Marketing Science Institute Working Paper Series 2026

Report No. 26-100

Interpretable Recommendations and Parameter-Grounded Explanations with Multi-Graph Attention

Yan Leng, Xiao Liu and Rodrigo Ruiz

manuscript

Interpretable Recommendations and Parameter-Grounded Explanations with Multi-Graph Attention

Yan Leng
McCombs School of Business, The University of Texas at Austin, yan.leng@mccombs.utexas.edu

Xiao Liu
New York University (NYU) - Leonard N. Stern School of Business, xliu@stern.nyu.edu

Rodrigo Ruiz
rodrigoruiz@alum.mit.edu

Deep learning recommender systems (RSs) increasingly combine ratings, attributes, and network signals, but the resulting models are often opaque and hard to explain. We propose an explanation-integrated framework in which prediction and explanation rely on the same internal signals. We develop MG-GAT, a multi-graph attention model that fuses user–user and business–business networks with auxiliary attributes and ratings. MG-GAT exposes two interpretable prediction-time signals: a Neighbor Importance Graph (NIG) that identifies which neighbors drive each recommendation and Feature Relevance (FR) that indicates which attributes make those neighbors influential. We use NIG and FR as auditable inputs to produce parameter-grounded explanations, defined as narratives generated by a fixed mechanism under a fixed decoding configuration. We implement this idea via a constrained LLM-based plan-and-critique procedure and show in a hallucination audit that grounding generation in model-derived evidence reduces contextual misattribution. Across two Yelp regions, MG-GAT achieves predictive performance comparable to strong attention- and knowledge-graph-based baselines. In a randomized human-subject experiment, MG-GAT-guided explanations improve engagement outcomes and user-perceived explanation quality—including trust, persuasiveness, and satisfaction—relative to similarity-based, social, and post-hoc SHAP-based explanations. These results show how RSs can be engineered to deliver interpretable predictions and user-centered explanations that are grounded in the same signals used for prediction.

Key words: recommender systems; explainable AI; design science research; large language models; networks

1. Introduction

Recommender systems (RSs) are essential for improving user experiences and driving transactions (Zhang and Curley 2018, Bauman et al. 2017, Bauman and Tuzhilin 2018), and clear explanations that users can understand and trust are equally important (Herlocker et al. 2000, Tintarev and Masthoff 2010, 2012a, Nunes and Jannach 2017). At the same time, integrating multiple signals into deep learning (DL) recommenders often increases opacity rather than clarity (Zhou et al. 2023,

Kunkel et al. 2019, Bauer et al. 2023). In practice, post-hoc surrogates such as SHAP (SHapley Additive exPlanations) approximate local importance for black-box models (Lundberg and Lee 2017).[1] However, the interpretable machine learning (IML) literature cautions that such model-agnostic techniques can misrepresent the predictor's computation (Rudin 2019), with large-scale evidence of frequent direction errors in feature importance (Ragodos et al. 2024). These considerations motivate treating interpretability as a first-class design objective: the reasons behind a recommendation should be surfaced at prediction time and connected directly to the model's internal logic, rather than approximated after the fact.

Yet, integrating network data, especially within DL methods, introduces additional complexities. The heterogeneity and noise inherent in network connections can hinder the learning of meaningful relationships (Bapna et al. 2017): in online marketplaces, users may have various types of connections (e.g., followers or friends), but not all are equally relevant for recommendations. Theories like weak tie theory (Granovetter 1977) and social role theory (Biddle 1986) suggest that the relevance of connections varies. This motivates a model that can both learn which ties are predictive and expose those learned tie-weights in a form users can audit.

Our research addresses the dual challenges of achieving interpretability and automated explanations while integrating rich network information in RSs. We ground our two-stage framework in design principles focused on key explanation quality criteria, including *transparency*, *trust*, *effectiveness*, *efficiency*, *persuasiveness*, and *satisfaction* (Tintarev and Masthoff 2012a,b). Guided by these principles, our approach delivers interpretable recommendations and user-centered explanations.

