Orclever
Vol. 5 No. 1 (2025)
DOI: 10.56038/ejrnd2026526734
www.orclever.com
The European Journal of Research and Development
e-ISSN: 2822-2296

EARS-XTSK: Privacy-Preserving Global Explainability in Cross-Silo Federated Two-Tower Recommendation Systems via Server-Side TSK Fuzzy Rule Distribution

Deniz Altay Avcı*

Adesso Turkey

0009-0006-6443-6008

* Corresponding author: denizaltayavci@gmail.com

DOI: 10.56038/ejrnd2026526734

Received: 2025-09-10 · Revised: 2025-11-06 · Accepted: 2025-12-24 · Published: 2025-12-30

Abstract

Recommender systems form the both the commercial and the computational backbone of personalization in modern digital ecosystems—from video streaming and e-commerce to news, finance, and social platforms. Yet, the data centralization they require conflicts with privacy regulations such as GDPR and KVKK, which are emphasized more than ever by both the policymakers and the end users, motivating the adoption of privacy-first and transparent solutions. In the privacy context, a prominent approach is Federated Learning (FL). Yet, on its own, it lacks the transparency and interpretability that a centralized solution offers under the same circumstances. We present EARS-XTSK, a privacy-preserving global explainability framework for cross-silo federated recommendation. Our industry setting comprises two real-life data providers in Türkiye: (i) a large online forum representing the user metadata side, and (ii) a commercial streaming platform providing item/content and interaction data. By design, raw user data never leaves partner premises. To enable the global FED-XAI layer without direct data access, we construct a faithful synthetic dataset conforming to the partners' schema, feature definitions, and scale, and we build an end-to-end clone of their pipelines to de-risk integration. Local sites train Two-Tower retrieval models (user tower on demographics/geo, item tower on multilingual text and metadata) and participate in NVFLARE-based FedAvg aggregation. As our core contribution, we introduce a server-side explainability engine that fuses Grad×Input saliency with a Takagi–Sugeno–Kang (TSK) fuzzy-rule layer to produce interpretable global explanations over item-side labels only (privacy-strict). We provide a complete, reproducible pipeline: dummy data generation (OMDb, SBERT, fastText), Two-Tower embedding/training, federated orchestration (NVFLARE), and Fed-XAI evaluation with visualization. Prototype runs demonstrate faithful global explanations without exposing user identifiers or raw features. In deployment, the same FED-XAI is practically guaranteed to work with the partners' real local Two-Tower models, since all upstream interfaces are schema-compatible.

Keywords: Reference: (will be filled by editorial office) Keywords: Recommender Systems, Federated Learning, Two-Tower Retrieval, Explainable AI, Federated Explainability, Fuzzy Rule-Based Systems, TSK Fuzzy Systems, NVFLARE., Recommender Systems

1. Introduction

1.1. Industrial and Scientific Context

Personalized recommendation is the driving factor behind engagement and monetization across **streaming**, **e-commerce**, **social**, and *news *platforms. In streaming media, recommendation quality directly impacts watch-time retention; in e-commerce, it influences click-through and basket size; in finance, it governs risk-aware product targeting; and in social platforms, it shapes information exposure. Modern stacks typically use a **retrieval → ranking** cascade, where a **Two-Tower** (dual-encoder) model supplies scalable candidate generation at web scale. At the same time, **GDPR/KVKK**, platform risk, and partner sensitivity make centralized data pooling and hence a "all-seeing explainability" approach increasingly untenable. *Federated Learning (FL)* mitigates this at the decision-making side by aggregating model updates instead of raw data. But **explainability in FL** is harder than in centralized setups: product teams and auditors need **consistent, global explanations**, while raw **per-user information must remain private**. This paper contributes EARS-XTSK, a practical, end-to-end solution:Our company (adesso Turkey) directs the global explainability (FED-XAI) layer, which is the main scope of this work.Because we cannot access real data, a

faithful synthetic twin had to be developed which matches schemas and feature ranges of two real providers (a large online forum and a Turkish streaming platform).Partners continue local research on real data for the non-XAI portions as well as local explainability methods, validating model utility and reinforcing the interpretability aspects of the EARS platform. Our server-side FED-XAI then operates on the aggregated Two-Tower model and surfaces global, item-only rationales via Grad×Input + TSK fuzzy fusion, never logging user attributes.Main contributions of our work can be summarized as follows:(1) A privacy-preserving, server-side explainability method for cross-silo Two-Tower models. (2) A reproducible pipeline: data generation, vectorization, Two-Tower training, NVFLARE federation, and FED-XAI evaluation/visualization, all schema-compatible with production partners.

1.2. Federated Learning for Recommendation Systems

Federated Learning (FL) enables collaborative model training without sharing raw data. Each client learns local parameters and only communicates updates—gradients or weights—to a central server performing an aggregation through certain aggregation algorithms designed for decentralized learning such as Federated Averaging (FedAvg).For personalization, FL must handle heterogeneous feature spaces and sparse interactions; the Two-Tower architecture has emerged as the industry standard for this setting because it naturally separates user and item representation learning. Let us begin by introducing the mathematics for a typical two-towerm topology.Formally, each tower learns an embedding function

$$u_i = f_{\{(\theta)_U\}}\left(x_i^{(U)}\right), v_j = f_{\{(\theta)_I\}}\left(x_j^{(I)}\right) \tag{1}$$

Where $x_i^{(U)}$ and $x_j^{(I)}$ denote user and item features, and the predicted relevance is

$$(\hat{y})_{ij} = \sigma(u_i v_j) \tag{2}$$

With $\sigma(\cdot)$ being the sigmoid activation function.The model is optimized with binary cross-entropy:$L_{\mathrm{BCE}} = -\sum_{(i,j)}\{[y_{ij}\ \log\ ((\hat{y})_{ij}) + (1-y_{ij})\ \log\ (((1-\hat{y})_{ij}))]\}$ (1)

1.3. From Federated Learning to Global Explainability

Existing FL systems focus on performance and privacy but seldom address *explainability*.Local XAI methods—e.g., SHAP, LIME, grad-CAM—cannot operate at the server level because by design, raw inputs strictly remain exclusive to client-side access.Global explainability instead interprets the *aggregated model* to reveal feature-group importance without the need of exposing any private data.EARS-XTSK implements this concept by performing gradient-based attribution and TSK fuzzy inference directly on the federated global model.

