IEEE Open Journal of the
Computer Society

Received 28 September 2024; revised 24 November 2024; accepted 27 November 2024. Date of publication 28 November 2024; date of current version 9 January 2025. The review of this article was arranged by Associate Editor Nandha Kumar Thulasiraman.

Digital Object Identifier 10.1109/OJCS.2024.3509221

CD-LLMCARS: Cross Domain Fine-Tuned Large Language Model for Context-Aware Recommender Systems

ADEEL ASHRAF CHEEMA [1] (Member, IEEE), MUHAMMAD SHAHZAD SARFRAZ [1] (Senior Member, IEEE), USMAN HABIB [2] (Senior Member, IEEE), QAMAR UZ ZAMAN [3], AND EKKARAT BOONCHIENG [4] (Senior Member, IEEE)

[1] Department of Computer Science, National University of Computer and Emerging Sciences, Chiniot-Faisalabad Campus, Chiniot 35400, Pakistan
[2] FAST School of Computing, National University of Computer and Emerging Sciences, Islamabad Campus, Islamabad 44000, Pakistan
[3] Department of Software Engineering, National University of Computer and Emerging Sciences, Chiniot-Faisalabad Campus, Chiniot 35400, Pakistan
[4] Department of Computer Science, Faculty of Science, Chiang Mai University, Chiang Mai 50200, Thailand

CORRESPONDING AUTHORS: Ekkarat Boonchieng; Muhammad Shahzad Sarfraz (e-mail: ekkarat.boonchieng@cmu.ac.th; shahzad.sarfraz@nu.edu.pk).

This work was supported by Fundamental Fund 2025, Chiang Mai University.

ABSTRACT Recommender systems are essential for providing personalized content across various platforms. However, traditional systems often struggle with limited information, known as the cold start problem, and with accurately interpreting a user's comprehensive preferences, referred to as context. The proposed study, CD-LLMCARS (Cross-Domain fine-tuned Large Language Model for Context-Aware Recommender Systems), presents a novel approach to addressing these issues. CD-LLMCARS leverages the substantial capabilities of the Large Language Model (LLM) Llama 2. Fine-tuning Llama 2 with information from multiple domains can enhance the generation of contextually relevant recommendations that align with a user's preferences in areas such as movies, music, books, and CDs. Techniques such as Low-Rank Adaptation (LoRA) and Half Precision Training (FP16) are both effective and resource-efficient, allowing CD-LLMCARS to perform optimally in cold start scenarios. Extensive testing of CD-LLMCARS indicates outstanding accuracy, particularly in challenging scenarios characterized by limited user history data relevant to the cold start problem. CD-LLMCARS offers precise and pertinent recommendations to users, effectively mitigating the limitations of traditional recommender systems.

INDEX TERMS Collaborative filtering, context-aware recommeder systems (CARS), cross-domain recommender systems (CDRS), large language models (LLMs), prompt engineering.

I. INTRODUCTION

Recommender Systems have become an important part of many digital platforms. These systems analyze large amounts of user history and employ smart algorithms to suggest relevant items to users. The objective of recommender systems is to improve user satisfaction and engagement by recommending users with items that match their interests [1]. Recommender systems are used in e-commerce by sites like Amazon [2] to suggest available products that customers are likely to purchase. Amazon's recommendation algorithms may forecast and suggest products aligned with a user's shopping behavior by examining previous purchases, browsing history, and search queries, resulting in increased sales and enhanced customer retention. Likewise, in the entertainment sector, platforms such as Netflix and Spotify have transformed how people discover material. Netflix [3] uses recommendation algorithms to provide films and television programs that align with a user's viewing history and geographical context. By analyzing watching trends, Netflix may generate personalized content recommendations, facilitating viewers' discovery of media that captivates their interest. Spotify [4] employs analogous algorithms to suggest music tracks and playlists that correspond with the user's preferences in genres, artists, and songs. There are several ways to build

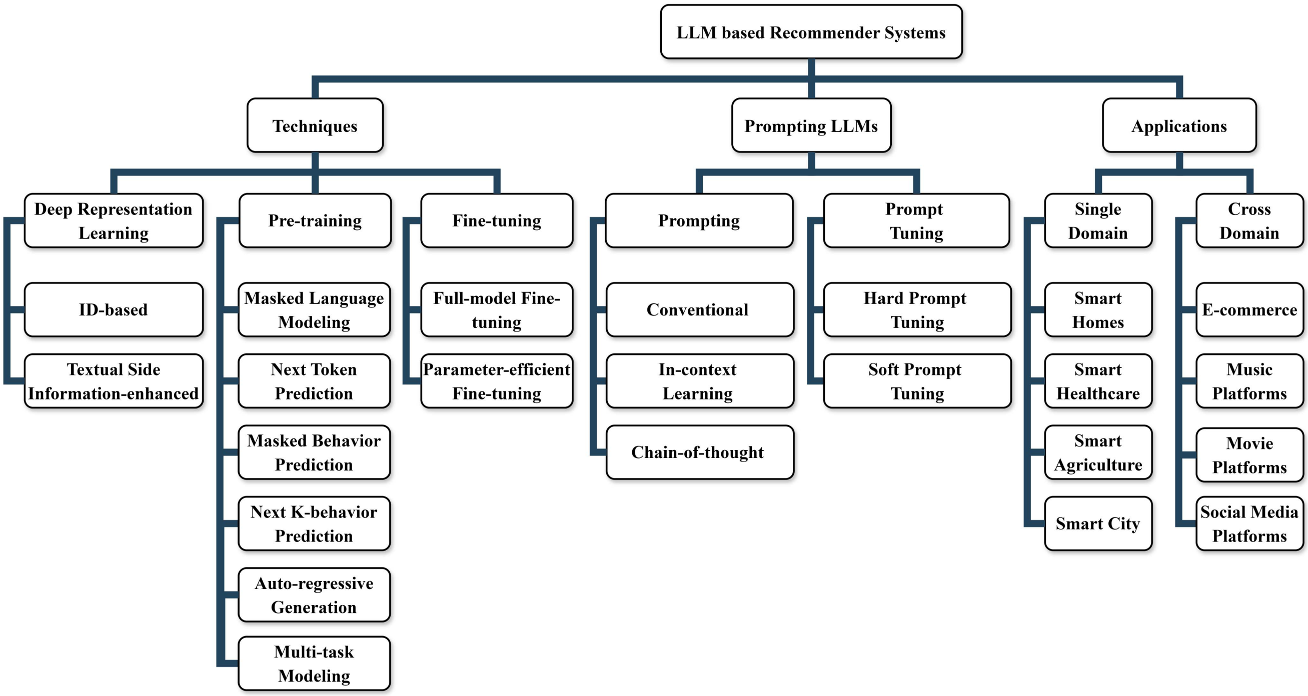

FIGURE 1. Detailed taxonomy of LLM-based recommender systems.