In the first stage, we design an interpretable DL framework, the Multi-Graph Attention Network (MG-GAT), which integrates network information, user and business attributes, and user–business ratings to predict missing entries in the rating matrix. To support model-traceable explanations, MG-GAT exposes two prediction-time signals: the Neighbor Importance Graph (NIG), the neighbor weights used by the model's own aggregation at prediction time, and Feature Relevance (FR), an additive decomposition that expresses each NIG in terms of contributions of observed attributes. Surfacing NIG/FR at prediction time supports transparency by making the ties and attributes driving each recommendation visible and auditable, and it supports trust by basing decisions (and later explanations) on the same model-internal signals rather than post-hoc surrogates.

In the second stage, we use MG-GAT's prediction-time evidence signals (NIG/FR) as control inputs to constrain LLM explanation generation. Rather than prompting an LLM on raw business metadata or full user history, we first use NIG to select a small, high-salience support set of neighbors and use FR to select a small set of salient observable attributes within that support. Concretely, for

[1] SHAP assigns each feature a marginal contribution to a model prediction based on Shapley values.

each recommendation, we identify the most influential user-side and business-side neighbors (per NIG) and the most influential observable attributes that account for their importance (per FR). We then constrain the generator to cite only this selected set of neighbors and attributes, rather than introducing unrelated facts or weakly relevant details. We implement this constraint via a brief plan-and-critique procedure: candidate explanation plans and narratives are generated and scored using NIG-weighted criteria aligned with our six explanation quality metrics.

Using Yelp data from a Canadian province and a U.S. state, we demonstrate the competitive performance of MG-GAT against existing recommendation algorithms, such as attention-based (Shimizu et al. 2022, Wu et al. 2019), heterogeneous information network-based (Fan et al. 2019, Zhang and Chen 2020, Cai et al. 2023), contrastive learning methods (Chen et al. 2023), and interpretable methods (Wang et al. 2019a, Pan et al. 2021). Through ablation studies, we identify factors contributing to this performance, showing that incorporating network and auxiliary data, as well as the model's core components enables competitive performance.

To assess the real-world efficacy of MG-GAT's explanations, we conducted a randomized user experiment that simulates preference-conditioned Yelp recommendations with five conditions. Relative to (i) the no-explanation control, (ii) an industry-style product-similarity rationale, (iii) a social-based explanation (Park et al. 2017), and (iv) a feature-based baseline generated via SHAP combined with an LLM, MG-GAT-guided explanations yield significantly higher acceptance/engagement intentions (Perceived Relevance and Future Interest) and higher user-rated explanation quality across the six criteria.[2] This pattern underscores the value of parameter-grounded explanations in driving a recommendation.

1.1. Salient Design Insights

Our framework is organized around three Salient Design Insights (SDIs) (Abbasi et al. 2024, Lee and Ram 2024, Lee et al. 2025) that operationalize the explanation goals required for high-quality explanations (Tintarev and Masthoff 2010), see Table 1. Effectiveness (i.e., helping users choose items that truly match their needs) is the non-negotiable baseline for any RS; each SDI then targets a complementary subset of user-centric desiderata so that interpretability is woven into the recommendation process rather than added after the fact.

- **SDI-1 (Trust & Transparency): Prediction-time rationale**. We expose the drivers of prediction through NIG (which neighbors matter) and FR (which attributes make them matter), rather than relying on post-hoc surrogates. This design aligns with process-transparency principles in IS and strengthens user trust (Bauer et al. 2023, Herlocker et al. 2000).

[2] Because the experimental conditions differ in both the *evidence interface* (what information is surfaced/selected) and, for some baselines, the *surface realization* mechanism, we interpret the treatment effects as comparisons of end-to-end explanation strategies rather than as isolating a single component in isolation.

- **SDI-2 (Efficiency): Dual-channel filtering to manage cognitive load**. While parameter exposure aids transparency, unstructured presentation can overwhelm users. We therefore provide two continuous salience signals from NIG and FR that compress high-dimensional network and feature information into a small set of cues per recommendation. This enables users to see quickly which neighbors and which attributes matter most and thereby improves interpretive efficiency (Gedikli et al. 2014, Hollender et al. 2010, Herm 2023).
- **SDI-3 (Persuasiveness & Satisfaction): Parameter-grounded generation.** We translate the NIG/FR-selected support into natural-language rationales under an evidence contract: narratives may cite only the selected neighbors/attributes, enforced via our constrained generation procedure.

