting personalized recommendations. New items have no prior engagement, affecting visibility. New communities, such as platform launches, create a lack of recommendation history, reducing content discoverability and user satisfaction (Sun & Peng, 2022).

Several approaches mitigate cold-start issues. Content-based filtering recommends items based on metadata, addressing new-item challenges. Preference elicitation collects initial user data at registration to generate recommendations (Pu & Chen, 2008). Demographic clustering segments users based on shared traits, enabling early personalization. However, these methods have limitations—demographic-based recommendations may be too generic, while emphasizing globally popular items can reduce content diversity (How Retailers Can Keep Up with Consumers, n.d.).

This study enhances NetEase Cloud Music’s recommendation system for cold-start users by evaluating multiple strategies. The DeCS-Inspired model performs well in data-rich environments, while demographic similarity and preference elicitation improve early recommendations. A hybrid model integrating these approaches balances accuracy and

scalability, improving personalization for new users (Music Streaming - China | Statista Market Forecast, n.d.).

The findings highlight the importance of hybrid models in addressing cold-start challenges. By combining demographic clustering, preference elicitation, and AI-enhanced techniques, NetEase Cloud Music can enhance engagement and user retention. Future research should refine these models to improve accuracy while ensuring scalability, leading to a more adaptive and efficient recommendation system.

2. Literature Review

Recommendation systems play a pivotal role in curating and ranking content to optimize user experiences across digital platforms. However, they face a fundamental challenge known as the cold-start problem, which occurs when insufficient historical data is available to generate accurate recommendations (Pandey & Rajpoot, 2016). Various methodologies, including demographic-based clustering, hybrid models, and deep learning techniques, have been developed to mitigate this limitation (Lam et al., 2008).

2.1 Challenges and Solutions for Cold-Start Scenarios

The cold-start problem manifests in three primary forms: new-item, cold-user, and new-community issues. Traditional Matrix Factorization (MF) models perform poorly in these contexts due to their reliance on past interactions. Meta-learning techniques address this limitation by leveraging patterns from similar users or items to improve recommendation quality (Zhu et al., 2020). Additionally, demographic-based models categorize users based on shared attributes, allowing for initial content personalization when behavioral data is

unavailable. Deep learning approaches, such as DropoutNet, enhance predictions by incorporating auxiliary metadata into training processes (Roy & Dutta, 2022; Steck et al., 2021).

Hybrid recommendation systems combine collaborative filtering (CF) and content-based filtering (CBF) to improve accuracy (Nguyen et al., 2007). For example, Spotify's "Discover Weekly" refines recommendations by integrating CF, user behavior, and contextual factors to tailor playlists (Kowald et al., 2020). Emerging architectures, including two-tower models and task-adaptive neural networks, further optimize cold-start recommendations by dynamically adjusting prediction mechanisms based on limited user interactions (Feng et al., 2021; Lin et al., 2021). The incorporation of Large Language Models (LLMs) has also facilitated synthetic user behavior generation, improving recommendation accuracy through enriched training data (Han et al., 2022; Wang et al., 2024).

2.2 Advanced Learning Models and Optimization Techniques

Demographic data allows systems to group users and items with similar characteristics, improving recommendation accuracy when historical data is lacking (Zhu, et al. 2020). Studies also highlight incorporating Deep Learning using auxiliary information, including - but not limited - to text, user metadata, and other additional information that could generate more accurate predictions (Roy e Dutta 2022), (Steck, et al. 2021). One example of a Deep Learning based approach is DropoutNet which focuses on training neural networks for cold-start situations, using dropout techniques to create the effect of missing data during the training phase (Steck, et al. 2021).

2.3 Emerging Trends in Cold-Start Recommendation Systems

Recent advancements in self-supervised learning and transformer-based architectures have revolutionized recommendation systems. BERT-based models leverage sequential user interactions to refine personalization strategies (Karabila et al., 2024). Zero-shot learning techniques, which generate predictions with minimal historical data, have also demonstrated strong performance in cold-start scenarios (Hafnar & Demšar, 2024). Frequent pattern clustering enhances content diversity by identifying behavioral similarities among users, mitigating the risk of recommendation redundancy (Panteli & Boutsinas, 2023).

2.4 Limitations and Future Perspective

Despite recent progress, existing cold-start mitigation techniques still present challenges. Demographic-based models risk oversimplifying user preferences, while deep learning solutions require substantial computational resources, limiting scalability (Steck et al., 2021). Future research should prioritize cost-effective, scalable solutions that balance efficiency with personalization while addressing algorithmic biases (Silva et al., 2019). Contextual embeddings and zero-shot learning hold promise as effective strategies for advancing recommendation models in data-scarce environments.

2.5 Summary

State-of-the-art recommendation systems integrate deep learning, hybrid modeling, and meta-learning techniques to enhance personalization for cold-start users. Advances in contextual embeddings, self-supervised learning, and synthetic user behavior modeling have significantly improved recommendation accuracy. Future developments should focus on

refining low-data solutions while ensuring computational efficiency and minimizing bias, ultimately fostering more effective and scalable recommendation systems.

3. Context and Data

3.1 NetEase Research Context

NetEase Cloud Music (NCM), launched in 2013, is a leading music streaming platform in China, known for its "Cloud Village" feature, which enhances user engagement through likes, shares, and comments (Zhang et al., 2022). This study utilizes a dataset comprising over 57 million interactions recorded in November 2019, capturing detailed user behavior and content metadata. The primary focus is on addressing the cold-start problem, ensuring a balance between personalization and diversity, and enhancing user engagement strategies.

3.2 Data Description

The dataset contains over 57 million impressions from the "Cloud Village" discovery tab, encompassing user interactions such as clicks, likes, shares, comments, and views. These data points provide granular insights into engagement trends. Additional datasets capture impression-related attributes, including user and creator demographics, enriching the understanding of interaction dynamics within the platform.