recommender systems. Collaborative filtering [5] looks at user-item ratings to find similarities between users or items, while content-based filtering [6] uses item features to recommend similar items to what the user has liked before. Hybrid recommender systems [7] combine both methods to improve accuracy and relevance. However, single-domain recommender systems often struggle with sparse data [8], making it hard to generate accurate recommendations when there are limited user interactions. They may also miss capturing a user's full range of interests, which leads to less relevant suggestions. To solve these problems, cross-domain recommender systems [9] have emerged as a promising solution. By leveraging data from multiple domains, cross-domain recommender systems can enhance their comprehension of user behavior. This compliments the dataset, mitigating sparse data challenges, while also improving the personalization and diversity of recommendations [10]. A customer's preference for consumption of science fiction content on Netflix may result in suggestions for science fiction literature on Amazon or futuristic-themed restaurants on Yelp, thereby enhancing the overall user experience. Despite these advancements, cross-domain recommender systems continue to face challenges with comprehending the dynamic context of recommendations [11]. Large language models (LLMs) demonstrate significant potential in this regard. Large Language Models (LLMs) are engineered to comprehend, produce, and manipulate human language effectively, owing to transformer architectures, a significant advancement in deep learning. Llama 2 [12], GPT-3 [13], and BERT [14] represent leading large language models characterized by complex architectures that incorporate increased layers, humongous datasets, and advanced training techniques compared to their predecessors. In the realm of recommender systems, large language models (LLMs) can enhance personalization and contextual relevance in recommendations [15]. Through the analysis of natural language data such as titles, reviews, comments, and search queries, LLMs can effectively discern user preferences and dynamic context. Their capacity to generate content enables the formulation of contextually relevant recommendations [16] that correspond with a user's historical interactions and present context, resulting in an enhanced user experience. The integration of LLMs into cross-domain recommender systems presents significant potential, since it addresses the constraints of conventional recommender systems and offers a more comprehensive insight into user preferences across domains. However, general LLMs without fine-tuning may miss important context that affects user preferences, leading to less effective recommendations. Fig. 1 illustrates the comprehensive taxonomy of LLM-based recommender systems, highlighting prospective avenues of evolution in this domain. The efficacy of LLMs in recommender systems can be significantly improved through fine-tuning [17]. Fine-tuning refers to the process of modifying a pre-trained large language model (LLM) for a particular task by training the LLM on a relevant dataset. This method enables the model to enhance its comprehension of the particular task. This study presents a Cross Domain fine-tuned Large Language Model for Context Aware Recommender System (CD-LLMCARS), which fine-tunes LLMs using a dataset that includes user ratings, product metadata, and user history data specific to the target domains. This leads to more precise and context-aware recommendations. Fine-tuned

LLMs are able to recognize subtle preferences and intrinsic contextual details that might be missed otherwise. The key contributions of this research are:

1) Develop a cross-domain recommender system (CD-LLMCARS) using fine-tuned LLMs to generate context-aware recommendations across different domains.
2) Address the challenge of data sparsity, especially in cold-start scenarios, by using LLMs to provide accurate recommendations with limited user interaction data.
3) Evaluate the effectiveness of CD-LLMCARS in various user interaction scenarios (cold and warm) to test its ability to generate personalized and contextually relevant recommendations.
4) Study the impact of different fine-tuning techniques on the performance and accuracy of LLM-based cross-domain recommendations.

The following sections of this article are organized as follows: Section II presents the Literature Review, Section III discusses the Materials and Methods, Section IV explains the Experimental Setup and Implementation, Section V provides the Results and Discussions, and finally, Section VI concludes the article with insights on Conclusion and Future Direction.

II. LITERATURE REVIEW

Recently, literature has emphasized the concept of Large Language Models (LLMs) for generating recommendations, in contrast to conventional approaches that generate and rank potential items. The GenRec framework has enhanced the efficacy of recommendations using LLMs and improved adaptability across various applications [18]. LLMs have proven useful for generating recommendations based on longer interaction histories, due to their superior ability to understand, translate, and converge on contexts [19]. Traditional large language models, on the other hand, are trained on larger knowledge bases and large language corpora. This means that their architecture isn't good for analyzing specific behavioral data and making suggestions in specific situations [20].

The CoLLM framework enhances collaborative information modeling by externalizing existing LLM models, providing both robustness and exceptional flexibility in altering the modeling method. CoLLM excels in enabling LLMs to perform exceptionally well in both warm and cold recommendation settings [21]. Furthermore, AgentCF, another model for collaborative filtering in recommender systems, uses agent-based simulation to model user-item interactions. Unconventionally, it considers both users and items as agents, allowing for the representation of their interactions using LLM-powered agents to record their mutually dependent connections [22].

The USER-LLM framework leverages user embeddings to provide contextualization for LLMs. Self-supervised pretraining on a wide range of user interactions generates these embeddings, revealing hidden user preferences and their changes over time. USER-LLM integrates embeddings into LLMs using cross-attention and soft-prompt techniques, which enables LLMs to flexibly adjust to user contexts [23].

LLM-REC enriches recommendations by supplementing original item descriptions, which frequently lack comprehensive details, with extensive language models. LLM-REC includes additional context, which allows recommendation algorithms to incorporate more intricate details and produce recommendations that more accurately correspond to user preferences [24].

The CHAT-REC system is a significant advancement in the recommendation process, as it effectively resolves the common issue of insufficient item descriptions. The ability of recommendation algorithms to comprehend and anticipate user preferences is often limited by the lack of details in conventional item descriptions, which prevents them from providing accurate recommendations. CHAT-REC enables recommendation engines to capture more complex information by utilizing LLMs to enrich these descriptions with additional context. The significant capacity of LLMs to enhance the performance and accuracy of recommender systems is demonstrated by the generation of recommendations that are more closely aligned with user preferences, which is facilitated by the incorporation of additional layers of information [25].

GPTRec is a generative transformer model for sequential recommendations. It is a big step forward in how recommendation systems handle complex goals that depend on each other. GPTRec has shown significant potential for enhancing recommendation accuracy and efficiency by implementing a new GPU memory-efficient SVD tokenization and a Next-K recommendation generating technique [26]. LLMs have demonstrated exceptional performance in recommendation tasks, particularly in situations where there is limited information about item preferences and only language-based preferences are accessible, such as near cold-start circumstances. Although LLMs are general-purpose models, they achieve comparable performance to fully supervised item-based collaborative filtering (CF) approaches. The competitive performance demonstrated by incorporating both item-based and language-based preferences highlights the potential of LLMs to improve recommender systems, especially in scenarios where standard item-based CF approaches may face challenges [27].

LLM-enhanced models handle the challenges posed by limited implicit feedback and the cold start problem by analyzing user interaction preferences and correcting inaccuracies in item attributes. A denoised augmentation robustification mechanism is implemented to ensure the quality of augmented data [38]. A-LLMRec has been shown to outperform existing collaborative filtering recommender systems, recommenders that include different modalities, and recommenders based on latent linear models across a range of scenarios, including those with new or less popular items, new users, limited data, and various domains. Two significant benefits obtained by fine-tuning neither pre-trained CF-RecSys nor LLMs are model-agnosticism and performance [39].

TABLE 1. A Comparative Overview of Cross-Domain Datasets in Recommender Systems Literature, Highlighting Key Characteristics and Specifications

Dataset	Size	Domain	Values
CiteULike [28]	16,800	Research papers	Likes
Amazon Reviews [29]	Varies	Various products	Reviews
Amazon Ratings [30]	Varies	Various products	Ratings
UCI HAR [31]	10,299	Human Activities	Values
PAMAP2 [32]	385,050	Physical Activities	Values
Epinions [33]	Varies	Various products	Ratings, Reviews
Yelp [34]	Varies	Businesses	Ratings, Reviews
TripAdvisor [35]	Varies	Travel Tourism	Ratings, Reviews
InCar [36]	Varies	In-car HMI	Values
Frappe [37]	26,000	Mobile app usage	Activity

CLLM4Rec efficiently gathers user and item collaborative and content information through language modeling techniques. By employing recommendation-oriented fine-tuning, CLLM4Rec effectively utilizes its pretrained understanding to provide recommendations in an efficient manner [17].