These three SDIs motivate our artifact: Stage 1 produces auditable prediction-time evidence (NIG/FR), and Stage 2 translates that evidence into support-limited natural-language rationales.

1.2. Contributions.

Our study advances design science research (DSR) on explainable recommender systems (ERS) and human-centered AI by developing and evaluating an explanation-integrated artifact in which both prediction and explanation are grounded in a shared, auditable evidential base.

(i) Design principle. We formalize parameter-grounded explanations for an attention-based RS as support-limited rationales generated by a fixed mechanism, whose inputs are the model's prediction-time evidence signals and observable attributes (Def. 3). This principle is explicitly scoped as a traceability contract: it ensures that user-facing narratives are generated from the same evidence interface used by the deployed predictor rather than from a separate post-hoc surrogate.

(ii) Reusable design knowledge. We distill three SDIs and a decision template that guide (a) which prediction-time evidence signals to surface, (b) how to compress them into user-digestible cues (salience filtering), and (c) how to translate them into auditable narratives without overwhelming users. This reusable guidance supports both ERS design and LLM-enabled decision support in settings where explanation faithfulness and auditability are central.

(iii) Artifact. We instantiate this principle in a multi-graph attention recommender that fuses user–user and business–business relations with ratings and attributes while exposing two auditable prediction-time signals: (a) an NIG capturing tie-level salience (which neighbors drive the prediction) and (b) an FR decomposition that expresses each pre-softmax attention logit as additive contributions of observable features (Defs. 1–2). We further provide formal characterizations that make these signals inspectable and usable for explanation (Props. 1 and 2).

(iv) Human experiments. We evaluate interpretability and explanations from the end-user perspective via a randomized, preference-conditioned vignette experiment with five conditions (including a SHAP+LLM feature-attribution baseline). Beyond offline metrics and ablations, we show

that MG-GAT-guided explanations improve engagement intentions and user-perceived explanation quality relative to strong baselines.

Table 1 Overview of the two–stage MG-GAT and LLM-guided explanation framework.

Dimension	Description
Input	**Recommendation:** user–user and business–business graphs, auxiliary user/business attributes, and observed ratings. **Explanation:** prediction-time signals NIG (which neighbors matter) and FR (which attributes make them matter); the Stage-2 search is constrained to these signals.
Output	**Recommendation:** predicted (completed) user–business ratings. **Explanation:** parameter–grounded natural-language rationales constrained to NIG/FR (support-limited to prediction-time NIG/FR signals; model-traceable) and selected via NIG-weighted critics.
Key desiderata and SDIs	• **SDI 1 [Trust & Transparency]** – transparent logic as auditable evidence: expose NIG and FR instead of relying on post-hoc surrogates (e.g., SHAP). • **SDI 2 [Efficiency]** – dual-channel filtering: two salience signals from NIG and FR compress high-dimensional network and feature information into a small set of cues per recommendation. • **SDI 3 [Persuasiveness & Satisfaction]** – parameter-grounded generation: a plan–and–critique search is constrained to NIG/FR and scored by NIG-weighted critics to impose support-based grounding constraints and enable auditing.

2. Literature Review

We build on two strands: prediction-focused RSs and explainable RSs. Below we summarize each, emphasizing explanation gaps and how our work is positioned relative to them.

2.1. Prediction-focused and Deep Learning-based Recommender Systems (RSs)

Prediction-focused RSs increasingly combine ratings with auxiliary text, attributes, and network/context signals, often via DL methods on graphs and heterogeneous information networks (Bobadilla et al. 2013, Zhou et al. 2023, Li et al. 2017, Adomavicius and Tuzhilin 2005, Sharma et al. 2024, Bauman and Tuzhilin 2022, 2018, Bauman et al. 2017, 2024). While these advances mitigate sparsity and lift accuracy, they remain largely prediction-first, offering limited, user-auditable rationales (Wang et al. 2021, Dziugaite and Roy 2015, Zhang and Curley 2018).