1.4. Project Structure and Partners

Two industrial partners provide the real use cases:Forum Partner (A): a large Turkish online discussion platform representing *user-centric* data.Streaming Partner (B): a national video-on-demand provider contributing *content-centric* metadata.Due to legal and organizational constraints, adesso Turkey—responsible for the global FED-XAI layer—does not access either partner's raw data.To enable development and testing, we built a synthetic yet schema-faithful dataset reproducing their data volumes, structures, and statistical properties.Local research at each partner validates all pre-XAI stages using real data, ensuring that when our global XAI module is deployed, it will operate seamlessly. A simplified overview of the current system architecture can be found in the Figure 1 provided below:

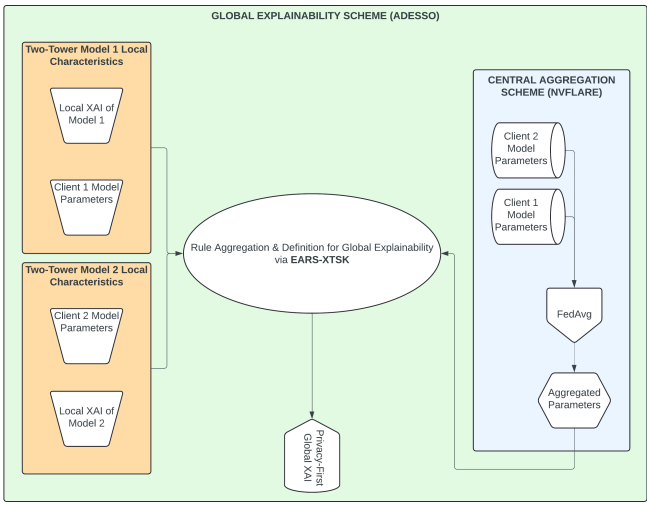

Figure 1: Cross-silo architecture of EARS-XTSK showing local Two-Tower models at Partner A and B, FedAvg aggregation at the central server, and server-side TSK Explainability.

Research on explainability in federated learning (FL), fuzzy rule-based systems, and global surrogate modeling has expanded rapidly in recent years. Furthermore, the usage of such systems for recommender systems is also becoming one of the frequently investigated topics. The EARS-XTSK framework sits at the intersection of these areas, especially where privacy-preserving cross-silo infrastructures must provide global transparency for deep recommendation models.

1.5. Federated Learning for Recommenders

Federated Learning (FL) enables collaborative model training without centralizing raw client data. McMahan et al. introduced Federated Averaging (FedAvg), showing that locally trained client models can be aggregated efficiently at the server and that the approach can remain effective under unbalanced and non-IID client data distributions [1]. Bonawitz et al. later proposed practical secure aggregation mechanisms that allow a server to compute aggregate model updates without observing each individual client contribution, strengthening the privacy guarantees of FL systems [2]. In a subsequent system-level study, Bonawitz et al. examined the engineering requirements of deploying FL at scale, including client orchestration, reliability, communication efficiency, and production constraints [3]. Kairouz et al. provide a comprehensive overview of FL research challenges, including statistical heterogeneity, privacy, security, communication bottlenecks, personalization, and robustness, all of which are directly relevant to federated recommender systems [4].FL has also been adapted to recommendation scenarios, where user interaction histories and preference signals are highly sensitive. Ammad-ud-din et al. proposed a federated collaborative filtering approach in which recommendation models are trained without requiring centralized access to user-item interaction data [5]. Qi et al. applied FL to news recommendation and incorporated privacy-preserving mechanisms to reduce the risk of exposing user interests during model learning [6]. Perifanis and Efraimidis introduced Federated Neural Collaborative Filtering, extending neural collaborative filtering into a federated setting and showing how recommender models can be trained while keeping user data decentralized [7]. Sun et al. survey federated recommendation systems and organize the field around data partitioning, privacy protection, communication efficiency, attacks and defenses, personalization, and open deployment challenges [8].These works demonstrate that FL is a viable training paradigm for privacy-preserving recommendation. However, their primary focus is model utility, privacy protection, and communication efficiency rather than global model interpretation. EARS-XTSK builds on this line of work but targets a different problem: explaining the behavior of a federated global recommender model after aggregation, without requiring access to raw user-side data.Large-scale recommendation systems commonly use dual-encoder or Two-Tower architectures to support efficient retrieval from large item corpora. Yi et al. studied large-corpus item recommendation with neural retrieval models and addressed sampling bias in candidate generation, highlighting the practical importance of scalable embedding-based recommendation [9]. This architectural logic aligns with EARS-XTSK, where user and item towers are trained locally and then aggregated through FL. However, Two-Tower embeddings are latent and similarity-driven, making their recommendation decisions difficult to interpret without an additional explanation layer.

1.6. Explainable AI in Federated Learning

Explainability in FL is more constrained than explainability in centralized machine learning because the explanation process itself must not reveal sensitive client-side information. Abadi et al. introduced deep learning with differential

privacy, showing how privacy-preserving noise mechanisms can be incorporated into gradient-based training to limit information leakage [10]. Truex et al. proposed a hybrid privacy-preserving FL approach that combines multiple privacy-enhancing techniques to reduce disclosure risks during collaborative training [11]. Hitaj et al. demonstrated that collaborative deep learning can leak sensitive information through adversarial reconstruction attacks, underscoring the need for privacy-aware model analysis and explanation mechanisms [12].Recent work has begun to examine the intersection of FL and explainability more directly. Daole et al. introduced OpenFL-XAI, a Python framework for federated learning of explainable artificial intelligence models, emphasizing the practical need for modular tools that combine federation and interpretability [13]. Zhang and Yu proposed LR-XFL, a logical-reasoning-based explainable federated learning framework in which local rule information is used to support more interpretable global reasoning [14]. Lopez-Ramos et al. reviewed the relationship between FL and XAI, concluding that the interaction between federated training dynamics and explanation quality remains an emerging and insufficiently studied area [15].These studies establish that FL and XAI can be combined, especially through rule-based or interpretable-by-design approaches. Nevertheless, existing FL-XAI research does not directly address the setting of EARS-XTSK: a server-side global explanation module for a deep federated Two-Tower recommender, where the server receives aggregated model parameters but cannot access raw user records. EARS-XTSK therefore extends this literature by producing global fuzzy-rule explanations from the aggregated recommender model while preserving strict separation from private client-side data.

