

HIVE: Human-In-the-Loop Framework for Explainable Recommender Systems with Veracity-Based Human Feedback

Yeonbin Son
Systems and Information Engineering
University of Virginia
Charlottesville, USA
ybson@virginia.edu

Bohyun Choi
Artificial Intelligence Convergence
Ewha Womans University
Seoul, South Korea
bohyunchoi@ewha.ac.kr

Jiyoun Park
Data Science
Ewha Womans University
Seoul, South Korea
gigipark@ewha.ac.kr

Yerim Choi
Data Science
Ewha Womans University
ai.m Inc.
Seoul, South Korea
yrchoi@ewha.ac.kr

Matthew L. Bolton*
Systems and Information Engineering
University of Virginia
Charlottesville, USA
matthewbolton@virginia.edu

Abstract

Previous research has demonstrated that incorporating human feedback in recommender systems can improve personalization and adapt to evolving user preferences. This idea has also been extended to explainable recommender systems (XRSs), though limited attention has been given to explanation information quality. Recently, a new metric named *Veracity* was proposed to evaluate the information quality of explanation content. *Veracity* decomposes information along two dimensions: *Fidelity* (whether explanations truthfully represent the recommended item) and *Attunement* (whether they align with the end user’s preferences). Based on this metric, we propose a novel human-in-the-loop framework for XRSs, called HIVE. HIVE collects human feedback on explanation veracity and leverages it to iteratively update user and item embeddings, thereby improving both recommendation performance and explanation quality through personalized factual correctness and preference alignment. We validated the framework through two simulation experiments: (1) an offline evaluation using dataset ground truth and (2) a pseudo-user study simulating human responses with large language models to represent diverse user personas. Results demonstrated that incorporating veracity-based feedback in an iterative loop measurably improves performance. These findings suggest that our framework offers a practical solution for developing more reliable and user-aligned XRSs. Future work will validate the approach with real users, extend it to other domains and modalities, and explore its potential to mitigate challenges such as cold-start.

Keywords

explainable recommender systems, human feedback, human-in-the-loop learning, veracity, information quality evaluation, human-AI interaction

1 Introduction

Recommender systems (RSs) aim to suggest items or contents that users are likely to prefer, based on user characteristics, item characteristics, and interactions between users and items [1]. Recently,

*Corresponding author.
© 2025 Copyright held by the owner/author(s).


Recommended Game	Explanation
 The Elder Scrolls V Skyrim	The Elder Scrolls V Skyrim is a single-player game , similar to previously played titles such as <i>Bastion</i> and <i>Dead Space</i> , which also offer a single-player experience.

Figure 1: Example of textual explanations using natural language to justify the recommendation.

there has been growing interest in explainable recommender systems (XRSs), which not only generate recommendations but also explanations of why the item is recommended to the user [21]. These explanations aim to enhance recommendation persuasiveness, user satisfaction, and system transparency [5]. Explanations can take different forms depending on the system’s purpose, and Figure 1 presents an example of a textual explanation. Textual explanations describe the reasoning behind the algorithm in natural language, typically highlighting feature-level information [13].

Recent advances in AI have sparked increasing interest in human-centered approaches that emphasize systems designed to collaborate with, communicate with, and ultimately augment people [2]. Rather than viewing AI merely as a tool for automation, this perspective highlights the importance of designing systems that enhance human capabilities, support decision-making, and align with human values. In the field of recommender systems, this shift has motivated studies that leverage human feedback to generate better results. For example, Cai et al. [4] proposed an LLM-based interactive framework, in which an XRS presents results to the user, receives descriptive feedback in a conversational format, and utilizes it in subsequent recommendation turns. Similarly, Ouyang et al. [15] presented multiple outputs from the same prompt to human annotators, who ranked them. A reward model was trained on this ranking data and subsequently used to improve LLM performance. Ghazimatin et al. [7] introduced a model-agnostic framework that explains recommendations by presenting a similar item along with overlapping features (e.g., genre, actor) and updates input representations based on binary user feedback (like/dislike) on those features. While effective, the feedback used in these studies is limited to expressions of preference or relative preference order. Such

an approach lacks detailed diagnostics and thus does not address discrepancies in how an XRS may interpret factual information differently from end users. This limitation makes it difficult to verify whether the explanations (central to the role of an XRS) actually deliver high-quality information.

A recent study introduced veracity as an objective metric for evaluating the information quality of explanations [17]. Veracity is defined as a composite of two dimensions: fidelity and attunement that specifically relate to the type of information conveyed in XRS explanations. When an XRS provides feature-level explanation, for example “*This item has [feature] that you may like*”, it conveys two types of information: (1) factual information about the features of the item (fidelity), and (2) evaluative information about the user’s orientation toward those features (attunement). User feedback is collected on both dimensions for each explanation, and the two are combined to compute a final veracity score. A higher score indicates more accurate and informative explanations. This metric is particularly important when dealing with modern systems that may be operating on incomplete or inaccurate knowledge or, when using technologies like LLMs, are prone to generating nonfactual content, known as the hallucination problem [10].