Graph-based RSs exemplify this gap. Learned embeddings and message passing are effective but hard to audit: weights on hidden representations do not reveal why a neighbor is predictive or which attributes drive that salience (Wang et al. 2019b, He et al. 2020, Hartford et al. 2018, Zhang and Chen 2020). Attention can localize signal (Shimizu et al. 2022, Fan et al. 2019, Gao et al. 2019, Wu et al. 2019), but it is usually over latent states; as debated in the "attention as explanation" literature, raw attention weights often do not provide user-scrutinizable reasons in feature space (Jain and Wallace 2019, Serrano and Smith 2019, Wiegreffe and Pinter 2019). Recent contrastive objectives further boost representations without making this linkage explicit (Cai et al. 2023, Chen et al. 2023).

MG-GAT is situated in this stream as an attention-based RS that is designed to expose prediction-time evidence signals in user-auditable form. Appendix A.4 (Table A4) summarizes how attention- and KG-based recommenders differ in (i) graph structure, (ii) what is exposed as an explanation signal, (iii) whether attention admits an explicit feature-level decomposition, and (iv) whether any user-facing text is parameter-grounded.

Graph Attention Network (GAT) (Veličković et al. 2018) operates on a single homogeneous graph and exposes attention over latent states; DualGAT and related social models (Wu et al. 2019, Fan et al. 2019) add a social graph but still provide only latent weights; KGAT (Wang et al. 2019a) learns attention over knowledge-graph relations and can surface relation/path evidence, but it does not provide an explicit decomposition of prediction-time salience into *observable feature-level* contributions; similarly, path-based methods (Xian et al. 2020, Zhu et al. 2021) surface KG paths but do not provide an explicit feature-space contribution decomposition aligned with prediction-time attention; review-attention models (Wang et al. 2018a, Zhang et al. 2014) highlight words without linking those signals directly to the parameters used for prediction.

Relative to prior attention- and KG-based RSs, MG-GAT (i) attends jointly over multiple types of relationships and exposes the resulting neighbor weights as a single NIG, (ii) provides an FR decomposition that expresses each pre-softmax attention logit as additive contributions of observable attributes, and (iii) reuses NIG/FR as explicit control inputs for Stage 2 so generated explanations are traceable to prediction-time evidence rather than to a post-hoc surrogate.

From the perspective of the attention literature, our framework can be viewed as a step toward making GAT architectures more interpretable and explainable: the FR decomposition and convex-combination propositions make explicit two properties that are often implicit in GAT-style models—attention logits in a linear first layer can be decomposed over observable features, and attention weights define barycentric coordinates over neighbor embeddings. We then leverage these properties to construct NIG/FR as structured explanation signals and extend them to the explanation layer by defining and analyzing parameter-grounded explanations for attention-based RSs.

2.2. Explainable Recommender Systems (ERS)

Explainability influences user trust, acceptance, and decision efficiency (Wang and Benbasat 2007, Zhang and Curley 2018, Kunkel et al. 2019), motivating ERS and the broader IML literature (Guidotti et al. 2018). IML methods are often grouped into post-hoc surrogates and models that expose model-mechanistic (architecturally constrained) signals. Post-hoc procedures such as SHAP and LIME approximate local importance but can suffer fidelity issues and non-trivial cost, with large-scale evidence of misattributed directions in deep settings (Rudin 2019, Zhu et al. 2021, Ragodos et al. 2024). By contrast, architecturally constrained approaches expose prediction-time signals from the deployed model itself; our work contributes to this stream by surfacing NIG/FR at prediction time and reusing them to generate user-facing rationales.

2.2.1. Non-LLM-based ERS. The first relevant stream of ERS work largely focuses on feature-based, social-based, KG-based, and textual explanations. Feature-based approaches highlight item or user attributes that are important for prediction (Wang et al. 2019a, Bauman et al. 2017, McInerney et al. 2018); for example, Bauman et al. (2017) identify the most valuable aspects of an item from user reviews. Social explanations assume that items are popular among a user's friends or similar users (Park et al. 2017). KG-based methods trace reasoning paths through entities and relations to justify recommendations (Huang et al. 2019, Xian et al. 2020, Shimizu et al. 2022, Zhu et al. 2021), while textual approaches use reviews or generated text to provide justifications, often with attention over words or documents (Chen et al. 2018a, Li et al. 2021).

