International Journal of Mathematical, Engineering and Management Sciences
Vol. 11, No. 1, 439-471, 2026
https://doi.org/10.33889/IJMEMS.2026.11.1.019

Evaluating the Long Tail in Recommendation System: A Systematic Review of Approaches, Datasets and Metrics

Santosh Saha
Department of Computer Science and Engineering,
Sikkim Manipal Institute of Technology, Majhitar,
Sikkim Manipal University, 737136, Gangtok, Sikkim, India.
E-mail: santosh_20211064@smit.smu.edu.in

Amrita Biswas
Department of Computer Science and Engineering,
Sikkim Manipal Institute of Technology, Majhitar,
Sikkim Manipal University, 737136, Gangtok, Sikkim, India.
Corresponding author: amrita.a@smit.smu.edu.in

Saumya Das
Department of Computer Science and Engineering – AI,
Brainware University, 700125, Barasat, Kolkata, India.
E-mail: saumya.das.1976@gmail.com

Nitai Paitya
Amity Institute of Information Technology,
Amity University, 700135, Kolkata, India.
E-mail: drnitaipaitya@gmail.com

(Received on July 8, 2025; Revised on October 29, 2025; Accepted on December 10, 2025)

Abstract
A recommendation system (RS) leverages machine learning by analyzing user behavior, and suggesting relevant products based on the user's preferences. Long-tail items, which were once leading products in their niche, became harder to find and newer items are heavily promoted to users, long-tail items, which can boost customer engagement and ensure that long-tail items remain visible. In this paper, we have provided extensive efforts to conduct a systematic review of the long-tail recommendation system, based on PRISMA 2020 guidelines. Studies published between 2012 and 2024 were identified, through a detailed search in ACM Digital Library, Science Direct, SpringerLink, IEEE Xplore, and Google Scholar. We conducted a detailed investigation into a long-tail recommendation system which focused on finding different categories, datasets, and evaluation metrics. This literature review provides an extensive overview of the selection of datasets, different categories of long-tail recommendation, and evaluation criteria for the researchers and individuals who are new to the domain of long-tail recommendation systems.

Keywords- Recommendation system, Long-tail, Long-tail recommender system, Artificial intelligence, Systematic literature review

1. Introduction

In the old days people used to rely heavily on recommendations that used to come from their parents, friends, and relatives. In today's world as internet is growing, more and more new discoveries taking place every day and new products or items are being introduced, so it is impossible for a person to keep up to date with everything happening. RS are online tools that are used to recommend products or items to the user. Back in 1992 RS was coined by Goldberg et al. (1992) with the collaborative filtering (CF) the concept of RS was introduced and since then numerous research work has been conducted in the field of RS. Users can use an RS to make better decisions about products, listen to music of their choice, or even read the news rather than searching the entire catalog.

The main aim of the RS is to provide the best product or item to the users, and this may be a movie that the user may like based on customization, the aim is to provide a useful product or item to potential customers. The RS can function as an assistant, recommending products based on the user's preference or the categories that the user visits. Consider online shopping sites, where the system recommends items based on what the user has previously purchased, or the product user is likely to buy based on their past purchased history. Since users of the RS rely on the assistance provided, the recommended system should be accurate and reliable, with the ability to predict user preferences as accurately as possible. The RS helps to establish stronger relationships with the customers and address the issue of information overload. Based on the user history RS, it makes suitable product recommendations to customers based on their interaction, interest, and profile information.

Traditional recommendation techniques can be either content-based, CF, or hybrid recommendation techniques. In content-based filtering (CB) items are recommended to the users who have previously preferred them based on the user's past historical data and suggests items that exhibit similar features (Lops et al., 2011). Content-based filtering (CB) approach primarily focuses on closely related items, which results in limited exploration beyond user preferences. Content-based filtering (CB) suffers from a lack of diversity, as it rarely introduces novel or unexpected items to the users. In CF, the recommendation relies on the behavior and preferences of multiple users with similar interests to recommend items. The underlying principle is that if two users have similar preferences in the past, then it is likely that the users will enjoy similar items in the future (Sharma et al., 2017). CF can be further classified into memory based and model based. The Model-based CF technique uses machine learning and statistical techniques to uncover latent patterns within the data, whereas memory-based CF technique relies mostly on similarity measures between users and items to generate recommendation lists (Aditya et al., 2016). Collaborative filtering (CF) approach suffers from the cold-start problem for new users or items, as well as the sparsity of user-item interactions, which may lead to inconsistent preferences. Together, these approaches enable the system to identify and suggest previously undiscovered yet relevant items. Hybrid recommendation technique on the other hand merges two or more approaches to minimize their drawbacks to strengthen the shortcomings of both models (Burke, 2002). Hybrid recommendation systems often encounter problems related to model complexity, high computational cost, and parameter tuning, while simultaneously struggling to maintain an optimal balance among multiple objectives such as accuracy, diversity, and novelty.

Non-traditional recommendations include deep learning(DL) based recommendation techniques that use several layers to solve the recommendation problem using artificial neural networks to discover hidden patterns (Batmaz et al., 2019). Knowledge graph-based recommendation techniques use nodes to represent entities and the relations between entities are basically the relationship are represented by edges (Guo et al., 2022). The matrix factorization-based recommendation technique generates a user-to-item interaction matrix using latent factor analysis (Mehta & Rana, 2017). Reinforcement based recommendation techniques are dynamic. They learn from user feedback and interactions and dynamically adjust to balance the recommendation strategies between new and less popular items (Afsar et al., 2023). A context-aware recommendation technique uses user location, time, and tastes to suggest niche items based on demographic information (Kulkarni & Rodd, 2020).

In 2004, Chris Anderson first proposed the term "long-tail" in a wired magazine article (Anderson, 2006). Long-tail items are those items which many have relatively low demand but constitute a significant portion of overall demand. The idea of long-tail recommendations addresses the issue of recommended systems that try to promote niche or less popular items. In the conventional recommended system, items based on CF receive more ratings due to existing users' interactions with similar items. Long-tail items are those that may have fewer interactions with the top-level item, but when combined with their niche item, long-tail items may experience very high sales, even though they have fewer interactions with the top-level item.

The paper's main contributions are summarized as follows:

(i) This paper provides a brief description of the research on long-tail recommendation system.
(ii) More than seventy-one scholarly articles on long-tail recommendation systems, collected from the last thirteen years have been analyzed and critically reviewed.
(iii) There are several categories on which the articles were assessed: cluster, graph, deep learning, neural networks, multi-objective, and traditional RSs.
(iv) It will help researchers to identify most used datasets for long-tail recommendations.
(v) Organizing of evaluation criteria into different categories is one of the important aspects of the study.
(vi) A conclusion on long-tail recommendation systems that address challenges and future prospects.

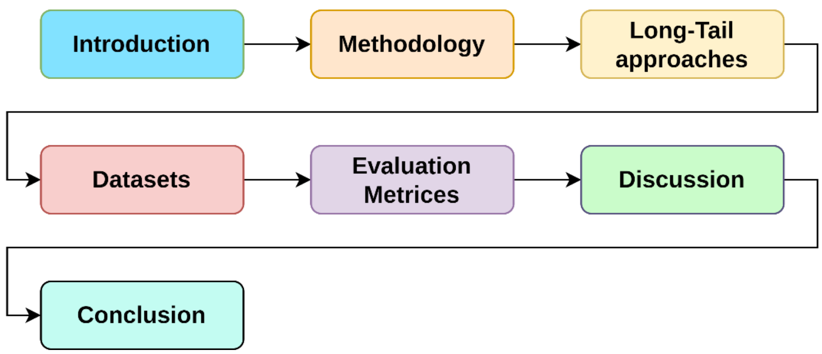

Figure 1. Logical flow of the paper.

The above **Figure 1** depicts the rest of the document's arrangement as a block diagram. Section 1 is used to provide the introduction of the paper and provide the motivation behind extensive study on long-tail recommendation system. Section 2 provides the methodology that is used to prepare the literature review dealing with the data collection and inclusion and exclusion criteria, materials and methods for conducting reviews. which further describes research questions and selection criteria. Section 3 provides different long-tail approaches and different categories of long-tail recommendation system. Section 4 talks about different methods used in long-tail recommendation system, datasets and. Section 5 deals with different evaluation methods. Section 6 will provide discussion and Section 7 will provide a conclusion and the future direction on long-tail recommendation system.

2. Methodology

This section describes the methodologies to review the work that has been done in long-tail recommendation system. The PRISMA 2000 is a preferred standardized reporting guideline designed to reflect methodological advancements in the identification, selection, appraisal, and synthesis of studies, to conduct transparent documentation of systematic literature reviews. To conduct a literature search on long-tail recommendation system we have reviewed articles published in the top reputed publishers such as ACM Digital Library, Science Direct, IEEE Xplore and Springer Link from the year 2012. **Figure 2** represents the year wise publication data and **Figure 3** represents the publisher wise literature percentage on long-Tail recommendation system. This review paper has undergone three major stages, planning, reviewing and finally the result.

2.1 Planning Stage

The planning stage defines the search strategies, based on the search strategies the published papers are reviewed. The purpose of the planning stage is to first formulate the research questions, keeping in mind the objectives of the research. A search string is developed and based on the search string the material has been collected from different databases. At this stage the collected materials are further reviewed based on the inclusion and exclusion. Finally, the collected materials are sorted based on relevance and the review will be conducted.

To meet the objective of this research the following research question has been considered:

- **RQ1:** What are the different approaches used in a long-tail recommendation system.
- **RQ2:** Which datasets are predominantly used in long-tail recommendation research and development?
- **RQ3:** What are the different evaluation criteria used for long-tail recommendation systems?

As a result of the research questions that were formulated and answered in this study, we were able to accomplish our contributions. Section 3 discusses different approaches to the long tail, which address RQ1. Section 4, which explored various dataset types, contained answers to RQ2. Section 5, which contained different types of evaluation criteria, provides the answer to RQ3. Each research question and its rationale are outlined in **Table 1**.

Table 1. Rationales of research question.

Research Question (RQ)	Rationale
RQ1: What are the different approaches used in a long-tail recommendation system.	This research aims to examine the different categories under which long-tail recommendations fall.
RQ2: Which datasets are predominantly used in long-tail recommendation research and development?	This research question aims to focus on the various dataset types that are frequently used in long-tail recommendation algorithms.
RQ3: What are the different evaluation criteria used for long-tail recommendation systems?	The aim of this research question is to identify the various evaluation criteria that are applied in long-tail recommendation systems.

2.2 Review Stage

After formulating the research questions and the objectives of the paper, keywords have been identified, and these keywords are used to search online scientific databases. Both the primary search string and modified search string combine Boolean ANDs and ORs to search for the keywords in title, abstract, and body of paper. The repositories contain papers published in the top journals and conferences.

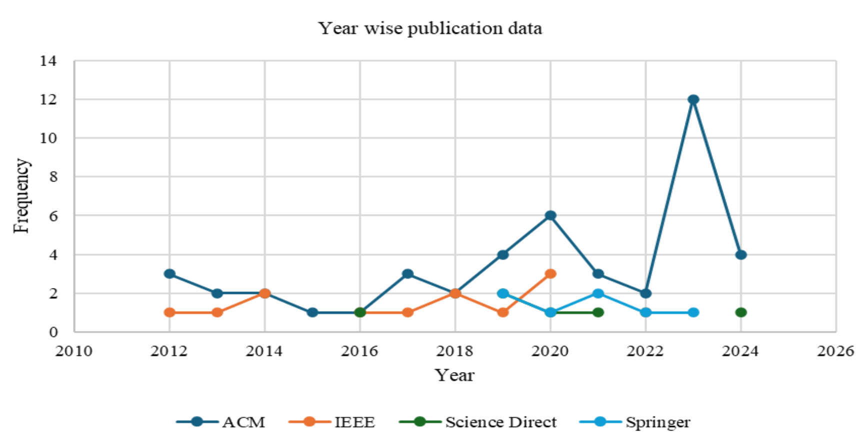

Figure 2. Year wise long-tail recommendation system publication details.

As a search string we have used keywords such as "Long-Tail", "Long-Tail Recommendation System", "Long-Tail Recommender System”. The primary search result was modified, based on time-based sorting. We've collected data in repositories from leading journals and conferences over the period of the past 12 years. Titled, abstracts, and conclusions based on keywords were manually reviewed and checked for the downloaded papers.

The screening and selection of the included articles were conducted systematically and adhered to the methodological standards outlined in the PRISMA 2020 (Page et al., 2021) guidelines. A detailed flow chart describing the selection process which includes identification, screening and inclusion is provided in **Figure 4**.

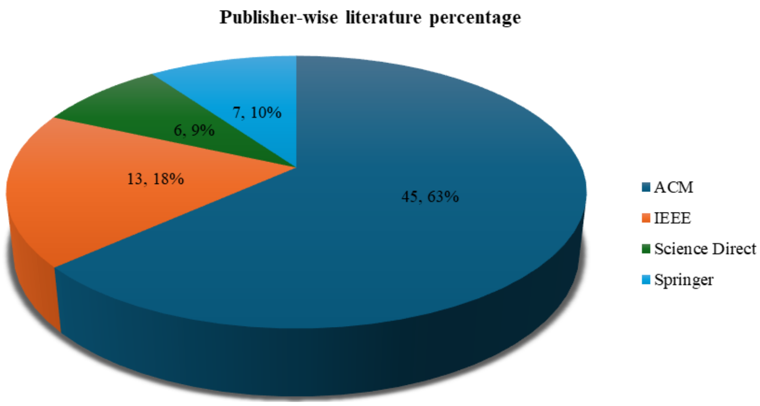

Figure 3. Publisher literature percentage of long-tail recommendation system articles.

