Understanding User Behavior in Cross-Domain Recommendation: An LLM-Based Approach

Dipak Falgun Meher, Ajay Krishna Vajjala, and David S. Rosenblum
Department of Computer Science
George Mason University
Fairfax, VA, USA
{dmeher, akrish, dsr}@gmu.edu

2025 International Joint Conference on Neural Networks (IJCNN) | 979-8-3315-1042-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/IJCNN64981.2025.11228754

***Abstract*—Cross-domain recommendation (CDR) has emerged as a promising solution to address the cold-start and sparsity issues faced by single-domain recommender systems. Users often exhibit varying interests and rating behaviors across domains, such as rating items in the Movies domain differently than in the Books domain. In this work, we empirically analyze whether state-of-the-art CDR algorithms make significantly better recommendations in a target domain when a user's rating behavior is consistent across domains. We propose a novel approach leveraging Large Language Models (LLMs) to quantify a consistency value that measures how consistently a user rates items across different domains. Our empirical analysis reveals that the performance of state-of-the-art CDR models does not consistently correlate with user behavior consistency across domain pairs, indicating limitations in their ability to effectively leverage this factor. These findings highlight the need for CDR algorithms that better utilize user behavior consistency to enhance recommendation performance.**

***Index Terms*—Large Language Models, Recommendation**

I. Introduction

As the availability of online information grows, users often feel overwhelmed by the sheer volume of content [1]–[4]. This overloads users, making it harder to find relevant content and increases dissatisfaction, which can lead to customer loss for companies [5]. To tackle these challenges, companies and e-commerce platforms use recommender systems to suggest items that match users' interests [6]–[8]. However, many of these systems struggle with the 'cold-start' problem, where they cannot provide recommendations to new users [9]. They also face the 'sparsity' problem, where insufficient user interactions hinder accurate recommendations for users [10].

Cross-domain recommendation (CDR) has recently emerged as an effective solution to address cold-start and sparsity issues by transferring knowledge from a source domain (e.g., Books) to a target domain (e.g., Movies) [1], [2], [11], [12]. While much of the research in this area has focused on developing advanced algorithms to enhance recommendation performance [13]–[15], less attention has been paid to how CDR systems perform when user behavior varies between domains. For example, users may exhibit consistent preferences across similar domains but behave differently when engaging with diverse ones. Recent findings by Krishna Vajjala et al. [16] indicate that CDR models do not always achieve statistically significant improvements in recommendation quality for similar domain combinations compared to dissimilar ones. These results underscore the importance of understanding how domain similarity and user behavior influence CDR performance.

In cross-domain recommendation, recommendations are driven by user rating behavior and how effectively this behavior translates across domains [11]. For instance, when transferring ratings from the Books domain to the Movies domain, it is intuitive to expect consistency in user ratings, as both domains often share attributes like narrative structures, genres, and themes [2]. A user who enjoys mystery novels, for example, is likely to appreciate mystery films due to similar plot elements and emotional engagement. However, transferring ratings from the Books domain to the Hotels domain presents a different scenario. These domains represent fundamentally distinct experiences: books are assessed based on elements like storytelling and character development, whereas hotels are rated based on amenities, location, and comfort. Such differences diminish the likelihood of consistent user preferences across these domains [16].

This leads to the question: "How do state-of-the-art CDR algorithms perform when users rate items consistently across domains versus inconsistently?" In this paper, we address this question by introducing a novel method to quantify user behavior consistency across domains. We then empirically evaluate its impact on CDR performance. To our knowledge, this is the first study to systematically quantify user behavior consistency in CDR and analyze how it influences algorithm performance under varying conditions of consistency.

To measure user rating consistency across domains, we leverage the reasoning capabilities of Large Language Models (LLMs) [17]. By utilizing user ratings, reviews, and textual data, we quantify consistency and evaluate how state-of-the-art CDR algorithms perform under consistent versus inconsistent user behavior. Our analysis reveals that CDR performance does not consistently correlate with user behavior consistency across domain pairs, highlighting a critical limitation in leveraging this factor effectively.

The findings of this study have important implications for real-world recommender systems. As CDR continues to be deployed in a variety of contexts, from e-commerce to other platforms, understanding how user behavior varies between domains becomes increasingly relevant. Our findings reveal that even advanced CDR models struggle to leverage consis-

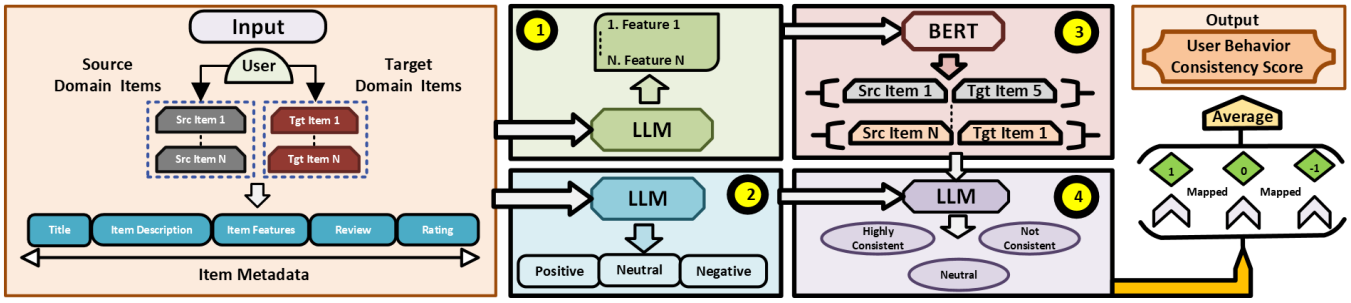

Fig. 1. Framework for Computing User Behavior Consistency: Overlapping users from the source and target domains are used as input. Item metadata (Title, Description, Features, Review, and Rating) is processed to generate item features. The framework includes four steps: 1. Item Feature Extraction: Common aspects between domains are extracted using an LLM. 2. User Sentiment Extraction: Sentiment is categorized as Positive, Neutral, or Negative. 3. Similar Item Extraction: BERT generates embeddings, and similarity is calculated to identify the most similar items across domains. 4. Behavior Consistency Calculation: User behavior is classified as Highly Consistent, Neutral, or Not Consistent (mapped to 1, 0, and -1), with scores averaged to compute overall consistency.

tent user behavior to improve recommendations, highlighting the need for further exploration. By addressing this gap, our work contributes to developing more reliable and adaptable CDR systems. We make our code and data are available via our GitHub repository [18].