3.3 Metadata

User interactions are recorded using binary fields (e.g., `isClick`, `isLike`, `isShare`, `isComment`), while `view time` quantifies engagement duration. `ImpressPosition` and `ImpressTime` contextualize user interactions. Unique identifiers such as `mlogId`, `songId`, and `artistId`

distinguish content, whereas attributes like `publishTime`, `contentId`, and `talkId` provide further contextual metadata. Demographic attributes – including age, gender, and user level – are crucial in refining recommendation accuracy and personalizing content delivery.

The table below (*Table 1*) provides a comprehensive overview of these metadata variables, summarizing the features that were used in our recommender systems:

Element	Variable	Description	Data Type
Position Element	<code>ImpressPosition</code>	Position of the card in the user's feed, starting from 1.	Integer
Engagement Element	<code>isClick</code>	Binary value indicating if the user clicked on the card (1 for yes, 0 for no).	Binary
	<code>isLike</code>	Binary value indicating if the user liked the card.	Binary
	<code>isComment</code>	Binary value indicating if the user commented on the card.	Binary
	<code>isViewComment</code>	Binary value indicating if the user viewed card comments.	Binary
	<code>isShare</code>	Binary value indicating if the user shared the card on social platforms.	Binary
	<code>isIntoPersonalHomepage</code>	Binary value indicating if the user accessed the creator personal homepage.	Binary
	<code>ImpressTime</code>	Epoch timestamp representing when a card was displayed to a user in millisecond.	Numeric
Card Element	<code>mlogViewTime</code>	Duration (in seconds) the user spent viewing the card.	Numeric
	<code>mlogId</code>	Unique identifier for each card displayed to users.	String
	<code>songId</code>	Unique identifier for each song in a card.	String
	<code>artistId</code>	Unique identifier for each artist in a card.	String
	<code>creatorId</code>	Unique identifier for the card creator.	String
	<code>contentId</code>	A set of content identifiers attributed to a card.	List
	<code>publishTime</code>	The number of days when the card is published till December 1st, 2019.	Integer
	<code>talkId</code>	Unique level that represents the anonymized topic of the card.	Integer
User Element	<code>cardType</code>	Type of card (e.g., video or image with background music).	Categorical
	<code>likesCount</code>	Total number of likes received by the card until the timestamp.	Integer
	<code>userId</code>	Unique identifier for each user.	String
	<code>registeredMonthCnt</code>	Number of months between a user's registration time and December 1st, 2019.	Integer
Demographics Element	<code>userLevel</code>	Activity intensity of a user, ranging from 0 to 10.	Integer
	<code>province</code>	Province of residence of the user (in Pinyin format).	String
	<code>age</code>	The age of a user.	Integer
	<code>gender</code>	The gender of a user.	String

Table 1 - NetEase Table Metadata: Variable Descriptions by Category

3.4 Cold-Start Problem: Definition and Challenges

The cold-start problem impairs recommendation system effectiveness, particularly for new users and content. This study focuses on the **New User Problem**, where minimal engagement history hinders personalized recommendations. Cold-start users are identified based on: **Registration Date** (users registered within the last six months), **Interaction Count** (users

with four or fewer recorded interactions), and **Activity Level** (users with a platform engagement level of three or lower).

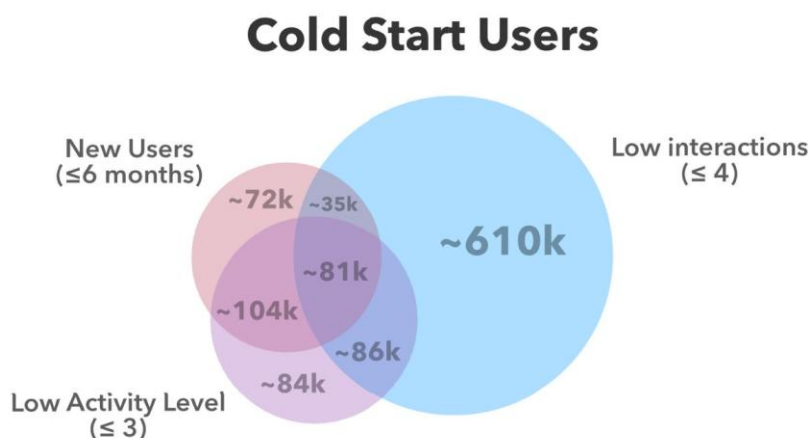


Figure 2 - Cold start users definition

These users constitute 3.88% (See Figure 2) of the dataset, presenting challenges in recommendation accuracy. Analysis shows that 25% of users had ≤ 4 interactions, 50% had fewer than seven, and 75% had more than 17 interactions, with an average of 26. The dataset mainly provides implicit feedback (e.g., clicks, shares), which lacks explicit user preference indicators. Additionally, early interactions often reflect exploration rather than long-term intent, complicating recommendation accuracy.

4. Exploratory Data Analysis

In the direction of understanding user behavior and data quality within the NetEase Cloud Music platform, we carefully analyzed key variables and interaction patterns present in the dataset. The analysis also addressed data inconsistencies, outliers, and the distribution of user engagement across the platform. Below is a detailed breakdown of the key findings and steps taken during the EDA.

4.1 Feature Explanation

The analysis focused on temporal variables such as `impressTime`, initially recorded as a Unix timestamp and converted into a readable datetime format. Additional features like `day`, `hour_of_day`, and `day_of_week` were created to analyze engagement patterns. Findings showed peak activity on Fridays and Saturdays, with user engagement rising in the afternoon and late at night (Appendix 1 and 2). These insights highlight the importance of temporal factors in content delivery strategies.