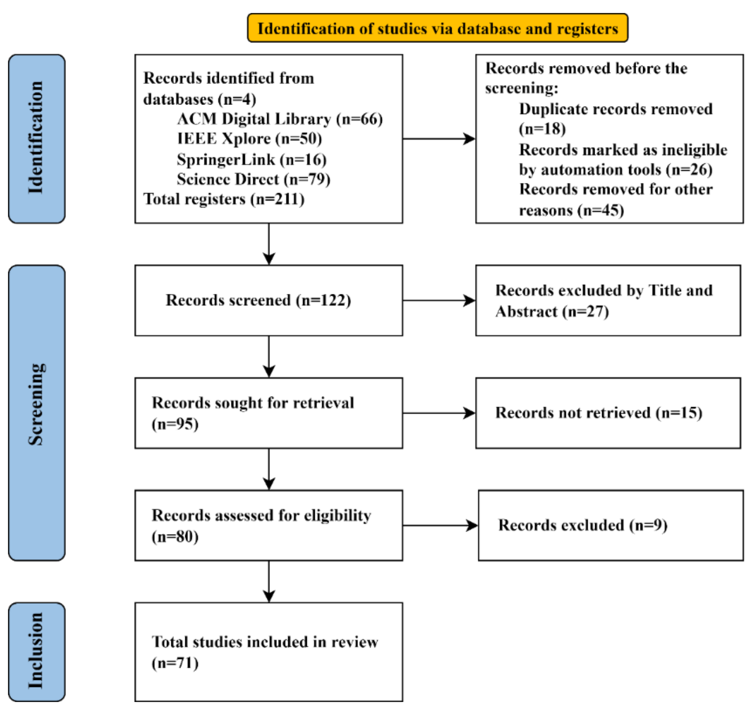

Figure 4. PRISMA 2020 workflow for SLR on long-tail recommendation system.

2.2.1 Identification (Search Online Databases)

We have obtained 211 articles, out of which 66 articles were from ACM Digital Library, 50 articles were from IEEE Xplore, 16 articles are from Springer Link and 79 articles from Science Direct. We use keywords for searching the databases with 'AND' or 'OR'. The searches that are used are as follows (long tail) AND (long tail recommender system) OR (Long-Tail Recommendation System) AND (collaborative filtering long tail) OR (content based long tail) AND (Hybrid long tail) OR (Multi Objective) OR (Graph) OR (Cluster).

2.2.2 Screening

After removing 18 duplicates articles, 26 articles were ineligible by the automation tool and 45 for other reasons. We also removed papers that did not have keywords like Long-Tail. At the end of the first phase of screening we retained 122 articles. After retaining 122 articles, nearly 27 articles records were excluded as the Title and Abstract criteria didn't match. Some of the records were not retrieved by nearly 15articles, then 14articles more record was excluded for other reasons as it does not match the eligible criteria. Finally, 71 articles were retained to be used for SLR.

2.3 Result Stage

The final SLR contains 71 papers from “journal articles” (JA) and “conference articles” (CA), which are sorted year wise from the year 2012 and selected based on their alignment with the research aim and questions. In **Table 2** we finalize the papers for the study.

Table 2. Selected papers for study.

PID	Year	Article	Method name	Document type
1	2012	Yin et al. (2012)	User-item based undirected weighted graph	JA
2	2012	Bonchi et al. (2012)	Center-piece subgraphs	CA
3	2012	Park & Han (2012)	Diversity-based recommendation strategy examining customer churn	CA
4	2012	Zhang et al. (2012)	Double-ranking strategy	CA
5	2013	Park (2013)	Adaptive clustering method	JA
6	2013	Shi (2013)	A novel cost flow concept based on a 1st order Markovian graph	CA
7	2013	Niemann & Wolpers (2013)	Context based CF	CA
8	2014	Hu et al. (2014)	Latent variable models	CA
9	2014	Ho et al. (2014)	Aggregate diversity enhancement	CA
10	2014	Hwang & Li (2014)	Economic model	CA
11	2014	Wang et al. (2014)	Cosine pattern-based recommendation	CA
12	2015	Seifert et al. (2015)	Collaborative-filtering	CA
13	2016	Huang et al. (2016)	Knowledge base query recommendation	CA
14	2016	Wang et al. (2016)	Multi-objective optimization	JA
15	2016	Luo & Xie (2016)	A probabilistic model	CA
16	2017	Johnson & Ng (2017a)	Tripartite graphs and Markov processes	CA
17	2017	Johnson & Ng (2017b)	Tripartite graphs based long-tail recommendations	CA
18	2017	Hu et al. (2017)	C-HMF, S-HMF	JA
19	2017	Li et al. (2017)	Handling of cold-start and long-tail recommendations	CA
20	2018	Wang et al. (2018b)	User’s experience based long tail recommendation	CA
21	2018	Luke et al. (2018)	Long-tail recommendation via extended tripartite graph modeling	CA
22	2018	Krishnan et al. (2018a)	Adversarial training for enhancing long-tail recommendations in neural CF	CA
23	2018	Krishnan et al. (2018)	Learning robust behavior representations in online platforms	CA
24	2019	Huang & Wu (2019)	Biterm topic model	JA
25	2019	Li et al. (2019)	Micro-video hashtag recommendation	CA
26	2019	Liu et al. (2019)	Real-time look-alike modeling with attention mechanisms	CA
27	2019	Tang et al. (2019)	Neural multi-temporal range mixture model (M3)	CA
28	2019	Meenakshi & Satpal (2019)	Long-tail web services using DL techniques	CA
29	2019	Hamedani & Kaedi (2019)	Long tail items through personalized diversification	JA
30	2019	Pang et al. (2019)	Weighted similarity mea sure based on NSGA-II	CA
31	2019	Agarwal et al. (2019)	Hybrid reranking framework in CF	CA

Table 2 continued..

32	2019	Garigliotti et al. (2019)	Generative probabilistic framework to rank contexts	CA
33	2020	Silva & Durão (2020)	Dynamic clustering and Markov chains	CA
34	2020	Niu et al. (2020)	Dual heterogeneous graph attention with GNNs	CA
35	2020	Silva et al. (2020)	Graph-based node similarity computation for users and items	JA
36	2020	Zhao et al. (2020)	Multi-latent representations	CA
37	2020	Yin et al. (2020)	Sequential modeling of long-tail user behavior	CA
38	2020	Liu & Zheng (2020)	Session-based recommendation TailNet, to improve long-tail recommendation performance	CA
39	2020	Jang et al. (2020)	Tail-item embedding for sequential recommendation	CA
40	2020	Sreepada & Patra (2020)	Long tail using few shots learning technique siamese networks	JA
41	2020	Qin (2021)	Neighborhood-based recommendation method	CA
42	2020	Pandey & Ankayarkanni (2020)	Random forest	CA
43	2020	Alshammari et al. (2020)	A switching multi-level recommender system	CA
44	2021	Lakshmi et al. (2021)	Adaptive correlation clustering-based recommender system	CA
45	2021	Achary & Patra (2021)	A Hybrid Graph-Driven Model for Long-Tail Items	CA
46	2021	Zhang et al. (2021)	Dual transfer learning framework	CA
47	2021	Ge et al. (2021)	Constrained Markov Decision Process (CMDP)	CA
48	2021	Wen et al. (2021)	knowledge-enhanced collaborative meta learner	CA
49	2021	Sreepada & Patra (2021)	Long tail Econophysics-inspired	JA
50	2022	Hu et al. (2022)	MASR (Memory Bank Augmented Long-tail Sequential Recommendation)	CA
51	2022	Mussi et al. (2022)	DynaLT (Dynamic pricing for the Long Tail)	CA
52	2022	Yalcin (2022)	Popularity-aware recommendation technique (PopHybrid)	CA
53	2023	Chen et al. (2023)	Session-based recommendation from calibration	JA
54	2023	Didi et al. (2023)	NSGA-II	CA
55	2023	Islam et al. (2023)	Maximum marginal sum of products (MMSP)	JA
56	2023	Yang et al. (2023)	Niche Walk Augmentation (NWA) and Tail Session Mixup (TSM)	CA
57	2023	Wei et al. (2023)	Meta graph learning	CA
58	2023	Zhao et al. (2023a)	Graph convolutional networks	CA
59	2023	Zhao et al. (2023b)	conversational recommender systems (CRS)	CA
60	2023	Papso (2023)	Product Universal Embedding Space (PUES)	CA
61	2023	Gong et al. (2023)	Full Index Deep Retrieval (FIDR)	CA
62	2023	Liu et al. (2023a)	LinRec linear attention mechanism	CA
63	2023	Kim et al. (2023)	MELT mutual enhancement of long-tailed	CA
64	2023	Ricci et al. (2023)	Meta-learning advisor networks	JA
65	2023	Liu et al. (2023b)	Co-occurrence embedding enhancement for Long-Tail (CoLT)	CA
66	2023	Zhang et al. (2023)	Enhancement for Long-Tail	CA
67	2024	Zhang et al. (2024)	Graph convolutional networks and Bayesian methods	JA
68	2024	Balasubramanian et al. (2024)	Unique sampling strategy to produce user interaction history	CA
69	2024	Lin et al. (2024)	User-item graph using multimodal similarity	CA
70	2024	Wu et al. (2024)	Integrates user-item collaborative method in LLM	CA
71	2024	Shafiloo et al. (2024)	Using users' dynamic to enhance the diversity of the items suggested	JA

3. Long-Tail Approaches

As we go through this section, we will examine the different approaches that can be used in long-tail recommendation systems illustrated in **Figure 5**.

RQ1. What are the different approaches used in a long-tail recommendation system.

In this section and sub-section of the systematic literature review we categorize the long-tail recommendation system into different categories based on the data retrieved in **Table 2**. Six different approaches to long-tail recommendation systems have been categorized, including cluster, graph, deep learning and neural networks, multi-objective, traditional recommendation, and other methods.

3.1 Cluster Based Approaches

In a cluster based long-tail recommendation system the user or item data are being clustered using similar items or similar user. Hu et al. (2022) used a new sequential recommendation framework, designed to predict the next item from users' history. The long-tail problem was solved by focusing on item recommendation using a novel "open-book" model, which combines memory banks with retriever-copy network. Five distinct datasets have been used to demonstrate the effectiveness of the cluster and centroid memory bank on the MASR model. The centroid-wise memory bank is represented with the formula.

$$mi = \frac{1}{|B|}\sum_{(b_j,y_j)\in B}\frac{b_j}{||b_j||} \tag{1}$$

Let $X = \{(b_0, y_0) \dots, (b_{|x|}, y_{|x|})\}$ denote a memory bank, where b_i is the feature and y_i represents label. X is updated by inserting pair $((b_i, y_i)$ during the training session. The author uses five different datasets to test the ecommerce application and evaluated using the hit ratio and nDCG. Mussi et al. (2022) used a novel online algorithm for dynamic pricing long-tail products using artificial intelligence. The online learning algorithm uses dynamic pricing that is used long-tail products that aggregate similar products using clustering algorithm which were evaluated in an offline synthetic setting and then on online for about two months, which increases the revenue for both long-tail and short-headed products. The author uses real world ecommerce website data to evaluate empirical regret.

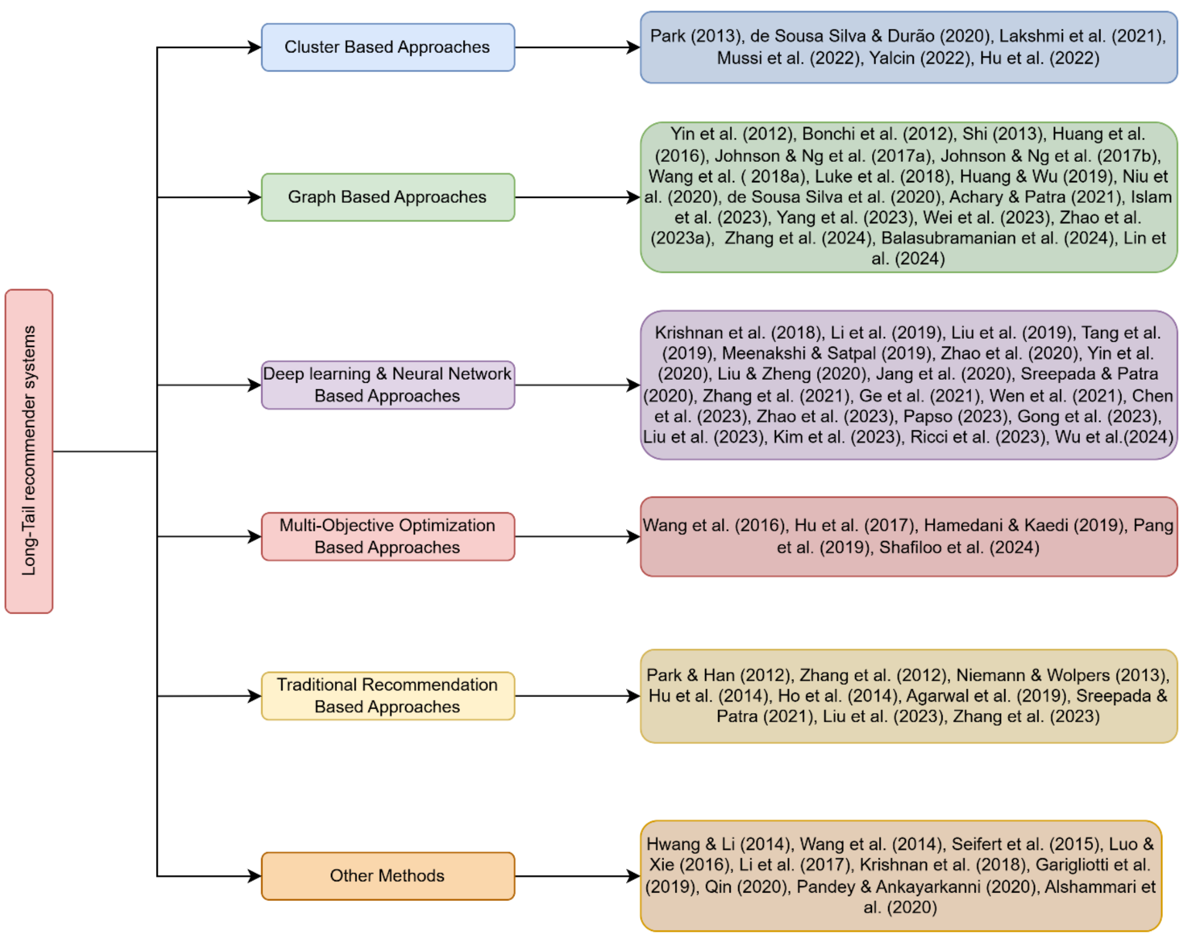

Figure 5. Systematic literature review diagram.