In this paper, we propose HIVE (Human-In-the-loop framework for XRS with Veracity-based feedback on Explanations). HIVE integrates two key ideas: (1) evaluating the information quality of explanations using the veracity metric and (2) iteratively leveraging this human feedback to improve model performance. Our framework incorporates user feedback to dynamically refine the initial user and item embeddings, rather than relying solely on static representations derived from the dataset. Importantly, our framework is model-agnostic, and thus it can be applied to any XRS as long as the generated explanations include feature-level information and the underlying user, item, and feature representations can be expressed in an embedding space. To evaluate our approach, we ran two simulation experiments in a game recommendation setting using the Steam [19] dataset, collected from a popular video game platform. The first experiment was an offline evaluation based on dataset ground truth. The second was a pseudo-user study, where LLM agents generated simulated responses instantiated with user-specific historical data, enabling the creation of realistic and personalized personas. The remainder of this paper is organized as follows: we first review background on veracity, then describe our method, present the experiments and results, and finally discuss implications and directions for future research.

2 Background

Veracity is a metric for evaluating the information quality of explanations generated by XRS [17]. The information embedded in an explanation is decomposed into two dimensions: fidelity and attunement. Fidelity and attunement are assessed separately and then combined to yield the final veracity score. To operationalize this evaluation, Son and Bolton [17] employed signal detection theory (SDT) to calculate sensitivity. SDT is a theoretical framework that shows how the system distinguishes signals from noise under uncertainty [16], and sensitivity is the SDT-based metric that measures this discriminative ability. Accordingly, the veracity

		<i>Fidelity</i> (tells the truth)	In reality, the item	
			Has the feature	Doesn't have the feature
XRS says	Item has the feature		Hit	False Alarm
	Item doesn't have the feature		Miss	Correct Rejection

(a) Fidelity Decision Outcomes

		<i>Attunement</i> (understands human)	User (Human)	
			Likes the feature	Doesn't like the feature
XRS says	User may like the feature		Hit	False Alarm
	User may not like the feature		Miss	Correct Rejection

(b) Attunement Decision Outcomes

Figure 2: Confusion matrices for deciding (a) fidelity and (b) attunement decision outcomes. Each outcome is determined by comparing the XRS’s explanation with the ground-truth reality.

sensitivity score represents the degree of information quality in an explanation.

For calculating the fidelity, attunement, and veracity sensitivity metrics (hereafter referred to as scores), the XRS generates multiple (rec, exp) pairs, with each explanation assumed to contain feature-level information. For each explanation, the fidelity and attunement outcomes are determined among the four decision categories: hit (H), miss (M), false alarm (FA), and correct rejection (CR). 2 illustrates how each outcome is determined. Specifically, fidelity is evaluated by comparing whether the explanation correctly states the presence or absence of an item feature with the item’s actual feature profile, where the signal corresponds to a feature that is truly present in the item and the noise corresponds to a feature that is truly absent. Attunement, on the other hand, is evaluated by comparing whether the explanation predicts that the user will like or dislike a feature with the user’s actual preference for the item. Here, signal corresponds to a feature the user actually likes and the noise corresponds to a feature the user does not like. After determining fidelity and attunement outcomes for all explanations, the total number of hits and false alarms are counted, and the hit rate (HR) and false alarm rate (FAR) are calculated using Equations (1) and (2). HR represents the proportion of signal trials where the signal is detected:

$$HR = \frac{\text{Number of hits}}{\text{Number of signal trials}}. \quad (1)$$

FAR is the proportion of noise-only trials where the signal is erroneously detected:

$$FAR = \frac{\text{Number of false alarms}}{\text{Number of noise trials}}. \quad (2)$$

These two values are then used to compute the sensitivity score, as shown in Equation (3).

$$A' = \begin{cases} 0.5 + \frac{(HR \cdot FAR)(1 + HR \cdot FAR)}{4 HR (1 - FAR)}, & \text{if } HR \geq FAR, \\ 0.5 + \frac{(FAR \cdot HR)(1 + FAR \cdot HR)}{4 FAR (1 - HR)}, & \text{otherwise.} \end{cases} \quad (3)$$

Case	Fidelity Outcome				Attunement Outcome				Veracity Outcome			
	H	M	FA	CR	H	M	FA	CR	H	M	FA	CR
1	1	0	0	0	1	0	0	0	1	0	0	0
2	0	1	0	0	0	1	0	0	0	1	0	0
3	0	0	1	0	0	0	1	0	0	0	1	0
4	0	0	0	1	0	0	0	1	0	0	0	1
5	1	0	0	0	0	0	0	1	0.5	0	0	0.5
6	0	0	0	1	1	0	0	0	0.5	0	0	0.5
7	0	0	1	0	0	1	0	0	0	0.5	0.5	0
8	0	1	0	0	0	0	1	0	0	0.5	0.5	0
9	0	1	0	0	0	0	0	1	0	1	0	0
10	0	1	0	0	1	0	0	0	0	1	0	0
11	0	0	1	0	1	0	0	0	0	0	1	0
12	0	0	1	0	0	0	0	1	0	0	1	0
13	0	0	0	1	0	0	1	0	0	0	1	0
14	0	0	0	1	0	1	0	0	0	1	0	0
15	1	0	0	0	0	1	0	0	0	1	0	0
16	1	0	0	0	0	0	1	0	0	0	1	0

Figure 3: Illustration of cases for determining the veracity outcome. Each case is derived by combining fidelity and attunement outcomes according to the defined decision rules.