Finally, in TALLRec [40], propose an efficient tuning framework designed to align large language models (LLMs) with recommendation tasks. By employing a two-stage tuning process (Alpaca tuning and rec-tuning) and utilizing LoRA for lightweight adaptation, TALLRec demonstrates strong performance in few-shot learning settings within single-domain recommendation tasks while exhibiting some cross-domain generalization capabilities. Overall, the progress made in recommender systems, especially with the addition of Large Language Models (LLMs), opens up a lot of room for making more advanced, cross-domain, and context-aware recommendation systems. Our suggested method, CD-LLMCARS, aims to use fine-tuned LLMs to get around common problems like lack of data and cold-start issues by making good suggestions with little information about how the user interacts with the system. By dealing with these problems, especially during cold start, CD-LLMCARS can give very specific suggestions across domains, changing based on the needs of both new and existing users. We will also test how well CD-LLMCARS works in a variety of user interaction situations to see how well it can make personalized recommendations that are relevant to the user's actions. Table 1 compares cross-domain datasets used in recommender systems literature, highlighting significant aspects such as dataset size, domain, and value kinds (e.g., ratings, reviews, or activity data). This overview emphasizes the diversity of datasets. These datasets allow the robust evaluation of recommender models across various scenarios.

III. MATERIALS AND METHODS

Cross-domain insights and real-time contextual data are combined in the CD-LLMCARS architecture to make more personalized and dynamic suggestions. It learns about user preferences across multiple domains by using large language models (LLMs). The system can provide recommendations that are both relevant to the user's overall preferences and

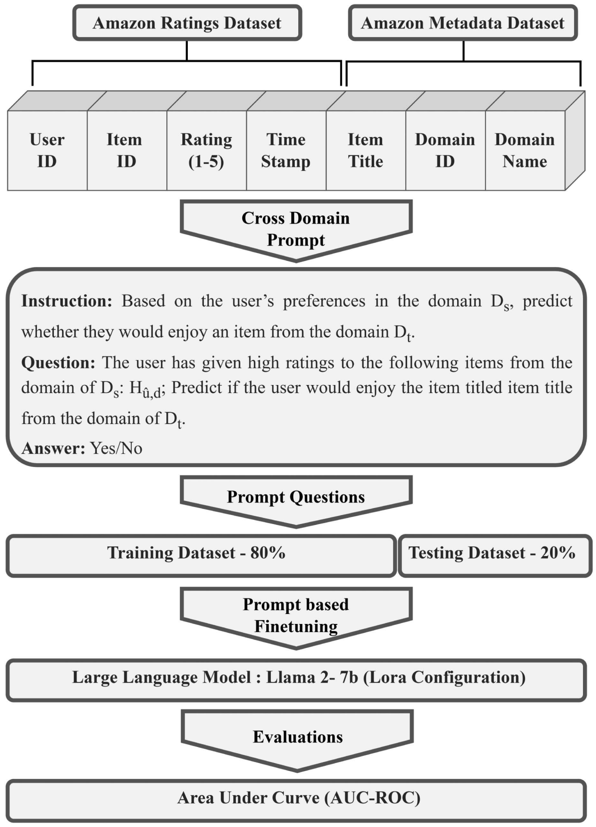

FIGURE 2. Architecture diagram of cross-domain fine-tuned large language model for context-aware recommender systems.

specific to their current context by taking into account factors like metadata and the user's activity.

Cross-domain data and contextual modeling are both used in the CD-LLMCARS architecture. This allows the system to make more accurate predictions by using data from more than one domain. The hybrid approach ensures suggestions that are accurate and diverse, taking into account both past and present user actions. It dynamically enables the system to adjust to users' changing requirements. The Cross-Domain Fine-Tuned Large Language Model for Context-Aware Recommender Systems (CD-LLMCARS) is shown in Fig. 2. The diagram shows how the model combines data about how users interact with different domains and fine-tunes Llama 2 to make personalized and context-aware suggestions, which improves its ability to adapt and be accurate across domains.

A. USER HISTORY

In this study, we focus on overlapping users who have interacted with multiple domains, while the items in these domains remain non-overlapping. This setup introduces complexity as items lack shared domain context. The proposed approach utilizes the historical data of overlapping users from one domain to predict preferences for non-overlapping items in another domain. This enables effective cross-domain knowledge transfer by fine-tuning the model to generalize across disjoint domains.

For the proposed model dataset of four domains is represented as $D \in \{d_1, d_2, d_3, d_4\}$ i.e., books, movies, music, and CDs, respectively. As a source domain D_s has a user set $U \in \{u_1, u_2, u_3, \ldots, u_m\}$ and an item set $I \in \{i_1, i_2, i_3, \ldots, i_n\}$.

The overlapping users are defined between the domains D as $\widehat{U} \in \{U_1 \cap U_2 \cap U_3 \cap U_4\}$. User history for each overlapping user $\widehat{U}$ within source domain D_s, where the item rating is greater than or equal to 4, is represented as $H_{\widehat{u},d} \in \{h_{1,1}, h_{1,2}, h_{1,3}, h_{1,4}, h_{2,1}, \ldots, h_{m,s}\}$. It is pertinent to note that in all domains the data of rating against users and items are sparse. The proposed system's goal is to generate cross-domain recommendations based on the historical data spanning across overlapping users.

B. PROMPT TEMPLATE

In order to ensure that the model can interpret user behavior from one context and apply it effectively to a new set of items, the prompt template [41] serves as a bridge between the source domain and target domain. Through the utilization of data obtained from the source domain, including user interactions, preferences, and past behavior, the model is designed to comprehend the patterns exhibited by the user. The provided information is subsequently combined with the queries pertaining to items within the target domain, enabling the model to produce recommendations that are both pertinent and tailored to the individual. The challenge lies in creating these prompts due to the limited availability of historical data and sparse user-item interactions. Hence, the prompts are categorized into two distinct groups, cold and warm scenarios.

The prompt template has been designed flexibly so that it can be used in cross-domain and recommendation situations. Although the user's history may come from a different domain, like music or books, the template retains its structure so that the LLM can understand the user's situation and make appropriate suggestions. Adaptability is a key part of making a recommendation system that can work across a wide range of content types and is both flexible and scalable.

C. LLM FINE-TUNING

When utilizing pre-trained LLMs for specific tasks, they need to be fine-tuned. In recommendation systems, LLMs need to be fine-tuned to capture domain-specific knowledge. This helps them understand how users and items interact better [42]. Fine-tuning involves training the model on specific recommendation datasets. This lets the LLM improve its knowledge and parameters so it can make better recommendations.

1) LOW-RANK ADAPTATION (LORA)

This study employs Low-Rank Adaptation (LoRA) to fine-tune Llama 2 [43]. Within particular layers of the LLM, LoRA adds additional low-rank trainable parameters, especially within the query, key, and value projection matrices of the transformer's multi-head attention mechanism. This allows the model to fit new tasks without updating most of its pre-trained weights. Two low-rank matrices are added to the layers where matrix multiplies take place by the LoRA method. Initially reducing the dimension to a predefined middle layer, these matrices then restore the original dimension, so mimicking the effects of varying the whole model weights. Targeting just particular modules (such as query, key, and value projections), LoRA reduces the computational overhead, enabling the fine-tuning of LLMs.

