LLM-Based Evaluation Methodology of Explanation Strategies

Ege Soyarar[1(✉)], Reyhan Aydogan[2,3], Berk Buzcu[4], and Davide Calvaresi[4]

[1] Computer Science, Ozyegin University, Istanbul, Turkey
ege.soyarar@ozu.edu.tr
[2] Artificial Intelligence and Data Engineering, Ozyegin University, Istanbul, Turkey
reyhan.aydogan@ozyegin.edu.tr
[3] Interactive Intelligence, Delft University of Technology, Delft, The Netherlands
[4] University of Applied Sciences and Arts Western Switzerland (HES-SO Valais-Wallis), Sierre, Switzerland
{berk.buzcu,davide.calvaresi}@hevs.ch

Abstract. As data privacy regulations, such as the EU AI Act and EU Data Act, become increasingly stringent, processing real user data for AI models like movie recommendation systems has grown more challenging. Moreover, the human-centric data collection and evaluation of Explainable AI (XAI) systems are often costly and time-consuming; making it hard to sustain. Hence, this study adopts the Synthetic Behavior Generation (SBG) approach, leveraging large language models (LLMs) to evaluate AI explanations while ensuring compliance with regulations and providing cost-effective solutions for human feedback. To assess the quality of these explanations, we utilize three different LLMs, which are fed synthetically generated user behaviors to evaluate explanations of an AI system as if they were real users. The evaluation focuses on key criteria such as convincingness, clarity, accuracy, and the impact on decision-making, facilitating a thorough assessment of explanation effectiveness. The results indicated that LLMs can deliver structured and consistent evaluations based on the provided synthetic user behavior.

Keywords: Synthetic Data Generation · Explainable AI (XAI) · Recommender Systems · Large Language Models (LLMs) · Explanation Evaluation

1 Introduction

The rise of Artificial Intelligence (AI) has influenced various fields such as healthcare and entertainment. One of the significant applications of AI is in the development of recommendation systems such as movie or food recommendations. Since understanding and justifying recommendations are essential for user trust and system transparency [10,23], the recent recommendation systems do not

D. Calvaresi et al. (Eds.): EXTRAAMAS 2025, LNAI 15936, pp. 85–103, 2026.
https://doi.org/10.1007/978-3-032-01399-6_6

only provide recommendations but also generate explanations for the reasoning behind them to enhance user experience and build rapport with their users [5,29]. Here, one of the interesting research questions is how to evaluate the effectiveness of the automatically generated explanations [13]. Typically, user studies are conducted where participants share their perceptions. However, real-world limitations such as time cost and financial expenses restrict researchers from performing exploratory analyses, AI system testing, and evaluation [11]. Key challenges include participant recruitment, experiment setup, and carrying out user surveys. To address these issues, we propose using large language models (LLMs), whose increasing popularity has positioned them as effective tools for explanation evaluation, to assess the generated explanations [4,28].

In our proposal, we utilize large language models (LLMs) to simulate a synthetic user seeking recommendations and explanations. To effectively mimic user behavior, these LLMs can be trained on either real users' comprehensive data, which captures their preferences and interactions with the system, or synthetic user data. The emergence of privacy regulations, such as GDPR, the EU AI Act [19] and the EU Data Act [6], has posed significant challenges regarding the use of actual user data for training AI models. Consequently, our study emphasizes the generation of synthetic user behaviors, which serve as inputs for the LLMs, enabling them to function appropriately and assess AI systems. We propose a method for synthesizing realistic user behaviors in the domain of movie recommendations. This context provides a concrete use case where user preferences and interactions can be procedurally generated while preserving statistical properties of real-world datasets. The resulting recommendation system is then enriched with the generated synthetic data. Furthermore, certain segments of the generated behaviors could be utilized as a test set to gain insights into how well the recommendation systems align with these synthetic behaviors. By assessing the quality of generated explanations in terms of convincingness, clarity, accuracy, and their impact on decision-making, we aim to clarify the dual roles of synthetic behavior generation and LLMs as both compliance-driven tools and enhancers of user engagement. This paper presents a preliminary study on the evaluation of explanation strategies using Synthetic Behavior Generation (SBG) and LLM-based assessment.

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 details our synthetic behavior generation and LLM-based evaluation framework, while Sect. 4 describes the experimental design and results. Finally, Sect. 5 concludes the paper with future directions.

2 Related Work

XAI has emerged as a critical field aimed at making complex machine learning models more transparent and understandable to both end-users and system developers [8]. While numerous approaches in XAI have been developed, including model-agnostic methods such as LIME (Local Interpretable Model-agnostic Explanations) [24] and SHAP (SHapley Additive exPlanations) [17]. LIME is

a technique designed to explain the predictions of complex machine learning models by approximating them locally around a specific instance using inherently explainable surrogate models (e.g., decision trees). By perturbing input features and observing changes in the model's predictions, LIME extracts the influence of individual features for specific predictions. On the other hand, SHAP borrows from the field of cooperative game theory to compute feature contributions, via Shapley values [17], by distributing the prediction outcomes among features according to their marginal contribution across all possible combinations. Grad-CAM [25] operates by highlighting areas within an input image that significantly influence the model's classification decisions. The algorithm highlights areas within an input image that significantly influences the model's classification decisions. At each step, the gradients of the target prediction class are computed with respect to the feature maps from the previous convolutional layer. These gradients are then averaged to produce a weight for each feature map, which are then used to create a final heat map. Grad-CAM uses a visual approach to provide intuitive explanations to understand Convolutional Neural Network (CNN) outcomes.