These approaches offer useful signals but also limitations. KG paths and multi-hop reasoning can be rich yet cognitively demanding, and path-shortening strategies may trade off interpretability and predictive performance (Shimizu et al. 2022, Pan et al. 2021). Textual methods depend on the availability of high-quality reviews, and even when they highlight salient snippets, the underlying DL models typically remain opaque, so the explanations may not accurately reflect how predictions are computed. Shimizu et al. (2022), for example, visualizes attention coefficients over KG embeddings but does not design these coefficients as signals for explanation generation.

Our work differs in that the explanation signals (NIG/FR) are built into the predictor as prediction-time objects with explicit geometric and feature-level interpretations. Rather than deriving explanations from paths or text alone, we expose model-internal salience over neighbors and attributes and then reuse these signals as inputs to the explanation mechanism.

2.2.2. LLM-based Recommendations. A recent stream of RSs combines existing ERS with LLMs (Chen et al. 2024, Wu et al. 2024). Some methods use LLMs directly for prediction, via one-shot and few-shot prompting adapted to RSs (Geng et al. 2023, Gao et al. 2023, Wang et al. 2024); others fine-tune LLMs to better exploit corpus knowledge and reasoning abilities (Li et al. 2023a,b,c). However, fine-tuned LLMs often act as black boxes, and updating billions of parameters is computationally expensive and sometimes infeasible with proprietary models.

A closely related line uses LLMs as explainers for black-box RSs. For example, RecExplainer (Lei et al. 2024) prompts an LLM as a surrogate model to explain RS behavior, and DRE (Gao et al. 2024) uses LLMs to reason about connections between user histories and recommended items. These methods are model-agnostic but provide post-hoc explanations that may not faithfully capture the internal prediction process (Zhu et al. 2021, Ragodos et al. 2024). In contrast, we rely on our interpretable model for prediction and use its prediction-time evidence signals within a constrained multi-step prompting structure, so explanations are explicitly parameter-grounded and aligned with the deployed predictor. This modular design reduces computational overhead relative to LLM fine-tuning while maintaining competitive performance and lowering hallucinations in the explanations.

3. Interpretable RS: Multi-Graph Attention Network

This section presents Stage-1 of our two-stage artifact, the Multi-Graph Attention Network (MG-GAT). MG-GAT predicts missing ratings by combining (i) a global homophily prior (graph Laplacian regularization) with (ii) local, context-specific tie weighting (attention). Crucially, it exposes two prediction-time evidence signals used directly in Stage-2 explanation generation: NIG, the neighbor weights used in aggregation, and FR, a feature-level decomposition of the pre-softmax attention logit. These signals are model-internal objects (not post-hoc surrogates) and form the auditable "evidence contract" for explanations.

3.1. Design Rationale of MG-GAT

Our design is guided by two complementary theoretical premises. First, homophily implies that connected users (or businesses) tend to be similar in expectation (McPherson et al. 2001), so network structure provides a useful global prior when individual rating histories are sparse. Second, ties on digital platforms are often heterogeneous and weakly typed: an observed edge indicates that a relationship exists, but not whether it is preference-relevant for the focal decision. Weak-tie theory and role perspectives emphasize that ties differ in strength and function (Granovetter 1977, Biddle 1986), motivating local, decision-specific reweighting of neighbors.

MG-GAT operationalizes this separation explicitly. Laplacian regularization encodes the global smoothness (homophily) prior, while attention learns local tie salience and exposes it as a prediction-time object (NIG), making the model's network reasoning auditable rather than implicit.

Finally, to ensure that "which neighbors matter" can be explained in terms users can scrutinize, we anchor tie salience to observable attributes rather than only to latent states. This yields FR, a feature-level decomposition of the attention logit that explains why a particular tie is influential and enables dual-channel salience filtering: a small set of influential neighbors and attributes can be surfaced to users and passed to Stage 2. These design choices directly support SDI-1/SDI-2 by producing auditable, prediction-time evidence signals (NIG/FR) that can be summarized and reused to constrain explanation generation (Section 4).