1.7. Explainability in Deep Learning

Deep learning explainability provides the methodological basis for the attribution component of EARS-XTSK. Guidotti et al. survey black-box explanation methods and classify them across local explanations, global explanations, rule extraction, surrogate modeling, and feature-importance analysis [16]. Ribeiro et al. introduced LIME, a model-agnostic method that explains individual predictions by fitting local surrogate models around the instance being explained [17]. Lundberg and Lee proposed SHAP, which unifies additive feature-attribution methods through a Shapley-value-based theoretical framework and provides consistency properties for feature importance [18]. Sundararajan et al. introduced Integrated Gradients, an attribution method for differentiable models grounded in sensitivity and implementation-invariance axioms [19].Gradient-based explanation methods are especially relevant for neural recommendation models because they can be computed directly from differentiable architectures. Simonyan et al. introduced saliency maps for neural networks by using gradients to identify input dimensions that strongly influence model outputs [20]. Ancona et al. analyzed gradient-based attribution methods and clarified the relationships among gradient, Gradient×Input, Integrated Gradients, DeepLIFT, and related techniques [21]. EARS-XTSK adopts Gradient×Input because it is computationally lightweight, compatible with neural Two-Tower models, and feasible for server-side evaluation over privacy-filtered or synthetic records.However, raw attribution vectors are often too high-dimensional and technical for direct use by product teams, auditors, or end users. For this reason, EARS-XTSK does not stop at feature-level saliency. Instead, it uses Gradient×Input as an intermediate attribution mechanism and then transforms attribution patterns into TSK fuzzy-rule explanations that provide more compact and human-readable global interpretation.

1.8. TSK Fuzzy Systems and Interpretable Surrogates

Fuzzy systems provide a natural foundation for converting continuous model behavior into interpretable linguistic rules. Zadeh introduced fuzzy set theory, establishing the idea that elements can belong to sets with graded membership rather than binary inclusion [22]. Takagi and Sugeno introduced fuzzy identification of systems, where fuzzy antecedents are combined with functional consequents to model nonlinear systems in an interpretable rule-based form [23]. Sugeno and Kang further developed structure identification for fuzzy models, contributing to the systematic construction of fuzzy model structures from data [24]. Wang and Mendel proposed a method for generating fuzzy rules by learning from examples, showing how rule bases can be constructed from observed data rather than manually specified alone [25].These foundational works support the use of TSK fuzzy systems as interpretable surrogates for complex model behavior. In EARS-XTSK, fuzzy membership functions map latent similarity and attribution-derived signals into linguistic regions such as low, medium, and high contribution. This allows the system to summarize the global behavior of a federated Two-Tower model without exposing raw user-side features.Recent fuzzy explainability work further supports this design choice. Aghaeipoor et al. proposed fuzzy rule-based explainer systems for deep neural networks, demonstrating how fuzzy rules can support both local explainability and broader global understanding of black-box neural models [26]. Ribeiro et al. introduced Anchors, a model-agnostic explanation method that produces high-precision propositional rules for individual predictions [27]. Lou et al. proposed intelligible models for classification and regression, showing that generalized additive structures can provide interpretable feature-level effects while maintaining predictive performance [28]. Caruana et al. later demonstrated the practical value of intelligible additive models in healthcare

risk prediction, where interpretability is essential for high-stakes decision-making [29].EARS-XTSK follows the same broader interpretability principle but applies it to a different technical setting. Rather than replacing the recommender with an inherently interpretable model, it preserves the deep federated Two-Tower architecture and adds a server-side TSK fuzzy-rule layer that acts as an interpretable global surrogate over the aggregated model's behavior.

1.9. Model Interpretation Techniques for Recommenders

Explainable recommendation has been widely studied in centralized settings. He et al. introduced Neural Collaborative Filtering, replacing traditional matrix factorization interaction functions with neural architectures capable of learning nonlinear user-item interactions [30]. Although this work is not primarily an explainability method, it is important because it shows how latent neural representations became central to modern recommendation, thereby motivating the need for post-hoc interpretation techniques. Wang et al. proposed explainable reasoning over knowledge graphs for recommendation, using path-based reasoning to connect users and items through interpretable relational structures [31]. Xian et al. introduced reinforcement knowledge graph reasoning, where an agent learns paths over a knowledge graph to generate both recommendations and inspectable reasoning chains [32]. Chen et al. proposed visually explainable recommendation, using visual attention mechanisms to connect recommendation decisions to interpretable image regions [33].These approaches show how modern recommendation models motivate different forms of interpretation: neural collaborative filtering highlights the opacity of learned user-item interaction functions, while graph-based and visually grounded recommenders provide explanations through graph paths, reasoning chains, attention mechanisms, or visual evidence.EARS-XTSK therefore addresses a distinct form of recommender interpretability. Its goal is not to explain a centralized recommender using complete user histories, but to explain an aggregated federated Two-Tower model through privacy-preserving global summaries. By combining Gradient×Input with TSK fuzzy rules, EARS-XTSK produces recommendation explanations that are compatible with cross-silo deployment constraints and item-side-only logging.

1.10. Synthetic Data, Privacy, and Governance

Synthetic data are useful when real data cannot be shared because of legal, privacy, or organizational constraints. Gonçalves et al. studied the generation and evaluation of synthetic patient data and emphasized that synthetic data should be assessed in terms of both utility and privacy risk [34]. Van Breugel et al. cautioned that synthetic data can introduce real downstream errors if treated as a perfect substitute for real data, highlighting the importance of clearly distinguishing between schema-faithful development data and behaviorally equivalent real-world data [35].This distinction is central to EARS-XTSK. The synthetic EARS dataset is not used to claim that synthetic user behavior perfectly reproduces partner behavior. Instead, it functions as a schema-faithful engineering and validation substrate that enables the development, testing, and reproducibility of the FED-XAI workflow without requiring access to private partner records.The implementation layer of EARS-XTSK is also shaped by practical FL deployment requirements. Roth et al. introduced NVIDIA FLARE as a framework for moving federated learning from simulation to real-world multiparty collaboration, supporting heterogeneous workflows and privacy-preserving distributed model development [36]. The official NVIDIA FLARE documentation defines the framework's job orchestration, aggregation, component, and deployment mechanisms, which directly inform the practical organization of the EARS-XTSK federated training and server-side explainability pipeline [37].