Commonly, there is a distinctive shortcoming of experimental datasets with known ground-truth explanations, given the subjective nature of explanations [1]. Evaluating the accuracy and reliability of explainable models depends on having known, interpretable baseline explanations against which their generated feature attributions can be compared to. Accordingly, the use of synthetic data generated with certain rulesets offers a promising solution to evaluate these models [1]. These XAI methods often rely on ground-truth data for validation so there remains a significant challenge in validating these explanations reliably, often due to the absence of ground-truth interpretability data. Although datasets like Netflix Prize and MovieLens [3,12] are widely used in recommender systems, they lack detailed behavioral context such as mood, time, and viewing environment. This gap limits their suitability for evaluating explanation methods that depend on rich user behavior signals. Previous research leveraging synthetic datasets has demonstrated their utility in systematically explanation generation methods [7]. In the research, the SBG facilitates controlled experimentation by generating synthetic data based on predefined context parameters, user profiles, and probabilistic models, enabling large-scale simulations for training deep learning models and from the models explanations are generated. Additionally, synthetic datasets have facilitated the quantitative assessment of the accuracy, understandability, and comprehensibility of various interpretability techniques [20].

Furthermore, recent research within the field sought to incorporate LLMs into the generation and evaluation of self-explanatory mechanisms [15,16,22,27]. For example, Ludos *et al.* blends the concepts of XAI and the linguistic power of LLMs [16]. Specifically, their approach involves three key steps: (i) the classical feature importance methods (e.g., SHAP) extract the important features and quantify their relevance; (ii) these extracted features are then given as input to a transformer-based LLM (e.g., GPT models), which generates human inter-

pretable explanations detailing how these features interact with the predictions; and (iii) further interactive interfaces powered by the same models to allow users to respond with customized queries and receive explanations accordingly in a dialogic manner. In order to evaluate the effectiveness of the LLM-generated explanations, Wang *et al.* adopt both subjective and objective evaluation metrics [26]. Subjectively, they employ human-grounded evaluations, utilizing a 5-point Likert scale to assess dimensions such as understandability, satisfaction, completeness, usefulness, and trustworthiness. Objectively, accuracy metrics measure the correctness of model decisions made by human evaluators when aided by LLM-generated explanations compared to traditional explanations. Their research reveals that LLMs can effectively judge the quality of explanations, particularly within the context of subjective assessments. However, objective evaluations show that there is a high degree of variability, where the LLM evaluations may still differ significantly from human judgments in certain contexts; thus, highlighting a limitation of automated evaluators. This variability could further be assessed using synthetic data, given that the synthetic data could further be personalized to reduce the gap. Accordingly, Francesco *et al.* offer further evaluation of explanations through the use of LLMs utilizing synthetic data in their evaluation bench [4]. They evaluate the alignment of LLM responses with human judgments across different experimental conditions (e.g., high versus low domain familiarity scenarios). LLM-generated results were summated under various conditions (with or without memory, aggregated vs. individual responses) to assess the consistency and reliability of their responses. The findings show that LLM based systems can reliably replicate human conclusions in tasks requiring straightforward prediction from provided explanations, while showing discrepancies in subjective tasks, such as confidence assessment. To mimic human evaluation process, they utilize synthetic data and structured prompts to guide LLM interactions. Their results highlight the potential for LLM-based evaluations to address scalability and reproducibility challenges that arise in human-user studies, with similar limitations to the latest literature, such as potential model training biases and constraints in generalizing findings across broader XAI methods and domains.

Despite these advancements, current literature has not fully explored how varying the complexity and characteristics of synthetic datasets impacts the performance and perceived usefulness of explanation methods. This paper aims to bridge this gap by systematically investigating how synthetic data configurations influence the effectiveness and interpretability of XAI techniques, contributing to both theoretical insights and practical guidelines for future research.

3 Methodology

The proposed systematic approach aims to achieve the objectives of generating synthetic behavior data for recommendations, evaluating the interpretability of the recommendation model, and assessing the quality of explanations using LLMs in order to reduce human-centric costs associated with data collection,

evaluation, and decision-making processes in the development of personalized recommendation systems; and provide transparent and understandable explanations. Hence, the research is divided to 4 key phases; (1) synthetic behavior data generation, (2) model training and evaluation, (3) explanation generation, and (4) explanation evaluation using an LLMs.

The system can be shown in Fig. 1, begins with generating synthetic behavior data that simulates user interactions, including features such as liked genre, spoken language, movie duration, and so forth, designed to mimic real-world behavior while enabling controlled experimentation. The data is grouped by user and split into training and test sets for personalized model training and evaluation. AI-Powered model generates recommendations according to the training input. Explanation Generator creates the proper explanations thanks to training data and feature importance values that are then calculated to interpret predictions and identify key features, which are used to generate natural language explanations linking feature importance, values, and outcomes. Finally, the explanations are evaluated by considering historical user data, recommendations via LLMs to assess their convincingness, understandability, accuracy, and impact on decisions.

3.1 Synthetic Behavior Data Generation

The process of user generation follows a probabilistic modeling approach, where attributes are assigned to each user based on predefined distributions. This ensures that the simulated users resemble real-world audience demographics, behaviors, and preferences. Each user is uniquely identified and has a set of attributes defining their demographics, geography, lifestyle, movie preferences, and behavioral constraints.

User Generation. Each generated user is assigned a unique identifier (userID), name, surname, and a collection of attributes that define their demographics, geographic location, language proficiency, lifestyle, preferences, and constraints.