3.2. Problem Formulation

Fig. 1 illustrates the architecture, and Table B1 in Appendix B summarizes notation. We observe a partially observed rating matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$, auxiliary user and business attributes $\mathbf{S}_u \in \mathbb{R}^{n \times s_u}$ and $\mathbf{S}_b \in \mathbb{R}^{m \times s_b}$, and user and business graphs $\mathbf{G}_u$ and $\mathbf{G}_b$ with Laplacians $\mathbf{L}_u$ and $\mathbf{L}_b$.

Our objective is to infer missing entries in $\mathbf{X}$ given $\mathbf{G}_u$, $\mathbf{G}_b$, $\mathbf{S}_u$, and $\mathbf{S}_b$. We cast this as matrix completion with graph regularization:

$$\mathcal{L} = \left\| \mathbf{\Omega}_{\text{training}} \circ (\mathbf{X} - \mathbf{U}^\top \mathbf{B}) \right\|_F^2 + \mathcal{R}(\mathbf{U}) + \mathcal{R}(\mathbf{B}), \tag{1}$$

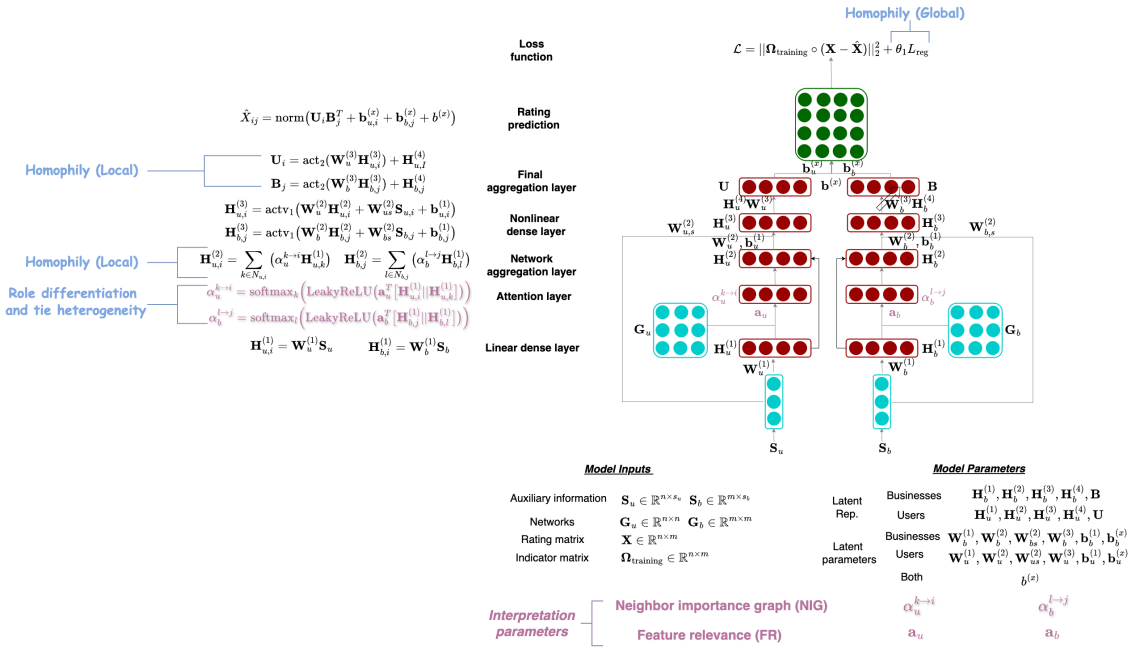


Figure 1 Overview of the Multi-Graph Attention Network (MG-GAT) used for the recommendation task.

where $\mathbf{U} \in \mathbb{R}^{n \times k_f}$ and $\mathbf{B} \in \mathbb{R}^{m \times k_f}$ are latent user and business representations; $\mathbf{\Omega}_{\text{training}}$ is an indicator matrix with ones at observed training entries and zeros otherwise; $\circ$ denotes the Hadamard product; and $\|\cdot\|_F$ is the Frobenius norm. The regularizers $\mathcal{R}(\mathbf{U})$ and $\mathcal{R}(\mathbf{B})$ impose graph-structured smoothness constraints described next.

