Research on Personalized Recommendation Algorithms Based on User Profile

Guo Hui[1], Zhou LiQing[2], Chen Mang[3]*, Xv ShiKun[4]
School of Computer Science and Engineering-Guilin University of Technology,
Guilin University of Technology, GUT Guilin, China[1, 4]
Network and Information Center-Guilin University of Technology, Guilin University of Technology, GUT, Guilin, China[2]
Business School-Guilin University of Technology, Guilin University of Technology, GUT, Guilin, China[3]

***Abstract*—In recent decades, recommendation systems (RS) have played a pivotal role in societal life, closely intertwined with people's everyday activities. However, traditional recommendation systems still require thorough consideration of comprehensive user profiles as they have struggled to provide more personalized and accurate recommendation services. This paper delves into the analysis and enrichment of user profiles, utilizing this foundation to tailor recommendations for individuals across domains such as movies, TV shows, and books. The paper constructs a chart comprising 246 types of user profile attributes, primarily covering dimensions like gender, age, occupation, and religious beliefs, among 16 other dimensions. This chart integrates approximately 1.2 million data points, encompassing information relevant to movies, TV shows, and novels. Through training on the dataset, the study has enhanced the model's recommendation effectiveness. Post-training, the recommendation accuracy surpasses that of pre-training based on proposed evaluation metrics. Furthermore, post-manual evaluation, the recommended results are more reasonable and align better with user profiles.**

***Keywords*—Recommender system; large language model; user profile; multi-disciplinary**

I. Introduction

With the continuous development of the Internet and information technology, the world enters the digital age, generating a vast amount of data and information daily. Filtering out content that genuinely interests and meets people's needs becomes increasingly challenging, and the problem of information overload hinders efficient retrieval and utilization of knowledge. Recommender systems play a crucial role in various fields, successfully providing users with personalized consumer goods and entertainment media recommendations. They also contribute to choices in tourism destinations, educational resources, personnel, services, and even lifestyles. Recommender systems represent one of the most widely applied applications in data mining and machine learning technologies. These technologies recommend relevant products to customers, such as movies to watch, items to purchase, and books to read. Over time, differences in user preferences become one of the most significant challenges recommender systems face [1]. In recent years, there have been substantial changes in the presentation of recommendations, especially on e-commerce and streaming platforms. As the quantity of content available on streaming platforms increases, finding content users want to watch becomes more challenging.

To tackle this issue, traditional recommendation systems employ machine learning techniques to optimize suggestions. In personalized movie recommendation systems, machine learning algorithms and user data are utilized [2]. The proliferation of streaming platforms has led to an abundance of movie choices, making it increasingly difficult for users to find relevant content. RS plays a crucial role in assisting users in making informed decisions automatically, helping them navigate through vast amounts of available data. In the realm of movie recommendations, two primary approaches are collaborative filtering, which compares user similarities, and content-based filtering, which considers user preferences [3]. However, machine learning algorithms have limitations in capturing the dynamic and evolving nature of recommendation problems over time, as they tend to extract superficial features. With the advancement of deep learning, both large and small models have emerged, with large models proving advantageous in meeting personalized user demands. Large Language Models (LLMs) have shown significant success across various domains, thanks to their comprehensive contextual understanding and generative capabilities. These models encode vast amounts of knowledge, possess robust reasoning abilities, and adeptly adapt to new tasks through context learning from examples [4]. Recent advancements in LLMs have enabled powerful logical and causal reasoning capabilities, allowing them to identify entities, actions, and causal chains using techniques such as self-attention and contextual embeddings. Large language models exhibit impressive results in a range of Natural Language Processing (NLP) tasks, thanks to their strong logical and causal reasoning abilities [5]. However, personalized user behaviors present challenges in effectively filtering content of genuine interest from vast data due to limited user and item interactions. Current efforts often neglect the consideration of comprehensive user profiles.

We use GPT-3.5 to generate the necessary dataset, ensuring that the recommended results closely align with individual characteristics. This dataset includes three domains: movies, TV Series, and books, totaling approximately 1.2 million entries. We create a graph of 246 personas, covering 16 aspects such as gender, age, occupation, and religious beliefs. We then combine and intersect to build secondary and tertiary datasets. These datasets, at various levels, describe the granularity of individual characteristics, prioritizing recommendations that are more tailored to personas rather than catering to the general interests of the public. Finally, we employ the popular Llama

*Corresponding Author.

model, achieving satisfactory results in both machine and human evaluations. This approach enhances the personalization of recommendations, aligning them more closely with users' individual needs and preferences.

Main contributions:

- In the realm of recommendation systems, we have identified that a more comprehensive and detailed user profile significantly impacts the quality of recommendation outcomes.
- Utilizing GPT-3.5, we have established a knowledge base comprising 16 user profiles and curated a dataset spanning three domains with a total volume of 1.2 million entries.
- Employing the Llama model, we have successfully delivered practical recommendations on the dataset.

II. Related Work

A. Traditional Recommendation Algorithms

Traditional recommendation algorithms can be categorized into three main types: collaborative filtering, content-based recommendation, and hybrid algorithms. Collaborative filtering analyzes past user-item interactions to predict future behavior. While effective in many scenarios, it struggles to capture all aspects of item features, leading to suboptimal recommendations. Content-based recommendation suggests items similar to those a user has interacted with in the past, focusing on surface-level item features. However, it often overlooks user behavior, limiting its effectiveness. Hybrid approaches combine collaborative filtering and content-based recommendation, aiming to improve accuracy and coverage. Yet, they still require more extensive user profile analysis. For instance, Badr et al. [6] developed a recommendation system using K-nearest neighbors and singular value decomposition, but it suffered from low efficiency due to the inability to capture all item features. Collaborative filtering algorithms struggle to extract deep user demands and interest preferences from deep-level interaction data, resulting in a significant bias between recommended content and user needs, yielding suboptimal results [7]. Similarly, Wu et al. [8] successfully combined collaborative filtering and content-based recommendation for book recommendations, and Pazzani et al. [9] suggested aggregating results from multiple algorithms to enhance performance. However, challenges persist. Zhang et al. [10] improved recommendations by combining collaborative filtering with grid-based algorithms, but further user profile feature mining is necessary for optimal results. Dineth et al. [11] utilized a weighted decomposition model but acknowledged the need for more work in capturing user-item relationships. In conclusion, traditional recommendation algorithms often rely on surface-level data, leading to limited accuracy. While efforts have been made to enhance recommendation effectiveness through various approaches, further exploration of deep user profiles and comprehensive user-item relationships is essential for significant improvements.