2) HALF-PRECISION (FP16) TRAINING

In our fine-tuning pipeline, we incorporate half-precision (FP16) training to further optimize resource usage. FP16 training reduces memory consumption and accelerates computations by storing and processing values in 16-bit floating-point format instead of the default 32-bit precision [44]. This significantly decreases the memory required during training while maintaining model performance.

The combination of LoRA and FP16 training ensures that the model can be fine-tuned with reduced memory requirements and faster execution. The following equation represents the update mechanism under FP16 training:

$$\begin{aligned} \mathrm{Y}_{\mathrm{FP16}} &= X_{\mathrm{FP16}} \cdot \mathrm{doubleDequant}(C1_{\mathrm{FP32}}, C2_{\mathrm{FP8}}, W_{\mathrm{NF4}}) \\ &\quad + X_{\mathrm{FP16}} \cdot L1_{\mathrm{FP16}} \cdot L2_{\mathrm{FP16}} \end{aligned} \tag{1}$$

In the (1) the variable X_{FP16} represents the input in half-precision format, W_{NF4} denotes the quantized weight matrix, and the `doubleDequant` function is used to restore weights from lower precision formats. Additionally, $L1_{\mathrm{FP16}}$ and $L2_{\mathrm{FP16}}$ are the LoRA's low-rank components used in the computation.

3) FREEZING LAYERS

To further optimize the fine-tuning process, we selectively freeze certain layers of the model. Freezing layers reduces the number of trainable parameters, lowering the computational cost and avoiding overfitting [45]. In our case, we freeze the first 31 layers of the Llama 2 7B model, which allows us to focus the fine-tuning process on the higher layers where more task-specific learning can occur. By freezing these deeper layers, we preserve the core knowledge encoded in the pre-trained model while enabling the upper layers to specialize in the task at hand. This strategy balances the need for effective fine-tuning with computational efficiency. Fig. 5 presents the model architecture of CD-LLMCARS, which consists of three key components: prompt construction, input, and output. This structure enables the model to process cross-domain user interactions effectively, generating context-aware recommendations through the fine-tuning of large language models.

D. CROSS-DOMAIN RECOMMENDATION GENERATION

The CD-LLMCARS system employs the fine-tuned Llama 2 model to generate recommendations in the target domain by leveraging the context provided in the prompts. The model infers user preferences in the target domain, incorporating user

Algorithm 1: Fine-Tuning CD-LLMCARS: Cross-Domain Fine-Tuned Large Language Model For Context-Aware Recommender Systems.

Result: Fine-tuned model M_{tuned}
Input : Pretrained model M, Dataset D, Learning rate α, Number of epochs n, Number of layers to freeze f
Output: Fine-tuned model M_{tuned}

Load and preprocess dataset D by tokenizing inputs and applying padding and truncation
Split dataset D into training set D_{train} and evaluation set D_{eval}
Initialize model M with quantization
Initialize tokenizer, setting the pad token to the EoS token
for *each layer in the model up to* f **do**
    param.requires_grad = False
end
Initialize LoRA configuration with dropout and targeted layer projections
Prepare training arguments
for *epoch = 1 to* n **do**
    for *each batch* B_i *in* D_{train} **do**
        Compute predictions $\hat{y}_i = M(x_i)$ for each input $x_i \in B_i$
        Compute loss $\mathcal{L}(\hat{y}_i, y_i)$
        Backpropagate gradients: $\nabla_\theta \mathcal{L}$
        Update model parameters $\theta = \theta - \alpha \nabla_\theta \mathcal{L}$
    end
    if *(epoch % eval_interval == 0)* **then**
        Evaluate the model on D_{eval}
        Compute evaluation metrics (e.g., accuracy, F1-score)
    end
end
Save the fine-tuned model M_{tuned}, tokenizer, and configuration
return M_{tuned}

history in the source domain. This cross-domain recommendation capability is useful in cold and warm scenarios, where user interaction data may be sparse or abundant, respectively. The flow of the proposed system is shown in Algorithm 1.

E. RECEIVER OPERATING CHARACTERISTIC - AREA UNDER THE CURVE (ROC-AUC)

The ROC-AUC (Receiver Operating Characteristic - Area Under the Curve) [46] is a widely used evaluation metric that measures the performance of a classification model by assessing its ability to distinguish between classes. In the case of recommender systems, it is used to determine how well the model can differentiate between relevant and irrelevant recommendations. The ROC curve plots the true positive rate (sensitivity) against the false positive rate (1-specificity) at various threshold settings, providing a visual representation of the trade-off between correctly identifying relevant items and avoiding irrelevant ones. The AUC value is the area under this curve, ranging from 0 to 1. A value of 1 represents perfect classification, meaning the model can perfectly separate relevant from irrelevant items. A value of 0.5 indicates the model performs no better than random guessing, while scores closer to 1 suggest strong discriminative ability. Higher ROC-AUC values signify that the model is making more accurate recommendations, offering a balanced assessment that is particularly useful for imbalanced datasets where the proportion of relevant to irrelevant items may be skewed.

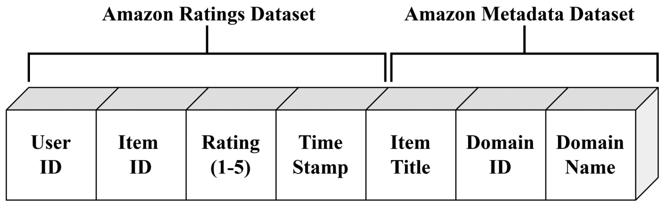

FIGURE 3. Dataset representation for the finetuning of the CD-LLMCARS prompt.

IV. EXPERIMENTAL SETUP AND IMPLEMENTATION

This provides the comprehensive implementation details to fine-tune the Llama 2 model for cross-domain recommendation systems. The core objective of this section is to explain how to leverage large language models to improve the accuracy and relevance of cross-domain recommendations.

A. DATASET

The Amazon Ratings Dataset and the Amazon Ratings Metadata Dataset mentioned in Table 1 are used for this study, which includes user ratings in four domains: Movies, CDs, Digital Music, and Books. The Amazon Ratings metadata dataset was obtained to provide textual representations of items in each domain. The datasets from multiple domains are used to explore user profiling and knowledge transfer in cross-domain by leveraging the extended contextual abilities of fine-tuned Llama 2. Fig. 3 illustrates the dataset representation used for fine-tuning the prompts in CD-LLMCARS. This representation organizes and structures the input data in a way that optimizes the model's ability to capture cross-domain contextual information, enhancing the accuracy of recommendations generated by the model.

1) DATA PREPROCESSING

To prepare the dataset for fine-tuning the Llama 2, rating data is further classified. The items with ratings above 4 given by any user are considered as highly liked items. This classification is fundamental in developing the user history of highly liked items into the prompts. Furthermore, to enhance the contextual capability of the dataset, item titles are merged from the Amazon Ratings Metadata Dataset as shown in Fig. 4 .