Overall, the reviewed literature supports the individual components of EARS-XTSK: federated training, privacy-preserving recommendation, neural attribution, fuzzy-rule explanation, recommender interpretability, synthetic-data-based development, and FL orchestration. The gap lies in their integration. EARS-XTSK contributes a unified framework for privacy-preserving global explainability in cross-silo federated Two-Tower recommendation, where server-side Gradient×Input attributions are transformed into TSK fuzzy rules without exposing raw user data.

2. Methodology

2.1. Synthetic Data Generation

2.1.1. Design and Metadata Sources

The dataset generator algorithm synthesizes user–item interactions using the OMDb API for realistic, verifiable and exhaustive movie/series metadata.It defines 10 demographic user classes distinguished by age, gender, and urbanity, each with *preferred genre tags* guiding positive and negative sample creation. There are five classes for each gender. Each gender has one outlier's class as well as the four main socioeconomical and geolocational breakdown-based classes.

2.1.2. User and Item Representation

Each user vector $x^{(U)} in (\mathrm{R})^3$ encodes:Gender (0 = female, 1 = male)Normalized age $a^{\sim} = \frac{a - a_{\min}}{a_{\max} - a_{\min}}$Location index normalized to [0, 1].Which implies $d_u = 3$.Item tower concatenates the following features with given dimensions:

Table 1: Synthetic data fields and dimensionalities used in the EARS pipeline.

Component	Source	Dimensionality
SBERT (title)	sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2	384
SBERT (summary)	sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2	384
fastText (keywords)	Truncated to	64
fastText (directors)	Truncated to	32
fastText (casts)	Truncated to	64
Year Scalar	Normalized	1
Genres	Multi-hot	G

Hence, we obtain:$d_I = 384 + 384 + 64 + 32 + 64 + 1 + \mid G \mid = 929 + \mid G \mid$with typical $|G| \approx 20 - 40$ and hence $d_I \approx 950$.

2.1.3. Interaction Labelling

Our test dataset contains 2000 users. Each user interacts with a random number of items between 5 and 50; for each positive interaction, a configurable ratio of negatives is also sampled. The final dataset $D = \left(x_i^{(U)}, x_j^{(I)}, y_{ij}\right)$ provides the foundation for both centralized and federated experiments.

2.1.4. Two-Tower Representation Learning Model Architecture

Implemented in Keras 3, the Two-Tower model comprises:u $= L2\left(W_2^{(U)} \phi\left(W_1^{(U)} x^{(U)} + b_1^{(U)}\right)\right)$, v $= L2\left(W_3^{(I)} \phi\left(W_2^{(I)} \phi\left(\left(W_1^{(I)} x\right)^{(I)} + b_1^{(I)}\right) + b_2^{(I)}\right) + b_3^{(I)}\right)$ (2)? Where $\phi(\cdot)$ is the ReLU and L2 normalization enforces $\mid$ u $(\mid)_2 = \mid$ v $(\mid)_2 = 1$. The final prediction follows Eq. (1). Training uses Adam optimizer with learning rate $(10)^{-3}$ and mini-batch 32.

2.1.5. Caching and Vocabularies

Each embedding call is cached to embed_cache.pkl; supporting JSONs store vocabularies (vocab_genre.json, vocab_location.json) and scaling parameters (age_minmax.json, year_minmax.json). This process, combined with seed generation logic, lays the groundwork for reliable and fast metadata reproducibility across experiments where different parametrical settings are tested.

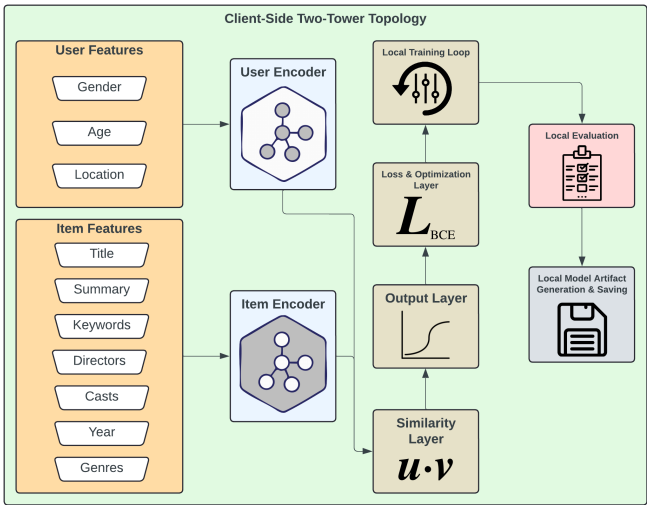

Figure 2: Two-Tower topology with Layer Summaries.

2.2. Federated Learning Setup

2.2.1. Federated Averaging

Each client c trains their two-tower model locally and sends their respective weights w_c and sample counts n_c. The server then computes the weighted mean:

$$w^{(t+1)} = frac\left\{ \sum_{c\left(n_c w_c^{(t)}\right)} \right\}\left\{ \sum_{c(n_c)} \right\}(3) \tag{3}$$

This is implemented through InTimeAcccumulateWeightedAggregator in NVFLARE 2.5.

2.2.2. Persistor, Client Executors, Staging and Orchestration and Resulting Algorithm

The Simple Persistor saves each aggregated model asglobal_round_XXXX_YYYYMMDDThhmmssZ.h5 with a companion manifest provided in Figure 3 below:

```
{
 "round_index": 25,
 "timestamp_utc": "20251112T084201Z",
 "weights_shape": [[3,64], ...],
 "source": "aggregated"
}
```

Figure 3: A Typical Companion Manifest.

Each client loads its shard, trains for a specified number of epochs and batch size, then returns a Shareable whose payload is a dictionary {w_0, ..., w_k}. Meanwhile, The Bash scripts and job-driven (rather than command-driven) architecture of our repository automate workspace creation, template seeding, and simulation launch. The process can be summarized in an algorithmic representation provided in Figure 4 below:

```
Algorithm 1: Federated Scatter-and-Gather Loop
for round t = 1 to T do
 Server broadcasts w^(t)
 Clients train locally → Δw_c
 Server aggregates via Eq.(3)
 Persistor saves global_round_t
end for
```

Figure 4: Federated Scatter-and-Gather Loop.