B. Large Models Recommended Models

LLMs have become vital in NLP and are gaining attention in RS. These models, trained on extensive datasets through self-supervised learning, excel in learning universal representations. Effective transfer techniques like fine-tuning and on-the-fly adjustments offer the potential to enhance recommender systems. Leveraging LLMs improves recommendation quality by providing high performance, quality representation of text features, and extensive external knowledge coverage to establish correlations between items and users [12]. Recent advancements in LLMs, as demonstrated by Zhong et al. [13], exhibit robust logical and causal reasoning capabilities. This progress is attributed to three key advantages. Firstly, LLMs' natural language understanding enables parsing meaning and relationships from text, identifying entities, actions, and causal chains through self-attention and contextual embeddings.BAO et al. [14] introduce BIGRec, a two-step grounding framework for Recommender Systems. This framework fine-tunes LLMs to connect them to the recommendation space, generating meaningful tokens for items and identifying corresponding actual items beyond surface-level feature extraction. Jin et al. [15] extend LLMs' context window with SelfExtend to exploit their long-context processing potential. However, the impact of user profiles on experimental results regarding existing item information remains unexplored. Wang et al. [16] utilize LLMs in various applications, emphasizing relationships between users and items, yet highlighting the need for datasets incorporating user profiles. Wang et al. [17] introduce the Probability Inference Layer (PIL) into Mistral, aiming to enhance information retrieval in NLP, stressing the importance of a comprehensive understanding of user profiles for personalized recommendations.

Based on the analysis above, it is evident that some current research neglects the role of user profiles, while others present incomplete user profiles, failing to acknowledge the significance of user profiles in recommendations.

III. Datasets

In the real-world social context, a deficiency exists in datasets related to user persona descriptions. Consequently, we construct the dataset necessary for our experiment. This dataset spans three domains: movies, television shows, and books. Simultaneously, we develop a comprehensive knowledge graph for character personas.

A. Character Profile Collection

The User profile dataset is mainly built on common human traits, comprising 16 distinct characters, each with different values, and evenly distributed attributes. Gender has two values: male and female. Age is grouped into eight categories: below 10, 11-20, 21-30, 31-40, 41-50, 51-60, 61-70, and above 70 years, catering to diverse age groups. Education includes six levels: elementary, junior high, high school, undergraduate, master's, and doctoral, accurately reflecting users' educational backgrounds. Religion options are Christianity, Islam, Hinduism, Buddhism, Taoism, and no religious beliefs, considering users' preferences. Interests like basketball, badminton, and table tennis cater to various age groups. Idol values are randomly assigned, covering celebrities like Chen

He, Chen Kun, Hu Ge, and Huang Bo. The "specialty" attribute spans multiple domains. Personality and dreams offer insights into individual characteristics. The "children" attribute reflects life experiences, while social media portrays usual lifestyles. Historical records include two recent highly-rated movies, TV series, and books from Douban. Various occupations are covered, providing a comprehensive understanding of users' backgrounds. Integrating these attributes enhances the accuracy and personalization of the recommendation system by considering age, education, religion, and occupation, delivering more targeted recommendations. Table I provides statistical information and a summary of relevant character details.

TABLE I. STATISTICAL ANALYSIS OF CHARACTER INFORMATION

Profile	Value
Sex	Male, Female
Age(years old)	<10,10-20,21-30,31-40,41-50,51-60,61-70,>70
Native place	Suzhou,Sanming,Zhangping,Hezhou,Haikou...
Star sign	Aries, Taurus,Gemini,Cancer,Leo,Virgo...
Qualification	Primary school, Junior high school, High school...
Religion	Christianity,Islam,Hinduism,Buddhism,Taoism...
Interest	playing basketball,playing badminton...
Idol	Zhang Fuqing,Shen Teng,Xv Zheng,Li Yannian...
Strong point	calligraphy,oil painting,Chinese painting...
Character	thoughtful person,calm person,suggestible person...
Dream	go to university,enter MIT for a master's degree...
Pet	cat,dog,hamster,rabbit,fish,duck
Child	son,daughter,no child
Daily	Weibo,Zhihu,Little red book,Douyin,wechat,QQ...
Historical record	The Untamed,The Bad Kids,Joy of Life...
Occupation	public health physician,landscape architect...

B. Data Collection

The user profile dataset is meticulously crafted to cater to users' individual preferences, rather than solely prioritizing top-rated choices. It comprises 16 distinct attributes, such as gender, age, and interests, ensuring a nuanced reflection of users' personalities. Movie recommendations are sourced from highly-rated films on Douban, guaranteeing quality from a diverse selection of genres and languages. An algorithm then suggests 20 movies for each character, with the top 10 most frequently occurring choices being finalized. This approach ensures tailored recommendations while avoiding an overemphasis on widely known works. TV series and books follow a similar principle, drawing from Douban ratings above 9 to deliver personalized content. Despite potential imperfections in attribute construction, the integration of the character dataset enhances recommendation accuracy, accommodating varied user preferences effectively.

The user profile dataset is an innovative application leveraging GPT-3.5 to offer personalized film recommendations. Each movie attribute receives independent recommendations, ensuring comprehensive attention. The process involves 246 character profiles, each with unique movie preferences. From a selection of 100 films, each profile receives 20 tailored recommendations. This approach guarantees personalized suggestions from a diverse film pool. Moreover, these 246 profiles offer users a varied selection, catering to different preferences. This dataset capitalizes on ChatGPT-3.5's vast training and implicit knowledge, providing accurate recommendations. Whether suggesting films for individual profiles or combinations, the aim is to help users find enjoyable results matching their preferences.

The dataset also incorporates combinations of user preferences to enhance the accuracy and diversity of recommendation results. By combining the values of two different user preferences, the recommendation results depict the intersection of individual recommendations for each preference, as shown in Fig. 1. This approach broadens the range of suggestions for each user preference and ensures the precision of the recommendation results. Leveraging the strengths of different user preferences collectively allows users to experience more comprehensive and personalized movie recommendations that better cater to their film preferences. To investigate the impact of the level of detail in user preference descriptions on recommendation results, the three-tier dataset is composed by merging descriptions from three individual user preferences. The recommendation results are derived from the intersection of recommendations from these three personal preferences, as depicted in Fig. 2.