To summarize, our contributions are as follows:

- We present a novel method for quantifying user behavior consistency across domains using LLMs.
- To the best of our knowledge, this is the first study to analyze the impact of user rating consistency across domains on cross-domain recommendation performance.
- We conduct a comprehensive empirical evaluation across 12 domain combinations and 4 state-of-the-art CDR algorithms to assess their performance when user behavior is consistent versus inconsistent.

II. Quantifying User Behavior Consistency

We present a method to quantify the consistency of user behavior when interacting with items across different domains. User behavior is inherently dynamic [19], as it changes based on the nature of the object being interacted with. This variability is amplified when the user shifts between domains. To address this challenge, we introduce a novel framework that leverages LLMs [20] to evaluate consistency in user behavior across domains. Figure 1 details the components of our approach. In the following sections, we provide more details about each component and how they work together to quantify user behavior consistency across domains.

There are four main components to our framework:

1) **Item Feature Extractor**: Generates item features from user reviews and metadata (e.g., titles, descriptions).
2) **Similar Item Pair Extractor**: Identifies similar item pairs across domains using embeddings to evaluate shared key similarities.
3) **Sentiment Extractor**: Extracts user sentiment from reviews and metadata for richer insights.
4) **Behavior Consistency Extractor**: Quantifies user behavior consistency across domains as Highly-consistent (1), Neutral (0), or Not-consistent (-1).

A. Item Feature Extractor

The first component in our framework is the Item Feature Extractor. This is a key component of our framework, which leverages LLMs to extract item features from user reviews and metadata (See Figure 1). The goal is to identify commonalities between domain pairs, such as Books-Movies and Movies-Music, as well as contrasting pairs like Books-Food and Electronics-Food. Figure 2 outlines the comparable aspects we have identified to extract common item features from these items. In this work, we used the Amazon Reviews dataset [21], and the domain combinations mentioned throughout the paper correspond to the domains within this dataset.

1) Reasoning for Feature Selection Across Domain Pairs : The features for each domain pair (Figure 2) were selected based on their relevance to the specific nature of items in each domain, ensuring meaningful and contextually appropriate comparisons. By focusing on domain-specific aspects, we avoid artificial similarities and emphasize the most critical elements that enable valid cross-domain comparisons.

For the Books-Movies domain, features like Character Development, Setting, and Plot Summary are central, as both rely on narrative structure and character engagement, making them ideal for cross-domain analysis. In the Movies-Music domain, the focus shifts to Theme/Emotional Tone and Rhythm/Pacing, as both evoke emotional responses, though they differ in delivery. Music's rhythm plays a role similar to pacing in film, allowing for comparison of emotional and atmospheric qualities. For Books-Food, we selected Cultural Significance and Sensory Elements, as books convey experiences through storytelling, while food does so through sensory interaction, enabling meaningful cultural and sensory comparisons. In Electronics-Food, we focus on Functionality, Design, and Production Background. Although different in purpose—one utilitarian, the other sensory—both share commonalities in design and presentation. Tradition vs. Innovation applies to both electronics (technology advancements) and food (culinary evolution), offering a meaningful comparison.

By aligning feature selection with each domain's strengths, we ensure that cross-domain comparisons are relevant and

Books-Movies	Movies-Music	Books-Food	Electronics-Food
1) Genre	1) Theme/Emotional Tone	1) Theme/Cultural Significance	1) Functionality and Usage
2) Theme	2) Cultural Significance	2) Target Audience	2) Target Consumer Demographics
3) Key Features	3) Target Audience and Appeal	3) Experience/Sensory Elements	3) Experience and Sensory Interaction
4) Plot Summary	4) Atmosphere/Aesthetic Style	4) Presentation/Storytelling	4) Design and Presentation
5) Character Development	5) Narrative/Story Elements	5) Historical/Contextual Elements	5) Technological or Production Background
6) Setting and World-Building	6) Rhythm and Pacing		6) Tradition vs. Innovation
7) Target Audience	7) Tradition vs. Innovation		
8) Pacing and Style			

Fig. 2. Comparable Aspects for Domain Pair Comparison: Comparable aspects extracted from user reviews and item metadata in the Amazon Reviews Dataset for Books-Movies, Movies-Music, Books-Food, and Electronics-Food domain pairs.

avoid forcing connections where they do not naturally exist, ensuring that cross-domain item similarity extraction remains accurate and valid.

Item Feature Extraction Prompt for Books Movies Domain Pair

You are an advanced AI model that extracts detailed, objective aspects of an item based on its title, description, features, and review. Your task is to identify and list these aspects, making careful inferences based on the provided data where appropriate. However, you must strictly avoid including any user sentiments, opinions, or personalized content. Focus solely on the requested aspects, ensuring that any inferences are based only on clear and explicit cues in the provided information. Do not stretch inferences or assumptions where details are vague or missing.
Title: title
Description: description
Features: features
Review: review
Focus on the following aspects, based on the explicit information provided and only make inferences when they are strongly supported by clear patterns in the data:

1) **Genre:** Identify the genre of the item based on explicit information and clear inferences. If not mentioned or cannot be confidently inferred, state 'Not mentioned'.
2) **Theme:** Describe the main themes or topics discussed in the item, making logical inferences only if they are strongly supported by explicit cues in the data. If not mentioned or cannot be confidently inferred, state 'Not mentioned'.
3) **Key Features:** List any unique features or aspects of the item mentioned in the review or description, and make logical inferences about their significance only when clearly indicated. If not mentioned or cannot be inferred, state 'Not mentioned'.
4) **Plot Summary:** Provide a brief summary of the plot or storyline based on the description or review. Only make inferences where the storyline is clear, and avoid guessing. If not mentioned or cannot be inferred, state 'Not mentioned'.
5) **Character Development:** Describe the development and depth of the main characters as mentioned in the review or description, including any logical inferences only if strongly supported by clear evidence. If not mentioned or cannot be inferred, state 'Not mentioned'.
6) **Setting and World-Building:** Detail the setting and the world in which the story takes place, based on explicit details and clear inferences. If not mentioned or cannot be inferred, state 'Not mentioned'.
7) **Target Audience:** Identify the intended target audience for the item, making logical inferences based on explicit cues in the provided information. If not mentioned or cannot be inferred, state 'Not mentioned'.
8) **Pacing and Style:** Describe the pacing of the story and the writing or cinematic style based on explicit information and any logical inferences, only if strongly supported by clear details. If not mentioned or cannot be inferred, state 'Not mentioned'.