2.2.3. Federated Explainability (FED-XAI)

In a nutshell, the FED-XAI module operates after federated training has produced a sequence of aggregated global models. Its goal is to evaluate the latest global model and to generate privacy-preserving, rule-based global explanations using two complementary mechanisms:Gradient × Input local attributionTSK-FRBS (Takagi–Sugeno–Kang Fuzzy Rule-Based System) global similarity-based reasoningThe workflow can be decomposed into stages and can be illustrated as shown in Figure 5 below:

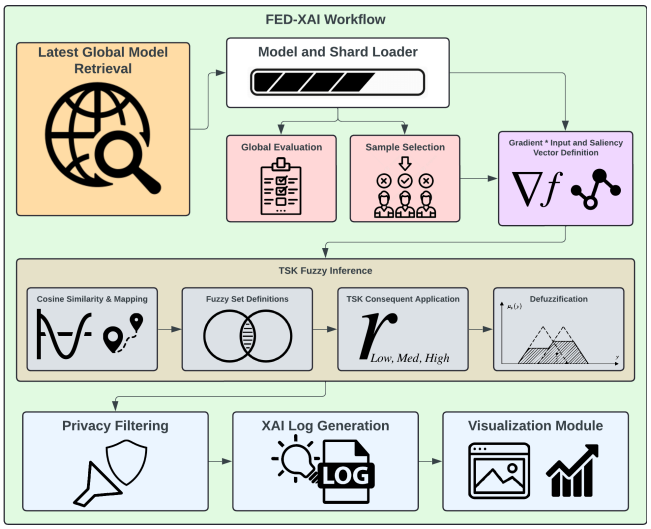

Figure 5: FED-XAI Workflow Illustration.

2.2.4. Rationale

While each client can interpret its local model, only the server-aggregated model reflects *collective* behavior. FED-XAI thus computes explanations at the server on the aggregated $y^{\{\widehat{()}\}} = \sigma((\mathrm{u})^{\top}\ \mathrm{v})$ surface, without ever receiving the raw inputs.

2.2.5. Gradient x Input Attribution

For each test sample $\left(x^{(U)}, x^{(I)}\right)$, we compute

$$g_U = frac\{\partial\widehat{}(y)\}\left(\partial x^{(U)}\right), g_I = frac\{\partial\widehat{}(y)\}\left(\partial x^{(I)}\right) \tag{4}$$

And define saliency vectors

$$s_U = g_U odot(x)^{(U)}, s_I = g_I odot(x)^{(I)}(4) \tag{5}$$

The concatenated explanation e then becomes

$$e = [s_U; s_I] in(\mathrm{R})^{d_U + d_I} \tag{6}$$

Feature-group contributions are obtained by summing $|(\mathrm{s})_{U,I}|$ over predefined index ranges.

2.2.6. TSK Fuzzy-Rule Scoring

To translate embedding similarity into interpretable linguistic scores, we employ a TSK Fuzzy Rule-Based System which carries out the following operations:Compute cosine similarity

$$c = \frac{u \cdot v}{(\{\|\ u\ \|)_2 (\|\ v\ \|)_2 + \varepsilon\}}\ ,\ \mathrm{m\ a\ p\ p\ e\ d\ t\ o}\ [0, 10]\ \ \mathrm{a\ s\ s{=}(c{+}1)\ dot.c\ 5} \tag{7}$$

Define fuzzy sets (triangles)

$$(\mu)_{\mathrm{L\ o\ w}}(s) = \mathrm{t\ r\ i\ m\ f}\ (s; 0, 0, 5), (\mu)_{\mathrm{M\ e\ d}}(s) = \mathrm{t\ r\ i\ m\ f}\ (s; 2, 5, 8), (\mu)_{\mathrm{H\ i\ g\ h}}(s) = \mathrm{t\ r\ i\ m\ f(s;5,\ 10,\ 10)} \tag{8}$$

Apply TSK consequents

$$r_{\mathrm{L\ o\ w}} = 0.2, r_{\mathrm{M\ e\ d}} = 0.5, r_{\mathrm{H\ i\ g\ h}} = 0.9 \tag{9}$$

Defuzzify

$$TSK_{\mathrm{s\ c\ o\ r\ e}} = frac\left\{\sum_{k\{(\mu)_k(s) r_k\}}\right\}\left\{\sum_{k\{(\mu)_k(s)\}}\right\}(5) \tag{10}$$

2.2.7. Privacy-Preserving Logging

Each explanation log L_i contains:

$$L_i = (\mathrm{id}, \mathrm{prediction}, \mathrm{TSK}_{\mathrm{score}}, \mathrm{ruleoutputs}, \mathrm{e}) \tag{11}$$

If –privacy-user-strict is active, as it is the case within the EARS platform's privacy-first design, user identifiers and labels are replaced by null.

2.2.8. Evaluation Protocol, Results and Discussion

The evaluation consists of three axes:FL Performance: metrics (AUC, Acc, BCE) across 50 FL rounds.XAI Performance: Visualization and Discussion of the ExplanationsPrivacy verification: Ensure absence of raw user metadata in any server-side artifact or explainibility module.

3. Results

- The BCE Loss, AUC and Accuracy vs. FL Rounds plots are provided in figures 6 and 7 below. The figures show that there are modest yet clear improvements in all three metrics over a 50-round federated learning simulation. The modesty of the improvements is due to the following factors:
- When weight drift was investigated, the following results were obtained: Norm round 1: 48.25, Norm round 50: 319.81, Delta norm: 298.63, Mean absolute per-weight change: 0.267, Cosine similarity round1 vs round50: ≈ 0.50 (substantial rotation). This shows that the weights change substantially, but only their effects on global evaluation metrics stay modest. This is caused by the fact that the initial, client-side models were already near a performance ceiling. Also, parametric optimization, especially the number of allowed local epoches and enabling decaying learning rate based on round count was observed to be able to substantially increase the effect of aggreagation on global evaluation metrics. Hence, since the main scope of our work is Federated Explainability and not Federated Learning, the current results were found to be sufficient for us to be able to move on with the next step.
-