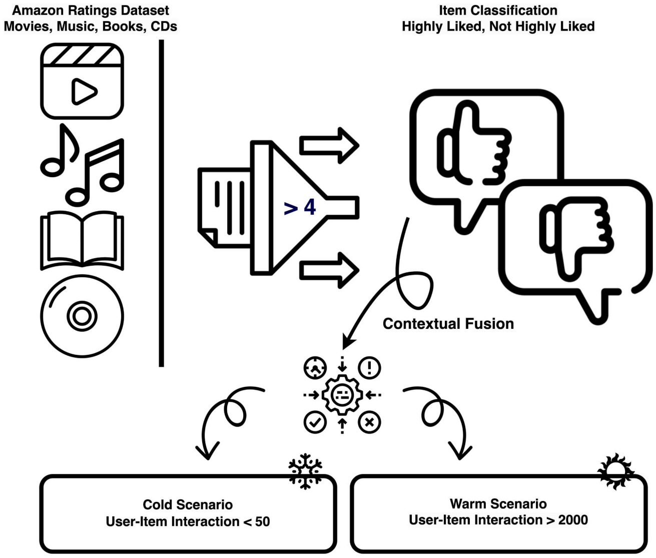

FIGURE 4. Data pre-processing pipeline.

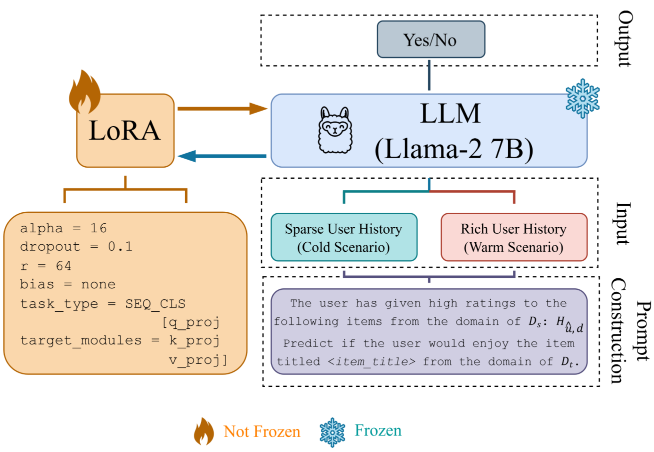

FIGURE 5. Model architecture of CD-LLMCARS, consisting of three major components: Prompt construction, input, and output.

TABLE 2. Amazon Ratings Dataset Characteristics of Overlapped Users in Books, CDs, Movies, and Digital Music Domains

Domain	#Users	#Items	#Ratings
Books	839	2,935,525	102189
CDs	839	544,442	115327
Movies	839	203,970	75829
Digital Music	839	465,392	5449

Moreover, to assess the model's performance under different user-item interaction scenarios, we meticulously divided the dataset into two distinct scenarios, i.e., the Cold Scenario and the Warm Scenario. Table 2 provides the characteristics of the Amazon Ratings Dataset for users who have overlapping interactions across four domains: Books, CDs, Movies, and Digital Music. It outlines the number of users, items, and ratings in each domain, highlighting the diversity of user engagement across different media types, which is crucial for evaluating cross-domain recommendation models like CD-LLMCARS. The scenarios are delineated into cold and warm scenarios.

TABLE 3. A Comparative Overview of Baseline LLMs, Highlighting Key Characteristics and Specifications

Baseline LLMs	# of Params	Layers	Size	Availability
GPT-3 [13]	175 billion	96	350GB	Licensed
BERT (Base) [47]	110 million	12	420MB	Public
T5 (Base) [48]	220 million	12	900MB	Public
RoBERTa (Base) [49]	125 million	12	500MB	Public
XLNet (Base) [50]	110 million	12	450MB	Public
OPT (66B) [51]	66 billion	64	130GB	Public
Llama 2 [12]	**7 billion**	**32**	**13GB**	**Public**

The bold entry establish the fact that we had chosen the highlighed LLM to fine-tune, and the results of our method respectively.

2) COLD SCENARIO

The set of users where the count of user-item interactions is fewer than 50. Such a scenario is indicative of a sparse user history environment. For which the model must rely on limited user-item interaction data to generate recommendations. This is a critical test of the model's ability to infer user preferences with minimal information, a significant challenge in recommender system development.

3) WARM SCENARIO

Contrastingly, the warm scenario contains the set of users where the count of user-item interactions is more than 2000. This scenario represents dense user history where ample user interaction data is available, allowing the model to draw on rich behavioral patterns to enhance recommendation accuracy. The warm scenario is illustrative of active users with extensive histories, enabling the model to perform more sophisticated analyses, including temporal patterns in user behavior. Evaluating the model in this scenario tests its capacity to scale and maintain high performance when handling large volumes of data, crucial for ensuring reliability in real-world applications where user engagement levels can vary widely.

B. LLAMA 2

For this research, the Meta Llama 2 model is used, a variant with 7 billion parameters, due to its robust architecture and capability to handle large-scale language tasks effectively. The model's extensive parameter set enables it to capture intricate patterns and relationships within the data. A comparative analysis of different LLMs is shown in Table 3.

C. CROSS DOMAIN PROMPT GENERATION

To enhance the fine-tuning of the Large Language Model (LLM) for cross-domain recommendations, this study leverages both source and target domains in the prompt construction process. The source domain consists of user history, specifically a set of items from each domain classified as highly liked by the user. The target domain includes items from other domains for which the LLM is tasked with generating recommendations. This is referred to as the answer segment of the LLM prompt. Based on the established user-item interactions, prompts are constructed for each user following their user history in each source domain against each item available in all target domains. The prompt template is as follows:

Instruction: Based on the user's preferences in the domain D_s, predict whether they would enjoy an item from the domain D_t.
Input: The user has given high ratings to the following items from the domain of D_s: $H_{\widehat{u},d}$; Predict if the user would enjoy the item titled item_title from the domain of D_t.
Output: Yes/No

D. EXPERIMENTAL SETUP

The experimental setup for training CD-LLMCARS involved leveraging resource-efficient techniques, including Low-rank Adaptation (LoRA) and half-precision (FP16) training, to optimize computational performance. The model was trained on a dataset containing 3,000 instances over a duration of approximately 16 to 20 hours. The hardware used for training consisted of 4 GPUs with a combined memory capacity of 64 GB, ensuring sufficient resources for handling the model's complexity. While the computational demands of the system are significant, this setup provided a reasonable balance between performance and resource utilization. To enhance inference efficiency, future iterations could incorporate techniques such as caching and computation reuse, which have shown promise in recent research for improving the scalability and runtime of large language models. This setup demonstrates the feasibility of training complex cross-domain recommender systems while maintaining resource efficiency.

V. RESULTS AND DISCUSSION

This section evaluates the performance of CD-LLMCARS against TALLRec [40] and CoLLM-SASRec [21] in terms of their ability to handle recommendations using LLMs. The experiments were conducted under both warm and cold scenarios using a fixed learning rate of 1e-3 across all models. The key evaluation metric was the ROC-AUC, which measures how well a model distinguishes between relevant and irrelevant recommendations, i.e., binary classification.

The use of LoRA and FP16 training not only reduces computational overhead but also makes CD-LLMCARS feasible for deployment in real-world systems. Compared to traditional fine-tuning methods, these techniques enable faster adaptation to diverse datasets, making the approach particularly suitable for large-scale recommendation systems. Additionally, CD-LLMCARS achieves improved accuracy metrics compared to existing state-of-the-art methods such as TALLRec and CoLLM-SASRec, as demonstrated in warm and cold scenarios.