Important: Ensure the extracted aspects capture sufficient detail to analyze user behavior consistency across domains, focusing on comparable features like emotional tone, thematic complexity, and narrative depth.
Important: Exclude user sentiments or opinions and provide only the relevant aspects. If certain information cannot be inferred confidently, state 'Not mentioned' instead of making assumptions.

Fig. 3. LLM Prompt for Item Feature Extraction

2) Prompt Design for Item Features Extraction: LLMs have demonstrated strong performance in event extraction and reasoning [22], [23], but their output is highly dependent on how they are prompted [24]. To effectively extract item features with shared commonalities across domains, we designed unique, tailored prompts for each domain pair used in this study, by leveraging the aspects in Figure 2. In this study, we experimented with the domain pairs Books-Movies, Movies-Music, Books-Food, and Electronics-Food. For intuitively similar domain pairs like Books-Movies, we focused on features such as Character Development and Plot Summary to capture key aspects of both mediums. For less similar pairs like Electronics-Food, broader features such as Experience, Sensory Interaction, and Design were emphasized, enabling higher-level comparisons while reducing the risk of poor results from overly general prompts.

To ensure prompt quality, we designed prompts to focus on broader themes, make logical comparisons, handle missing information by returning "Not mentioned," and avoid forced inferences. This approach ensures meaningful comparisons without speculative responses, maintaining consistency in cross-domain feature extraction. For example, the Books-Movies prompt (Figure 3) follows these guidelines to achieve high-quality feature extraction. Responses for a Book and Movie item (Figures 6 and 7) demonstrate that the LLM effectively extracted common features with structured, precise outputs across all eight aspects. In the Genre" category, both responses returned "Not Mentioned" due to insufficient data, as instructed. In the "Pacing and Style" category for the Books item, despite missing specific data, the LLM made confident, logical inferences based on observable patterns, avoiding vagueness. Other features were extracted accurately, providing precise details without over-generalization, highlighting the prompt's effectiveness in guiding high-quality responses.

B. Similar Item Pair Extractor

The second component of our framework is the Similar Item Pair Extractor. This component focuses on generating the most similar item pairs between domains to assess consistency in user behavior. The goal is to examine how users interact with items that share significant similarities across domains. For instance, we can compare a user's interaction with the Harry Potter book in the Books domain and the Harry Potter movie in the Movies domain to evaluate whether their behavior remains consistent or differs between mediums. By analyzing changes in rating patterns, we can observe whether the user prefers the book over the movie, despite the content being the same but presented in different forms.

This approach emphasizes finding examples like Harry Potter across domains because they provide meaningful comparisons for effective cross-domain analysis. Even for domains that seem unrelated, such as Electronics and Food, we can draw comparisons by focusing on shared aspects like functionality, design, and sensory interaction, ensuring that selected items have comparable features.

We introduce a LLM-based technique (See Figure 4) that converts item features into rich numerical representations, known as embeddings [25]. Embeddings capture the semantic properties of textual data by transforming it into continuous numerical vectors, which are used to calculate item similarity. For this task, we primarily use the BERT-base-uncased model [26], pretrained on a large corpus of text. This model excels at generating contextual embeddings, allowing the same word to have different representations based on its sentence context. This results in more accurate representations, where similar meanings produce similar embeddings [27].

For this component, we generate embeddings for the item features of all items that a user has interacted with in both

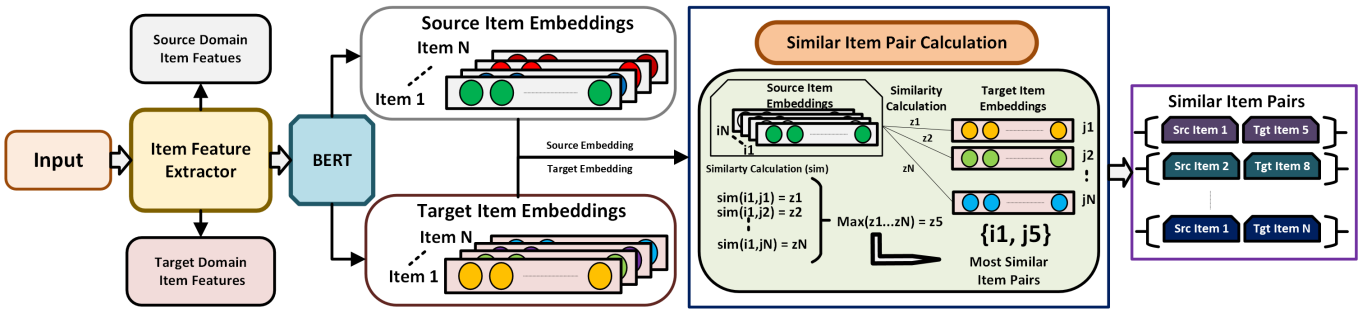

Fig. 4. Similar Item Pair Extractor: This figure illustrates the process of the Similar Item Pair Extractor. Overlapping users from the source and target domains, along with their item interactions, are used as input. Titles, Item Descriptions, Features, Reviews, and Ratings are processed in the Item Feature Extractor to generate item features. These features are embedded using BERT, and similarity is calculated to identify the most similar item pairs.

domains. These embeddings are then used to find the most similar item pairs across domains. For each item, we compare the embeddings in the source and target domains to find the best match. We use cosine similarity [28] for this comparison. Let src_{i_1} be the embedding of item i_1 from the source domain, and tgt_{j_1} be the embedding of item j_1 from the target domain. The cosine similarity is computed as:

$$i_1^{\text{sim}} = \frac{src_{i1} \cdot tgt_{j1}}{\|src_{11}\|\|tgt_{j1}\|} \tag{1}$$

where ${i_1}^{sim}$ reflects the similarity score between item i_1 from the source domain and item j_1 from the target domain, indicating how similar the two items are. A score closer to 1 indicates greater similarity. We compute the similarity score for item i_1 with all target domain items and select the item with the highest similarity score as the best match for i_1.