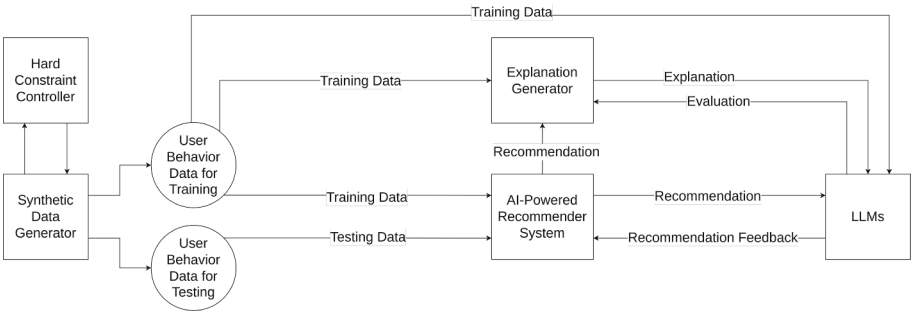


Fig. 1. Proposed System Architecture.

Demographic attributes form the foundation of user profiles in the synthetic dataset. These attributes define age, gender, ethnicity, marital status, employment status, and lifestyle. Each demographic attribute is assigned based on weighted probability distributions, ensuring some groups are more prevalent while maintaining variability. This approach allows the dataset to mimic population trends and regional differences, improving the reliability of behavioral simulations.

- **Gender**: Male, Female, Not Mentioned
- **Age Range**: Users are grouped into eight age categories: Under 13, 14–17, 18–24, 25–34, 35–44, 45–54, 55–64, 65+
- **Ethnicity**: Non-Hispanic White, Hispanic, Black, Asian, Mixed, Other
- **Marital Status**: Single, Married, Divorced, Widowed
- **Employment Status**: Employed, Student, Unemployed, Retired
- **Lifestyle**: Active, Sedentary, Balanced, Busy, Relaxed

Additionally, attributes have dependencies to maintain realism. Logical dependencies include:

- A person under 18 must have "Single" marital status or "Student" employment status.
- A person aged 65+ is likely to be "Retired."
- A person aged 18-24 is more likely to be a "Student" than in other employment categories.

Geographic and linguistic attributes play a crucial role in defining user profiles, influencing content accessibility, cultural preferences, and viewing habits. The framework integrates country of origin, living country, current location, and spoken languages as key attributes. A user's country of origin is probabilistically assigned based on global population distributions. Migration tendencies are modeled, allowing a portion of users to relocate with controlled probabilities, reflecting real-world mobility trends. A user's current location is determined based on their living country, with major cities assigned according to urban population densities. Language familiarity significantly impacts content accessibility and user engagement. Users are assigned spoken languages based on their country of origin's official languages, with additional probabilities accounting for multilingualism due to regional influences. For example, a user from Spain is more likely to know Italian than a user from Turkey, reflecting linguistic similarities and geographical proximity.

To model diverse user behaviors, the framework incorporates genre preferences, aversions, and award sensitivity, ensuring meaningful distinctions in taste and selection patterns. Some users consistently watch a narrow set of genres, while others explore a broad variety.

- **Liked Genres**: Each user has up to 3 preferred genres.
- **Disliked Genres**: Each user has up to 3 disliked genres (mutually exclusive with liked genres).

– **Award Sensitivity**: A binary trait indicating whether a user prefers critically acclaimed movies. Approximately 20% of users are labeled as "Award Hunters," affecting behavioral constraints and satisfaction scores.

Hard constraints define strict rules restricting what a user can watch under specific circumstances. These constraints ensure realistic behavior patterns, preventing unlikely or unrealistic viewing scenarios. Each user has a 20% probability of receiving a hard constraint, assigned based on demographic attributes in Table 1.

Table 1. Filtering constraints and their rules

Constraint	Filtering Rule
under_13_constraint	Excludes NC-17 and R movies, if user is under 13
13_17_constraint	Excludes NC-17 movies, if user's age in between 13 and 17
no_morning_thriller_horror	Excludes Thriller and Horror movies in the morning
no_long_movie_constraint	Excludes movies longer than 2 h, if lifestyle is "Active" or "Busy"
strict_award_hunter	Includes only award-winning movies
only_known_languages	Includes only movies in the user's spoken languages
no_constraint	All movies available

Beyond demographic, geographic, linguistic, preference-based attributes and hard constraints, each user is also assigned individualized probabilistic tendencies that influence their movie-watching behaviors. These probabilities dictate when, where, and how users engage with movies, ensuring that behavior generation remains realistic and personalized. Each user is assigned five key behavioral probabilities, a numerical score (1 to 5) for the tendency of the watch and weights that will be used to calculate satisfaction score:

– **Companion Preferences**: Determines how often the user watches alone, with friends, family, partner, or group.
– **Seasonal Preferences**: Indicates if a user prefers watching movies more in certain seasons; winter, spring, summer, autumn.
– **Day-of-Week Preferences**: Defines whether a user is more likely to watch movies on days of week.
– **Time-of-Day Preferences**: Dictates if a user prefers watching movies in the morning, afternoon, evening, or at night.
– **Location Preferences**: Determines where a user is most likely to watch movies (home, cinema, public transportation, workplace, and community center).
– **Watch Tendency**: A numerical score (1 to 5) that affects how persistent a user is in attempting to watch a movie.
– **Satisfaction Weights**: Each user is assigned a unique set of satisfaction weights that influence how different factors contribute to their perceived enjoyment of a movie. These weights probabilistically determine the impact of genre compatibility, disliked genres, language familiarity, IMDb ratings, user mood, rewatch history, and award recognition on the final satisfaction score.

By allowing individual users to prioritize different aspects of a movie, this component ensures a more nuanced and user-specific satisfaction evaluation.