This yields the fidelity and attunement scores, which range between 0.5 and 1, with higher values indicating greater informativeness.

The veracity outcome of each explanation is derived by combining the fidelity and attunement outcomes. Note that among the decision outcomes, H and CR correspond to correctness, whereas M and FA correspond to incorrectness. If the fidelity and attunement outcomes are identical, the veracity outcome is assigned the same label. If one belongs to the correctness set and the other to the incorrectness set, the incorrectness outcome is prioritized, as incorrect information is considered more critical. If the two outcomes differ, but both belong to the same set, equal credit is assigned to each (i.e., half-and-half). This process is illustrated in 3. After determining the veracity outcomes for all explanations, the same procedure described earlier is applied to compute HR and FAR, which are then used to calculate the final veracity score. The resulting veracity score thus provides a unified and objective measure of explanation quality, capturing both fidelity and attunement.

3 Methods

In this section, we introduce our novel framework HIVE, which consists of three modules. HIVE begins with a recommendation module that initiates the process of generating output. The backbone explainable recommendation algorithm can take various forms of input, such as knowledge graphs, text, or images, but ultimately represents information through user, item, and feature embeddings. For a given target user, this module generates a set of recommendations. Each recommendation is accompanied by a feature-level explanation that specifies which item feature is the most responsible for the recommendation, as shown in Figure 1.

Next, we collect human feedback on each feature’s information quality along two veracity dimensions: fidelity (factual correctness about the item) and attunement (preference alignment for the user). For fidelity, we ask “Is this feature truly representative of the item?” For attunement, we ask “Do you like this feature?” We adopt continuous degrees $d_{v,f} \in [0, 1]$ and $a_{u,f} \in [0, 1]$ for fidelity and attunement, respectively, where v is an item, u is a user, and f indexes a feature. Higher values indicate stronger correctness (fidelity) or preference (attunement).

In the third module, we update item and user embeddings to reflect user opinion based on the collected feedback, allowing the system to naturally adapt to preference shifts or inconsistencies over time. Let \vec{u} , \vec{v} , and \vec{f} denote the embeddings for a user, an item, and a feature, respectively. Item and user embeddings are updated via two degree-weighted objectives capturing (1) item-feature factual alignment (fidelity) and (2) user-feature preference alignment (attunement). This feedback allows us to calculate a total objective loss function based on loss of both fidelity and attunement. It also allows us to consider the veracity metric (see Section 2) in analysis results. The following builds the total objective loss from the fidelity and attunement loss functions. Note that, in the formulation for these functions, $\text{sim}(\cdot, \cdot)$ indicates cosine similarity, and $\mathcal{F}(v)$ and $\mathcal{F}(u)$ are candidate feature sets attached to item v and user u .

3.1 Fidelity Loss

For each item v , we rank $\mathcal{F}(v)$ by $d_{v,f}$ to build a positive fidelity feature set \mathcal{P}_v and a negative fidelity feature set \mathcal{N}_v . The fidelity objective encourages the item embedding \vec{v} to move closer to features judged factually correct and farther from features judged incorrect. It is defined in Equation (4).

$$\mathcal{L}_{fid} = \sum_v \frac{1}{Z_v} \sum_{f_p \in \mathcal{P}_v} \sum_{f_n \in \mathcal{N}_v} d_{v,f_p} (1 - d_{v,f_n}) \max(0, m - \text{sim}(\vec{v}, \vec{f}_p) + \text{sim}(\vec{v}, \vec{f}_n)) \quad (4)$$

Here, d_{v,f_p} and d_{v,f_n} denote the degrees of factual correctness (fidelity), elicited from user feedback, for the related feature f_p and the unrelated feature f_n of item v , respectively. The pairwise weight $d_{v,f_p} (1 - d_{v,f_n})$ increases when the positive feature is judged more correct and the negative feature less correct, thereby focusing updates on high-confidence pairs while down-weighting ambiguous cases (both ≈ 0.5). We normalized by $Z_v = \sum_{f_p \in \mathcal{P}_v} \sum_{f_n \in \mathcal{N}_v} d_{v,f_p} (1 - d_{v,f_n})$ to obtain a per-item weighted mean, which stabilizes gradient magnitudes across items with different numbers and degree distributions of candidate features.