In our context, we interpret $\mathbf{U}$ as a latent representation that captures users' preferences for businesses, and we interpret $\mathbf{B}$ as a business latent representation. We set $\mathcal{R}(\mathbf{U}) = \text{Tr}(\mathbf{U}^T \mathbf{L}_u \mathbf{U})$ and $\mathcal{R}(\mathbf{B}) = \text{Tr}(\mathbf{B}^T \mathbf{L}_b \mathbf{B})$, encouraging globally smooth embeddings under homophily (Cai et al. 2010, Li and Yeung 2009, McPherson et al. 2001, Leng et al. 2020).

3.3. Multi-Graph Attention Network

The Laplacian regularizer imposes global smoothness, but observed networks often contain heterogeneous and noisy ties (Section 3.1). We therefore introduce an interpretable graph attention network (GAT) (Veličković et al. 2018) that learns per-edge weights, emphasizing informative neighbors and downweighting irrelevant ones. This local weighting both improves prediction and exposes which neighbors drive each recommendation. (We provide a component-wise comparison to attention- and KG-based recommenders in Appendix A.4, Table A4.)

Next, we explain MG-GAT in detail. We first impose a linear transformation (a linear dense layer) on the auxiliary information to maintain the interpretability of these features:

$$\begin{aligned} \text{user } i\text{: } \mathbf{H}_{u,i}^{(1)} &= \mathbf{W}_u^{(1)} \mathbf{S}_{u,i}, \\ \text{business } j\text{: } \mathbf{H}_{b,j}^{(1)} &= \mathbf{W}_b^{(1)} \mathbf{S}_{b,j}, \end{aligned} \tag{2}$$

where $\mathbf{W}_u^{(1)} \in \mathbb{R}^{d_u^{(0)} \times s_u}$ and $\mathbf{W}_b^{(1)} \in \mathbb{R}^{d_b^{(0)} \times s_b}$ are coefficient matrices applied to user i's attributes $\mathbf{S}_{u,i}$ and business j's attributes $\mathbf{S}_{b,j}$; s_u and s_b are the dimensions of the auxiliary information for users and businesses; and $d_u^{(0)}$ and $d_b^{(0)}$ are the dimensions of the node embeddings for the user and business after the first transformation.

Next, we introduce the graph attention mechanism and two key definitions as the inputs and outputs of this mechanism to make the architecture interpretable.

Definition 1 (Neighbor Importance Graph (NIG)). Let $G = (V, E)$ be a (possibly undirected) graph. For message passing, we view each observed edge as inducing directed edges $(k \to i)$ and define the in-neighborhood of i as $N_i = \{k : (k \to i) \in E\}$. The *Neighbor Importance Graph* is the collection of nonnegative weights $\{\alpha^{k\to i}\}_{(k\to i)\in E}$ such that for every node i, $\alpha^{k\to i} \in [0, 1]$ and

$$\sum_{k \in N_i} \alpha^{k\to i} = 1.$$

We denote the incoming weight vector by $\alpha^{\cdot\to i} \in \mathbb{R}^{|N_i|}$ and interpret $\alpha^{k\to i}$ as the *importance* of neighbor k when aggregating information to form i's representation. □

Interpretation. NIG answers: *which specific neighbors drove this recommendation, and by how much.* In MG-GAT, the weights $\alpha^{k\to i}$ are *prediction-time* attention coefficients used by the deployed model to construct node embeddings (Eq. (6)), rather than post-hoc attributions. Because platform ties are weakly typed and heterogeneous—some links are preference-relevant while others are noisy or incidental—NIG captures local heterogeneity by upweighting predictive neighbors and downweighting less-informative ties (Granovetter 1977, Biddle 1986, Crandall et al. 2010). The Laplacian regularizer provides a global homophily prior, while NIG refines it locally in a task- and context-specific way.

Instantiation via attention. Given a binary graph ($\mathbf{G}_u$ or $\mathbf{G}_b$) and intermediate node embeddings $\mathbf{H}^{(1)}$, the graph-attention layer computes an unnormalized score $e_{k\to i}$ for each directed edge and applies a neighbor-wise softmax to obtain $\alpha^{k\to i} = \mathrm{softmax}_i(e_{k\to i})$. This construction satisfies Definition 1 by design and yields two NIGs: $\{\alpha_u^{k\to i}\}$ on $\mathbf{G}_u$ and $\{\alpha_b^{l\to j}\}$ on $\mathbf{G}_b$.