By assigning each user personalized and unique probabilistic profiles, the framework ensures that; user behaviors are diverse rather than deterministic, seasonal, weekly, and daily preferences shape behavior dynamically and location, companionship, and persistence vary per user, preventing unrealistic uniformity. These user-specific probabilities directly influence movie-watching decisions, ensuring that generated behaviors align with real-world tendencies.

Behavior Generation and User Satisfaction Estimation. The behavior generation phase simulates user interactions with movies over a defined period, incorporating personal preferences, contextual influences, and predefined constraints. This approach ensures that the generated behaviors align with real-world movie-watching tendencies. Additionally, a user satisfaction estimation model quantifies how well a selected movie aligns with user expectations.

The process consists of three main stages: Initialization, Behavior Decision, and Behavior Recording. This structured methodology enables the creation of realistic user behavior data, capturing the complexity of human decision-making in movie-watching scenarios. Before simulating user behavior, the system initializes key components, including user profiles, preferences, and movie data preparation. User and preference data are generated as described in Sect. 2. The movie dataset is then preprocessed into filtered subsets, ensuring that all hard constraints (e.g., age restrictions, language preferences) are accounted for. The simulation is based on a day-driven framework, where each user's movie-watching decisions are evaluated on a daily basis over the defined period. This time-driven approach enables the model to capture temporal variations such as seasonality, day-of-week preferences, and other time-dependent factors. To facilitate this, a day-mapping mechanism systematically assigns each simulation day to a corresponding season and day of the week.

For each simulated day, the system iterates over every user to determine whether they will watch a movie. This event decision is guided by a multi-step process that begins with computing the base probability of watching a movie, which is derived by the user's seasonal, day-of-week and time-of-day probabilities for that particular day. The Watch Tendency Mechanism acts as a persistence factor, enabling multiple attempts—up to the assigned watch tendency value—if the initial probability does not immediately result in a movie-watching decision. Hence, if the watch probability is exceeded during any of the attempts, a movie is selected from the movie subset according to the user's hard constraint; otherwise, no behavior is recorded for that user.

Once a user decides to watch a movie, contextual factors (e.g., location, companion, etc.) are assigned based on predefined probabilities. Thus, the output format can be show in the Table 2.

After completing all the movie watching event information, the system calculates the user's satisfaction score using a weighted function that integrates multiple factors via Eq. 1. The significant factors are:

Table 2. User Behavior Data

Column Name	Description	Example Value
day_number	Number that represent day	0
date	The simulated calendar date for the recorded behavior	2025-01-01
season	The season corresponding to the given date	Winter
day_of_week	The day of the week	Wednesday
time_of_day	The specific period of the day when the user watched the movie (Morning, Afternoon, Evening, Night)	Evening
userId	Unique identifier assigned to the user	U0016
movieId	Unique identifier of movie watched	M00780
location	The location where the user watched the movie (Home, Cinema, Workplace, etc.)	Home
companions	The group with whom the user watched the movie (Alone, Partner, Friends, Family, Group)	Friends
user_mood	The emotional state of the user before watching the movie (Happy, Neutral, Sad)	Sad
satisfaction_score	The computed satisfaction score (0 to 1) representing the user's experience	0.56

- **Genre Compatibility**: Positive influence from liked genres, negative influence from disliked genres.
- **Language Match**: Whether the movie language aligns with the user's known languages.
- **IMDb Rating**: Movies rated below 5.5 are penalized, while higher-rated movies contribute positively.
- **User Mood**: Satisfaction is adjusted based on mood (higher for happy users, lower for sad users).
- **Rewatching Factor**: If a movie is rewatched more than one by other users, it has a good impact on candidate user.
- **Award Bonus**: If an "Award Hunter" user watches an awarded movie, an additional boost is applied.

$$\begin{aligned} \text{satisfaction} = {} & (\text{liked\_genre\_match\_score} \times w_{\text{liked\_genre\_match}}) + \\ & (\text{disliked\_genre\_match\_score} \times w_{\text{disliked\_genre\_match}}) + \\ & (\text{language\_match} \times w_{\text{language\_match}}) + \\ & (\text{imdb\_rating\_normalized} \times w_{\text{imdb\_rating}}) + \\ & (\text{user\_mood\_score} \times w_{\text{user\_mood}}) + \\ & (\min(\text{number\_of\_rewatches}, 3) \times w_{\text{rewatch\_factor}}) + \\ & (\text{award\_bonus} \times w_{\text{award\_bonus}}) \end{aligned} \tag{1}$$

While preprocessing the generated data, Jaccard Similarity was utilized to quantify the alignment between user preferences and movie attributes [2]. Specifically, Jaccard scores were calculated for language, liked genres, and disliked genres, capturing the degree of similarity between a user's preferences and a movie's characteristics. To illustrate, a user likes "Horror" and "Action" as genres; a movie's genres are "Horror" and "Comedy". Then, the Jaccard Similarity would be equal to $1/4$. These similarity scores were then incorporated as features in the tree-based models.

Lastly, the "recommended" column is created to personalize movie suggestions based on user satisfaction score. For each user, movies with a satisfaction score above their average are marked as 1 (recommended), while others are marked as 0. This ensures recommendations are user-specific rather than based on a fixed threshold.