Our approach uses common aspects between domain pairs as a prompt template, but this can introduce structural bias, inflating item similarity even when it is inaccurate. To address this, we implemented the template denoising method by Jiang et al. [29], which removes structural bias by subtracting the template embedding from the full sentence embedding. Specifically, we treat the common aspects of a domain pair as the template, while the entire LLM-generated response, including the prompt template and its corresponding content, is used to generate the final embedding.

C. Sentiment Extractor

The third component in our framework, the sentiment extractor, incorporates user sentiment to provide context on how users feel about items. Sentiment reflects the user's attitude toward an item, capturing their feelings, which often indicates the depth of their engagement and summarizes the quality of their experience [30]. Consistent sentiment across domains adds significant value in assessing user behavior consistency. While ratings provide a numerical evaluation, they often overlook the emotional nuances of user interaction [31].

By incorporating user reviews and metadata, we gain an accurate understanding of sentiment, as this combination reveals the reasoning behind the rating and highlights specific attributes driving the user's feelings. For example, a user may give a movie a 4-star rating but express mixed feelings about the plot or acting. Ratings alone fail to capture this complexity, which user reviews and item metadata reveal. This richer information provides deeper insights into user preferences and behavior across domains, improving consistency assessment.

To capture this richness, we introduced a LLM-based technique (see Figure 1) that analyzes user reviews, ratings, and item metadata. This technique categorizes sentiment into three outputs: Positive, Neutral, and Negative. The prompts are tailored to each domain pair, but the same prompt is used consistently across both domains in the pair. To ensure quality, the prompts are designed to guide the model in delivering clear and consistent sentiment classifications by combining review text and ratings. While ratings serve as strong sentiment indicators, contextual information from reviews helps resolve conflicts logically. The prompts also account for domain-specific factors, such as storytelling in books or acting in movies, ensuring sentiments reflect each domain's unique characteristics. Additionally, the model generates relevant explanations derived from the review content. Due to space constraints, the Sentiment Extractor prompt is not included in this paper but is available in our GitHub repository [18].

D. Behavior Consistency Extractor

The fourth component in our framework is the behavior consistency extractor. This component combines information from the previous extractors to output a consistency value for a given user. It evaluates item features and user sentiment for similar item pairs that the user has interacted with, categorizing user behavior consistency into three levels: Highly Consistent, Neutral, and Not Consistent. The prompt is designed to guide the LLM in accurately assessing consistency when users interact with similar items across domains. The prompt, designed for the Books-Movies domain and focusing on both obvious and subtle patterns of consistency, is provided in Figure 5.

The prompt is customized for each domain pair to capture specific nuances and avoid generalizations. To ensure quality in assessing cross-domain user behavior consistency, the prompts emphasize specific, observable patterns rather than abstract conclusions. They highlight factors unique to each domain pair, such as narrative complexity in Books-Movies or

functionality in Food-Electronics. The design also accounts for incomplete data by avoiding speculative responses, grounding correlations in the user's specific preferences.

Behavior Consistency Extraction Prompt for Books Movies domain

You are an advanced AI model that rigorously determines the consistency between user behaviors based on their preferences and interactions across different domains. Focus on clear, meaningful alignments in behavior, considering observable patterns such as thematic focus, narrative complexity, and user engagement across domains like books and movies. Avoid abstract or unrelated inferences and ignore emotional tone or sentiment. Base conclusions solely on specific, measurable patterns in user behavior.
Below are the item details for one item in each domain the user has interacted with:
For the Books Domain:
- **Book Item:** book_item_features

For the Movies Domain:
- **Movie Item:** movie_item_features

Based on the item details from the two domains (Books and Movies), assess the consistency of the user's behavior across these domains. Identify clear connections between preferences, thematic interests, and engagement, focusing on patterns in user behavior. Consider consistent preferences for themes, complexity, or genres, even when expressed through different mediums.
When assessing consistency, use the following guidelines:
1) **Highly-consistent:** The user's behavior shows clear and meaningful alignment across both domains, with significant similarities in preferences or behavior patterns that are relevant, even if these are subtle or abstract. This alignment goes beyond simply liking or disliking both items.
2) **Neutral:** The user's behavior shows some similarities, but there is no strong or clear pattern of consistency or divergence across the domains.
3) **Not-consistent:** The user's behavior is noticeably inconsistent across both domains, with clear differences in preferences or behavior patterns, and no significant alignment is observed in how they engage with the content across the two domains.

Please classify the user's behavior as one of the following: 'Highly-consistent,' 'Neutral,' or 'Not-consistent.' Provide a brief explanation of your decision.

Fig. 5. LLM Prompt for Getting User Behavior Consistency

E. Quantifying Behavior Consistency

After going through the four different extractors in our proposed approach (see Figure 1), we need a way to use that information to quantify user behavior consistency. As mentioned in the "Behavior Consistency Extractor", We categorize behavior consistency for each similar item pair of a user into three categories: Highly Consistent, Neutral, and Not Consistent. These are mapped to numerical values 1, 0, and -1, respectively. This mapping provides a clear distinction between the categories, where 1 represents strong alignment, 0 indicates ambiguity, and -1 reflects clear inconsistency. The mapping simplifies interpretation and decision-making.

After mapping, we average the consistency values to compute a user's final behavior consistency score, offering a holistic view of their consistency across item pairs and capturing the most prevalent features. Let a_1 represent the mapped user behavior consistency score for the first similar item pair, and a_n represent the score for the nth similar item pair for an overlapping user u_1 in the cross-domain context. The average user behavior consistency score for user u_1 is computed as:

$$a_{u_1} = \frac{\sum_{i=1}^{n} a_i}{n} \tag{2}$$

where a_{u_1} represents a user behavior consistency score for user u_1 based on their interactions with similar item pairs. We repeat this process for each user to calculate their behavior consistency score. Once we have scores for all users, we average them to obtain the overall behavior consistency score for all overlapping users across both domains. **This quantifies user behavior consistency across domains**.