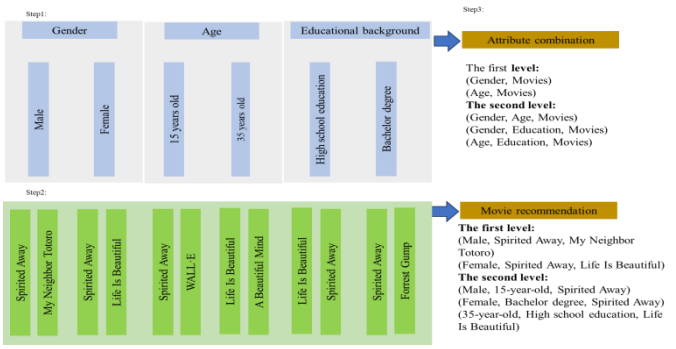


Fig. 1. Steps in the construction of secondary data set.

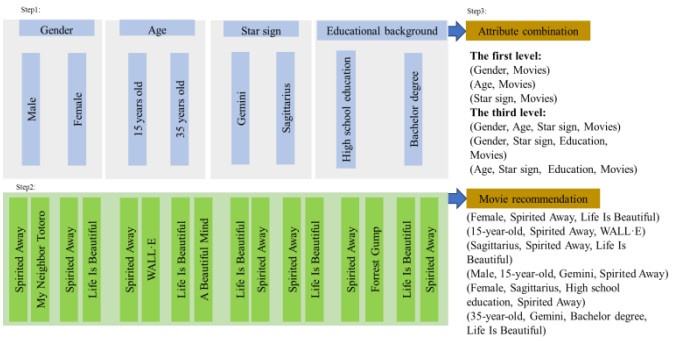


Fig. 2. Steps for the construction of the three-tier dataset.

In summary, the construction and application of user profile datasets serve as an effective method to enhance recommendation systems. By considering user characteristics and preferences, recommended results are more aligned with user interests. Overall, the improved dataset is more refined and comprehensive, taking into account various factors such as

user age, educational background, religious beliefs, and occupation. Works with a Douban rating exceeding 9 points are included in the candidate recommendation scope to alleviate bias in the dataset. Simultaneously, such a selection of works enhances the universality of research results and avoids potential biases. This screening criterion also ensures the quality and popularity of candidate items accurately, making recommended results more likely to align with user tastes. During the recommendation process, each user profile attribute is matched with candidate items, selecting the top 10 most frequently occurring movies, TV shows, and books as the final recommendation results. This role-based recommendation strategy enhances the personalization of the recommendation system, making it easier for users to accept recommendations and strengthening their trust in the system. This, in turn, provides users with a better overall experience, improving the accuracy of the recommendation system and user satisfaction.

IV. Methodology

We employ the LLaMa model in the experiment to train on the dataset we construct. Our prompt is "Given the personality 'Xiao He is an eight-year-old boy from Hezhou City.' The recommended novels are:", and the output after training the LLaMa model is "Harry Potter and the Philosopher's Stone, The Lion, Charlie and the Chocolate Factory, Matilda, The Wind in the Willows, The Secret Garden, The Adventures of Tom Sawyer, The Adventures of Huckleberry Finn." We obtain different predictions by inputting various personas, as illustrated in Fig. 4. The results generated by the recommendation are evaluated against human assessments by calculating the recommendation accuracy. The architecture of the LLaMa model used in the experiment adopts a Transformer Decoder structure with several optimizations in the details. These optimizations include Pre-normalization, SwiGLU activation function, and RoPE rotational position encoding. In the case of Pre-norm, unlike the native Transformer's post-norm approach, which normalizes after each sub-layer output, LLaMa chooses to normalize the data before each sub-layer input. Pre-norm training is more stable than post-norm training and can achieve good results even when training large Transformer models without warm-up operations. Additionally, LLaMa introduces RMSNorm to replace the traditional Layer Norms. As a variant of Layer Norm, RMSNorm differs from Layer Norm in its normalization method, directly dividing by the root mean square instead of subtracting the mean and dividing by the variance. This change makes LLaMa more flexible in model normalization. Furthermore, LLaMa adjusts the activation function using SwiGLU instead of the original ReLU activation function. SwiGLU combines the Swish and GLU functions, introducing a more complex and nonlinear activation mechanism to enhance the model's expressive power when handling complex data. Through these meticulous improvements, LLaMa significantly enhances its model in terms of performance and stability.

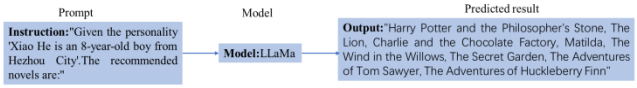


Fig. 3. Experimental workflow diagram.

V. Experiments and Analysis of Results

In the experiment, we utilize a dataset that we have previously constructed. This dataset comprises three research domains and three levels. Relevant data regarding the dataset can be found in Table VI. We employ the alpaca model for recommendations. Finally, we validate the recommendation performance of the dataset before and after training. Fig. 3 shows the experimental workflow diagram.

A. Experimental Setup

The experiment uses a GPU, specifically the RTX 3090(24GB) model, along with PyTorch version 1.11.0, Python version 3.8, and CUDA version 11.3. The experiment is configured with a learning rate of 0.0001, trains for 3 epochs, and has a batch size of 1024. Additionally, the value of the newly generated maximum token is set to 128, the output length is 256, and the micro-batch size is 32, among other parameters. The total duration of the experimental training is 320 hours.

B. Datasets, Baseline and Matrices

1) Datasets. We use a self-constructed dataset to evaluate the recommendation capabilities of large-scale models. This evaluation involves the incorporation of open-source datasets such as ABC, ML-1M, Amazon Beauty, and Amazon Clothing. The ABC dataset [18] consists of a collection of one million Computer-Aided Design (CAD) models. We specifically extract and clean natural language corpora from all non-default text strings in the ABC dataset. This includes the names of parts and modeling features, along with the names of documents containing them, for comparative baseline effectiveness. Table II provides an overview of relevant information for the ML-1M, Amazon Beauty, and Amazon Clothing datasets.

TABLE II. Statistical Information of the Dataset

Name	Users	Items	Actions
ML-1M	6041	3417	999611
Amazon Beauty	22363	12101	198502
Amazon Clothing	39387	23033	278677

2) Baseline. Experiments are being conducted on the ABC, ML-1M, Amazon Beauty, and Amazon Clothing datasets, comparing the recommendation performance of nine baseline models with our trained large-scale model.