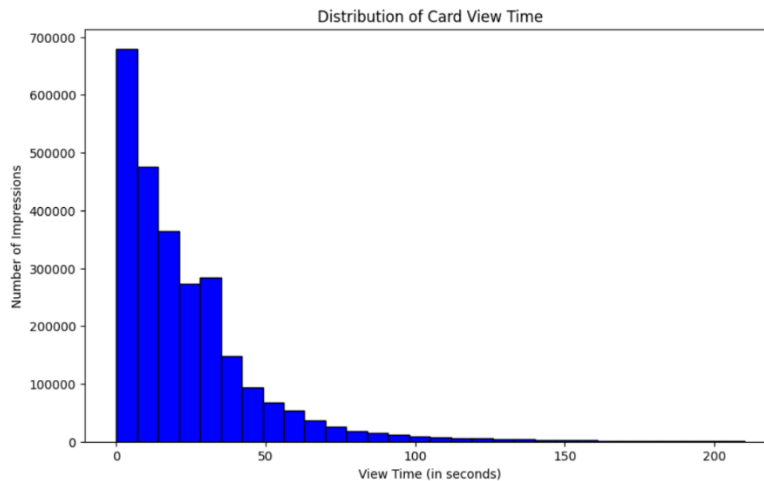


Figure 3 – Long-tail distribution of users interactions

Data validation revealed inconsistencies, such as 3.21% of records with an invalid `impressPosition` of 0, which were removed to maintain dataset integrity. Additionally, discrepancies in `isClick` and `mlogViewTime` were identified, with cases where users clicked but registered no viewing time and vice versa. These anomalies were flagged and excluded to enhance dataset accuracy. Notably, 95.2% of `mlogViewTime` values were 0 or null, indicating low engagement levels and a long-tail distribution of user interaction (See Figure 3).

User behavior analysis using `userId` examined click-to-impression ratios to assess engagement levels. Similar analysis on `mlogId` identified high-performing content based on impressions and clicks. Popularity trends were also explored through `songId` and `artistId`, revealing a concentration of engagement around specific artists and tracks.

Content-type analysis showed that image-based content (53.9%) was slightly more prevalent than video content (46.1%). `PublishTime` data indicated that most content was created about two months before the data collection period. Demographic insights categorized users by age and gender, revealing that 35.31% of users were male, while 37% had unknown gender information. Registration data was refined, removing erroneous records predating the platform's launch in April 2013.

These findings enhance understanding of user behavior, content popularity, and engagement trends, supporting improvements in recommendation algorithms and user experience.

4.2 Summary

The EDA provided valuable insights into user behavior and data quality. Engagement levels were low, with only 3.82% of impressions leading to actions, posing challenges for predicting user interaction. Comments were significantly less frequent than clicks, likes, and shares, suggesting the need for strategies to encourage more diverse user participation.

Temporal analysis revealed distinct engagement patterns, with a long-tail distribution where a small subset of users and content accounted for most interactions. Addressing data inconsistencies and refining engagement models will enhance recommendation precision, ultimately improving personalization and user experience on NCM.

5. Methods

In designing an effective recommender system for NetEase Cloud Music, we implemented multiple approaches, with a primary emphasis on the **User Similarity-Based Recommendation Algorithm**. This technique utilized demographic data, converting it into vectorized representations and computing similarity scores using the FAISS library to enhance recommendation accuracy.

To supplement this core approach, we explored additional methodologies:

- **User Preference Elicitation:** A content-based filtering approach assuming that cold users had previously interacted with specific music cards, enabling initial preference predictions.
- **DeCS-Inspired Model:** A deep neural network integrating user and item embeddings with side information, leveraging dimensionality reduction to improve recommendation quality.
- **Discriminant Frequent Item-Sets:** Applied K-means clustering to categorize users based on demographics and extracted frequently consumed music sets within each cluster to inform recommendations.

5.1 Evaluating Recommender Systems

Given the dataset's size, comprising extensive demographic profiles, creator statistics, impression data, and user activity logs, computational efficiency was a key challenge. The volume of interactions and the application of machine learning models led to high resource

consumption, requiring an optimized approach. Peak computational usage reached **51 GB of RAM, 225 GB of disk storage, and 100 compute units**.

To address these constraints, we employed a **5% representative sample** of the dataset for exploratory analysis and initial model validation. This strategic sampling approach facilitated efficient model testing without exceeding system memory limits while ensuring the methodologies were scalable. The iterative evaluation process allowed us to refine and optimize the recommendation pipeline, ensuring its adaptability for full-scale deployment as computational resources permitted.

6. User Similarity-Based Recommendation Algorithm

Before starting a more complex and detailed algorithm, it is important to establish a baseline algorithm. Our baseline algorithm is a Demographic Filtering Recommender System which focuses on **user demographics to suggest user similarity** in tackling the cold user problem. The main idea is to utilize information from the user's interaction history (non-cold users) to suggest the most relevant cards to users with no prior interactions (cold users). To achieve this, the user data will be transformed into a vector space, enabling comparisons between users and the identification of similar profiles.

For this algorithm, we have assumed no interaction data is available for the defined cold users. Therefore, the only data available to describe these users is their profile information, e.g., age, gender, and location.

6.1 Feature Engineering

The dataset does not contain any explicit user ratings for card interactions, as mostly found in other datasets, as input for developing personalized recommendation systems. To deal with this restriction, we created an induced scoring mechanism that assigns a proxy score to every such user-card interaction. It essentially takes interaction metadata into account for defining weights based on importance and frequency of interaction types. This induced score can bridge the void created by the absence of explicit ratings in the meaningful quantitative representation of user preferences. The sections below provide the underlying mathematical formulation, rationale behind weight selection, and importance of this mechanism.

The Weighted Interaction Score (S_{weighted}) is calculated as follows:

$$S_{\text{weighted}} = C \cdot w_C + M \cdot w_M + H \cdot w_H + S \cdot w_S + V \cdot w_V + L \cdot w_L + T \cdot w_T$$

Where: S_{weighted} = Weighted Interaction Score, summarizing the user's level of engagement;
C = Binary indicator for "isClick" (0 or 1); M = Binary indicator for "isComment" (0 or 1);
H = Binary indicator for "isIntoPersonalHomepage" (0 or 1); S = Binary indicator for "isShare" (0 or 1); V = Binary indicator for "isViewComment" (0 or 1); L = Binary indicator for "isLike" (0 or 1); T = Normalized "mlogViewTime."