TABLE 4. Warm and Cold Scenario Results Comparison Across TALLRec, CoLLM, and CD-LLMCARS

Model	ROC-AUC
TALLRec (Movie)	67.24
TALLRec (Book)	60.39
TALLRec (Both)	67.59
CoLLM-SASRec (Movie)	72.35
CoLLM-SASRec (Book)	77.46
CD-LLMCARS (Warm)	**79.77**
CD-LLMCARS (Cold)	**75.2**

The bold entry establish the fact that we had chosen the highlighed LLM to fine-tune, and the results of our method respectively.

A. WARM SCENARIO

In the warm scenario, where ample user history is available, CD-LLMCARS achieved the highest performance, with an impressive ROC-AUC of 79.77, as shown in Table 4. This result demonstrates the model's ability to leverage cross-domain data effectively. By fine-tuning large language model Llama 2 across multiple domains such as books, CDs, movies, and digital music, CD-LLMCARS captured intrinsic user preferences and generated more personalized and accurate recommendations. In comparison, TALLRec (Movie) scored 67.24 ROC-AUC, and TALLRec (Book) scored 60.39 ROC-AUC. TALLRec (Both), even when trained on both domains, scored to 67.59 ROC-AUC, showing its limited ability to generalize across domains. Whereas, CoLLM-SASRec, which incorporates sequential learning, performed better than TALLRec, achieving 72.35 ROC-AUC in the movie domain and 77.46 ROC-AUC in the book domain. Despite these improvements, it still falls short of CD-LLMCARS in terms of cross-domain adaptability. The superior performance of CD-LLMCARS in the warm scenario indicates its strong ability to generalize across domains, providing more relevant recommendations by exploiting fine-tuned Llama 2. The results reflect the model's strength in understanding user behavior by incorporating and leveraging the intrinsic context to generate cross-domain recommendations.

B. COLD SCENARIO

The cold scenario is particularly challenging due to the lack of sufficient user history data. Despite this, CD-LLMCARS continued to outperform the other models, achieving a ROC-AUC of 75.2 as shown in Table 4, which, while lower than its warm scenario performance, still shows robustness in sparse data environments. This illustrates the model's capacity to infer meaningful context from minimal data through cross-domain fine-tuning.

In contrast, TALLRec struggled significantly, with TALLRec (Movie) and TALLRec (Book) performing poorly, as expected in cold scenarios. The combined model also saw no

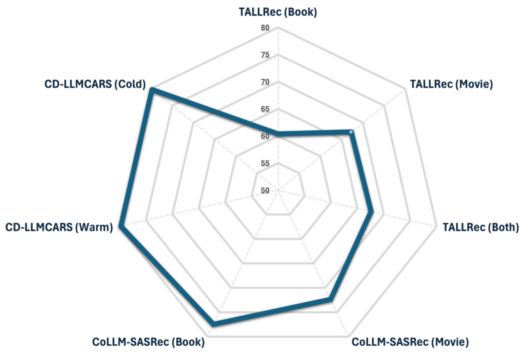

FIGURE 6. **Warm and cold scenario results comparison across TALLRec, CoLLM, and CD-LLMCARS.**

significant improvement. CoLLM-SASRec, while performing better than TALLRec, could not reach the same level of cold-start adaptability as CD-LLMCARS.

These results highlight that CD-LLMCARS's fine-tuning process is particularly effective in addressing data sparsity issues. The model leverages cross-domain knowledge to compensate for the sparsity of user history in cold situations, making it ideal for real-world applications where such conditions are common.

C. CROSS-DOMAIN LEARNING

CD-LLMCARS benefits from its ability to transfer knowledge across domains, which is crucial for improving recommendation accuracy. By fine-tuning Llama 2 with multiple datasets, the model gains a more comprehensive understanding of user preferences. Comparisons between LLM-based and non-LLM-based cross-domain recommender systems are difficult due to their distinct approaches—non-LLM methods rely on numerical patterns, while LLMs use semantic understanding and contextual prompts. These differences highlight LLMs' unique ability to handle complex contextual and cross-domain challenges. TALLRec shows limited performance improvements when working across multiple domains. Its single-domain focus restricts its ability to generalize, especially in cold conditions, where it performs worse than other models. While CoLLM-SASRec outperforms TALLRec, particularly in single-domain scenarios, its performance does not scale as effectively as CD-LLMCARS when cross-domain interactions are introduced.

The results clearly demonstrate that CD-LLMCARS is the most robust model across both warm and cold scenarios, as shown in Fig. 6. It consistently outperforms TALLRec and CoLLM-SASRec, achieving higher ROC-AUC scores in both scenarios.

D. PERFORMANCE EVALUATION

The ability of CD-LLMCARS to handle cross-domain data and sparse user interactions makes it highly suitable for real-world recommendation systems. The model's ability to maintain high performance in challenging cold situations further underscores its effectiveness in generating contextually relevant and personalized recommendations. CD-LLMCARS demonstrates practical applicability in real-world scenarios such as personalized recommendations on e-commerce platforms and media streaming services. By leveraging LoRA and FP16 training, the model significantly reduces computational overhead as shown in Table 5, enabling efficient deployment in large-scale systems. Compared to traditional methods, CD-LLMCARS offers enhanced scalability and accuracy by integrating contextual and cross-domain knowledge. This makes it particularly effective in environments with diverse user preferences and sparse data.

TABLE 5. **Hardware Configurations, Training, and Inference Times for CD-LLMCARS**

Hardware Configurations	
GPU	4 (NVIDIA A2)
Memory	64 GB
Training Data	
Number of Prompts	3000
Train Time	
Cold Scenario	600 min (avg)
Warm Scenario	960 min (avg)
Inference Time	
Cold Scenario	322 s (avg)
Warm Scenario	326 s (avg)

VI. CONCLUSION AND FUTURE DIRECTION

CD-LLMCARS is a cross-domain fine-tuned large language model (LLM) designed to provide accurate, context-aware recommendations by leveraging user interaction data across multiple domains. The system effectively addresses key challenges in traditional recommender systems, such as sparse data and cold-start problems, by employing advanced techniques like Low-rank Adaptation (LoRA) and half-precision (FP16) training. These optimizations not only enhance recommendation accuracy but also ensure resource efficiency, making CD-LLMCARS well-suited for large-scale applications. Its ability to generalize across diverse domains—such as books, movies, CDs, and digital music—offers a scalable and personalized user experience. The model consistently outperforms baselines like TALLRec and CoLLM, achieving high ROC-AUC scores and delivering relevant recommendations in both warm and cold-start scenarios.

A standout feature of CD-LLMCARS is its capacity to effectively capture contextual information, overcoming the limitations of conventional cross-domain systems and enabling practical solutions for real-world use cases. However, the model's reliance on overlapping domain data can limit its applicability in scenarios where such data is unavailable. Furthermore, while its Yes/No classification framework is effective, it lacks a ranking mechanism, which restricts its direct applicability to ranking-based systems commonly used in recommendations.