Yalcin (2022) proposes PopHybrid method to improve recommendation quality by combining multiple recommendation methods to optimize their shortcomings. PopHybrid method selects the less-biased method based on item popularity as a final recommendation model for the user. The system tests the model on two different datasets Movielens and Yahoo, using metrics Average popularity of the recommended items (APRI) and ration of popular item (RPI). Park (2013) proposed an adaptive clustering method which clusters user and items based on their dynamic behavior. This method enhances the less popular item by using diversity in RS. The author tries to solve the problem of e-commerce (E-comm) domain with two real world datasets Movielens and bookcrossing, and evaluated using mean absolute error and root mean square error which improve the visibility of less popular items. Lakshmi et al. (2021) proposed adaptive clustering which focuses on correlation of long-tail items. The author developed adaptive clustering which distinguishes popular items and less popular items. The long-tail items are clustered based on their similarities using correlation clustering. The model was tested on Movielens dataset with mean absolute error and root mean square error to evaluate the accuracy of rating prediction by including both head and the tail items. de Sousa Silva & Durão (2020) proposed a new algorithm based on Markov chain method that organizes items based on the relevance using dynamic clustering approaches to solve long tail item recommendation. The author combines different techniques first clustering based on dynamic parameter and then applies Markov chain method without negatively affecting the accuracy of the prediction. A detailed summary of the cluster-based RS is presented in **Table 3**.

Table 3. Cluster based recommendation system.

Article	Method name	Data sets	Evaluation criteria	Application area
Hu et al. (2022)	MASR (Memory Bank Augmented Long-tail Sequential Recommendation)	MovieLens-1M, Musical, Video, Diginetica, Yoochoose	Hit Ratio HR@20 = 0.4828, HR@10 = 0.3802 NDCG N@20 = 0.2381, N@10 = 0.2121	E-comm
Mussi et al. (2022)	DynaLT (Dynamic pricing for the Long Tail)	real-world E-comm website	Long-tail Global Performance DynaLT 1.4	E-comm
Yalcin (2022)	Popularity-aware recommendation technique (PopHybrid)	MovieLens, Yahoo	Average Popularity of the Recommended Items (APRI), Ratio of Popular Items (RPI)	E-comm
Park (2013)	Adaptive Clustering Method	MovieLens, BookCrossing	Mean absolute error (MAE), RMSE	General
Lakshmi et al. (2021)	Adaptive Correlation Clustering-Based Recommender System	MovieLens	MAE, RMSE	Movie
de Sousa Silva & Durão (2020)	Dynamic Clustering and Markov chains	MovieLens	Recall, Diversity, Popularity	E-comm

In conclusion, we noted that cluster-based long-tail approaches had been employed mainly in e-commerce, movies, and general-purpose applications. More elaborate recommendations could be obtained by clustering the items and users by sequential, dynamic, popularity-based, and adaptive methods.

3.2 Graph Based Approaches

In Graph based approach, data of user and items are stored in the form of nodes and edges, which is basically used to manage the relationship between the user and the items. Organizing data of user and items in a form of graph can be used to predict link between the user and the item. By using an undirected edge-weighted graph, Yin et al. (2012) proposes Hitting time as a method for improving accurate and diverse to recommend niche product for the long tail item recommendations. By enabling time-space efficient generation for rare queries, Bonchi et al. (2012) propose long-tail queries on center-piece subgraphs, which are being used in web search. The author constructed a query-flow graph having term node, query node and their connected link node with highly correlated queries. This paper (Shi, 2013) addresses the need to improve the overall recommendation quality, the author proposes graph-based recommendation to reduce the problem of long-tail recommendation by balancing different criteria. The methodology involves graph-based cost flow concept for recommendation. It uses two real world datasets Movielens and lastfm, it is considered that alone accuracy is not enough for giving correct prediction other criteria should also be considered. This study by (Huang et al., 2016) investigates long tail queries using the knowledge-based approach to extract entities from the query suggestion process in web search. In this article, the author uses a query-flow graph for solving the problem of long-tail query recommendation.

The paper by Johnson & Ng (2017a) addresses the problem of items long-tail with tripartite graphs. The author tries to highlight the importance of niche products by improving long-tail item recommendations with the Markov process. The author uses the Movielens dataset to evaluate the recall and diversity to improve the recommendation. Johnson & Ng (2017b) in his paper investigates the importance of graph for solving the problem of long-tail recommendation. The paper discusses various algorithms for graph traversing to identify similar users and the items, focusing on the random walker's traversal to identify long-tail items.

Wang et al. (2018b) in his paper investigates enhancement of user experience for the long-tail recommendation. The paper discusses user rating by developing recommendation scores that are used for knowing user experience. In this study, user personal experience has been used to produce top-on recommendations. The results suggest that user experience has a significant impact on their preference for item recommendation. Luke et al. (2018) in his paper addressed the problem of traditional RS by graph-based recommendation approach. The author proposed random walkers on tripartite graphs to modify the hitting time algorithm to enhance the likelihood by improving the traditional RS. This approach aims to improve the traditional RS which focuses on top-n items, leaving long tail items behind. The methodology used in this paper involves enhanced tripartite graph combining with Markov process to recommend long-tail recommendation.

Huang & Wu (2019) proposes two components to deal with long-tail recommendations. The item profile extractor captures consumer sentiment, while the similar item extractor identifies equivalent substitute products. It uses Amazon and Airbnb datasets to evaluate Precision, Novelty, and Diversity on the dataset. Niu et al. (2020) addresses the problem, that includes between user queries and shop names, for delivering good search results for long-tail queries. The author proposes graph neural network and attention network to deal the long-tail queries.

de Sousa Silva et al. (2020) improves the long-tail item recommendation by combining graph similarities with clustering techniques. To enhance the visibility of the RS hitting time algorithm has been combined with clustering techniques. The study was conducted on Movielens dataset, to evaluate the RS metrics like recall, diversity and popularity have been used. Achary & Patra (2021) proposes a graph-based approach to deal with the challenges of recommending relevant long-tail items. The author combines traditional approach with graph-based approach. The Movielens datasets were used to evaluate precision and novelty metrices, which provide very satisfying results. Islam et al. (2023) investigates how to compute the top-k sets through maximum margin sum product results for long-tail items. The author uses several real-world datasets to find top-k item sets with highest score, subject to diverse list with respect to previously selected sets to achieve equitable top-k results.

Yang et al. (2023) proposed graph-based data augmentation to enhance long-tail recommendation. The author uses two different methods first is finding niche walks and second is tail session mixup. The focus is to address the problem of data sparsity and item popularity using real real-world datasets. The proposed methodology shows superior performance evaluating different metrices like Hit Rate, Mean Reciprocal Rank and Coverage. To enhance the problem of long-tail item recommendation (Wei et al., 2023) proposed meta-graph learning framework. The author uses meta graph learning to optimize edge generator for item recommendation. Two real-world datasets have been used to assess model performance using NSCG and Hit Ratio. Zhao et al. (2023a) proposes a long-tail augmentation approach through a graph convolutional network to address the problem of data sparsity. The method focuses on enhancing the tail node by predicting neighbor information based on the resulting graph. Three benchmark datasets, including precision, recall, F1, NDCG, and Hit Ratio, have been used to evaluate the results. Zhang et al. (2024) proposes a method based on graph convolution network combining with Bayesian method to address challenges related to misinformation on social media or other platforms. The method focuses on exploiting and capturing interaction data. To detect misinformation the model was tested on two public twitter datasets, the proposed long-tail strategy significantly enhanced misinformation detection capabilities.

Balasubramanian et al. (2024) proposes a novel method, where the less popular items receive inadequate recommendations. The method uses user history and aims to enhance personalization and improve the recommendation for both long-tail and popular items. The paper uses two real world benchmark datasets MovieLens and BookCrossing. For evaluating the model Hit ratio and nDCG metrics were used which significantly enhances recommendation performance. Lin et al. (2024) proposes a novel method, to multimodal RS by the limited interaction data of long-tail items and representation of user modality preference. The author uses methods that enhance the user-item graph using multimodal similarity to improve the representation of long-tail items. The model uses 4 categories of Amazon datasets which significantly outperform the state-of-the-art methods. A detailed summary of graph-based RS has been presented in **Table 4**.

The section is summarized through the representation of the user and item in the form of a graph, which can be a tripartite graph, an undirected graph, or a centerpiece graph. Niche walks, maximum marginal sums of products, and even graph convolution networks can be used to identify long-tail products. Graph-based engines are mainly used in e-commerce, search engines, and general websites.

Table 4. Graph based recommendation system.

Article	Method name	Data sets	Evaluation criteria	Application area
Yin et al. (2012)	User-item interaction, undirected edge-weighted graph	MovieLens, Douban	Recall@N, Popularity, Similarity, Diversity, Efficiency	E-comm
Bonchi et al. (2012)	Center-Piece Subgraphs	Yahoo, MSN	efficiency comparing the average runtime per query	Search Engines
Shi (2013)	A novel cost flow concept based on a 1st order Markovian graph	MovieLens, Last.fm	Accuracy = 0.632-0.205, Similarity = 1.23-0.182, Diversity = 535-1041, Long-tail= 0.014-0.003	General
Huang et al. (2016)	Knowledge base Query Recommendation	YAGO	Coverage, Precision	search engines
Johnson & Ng (2017a)	Tripartite graphs and Markov processes	MovieLens	Recall, Diversity	Online shopping, movie or music
Johnson & Ng (2017b)	Tripartite graphs	MovieLens	Recall, Diversity	E-comm
Wang et al. (2018b)	Long Tail based on User's Experience	RateBeer	Accuracy=22.90, recall=12.33, coverage=26.54, F-measure=16.03	E-comm
Luke et al. (2018)	Long-Tail Items recommendation using extended tripartite graphs	MovieLens	Recall, Diversity	E-comm
Huang & Wu (2019)	Biterm Topic Model	Amazon, Airbnb	Precision, Novelty, Diversity	General

Table 4 continued…

Niu et al. (2020)	Graph neural networks and a dual heterogeneous graph attention network integrated with a two-tower architecture	E-commerce platform Taobao	AUC= 0.8800, GAUC=0.7581 MRR=0.9314, HR@1=0.9435, HR@5=0.9961	E-comm
de Sousa Silva et al. (2020)	Graph-based technique to calculate node similarity between users and items	MovieLens-100K	Accuracy, Diversity, Popularity	E-comm
Achary & Patra (2021)	Graph Based Hybrid Approach for Long-Tail Item Recommendation	MovieLens-100K, MovieLens-1M	100K Precision=0.469, Novelty=0.010 1M Precision=0.47, Novelty=0.0018	E-comm
Islam et al. (2023)	Maximum Marginal Sum of Products (MMSP)	Yelp, IMDB-top 1000, IMDB, Airbnb, Synthetic, Makeblobs	Recall=91%	E-comm
Yang et al. (2023)	Niche Walk Augmentation (NWA) and Tail Session Mixup (TSM)	Nowplaying, Diginetica, Retailrocket, Yoochoose 1/4	Hit Ratio=20.15, Coverage=86.96	E-comm
Wei et al. (2023)	Meta Graph Learning	MovieLens-1M, Bookcrossing	nDCG, Hit Ratio	General
Zhao et al. (2023a)	Graph Convolutional Networks	Yelp2018, Amazon-Book, MovieLens-25M	Recall(Yelp)=0.0732, (Amazon)= 0.0522, (ML25)=0.3579, nDCG(Yelp)= 0.0604, (Amazon)= 0.0415, (ML25)=0.2509	E-comm
Zhang et al. (2024)	Graph Convolutional Networks	Twitter	Accuracy, F1-score, MCC	Social media
Balasubramanian et al. (2024)	Graph Convolutional Networks	MovieLens-1M, BookCrossing	HR@k, nDCG@k	General
Lin et al. (2024)	Graph Convolutional Networks	Amazon	Recall@K, nDCG@K	E-comm

3.3 Deep Learning & Neural Network Based Approaches

A subfield of machine learning, can be further classified as Multilayer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Artificial Neural Network (ANN), and Autoencoder (AE). Krishnan et al. (2018a) propose neural CF to improve the long-tail problem. The author focuses on overcoming sparsity and implementing adversarial learning guided by user feedback data. M. Li et al. (2019) investigates micro-videos retrieval across social media platform with graph convolutional network to improve the accuracy between recommended videos and users. Due to the lack of behavior features, long-tail items have difficulties getting recommended. Liu et al. (2019) proposes an attention-based mechanism to address the issue. The author uses a real-time attention mechanism to enhance user representation learning. User representation learning learns from different features of user fields, and look-alike learning uses local and global attention to construct a relation to the target users. Tang et al. (2019) build a neural network model to deal with both short-term and long-term dependencies. Meenakshi & Satpal (2019) proposes a deep neural network on webpages to enhance website ranking and data sparsity and cold start issues. Zhao et al. (2020) uses conversational recommender systems (CRS) to address the problem. Conversational recommender systems (CRS) tend to recommend top items leaving long-tail items behind. To address the problem the author proposes pre-training to enhance to recommend items from long-tail.