3.2 Model Training and Feature Importance

The model training phase involved developing personalized recommendation models for each user using their synthetic behavior data. Decision trees offer inherent advantage of interpretability due to its hierarchical structure and clear decision-making logic [5]. When combined with feature importance values, the explainability is further enhanced by providing a fair contribution of each feature to the prediction [18]. Each user's data is partitioned into training and testing sets, and recommender models are trained on user-specific datasets. Afterwards, each tree's predictions are interpreted by using feature importance values, which quantify the contribution of each feature (e.g., movie duration, companion) to the recommendation outcome. To ensure clear and persuasive explanations, the

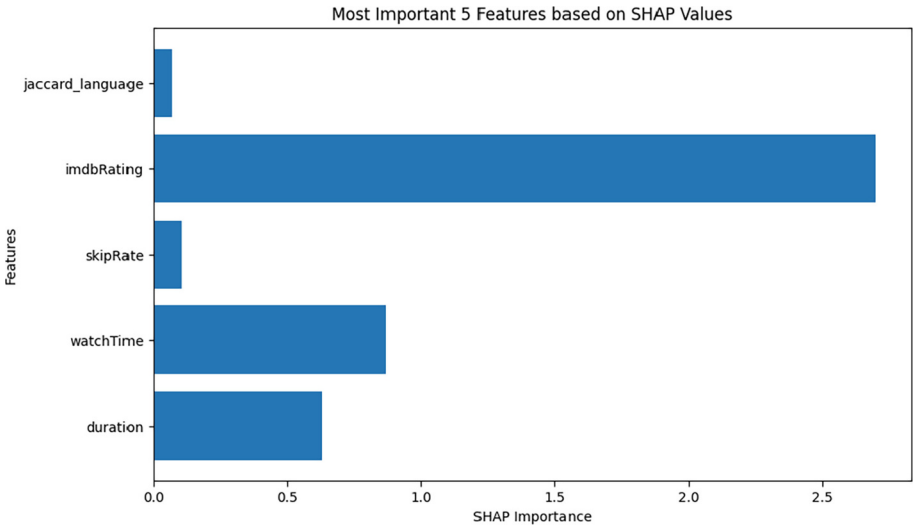


Fig. 2. Example Feature Importance Bar Chart for user "U0099".

top five most important features are utilized for generating explanations as in the Fig. 2.

3.3 Explanation Generation

This research introduces a Post-Hoc explanation generation method to enhance the understandability, transparency, and user engagement of a recommender system, improving the clarity of recommendations. The combination of Decision Trees and feature importance values aids in understanding the impact of features in machine learning models, unlike other complex models [14]. The extracted feature importance values from the model, and the five most significant features are selected for explanation.

The strategy for generating user-specific explanations is outlined in Algorithm 1. First, a model is trained with user data in (Line 2), then calling the function to get prediction from recommendation which is the data that needs to be explained (Line 3). The most important five feature importance values are extracted by calling function of model (Line 4). The system evaluates the prediction (Line 5), determining whether a content was suggested (label = 1) or not (label = 0). Next, feature importance values are analyzed to assess their influence on the recommendation (Line 6). If a user's feature aligns negatively with a highly influential attribute (feature importance value less than 0) and recommendation behavior's feature matches with the negative effect (Line 7), a negative explanation is generated using predefined negative explanation templates (Line 8). Lastly, the final sentence is selected from "NOT_RECOMMENDATION_MAPPING" to explain behavior is not recommended (Line 11). Same feature importance analysis is applied if prediction is equal to 1- (Line 12-13). Conversely, if an influential feature is positively contributes to the recommendation (feature importance value greater than 0) and recommendation behavior's feature has positive impact (Line 14), a positive explanation is selected from the corresponding positive explanation vocabulary (Line 15). Finally, a sentence about pointing the recommendation is added (Line 18).

Determining whether a feature value has a positive influence is straightforward for binary columns. However, for some numeric features; Jaccard Similarity for language, liked and disliked genres—the impact is less explicitly defined. To address this, a threshold-based approach is applied: if the Jaccard Similarity for a feature is greater than 0, it is considered positive; otherwise, it is treated as negative. This method ensures a consistent interpretation of similarity-based features within the recommendation model. For other numeric features, values below the user's average are considered negative, while values above are positive. For example, if a duration is 60 min, but the user's average preferred duration is 100 min, 60 min is classified as a negative value. An explanation is retrieved for each feature and once this process is completed for five features, the final explanation is constructed. This process repeats for each user until all test data is processed, making the explanations ready for LLM evaluation.

Algorithm 1. Generate Explanations for Recommendations

Require: training_data: A dataset containing user-specific behavior
recommendation: A single behavior that needs to be explained
Ensure: explanation
Initialize an empty list *explanations*
model = GenerateUserModel(training_data)
prediction = model.predict(recommendation)
feature_importance = model.GetTop5FeatureImportance()
if prediction == 0 **then**
  for each feature in *feature_importance* **do**
    if feature["FeatureImportanceValue"] < 0 AND recommendation[feature] == False **then**
      Select a negative explanation from *EXPLANATION_VOCAB*
    end if
  end for
  Append a negative sentence to explanation from *NOT_RECOMMENDATION_MAPPING*
else
  for each feature in *feature_importance* **do**
    if feature["FeatureImportanceValue"] > 0 AND recommendation[feature] == True **then**
      Select a positive explanation from *EXPLANATION_VOCAB*
    end if
  end for
  Append a positive sentence to explanation from *RECOMMENDATION_MAPPING*
end if
return *explanation*

3.4 LLM-Based Evaluation

The evaluation of the generated explanations is a critical and novel step in assessing the effectiveness and usability of both synthetic behavior data and recommendation, since human surveys and participation for data collection and evaluation are time consuming and expensive [4]. The research that compares cost of human LLM shows that dramatic difference in expenses [28]. In addition to broad domain knowledge, the fact that LLMs can be used as judge (LLM-as-a-Judge) thanks to the prompt makes them valuable for this study.