Several opportunities exist for further refinement and research. Integrating more complex user behavior patterns, such as sequential and temporal dynamics, could allow the model to better capture evolving preferences over time. Improvements in prompt engineering and adaptive learning techniques may also enhance the generation of contextually rich recommendations. Expanding CD-LLMCARS to less-related domains, such as healthcare, education, or travel, could broaden its scope and provide insights into cross-domain interactions in novel contexts. Exploring hybrid models, such as combining LLMs with graph-based systems, could reveal deeper user-item relationships, further improving recommendation accuracy.

Future work could also focus on optimizing CD-LLMCARS for real-time recommendations in dynamic, large-scale environments. Techniques like transfer learning between closely and distantly related domains may address data sparsity issues more effectively, while multi-modal data integration—incorporating text, images, and audio—has the potential to enrich the recommendation process and enhance the user experience. Additionally, the development of standardized evaluation metrics to compare LLM-based and traditional recommender systems would provide clearer insights into their relative strengths and weaknesses, bridging the gap between these paradigms.

From an ethical perspective, the use of cross-domain user data raises concerns about privacy and security. Addressing these challenges through federated learning, privacy-preserving mechanisms, and compliance with data protection regulations is essential to ensuring responsible data usage. Enhancing transparency and fairness in recommendations will also play a crucial role in building user trust.

In summary, CD-LLMCARS represents a significant advancement in cross-domain recommender systems, leveraging LLM fine-tuning to deliver accurate, scalable, and contextually aware recommendations. While limitations remain, such as reliance on overlapping domain data and the absence of a ranking mechanism, future research and enhancements promise to address these gaps. With its strong foundation and potential for improvement, CD-LLMCARS has the capacity to revolutionize personalized recommendations across a wide range of domains.

REFERENCES

[1] F. O. Isinkaye, Y. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informat. J.*, vol. 16, pp. 261–273, 2015.

[2] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76–80, Jan./Feb. 2003.

[3] C. A. Gomez-Uribe and N. Hunt, "The netflix recommender system: Algorithms, business value, and innovation," *ACM Trans. Manage. Inf. Syst.*, vol. 6, no. 4, pp. 1–19, 2015.

[4] K. Jacobson, V. Murali, E. Newett, B. Whitman, and R. Yon, "Music personalization at spotify," in *Proc. 10th ACM Conf. Recommender Syst.*, 2016, pp. 373–373.

[5] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web: Methods and Strategies of Web Personalization*. Berlin, Germany: Springer, 2007, pp. 291–324.

[6] J. M. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web: Methods and Strategies of Web Personalization*. Berlin, Germany: Springer, 2007, pp. 325–341.

[7] R. Burke, "Hybrid recommender systems: Survey and experiments," *User Model. User-Adapted Interaction*, vol. 12, pp. 331–370, 2002.

[8] N. Idrissi and A. Zellou, "A systematic literature review of sparsity issues in recommender systems," *Social Netw. Anal. Mining*, vol. 10, no. 1, 2020, Art. no. 15.

[9] I. Fernández-Tobías, I. Cantador, M. Kaminskas, and F. Ricci, "Cross-domain recommender systems: A survey of the state of the art," in *Proc. Spanish Conf. Inf. Retrieval*, 2012, pp. 919–959.

[10] T. Zang, Y. Zhu, H. Liu, R. Zhang, and J. Yu, "A survey on cross-domain recommendation: Taxonomies, methods, and future directions," *ACM Trans. Inf. Syst.*, vol. 41, no. 2, pp. 1–39, 2022.

[11] F. Zhu, Y. Wang, C. Chen, J. Zhou, L. Li, and G. Liu, "Cross-domain recommendation: Challenges, progress, and prospects," in *Proc. 30th Int. Joint Conf. Artif. Intell., IJCAI 2021*, 2021, pp. 4721–4728.

[12] H. Touvron et al., "Llama 2: Open foundation and fine-tuned chat models," 2023, *arXiv:2307.09288*.

[13] L. Floridi and M. Chiriatti, "GPT-3: Its nature, scope, limits, and consequences," *Minds Mach.*, vol. 30, pp. 681–694, 2020.

[14] Y. Hao, L. Dong, F. Wei, and K. Xu, "Visualizing and understanding the effectiveness of BERT," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, pp. 4143–4152, 2019.

[15] Z. Zhao et al., "Recommender systems in the era of large language models (LLMs)," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 11, pp. 6889–6907, 2024.

[16] Y. Zhang et al., "Language models as recommender systems: Evaluations and limitations," in *Proc. NeurIPS 2021 Workshop on I (Still) Can't Believe It's Not Better*, 2021.

[17] Y. Zhu, L. Wu, Q. Guo, L. Hong, and J. Li, "Collaborative large language model for recommender systems," in *Proc. ACM Web Conf.*, 2024, pp. 3162–3172, 2024.

[18] J. Ji et al., "GenRec: Large language model for generative recommendation," in *Eur. Conf. Inf. Retrieval*, 2024, pp. 494–502.

[19] Z. Zheng, W. Chao, Z. Qiu, H. Zhu, and H. Xiong, "Harnessing large language models for text-rich sequential recommendation," in *Proc. ACM Web Conf.*, 2024, pp. 3207–3216.

[20] Z. Chu et al., "Leveraging large language models for pre-trained recommender systems," in *Proc. 17th ACM Conf. Recommender Syst.*, 2023, pp. 1096–1102.

[21] Y. Zhang et al., "CoLLM: Integrating collaborative embeddings into large language models for recommendation," 2023, *arXiv:2310.19488*.

[22] J. Zhang et al., "AgentCF: Collaborative learning with autonomous language agents for recommender systems," in *Proc. ACM Web Conf.*, 2024, pp. 3679–3689.

[23] L. Ning et al., "User-LLM: Efficient LLM contextualization with user embeddings," 2024, *arXiv:2402.13598*.

[24] H. Lyu, S. Jiang, H. Zeng, Y. Xia, and J. Luo, "LLM-Rec: Personalized recommendation via prompting large language models," *Findings Assoc. Comput. Linguistics: NAACL 2024*, pp. 583–612, 2024, doi: 10.18653/v1/2024.findings-naacl.39.

[25] Y. Gao, T. Sheng, Y. Xiang, Y. Xiong, H. Wang, and J. Zhang, "Chat-rec: Towards interactive and explainable llms-augmented recommender system," 2023, *arXiv:2303.14524*.

[26] V. A. Petrov and C. Macdonald, "Generative sequential recommendation with GPTRec," 2023, *arXiv:2306.11114*.

[27] S. Sanner, K. Balog, F. Radlinski, B. Wedin, and L. Dixon, "Large language models are competitive near cold-start recommenders for language-and item-based preferences," in *Proc. 17th ACM Conf. Recommender Syst.*, 2023, pp. 890–896.

[28] A. Harpale, Y. Yang, S. Gopal, D. He, and Z. Yue, "Citedata: A new multi-faceted dataset for evaluating personalized search performance," in *Proc. 19th ACM Int. Conf. Inf. Knowl. Manage.*, 2010, pp. 549–558.

[29] P. Keung, Y. Lu, G. Szarvas, and N. A. Smith, "The multilingual amazon reviews corpus," *Proc. 2020 Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 4563–4568.

[30] J. Ni, J. Li, and J. McAuley, "Justifying recommendations using distantly-labeled reviews and fine-grained aspects," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 188–197.

[31] N. Sikder and A.-Al Nahid, "KU-HAR: An open dataset for heterogeneous human activity recognition," *Pattern Recognit. Lett.*, vol. 146, pp. 46–54, 2021.