Yin et al. (2020) proposes a framework that addresses the long-tailed distribution issues through transferable learning parameters through optimization and feature perspectives. The author employs gradient optimizer and adversarial training to balance the performance of head and tail users. The model is evaluated using Hit ratio (HR) and nDCG. Liu & Zheng (2020) uses session data to predict users' next actions. It uses numerous applications to record users' session data. This session data provides users with diverse information that increases the likelihood of serendipitous suggestions. The author proposes TailNet, a DL model that uses user preferences between long-tail and short-headed items. In his paper, Jang et al. (2020) addresses the problems of recommending tail items. The author proposes a CITIES framework to improve the recommendation performance of tail items using contextual embeddings. The author uses two real-world public datasets: Yelp and Amazon. Yelp consists of user reviews of local businesses, and the Amazon dataset contains user products. Sreepada & Patra (2020) proposes a novel approach to overcome the rating problem in the RS, using a deep Siamese network to recommend long-tail items. The proposed framework uses two real-world datasets Movielens and Netflix to evaluate the model. The author uses few-shot techniques to improve the performance of the RS. Zhang et al. (2021) proposes a dual transfer learning through model-level and item-level to improve long-tail item recommendation. It uses real-world public datasets Movielens and Bookcrossing to evaluate the model using hit ratio and NDCG metrics which improves performance for tail items. The study investigates (Ge et al., 2021), that item popularity dynamically changes over time, which may affect user engagement to maintain for a longer time. To maintain the recommendation process, the author suggested the use of reinforcement learning. Using the Movielens dataset, the model outperforms baseline method's recommendation accuracy.

To enhance low resource recommendation in traditional CF (Wen et al., 2021) propose a knowledge graph as a method to address this problem. Using a knowledge graph the study aims to improve the accuracy for users with few interactions, which is often overlooked in traditional RS. Two real-world benchmark datasets Movielens and twitter datasets have been used to evaluate the model which outperforms on all the parameters in handling long-tail user recommendations. Chen et al. (2023) proposes a calibration module that predicts the ratio of tail items in the recommendation list from the ongoing session. According to the author, the primary goal of his work is to reduce popularity bias in recommendations from a user-centric perspective. This study examines the natural language conversation by establishing a relationship between pre-train and retrieval techniques (Zhao et al., 2023b) to improve rarely mentioned items. In this study, two public datasets are used to examine the effectiveness of natural language conversation in resolving long-tail issues. Papso (2023) proposes methods to identify relations among products that are purchased together to fulfill a joint demand in a sparse e-commerce network with limited engagement. The proposed methodology uses pre-trained DL models which are being used to fine-tune e-commerce to enhance product recommendation. Gong et al. (2023) propose a DL technique to address the problem of long-tail item recommendations. The deep retrieval method uses user-item interaction in conjunction with demographic information about the users. The limitation of the model is due to the lack of user-item interaction and changing demographic information. Liu et al. (2023a) proposes a methodology to improve the performance of the sequential recommendation model. To identify and utilize long-term dependencies in user behavior, we need to identify and utilize long-term dependencies. Two benchmark datasets were used to improve the accuracy of predicting the next recommendation item. Kim et al. (2023) introduced a way to address the problem of long-tailed problem of both users and items, rather than focusing on one. The paper proposes a bilateral branch that is trained to mutually enhance each other users and items without sacrificing performance of head items or users. As a result of noise and class imbalance present in social image classification, Ricci et al. (2023) propose methods to resolve the problem of image classification. The study aims to enhance the performance of classification tasks, in real-life environments, where the data is noisy and long-tail distributions. The proposed method uses an effective way to train a model, that can improve the performance of rare classes. Wu et al. (2024) proposes a method to enhance LLM for recommending task by integrating prompt based on user-item interactions. The author employs reinforcement learning framework to explore collaborative information the reasoning capability for providing recommendation. The algorithm updates based on user feedback and optimized recommendation through a reinforcement learning method. A detailed summary of deep learning-based RS is presented in **Table 5**.

We summarize this section by using a pre-trained model to enhance recommendation, i.e. fine-tuning. Getting items recommended is difficult because of the lack of behavioral characteristics between the user and the item.

Table 5. Deep learning based recommendation system.

Article	Method name	Data sets	Evaluation criteria	Application area
Krishnan et al. (2018a)	Adversarial training strategy to enhance long-tail recommendations for users with Neural CF (NCF) models.	Movielens-20M, Ask-Ubuntu Stack Exchange	Recall at K, and nDCG	Q&A forums and movie recommendation
Li et al. (2019)	Micro-video hashtag recommendation	INSVIDEO	Recall@K, nDCG@K	Micro-video
Liu et al. (2019)	Real-time attention-based look-alike model (RALM)	Wechat	precision@K and AUC	Chat
Tang et al. (2019)	Neural Multi-temporal range Mixture Model (M3)	MovieLens, YouTube	mAP	General
Meenakshi & Satpal (2019)	long-tail web services using DL techniques	not specified	Recall	Web services
Zhao et al. (2020)	Multi-latent representations	Amazon, Goodreads, and MovieLens	RMSE	E-comm
Yin et al. (2020)	Long-tailed sequential user behavior modeling	Amazon, MovieLens, MovieTweetings3	Hit Ratio (HR), nDCG	E-comm
Liu & Zheng (2020)	Session-based recommendation TailNet, to improve long-tail recommendation performance.	YOOCHOOSE, 30MUSIC	Recall, Coverage, Tail-Coverage	MovieTweetings3
Jang et al. (2020)	Contextual Inference of Tail-item Embeddings for Sequential Recommendation	Yelp, Amazon	hit ratio, mean reciprocal rank	E-comm
Sreepada & Patra (2020)	Long tail using few shot learning technique siamese networks	MovieLens, Netflix	Precision, Recall, F1, Binary Preference Relation (bpref)	E-comm
Zhang et al. (2021)	Dual Transfer Learning Framework	Movielens-1M, BookCrossing	Recall, Precision, nDCG	E-comm, online movie
Ge et al. (2021)	Constrained Markov Decision Process (CMDP)	Movielens-100K and Movielens-1M	Short-term evaluation and Long-term evaluation	E-comm
Wen et al. (2021)	Knowledge-enhanced collaborative meta learner	Movielens, Twitter	MSE, Area Under the Precision-Recall Curve (PR-AUC), Relative Cross Entropy (RCE)	General
Chen et al. (2023)	Session-based Recommendation from Calibration	YOOCHOOSE, Last.fm	Recall@N, MRR@N, Coverage@N, TailCoverage@N, Tail@N	E-comm
Zhao et al. (2023b)	Conversational recommender systems (CRS)	ReDial and Inspired	Recall, Coverage	Movie recommendation
Papso (2023)	Product Universal Embedding Space (PUES)	Amazon Review	HR and nDCG metrics	E-comm
Gong et al. (2023)	Full Index Deep Retrieval (FIDR)	MovieLens-1M, KuaiRec and Douyin Ads	Precision, Recall, and F-measure	E-comm
Liu et al. (2023a)	LinRec Linear Attention Mechanism	MovieLens-1M, Gowalla	Recall, Mean Reciprocal Rank (MRR), and nDCG	E-comm, online movie
Kim et al. (2023)	MELT Mutual Enhancement of Long-Tailed	eight real-world datasets	precision, recall, nDCG, and Hit Ratio	E-comm, social networking, and online advertising
Ricci et al. (2023)	Meta-learning Advisor Networks	CIFAR10, CIFAR100, ImageNet-LT, Places-LT, and Clothing1M datasets	improvements by 12.33%	Social image classification
Wu et al. (2024)	LLM to Improve Long-tail	Amazon	AUC, F1	E-comm

3.4 Multi-Objective Based Approaches

This section demonstrates the capability of the recommendation technique to recommend popular items those are top rated as well as long-tail items using multi-objective optimization. The aim of multi-objective optimization is to identify all possible Pareto solutions based on a Pareto optimal front, which visualizes trade-offs between objectives. Wang et al. (2016) propose a framework to balance the recommendation of popular items and long-tail items. Through the implementation of long-tail items, the author addresses the challenge of maintaining novelty and accuracy while optimally optimizing two contradictory objective functions. The experiment was conducted on two datasets Movielens and Jester dataset, which successfully generated effectively balances the accuracy and novelty of the recommendation items. Hu et al. (2017) propose a method for enhancing credibility for users and focusing on specialization for long-tail items for enhancing recommendation. As the recommender system faces data sparsity challenges and cold start issues, the model addresses these issues. The two objective functions of highlighting credibility and specialty are multi-objective optimization problems. According to Hamedani & Kaedi (2019), a personalized diversification approach to the long-tail recommendation problem can enhance the performance of the overall RS. Diversity, long-tail, and accuracy are three objective functions used in this method. Based on Movielens and Netflix datasets, the proposed method was evaluated regarding precision and RMSE. Pang et al. (2019) proposes an algorithm to improve long-tail, while maintaining both accuracy and coverage. The author uses multi-objective optimization using a weighted similarity based method on NSGA-II. Due to the focus on accuracy, long-tailed items do not get recommended, the author mainly focuses on user satisfaction and maintaining the overall performance of recommendations. Shafiloo et al. (2024) proposes a method that dynamically collect user preferences, to enhance the recommendation based on user changing interests over time. The proposed method utilizes age prediction and multi-objective optimization methods to recommend products based on user tastes. To evaluate the model precision, novelty, and aggregate diversity criteria are being used which outperform the traditional algorithm. A detailed summary of multi-objective based RS is presented in **Table 6**.

Table 6. Multi-objective based recommendation system.

Article	Method name	Data sets	Evaluation criteria	Application area
Wang et al. (2016)	Multi-objective optimization	MovieLens, Jester, Netflix	Precision, Novelty, Diversity	General
Hu et al. (2017)	C-HMF, S-HMF	Epinions Dataset	MAE, precision, recall, Average precision, nDCG)	Online shopping
Hamedani & Kaedi (2019)	Long Tail Items through Personalized Diversification	Movielens, Netflix	precision, RMSE, diversity	General
Pang et al. (2019)	weighted similarity mea sure based on NSGA-II	MovieLens, Netflix	Accuracy, coverage	Movie
Shafiloo et al. (2024)	Multi-objective optimization using memetic algorithm	MovieLens	Precision, novelty	Movie

3.5 Traditional Recommendation Based Approaches

Traditional RS primarily focuses on accuracy, which primarily recommend products that are closely aligned with the user's previous interests. It uses different traditional approaches to recommend products to the user using content-based, CF and hybrid methods. Park & Han (2012) propose that diversity in product recommendations can help a company succeed economically. By examining customer diversity, the study hopes to reduce dissatisfaction and boost profitability at the company. Zhang et al. (2012) investigates items that are not so popular, but as time passes gain popularity in a sub-set of the product space. The main aim is to reduce the reliance on popular items and to recommend popular items from their niche category. The study balances both popular and unpopular items by adjusting the bias to achieve varying levels of accuracy and diversity in the recommendation. To surprise customers, Niemann & Wolpers (2013) propose presenting users with a wide range of novel and diverse items, including niche items. The author uses a CF approach rather than an association mining approach to generate characteristic vectors based on the co-occurrence patterns of items. According to Hu et al. (2014), LDA is used to identify the categories that describe a user's interest using the nearest neighbor to produce recommendations efficiently. The experiments positively impact user engagement and can significantly enhance user interaction and business outcomes.