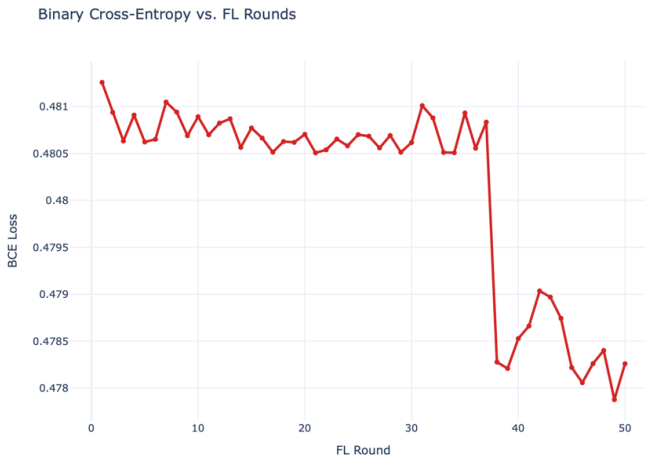

Figure 6: BCE Loss vs. FL Rounds.

-

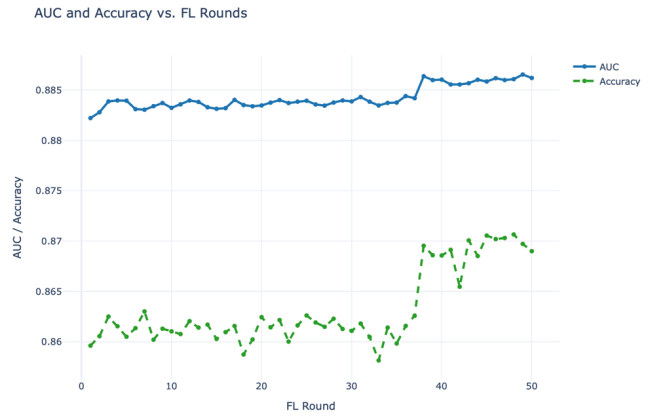

Figure 7: AUC & Accuracy vs. FL Rounds.

Vol. 5 No. 1 (2025)
DOI: 10.56038/ejrnd2026526734
www.orclever.com
The European Journal of Research and Development
e-ISSN: 2822-2296

3.1. XAI Performance

- The following figures (Figure 8 and 9) provide two different item suggestions for two different users along with the visualizations of TSK explanations. Here, we can see that each recommendation has occurred due to different amounts of contribution from different user or item properties. For instance, the recommendation of Psycho (1960) was based more on locational features and movie titles (titles are the short descriptions of the movies, not their actual titles). Its more detailed summary shows half as less contribution. Age and gender were the trailing factors for this recommendation. The remaining fields provided lower contributions. The movie's drama genre was also a large contributor. It can be deduced that such a user would like content in Drama genre. The Space Dandy (2014) recommendation, on the other hand, was based primarily on gender field. Also, this time, the summary had nearly as much effect as the title field. The trailing contributors are age and location data, as well as the genres of the item. It can be deduced that a user with similar characteristics to this user would like to watch Comedy and Animation genres.
- After temporarily lifting the privacy checks and verifying with said users' metadata in the original dataset, it has been seen that the users in fact contain similar items watched in alignment with the below recommendations. Hence, despite the privacy constraints, EARS-XTSK Federated Explainability scheme is able to deliver accurate and detailed feature importance information that can easily be converted into linguistic, end-user friendly insights for a recommender engine.

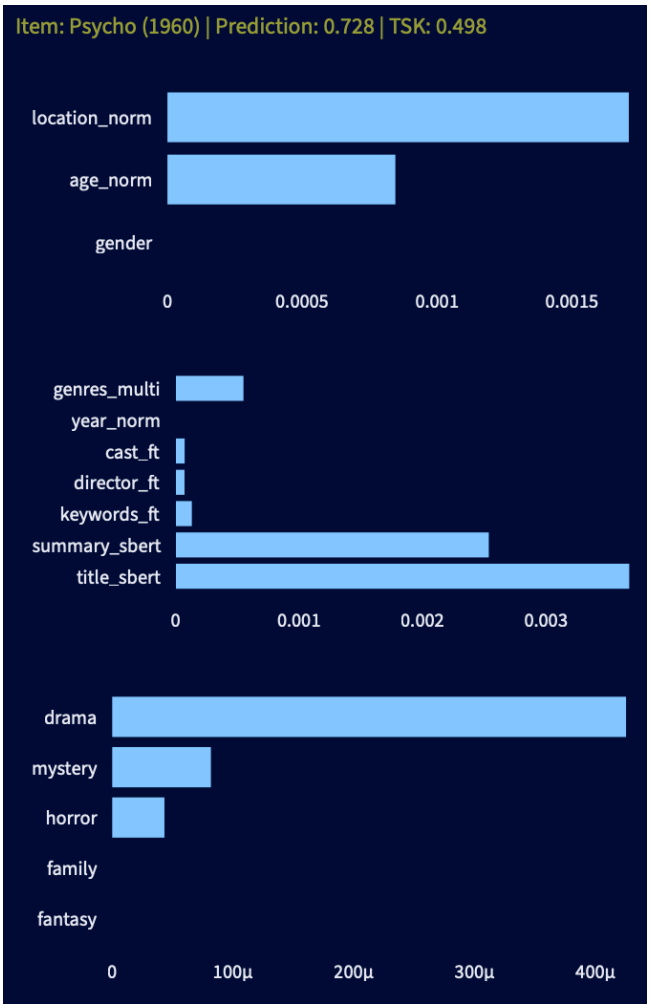

Figure 8: Explanations for Suggesting Psycho (1960) to a User.