TechNet [19], which covers fundamental concepts in all technical fields and their semantic correlations, has been mining the complete United States patent database since 1976. Natural language processing techniques are employed to extract terms from many patent texts to generate TechNet. The latest word embedding algorithms are then used to vectorize these terms and establish semantic relationships.

FastText [20], based on a new approach to the skipgram model, represents each word as a bag of character n-grams. A vector is associated with each character n-gram, and words are described as the sum of these representations. This allows for fast model training on large corpora and enables the

computation of word representations for words not present in the training data.

DistilBERT [21], a method to pre-train a smaller general-purpose language representation model, allows for fine-tuning on various tasks. During the Before-training phase, knowledge distillation is employed, demonstrating the ability to reduce the size of the BERT model by 40% while retaining 97% of language understanding and improving by 60%. A triple loss is introduced, combining language modeling, distillation, and cosine distance to leverage the inductive biases learned by the larger model during Before-training.

DistilBERT-FT [22] refers to the same pre-trained model, which underwent additional fine-tuning on the cleaned version of the ABC corpus that we contributed to in this work.

FDSA [23] initially integrates various heterogeneous features of the ensemble project into feature sequences with different weights, employing a vanilla attention mechanism. Subsequently, FDSA applies separate self-attention blocks to project-level and feature-level sequences, modeling project and feature transition patterns. Finally, the outputs of these two blocks are consolidated into a fully connected layer for the following project recommendation.

BERT4Rec [24] employs deep bidirectional self-attention to model user behavior sequences. By utilizing the sequential recommendation task with a cloze-style objective, predicting a randomly masked item sequence, the model effectively leverages both left and right context through a shared context, preventing information leakage and training the bidirectional model efficiently. The bidirectional representation model recommends incorporating information from the left and right sides for each item in a user's historical behavior.

CL4SRec [25] leverages the traditional next-item prediction task and employs a contrastive learning framework to obtain self-supervised signals from the raw user behavior sequences. Consequently, it can extract more meaningful user patterns and further efficiently encode user representations.

DuoRec [26] introduces a contrastive regularization approach to reshape the distribution represented in sequences. Conventional recommendation tasks typically measure the similarity between sequence representations and item embeddings in the same space through dot products. DuoRec achieves this by implicitly applying regularization to item embedding distributions. Furthermore, DuoRec proposes a model-level enhancement based on Dropout to achieve better semantic retention.

LLMRG [13] consists of four components: Chain Graph Reasoning, Divergent Expansion, Self-Validation and Scoring, and Knowledge Base Self-Improvement. The resulting reasoning graph is encoded using graph neural networks as an additional input, enhancing traditional recommendation systems without requiring different user or item information.

3) Metrics. To evaluate the accuracy of the recommendation results, we compare the similarity of output results before and after training. We employ cosine similarity to measure similarity, with values ranging from 0 to 1. Based on the maximum and minimum values of similarity, the range is divided into segments: 0-0.1, 0.1-0.3, 0.3-0.5, 0.5-0.7, and greater than 0.7, corresponding to ratings of 1-5. In manual evaluation, the manual scoring system is based on users' ratings of the recommended results, serving as an evaluation of both user satisfaction with the recommendations and the reasonableness of the recommendations themselves scores associated with "unsatisfactory recommendation," "satisfactory recommendation," and "excellent recommendation" are respectively assigned as 1, 2, and 3.

C. Case Study

Based on the content of the instructions and recommendations, corresponding profiles are made. Table III presents some examples recommended by the large model, including the predicted results for the listed profiles.

TABLE III. SOME INDIVIDUALS SET RECOMMENDATION INSTANCES

Prompt	Output
Xiao He is a boy,please recommend 20 movies.	Spirited Away,My Neighbor Totoro,Wall.E,The Lion King,Up,Howl's Moving Castle,The Sound of Music,The Little Prince,Coco,The Truman Show,Forrest Gump,The Pursuit of Happyness,Life Is Beautiful,Schindler's List,The Godfather,Hachi: A Dog's Tale,Flipped,The Intouchables,The Legend of Sealed Book,A Dream in Red Mansions
Xiao He is an 8-year-old boy.	Life Is Beautiful, The Sound of Music, Up, Hachi: A Dog's Tale, The Lion King, Coco, My Neighbor Totoro, Forrest Gump, The Little Prince, Spirited Away, The Truman Show
Xiao He is an 8-year-old boy who is from Hezhou City.	Spirited Away, Life Is Beautiful, The Truman Show, Forrest Gump, The Lion King, The Sound of Music

D. Analysis of Results

To comprehensively assess the effectiveness of the experiment, we conduct a multifaceted description and analysis of the experiment and its results from five aspects. These aspects include comparing data before and after training, making comparisons across different domains, examining differences between various levels of datasets, comparing machine evaluation with human evaluation, and conducting exploratory experiments.

1) Comparison of recommended results before and after data training. The experiment begins by utilizing an untrained model, Llama1, for a generation. The generation process relies on predetermined prompts. Following this, the experiment results are organized and modified. Subsequently, the model undergoes training using the original data, and the trained data is employed to generate recommendation results. Likewise, the recommendation results are organized and modified. In the final step, the effectiveness of the recommendation results before and after training is evaluated. This evaluation involves calculating the cosine similarity between the before-training data and the original data, comparing the accuracy of recommendations before and after training, and assessing the overall impact of training on the recommendation outcomes.

Fig. 4 illustrates the average recommendation accuracy of the first-level dataset before and after initial data training.

Each value represents the average derived from 246 data points, indicating recommendation accuracy within specific domains. In the domain of movie recommendations, accuracy increased from 0.5589 before training to 0.5873 after, a 5.08% improvement. For TV Series recommendations, accuracy improved from 0.2858 to 0.2872, a 0.49% increase. Book recommendations saw accuracy rise from 0.6971 to 0.7412, a 6.33% improvement. Comparing average accuracy before and after training, it's clear that post-training accuracy generally surpasses pre-training accuracy. Secondary dataset averages are also presented in Fig. 4. In the movie domain, accuracy improved from 0.3551 to 0.3731, a 5.07% increase. For TV Series, it rose from 0.0857 to 0.1056, a notable 23.22% improvement. Book recommendations saw an increase from 0.3796 to 0.4833, showcasing a 27.32% improvement. Tertiary dataset analysis follows, where movie accuracy increased from 0.3110 to 0.3277, a 5.37% improvement, TV Series accuracy from 0.0837 to 0.0976,a 16.61% improvement, and book accuracy from 0.5475 to 0.5589,a 2.08% improvement. Overall, post-training accuracy is consistently higher, aligning better with user preferences.