Ho et al. (2014) investigates the impact of discovering long-tail items to improve user experience and diversification of the items to increase quality and quantity of the items. The study lies in the contribution in the field of RS by enhancing the diversity of recommendation which leads to richer user experience, increase in user satisfaction and provide engagement with niche items in various domains. In their paper, Agarwal et al. (2019) proposes a hybrid collaborative framework for enhancing diversity in long-tail items. Aiming to improve customer experience by improving the diversity of recommendations and making them relevant and engaging, the study aims to improve customer experience. Two real-world datasets - Movielens and Netflix, were used to assess the model's precision and diversity. Sreepada & Patra (2021) proposes a method to selectively inject ratings into a long-tail item list to improve the visibility of the long-tail items. Equitable distribution enhances the visibility of diverse items and can improve item visibility and may increase the sale of less popular items. Liu et al. (2023b) proposes a method that aims to improve the performance of tail items that have insufficient context for embedded learning. The author uses 3 datasets like beauty, retail rocket and books to evaluate the performance using metrics such as recall and hit rate. Zhang et al. (2023) proposes a novel method that reduces the differences in the learning process between memorization and generalization. The author's aim is to predict user engagement, while maintaining overall recommendation effectiveness. A detailed summary of the Traditional recommendation-based approach is presented in **Table 7**.

Table 7. Traditional recommendation system.

Article	Method name	Data sets	Evaluation criteria	Application area
Park & Han (2012)	Diversity-based recommendation strategy examining customer churn	Offline retailer	Churn Rate Analysis	Marketing, Customer Relationship Management
Zhang et al. (2012)	Double-ranking strategy	Movielens, Yahoo	concentration index (CI), the Gini index (GI), Precision and Recall	E-comm, Media Streaming
Niemann & Wolpers (2013)	Context based CF	MovieLens, Netflix	Diversity, Novelty and Accuracy	E-comm
Hu et al. (2014)	Latent Variable Models	Etsy	Conversion Rate, Pages Viewed Rate, Activity Feed Visit Rate, User Follow Rate, and Item Favorite Rate	E-comm
Ho et al. (2014)	Aggregate diversity enhancement	MovieLens	Prediction, Coverage, Long-tail recommendation	General
Agarwal et al. (2019)	Hybrid Reranking framework in CF	MovieLens, Netflix	Precision, Diversity	E-comm
Sreepada & Patra (2021)	Long tail Econophysics-inspired	MovieLens, Netflix, Bookcrossing	Precision, Recall, Aggregate Diversity, Long Tail Diversity, Weighted Long Tail Coverage, Long Tail Item Relevance	E-comm
Liu et al. (2023b)	Co-occurrence embedding enhancement for Long-Tail (CoLT)	Beauty, RetailRocket and Books	Recall and Hit Rate	E-comm
Zhang et al. (2023)	Enhancement for Long-Tail	MovieLens-1M and BookCrossing	Hit Ratio and nDCG	E-comm

This section is summarized as a traditional RS that uses traditional methods to recommend products to the users. Traditional RS, where these methods mainly focus on recommendation of the top-n products leaving the long-tail products behind. uses methods like ranking, double-ranking, embedding content-based, CF and hybrid RSs.

3.6 Other Methods

Other methods used for product recommendation do not use traditional methods or any other methods discussed previously. According to Hwang & Li (2014), content monetization can be enhanced through personalized recommendations that are based on user preferences. By understanding customer preferences, patterns can be developed for increasing sales using products from different categories. According to the findings, the recommendation is closely related to the variety and price of content and is directly influenced by factors such as social and personal factors. Based on a cosine pattern, Wang et al. (2014) propose a pattern-based method for recommending popular and niche items. In this study, two real-world datasets, last.fm and Movielens were used to investigate metrics like recall and f-measure. To improve recommendation, Seifert et al. (2015) proposes a web-based system that incorporates user interaction. The author focuses on user-personalized experience on feature-rich data sets to enable personalization in the long-tail domain and collecting data from web users. A probabilistic model based on similarity measures and spreading algorithms was proposed by the author to deal with the long-tail problem (Luo & Xie, 2016). Users' behavior is used to recommend items from long tail lists based on the probabilistic model. Li et al. (2017) have divided the items into two categories to address the issue of long-tail: popular items have a lower rank while sparse items have a lower rank. To improve user engagement and overall satisfaction, the author uses an iterative approach.

Krishnan et al. (2018b) propose a Bayesian approach that is used to partition users based on their behavior and latent patterns. The author interprets relevant information from user behavior analysis that reveals user engagement patterns, preferences, and interactions. User behavior explains user preferences, engagement, and the effectiveness of the recommended algorithm for enhancing the user experience. As a solution to the problem of long-tail items, Garigliotti et al. (2019) proposes a probabilistic approach based on support information. A context retrieval approach was used to identify long-tail entities and relevant items from the items examined by the author. Comparing context retrieval to existing approaches, the study improves the retrieval of items using context retrieval. Qin (2021) proposed neighboring based method to increase long-tail recommended problem using user's historical data. It uses behavioral patterns of user, such as browsing history, comments etc. The author uses singular value decomposition on user and items to compare the long-tail items using nearest neighbor. Pandey & Ankayarkanni (2020) proposed a method to deal with the problem of a long-tail recommendation system by storing users purchased data and recommending a product based on their history. The author first stored the user’s data based on a clustering approach and used user purchase history to make classification of long-tailed items. Alshammari et al. (2020) proposed a method that interchanges between CF and Content-based filtering to enhance the ability of the model to suggest relevant items to the users. The author conducted a hybrid multi-level algorithm to enhance the suggestion of relevant items, despite their lower popularity. Didi et al. (2023) proposed a method that is used to click long-tail items in conjunction with head items, adding long-tail items to the recommendation list will reduce popularity bias and enhance user satisfaction. A detailed summary of the other RS has been presented in **Table 8**.

This section is summarized through the other methods that are used for recommending products to the customers like using probabilistic models, cosine patterns, and neighborhood-based methods.

Table 8. Other recommendation system.

Article	Method name	Data sets	Evaluation criteria	Application area
Hwang & Li (2014)	Economic model	Online channel, e-store	Equilibrium price and profit	E-comm
Wang et al. (2014)	Cosine pattern-based recommendation	MovieLens, Last.fm	Recall, precision and f-measure	E-comm
Seifert et al. (2015)	Collaborative-filtering	Web-based user data	Accuracy	E-comm
Luo & Xie (2016)	A probabilistic model	MovieLens	RMSE, precision and recall	E-comm
Li et al. (2017)	Handling of cold-start and long-tail recommendations	Flickr, BlogCatalog, YouTube, Hetrec11-LastFM	Precision, recall	Online shopping, movie or music recommendation, and news or article recommendations
Krishnan et al. (2018b)	Learning robust behavior representations in online platforms	Coursera, stack-exchange	Precision, Recall, F1-score, AUC	Stack-Exchanges are community Q&A websites, Coursera feature video lectures
Garigliotti et al. (2019)	Generative probabilistic framework to rank contexts	Wikipedia	MAP, MRR	News articles
Qin (2021)	Neighborhood-based recommendation method	Movielens-1M	Coverage, LTRate	Not defined
Pandey & Ankayarkanni (2020)	Random forest	Online transactional data	Not defined	E-comm
Alshammari et al. (2020)	A switching multi-level recommender system	Movielens	MAE, RMSE	Movie

4. Datasets

RQ2: Which datasets are predominantly used in long-tail recommendation research and development?

Datasets play a very crucial role when it comes to testing a model, it acts as a backbone. The RS uses datasets to store the user's past preferences, item attributes, item ratings, and so on. After looking at **Table 2**, we found these are the different categories under which the long-tail recommendation datasets fall. Long-tail recommendation systems use several different types of datasets, which are discussed in this section. We found that the datasets used in long-tail recommender systems are grouped under the following categories based on **Table 2**. A list of datasets has been examined for all the articles listed in **Table 2** to answer the research question. After collecting the datasets these datasets have been put into different categories and presented, then we have also calculated the frequency of each datasets and presented it in a tabular form.

Table 9. Datasets.

Category	Datasets
Movie/media recommendation	ML-(MovieLens 100K, 1M, 20M, 25M), Netflix, Yahoo, Douban, Jester, Last.fm, Hetrec11-LastFM, INSVIDEO, YouTube, BookCrossing, Musical (MovieLens-1M + Musical), Gowalla, KuaiRec, Nowplaying
E-Commerce/online retail	Amazon (various versions, including Amazon Review, Amazon-Book), Goodreads, Etsy, E-commerce platform Taobao, ReDial and Inspired, RetailRocket, Beauty, RetailRocket, and Books
Social media/network	Yelp (including Yelp2018), Flickr, BlogCatalog, Wechat, Twitter
Collaborative filtering/reviews	Epinions Dataset, YAGO, RateBeer
General purpose/academic	Coursera, Stack Exchange (Ask-Ubuntu, Stack Exchange), Wikipedia, YOOCHOOSE, 30MUSIC, Synthetic and Makeblobs
Browser-based query and ratings	queries and ratings dataset from browser-based setting
Multi-domain datasets	CIFAR10, CIFAR100, ImageNet-LT, Places-LT, and Clothing1M datasets, IMDB-top 1000, IMDB, MovieTweetings3, Movielens, Yahoo, MovieLens, Jester, Netflix, Amazon, Goodreads, MovieLens, Yelp, IMDB, Airbnb, Synthetic and Makeblobs
Online transactions/retail	Online transactional data, offline retailer, online channel, e-store
Mixed datasets	CIFAR10, CIFAR100, ImageNet-LT, Places-LT, and Clothing1M, Hetrec11-LastFM, Retailrocket, Real-world E-commerce website, ReDial and Inspired, Yelp, IMDB-top 1000, IMDB, Airbnb, Synthetic and Makeblobs

The top five datasets that are mostly used in long-tail recommendation systems that are collected from **Table 2** are as follows.

Table 10. Datasets information.

Serial no.	Dataset name	Frequency
1.	MovieLens[1]	37
2.	Netflix	6
3.	Amazon[2]	6
4.	BookCrossing	5
5.	YOOCHOOSE	4

Table 11. Popular datasets used in long-tail recommendation.

Dataset	Users	Movies	Ratings	Sparsity (%)
MovieLens 100K	943	1682	100,000	94%
MovieLens 1M	6,040	3,900	1,000,209	96%
MovieLens 10M	71,567	10,681	10,000,000	99%
MovieLens 20M	138,493	27,278	20,000,263	99%
MovieLens 25M	162,541	62,423	25,000,095	99%
Netflix Prize	480,000	17,770	100M	99%
Amazon (Electronics)	900K	300K	1.2M	99%
Amazon (Books)	6.6M	3.6M	12M	99%
Amazon (Movies)	800K	200K	2.2M	99%
Book-Crossing	278K	271K	1.15M	99%

After examining **Table 2** and **Table 9**, it has been found that most of the datasets are highly sparse (99.9%). For datasets like Movielens[1] and Netflix, the ratings are explicitly provided, while for datasets like Amazon, Yoochoose, and Book-Crossing the ratings are implicitly provided listed in **Table 11**. From the above **Table 10**, which mentions the frequency of the top 5 datasets, the Movielens dataset is the most frequently selected one, followed by Netflix and so on.

5. Evaluation Metrics

RQ3: What are the different evaluation criteria used for long-tail recommendation systems?

Articles listed in table 2 have been verified to evaluate the metrices used for evaluation. After examining all the articles, it has been found that there are various types of metrics that are used to evaluate an algorithm's performance as illustrated in **Figure 6**.

[1] http://grouplens.org/datasets/movielens/
[2] http://jmcauley.ucsd.edu/data/amazon/

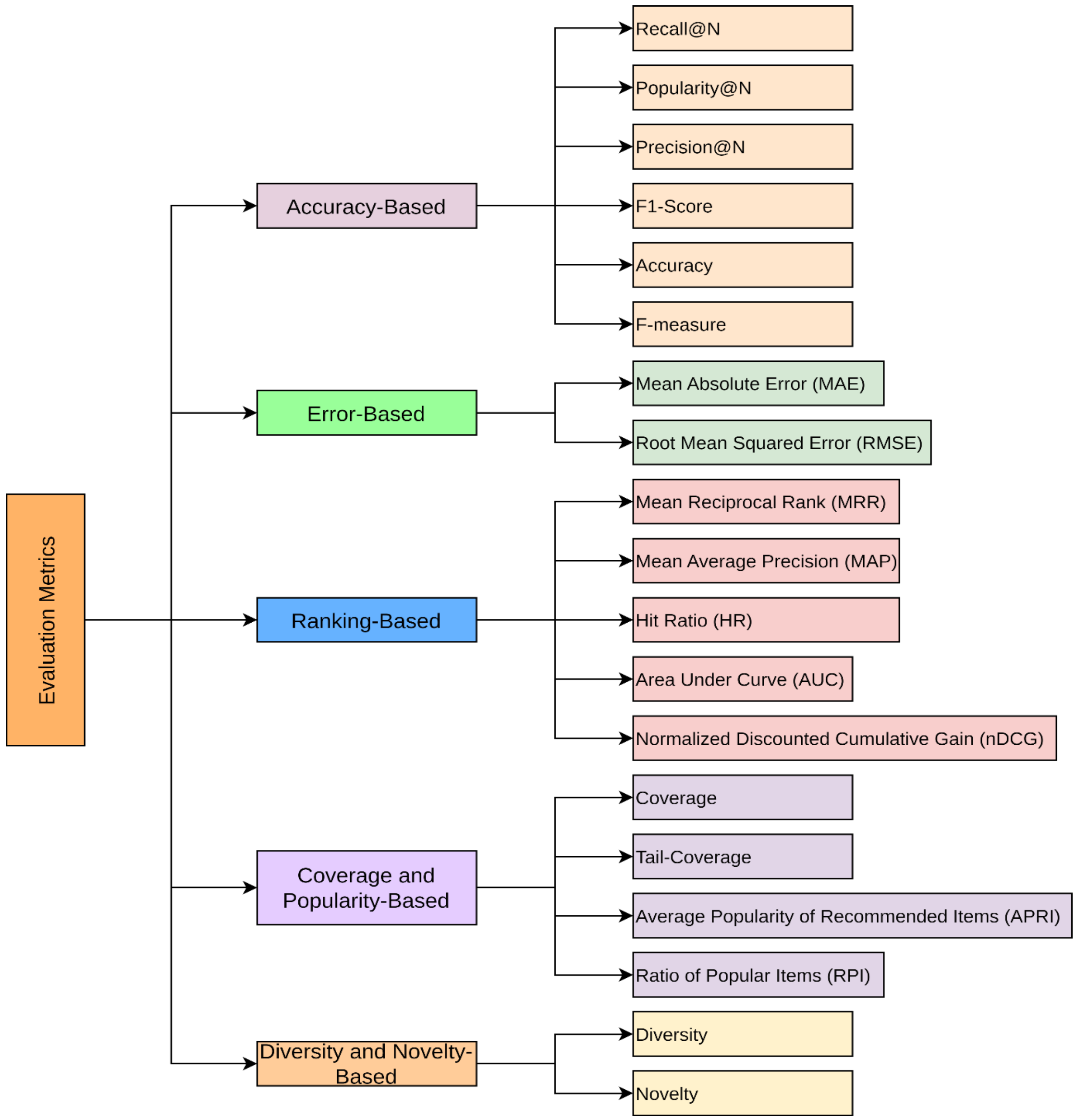

Figure 6. Evaluation metrices.