III. Experimental Setup

We conduct thorough experiments to address the following research questions: **RQ1**: How consistently do users rate items between similar domains compared to dissimilar domains? **RQ2**: To what extent are state-of-the-art CDR approaches able to leverage user behavior consistency across domains?

A. Dataset

Following established methods [1], [11], we use the **Amazon Reviews 2023 dataset** [21], containing 571,540,000 reviews from 54,510,000 users across 48,190,000 items. For evaluation, we selected six categories: Books, Movies, Music, Electronics, Food, and Video Games. Books, Movies, and Music represent similar domains with shared commonalities, while Electronics, Food, and Video Games are diverse domains with fewer shared characteristics. For user behavior consistency evaluation, datasets were grouped into Similar Domains (Books, Movies, Music) and Non-Similar Domains (Electronics, Food, Video Games). We ensured 100% user overlap and included users with at least 10 interactions. Tables I and II detail the dataset. To minimize bias and enhance computational efficiency, we randomly sampled 100 overlapping users from the six datasets for evaluation.

TABLE I
DATASET STATISTICS FOR SIMILAR DOMAINS

Domain	Total	Unique Users	Unique Items
Books	948971	29772	562092
Movies	879405	29772	232777
Music	672949	29772	256074

TABLE II
DATASET STATISTICS FOR NON-SIMILAR DOMAINS

Domain	Total	Unique Users	Unique Items
Electronics	363428	10654	159870
Food	204046	10654	89555
Video Games	118060	10654	36485

B. Implementation Details

We used the Llama-3-8B-Instruct model [32] as our LLM to generate Item Features, Sentiment, and User Behavior Consistency. We set the model parameters to, `do_sample=True`, `temperature=0.6`, `top_p=0.9`, and `max_new_tokens=1024`. For embedding generation in the Similar Item Pair Extractor, we employed the BERT-base-uncased model [33]. After extracting Item Features, we preprocessed the data to retrieve common aspects of the items from user reviews and item metadata. If the data is insufficient to provide information on a particular aspect, the LLM returns "Not Mentioned,". If 80% or more of the aspects for an item return "Not Mentioned," we remove that record.

After Similar Item Pair Extraction, we filtered out pairs with low similarity scores, often caused by LLM-generated errors or irrelevant outputs (e.g., special characters). If the LLM failed to generate a valid response during Item Feature Extraction, it returns "Failed to generate a valid response," and those records were also filtered. This process ensured that only items with sufficient and reliable features remained for comparison.

C. Baselines

We selected four key baselines, informed by recent advancements in cross-domain recommendation (CDR) research:

- **TGT** [34]: A single-domain Matrix Factorization (MF) model, trained only on the data from the target domain.

- **CMF** [35]: CDR model that extends MF by factorizing rating matrices from multiple domains at once.
- **EMCDR** [12]: A mapping-based CDR approach, which employs a linear function as a bridge between domains.
- **PTUPCDR** [11]: A bridge-based CDR model, which utilizes a meta-network and a linear mapping function to establish user-specific bridges across domains.

We used the open-source code from the authors of PTUPCDR [11] and filtered the data to include users with at least 10 interactions. Six datasets were selected and categorized into similar domains (Books, Movies, Music) and non-similar domains (Electronics, Food, Video Games). Overlapping users were retained within each category, and the test set for each target domain comprised 10,000 interactions. (See Tables I, II)

D. Experiment Design

In this study, each domain was selected as the target domain, with the remaining two domains serving as source domains, resulting in 12 domain combinations. Results were calculated for each baseline algorithm to evaluate how state-of-the-art CDR approaches perform with different source domains for the same target domain. The objective was to determine whether these approaches effectively leverage user behavior consistency to improve recommendation performance. To assess statistical significance, we conducted paired t-tests [36], comparing performance across domain pairs with a constant target domain and varying source domains. A p-value below 0.05 indicated statistical significance.

IV. Results

A. RQ1: User Behavior Consistency

We designed prompts to guide the LLM in Item Feature Extraction, Sentiment Extraction, and User Behavior Consistency Extraction. This paper focuses on Item Feature Extraction results and their role in computing similar item pairs effectively.

TABLE III
User Behavior Consistency Across Domain Pairs

Domain Pairs	User Behavior Consistency Score
Books-Movies	0.8732
Movies-Music	0.7731
Books-Food	0.4021
Electronic-Food	0.2324

1) Item Feature Response Evaluation: Figures 6 and 7 show the LLM's responses for two items from the Books and Movies domains, selected as similar item pairs in the Similar Item Pairs Extraction module. The LLM effectively extracted common features, providing structured and precise responses for all eight aspects as directed by the prompt. For instance, in the "Genre" category, both responses returned "Not Mentioned" due to insufficient data, following the prompt's instructions.

In the "Pacing and Style" category for the Book item, although specific data was missing, the LLM made confident and logical inferences based on repeated observable patterns without vague details. The remaining features were also extracted accurately, offering clear and precise information. This demonstrates the prompt's effectiveness in guiding the LLM to produce high-quality responses. The next section examines their role in computing similar item pairs.

2) Similar Item Pair Results: Figures 6 and 7 are the two

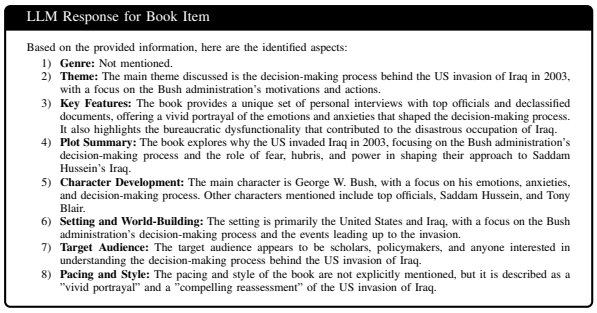

Fig. 6. LLM Response for Item in the Books Domain

items that were selected as the most similar to each other by the Similar Item Pair Extractor (see Figure 4). The reason for this is that both items revolve around the central theme of the U.S. invasion of Iraq in 2003, specifically focusing on the motivations and actions of the Bush administration. Both the book and the movie take an investigative approach to their subject matter: the book uses interviews and declassified documents to explore the decision-making process, while the film focuses on investigative journalism to uncover the truth behind the Bush administration's claims.