3.2 Attunement Loss

Symmetrically, for each user u , we rank $\mathcal{F}(u)$ by $a_{u,f}$ to construct the positive attunement feature set \mathcal{P}_u and a negative attunement feature set \mathcal{N}_u . The attunement objective encourages the user embedding \vec{u} to move closer to features the user likes and farther from features the user dislikes. It is defined as follows in Equation (5).

$$\mathcal{L}_{att} = \sum_u \frac{1}{Z_u} \sum_{f_l \in \mathcal{P}_u} \sum_{f_d \in \mathcal{N}_u} a_{u,f_l} (1 - a_{u,f_d}) \max(0, m - \text{sim}(\vec{u}, \vec{f}_l) + \text{sim}(\vec{u}, \vec{f}_d)) \quad (5)$$

where a_{u,f_l} and a_{u,f_d} indicate the degrees of u ’s preference alignment (attunement), for the liked feature f_l and the disliked feature f_d , respectively. The pairwise weighting $a_{u,f_l} (1 - a_{u,f_d})$ and the normalization $Z_u = \sum_{f_l \in \mathcal{P}_u} \sum_{f_d \in \mathcal{N}_u} a_{u,f_l} (1 - a_{u,f_d})$ follow the same rationale as in the fidelity loss (Equation (4)).

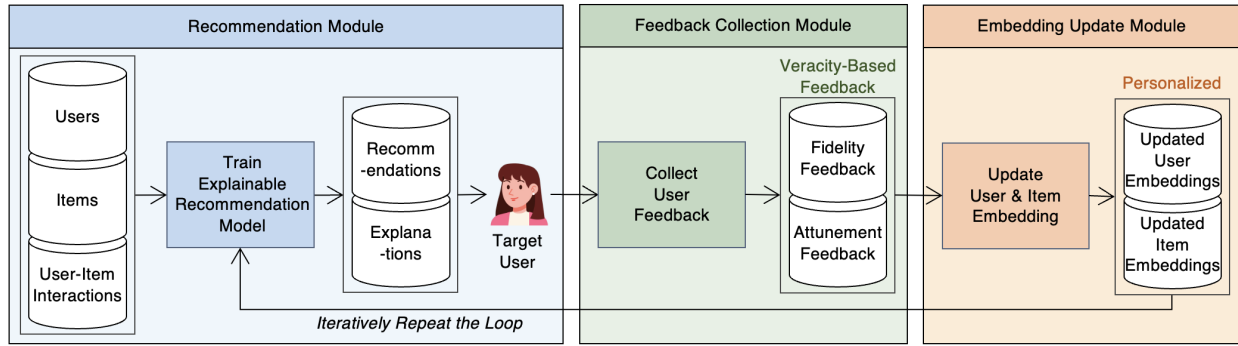


Figure 4: Illustration of the proposed framework. The recommendation module generates item and feature-level explanations. The feedback collection module gathers veracity-based human feedback, which is used to update item and user embeddings in a user-specific manner. These updates are inputs to the backbone recommendation algorithm to improve future results.

3.3 Veracity Loss

The final veracity loss \mathcal{L} is defined as the combination of the fidelity and attunement losses, as in Equation (6).

$$\mathcal{L}_{ver} = \mathcal{L}_{fid} + \mathcal{L}_{att} \quad (6)$$

During optimization, the user and item embeddings \vec{u} and \vec{v} are optimized to be \vec{u}^* and \vec{v}^* , respectively. These updated embeddings are subsequently re-integrated into the backbone explainable recommendation algorithm to produce refined recommendations and more personalized explanations.

4 Experiments

Our hypothesis was that using our novel veracity-based human feedback would improve both the performance of recommendations and the quality of explanations. We start with an initial set of recommended items and explanations generated by a backbone explainable recommendation algorithm in round 0 (R0). As part of this round, we collected feedback on the explanations and updated the results accordingly. This feedback loop was repeated over two rounds (R0 \rightarrow R1 and R1 \rightarrow R2) in two different evaluation simulations. The first was an offline evaluation, where feedback was derived from the dataset’s ground-truth user preferences. The second was a pseudo-user study, where LLM agents were conditioned on actual user histories to simulate realistic, persona-based feedback. We conducted both types of evaluation as they provide us with slightly different information. The offline study establishes a performance baseline under the assumption of perfect information. The pseudo-user study gives us a more realistic setting by introducing the kinds of discrepancies that can arise between an XRS’s understanding of reality and a human user’s subjective experience. Importantly, the pseudo-user study was a deliberate precursor to a real user study. By first validating our approach in a controlled, low-cost, scalable simulation, we hoped to refine our method, identify any pitfalls, and ensure the feasibility of the feedback loop before committing to a full-scale human subject experiment.