5.1 Accuracy-Based Metrics

5.1.1 Recall@N

Recall@N metric is used to evaluate the performance of the recommendation algorithms (de Sousa Silva et al., 2020). It is used to measure the accuracy of the relevant items that have been retrieved in the top-N recommendation list from the total no of relevant items. Recall is used as an indicator that tells how the user likes a particular item from the recommended items. To calculate Recall@N, we must consider hit@N, if hit@N is equals to 1 that means the items is in the top-N list, or else it will return 0.

$$Recall@N = \frac{\sum hit@N}{|L|} \tag{2}$$

Hit@N is used as a test case to calculate the Recall@N, which is used to check how many times a particular item appears from the testcase in the top@N items. Where |L| is the number of available testcases. The result of the recall determines how much user prefers an item from the recommended list.

5.1.2 Popularity@N

Popularity metric is used to measure the performance of an item, and how likely a recommended item is popular among the users (de Sousa Silva et al., 2020). The Popularity@N defines how a particular item is popular among the recommended items. The higher the value of N the more likely that the item is popular among the users.

$$Popularity = \frac{R_u}{\sum |R_d|} \tag{3}$$

$$R_u = \frac{\sum |R_r|}{|U| \,.top@N} \tag{4}$$

R_u represents the rating that has been normalized among the users and the recommended items. R_d represents the overall rating of the dataset. U represents the set of users for calculating the popularity of a recommended item. The top@N is the recommended list of items for the user set. Therefore, the higher value of the popularity metrics will give the popularity of an item, keeping in mind the list of items as the list grows the value of an item may decrease.

5.1.3 Precision@N

Precision metrics is an accuracy-based measure which is widely used to measure none of relevant items from a given list of recommended items. Precision@N is used to measure relevant items for the list of top@N recommended list (Hu et al., 2017).

$$Precision@N = \frac{\sum_{N=1}^{N} rel(N)}{N} \tag{5}$$

Average precision@K is used to find the average of the result that is obtained from the precision@K (Hu et al., 2017).

$$Average\ precision@K = \frac{\sum_{K=1}^{K} rel(k)\ x\ precision@k}{\min\ (K,N)} \tag{6}$$

5.1.4 F1-Score

F1 Score in RS is used to balance the list between precision and recall (Sreepada & Patra, 2020). Precision measures are used to compute the relevant items present in the recommended list and the recall measure is used to compute the number of recommended items that are present in the relevant list of items.

$$F1 - Score = \frac{2\ x\ Precision\ x\ Recall}{Precision + Recall} \tag{7}$$

5.1.5 Accuracy

Accuracy measures focus on the overall correctness of the prediction by balancing between relevant and irrelevant items in the recommended list (Wang et al., 2018b).

$$Accuracy = \frac{\Sigma_{ui} |R(u_i) \cap T(u_i)|}{\Sigma_{ui} |R(u_i)|} \tag{8}$$

where, $R(u_i)$ is the recommended set in term of number, u_i is the set of users and $T(u_i)$ is the test set.

5.1.6 F-Measure

The F-measure is used to evaluate weighted harmonic mean of accuracy and recall controlled by B, i.e. used to prioritize one metric over the other (Wang et al., 2018b).

$$F - \ measure = \frac{2 * accuracy * recall}{accuracy + recall} \tag{9}$$

5.2 Error-Based Metrics

The Error-based metrics are used to measure the performance of RS by measuring the differences between predicted and actual user ratings or preferences. These metrics are used for numerical predictions like ratings to quantify the accuracy of predicted values for continuous outputs. While using error-based methods it is important to keep in mind that lower values provide better results. Some of the Regression-based Metrics are Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root Mean Squared Error (RMSE).

5.2.1 Mean Absolute Error (MAE)

The MAE is a measure to find the absolute difference between the predicted and the actual ratings, by treating all errors equally (Alshammari et al., 2020). MEA works well with the sparse dataset and provides a more balanced view of the data if the dataset contains noise and outliers that do not reflect true user preferences.

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |p_i - \ r_i| \tag{10}$$

p_i represents the predicted rating, and r_i represents the actual rating in the equations above.

5.2.2 Root Mean Squared Error (RMSE)

The RMSE is a measure to find square root of the average squared difference between the actual and the predicted values (Alshammari et al., 2020). RMSE is one of the popular metrics used in rating prediction in RS as it balances interpretability.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (pi - ri)^2} \tag{11}$$

5.3 Ranking-Based Metrics

The Ranking-based metrics evaluate the items by showing their relevance based on the users or query. It is used to order the recommended items based on user query that how the ranking is evaluated based on user preferences or user query. It is commonly used to rank the items from the recommended list.

5.3.1 Mean Reciprocal Rank (MRR)

The MRR metric orders the item to evaluate the effectiveness of the ranking system of the RS.

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \tag{12}$$

$|Q|$ is the number of queries or users, i is the specific query and $rank_i$ is the position of the first relevant result of the i query.

5.3.2 Mean Average Precision (MAP)

The MAP metric is used to evaluate the performance of a recommended model that calculates the average precision of multiple queries and then compute the meaning of its value (Tang et al., 2019). It is used to evaluate the recommended model and how well the recommended system ranks the relevant item from the given list of recommended items.

$$MAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{13}$$

In the above equation, *n* represents no of queries or users, AP_i represents average precision of the *i*-th query.

5.3.3 Hit Ratio (HR)

The HR@N metric is a commonly used measure which is used to evaluate the predictive metric of a recommended model (Yin et al., 2020). It is used to evaluate the desired items from the list of top-N recommended list.

$$HR@K = \frac{1}{|U|}\sum_{u \in U} I(R_{u.gu} \leq K) \tag{14}$$

where, *u* is the user, I is an indicator function, gu is the item of user *u*, and $R_{u.gu}$is the rank of gu generated by the model.

5.3.4 Area Under Curve (AUC)

The AUC metric is used to evaluate binary classification. It is used to calculate the probability, which plots the curve of randomly chosen relevant items ranking higher than randomly chosen irrelevant items.

$$AUC = \frac{1}{|P||N|}\sum_{i=1}^{|P|}\sum_{j=1}^{|N|} \| (\hat{y_i} > \hat{y_j}) \tag{15}$$

5.3.5 Normalized Discounted Cumulative Gain (nDCG)

This metric is used to rank the quality of the ranking system (Yin et al., 2020). The metrics place the item based on the position of the relevant item in the list by giving weight to the item placement. The higher value of the metrics provides better performance of the model.

$$NDCG@K = \frac{1}{|U|}\sum_{u \in U} \frac{I(R_{u.gu} \leq K)}{\log_2(R_{u.gu}+1)} \tag{16}$$

5.4 Coverage and Popularity Based Metrics

5.4.1 Coverage

The coverage measures are used to evaluate how many items were recommended for the entire list of items set (Chen et al., 2023).

$$Coverage@N = \frac{|\cup_{s \in S_{te}} RL_s|}{|I|} \tag{17}$$

5.4.2 Tail-Coverage

The tail coverage measure is used to evaluate how many item from the tail item list is recommended in the current testing set (Chen et al., 2023).

$$TailCoverage@N = \frac{|\cup_{s \in S_{te}} (RL_s \cap I_{Tail})|}{|I_{Tail}|} \tag{18}$$

5.4.3 Average Popularity of Recommended Items (APRI)

The APRI metric is used to calculate the average popularity of an item that has been recommended from the top-N list to the user (Yalcin, 2022). The metric is used to calculate the popularity of each item in the dataset then it is used to divide users who provide the rating of individual unit then it calculates the average of that and present the item in the recommended list.

$$APRI = \sum_{i \in K} \frac{P_i}{|K|} \tag{19}$$

5.4.4 Ratio of Popular Items (RPI)

The RPI metrics is used to calculate the percentage of the popular item from the top-N list of recommended items (Yalcin, 2022). The list that is generated will be following Pareto principle to check whether the item in the head item falls under 20% of the top list or not.

$$RPI = \frac{\sum_{i \in K} \mathbb{1}(i \in H)}{|K|} \tag{20}$$

Having lower APRI and RPI will result in having more items from the long tail list in the recommended list. The controlling parameter $\mathbb{1}(.)$ returns 1 if it is correct or returns 0.

5.5 Diversity, Novelty

5.5.1 Diversity

The diversity metrics are used to provide users with different varieties of items (de Sousa Silva et al., 2020). To compute this metrics, for a set of users, if each user has been provided with a list of top@K items. Let assume for a set of 10 users, and for each user top@10 ranking is provided with a recommended item. Then we have a total of 100 recommended items for all 10 users.

$$Diversity = \frac{|\cup I_u \in I|}{|\cup| .top@N} \tag{21}$$

where, I_u is a single unique item that has been recommende to all users. The I element is an item from the dataset, set of users are represented as $\cup$ and the number of items that is being recommanded to each user is represents as $top@N$, which may vary based on the required list of recommendations.

5.5.2 Novelty

The novelty metrics of RS is used to recommend new or novel items to the target users list (Wang et al., 2016).

$$novelty = \frac{1}{mk} \sum_{u=1}^{m} \sum_{i \in L_u} di \tag{22}$$

where, in L_u u is the user and L is the top-k list of a user, m is number of users and di is the degree of item i, i.e., the users who have rated the item i.

After examining the entire system, it has been clear that novelty, diversity, accuracy and nDCG are the important metrics for RS, for user engagement and better product selling.

6. Discussion

The RS is more about understanding the relationship between the user and the item. Hence, there are several ways to generate a recommendation based on the user's preferences.

Clustering methods are used to improve the recommendation by grouping users who have similar interest, to group them into multiple groups or clusters (Roy & Dutta, 2022). The main purpose of clustering approach is to group users or items of similar interest to group together and dissimilar interest into different clusters based on users, or items of their choice. Clustering can be formed using k-means, fuzzy c-means, hierarchical and density-based clustering (Koohi & Kiani, 2016; Diplaris et al., 2018; Wang et al., 2018a, Ahuja et al., 2019; Selvi & Sivasankar, 2019). The accuracy metrics of long tail recommendation can be improved through clustering methods. For long-tail items, recommendations mainly rely on ratings from densely clustered groups, while for head items, they relieve on individual ratings. The advantage of clustering methods in long-tail recommendations increase accuracy, tailored recommendation and handling diversity (Shi, 2013; Huang et al., 2016).

Graph based RS includes the ability to capture the complex relationship between the user and the items. The graph-based recommendation enables the discovery of less popular items that is not possible with the traditional RS. The graph recommendation uses bipartite, tripartite graph to enhance the item recommendations for long-tail items (Luke et al., 2018). The graph-based recommendation can provide locally related recommendations with high efficiency, based on the contextual understanding of the relationships (Huang & Wu, 2019). The disadvantage of graph-based method is the quality may degenerate if the dataset becomes larger and sparser and the low connectivity of long-tail item in the graph may affect the diversity of the recommendation (Liu et al., 2019).

Deep-learning based methods have an ability to handle large-scale, high-dimensional data for capturing complex user item interactions. The DL based RS uses techniques like multilayer perception, autoencoders, convolutional neural network, recurrent neural network, generative adversarial network to advance deep reinforcement models (Hu et al., 2017). DL is used in RS when there is huge amount of data complexity or when there are large number of training instances (Hamedani & Kaedi, 2019). Long-tail recommendations still require more emphasis to be given to users and items relationships. The advantage of DL in long-tail recommendation is that it can scale huge amounts of data. DL based methods face issues because of cold start problems, non-structured data and more computational cost.