The precise responses from the Item Feature Extractor produced embeddings that effectively captured semantics, enabling accurate computation of similar item pairs. This validates our approach for calculating similar item pairs, which involves generating high-quality embeddings from accurate Item Feature responses, removing structural bias, and comparing items across domains.

3) User Behavior Consistency Results: Table III presents the User Behavior Consistency results for four domain pairs: Books-Movies, Books-Food, Movies-Music, and Electronics-Food. The results show that the consistency score is higher for the Books-Movies and Movies-Music domain pairs compared to the Books-Food and Electronics-Food pairs. This aligns with intuition, as Books-Movies and Movies-Music share many common aspects. For instance, many movies are adaptations of books, and music plays a role in movies, leading users to behave more consistently across these domains.

In contrast, the behavior consistency score is lower for the Books-Food and Electronics-Food domain pairs, which also aligns with expectations. These domains are fundamentally different, and while our prompts attempt to extract shared aspects, users tend to interact less consistently when engaging with such diverse domains. These results highlight stronger alignment in similar domains compared to diverse ones, emphasizing the need for domain-specific tailoring in cross-domain recommendation systems to better capture user preferences and behavioral nuances.

LLM Response for Movie Item

Based on the provided information, here are the identified aspects:

1) **Genre:** Not mentioned.
2) **Theme:** The main theme discussed is the propaganda campaign launched by the Bush administration to sell the Iraq war to the American public, as well as the role of the media in reporting on the war. Additionally, the theme of the importance of fact-checking and verifying information is also implied.
3) **Key Features:** The film uses made-up sources to convey actual events and phrases used in the run-up to the invasion. It also highlights the work of Knight Ridder reporters Jonathan Landay and Warren Strobel, who investigated the White House's claims about Iraq and found many problems.
4) **Plot Summary:** The film follows the story of two real-life reporters, Jonathan Landay and Warren Strobel, as they investigate the White House's claims about Iraq and find many problems. The story delves into the propaganda campaign launched by the Bush administration and the role of the media in reporting on the war.
5) **Character Development:** The main characters mentioned are Jonathan Landay and Warren Strobel, two real-life reporters who are portrayed as diligent and thorough in their investigation. Their motivations are to uncover the truth about the White House's claims about Iraq, and their conflicts arise from their efforts to fact-check and verify information in the face of government and media opposition.
6) **Setting and World-Building:** The story takes place in the United States and Iraq, specifically during the time leading up to the 2003 invasion of Iraq. The world-building is focused on the political and media landscape of the time, highlighting the propaganda campaign and the role of the media in reporting on the war.
7) **Target Audience:** The intended target audience appears to be adults interested in politics, history, and media studies, as the film deals with complex topics and themes.
8) **Pacing and Style:** The pacing of the story is likely to be slow and methodical, focusing on the investigative work of the reporters and the propaganda campaign. The writing style is likely to be informative and detailed, with a focus on presenting facts and evidence rather than sensationalism.

Fig. 7. LLM Response for Item in the Movies Domain

B. RQ2: Recommendation Performance

The results for the four algorithms—TGT, CMF, EMCDR, and PTUPCDR are shown in Tables IV and V.

TABLE IV
SIMILAR DOMAIN COMBINATIONS CDR RESULTS

	TGT		CMF		EMCDR		PTUPCDR	
Target Domain	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Target Music								
Books → Music	4.6749	5.6684	3.5494	4.3436	4.1905	4.3478	3.0505	3.7519
Movies → Music	4.736	5.999	2.7497	3.561	3.4565	3.7597	2.8467	3.5502
Two-Tailed t-test	$p < 0.74$	$p < 0.84$	$\mathbf{p < 2.4e{-}55}$	$\mathbf{p < 1.5e{-}50}$	$\mathbf{p < .7e{-}12}$	$\mathbf{p < 0.2e{-}10}$	$\mathbf{p < 1.4e{-}6}$	$\mathbf{p < 2.4e{-}6}$
Target Movies								
Books → Movies	4.8531	5.825	3.537	4.2609	4.1668	4.3336	3.8453	4.313
Music → Movies	4.8416	5.7908	2.483	3.2607	2.7733	3.2658	2.0942	2.7724
Two-Tailed t-test	$p < 0.77$	$p < 0.76$	$\mathbf{p < 3.3e{-}23}$	$\mathbf{p < 1.7e{-}21}$	$\mathbf{p < 1.4e{-}41}$	$\mathbf{p < 3.5e{-}36}$	$p < 0.59$	$p < 0.57$
Target Books								
Movies → Books	4.7202	5.7168	3.537	4.2609	4.1668	4.3336	3.8453	4.313
Music → Books	4.969	5.9715	3.8793	4.6374	4.1964	4.3587	3.9012	4.4018
Two-Tailed t-test	$p < 0.30$	$p < 0.40$	$p < 0.1$	$p < 0.09$	$p < 0.90$	$p < 0.60$	$p < 0.08$	$p < 0.06$

1) For Similar Domain Pairs: For the Music target domain, PTUPCDR outperforms other algorithms when Books and Movies are used as source domains. The results for PTUPCDR, EMCDR, and CMF are statistically significant, demonstrating their ability to leverage user behavior consistency across domains. In contrast, TGT, as a single-domain recommender, cannot transfer information from source domains, limiting its ability to utilize user behavior consistency and resulting in statistically insignificant outcomes.

For the Movies target domain, PTUPCDR outperforms other algorithms when Books and Music are used as source domains. However, its results, along with TGT, are statistically insignificant, unlike CMF and EMCDR, which achieve significant outcomes. This indicates that while PTUPCDR benefits from architectural advantages, it does not fully utilize user behavior consistency in this scenario. In contrast, CMF and EMCDR, though trailing PTUPCDR in overall performance, effectively leverage user behavior patterns to produce statistically significant results, highlighting their strength in capturing cross-domain relationships.