-

Vol. 5 No. 1 (2025)
DOI: 10.56038/ejrnd2026526734
www.orclever.com
The European Journal of Research and Development
e-ISSN: 2822-2296

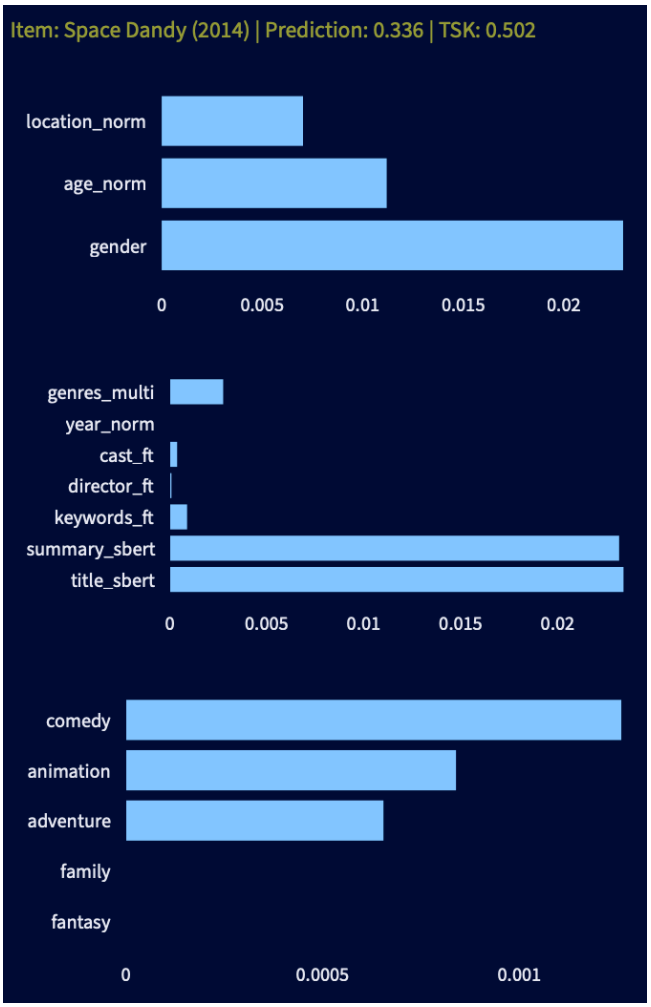

Figure 9: Explanations for Suggesting Space Dandy (2014) to a User.

3.2. Privacy Verification

- Thanks to our privacy-preserving logging approach, no sensitive user raw data was observed to be accessible by the global explainability module. The variable names, types, descriptions and their privacy status are provided in Table 2 below:

Table 2: Fields stored in each JSON explanation log with privacy levels.

Field	Type	Description	Privacy level
prediction	float	Model score in [0, 1] for the user–item pair.	Public
tsk_score	float	TSK-FRBS overall fuzzy score aligned with the prediction.	Public
rule_outputs	object	Per-rule fuzzy activations, e.g., {rule_name: strength}.	Public
explanation	object/array	Grad×input attributions. Usually provided as concatenated vector or split as {user: […], item: […]}.	Public (no PII)
item_label	object/string	Item-side label(s) for readability (e.g., title, genres).	Public (item metadata only)
model_checkpoint	string	Filename of the aggregated model used.	Internal (path/filename only)
timestamp	string (ISO-8601)	Time when the explanation was produced.	Public
item_id	string/int	Internal item identifier used by the pipeline.	Public/Internal
feature_names	object	Names for features, e.g., {user: […], item: […]}.	Public
groups	object	Item feature grouping (e.g., genres, keywords, year).	Public
top_k	object	Convenience view of top-K features by absolute attribution.	Public
dataset	string	Source dataset path or tag used for labels.	Internal
split	string	Which split was used (e.g., test).	Public
y_true	0/1	Ground truth label if logging enabled.	Restricted (omitted under strict privacy)
user_id	string/int	User identifier.	Private (omitted under strict privacy)
user_label	object/string	Any user-side descriptive label.	Private (omitted under strict privacy)

- Practical Implications
- Industrial Readiness
- Because NVFLARE already supports on-prem orchestration and secure aggregation, EARS-XTSK can be deployed without major backend/infrastructure changes.The explainability outputs (TSK scores + Grad×Input vectors) are JSON-serializable and can feed any BI or compliance dashboard.
- Regulatory Alignment
- Under GDPR Art. 22 and KVKK Art. 11, users have the right to *"meaningful information about the logic involved."*Our fuzzy linguistic rules ("Low / Medium / High similarity") satisfy this requirement while maintaining zero access to personal data.
- Integration w/ Partner Environments
- The same pipeline can run inside each partner's secure enclave, returning only aggregated global models to adesso Turkey.Hence, the explainability analysis remains compliant yet operationally consistent across real and synthetic domains.

DiscussionThis paper presented EARS-XTSK, an end-to-end, privacy-preserving, globally explainable recommender system architecture.By uniting:

A synthetic data generator matching industrial schemas,Cross-silo Two-Tower training under NVFLARE FedAvg, andA novel *server-side TSK fuzzy-rule explainability module* grounded in grad×input attributions,we achieve inter-

pretable global reasoning without accessing any raw user data.The framework demonstrates that federated architectures can provide not only accuracy but also compliance-grade transparency.

4. Discussion

4.0.1. Industrial Readiness

Because NVFLARE already supports on-prem orchestration and secure aggregation, EARS-XTSK can be deployed without major backend/infrastructure changes.The explainability outputs (TSK scores + Grad×Input vectors) are JSON-serializable and can feed any BI or compliance dashboard.

4.0.2. Regulatory Alignment

Under GDPR Art. 22 and KVKK Art. 11, users have the right to _“meaningful information about the logic involved.”_Our fuzzy linguistic rules (“Low / Medium / High similarity”) satisfy this requirement while maintaining zero access to personal data.

4.0.3. Integration w/ Partner Environments

The same pipeline can run inside each partner’s secure enclave, returning only aggregated global models to adesso Turkey.Hence, the explainability analysis remains compliant yet operationally consistent across real and synthetic domains.

5. Conclusion

- Future work is currently in progress and contains trainable neuro-fuzzy consequents jointly optimized with the global model, secure aggregation of client-side local explainability histograms, differential-privacy calibration for explanation vectors and domain-specific visual dashboards for analysts that employs fine-tuned LLMs, aiming to generate end user-friendly, verbal explanations as to why a certain item has been recommended to them.

6. Acknowledgements

This work is conducted under the EARS Project (Environment Adaptive Recommendation System), an ITEA project whose FED-XAI sections are being coordinated by adesso Turkey with contributions from partner organizations in the media and online-community sectors.We thank all collaborators for providing project coordination, use-case, data and client schemas alongside validation support.

7. References

[1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in Proc. 20th Int. Conf. Artificial Intelligence and Statistics (AISTATS), 2017, pp. 1273–1282.

[2] K. Bonawitz et al., “Practical secure aggregation for privacy-preserving machine learning,” in Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS), 2017, pp. 1175–1191.