The associated weights (w) are defined as follows:

$$w_C = 1, w_M = 5, w_H = 2, w_S = 4, w_V = 2, w_L = 3, w_T = 1$$

We formulate a positive interaction as being an interaction where:

$$S_{\text{weighted}} \geq 1$$

Positive interactions help filter engaged users, ensuring a sufficient dataset for testing the algorithm. By requiring at least one positive interaction, the evaluation avoids pessimistic results caused by an insufficient item pool. The scoring mechanism compensates for the lack of explicit ratings by aggregating interaction metadata and normalized features, balancing precision, scale, and personalization. This weighted scoring enhances performance metrics and serves as a key factor for future system improvements.

6.2 Data Preprocessing

Similarities among the demographic data were computed through a series of transformations. This included changing different data types to a regular format which is understandable to a machine learning algorithm.

Numerical Attributes: The variables like “*registeredMonthCnt*” and “*level*” were normalized using min-max scaling. It converted the input to a range between 0 and 1 and did not let any attribute skew the similarity comparison. The importance of normalization particularly lies in distance-based methods as different scales of an attribute may interfere with the ratings.

The categorical attributes included the demographic variables like “*province*”, “*gender*”, and “*age_group*” in the one-hot encoding. Therefore, it represents every category with a binary vector, not allowing for the categorical nature of data. For example, a column for gender having values “male”, “female” and “unknown” will be converted into three binary columns making it possible for the numerical algorithms to process the features.

Post transformation, high-dimensional vector was formed for the demographic profile of each user. It includes normalized numerical features along with one hot encoded categorical features. These vectors formed the basis for the computation of similarity between users.

6.3 Similarity Index

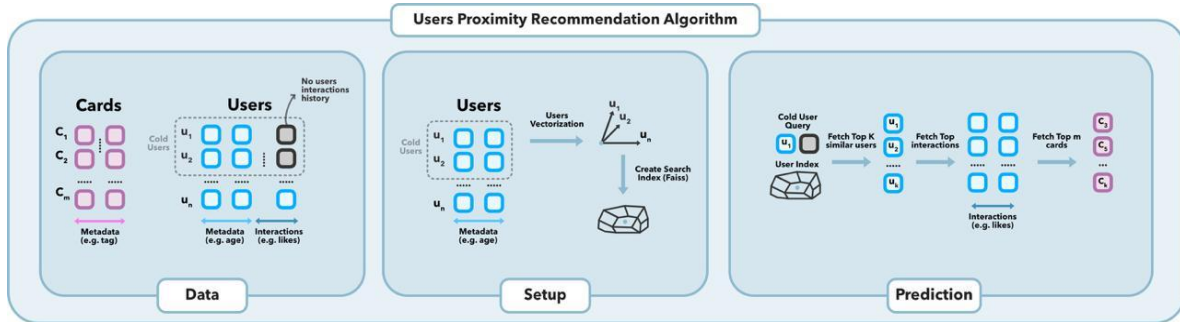


Figure 4 - User Similarity-Based Recommendation Algorithm Structure

The structure of the algorithm is illustrated in *Figure 4*, which will be explained and discussed in the following sections. Pre-processing the user-demographics table provides a vector representation. This vector is unique for each user and has all available information. The user similarity was calculated using the FAISS (Facebook AI Similarity Search) library (Douze, et al. 2024). Note that before using the FAISS library, we have used the cosine similarity that uses the cosine of the angle between the two vectors and despite this measure being very much used we opted for the FAISS library for faster search reasons, that is, FAISS library is an index mechanism which stores vectors in clusters and allows for fast search similar items retrieval using k-nearest neighbor retrieval. Besides, FAISS can perform an inner product index like cosine similarity if the vectors are normalized, since ours are normalized and this library is optimized for quickly finding the most similar users, even in large datasets we opt for this one instead of cosine similarity. After transforming the demographic information, it was used to create a similarity index through the application of FAISS:

1. **Vector preparation:** Each user's demographic vector is normalized to unit length; hence, the comparison is not dependent on any scale differences. It assures that all

similarity measures, based on the assumption that measures are based on angles between vectors (cosine similarity) do not be considered for their magnitudes.

2. **Index construction:** Different indexing methods are available for FAISS, each optimized for specific similarity computation. In this study, however, the use of the inner product index was preferred; because, when applied to the normalized vectors, it computes cosine similarity-which is the most popular measure in recommender systems due to its simplicity and computational efficiency.
3. **Index Population:** The normalized user vectors were put into the FAISS index to create a searchable structure. This index allows fast retrieval of the k nearest neighbors for any user query.

Speed and scalability enabled FAISS to surpass traditional similarity measures like brute-force cosine similarity. This becomes beneficial especially with large datasets as the present research indicates.

6.4 Recommendation based on similarity

For each cold-start user, the system identifies the k most similar non-cold users. As stated before, these are users with existing interaction histories, as their behavioral data provides valuable insights into which items (cards) are most relevant.

Post, the similar users are identified, the system generates recommendations by exploiting the past interactions of these similar users. For each similar user, the system selects the most relevant items based on a weighting metric (e.g., interaction frequency or engagement level). This weighting considered both the frequency and weight of the user and similar users interacting with that item. Selecting the top n items thus comprised recommendations that

are relevant yet diverse. In cases where no sufficiently similar users could be identified, globally popular items were recommended as a fallback.

6.5 Recommendation Evaluation

The system's performance is evaluated by comparing recommended items with actual user interactions using standard recommendation metrics. Precision@k measures the proportion of relevant recommendations, while Recall@k assesses how many relevant items were effectively suggested. Hit Rate@k indicates whether at least one relevant item appeared in the recommendations. MRR@k evaluates the ranking of the first relevant item, giving higher weight to top-ranked suggestions, and nDCG@k considers the overall ranking quality by prioritizing items placed higher. The evaluation uses k values of 1, 3, 5, and 10, reflecting practical interface constraints where recommendations typically range between one and ten items per screen.

6.6 Results and Interpretation

The assessment of the recommendation system uncovered highly enlightening findings about its efficacy for typical users, as well as cold-start ones, using measures such as Precision, Recall, HitRate, MRR, and nDCG. This section critically appraises the results (*Table 2* and *Table 3*), outlining some implications and issues raised by the key findings.