Fig. 4. Comparison of data before and after training the primary dataset.

2) Comparison of recommendations between different research areas. The experiment uses datasets from three domains: movie recommendations, TV Series recommendations, and book recommendations. Accuracy is assessed by applying the cosine similarity formula to evaluate the precision of recommendation results in various domains.

The Fig. 5 compares recommendation accuracy across different recommendation domains at three levels of data before training on the original dataset. In Fig. 6, the accuracy for movies in the first-level data is 0.5563, for TV series is 0.2858, and for books is 0.6971. In the second-level data, the accuracy for movies is 0.3551, for TV series is 0.0857, and for books is 0.3796. In the third-level data, the accuracy for movies is 0.3110, for TV series is 0.0837, and for books is 0.5475. Comparing movie recommendations across the three data levels, first-level accuracy exceeds second-level accuracy by 0.2012, a 36.15% decrease. Second-level accuracy, at 0.0441, is higher than third-level accuracy, reflecting a 12.42% decline. For TV series recommendations, first-level accuracy is 0.2001 higher than second-level accuracy, a 70.01% drop. Second-level accuracy is 0.002 higher than third-level accuracy, showing a 2.33% decrease. In book recommendations, first-level accuracy is 0.3175 higher than second-level accuracy, a 45.55% reduction. Second-level accuracy, at 0.0321, is higher than the third level, resulting in an 8.46% decrease. Book recommendations exhibit the highest accuracy among the three levels, while TV series recommendations have the lowest. This analysis indicates that data not trained by the Llama model yields varying recommendation effects across different recommendation domains.

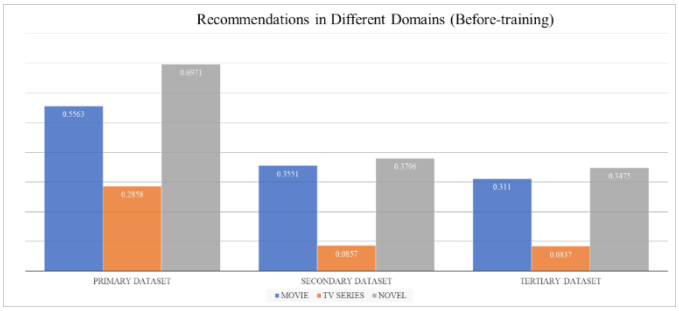


Fig. 5. Comparison of recommendation accuracy in different recommendation domains before training.

Fig. 6 compares recommendation accuracy among first-level, second-level, and third-level data in various recommendation domains post-model training. As illustrated in Fig. 5, in first-level data recommendations, the accuracy for movies is 0.5873; for second-level recommendations, it is 0.2855; and for third-level recommendations, it is 0.7412. In second-level data recommendations, the accuracy for movies is 0.3731; for TV Series, it is 0.1056; and for books, it is 0.48336. Regarding third-level data recommendations, the accuracy for movies is 0.3277; for TV Series, it is 0.0976; and for books, it is 0.5400. Across these three levels of data recommendations, books consistently display the highest accuracy, while TV Series consistently exhibit the lowest accuracy. The analysis above indicates that data trained with the Llama model yields varying recommendation effects in different recommendation domains. However, before and after training, the recommendation accuracy for books remains consistently the highest, while TV Series consistently have the lowest accuracy.

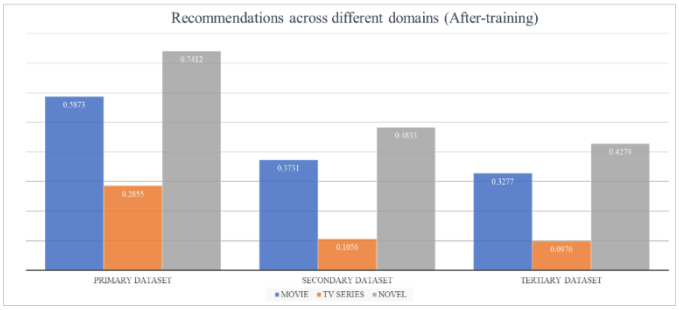


Fig. 6. Comparison of recommendation accuracy across different recommendation domains after training.

3) Analysis of experimental results between different models. To validate the performance of the proposed method, we conduct a comparative analysis by contrasting experimental results with various benchmark models. By comparing recommendation accuracies on different datasets and models, we observe that the model trained on the dataset we construct achieves favorable recommendation outcomes when applied to several other datasets. The experimental

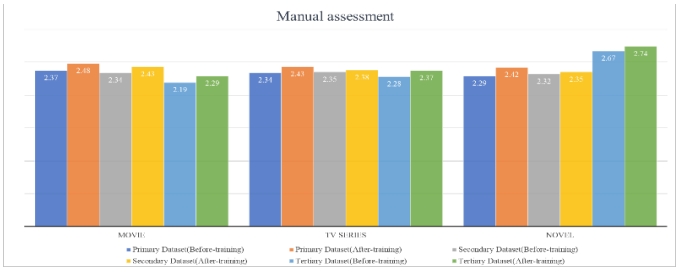
results are presented in Table IV. On dataset ABC, the DistilBERT-FT model performs the best. It achieves optimal recommendation results by cleaning and fine-tuning the ABC corpus. On datasets ML-1M, Amazon Beauty, and Amazon Clothing, the LLMRG model exhibits superior performance. It enhances recommendations by utilizing graph neural networks as additional inputs, based on the generated inference graphs. However, compared to the findings of this study, its performance is slightly inferior. This study establishes comprehensive user profiles, providing more precise descriptions of users, thereby achieving higher recommendation accuracy.