[3] K. Bonawitz et al., “Towards federated learning at scale: System design,” in Proc. Machine Learning and Systems (MLSys), vol. 1, 2019.

[4] P. Kairouz et al., “Advances and open problems in federated learning,” Foundations and Trends in Machine Learning, vol. 14, no. 1–2, pp. 1–210, 2021.

[5] M. Ammad-ud-din, E. Ivannikova, S. A. Khan, W. Oyomno, Q. Fu, K. E. Tan, and A. Flanagan, “Federated collaborative filtering for privacy-preserving personalized recommendation system,” arXiv:1901.09888, 2019.

[6] T. Qi, F. Wu, C. Wu, Y. Huang, and X. Xie, “Privacy-preserving news recommendation model learning,” arXiv:2003.09592, 2020.

[7] V. Perifanis and P. S. Efraimidis, “Federated neural collaborative filtering,” Knowledge-Based Systems, vol. 242, Art. no. 108441, 2022.

[8] Z. Sun, Y. Xu, Y. Liu, W. He, L. Kong, F. Wu, Y. Jiang, and L. Cui, “A survey on federated recommendation systems,” IEEE Transactions on Neural Networks and Learning Systems, vol. 36, no. 1, pp. 6–20, 2025.

[9] X. Yi, J. Yang, L. Hong, D. Z. Cheng, L. Heldt, A. Kumthekar, Z. Zhao, L. Wei, and E. H. Chi, "Sampling-bias-corrected neural modeling for large corpus item recommendations," in Proc. ACM Conf. Recommender Systems (RecSys), 2019, pp. 269–277.

[10] M. Abadi et al., "Deep learning with differential privacy," in Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS), 2016, pp. 308–318.

[11] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, and Y. Zhou, "A hybrid approach to privacy-preserving federated learning," in Proc. 12th ACM Workshop on Artificial Intelligence and Security (AISec), 2019, pp. 1–11.

[12] B. Hitaj, G. Ateniese, and F. Pérez-Cruz, "Deep models under the GAN: Information leakage from collaborative deep learning," in Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS), 2017, pp. 603–618.

[13] M. Daole, A. Schiavo, J. L. Corcuera Bárcena, P. Ducange, F. Marcelloni, and A. Renda, "OpenFL-XAI: Federated learning of explainable artificial intelligence models in Python," SoftwareX, vol. 23, Art. no. 101505, 2023.

[14] Y. Zhang and H. Yu, "LR-XFL: Logical reasoning-based explainable federated learning," arXiv:2308.12681, 2023.

[15] L. M. Lopez-Ramos, F. Leiser, A. Rastogi, S. Hicks, I. Strümke, V. I. Madai, T. Budig, A. Sunyaev, and A. Hilbert, "Interplay between federated learning and explainable artificial intelligence: A scoping review," arXiv:2411.05874, 2024.

[16] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," ACM Computing Surveys, vol. 51, no. 5, Art. no. 93, pp. 1–42, 2018.

[17] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?' Explaining the predictions of any classifier," in Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD), 2016, pp. 1135–1144.

[18] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems, vol. 30, 2017.

[19] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in Proc. Int. Conf. Machine Learning (ICML), 2017, pp. 3319–3328.

[20] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," arXiv:1312.6034, 2013.

[21] M. Ancona, E. Ceolini, C. Öztireli, and M. Gross, "Towards better understanding of gradient-based attribution methods for deep neural networks," in Proc. Int. Conf. Learning Representations (ICLR), 2018.

[22] L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, no. 3, pp. 338–353, 1965.

[23] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-15, no. 1, pp. 116–132, 1985.

[24] M. Sugeno and G. T. Kang, "Structure identification of fuzzy model," Fuzzy Sets and Systems, vol. 28, no. 1, pp. 15–33, 1988.

[25] L.-X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," IEEE Transactions on Systems, Man, and Cybernetics, vol. 22, no. 6, pp. 1414–1427, 1992.

[26] F. Aghaeipoor, M. Sabokrou, and A. Fernández, "Fuzzy rule-based explainer systems for deep neural networks: From local explainability to global understanding," IEEE Transactions on Fuzzy Systems, vol. 31, no. 9, pp. 3069–3080, 2023.

[27] M. T. Ribeiro, S. Singh, and C. Guestrin, "Anchors: High-precision model-agnostic explanations," in Proc. AAAI Conf. Artificial Intelligence, vol. 32, no. 1, 2018, pp. 1527–1535.

[28] Y. Lou, R. Caruana, and J. Gehrke, "Intelligible models for classification and regression," in Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD), 2012, pp. 150–158.

[29] R. Caruana et al., "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission," in Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD), 2015, pp. 1721–1730.

[30] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in Proc. Int. Conf. World Wide Web (WWW), 2017, pp. 173–182.

[31] X. Wang, D. Wang, C. Xu, X. He, Y. Cao, and T.-S. Chua, "Explainable reasoning over knowledge graphs for recommendation," in Proc. AAAI Conf. Artificial Intelligence, vol. 33, no. 1, 2019, pp. 5329–5336.

[32] Y. Xian, Z. Fu, S. Muthukrishnan, G. de Melo, and Y. Zhang, "Reinforcement knowledge graph reasoning for explainable recommendation," in Proc. Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR), 2019, pp. 285–294.

[33] X. Chen, Y. Zhang, H. Xu, Y. Cao, Z. Qin, and H. Zha, "Visually explainable recommendation," arXiv:1801.10288, 2018.

[34] A. Gonçalves, P. Ray, B. Soper, J. Stevens, L. Coyle, and A. P. Sales, "Generation and evaluation of synthetic patient data," BMC Medical Research Methodology, vol. 20, Art. no. 108, 2020.

[35] B. van Breugel, Z. Qian, and M. van der Schaar, "Synthetic data, real errors: How (not) to publish and use synthetic data," in Proc. Int. Conf. Machine Learning (ICML), 2023, pp. 34793–34808.

[36] H. R. Roth et al., "NVIDIA FLARE: Federated learning from simulation to real-world," arXiv:2210.13291, 2022.

[37] NVIDIA, "NVIDIA FLARE documentation," NVIDIA FLARE 2.7.0 documentation. [Online]. Available: https://nvflare.readthedocs.io/en/main/. Accessed: May 11, 2026.