Metric - Normal Users	k=1	k=3	k=5	k=10
Precision	0.0306	0.023	0.022	0.0173
Recall	0.0222	0.0458	0.071	0.1076
HitRate	0.0306	0.0679	0.106	0.1629
MRR	0.0306	0.0455	0.0542	0.062
nDCG	0.0306	0.04	0.0508	0.0636

Table 2 - Results for normal users

Metric - cold users	k=1	k=3	k=5	k=10
Precision	0.0227	0.0303	0.0227	0.0227
Recall	0.0227	0.0909	0.1136	0.2273
HitRate	0.0227	0.0909	0.1136	0.2273
MRR	0.0227	0.053	0.0576	0.0735
nDCG	0.0227	0.0628	0.0716	0.1091

Table 3 - Results for cold user

For k=1, the system's Precision across all users is about 3.06%, indicating that the top-ranked recommendation often holds some immediate value. As k increases, Recall improves significantly (over 10% at k=10), showing that a broader recommendation list captures more of the user's interests. However, this expansion naturally lowers Precision, and while nDCG and MRR improve at larger k values, relevant items are not always ranked optimally without behavioral signals.

For cold-start users, the system achieves a modest Precision of 2.27% at k=1, demonstrating that demographic-driven logic and popularity fallbacks provide initial relevance. Increasing k (e.g., k=3 or k=10) significantly enhances Recall, showing that demographic similarity uncovers relevant items for new users. However, MRR and nDCG suggest that while relevant items appear in recommendations, they are not always prioritized effectively. This highlights

the system's limitation: demographic attributes provide minimally relevant recommendations but cannot match the personalization achieved through behavioral data.

Despite these constraints, the system reduces randomness and ensures cold-start users receive somewhat relevant content, especially at higher k values. However, increasing top- k recommendations expands coverage while still lacking Precision, emphasizing the need for more sophisticated modeling. The system also faces computational challenges due to dataset size (over 5 million records), limiting its ability to refine relevance further.

Ultimately, this baseline demonstrates that new users can receive personalized recommendations despite the absence of prior interactions. As k increases, Precision improves, but the low top- k ranking quality suggests the necessity of integrating machine learning-based weighting, hybrid models, or behavioral signals to enhance personalization and item prioritization.

6.7 Summary

The demographic similarity-based recommendation algorithm effectively addresses cold-start problems by generating initial recommendations based on user demographics. Under $k=1$, it provides some relevant suggestions, and expanding to $k=3$, $k=5$, or $k=10$ improves recall, demonstrating its effectiveness in capturing relevant items within demographic clusters. However, moderate MRR and nDCG scores highlight its limitation in ranking quality due to the absence of behavioral signals.

For users with interaction histories, the algorithm achieves modest precision and higher recall at larger k values but struggles with personalized top-ranked recommendations. This

reinforces its key limitation: demographic attributes alone cannot replace user behavior signals.

Despite these drawbacks, the research supports demographic-based approaches, showing promise for cold-start scenarios. Integrating demographic insights with behavioral data offers a clear path for improved personalization. As a scalable baseline, this method provides a foundation for future advancements while balancing recommendation quality and computational efficiency in dynamic environments.

7. Conclusion and Discussion

7.1 Key Findings

This study explores how machine learning and data-driven methods can enhance recommendation systems for cold-start users on NetEase Cloud Music, where limited interaction history makes personalization challenging. Four models were designed to tackle this issue: a similarity-based algorithm leveraging demographic attributes, a metadata-driven filtering system interpreting early interactions, a deep learning model integrating side information embeddings, and a clustering-based approach identifying shared behavioral patterns. Each model has distinct strengths and weaknesses, making them suitable for different recommendation scenarios.

The User Similarity-Based Algorithm generates recommendations using FAISS-based searches on demographic data such as province, gender, and age group. While effective for new users, it lacks adaptability to evolving preferences, requiring further refinement in dynamic weighting. The User Preference Elicitation Model employs metadata-driven filtering with predefined weights to infer preferences from interactions like clicks and likes.

Though efficient, its static nature limits diversity, suggesting the need for adaptive machine learning techniques. The DeCS-Inspired System leverages deep learning and structured embeddings to enhance cold-start recommendations. The inclusion of Singular Value Decomposition (SVD) boosts efficiency, yet ranking precision remains an area for improvement. The Frequent Item Set Model clusters users based on co-occurrence patterns, offering scalability but favoring popularity-based recommendations over personalization. While each model has trade-offs, combining demographic initialization with deep-learning refinement presents a balanced approach to improving recommendation efficiency and adaptability.

Metric	k	User Similarity-Based	Users Preference Elicitation	DeCS-Inspired	Discriminant Frequent Item-sets
Precision	1	2.27	0.49	2.15	0.66
	3	3.03	0.52	2.49	0.79
	5	2.27	0.47	5.23	0.68
	10	2.27	0.49	23.44	0.43
Recall	1	2.27	0.02	1.97	0.38
	3	9.09	0.08	2.87	1.36
	5	11.36	0.12	5.5	1.96
	10	22.73	0.12	28.75	2.45
HitRate	1	2.27	0.49	2.15	0.17
	3	9.09	1.36	3.23	0.55
	5	11.36	2.14	6.45	0.69
	10	22.73	2.14	32.56	0.71
MRR	1	2.27	0.49	2.15	0.17
	3	5.3	0.83	2.51	0.33
	5	5.76	0.98	3.21	0.37
	10	7.35	0.98	13.66	0.37
nDCG	1	2.27	0.49	2.15	0.17
	3	6.28	0.50	2.48	0.24
	5	7.16	0.46	4.02	0.25
	10	10.91	0.30	15.81	0.23

Table 4 - Comparison of Recommendation Models Across Evaluation Metrics

To evaluate and compare these systems, we conducted a comprehensive analysis using **five recommendation evaluation metrics per model**, which are now consolidated into a single representation for clarity and comparison (*Table 4*).