4.1 Dataset

For both simulations, we considered the video game recommendation scenario using the Steam dataset. Steam is a popular video

game platform [19], and this study utilized two publicly available datasets [6, 18], which were integrated into a unified knowledge graph based on the name of the video game. The first dataset contains game characteristics, while the second includes user-game interactions. This includes data about things such as whether a user purchased or played a game and their total playtime. Note that there are cases where a user purchased a game but never played it. The integrated knowledge graph comprises 10,500 user nodes, 3,005 game nodes, 5,138 feature nodes (including developer, genres, categories, and release year), and 104,096 relationships.

Introduction	You are a user of a game recommendation system. Your task is to evaluate ...
Evaluation Dimensions	Fidelity refers to ... Attunement refers to ...
Persona Setting	You must adopt the role of the user. Based on the provided gameplay history, ...
Data	User: [User game history] Explanation: [Explanation text]
Output	Return a CSV with ...

Figure 5: Outline of the prompt template.

4.2 Experimental Settings

The experiment involved collecting simulated human feedback using two different approaches (offline and with a pseudo-user study), integrating the feedback into the next round’s results, and checking whether performance is improved. Furthermore, we compared the performance of HIVE with ELIXIR [7], a state-of-the-art model-agnostic framework that also incorporates human feedback for XRS [7]. While ELIXIR also collects feedback on features related to recommended items, this feedback is not user-specific and reflects only overall like/dislike opinions. It does not collect feedback about the information quality in explanations. Thus, our comparative experiments allow us to examine whether veracity-based user feedback provides more informative signals for improving XRS performance.

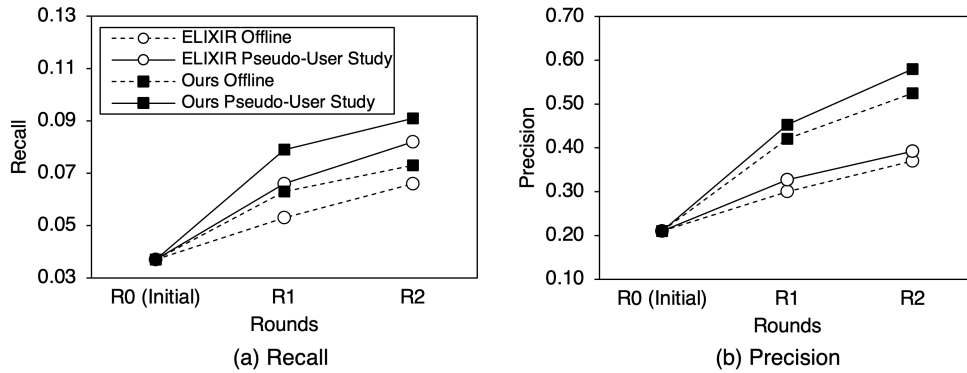


Figure 6: Recommendation performance in terms of precision (squares, left y-axis) and recall (circles, right y-axis) across iterations (R0 to R2). Solid lines show offline results; dotted lines show pseudo-user study results. Both metrics show consistent gains, indicating that iterative human feedback enhanced the alignment between user preferences and recommended items.

4.2.1 Backbone Algorithm. For generating recommendations and explanations, we adopted the Tower Bridge-Network (TB-Net) [20] as the backbone algorithm. The TB-Net performs bidirectional embedding propagation over a knowledge graph to capture key semantic factors for explainable recommendation. It takes a knowledge graph that includes users, items, and feature nodes as inputs. To address the potential incompleteness of the predefined knowledge graph, we updated it by extracting additional entities from each game’s textual reviews using LLMs. As an output, TB-Net generates recommended items and a path between the target user and the item by identifying an important intermediate node. These outputs are then passed to LLMs to generate refined natural language explanations.

4.2.2 Feedback. We collected simulated fidelity and attunement feedback ratings for each feature-level explanation in both experiments (as per Section 3). The feedback was recorded as a continuum ranging from 0.0 (disagree/dislike) to 1.0 (agree/like). Among all the users in our dataset, we randomly selected 20 users who had more than 50 liked games for the test set. For each user, the system recommended 10 games, each accompanied by 10 feature-level explanations. For the recommendation performance evaluation, we assessed whether 200 games recommended to the 20 users were appropriate. For the explanation quality evaluation, we examined the 2,000 feature-level explanations generated for these games.

4.2.3 Simulation Approaches. Now, we describe each feedback simulation process in detail. For the offline simulation, we used the playtime attribute in the dataset to determine the ground truth for user preferences. Since games vary in their typical play duration, we first normalized each user’s playtime by dividing it by the game’s global average playtime [11]. Next, we averaged the normalized playtimes for each user. A game was labeled as user-liked if the user’s normalized playtime for that game exceeded their personal average. Otherwise, it was labeled as a user-disliked game.

As part of the pseudo-user study, we employed the pre-trained LLM, GPT-4o mini [14], as an automatic feedback simulator. The pseudo-user study was conducted to enable robust preliminary testing before a real user study (see Section 5). This was accomplished by generating diverse personas based on each user’s behavioral

history using an LLM (as per [3]) and using these to give feedback on explanations as a “pseudo” user. GPT-4o mini was chosen for its strong reasoning ability and effectiveness in instruction-following tasks [12]. When running the evaluator, the parameter ‘temperature’ was set to 0.1 to constrain randomness.