TABLE IV. ACCURACY STATISTICS OF RECOMMENDATION RESULTS AMONG DIFFERENT MODELS

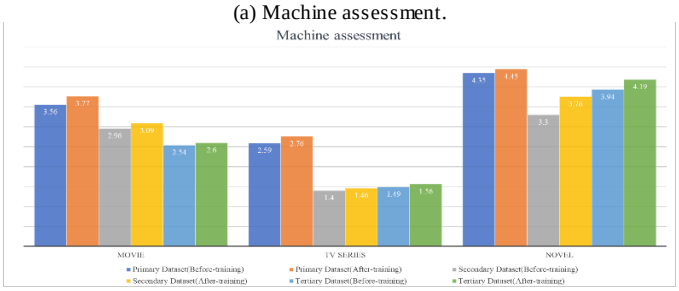
Dataset	Model	Accuracy
ABC	TechNet	0.018
	FastText	0.208
	DistilBERT	0.272
	DistilBERT-FT	0.321
Movie-3	**Our-LLM（Before-training）**	**0.311**
	Our-LLM（After- training）	**0.328**
ML-1M	FDSA	0.091
	BERT4Rec	0.112
	CL4SRec	0.114
	DuoRec	0.201
	LLMRG	0.227
tv-1	**Our-LLM（Before-training）**	**0.286**
	Our-LLM（After- training）	**0.287**
Amazon Beauty	FDSA	0.024
	BERT4Rec	0.020
	CL4SRec	0.040
	DuoRec	0.055
	LLMRG	0.062
tv-2	**Our-LLM（Before-training）**	**0.086**
	Our-LLM（After- training）	**0.106**
Amazon Clothing	FDSA	0.012
	BERT4Rec	0.013
	CL4SRec	0.017
	DuoRec	0.019
	LLMRG	0.021
tv-3	**Our-LLM（Before-training）**	**0.084**
	Our-LLM（After- training）	**0.098**

4) Machine and manual evaluation of recommended results. The comparison between the accuracy of recommendation results calculated manually and those computed by machines highlights the ability to assess the reasonableness of machine-generated recommendations through human evaluation.

Fig. 7 illustrates the comparison between machine evaluation and manual assessment of data across various levels. We manually evaluated one hundred data entries, assigning ratings from 1 to 5 based on original recommended results, Before-training, and After-training data, aiming to assess recommendation reasonableness compared to the original data. Scores were assigned based on perceived appropriateness, and the average score represented the overall manual assessment. In the movie recommendation field at level 1, machine evaluation scores were 3.57 after training and 3.77 after training, indicating a 0.20 increase. For TV Series recommendations, scores increased by 0.17 from 2.59 to 2.76 after training, while for Novel recommendations, they increased by 0.10 from 4.35 to 4.45. On a secondary dataset in the movie recommendation field, machine evaluation scores increased by 0.10 after training from 2.96 to 3.09. For TV Series recommendations, scores increased by 0.06 from 1.40 to 1.46, and for book recommendations, they increased by 0.46 from 3.30 to 3.76. On a three-level dataset in the movie recommendation field, scores increased by 0.06 after training from 2.54 to 2.60. For TV Series recommendations, scores increased by 0.07 from 1.49 to 1.56, and for book recommendations, they increased by 0.64 from 3.94 to 4.58.



(a) Machine assessment.



(b) Manual assessment.

Fig. 7. Machine assessment and manual assessment.

In the realm of movie recommendations, manual assessment scores on a level 1 dataset improved from 2.37 to 2.48 post-training, indicating a 0.11 enhancement. Similarly, for TV series recommendations, scores rose from 2.34 to 2.43, showing a 0.09 increase, while book recommendations experienced a boost from 2.29 to 2.42, marking a 0.13 improvement. Although machine evaluation didn't notably advance after training, manual assessment yielded more reasonable and accurate results. Overall, post-training data resulted in higher machine evaluation scores, indicating more rational recommendations. On a secondary dataset for movie recommendations, manual evaluation scores increased from

2.34 to 2.43 post-training, representing a 0.09 improvement. TV series recommendations saw scores rise from 2.35 to 2.38, a 0.03 improvement, and book recommendations increased from 2.32 to 2.35. In all three domains, post-training data, both in machine and manual evaluations, provided more reasonable and accurate results. However, scores for machine and manual assessment of secondary data were generally lower, possibly due to less smooth integration of personas. On a three-level dataset for movie recommendations, human evaluation scores improved from 2.19 to 2.29 post-training, a 0.10 increase. For TV series recommendations, scores rose from 2.28 to 2.37, a 0.09 improvement, and for book recommendations, scores increased from 2.67 to 2.74, a 0.07 improvement. Overall, third-level data showed higher accuracy and score values across all categories in both machine and human evaluations, offering more reasonable and user-preference-aligned recommendations.

TABLE V. STATISTICAL ANALYSIS OF MACHINE ASSESSMENT (J) BEFORE AND AFTER TRAINING, MANUAL ASSESSMENT (R), AND THE PEARSON CORRELATION COEFFICIENT (P) BETWEEN THE TWO

Recommended field	Dataset level	Before-training			After-training		
		J	*R*	*P*	*J*	*R*	*P*
Movie	First level	3.57	2.37	0.55	3.77	2.48	0.69
	Second level	2.96	2.34	0.59	3.09	2.43	0.69
	Third level	2.54	2.19	0.30	2.60	2.29	0.24
TV Series	First level	2.59	2.34	0.45	2.76	2.43	0.47
	Second level	1.40	2.35	0.53	1.46	2.38	0.51
	Third level	1.49	2.28	0.40	1.56	2.37	0.42
Novel	First level	4.36	2.29	0.60	2.49	2.42	0.62
	Second level	3.30	2.32	0.58	3.76	2.35	0.32
	Third level	3.94	2.67	0.66	4.19	2.74	0.48

The table presented in Table V illustrates the Pearson correlation coefficients between the average scores obtained from machine evaluations and human evaluations, as well as the correlation coefficients for three ranking datasets within three recommended domains. A correlation coefficient (P) of 1 indicates a perfect positive linear relationship, signifying that as one variable increases, the other variable increases proportionally. Conversely, a P value of -1 signifies a perfect negative linear relationship, implying that as one variable increases, the other variable decreases proportionally. A P value of 0 suggests no linear relationship between the two variables.

5) Exploratory experiment. Word frequency statistics and analysis in each recommendation result.

Using distinctive words from the titles of movies, TV Series, and books for statistical analysis, we assess the data based on the frequency of word occurrences. Words such as "a," "an," "the," and other articles are excluded from the titles to ensure the credibility of the evaluation results.