Performance evaluation was conducted using key metrics, including Precision@K, Recall@K, Mean Reciprocal Rank (MRR), Normalized Discounted Cumulative Gain (nDCG), and Hit Rate. The User Similarity-Based Algorithm demonstrated high precision and strong recall, making it effective for cold-start users but less responsive to changes in user preferences. The User Preference Elicitation Model exhibited low precision but reasonable recall, which resulted in redundancy and weaker ranking quality. The DeCS-Inspired System outperformed the others in ranking capability, achieving the highest nDCG and MRR scores but requiring substantial computational power. The Frequent Item Set Model, while emphasizing interpretability, struggled with lower precision and recall, limiting its effectiveness for personalization.

The suitability of each model depends on the specific use case. The User Similarity-Based Algorithm and the User Preference Elicitation Model are best suited for cold-start scenarios, leveraging demographic and metadata information to generate recommendations. The DeCS-Inspired System excels in long-term personalization, providing superior ranking and recall for users with historical interactions. The Frequent Item Set Model is ideal for platforms prioritizing scalability and interpretability, though it lacks strong ranking optimization. A hybrid approach, combining demographic-based initialization with deep-learning refinement, presents a balanced solution that enhances efficiency, personalization, and adaptability.

7.2 Limitations

Despite their strengths, each model presents limitations that impact their scalability and adaptability. The User Similarity-Based Algorithm is constrained by its reliance on static

attributes, making it less effective in dynamic recommendation environments. The User Preference Elicitation Model produces repetitive recommendations, reducing content diversity and limiting user engagement. The DeCS-Inspired System, while highly effective in ranking, requires significant computational resources, making real-time implementation challenging. The Frequent Item Set Model, though easy to interpret, sacrifices precision and recall, reducing its ability to deliver highly personalized recommendations. A common challenge across all models is the reliance on offline evaluation, which prevents real-time adaptation to user behavior and market trends. Computational constraints also hindered deeper exploration of dynamic weighting mechanisms, continuous updates, and broader dataset integration. Addressing these challenges is essential for developing more scalable and adaptive recommendation methodologies.

7.3 Research Contributions

This study contributes to the business and technological advancement of recommender systems by demonstrating how integrating multiple data sources enhances cold-start recommendations, ultimately improving user retention and monetization strategies. It provides practical insights for businesses looking to optimize customer engagement through data-driven personalization techniques. The research introduces metadata-driven similarity-based recommendation methods that allow platforms to better understand and serve new users, while a clustering-based approach ensures segmentation strategies align with user preferences. Additionally, the deep learning-based model leveraging embeddings enhances long-term personalization, helping businesses maximize customer lifetime value. The interaction-weighting framework, which prioritizes the rarity and frequency of interactions, refines user targeting and ensures more accurate recommendations, increasing conversion

rates and content discoverability. Establishing a standardized evaluation framework facilitates systematic performance comparisons, allowing companies to measure the effectiveness of their recommendation strategies and refine them accordingly. By addressing the cold-start problem with diverse methodological approaches, this research lays a foundation for businesses to create scalable, adaptive, and financially sustainable recommendation systems that drive growth and competitive advantage in digital markets.

7.4 Implications for Practice

Effectively addressing the cold-start problem is crucial for enhancing user engagement and long-term retention. Platforms such as NetEase Cloud Music must leverage demographic and contextual data to improve initial recommendations, creating a seamless and engaging user experience. In addition to improving onboarding experiences, effective cold-start solutions ensure that evolving user preferences are continuously incorporated into recommendation strategies. These solutions also facilitate the discovery of emerging artists by promoting new content in personalized recommendation pools, supporting content diversification and platform inclusivity. From a business perspective, overcoming cold-start challenges enables platforms to differentiate themselves in competitive markets, leading to increased user retention and higher revenue through improved content discoverability. By integrating algorithmic efficiency with a user-centric design, platforms can optimize both engagement and monetization, ensuring long-term sustainability.

7.5 Future Research Directions

Future research should focus on enhancing data strategies, advancing model development, and addressing privacy and security concerns. Expanding metadata, such as song attributes

and creator profiles, will improve clustering and personalization, while extending data collection beyond a single month will allow for a more comprehensive temporal analysis of user behavior. Developing online recommendation systems capable of real-time updates is essential to overcoming current computational constraints. Further refinement of the DeCS-Inspired Model can improve ranking precision and efficiency, making it more suitable for large-scale deployments. Ensuring fairness and security in recommendation systems remains a critical challenge, requiring robust approaches to mitigate adversarial attacks and protect user data. Increasing transparency and allowing users to control their recommendations will enhance trust and engagement, ensuring ethical AI implementation. Advancing personalization, adaptability, and computational efficiency will bridge the gap between cold-start challenges and long-term user satisfaction, fostering more meaningful digital experiences and driving sustainable business growth.

8. References

- s.d. “AI in Music Streaming: Personalization at its Best.” *AI in Music Streaming: Personalization at its Best*.
- s.d. “AI-based Recommendation System Market Report.” *AI-based Recommendation System Market Report*.
- Bobadilla, Jesús, Fernando Ortega, Antonio Hernando, e Jesús Bernal. 2012. “A collaborative filtering approach to mitigate the new user cold start problem.” *Knowledge-based systems* (Elsevier) 26: 225–238.
- Chadha, Aman, e Vinija Jain. 2022. “Recommender Systems Primer - Cold Start.” *Recommender Systems Primer - Cold Start*. www.vinija.ai.
- Chakravorti, Bhaskar. 2018. “Competing in the Huge Digital Economies of China and India.” *Harvard Business Review* 6.
- Douze, Matthijs, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, e Hervé Jégou. 2024. “The Faiss library.”
- Elahi, Mehdi, Farshad Bakhshandegan Moghaddam, Reza Hosseini, Mohammad Hossein Rimaz, Nabil El Ioini, Marko Tkalcic, Christoph Trattner, e Tammam Tillo. 2021. “Recommending videos in cold start with automatic visual tags.” *Adjunct Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*. 54–60.
- Feng, Philip J., Pingjun Pan, Tingting Zhou, Hongxiang Chen, e Chuanjiang Luo. 2021. “Zero shot on the cold-start problem: Model-agnostic interest learning for recommender systems.” *Proceedings of the 30th ACM international conference on information & knowledge management*. 474–483.
- Hafnar, Davor, e Jure Demšar. 2024. “Zero-Shot Reasoning: Personalized Content Generation Without the Cold Start Problem.” *arXiv preprint arXiv:2402.10133*.