We adopted a role-play prompting (single turn method) [9] for simulating pseudo-users. This is a simple yet effective approach where the LLM is explicitly instructed to assume the role of a user based on a historical profile, such as gameplay history. Each instance provided the model with (1) a simulated user persona constructed from historical gameplay data, (2) the recommended item’s feature set and explanation, and (3) a structured prompt describing the evaluation task. This setup enables automated evaluation that reflects a user-aligned perspective without requiring real-time human participation. The prompt template used is briefed in Figure 5.¹

4.2.4 Evaluation Metrics. As evaluation measures, we adopted precision and recall for calculating recommendation performances, two of the most widely used metrics for recommender systems [1]. Precision is defined as the number of recommended items that the user likes divided by the total number of recommended items. Recall is defined as the number of recommended items that the user likes divided by the total number of items the user previously liked. For evaluating explanation quality, we adopted the fidelity, attunement, and veracity scores proposed by Son and Bolton [17]. As detailed in Section 2, these scores are calculated based on signal detection theory, with fidelity and attunement sensitivities combined into an overall veracity score.

4.3 Results

Figure 6 reports the changes in recommendation performance in terms of recall and precision over iterations for ours (HIVE) compared with the baseline framework (ELIXIR). In terms of precision, both methods improved steadily across rounds. In the offline evaluation, our framework increased from 0.21 at R0 to 0.42 at R1 and 0.50 at R2, whereas ELIXIR rose from 0.21 to 0.30 and 0.38. In the pseudo-user study, our framework improved from 0.21 at R0 to

¹The full template can be found at <https://github.com/13207418/hitl-xrs>.

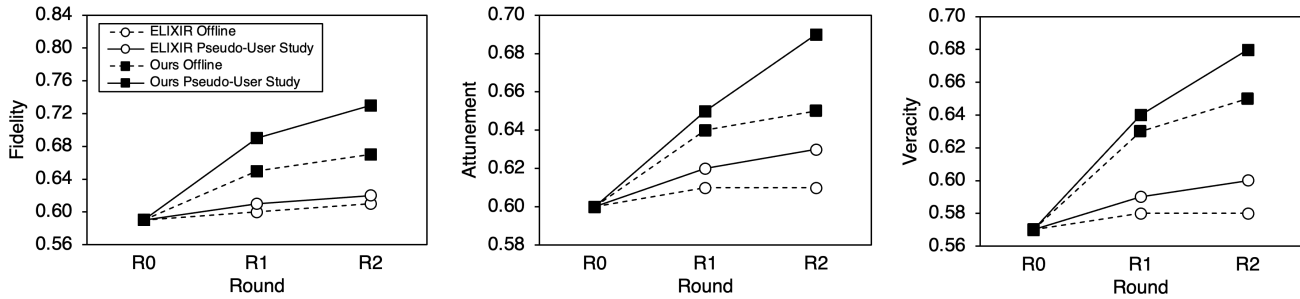


Figure 7: Explanation quality based on fidelity, attunement, and veracity score over iterations for the offline evaluation (solid line) and pseudo-user study (dashed line). These SDT sensitivity scores range from 0.5 (no detection) to 1 (perfect; see [17]).

0.46 at R1 and 0.59 at R2, compared to ELIXIR’s 0.21, 0.32, and 0.39. Recall followed a similar trend. In the offline evaluation, recall for our framework increased from 0.037 at R0 to 0.066 at R1 and 0.079 at R2, whereas ELIXIR increased from 0.037 to 0.050 and 0.066. In the pseudo-user study, our framework increased from 0.037 at R0 to 0.077 at R1 and 0.091 at R2, while ELIXIR improved from 0.037 to 0.053 and 0.074. Overall, both methods benefited from iterative feedback, but our framework achieved comparatively higher gains in both precision and recall. Note also that recall values remain lower than precision due to the limited recommendation list size (10 per user) compared to the average number of 56 user-liked games.

Figure 7 shows the improvement in explanation quality in terms of the fidelity, attunement, and veracity scores. Fidelity increased from 0.59 at R0 to 0.63 at R1 and 0.65 at R2 in the offline evaluation, and from 0.59 to 0.66 and 0.70 in the pseudo-user study. Attunement scores rose from 0.60 to 0.61 and 0.62 in the offline evaluation, and from 0.60 to 0.63 and 0.73 in the pseudo-user study. Finally, the combined veracity [17] score also increased from 0.57 (R0), to 0.61 (R1), and ultimately 0.63 (R2) in the offline evaluation, and from 0.57 to 0.63 and 0.68 in the pseudo-user study. Compared with ELIXIR, which showed steady but modest gains, our framework consistently achieved higher explanation quality across all three measures.