Multi-Objective optimization method is used to optimize multiple contradictory objective functions simultaneously (Park & Han, 2012; Pang et al., 2019). The recommendation tries to balance various objectives or metrices which are often conflicting with each other (Zhang et al., 2012). These objectives derive a set of pareto optimal solutions, which means one objective does not worsen another objective to find non dominated solutions (Niemann & Wolpers, 2013). The advantages of multi-objective optimization are they can handle conflicting objectives, can provide pareto optimal solutions and even can explore a broader search area using multiple solutions at the same time (Park, 2013). The disadvantage of multi-objective optimization is that they can increase complexity, difficulties in parameter tuning and can complicate the decision-making process.

Traditional Recommender Systems: In traditional RS various methods are used to find the similarity among the users, and items they purchased which may include CF, content-based filtering and the combination of both to form a hybrid RS. The key technologies used in traditional RS are user rating, similarity metrics, item popularity, matrix factorization, single value decomposition, ranking strategies which try to capture the user-item or item-item relationship (Koren et al., 2009; Shi et al., 2014; Alamdari et al., 2020; Walek & Fojtik, 2020). Traditional RS facing problems like cold start, data sparsity (Lika et al., 2014). The problem with traditional recommendation is that it cannot handle huge amounts of data, the problem of scalability is still there. Traditional recommendation can be used in smaller applications with fewer features.

While the traditional recommendation method may primarily focus on head items, by often neglecting the tail items. Other recommended methods use different strategies that do not belong to any group and at the same time can deal with both head items and long-tail items. The other RS might increase the complexity in implementation and need more computational resources (Didi et al., 2023).

Over the past few years, long-tail recommendation systems have become increasingly popular both in academia and in industry. As a result, long-tail recommendation research is still very young and has room to improve in the aforesaid directions. For top-n recommendations matrix factorization, clustering and nearest neighbors' techniques work well, but dealing with long-tailed items these techniques don't work well, whereas for long-tail items graphs work better than the traditional one. Therefore, a combination of methods is required to find an accurate and reliable recommendation since every method has its own advantages and disadvantages.

7. Conclusion

This paper presented a systematic literature review on long-tail recommendations, focusing on the different approaches in long-tail recommendation system. By promoting long-tail products, companies can not only increase sales but also enhance user engagement. As a result, the long-tail recommendation system is an important topic for strategically growing sales. This paper systematically explains different approaches to the long-tail recommendation system, which has been divided into six different groups, and then further explains these approaches in detail. This paper provides different categories of datasets that can be used for dealing with long-tail recommendations. A table of different categories of long-tail datasets used in various applications or domains has been provided.

As the review and analysis are conducted, we can say that soon we may see cross approaches, that may dominate the recommendation system. As the datasets are clearly defined based on different categories, researchers can use various datasets based on their domain categories. Previously, it has been observed based on the study that a single dataset is mostly used in most domains. As part of the study, evaluation criteria have also been categorized, which may help researchers decide which types of metrics are useful for their work.

Conflict of Interest

The authors confirm that there is no conflict of interest to declare for this publication.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors would like to thank the editor in chief, guest editors and anonymous reviewers for their comments that help improve the quality of this work.

AI Disclosure

The author(s) declare that no assistance is taken from generative AI to write this article.

References

Achary, N.S., & Patra, B.K. (2021). Graph based hybrid approach for long-tail item recommendation in collaborative filtering. In *Proceedings of the 3rd ACM India Joint International Conference on Data Science & Management of Data* (pp. 426-426). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3430984.3431058.

Aditya, P.H., Budi, I., & Munajat, Q. (2016). A comparative analysis of memory-based and model-based collaborative filtering on the implementation of recommender system for E-commerce in Indonesia: a case study PT X. In *2016 International Conference on Advanced Computer Science and Information Systems* (pp. 303-308). IEEE. Malang, Indonesia. https://doi.org/10.1109/ICACSIS.2016.7872755.

Afsar, M.M., Crump, T., & Far, B. (2023). Reinforcement learning based recommender systems: a survey. *ACM Computing Surveys*, *55*(7), 1-38. https://doi.org/10.1145/3543846.

Agarwal, P., Sreepada, R.S., & Patra, B.K. (2019). A hybrid framework for improving diversity and long tail items in recommendations. In: Deka, B., Maji, P., Mitra, S., Bhattacharyya, D.K., Bora, P.K., Pal, S.K. (eds) *Pattern Recognition and Machine Intelligence*. Springer International Publishing, Cham, pp. 285-293. https://doi.org/10.1007/978-3-030-34872-4_32.

Ahuja, R., Solanki, A., & Nayyar, A. (2019). Movie recommender system using K-means clustering and K-nearest neighbor. In *2019 9th International Conference on Cloud Computing, Data Science & Engineering* (pp. 263-268). IEEE. Noida, India. https://doi.org/10.1109/CONFLUENCE.2019.8776969.

Alamdari, P.M., Navimipour, N.J., Hosseinzadeh, M., Safaei, A.A., & Darwesh, A. (2020). A systematic study on the recommender systems in the e-commerce. *IEEE Access*, *8*, 115694-115716. https://doi.org/10.1109/ACCESS.2020.3002803.

Alshammari, G., Jorro-Aragoneses, J.L., Kapetanakis, S., Polatidis, N., & Petridis, M. (2020). A switching approach that improves prediction accuracy for long tail recommendations. In: Bi, Y., Bhatia, R., Kapoor, S. (eds) *Intelligent Systems and Applications*. Springer International Publishing, Cham, pp. 18-28. https://doi.org/10.1007/978-3-030-29516-5_3.

Anderson, C. (2006). *The long tail: why the future of business is selling less of more*. Hyperion, New York.

Balasubramanian, K., Alshabanah, A., Markowitz, E., Steeg, G.V., & Annavaram, M. (2024). Biased user history synthesis for personalized long-tail item recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems* (pp. 189-199). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3640457.3688141.

Batmaz, Z., Yurekli, A., Bilge, A., & Kaleli, C. (2019). A review on deep learning for recommender systems: challenges and remedies. *Artificial Intelligence Review*, *52*(1), 1-37. https://doi.org/10.1007/s10462-018-9654-y.

Bonchi, F., Perego, R., Silvestri, F., Vahabi, H., & Venturini, R. (2012). Efficient query recommendations in the long tail via center-piece subgraphs. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 345-354). Association for Computing Machinery, New York, USA. https://doi.org/10.1145/2348283.2348332.

Burke, R. (2002). Hybrid recommender systems: survey and experiments. *User Modeling and User-Adapted Interaction*, *12*(4), 331-370. https://doi.org/10.1023/A:1021240730564.

Chen, J., Wu, W., Shi, L., Zheng, W., & He, L. (2023). Long-tail session-based recommendation from calibration. *Applied Intelligence*, *53*(4), 4685-4702. https://doi.org/10.1007/s10489-022-03718-7.

de Sousa Silva, D.V., & Durão, F.A. (2020). Dynamic clustering personalization for recommending long tail items. *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems, 21,* 417-425. https://doi.org/10.15439/2020F157.

de Sousa Silva, D.V., de Oliveira, A.C., Almeida, F., & Durão, F.A. (2020). Exploiting graph similarities with clustering to improve long tail items recommendations. In *Proceedings of the Brazilian Symposium on Multimedia and the Web* (pp. 193-200). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3428658.3431078.

Didi, T., Guy, I., Livne, A., Dagan, A., Rokach, L., & Shapira, B. (2023). Promoting tail item recommendations in e-commerce. In *Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization* (pp. 194-203). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3565472.3592968.

Diplaris, S., Satsiou, A., Følstad, A., Vafopoulos, M., & Vilarinho, T. (2018). *Internet science*. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-77547-0.

Garigliotti, D., Albakour, D., Martinez, M., & Balog, K. (2019). Unsupervised context retrieval for long-tail entities. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval* (pp. 225-228). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3341981.3344244.

Ge, Y., Liu, S., Gao, R., Xian, Y., Li, Y., Zhao, X., Pei, C., Sun, F., Ge, J., Ou, W., & Zhang, Y. (2021). Towards long-term fairness in recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining* (pp. 445-453). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3437963.3441824.

Goldberg, D., Nichols, D., Oki, B.M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, *35*(12), 61-70. https://doi.org/10.1145/138859.138867.

Gong, Z., Wu, X., Chen, L., Zheng, Z., Wang, S., Xu, A., Wang, C., & Wu, F. (2023). Full index deep retrieval: end-to-end user and item structures for cold-start and long-tail item recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems* (pp. 47-57). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3604915.3608773.

Guo, Q., Zhuang, F., Qin, C., Zhu, H., Xie, X., Xiong, H., & He, Q. (2022). A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, *34*(8), 3549-3568. https://doi.org/10.1109/TKDE.2020.3028705.

Hamedani, E.M., & Kaedi, M. (2019). Recommending the long tail items through personalized diversification. *Knowledge-Based Systems*, *164*, 348-357. https://doi.org/10.1016/j.knosys.2018.11.004.

Ho, Y.-C., Chiang, Y.-T., & Hsu, J.Y.-J. (2014). Who likes it more?: mining worth-recommending items from long tails by modeling relative preference. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining* (pp. 253-262). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/2556195.2566589.

Hu, D.J., Hall, R., & Attenberg, J. (2014). Style in the long tail: discovering unique interests with latent variable models in large scale social E-commerce. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1640-1649). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/2623330.2623338.

Hu, L., Cao, L., Cao, J., Gu, Z., Xu, G., & Wang, J. (2017). Improving the quality of recommendations for users and items in the tail of distribution. *ACM Transactions on Information Systems*, *35*(3), 1-37. https://doi.org/10.1145/3052769.

Hu, Y., Liu, Y., Miao, C., & Miao, Y. (2022). Memory bank augmented long-tail sequential recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (pp. 791-801). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3511808.3557391.

Huang, X., & Wu, F. (2019). A novel topic-based framework for recommending long tail products. *Computers & Industrial Engineering*, *137*, 106063. https://doi.org/10.1016/j.cie.2019.106063.

Huang, Z., Cautis, B., Cheng, R., & Zheng, Y. (2016). KB-enabled query recommendation for long-tail queries. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management* (pp. 2107-2112). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/2983323.2983650.

Hwang, T.K., & Li, Y.M. (2014). Optimal recommendation and long-tail provision strategies for content monetization. In *2014 47th Hawaii International Conference on System Sciences* (pp. 1316-1323). IEEE. Waikoloa, USA. https://doi.org/10.1109/HICSS.2014.169.

Islam, M.M., Asadi, M., & Roy, S.B. (2023). Equitable top-k results for long tail data. *Proceedings of the ACM on Management of Data*, *1*(4), 1-24. https://doi.org/10.1145/3626727.

Jang, S., Lee, H., Cho, H., & Chung, S. (2020). CITIES: contextual inference of tail-item embeddings for sequential recommendation. In *2020 IEEE International Conference on Data Mining* (pp. 202-211). Sorrento, Italy. https://doi.org/10.1109/ICDM50108.2020.00029.

Johnson, J., & Ng, Y.K. (2017a). Enhancing long tail item recommendations using tripartite graphs and Markov process. In *Proceedings of the International Conference on Web Intelligence* (pp. 761-768). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3106426.3106439.

Johnson, J., & Ng, Y.K. (2017b). Using tripartite graphs to make long tail recommendations. In *2017 8th International Conference on Information, Intelligence, Systems & Applications* (pp. 1-6). IEEE. Larnaca, Cyprus. https://doi.org/10.1109/IISA.2017.8316436.

Kim, K., Hyun, D., Yun, S., & Park, C. (2023). MELT: mutual enhancement of long-tailed user and item for sequential recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 68-77). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3539618.3591725.

Koohi, H., & Kiani, K. (2016). User based collaborative filtering using fuzzy C-means. *Measurement*, *91*, 134-139. https://doi.org/10.1016/j.measurement.2016.05.058.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, *42*(8), 30-37. https://doi.org/10.1109/MC.2009.263.

Krishnan, A., Sharma, A., & Sundaram, H. (2018b). Insights from the long-tail: learning latent representations of online user behavior in the presence of skew and sparsity. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (pp. 297-306). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3269206.3271706.

Krishnan, A., Sharma, A., Sankar, A., & Sundaram, H. (2018a). An adversarial approach to improve long-tail performance in neural collaborative filtering. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (pp. 1491-1494). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3269206.3269264.

Kulkarni, S., & Rodd, S.F. (2020). Context aware recommendation systems: a review of the state of the art techniques. *Computer Science Review*, *37*, 100255. https://doi.org/10.1016/j.cosrev.2020.100255.

Lakshmi, S.S., AdiLakshmi, T., & Abhinith, B. (2021). An adaptive correlation clustering-based recommender system for the long-tail items. In: Satapathy, S.C., Bhateja, V., Favorskaya, M.N., & Adilakshmi, T. (eds) *Smart Computing Techniques and Applications*. Springer, Singapore, pp. 505-513. https://doi.org/10.1007/978-981-16-1502-3_50.

Li, J., Lu, K., Huang, Z., & Shen, H.T. (2017). Two birds one stone: on both cold-start and long-tail recommendation. In *Proceedings of the 25th ACM International Conference on Multimedia* (pp. 898-906). Association for Computing Machinery, New York, USA. https://doi.org/10.1145/3123266.3123316.

Li, M., Gan, T., Liu, M., Cheng, Z., Yin, J., & Nie, L. (2019). Long-tail hashtag recommendation for micro-videos with graph convolutional network. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 509-518). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3357384.3357912.