The LLM-based evaluation focuses on four key criteria: convincingness, understandability, accuracy, and decision impact. They all are designed to measure how persuasive, clear, accurate, and impactful the explanations are in the context of the recommendation system. The evaluation process begins with the preparation of inputs, where the historical behavior data of user, recommended behavior and generated explanations are formatted into a structured input for the LLM. The goal of passing historical data is that LLM behaves like the real user and evaluate the explanation. Additionally, detailed instructions, input and output format and examples, evaluation criteria questions are passed in the Prompt for Evaluating Movie Recommendations.

Prompt for Evaluating Movie Recommendations
You are an AI evaluator tasked with analyzing movie recommendations for a user. You will be provided with the following data for each test case:

- **Historical Behavior Data**: A DataFrame containing the user's past behavior (e.g., movies watched, liked, disliked, etc.). Contains whether the user actually liked (1) or disliked (0) the movie
- **Recommendation**: A recommended behavior from AI-Powered Recommender.
- **Explanation**: The explanation provided by the model for the prediction.

For each test case, answer the following questions:

1. **How convincing is the explanation to watch/not watch the movie?** Rate between 0 and 10, where 0 is not convincing at all and 10 is extremely convincing.
2. **How easy is the explanation to understand?** Rate between 0 and 10, where 0 is very difficult to understand and 10 is very easy to understand.
3. **How accurate is the explanation according to your desire?** Rate between 0 and 10, where 0 is completely inaccurate and 10 is completely accurate.
4. **Does the explanation affect your decision to change your real decision?** Answer as "Yes", "No", or "Same".

Example Input:

- Historical Behavior Data: A DataFrame with columns like "movieId", "duration", "genres", "language", etc.
- Test Behavior Data: A DataFrame with columns like "movieId", "duration", "genres", etc.
- Predictions: [1, 0, 1]
- Explanations: ["The movie's genre matches your preferences, which increased the likelihood of recommendation.", "The movie's duration is too long, which decreased the likelihood of recommendation.", "The movie has won awards, which increased the likelihood of recommendation."]
- Real Values: [1, 1, 0]

Example Output:

```
[
    [7, 9, 8, "Same"],
    [9, 9, 8, "Yes"],
    [3, 7, 5, "No"]
]
```

4 Results

The purpose of this section is to present the outcomes of our approach tailored for the movie domain case study. 100 users are generated by SBG with unique 4893 samples. Tree-based model, SHAP values are determined as recommendation model and feature importance illustration respectively for personalized movie recommendation system and explanation generation. To handle explanation evaluation, three LLMs are selected; deepseek-r1 [9], gpt-4o-mini and gpt-3.5-turbo [21]. Expectation from LLMs is answering the evaluation questions below:

- How convincing is the explanation to watch/not watch the movie? Rate between 0 and 10, where 0 is not convincing at all and 10 is extremely convincing.
- How easy is the explanation to understand? Rate between 0 and 10, where 0 is very difficult to understand and 10 is very easy to understand.
- How accurate is the explanation according to your desire? Rate between 0 and 10, where 0 is completely inaccurate and 10 is completely accurate.
- Does the explanation affect your decision to change your real decision? Answer as "Yes", "No" and "Same".

To determine the most suitable tree-based model for personalized movie recommendations, three different models are evaluated: CatBoost (max_depth=5, learning_rate=0.1), Decision Tree (max_depth=5), and Random Forest (max_depth=5, n_estimator=50). Table 3 illustrates a comparative analysis of their performance across Accuracy, Precision, Recall, and F1 Score.

Table 3. Comparison of Tree-based Models for Movie Recommendation

Model	Accuracy	Precision	Recall	F1 Score
CatBoost	0.75	0.63	0.68	0.58
Decision Tree	0.78	0.67	0.69	0.65
Random Forest	0.75	0.68	0.71	0.65

This comparison was made using the entire dataset. The Decision Tree model achieved the highest accuracy (0.78) while maintaining a balanced trade-off between precision (0.67) and recall (0.69), resulting in an F1 score of 0.65. Although the Random Forest model exhibited slightly higher precision and recall, the Decision Tree was selected due to its superior accuracy, lower computational complexity, and enhanced interpretability.

Since the users have different amounts of test data, they are classified into "Small"(1 to 3 sample), "Mid"(4 to 6 sample), and "Large" (7 and above) Scale Data groups. Consequently, the further evaluations were answered based on these data categories.

Table 4. Comparison of User Groups with Decision Tree Model

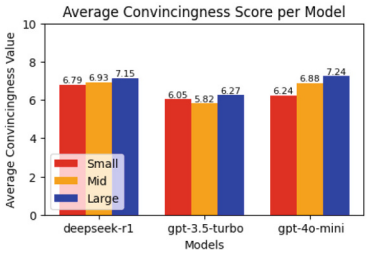
User Groups	Accuracy	Precision	Recall	F1 Score
Small	0.40	0.39	0.40	0.40
Medium	0.65	0.64	0.65	0.65
Large	0.88	0.61	0.88	0.88

Table 4 compares four average performance metrics of Decision Tree Model—Accuracy, Precision, Recall, and F1 Score—across three user groups of different sizes. For the Small group, the model shows relatively low performance, suggesting that limited user data hinders the model's ability to correctly identify and classify positive instances. In contrast, the Medium group exhibits moderate improvements in all metrics (Accuracy = 0.65, Precision = 0.64, Recall = 0.65, F1 = 0.65), indicating that increased data contributes to more reliable predictions. Notably, the Large group achieves a high Accuracy (0.88) and Recall (0.88), as well as an elevated F1 Score (0.88), which implies that extensive user data substantially enhances the model's ability to correctly identify true positives. However, Precision (0.61) is lower relative to Recall (0.88), suggesting that while the model captures most relevant cases, it also predicts more positives that may not be correct. Overall, these results underscore the importance of data availability for achieving robust and well-balanced predictive performance.