For the Books target domain, PTUPCDR outperforms other algorithms when Movies and Music are used as source domains. However, none of the algorithms achieve statistically significant results. This result may be influenced by nuanced differences in user interaction patterns across the domains. While Books, Movies, and Music are thematically similar, slight variations in how users engage with content could affect the algorithms' ability to fully leverage user behavior consistency, thereby impacting statistical significance.

TABLE V
NON-SIMILAR DOMAIN COMBINATIONS CDR RESULTS

	TGT		CMF		EMCDR		PTUPCDR	
Target Domain	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Target Electronics								
Food → Electronics	4.9039	5.8994	4.1208	4.9388	4.1862	4.3916	3.7776	4.4262
Games → Electronics	4.8452	5.8412	4.1168	4.9837	4.1809	4.3895	3.7722	4.3784
Two-Tailed t-test	$p < 0.06$	$p < 0.13$	$p < 0.96$	$p < 0.31$	$p < 0.96$	$p < 0.67$	$p < 0.20$	$p < 0.15$
Target Food								
Electronics → Food	4.9009	5.8955	3.9398	4.7468	4.2064	4.3959	4.0654	4.5276
Games → Food	4.8819	5.8977	4.5919	5.5611	4.2431	4.4315	4.2273	4.6176
Two-Tailed t-test	$p < 0.22$	$p < 0.26$	$\mathbf{p < 7.5e{-}38}$	$\mathbf{p < 7.14e{-}45}$	$\mathbf{p < 1.17e{-}09}$	$\mathbf{p < 6.91e{-}08}$	$p < 0.08$	$p < 0.06$
Target Games								
Food → Games	4.8236	5.7711	4.4582	5.3611	4.2018	4.3900	4.1873	4.4639
Electronics → Games	4.7908	5.7186	3.9009	4.7521	4.1588	4.3483	4.0153	4.3447
Two-Tailed t-test	$p < 0.89$	$p < 0.84$	$\mathbf{p < 4.7e{-}32}$	$\mathbf{p < 1.93e{-}34}$	$\mathbf{p < 6.3e{-}07}$	$\mathbf{p < 8.5e{-}08}$	$p < 0.08$	$p < 0.06$

2) Non-similar Domain Pairs: For the Electronics target domain, PTUPCDR outperforms other algorithms when Food and Games are used as source domains. However, none of the algorithms achieve statistically significant results, likely due to weak behavioral correlations between Electronics and the source domains. The diverse nature of user preferences and engagement patterns across these heterogeneous domains challenges the algorithms' ability to leverage behavior consistency. This underscores a limitation in current cross-domain recommendation methods and highlights the need for further refinement to capture nuanced relationships in such scenarios.

For the Food target domain, PTUPCDR outperforms other algorithms when Electronics and Games are used as source domains. However, PTUPCDR and TGT yield statistically insignificant results, unlike EMCDR and CMF, which achieve statistical significance. This suggests that while PTUPCDR excels in overall performance, it falls short in fully utilizing user behavior consistency in this scenario. In contrast, EMCDR and CMF effectively leverage behavioral patterns across domains, delivering significant outcomes. Similar trends are observed for the Games target domain with Food and Electronics as source domains, highlighting the need to refine PTUPCDR for better utilization of cross-domain behavioral correlations.

Due to computational constraints, we could not calculate user behavior consistency for domains involving games or evaluate CDR on them. However, the empirical evaluation across 12 domain combinations and four state-of-the-art CDR algorithms provides a robust assessment of performance when user behavior is consistent versus inconsistent.

V. RELATED WORK

Cross-domain recommendation (CDR) has emerged as a key area of interest due to its ability to address challenges such as data sparsity and cold-start issues in traditional recommender systems [2], [11], [12], [16]. Most CDR methods focus on transferring knowledge across domains using embedding mapping, transfer learning, and collaborative filtering [37]. State-of-the-art approaches, such as CoNet by Hu et al. [2], utilize neural networks to transfer knowledge between domains like Books and Movies. Vajjala et al. [1] further advanced CDR with VRCDR, employing geometric methods.

Large language models (LLMs) have shown promising results in recommender systems, particularly in leveraging

textual data like user reviews and item metadata [14], [38], [39]. Zhao et al. [3] highlight that LLMs enhance top-K recommendations, rating predictions, and explainable recommendations through advanced NLP capabilities [38]. However, most existing methods overlook a critical aspect of CDR: user behavior consistency across domains. Xi et al. [40] proposed Behavior Alignment as a metric to evaluate how well LLM-based conversational recommendation systems align with human recommenders, focusing on conversational strategies. While their work evaluates alignment in conversational flows [41], our study introduces a distinct approach by explicitly quantifying user behavior consistency across domains. Unlike Behavior Alignment, which assesses conversational dynamics, we focus on whether users exhibit consistent preferences and behaviors when interacting with items across different domains, using item metadata.

VI. Conclusion

In this study, we introduced a novel approach leveraging LLMs to quantify user behavior consistency across domains and explored its impact on cross-domain recommendation (CDR) systems. To our knowledge, this is the first method explicitly quantifying behavior consistency in CDR. Our findings revealed that CDR performance does not consistently correlate with behavior consistency, with significant variability even in similar domain pairs. This suggests that while behavior consistency provides valuable insights, its potential remains underutilized due to limitations in current CDR models and their inability to capture nuanced cross-domain relationships. These findings highlight the need for advanced recommendation algorithms that better incorporate user behavior consistency to improve accuracy across domains. Future work will focus on developing novel CDR algorithms that fully leverage consistent user rating behavior across domains.

References

[1] A. Krishna Vajjala, D. Meher, S. Pothagoni, Z. Zhu, and D. Rosenblum, "Vietoris-rips complex: A new direction for cross-domain cold-start recommendation," in *SDM*, 2024.

[2] G. Hu, Y. Zhang, and Q. Yang, "Conet: Collaborative cross networks for cross-domain recommendation," in *CIKM*, 2018.

[3] Z. Zhao, W. Fan, J. Li, Y. Liu, X. Mei, Y. Wang, Z. Wen, F. Wang, X. Zhao, J. Tang, and et al., "Recommender systems in the era of large language models (llms)," *TKDE*, 2024.