[32] A. Reiss, "PAMAP2 physical activity monitoring," *UCI Mach. Learn. Repository*, 2012, doi: 10.24432/C5NW2H.

[33] J. Tang, H. Gao, and H. Liu, "mTrust: Discerning multi-faceted trust in a connected world," in *Proc. 5th ACM Int. Conf. Web Search Data Mining*, 2012, pp. 93–102.
[34] N. Asghar, "Yelp dataset challenge: Review rating prediction," 2016, *arXiv:1605.05362*.
[35] M. Nilashi, O. Ibrahim, E. Yadegaridehkordi, S. Samad, E. Akbari, and A. Alizadeh, "Travelers decision making using online review in social network sites: A case on tripadvisor," *J. Comput. Sci.*, vol. 28, pp. 168–179, 2018.
[36] R. P. Nelson et al., "MoLa InCar AR: Dataset for action recognition," 2022, doi: 10.34622/datarepositorium/1S8QVP.
[37] L. Baltrunas, K. Church, A. Karatzoglou, and N. Oliver, "Frappe: Understanding the usage and perception of mobile app recommendations in-the-wild," *CoRR abs/1505.03014*, 2015.
[38] W. Wei et al., "LLMRec: Large language models with graph augmentation for recommendation," in *Proc. 17th ACM Int. Conf. Web Search Data Mining*, 2024, pp. 806–815.
[39] S. Kim, H. Kang, S. Choi, D. Kim, M. Yang, and C. Park, "Large language models meet collaborative filtering: An efficient all-round LLM-based recommender system," in *Proc. 30th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2024, pp. 1395–1406.
[40] K. Bao, J. Zhang, Y. Zhang, W. Wang, F. Feng, and X. He, "TALLRec: An effective and efficient tuning framework to align large language model with recommendation," in *Proc. 17th ACM Conf. Recommender Syst.*, 2023, pp. 1007–1014.
[41] K. Lyu, H. Zhao, X. Gu, D. Yu, A. Goyal, and S. Arora, "Keeping LLMs aligned after fine-tuning: The crucial role of prompt templates," in *Proc. ICLR 2024 Workshop Reliable Responsible Found. Models*, 2024, *arXiv:2402.18540*.
[42] L. Zhang, P. Liu, Y. Deldjoo, Y. Zheng, and J. A. Gulla, "Understanding language modeling paradigm adaptations in recommender systems: Lessons learned and open challenges," 2024, *arXiv:2404.03788*.
[43] W. Xia, C. Qin, and E. Hazan, "Chain of LoRa: Efficient fine-tuning of language models via residual learning," 2024, *arXiv:2401.04151*.
[44] K. VM, et al., "Fine tuning LLM for enterprise: Practical guidelines and recommendations," *CoRR*, 2024, doi: 10.48550/ARXIV.2404.10779.
[45] Q. Liu, N. Chen, T. Sakai, and X.-M. Wu, "Once: Boosting content-based recommendation with both open-and closed-source large language models," in *Proc. 17th ACM Int. Conf. Web Search Data Mining*, 2024, pp. 452–461.
[46] F. H. D. Olmo and E. Gaudioso, "Evaluation of recommender systems: A new approach," *Expert Syst. Appl.*, vol. 35, no. 3, pp. 790–804, 2008.
[47] J. Devlin, M.-W. Chang, K. Lee, and K. N. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. naacL-HLT*, vol. 1, 2018, Art no. 2.
[48] J. Ni et al., "Sentence-T5: Scalable sentence encoders from pre-trained text-to-text models," *Findings Assoc. Comput. Linguistics: ACL 2022*, pp. 1864–1874, 2022, *arXiv:2108.08877*.
[49] Y. Liu et al., "RoBERTa: A robustly optimized bert pretraining approach," 2019, *arXiv:1907.11692*.
[50] Z. Yang, "XLNet: Generalized autoregressive pretraining for language understanding," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, Art. no. 517.
[51] S. Zhang et al., "OPT: Open pre-trained transformer language models," 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:248496292

ADEEL ASHRAF CHEEMA (Member, IEEE) is currently working toward the Ph.D. degree with the FAST National University of Computer and Emerging Sciences Chiniot -Faisalabad Campus, Chiniot, Pakistan. He is also an Assistant Professor with the Department of Computer Science, FAST National University of Computer and Emerging Sciences. His research interests include recommender systems and information retrieval techniques.

MUHAMMAD SHAHZAD SARFRAZ (Senior Member, IEEE) is currently a Professor with the School of Computing and Director with the NUCES Chiniot-Faisalabad Campus, Chiniot, Pakistan. His international experience includes, a Visiting Researcher with the Digital Image Processing Lab, National Institute of Informatics (NII), Tokyo, Japan. He is an academic specializing in computer science and geospatial technologies, with a focus on ICT tools for health, agriculture, and location-based systems. His work emphasizes the use of web-based and open-source tools for Remote Sensing and Geographical Information Systems (GIS). Dr. Shahzad is a Member of professional organizations such as ACM, ISPRS, Canadian Remote Sensing Society, and the International Association of Chinese Professionals in GIS. He was a resource person for the World Health Organization (WHO) and Pak One Health Alliance, conducting workshops on analytical skills and information utilization.

USMAN HABIB (Senior Member, IEEE) received the master's degree from the Norwegian University of Science and Technology (NTNU), Trondheim, Norway, and the Ph.D. degree from the ICT Department, Technical University of Vienna, Vienna, Austria. He was with the Ghulam Ishaq Khan Institute of Engineering Sciences and Technology (GIKI), Swabi, Pakistan, and COMSATS University, Abbottabad Campus, Abbottabad, Pakistan. With more than 18 years of teaching and research experience, since 2006, he has been completing various industrial projects along with academia. He is currently an Associate Professor and Head of the Software Engineering Department, FAST School of Computing, National University of Computer and Emerging Sciences (NUCES), Islamabad, Pakistan. He also actively engages in research and has authored numerous conference and journal publications. His research interests include machine learning, data analytics, pattern recognition, security, and medical image processing.

QAMAR UZ ZAMAN received the M.S. degree in software engineering from the University of Engineering and Technology (UET), Taxila, Pakistan, and the Ph.D. degree from the Capital University of Science and Technology (CUST), Islamabad, Pakistan. He was a Senior Lecturer with the Department of Software Engineering, CUST, and Lecturer with COMSATS University, Abbottabad Campus, Abbottabad, Pakistan. In 2022, he joined the FAST National University of Computer and Emerging Sciences (NUCES), Chiniot-Faisalabad (CFD) Campus, Chiniot, Pakistan, as an Assistant Professor. He has authored multiple papers in respected journals and conferences. His research interests include software verification and validation, natural language processing and expert systems.

EKKARAT BOONCHIENG (Senior Member, IEEE) received the B.S. degree in computer science from Khon Kaen University, Khon Kaen, Thailand, the M.S. degree in computer science from the University of New Haven, West Haven, CT, USA, and the Ph.D. degree in computer science from the Illinois Institute of Technology, Chicago, IL, USA. He is currently an Associate Professor of computer science with Chiang Mai University, Chiang Mai, Thailand, and Director of the Center of Excellence in Community Health Informatics, CMU. His research interests include computer networks, image processing, data science and biomedical engineering.