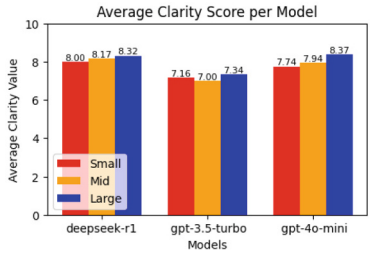
The Fig. 3a demonstrates the average convincingness scores across LLMs and three user groups. Across all models, an increase in the scale of data groups correlates with an improvement in the convincingness score. In other words, larger datasets lead to higher-quality, more persuasive explanations with a solid background, likely because the model has more information to learn patterns and make more informed decisions. As a comparison of LLMs, gpt-4o-mini provides more persuasive explanations when ample user data is available. In contrast, gpt-3.5-turbo displays relatively lower scores across all user groups, implying that its explanations are generally perceived as less convincing compared to the other models. Meanwhile, deepseek-r1 exhibits a steady and moderately high performance, maintaining balanced convincingness across different user group sizes.

As a next metric, the clarity score is evaluated across various models and user groups. From the Fig. 3b , it can be observed that deepseek-r1 achieves the highest clarity for the Small (8.00) and Mid (8.17) groups, indicating that its explanations are perceived as more comprehensible when user data is relatively limited or moderate. However, gpt-4o-mini attains the highest clarity (8.37) in the Large group, suggesting that it provides clearer explanations when substantial user data is available. In contrast, gpt-3.5-turbo exhibits lower clarity scores (ranging from 7.00 to 7.34) for all user group sizes. Similar to the Convincingness, high amount of data provides more clear explanations to the LLMs.

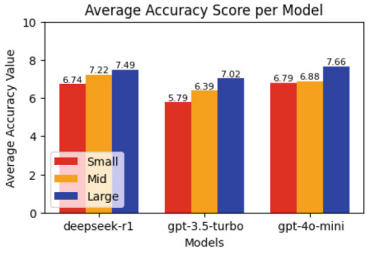
The Fig. 3c illustrates that accuracy finding via LLMs. In general, each model's accuracy increases with user group size, indicating that more extensive



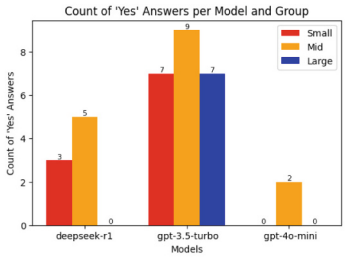
(a) Convincingness Score comparison between LLMs and User Groups



(b) Clarity Score comparison between LLMs and User Groups



(c) Accuracy Score comparison between LLMs and User Groups



(d) Impact of Explanations on Decision Change

Fig. 3. LLM Evaluation Metrics.

user data enhances recommendation precision. gpt-4o-mini achieves the highest score (7.66) in the Large group, deepseek-r1 maintains a steady improvement from 6.74 (Small) to 7.49 (Large), while gpt-3.5-turbo starts lower (5.79) but rises to 7.02, emphasizing the role of sufficient user data in boosting accuracy.

The last metric is about how often users reported that an explanation caused them to change their real decision ("Yes") across different language models. In this Fig. 3d , The gpt-3.5-turbo model yields the highest count of "Yes" responses for all user groups comparing with the other models. Meanwhile, other models records few or zero "Yes" responses, it means that they are more attached to their own preferences.

5 Conclusion

The increasing impact of data regulations, such as EU Data Act and EU AI Act, has needed innovative approaches to AI model development and evaluation. Our study explored a novel approach to evaluating explainable AI systems thanks to LLMs using SBG without using human-centric data. There are two crucial findings, one of them underscores that the potential of SBG effectively simulate coherent user behavior from various user background. Using synthetic data significantly reduces time and financial costs compared to real datasets

while offering a practical and efficient alternative. Secondly, LLMs as an evaluator provide structured, consistent evaluations of explanation quality, focusing on crucial aspects such as convincingness, clarity, accuracy, and impact on decision-making like a real user behaved.

By giving LLMs synthetic user behaviors, we found that LLMs are able to provide detailed and accurate evaluations, much like human evaluators do with a moderate amount of data. The capability of LLMs to analyze and interpret synthetic data, and evaluate explanations, represents a significant advancement in the development of trustworthy AI. This methodology not only eases the evaluation process but also provides a robust framework for sustainable improvement on AI applications. Especially, this advantage is particularly beneficial for the sectors that privacy concerns are prior such as healthcare, education.

As future work, extending this approach to other domains, such as healthcare or e-commerce, and developing standardized metrics for explanation quality will further advance the field. In this work, a Decision Tree is employed as the AI model due to its simplicity and interpretability; however, more advanced implementations, such as Neural Networks, can also be utilized for improved performance. Different prompting techniques or fine-tuned LLMs that specialized for a user might improve the user engagement and satisfaction. Finally, hybrid approaches combining LLM evaluations with targeted human feedback could balance scalability with feedback, brings us to more robust and trustworthy AI systems.