5 Discussion

This research introduces HIVE, a human-in-the-loop framework for XRSs that leverages veracity-based human feedback to iteratively refine and enhance both recommendation outputs and their associated explanations. HIVE is grounded in the veracity metric, which decomposes the information quality of explanations into two dimensions: fidelity and attunement. Human feedback along these dimensions is used to update user and item embeddings, which are then fed into the backbone explainable recommendation algorithm to improve future results. Because the embeddings are updated after each round, HIVE inherently accommodates evolving or even conflicting feedback. In other words, changes in user preferences or inconsistencies across sessions are incorporated into the updated representations rather than being treated as errors. To assess the effectiveness of HIVE, we conducted two simulation experiments: an offline evaluation based on the dataset’s ground truth and a pseudo-user study with LLM agents. We hypothesized that our framework would improve both predictive recommendation performance and

explanation quality. Results across two iterations confirmed this hypothesis, demonstrating that veracity-based human feedback contributes to enhanced performance in XRSs.

As shown in 7, fidelity scores exhibited a consistent upward trend in both evaluations. This indicates that the iterative feedback loop effectively enhances the factual alignment of explanations with item features. Notably, the improvement was substantially larger for the pseudo-user study than for the offline evaluation. Similar trends were also seen for attunement and veracity, with larger improvement in attunement, particularly observed in the pseudo-user study going from R1 to R2. We attribute this to the offline evaluation being based on a fixed set of ground truth data, which led to more conservative and stable improvements. In contrast, the pseudo-user study employed flexible evaluation responses informed by each persona’s behavioral history. The LLM-based simulated feedback may have facilitated more effective refinement, potentially due to its ability to capture nuanced, user-aligned rationales not present in the static knowledge graph, especially for the attunement dimension. These findings highlight a potential major advantage of HIVE. Overall, performance improvements were consistently larger in the pseudo-user study compared to the offline evaluation. Thus, it appears that accounting for human subjective interpretation of information improves performance beyond what can be achieved by just looking at the ground truth.

Beyond these contributions, we demonstrated that HIVE outperformed the existing SOTA framework, ELIXIR. While both approaches leverage user feedback on explanations to refine future recommendations, the key distinction lies in how feedback is represented and utilized. ELIXIR relies on binary like/dislike signals on (rec, exp) pairs, which provide useful but coarse adjustments to user preference models. In contrast, HIVE collects feedback along two veracity dimensions and captures these as continuous degrees rather than binary judgments. These graded signals are directly incorporated into fidelity and attunement loss functions, allowing the system to update embeddings with finer granularity. Based on our results, this richer representation of feedback translates into more substantial and consistent improvements in both recommendation accuracy and explanation quality, highlighting the value of modeling explanation veracity beyond simple preference signals.

Although outside the primary scope of this research, one additional implication emerges from our pseudo-user feedback simulation. We observed that attunement scores, which capture alignment with user preferences, improved more substantially in the pseudo-user study than in the offline evaluation. This suggests that LLM-based personas may serve as a promising direction for addressing the long-standing cold-start problem in recommender systems. The cold-start problem, a critical challenge in RS research, arises because recommendation performance improves with larger amounts of user preference data. Yet, in real-world systems such data are often sparse or difficult to collect [8]. Existing work has largely focused on extracting the best possible performance from limited datasets. However, our findings indicate that a complementary approach could be to collect an initial sample of preference information from real users, instantiate an LLM-based target user persona, and then elaborate or augment this information to generate sufficient synthetic preference data. Feeding this enriched data into the recommendation algorithm may yield stronger performance and offer a practical path forward in mitigating cold-start scenarios.

While this study shows clear benefits of HIVE, it also raises questions for future research. First, while veracity-based feedback appeared to be successful in our analysis, it remains unclear how its performance will compare with that of other mechanisms utilizing human feedback. Along these same considerations, ELIXIR is based on user feedback concerning the overlapped feature of a recommended item and an explanation item, whereas HIVE is based on feedback about explanations itself. This is by design as veracity (and its fidelity and attunement dimensions) is meant to enable the assessment of explanations independently of recommendations. This suggests that additional information could be gained by HIVE also incorporating direct feedback about the recommended item. These subjects should be investigated in future analyses. Next, although HIVE is applicable to any XRS beyond those generating textual explanations, our experiments were limited to the assumption that explanations are in text format. Future work should investigate how veracity concepts can be translated and applied to other modalities. In addition, our evaluation was limited to a single game recommendation scenario; therefore, applying the framework to diverse datasets and domains will be necessary for demonstrating its generalizability. Finally, the results presented here are preliminary. Future work will seek to replicate the conditions of our simulation in a human subjects study to validate our findings. Together, these directions will be crucial for establishing HIVE as a practical framework that advances both the accuracy and user-centeredness of future recommender systems.

GenAI Usage Disclosure

GenAI tools were used during the research, but the authors are fully accountable for the content. A sufficient detailed statement of the exact use is included in this paper.