The figure presented in Fig. 8 illustrates the frequency of word occurrences in the top recommendations across three recommended domains for the first-level data after training the Llama model. In Fig. 8(a), the chart displays the maximum 20 words with the highest frequency of occurrence in the recommended movie names after training. Words with occurrences exceeding 200 include 'Godfather' and the movies associated with this word are 'The Godfather' and 'The Godfather Part II.' Importantly, there are differences between the films recommended after training and those recommended without data training. Fig. 8(b) showcases a bar chart representing the word occurrences in the top 20 words found in the recommended TV Series names after training the first-level data. Among these, words with occurrences surpassing 200 include 'abbey', 'bad', 'breaking', 'cards', 'dead', 'downtown', 'game', 'house', 'thrones', and 'walking'. These ten words are associated with TV Series such as 'Downton Abbey', 'Breaking Bad', 'House of Cards', 'The Walking Dead', and 'Game of Thrones'. Popular TV Series remain broadly consistent when comparing the results before and after data training. In Fig. 8(c), the chart represents the top 20 words in the recommended books, showing that the word frequencies after training are higher than before training. The above analysis reveals user preferences as well as the most popular works.

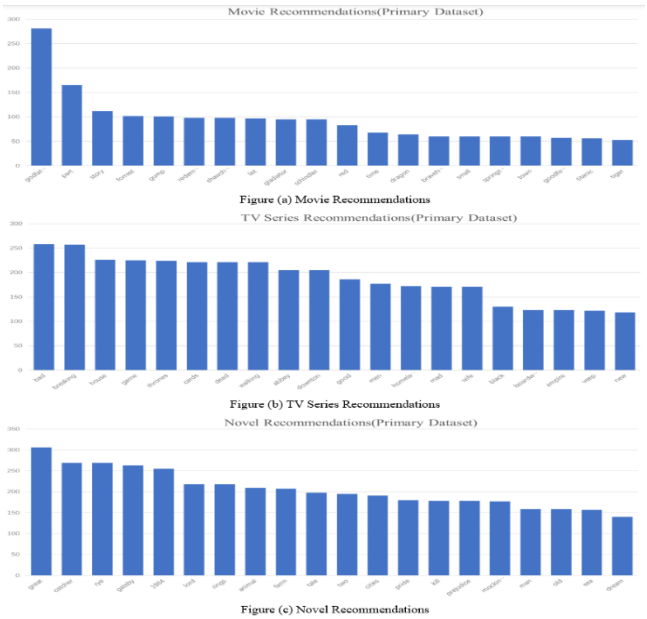


Fig. 8. The word frequency statistics in three domains of data at the after-training level.

The figure presented in Fig. 9 illustrates the word frequency in secondary data recommended across three domains following the training of the Llama model. In Fig. 9(a), the chart displays the top 20 words in movie recommendations, where the term 'godfather' stands out with a frequency of over 1000 times. This indicates a significant popularity for movies like 'The Godfather' and 'The Godfather Part II'. Moving on to Fig. 9(b), the top 20 words in TV Series recommendations are depicted, with six words having frequencies exceeding 2000. Notably, these high-frequency words are associated with three specific TV Series—' Breaking Bad', 'House of Cards' and 'Game of Thrones'. Fig. 9(c) reveals that the top 20 words in book recommendations are linked to widely-read books, with some words representing various

terms referring to the same book, aligning more closely with profiles.

The figure displayed in Fig. 10 illustrates the frequency of word occurrences in the recommendations across three domains for the three-level data following training with the Llama model. In comparison to the first and second-level data, the third-level data offers more precise profile descriptions, and the recommended movies, TV Series, and books are more accurate. The three graphs depicted in Fig. 10 represent the top 20 words with the highest occurrences in each recommendation domain. Each domain's maximum of 20 words highlights the most popular movies, TV Series, and books within their respective domains, facilitating a representative comparison.

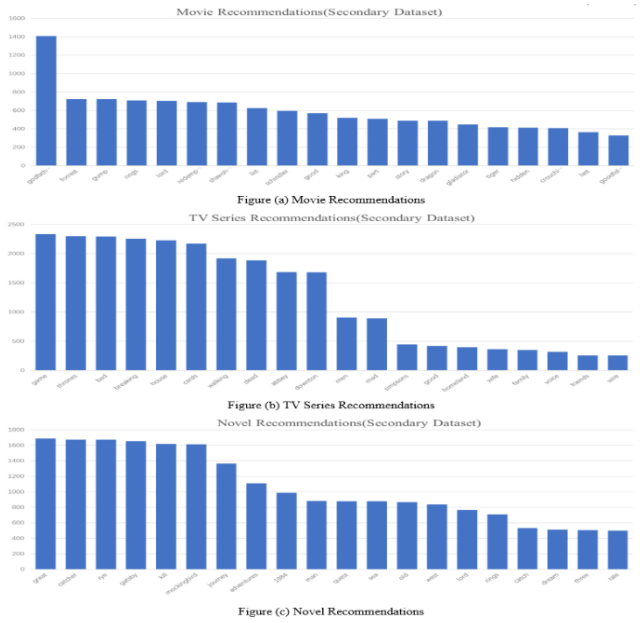


Fig. 9. After training, word frequency statistics were conducted in three domains for secondary data.

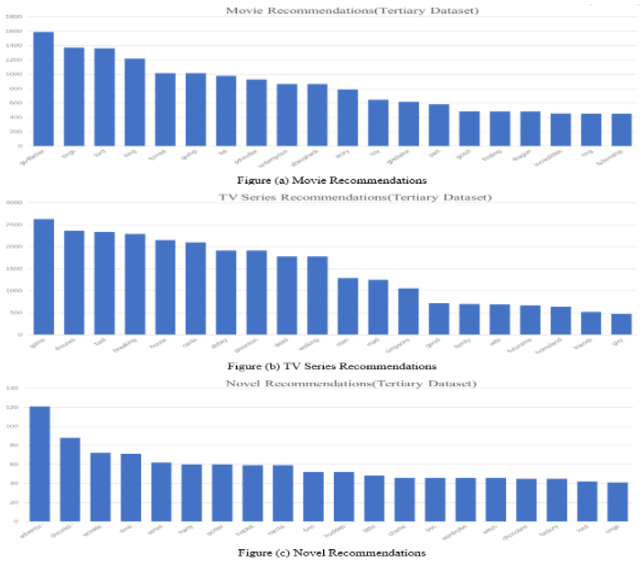


Fig. 10. After training, the word frequency statistics were conducted for three domains in the tertiary-level data.