[4] S. Rajput, N. Mehta, A. Singh, R. Hulikal Keshavan, T. Vu, L. Heldt, L. Hong, Y. Tay, V. Tran, J. Samost *et al.*, "Recommender systems with generative retrieval," *NeurIPS*, 2024.

[5] A. Felfernig, L. Boratto, M. Stettinger, and M. Tkalčič, *Group recommender systems: an introduction*, 2024.

[6] A. Shankar, P. Perumal, M. Subramanian, N. Ramu, D. Natesan, V. Kulkarni, and T. Stephan, "An intelligent recommendation system in e-commerce using ensemble learning," *MTA*, 2024.

[7] J. Schafer, J. Konstan, and J. Riedl, "Recommender systems in e-commerce," in *EC*, 1999.

[8] F. Karimova, "A survey of e-commerce recommender systems," *ESJ*, 2016.

[9] X. Lam, T. Vu, T. Le, and A. Duong, "Addressing cold-start problem in recommendation systems," in *IMCOM*, 2008.

[10] Y. Chen, C. Wu, M. Xie, and X. Guo, "Solving the sparsity problem in recommender systems using association retrieval." *J. Comput.*, 2011.

[11] Y. Zhu, Z. Tang, Y. Liu, F. Zhuang, R. Xie, X. Zhang, L. Lin, and Q. He, "Personalized transfer of user preferences for cross-domain recommendation," in *WSDM*, 2022.

[12] T. Man, H. Shen, X. Jin, and X. Cheng, "Cross-domain recommendation: An embedding and mapping approach." in *IJCAI*, 2017.

[13] H. Ma, R. Xie, L. Meng, X. Chen, X. Zhang, L. Lin, and J. Zhou, "Triple sequence learning for cross-domain recommendation," *TOIS*, 2024.

[14] A. Petruzzelli, C. Musto, L. Laraspata, I. Rinaldi *et al.*, "Instructing and prompting large language models for explainable cross-domain recommendations," in *RecSys*, 2024.

[15] S. Zhang, Q. Miao, P. Nie, M. Li, Z. Chen, F. Feng, K. Kuang, and F. Wu, "Transferring causal mechanism over meta-representations for target-unknown cross-domain recommendation," *TOIS*, 2024.

[16] A. Vajjala, A. Vajjala, Z. Zhu, and D. Rosenblum, "Analyzing the impact of domain similarity: A new perspective in cross-domain recommendation," in *IJCNN*, 2024.

[17] W. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang *et al.*, "A survey of large language models," *arXiv*, 2023.

[18] "Github repo," 2024. [Online]. Available: https://github.com/dipakmeher/User_Behavior_Consistency_in_CDR

[19] P. Panzarasa, T. Opsahl, and K. Carley, "Patterns and dynamics of users' behavior and interaction: Network analysis of an online community," *JASIST*, 2009.

[20] A. Vaswani, "Attention is all you need," *NeurIPS*, 2017.

[21] D. Basaran, E. Ntoutsi, and A. Zimek, "Redundancies in data and their effect on the evaluation of recommendation systems: A case study on the amazon reviews datasets," in *SDM*, 2017.

[22] F. Huang, Q. Huang, Y. Zhao, Z. Qi, B. Wang, Y. Huang, and S. Li, "A three-stage framework for event-event relation extraction with large language model," in *NeurIPS*, 2023.

[23] A. Kalyanpur, K. Saravanakumar, V. Barres, J. Chu-Carroll, D. Melville, and D. Ferrucci, "Llm-arc: Enhancing llms with an automated reasoning critic," *arXiv*, 2024.

[24] G. Marvin, N. Hellen, D. Jjingo, and J. Nakatumba-Nabende, "Prompt engineering in large language models," in *ICDICI*, 2023.

[25] X. Ma, Z. Wang, P. Ng, R. Nallapati, and B. Xiang, "Universal text representation from bert: An empirical study," *arXiv*, 2019.

[26] S. Behera and R. Dash, "Fine-tuning of a bert-based uncased model for unbalanced text classification," in *ICAC*, 2022.

[27] G. Wiedemann, S. Remus, A. Chawla, and C. Biemann, "Does bert make any sense? interpretable word sense disambiguation with contextualized embeddings," *arXiv*, 2019.

[28] T. Thongtan and T. Phienthrakul, "Sentiment classification using document embeddings trained with cosine similarity," in *ACL-SRW*, 2019.

[29] T. Jiang, J. Jian, S. Huang, Z. Zhang, D. Wang, F. Zhuang, F. Wei, H. Huang, D. Deng, and Q. Zhang, "Promptbert: Improving bert sentence embeddings with prompts," *arXiv*, 2022.

[30] H. Tran and M. Shcherbakov, "Detection and prediction of users attitude based on real-time and batch sentiment analysis of facebook comments," in *CSoNet*, 2016.

[31] D. Yu, Y. Mu, and Y. Jin, "Rating prediction using review texts with underlying sentiments," *IPL*, 2017.

[32] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan *et al.*, "The llama 3 herd of models," *arXiv*, 2024.

[33] M. Geetha and D. Renuka, "Improving the performance of aspect based sentiment analysis using fine-tuned bert base uncased model," *IJIN*, 2021.

[34] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, 2009.

[35] A. Singh and G. Gordon, "Relational learning via collective matrix factorization," in *KDD*, 2008.

[36] H. Hsu and P. Lachenbruch, "Paired t test," *Wiley StatsRef*, 2014.

[37] H. Su, Y. Zhang, X. Yang, H. Hua, S. Wang, and J. Li, "Cross-domain recommendation via adversarial adaptation," in *CIKM*, 2022.

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

[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 *KDD*, 2024.

[40] Y. Xi, W. Liu, J. Lin, B. Chen, R. Tang, W. Zhang, and Y. Yu, "Memocrs: Memory-enhanced sequential conversational recommender systems with large language models," *arXiv*, 2024.

[41] D. Yang, F. Chen, and H. Fang, "Behavior alignment: A new perspective of evaluating llm-based conversational recommendation systems," in *SIGIR*, 2024.