- Han, Cuize, Pablo Castells, Parth Gupta, Xu Xu, e Vamsi Salaka. 2022. “Addressing cold start in product search via empirical bayes.” *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3141–3151.
- s.d. “How Retailers Can Keep Up with Consumers.” *How Retailers Can Keep Up with Consumers*.
- Karabila, Ikram, Nossayba Darraz, Anas El-Ansari, Nabil Alami, e Mostafa El Mallahi. 2024. “Addressing data sparsity and cold-start challenges in recommender systems using advanced deep learning and self-supervised learning techniques.” *Journal of Experimental & Theoretical Artificial Intelligence* (Taylor & Francis) 1–31.
- Kowald, Dominik, Markus Schedl, e Elisabeth Lex. 2020. “The unfairness of popularity bias in music recommendation: A reproducibility study.” *Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II 42*. 35–42.
- Kumar, Pushpendra, e Ramjeevan Singh Thakur. 2018. “Recommendation system techniques and related issues: a survey.” *International Journal of Information Technology* (Springer) 10: 495–501.
- Lam, Xuan Nhat, Thuc Vu, Trong Duc Le, e Anh Duc Duong. 2008. “Addressing cold-start problem in recommendation systems.” *Proceedings of the 2nd international conference on Ubiquitous information management and communication*. 208–211.
- Li, Jingjing, Ke Lu, Zi Huang, e Heng Tao Shen. 2017. “Two birds one stone: on both cold-start and long-tail recommendation.” *Proceedings of the 25th ACM international conference on Multimedia*. 898–906.
- Lika, Blerina, Kostas Kolomvatsos, e Stathes Hadjiefthymiades. 2014. “Facing the cold start problem in recommender systems.” *Expert systems with applications* (Elsevier) 41: 2065–2073.
- Lin, Xixun, Jia Wu, Chuan Zhou, Shirui Pan, Yanan Cao, e Bin Wang. 2021. “Task-adaptive neural process for user cold-start recommendation.” *Proceedings of the Web Conference 2021*. 1306–1316.

- Liu, Haifeng, Zheng Hu, Ahmad Mian, Hui Tian, e Xuzhen Zhu. 2014. “A new user similarity model to improve the accuracy of collaborative filtering.” *Knowledge-based systems* (Elsevier) 56: 156–166.
- Melville, Prem, e Vikas Sindhwani. 2010. “Recommender systems.” *Encyclopedia of machine learning* 1: 829–838.
- s.d. “Music Streaming - China | Statista Market Forecast.” *Music Streaming - China | Statista Market Forecast*.
- s.d. “Music Streaming - Worldwide | Statista Market Forecast.” *Music Streaming - Worldwide | Statista Market Forecast*.
- Nguyen, An-Te, Nathalie Denos, e Catherine Berrut. 2007. “Improving new user recommendations with rule-based induction on cold user data.” *Proceedings of the 2007 ACM conference on Recommender systems*. 121–128.
- Pandey, Anand Kishor, e Dharmveer Singh Rajpoot. 2016. “Resolving Cold Start problem in recommendation system using demographic approach.” *2016 International Conference on Signal Processing and Communication (ICSC)*. 213-218.
doi:10.1109/ICSPCom.2016.7980578.
- Panteli, Antiopi, e Basilis Boutsinas. 2023. “Addressing the cold-start problem in recommender systems based on frequent patterns.” *Algorithms* (MDPI) 16: 182.
- Pu, Pearl, e Li Chen. 2008. “User-involved preference elicitation for product search and recommender systems.” *AI magazine* 29: 93–93.
- Roy, Deepjyoti, e Mala Dutta. 2022. “A systematic review and research perspective on recommender systems.” *Journal of Big Data* (Springer) 9: 59.
- Silva, Nícollas, Diego Carvalho, Adriano C. M. Pereira, Fernando Mourão, e Leonardo Rocha. 2019. “The pure cold-start problem: A deep study about how to conquer first-time users in recommendations domains.” *Information Systems* (Elsevier) 80: 1–12.

- Singh, Neetu, e Sandeep Kumar Singh. 2024. “A systematic literature review of solutions for cold start problem.” *International Journal of System Assurance Engineering and Management* (Springer) 1–35.
- Son, Le Hoang. 2016. “Dealing with the new user cold-start problem in recommender systems: A comparative review.” *Information Systems* (PERGAMON-ELSEVIER SCIENCE LTD THE BOULEVARD, LANGFORD LANE, KIDLINGTON ...) 58: 87–104.
- Song, Yading, Simon Dixon, e Marcus Pearce. 2012. “A survey of music recommendation systems and future perspectives.” *9th international symposium on computer music modeling and retrieval*. 395–410.
- Steck, Harald, Linas Baltrunas, Ehtsham Elahi, Dawen Liang, Yves Raimond, e Justin Basilico. 2021. “Deep learning for recommender systems: A Netflix case study.” *AI Magazine* 42: 7–18.
- Sun, Anchen, e Yuanzhe Peng. 2022. “A survey on modern recommendation system based on big data.” *arXiv e-prints* arXiv–2206.
- Tahmasebi, Faryad, Majid Meghdadi, Sajad Ahmadian, e Khashayar Valiollahi. 2021. “A hybrid recommendation system based on profile expansion technique to alleviate cold start problem.” *Multimedia Tools and Applications* (Springer) 80: 2339–2354.
- Vozalis, Emmanouil, e Konstantinos G. Margaritis. 2003. “Analysis of recommender systems algorithms.” *The 6th Hellenic European Conference on Computer Mathematics & its Applications*. 732–745.
- Wang, Chunyang, Yanmin Zhu, Haobing Liu, Tianzi Zang, Jiadi Yu, e Feilong Tang. 2022. “Deep meta-learning in recommendation systems: A survey.” *arXiv preprint arXiv:2206.04415*.
- Wang, Shixiong, Wei Dai, e Geoffrey Ye Li. 2024. “Distributionally Robust Outlier-Aware Receive Beamforming.” *2024 IEEE 34th International Workshop on Machine Learning for Signal Processing (MLSP)*. 1–6.