References

1. Amiri, S.S., et al.: Data representing ground-truth explanations to evaluate XAI methods. CoRR **abs/2011.09892** (2020). https://arxiv.org/abs/2011.09892
2. Ayub, R., Ghazanfar, M.a., Maqsood, M., Saleem, A.: A Jaccard base similarity measure to improve performance of CF based recommender systems, pp. 1–6 (01 2018). https://doi.org/10.1109/ICOIN.2018.8343073
3. Bennett, J., Lanning, S.: The netflix prize (2007)
4. Bona, F.B.D., Dominici, G., Miller, T., Langheinrich, M., Gjoreski, M.: Evaluating explanations through LLMs: Beyond traditional user studies (2024). https://arxiv.org/abs/2410.17781
5. Buzcu, B., et al.: Towards interactive explanation-based nutrition virtual coaching systems (2023). https://doi.org/10.21203/rs.3.rs-3110083/v1
6. Casolari, F., Buttaboni, C., Floridi, L.: The EU data act in context: a legal assessment. Int. J. Law Inform. Technol. **31**(4), 399–412 (02 2024). https://doi.org/10.1093/ijlit/eaae005
7. Contreras, V., Schumacher, M., Calvaresi, D.: Explanation of deep learning models via logic rules enhanced by embeddings analysis, and probabilistic models. In: Calvaresi, D., et al. (eds.) Explainable and Transparent AI and Multi-Agent Systems, pp. 155–183. Springer Nature Switzerland, Cham (2024)
8. Das, A., Rad, P.: Opportunities and challenges in explainable artificial intelligence (xai): A survey (2020). https://arxiv.org/abs/2006.11371
9. DeepSeek-AI, Guo, D., et al.: Deepseek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning (2025). https://arxiv.org/abs/2501.12948

10. Ge, Y., et al.: A survey on trustworthy recommender systems **3**(2) (Nov 2024). https://doi.org/10.1145/3652891
11. Gonzales, A., Guruswamy, G., Smith, S.R.: Synthetic data in health care: a narrative review. PLOS Digital Health **2**, e0000082 (2023). https://doi.org/10.1371/journal.pdig.0000082
12. Harper, F.M., Konstan, J.A.: The movielens datasets: history and context. ACM Trans. Interact. Intell. Syst. (tiis) **5**(4), 1–19 (2015)
13. Hulstijn, J., Tchappi, I., Najjar, A., Aydoğan, R.: Metrics for evaluating explainable recommender systems. In: Calvaresi, D., et al. (eds.) Explainable and Transparent AI and Multi-Agent Systems, pp. 212–230. Springer Nature Switzerland, Cham (2023)
14. Le, T.T.H., Kim, H., Kang, H., Kim, H.: Classification and explanation for intrusion detection system based on ensemble trees and shap method. Sensors **22**(3) (2022). https://doi.org/10.3390/s22031154
15. Lin, C.S., Tsai, C.N., Su, S.T., Jwo, J.S., Lee, C.H., Wang, X.: Predictive prompts with joint training of large language models for explainable recommendation. Mathematics **11**(20) (2023). https://doi.org/10.3390/math11204230
16. Lubos, S., Tran, T.N.T., Felfernig, A., Polat Erdeniz, S., Le, V.M.: LL,-generated explanations for recommender systems. In: Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization, pp. 276–285. UMAP Adjunct '24, Association for Computing Machinery, New York, NY, USA (2024). https://doi.org/10.1145/3631700.3665185
17. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Guyon, I., et al., (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 4765–4774. Curran Associates, Inc. (2017). http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf
18. Mathew, J., Chitra, R., Stephen, C., Koshy, R.S.: Integration of explainable artificial intelligence (xai) in the development of disease prediction and medicine recommendation system. In: 2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI), vol. 2, pp. 1–5 (2024). https://doi.org/10.1109/IATMSI60426.2024.10503250
19. Novelli, C., Casolari, F., Rotolo, A., Taddeo, M., Floridi, L.: Taking AI risks seriously: a new assessment model for the AI act. AI and Society **39**(5), 2493–2497 (Jul 2023). https://doi.org/10.1007/s00146-023-01723-z
20. Oblizanov, A., Shevskaya, N., Kazak, A., Rudenko, M., Dorofeeva, A.: Evaluation metrics research for explainable artificial intelligence global methods using synthetic data. Appl. Syst. Innov. **6**(1) (2023). https://www.mdpi.com/2571-5577/6/1/26
21. OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Others: Gpt-4 technical report (2024). https://arxiv.org/abs/2303.08774
22. Pang, A., Jang, H., Fang, S.: Generating descriptive explanations of machine learning models using LLM. In: 2024 IEEE International Conference on Big Data (BigData), pp. 5369–5374 (2024). https://doi.org/10.1109/BigData62323.2024.10825667
23. Radanliev, P., Santos, O., Brandon-Jones, A., Joinson, A.: Ethics and responsible AI deployment. Front. Artif. Intell. **7** (2024). https://doi.org/10.3389/frai.2024.1377011
24. Ribeiro, M.T., Singh, S., Guestrin, C.: "why should I trust you?": Explaining the predictions of any classifier. CoRR **abs/1602.04938** (2016). http://arxiv.org/abs/1602.04938

25. Selvaraju, R.R., Das, A., Vedantam, R., Cogswell, M., Parikh, D., Batra, D.: Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. CoRR **abs/1610.02391** (2016). http://arxiv.org/abs/1610.02391
26. Wang, B., Li, Y., Zhou, J., Chen, F.: Can LLM assist in the evaluation of the quality of machine learning explanations? (2025). https://arxiv.org/abs/2502.20635
27. Zhang, J., Liu, J., Luo, D., Neville, J., Wei, H.: Llmexplainer: Large language model based Bayesian inference for graph explanation generation (2024). https://arxiv.org/abs/2407.15351
28. Zhang, X., et al.: Large language models as evaluators for recommendation explanations, pp. 33–42 (2024). https://doi.org/10.1145/3640457.3688075
29. Zhang, Y., Chen, X.: Explainable recommendation: a survey and new perspectives. Found. Trends® Inform. Retrieval **14**(1), 1–101 (2020). https://doi.org/10.1561/1500000066