References

- [1] Charu C. Aggarwal. 2016. *Recommender Systems*. Springer International Publishing, Cham.
- [2] Jan Auernhammer. 2020. Human-centered AI: The role of Human-centered Design Research in the development of AI. In *DRS International Conference (Synergy)*. Design Research Society, Online, 11–14.
- [3] Savita Bhat, Ishaan Shukla, and Shirish Karande. 2025. Know Thyself: Validating Knowledge Awareness of LLM-based Persona Agents. In *Proceedings of the 5th Workshop on Trustworthy NLP*. Association for Computational Linguistics, Albuquerque, New Mexico, 321–334.
- [4] Shihao Cai, Jizhi Zhang, Keqin Bao, Chongming Gao, and Fuli Feng. 2024. FLOW: A Feedback Loop Framework for Simultaneously Enhancing Recommendation and User Agents. *arXiv preprint arXiv:2410.20027* (2024).
- [5] Xu Chen, Yongfeng Zhang, and Ji-Rong Wen. 2022. Measuring "Why" in Recommender Systems: A Comprehensive Survey on the Evaluation of Explainable Recommendation. *arXiv preprint arXiv:2202.06466* (2022).
- [6] Nik Davis. 2022. Steam Store Games. <https://www.kaggle.com/datasets/nikdavis/steam-store-games>. Accessed: 2025-05-30.
- [7] Azin Ghazimatin, Soumajit Pramanik, Rishiraj Saha Roy, and Gerhard Weikum. 2021. ELIXIR: Learning from User Feedback on Explanations to Improve Recommender Models. In *Proceedings of the Web Conference 2021*. ACM, Ljubljana Slovenia, 3850–3860.
- [8] Jyotirmoy Gope and Sanjay Kumar Jain. 2017. A survey on solving cold start problem in recommender systems. In *2017 International Conference on Computing, Communication and Automation*. 133–138.
- [9] Zhiguang Han and Zijian Wang. 2024. Rethinking the Role-play Prompting in Mathematical Reasoning Tasks. In *Proceedings of the 1st Workshop on Efficiency, Security, and Generalization of Multimedia Foundation Models*. ACM, Melbourne, VIC, Australia, 13–17.
- [10] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems* 43, 2 (March 2025), 1–55.
- [11] IGN Entertainment Inc. 2025. HowLongToBeat.com | Game Lengths, Backlogs and more! <https://howlongtobeat.com/> Accessed: 2025-06-05.
- [12] Sho Isogai, Shinpei Ogata, Yutaro Kashiwa, Satoshi Yazawa, Kozo Okano, Takao Okubo, and Hironori Washizaki. 2024. Toward Extracting Learning Pattern: A Comparative Study of GPT-4o-mini and BERT Models in Predicting CVSS Base Vectors. In *2024 IEEE 35th International Symposium on Software Reliability Engineering Workshops*. IEEE, Tsukuba, Japan, 127–134.
- [13] Weiqing Li, Yue Xu, Yuefeng Li, and Yinghui Huang. 2025. Display Content, Display Methods and Evaluation Methods of the HCI in Explainable Recommender Systems: A Survey. *arXiv preprint arXiv:2505.09065* (2025).
- [14] OpenAI. 2024. GPT-4o mini: Advancing Cost-Efficient Intelligence. <https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/>. Accessed: 2025-06-03.
- [15] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, Vol. 35. Curran Associates, Inc., 27730–27744.
- [16] William Peterson, Theodore Birdsall, and William Fox. 1954. The theory of signal detectability. *Transactions of the IRE Professional Group on Information Theory* 4, 4 (1954), 171–212.
- [17] Yeonbin Son and Matthew Bolton. 2025. Towards a Signal Detection Based Measure for Assessing Information Quality of Explainable Recommender Systems. In *2025 IEEE Conference on Artificial Intelligence*. IEEE, Santa Clara, CA, USA, 1–6.
- [18] Tamber. 2018. Steam Video Games. <https://www.kaggle.com/datasets/tamber/steam-video-games>. Accessed: 2025-05-30.
- [19] Valve Corporation. 2025. Welcome to Steam. <https://store.steampowered.com/>. Accessed: 2025-05-30.
- [20] Shendi Wang, Haoyang Li, Caleb Chen Cao, Xiao-Hui Li, Ng Ngai Fai, Jianxin Liu, Xun Xue, Hu Song, Jinyu Li, and Guangye Gu. 2022. Tower Bridge Net (TB-Net): Bidirectional Knowledge Graph Aware Embedding Propagation for Explainable Recommender Systems. In *2022 IEEE 38th International Conference on Data Engineering*. IEEE, Kuala Lumpur, Malaysia, 3268–3279.
- [21] Yongfeng Zhang and Xu Chen. 2020. Explainable Recommendation: A Survey and New Perspectives. *Foundations and Trends® in Information Retrieval* 14, 1 (2020), 1–101.