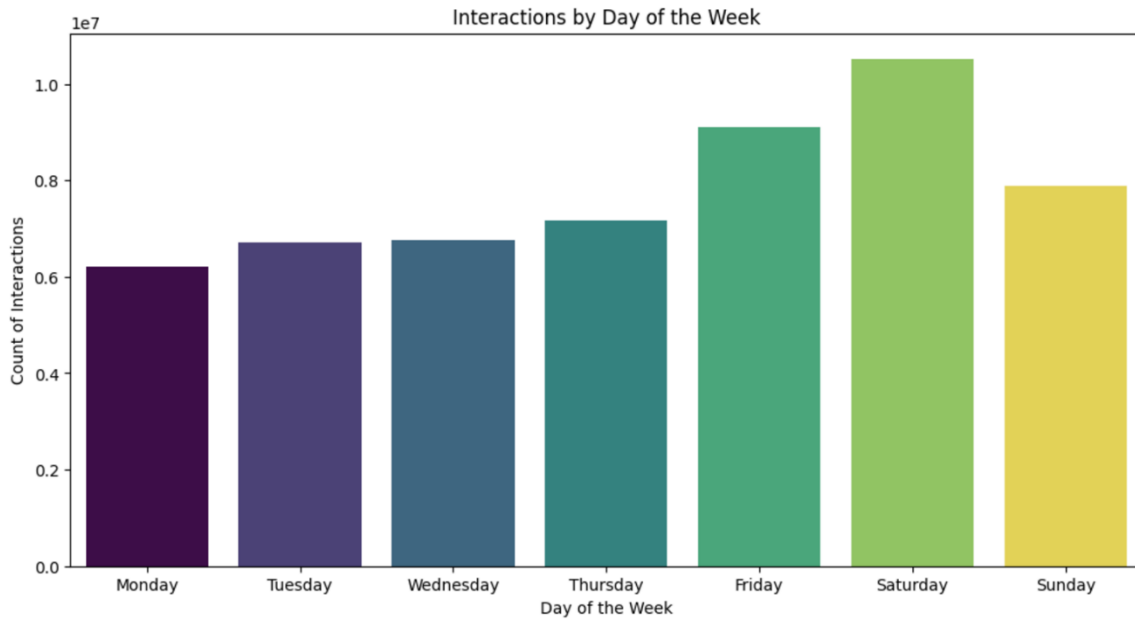
s.d. “What Netflix's Recommendation Systems Can Teach Us About The Computing Challenges Of The Near Future.” *What Netflix's Recommendation Systems Can Teach Us About The Computing Challenges Of The Near Future*.

<https://www.forbes.com/councils/forbestechcouncil/2021/02/19/what-netflixs-recommendation-systems-can-teach-us-about-the-computing-challenges-of-the-near-future/>.

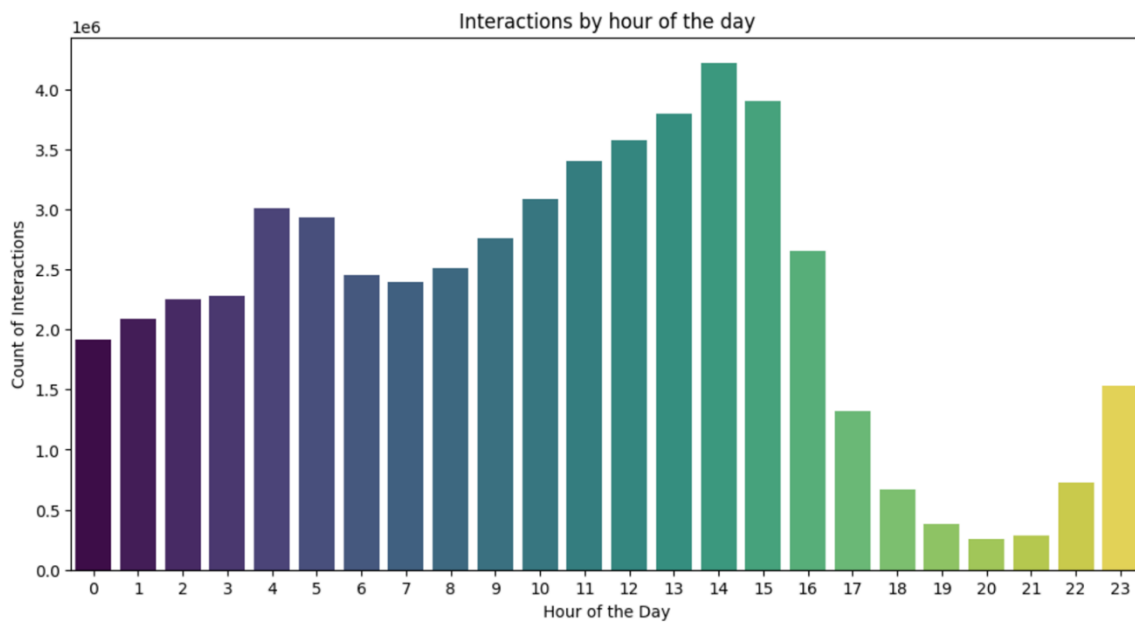
Zhang, Dennis J., Ming Hu, Xiaofei Liu, Yuxiang Wu, e Yong Li. 2022. “NetEase cloud music data.” *Manufacturing & Service Operations Management (INFORMS)* 24: 275–284.

Zhu, Ziwei, Shahin Sefati, Parsa Saadatpanah, e James Caverlee. 2020. “Recommendation for new users and new items via randomized training and mixture-of-experts transformation.” *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1121–1130.

9. Appendix



Appendix 1 - Interactions by day of week



Appendix 2 - Interactions by hour of the day