Lika, B., Kolomvatsos, K., & Hadjiefthymiades, S. (2014). Facing the cold start problem in recommender systems. *Expert Systems with Applications*, *41*(4), 2065-2073. https://doi.org/10.1016/j.eswa.2013.09.005.

Lin, G., Zhen, M., Wang, D., Long, Q., Zhou, Y., & Xiao, M. (2024). GUME: graphs and user modalities enhancement for long-tail multimodal recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management* (pp. 1400-1409). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3627673.3679620.

Liu, L., Cai, L., Zhang, C., Zhao, X., Gao, J., Wang, W., Lv, Y., Fan, W., Wang, Y., He, M., Liu, Z., & Li, Q. (2023a). LinRec: linear attention mechanism for long-term sequential recommender systems. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 289-299). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3539618.3591717.

Liu, S., & Zheng, Y. (2020). Long-tail session-based recommendation. In *Fourteenth ACM Conference on Recommender Systems* (pp. 509-514). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3383313.3412222.

Liu, Y., Ge, K., Zhang, X., & Lin, L. (2019). Real-time attention based look-alike model for recommender system. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2765-2773). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3292500.3330707.

Liu, Y., Zhang, X., Zou, M., & Feng, Z. (2023b). Co-occurrence embedding enhancement for long-tail problem in multi-interest recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems* (pp. 820-825). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3604915.3608835.

Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: state of the art and trends. In: Ricci, F., Rokach, L., Shapira, B., & Kantor, P.B. (eds) *Recommender Systems Handbook*. Springer, US, pp. 73-105. https://doi.org/10.1007/978-0-387-85820-3_3.

Luke, A., Johnson, J., & Ng, Y.K. (2018). Recommending long-tail items using extended tripartite graphs. In *2018 IEEE International Conference on Big Knowledge* (pp. 123-130). IEEE. Singapore. https://doi.org/10.1109/ICBK.2018.00024.

Luo, S., & Xie, J. (2016). A probabilistic perspective model for recommendation considering long tail effect. In *2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress* (pp. 147-150). IEEE. Auckland, New Zealand. https://doi.org/10.1109/DASC-PICom-DataCom-CyberSciTec.2016.43.

Meenakshi, M. & Satpal. (2019). A novel approach web services based long tail web services using deep neural network. In *2019 International Conference on Issues and Challenges in Intelligent Computing Techniques* (pp. 1-9). IEEE. Ghaziabad, India. https://doi.org/10.1109/ICICT46931.2019.8977655.

Mehta, R., & Rana, K. (2017). A review on matrix factorization techniques in recommender systems. In *2017 2nd International Conference on Communication Systems, Computing and IT Applications* (pp. 269-274). IEEE. Mumbai, India. https://doi.org/10.1109/CSCITA.2017.8066567.

Mussi, M., Genalti, G., Trovò, F., Nuara, A., Gatti, N., & Restelli, M. (2022). Pricing the long tail by explainable product aggregation and monotonic bandits. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 3623-3633). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3534678.3539142.

Niemann, K., & Wolpers, M. (2013). A new collaborative filtering approach for increasing the aggregate diversity of recommender systems. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 955-963). Association for Computing Machinery. New York, USA.

Niu, X., Li, B., Li, C., Xiao, R., Sun, H., Deng, H., & Chen, Z. (2020). A dual heterogeneous graph attention network to improve long-tail performance for shop search in e-commerce. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 3405-3415). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3394486.3403393.

Page, M.J., McKenzie, J.E., Bossuyt, P.M., Boutron, I., Hoffmann, T.C., Mulrow, C.D., Shamseer, L., Tetzlaff, J.M., Akl, E.A., Brennan, S.E., Chou, R., Glanville, J., Grimshaw, J.M., Hróbjartsson, A., Lalu, M.M., Li, T., Loder, E.W., Mayo-Wilson, E., McDonald, S., McGuinness, L.A., Stewart, L.A., Thomas, J., Tricco, A.C., Welch, V.A., Whiting, P., & Moher, D. (2021). The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ, 372*(71), 1-9. https://doi.org/10.1136/bmj.n71.

Pandey, A.K., & Ankayarkanni, B. (2020). Recommending e-commerce products on cold start and long tail using transaction data. In *2020 4th International Conference on Trends in Electronics and Informatics* (pp. 661-663). IEEE. Tirunelveli, India. https://doi.org/10.1109/ICOEI48184.2020.9143009.

Pang, J., Guo, J., & Zhang, W. (2019). Using multi-objective optimization to solve the long tail problem in recommender system. In: Yang, Q., Zhou, Z.H., Gong, Z., Zhang, M.L., Huang, S.J. (eds) *Advances in Knowledge Discovery and Data Mining*. Springer International Publishing, Cham, pp. 302-313. https://doi.org/10.1007/978-3-030-16142-2_24.

Papso, R. (2023). Complementary product recommendation for long-tail products. In *Proceedings of the 17th ACM Conference on Recommender Systems* (pp. 1305-1311). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3604915.3608864.

Park, S.H., & Han, S.P. (2012). Empirical analysis of the impact of product diversity on long-term performance of recommender systems. In *Proceedings of the 14th Annual International Conference on Electronic Commerce* (pp. 280-281). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/2346536.2346592.

Park, Y.J. (2013). The adaptive clustering method for the long tail problem of recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, *25*(8), 1904-1915. https://doi.org/10.1109/TKDE.2012.119.

Qin, J. (2021). Long-Tail recommendation framework using frequent neighbors. In *Proceedings of the 2020 8th International Conference on Information Technology: IoT and Smart City* (pp. 6-12). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3446999.3447001.

Ricci, S., Uricchio, T., & Bimbo, A.D. (2023). Meta-learning advisor networks for long-tail and noisy labels in social image classification. *ACM Transactions on Multimedia Computing, Communications, and Applications*, *19*(5), 1-23. https://doi.org/10.1145/3584360.

Roy, D., & Dutta, M. (2022). A systematic review and research perspective on recommender systems. *Journal of Big Data*, *9*(1), 59. https://doi.org/10.1186/s40537-022-00592-5.

Seifert, C., Schlötterer, J., & Granitzer, M. (2015). Towards a feature-rich data set for personalized access to long-tail content. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing* (pp. 1031-1038). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/2695664.2695671.

Selvi, C., & Sivasankar, E. (2019). A novel optimization algorithm for recommender system using modified fuzzy c-means clustering approach. *Soft Computing*, *23*(6), 1901-1916. https://doi.org/10.1007/s00500-017-2899-6.

Shafiloo, R., Kaedi, M., & Pourmiri, A. (2024). Considering user dynamic preferences for mitigating negative effects of long-tail in recommender systems. *Information Sciences*, *669*, 120558. https://doi.org/10.1016/j.ins.2024.120558.

Sharma, R., Gopalani, D., & Meena, Y. (2017). Collaborative filtering-based recommender system: approaches and research challenges. In *2017 3rd International Conference on Computational Intelligence & Communication Technology* (pp. 1-6). IEEE. Ghaziabad, India. https://doi.org/10.1109/CIACT.2017.7977363.

Shi, L. (2013). Trading-off among accuracy, similarity, diversity, and long-tail: a graph-based recommendation approach. In *Proceedings of the 7th ACM Conference on Recommender Systems* (pp. 57-64). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/2507157.2507165.

Shi, Y., Larson, M., & Hanjalic, A. (2014). Collaborative filtering beyond the user-item matrix: a survey of the state of the art and future challenges. *ACM Computing Surveys*, *47*(1), 1-45. https://doi.org/10.1145/2556270.

Sreepada, R.S., & Patra, B.K. (2020). Mitigating long tail effect in recommendations using few shot learning technique. *Expert Systems with Applications*, *140*, 112887. https://doi.org/10.1016/j.eswa.2019.112887.

Sreepada, R.S., & Patra, B.K. (2021). Enhancing long tail item recommendation in collaborative filtering: an econophysics-inspired approach. *Electronic Commerce Research and Applications*, *49*, 101089. https://doi.org/10.1016/j.elerap.2021.101089.

Tang, J., Belletti, F., Jain, S., Chen, M., Beutel, A., Xu, C., & Chi, E.H. (2019). Towards neural mixture recommender for long range dependent user sequences. In *The World Wide Web Conference* (pp. 1782-1793). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3308558.3313650.

Walek, B., & Fojtik, V. (2020). A hybrid recommender system for recommending relevant movies using an expert system. *Expert Systems with Applications*, *158*, 113452. https://doi.org/10.1016/j.eswa.2020.113452.

Wang, S., Gong, M., Li, H., & Yang, J. (2016). Multi-objective optimization for long tail recommendation. *Knowledge-Based Systems*, *104*, 145-155. https://doi.org/10.1016/j.knosys.2016.04.018.

Wang, S., Tang, J., Wang, Y., & Liu, H. (2018a). Exploring hierarchical structures for recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, *30*(6), 1022-1035. https://doi.org/10.1109/TKDE.2018.2789443.

Wang, Y., Wang, J., & Li, L. (2018b). Enhancing long tail recommendation based on user's experience evolution. In *2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design* (pp. 25-30). IEEE. Nanjing, China. https://doi.org/10.1109/CSCWD.2018.8465216.

Wang, Y., Wu, J., Wu, Z., Yuan, H., & Zhang, X. (2014). Popular items or niche items: flexible recommendation using cosine patterns. In *2014 IEEE International Conference on Data Mining Workshop* (pp. 205-212). IEEE. Shenzhen, China. https://doi.org/10.1109/ICDMW.2014.157.

Wei, C., Liang, J., Liu, D., Dai, Z., Li, M., & Wang, F. (2023). Meta graph learning for long-tail recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 2512-2522). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3580305.3599428.

Wen, B., Deng, S., & Chen, H. (2021). Knowledge-enhanced collaborative meta learner for long-tail recommendation. In: Chen, H., Liu, K., Sun, Y., Wang, S., & Hou, L. (eds) *Knowledge Graph and Semantic Computing: Knowledge Graph and Cognitive Intelligence*. Springer, Singapore, pp. 322-333. https://doi.org/10.1007/978-981-16-1964-9_26.

Wu, J., Chang, C.C., Yu, T., He, Z., Wang, J., Hou, Y., & McAuley, J. (2024). CoRAL: collaborative retrieval-augmented large language models improve long-tail recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 3391-3401). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3637528.3671901.

Yalcin, E. (2022). PopHybrid: a novel item popularity-aware hybrid approach for long-tail recommendation. In *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications* (pp. 1-6). IEEE. Ankara, Turkey. https://doi.org/10.1109/HORA55278.2022.9800006.

Yang, H., Choi, Y., Kim, G., & Lee, J.H. (2023). LOAM: improving long-tail session-based recommendation via niche walk augmentation and tail session mixup. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 527-536). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3539618.3591718.

Yin, H., Cui, B., Li, J., Yao, J., & Chen, C. (2012). Challenging the long tail recommendation. *Proceedings of the VLDB Endowment, 5*(9), 896-907. https://doi.org/10.48550/ARXIV.1205.6700.

Yin, J., Liu, C., Wang, W., Sun, J., & Hoi, S.C.H. (2020). Learning transferrable parameters for long-tailed sequential user behavior modeling. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 359-367). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3394486.3403078.

Zhang, G., Zhang, S., & Yuan, G. (2024). Bayesian graph local extrema convolution with long-tail strategy for misinformation detection. *ACM Transactions on Knowledge Discovery from Data*, *18*(4), 1-21. https://doi.org/10.1145/3639408.

Zhang, M., Hurley, N., Li, W., & Xue, X. (2012). A double-ranking strategy for long-tail product recommendation. In *2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology* (pp. 282-286). IEEE. Macau, China. https://doi.org/10.1109/WI-IAT.2012.20.

Zhang, Y., Cheng, D.Z., Yao, T., Yi, X., Hong, L., & Chi, E.H. (2021). A model of two tales: dual transfer learning framework for improved long-tail item recommendation. In *Proceedings of the Web Conference 2021* (pp. 2220-2231). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3442381.3450086.

Zhang, Y., Wang, R., Cheng, D.Z., Yao, T., Yi, X., Hong, L., Caverlee, J., & Chi, E.H. (2023). Empowering long-tail item recommendation through cross decoupling network. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 5608-5617). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3580305.3599814.

Zhao, Q., Wu, Z., Zhang, Z., & Zhou, J. (2023a). Long-tail augmented graph contrastive learning for recommendation. In: Koutra, D., Plant, C., Gomez Rodriguez, M., Baralis, E., Bonchi, F. (eds) *Machine Learning and Knowledge Discovery in Databases: Research Track*. Springer Nature, Switzerland, pp. 387-403. https://doi.org/10.1007/978-3-031-43421-1_23.

Zhao, X., Zhu, Z., Zhang, Y., & Caverlee, J. (2020). Improving the estimation of tail ratings in recommender system with multi-latent representations. In *Proceedings of the 13th International Conference on Web Search and Data Mining* (pp. 762-770). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3336191.3371810.

Zhao, Z., Zhou, K., Wang, X., Zhao, W.X., Pan, F., Cao, Z., & Wen, J.R. (2023b). Alleviating the long-tail problem in conversational recommender systems. In *Proceedings of the 17th ACM Conference on Recommender Systems* (pp. 374-385). Association for Computing Machinery. New York, USA. https://doi.org/10.1145/3604915.3608812.