VI. Conclusion and Prospects

In this paper, we investigate the significant impact of user profiles on recommendation outcomes within the recommendation domain. We employ the Llama model to train on the original dataset. This approach illustrates how Llama brings interpretability to recommendation systems and aligns more closely with user interests without requiring additional information. We conduct experiments using datasets generated by ChatGPT-3.5 in three distinct recommendation domains and across three levels of datasets. Among the mentioned domains, book recommendations show the most promising results, with the first-level dataset yielding the most effective recommendations. In the experiments, we evaluate the effectiveness of recommendation methods based on user profiles. The precision of recommendation results is calculated using cosine similarity, and human evaluators assign scores based on the reasonableness of the recommendations. The higher the human-assigned score, the more reasonable the recommendations, aligning better with user preferences and human values. The future experiments will involve constructing more detailed user profiles, such as socioeconomic status, psychological conditions, etc., and exploring outcomes on other more advanced models, such as LLaMa2.

TABLE VI. Self-Build Dataset Related Data Statistics

Dataset	Recommended field	Dataset level	Data volume
TOTAL	Movie	First level(movie-1)	247
		Second level(movie-2)	28043
		Third level(movie-3)	425431
	TV Series	First level(tv-1)	247
		Second level(tv-2)	28043
		Third level(tv-3)	400157
	Novel	First level(novel-1)	247
		Second level(novel-2)	28043
		Third level(novel-3)	439528

Acknowledgment

Gratitude is extended to all contributors of this article for their assistance, with our work supported by the National Social Science Foundation (Grant No. 22BTQ081).

References

[1] Alabduljabbar R, Alshareef M, Alshareef N. Time-aware Recommender Systems: A Comprehensive Survey and Quantitative Assessment of Literature[J]. IEEE Access, 2023.

[2] MODI P, KUMAR A, KAPOOR B. Filmview: A Review Paper on Movie Recommendation Systems[J]. 2023.

[3] Fiagbe R. Movie Recommender System Using Matrix Factorization[J]. 2023.

[4] Bao K, Zhang J, Wang W, et al. A bi-step grounding paradigm for large language models in recommendation systems[J]. arXiv preprint arXiv:2308.08434, 2023.

[5] Wang L, Lim E P. Zero-Shot Next-Item Recommendation using Large Pretrained Language Models[J]. arXiv preprint arXiv:2304.03153, 2023.

[6] Erritali M, Hssina B, Grota A. Building Recommendation Systems Using the Algorithms KNN and SVD[J]. Int. J. Recent Contributions Eng. Sci. IT, 2021, 9(1): 71-80.

[7] Wang Z, Wang Z, Li X, et al. Exploring multi-dimension user-item interactions with attentional knowledge graph neural networks for recommendation[J]. IEEE Transactions on Big Data, 2022, 9(1): 212-226.

[8] Wu L, He X, Wang X, et al. A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation[J]. IEEE Transactions on Knowledge and Data Engineering, 2022, 35(5): 4425-4445.

[9] Pazzani M, Billsus D. Learning and revising user profiles: The identification of interesting web sites[J]. Machine learning, 1997, 27: 313-331.

[10] Zhang Y C, Blattner M, Yu Y K. Heat conduction process on community networks as a recommendation model[J]. Physical review letters, 2007, 99(15): 154301.

[11] Jayathilaka D K, Kottage G N, Chankuma K C, et al. Hybrid Weight Factorization Recommendation System[C]//2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer). IEEE, 2018: 209-214.

[12] Wu L, Zheng Z, Qiu Z, et al. A Survey on Large Language Models for Recommendation[J]. arXiv preprint arXiv:2305.19860, 2023.

[13] Wang Y, Chu Z, Ouyang X, et al. Enhancing recommender systems with large language model reasoning graphs[J]. arXiv preprint arXiv:2308.10835, 2023.

[14] Bao K, Zhang J, Wang W, et al. A bi-step grounding paradigm for large language models in recommendation systems[J]. arXiv preprint arXiv:2308.08434, 2023.

[15] Jin H, Han X, Yang J, et al. LLM Maybe LongLM: Self-Extend LLM Context Window Without Tuning[J]. arXiv preprint arXiv:2401.01325, 2024.

[16] Patil D D, Dhotre D R, Gawande G S, et al. Transformative Trends in Generative AI: Harnessing Large Language Models for Natural Language Understanding and Generation[J]. International Journal of Intelligent Systems and Applications in Engineering, 2024, 12(4s): 309-319.

[17] Wang B, Wang S, Ouyang Q. Probabilistic Inference Layer Integration in Mistral LLM for Accurate Information Retrieval[J]. 2024.

[18] Koch S, Matveev A, Jiang Z, et al. Abc: A big cad model dataset for geometric deep learning[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 9601-9611.

[19] Sarica S, Luo J, Wood K L. TechNet: Technology semantic network based on patent data[J]. Expert Systems with Applications, 2020, 142: 112995.

[20] Bojanowski P, Grave E, Joulin A, et al. Enriching word vectors with subword information[J]. Transactions of the association for computational linguistics, 2017, 5: 135-146.

[21] Sanh V, Debut L, Chaumond J, et al. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter[J]. arXiv preprint arXiv:1910.01108, 2019.

[22] Meltzer P, Lambourne J G, Grandi D. What's in a Name? Evaluating Assembly-Part Semantic Knowledge in Language Models Through User-Provided Names in Computer Aided Design Files[J]. Journal of Computing and Information Science in Engineering, 2024, 24(1): 011002.

[23] Zhang T, Zhao P, Liu Y, et al. Feature-level Deeper Self-Attention Network for Sequential Recommendation[C]//IJCAI. 2019: 4320-4326.

[24] Sun F, Liu J, Wu J, et al. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer[C]//Proceedings of the 28th ACM international conference on information and knowledge management. 2019: 1441-1450.

[25] Xie X, Sun F, Liu Z, et al. Contrastive learning for sequential recommendation[C]//2022 IEEE 38th international conference on data engineering (ICDE). IEEE, 2022: 1259-1273.

[26] Qiu R, Huang Z, Yin H, et al. Contrastive learning for representation degeneration problem in sequential recommendation[C]//Proceedings of the fifteenth ACM international conference on web search and data mining. 2022: 813-823.