We parameterize per-edge salience as a function of focal and neighbor attributes. For each directed edge $(k \to i)$, we first compute a *linear pre-activation score*

$$\tilde{e}^u_{k\to i} = \mathbf{a}_u^\top \left[\mathbf{H}^{(1)}_{u,i} \,\|\, \mathbf{H}^{(1)}_{u,k}\right], \qquad \tilde{e}^b_{l\to j} = \sum_{g\in\mathcal{G}_b} \omega_g \, \mathbf{a}_b^\top \left[\mathbf{H}^{(1)}_{b,j} \,\|\, \mathbf{H}^{(1)}_{b,l}\right], \tag{3}$$

then apply an element-wise nonlinearity before softmax:

$$e^u_{k\to i} = \mathrm{LeakyReLU}(\tilde{e}^u_{k\to i}), \qquad e^b_{l\to j} = \mathrm{LeakyReLU}(\tilde{e}^b_{l\to j}). \tag{4}$$

We normalize them by a node-wise softmax:

$$\alpha_u^{k\to i} = \frac{\exp(e^u_{k\to i})}{\sum_{k'\in N_i^u} \exp(e^u_{k'\to i})}, \qquad \alpha_b^{l\to j} = \frac{\exp(e^b_{l\to j})}{\sum_{l'\in N_j^b} \exp(e^b_{l'\to j})}. \tag{5}$$

By construction, $\alpha^{k\to i} \geq 0$ and $\sum_{k\in N_i} \alpha^{k\to i} = 1$, satisfying Definition 1.

We next introduce Feature Relevance, the second key concept for interpretability.

DEFINITION 2 (FEATURE RELEVANCE (FR)). Feature Relevance (FR) is the feature-space coefficient vector that decomposes the *linear pre-activation attention score* $\tilde{e}$ into additive contributions from the focal node's attributes and the neighbor's attributes. With the linear first layer in Eq. (2) and the score in Eq. (3), the user-side score admits the decomposition

$$\tilde{e}^{u}_{k\to i} = \mathrm{FR}^{u}_{\mathrm{self}}{}^{\top}\mathbf{S}_{u,i} + \mathrm{FR}^{u}_{\mathrm{nb}}{}^{\top}\mathbf{S}_{u,k},$$

where $\mathrm{FR}^{u}_{\mathrm{self}} = \mathbf{a}^{\top}_{u,\mathrm{self}}\mathbf{W}^{(1)}_{u}$ and $\mathrm{FR}^{u}_{\mathrm{nb}} = \mathbf{a}^{\top}_{u,\mathrm{nb}}\mathbf{W}^{(1)}_{u}$. Similarly, the business-side score admits

$$\tilde{e}^{b}_{l\to j} = \mathrm{FR}^{b}_{\mathrm{self}}{}^{\top}\mathbf{S}_{b,j} + \mathrm{FR}^{b}_{\mathrm{nb}}{}^{\top}\mathbf{S}_{b,l},$$

where $\mathrm{FR}^{b}_{\mathrm{self}} = \mathbf{a}^{\top}_{b,\mathrm{self}}\mathbf{W}^{(1)}_{b}$ and $\mathrm{FR}^{b}_{\mathrm{nb}} = \left(\sum_{g\in\mathcal{G}_b}\omega_g\right)\mathbf{a}^{\top}_{b,\mathrm{nb}}\mathbf{W}^{(1)}_{b}$. □

We design Eq. (2) to be linear by design so that the contribution of auxiliary attributes to neighbor importance can be learned directly.

Importantly, FR is a global coefficient object learned in Stage 1, while feature salience for a specific edge instance is obtained by combining FR with observed attributes (e.g., elementwise coefficient × feature value in the decomposition of $\tilde{e}$). This distinction ensures that Stage 2 uses FR to rank instance-specific attribute contributions without treating FR itself as an instance-level importance score. The attention weights α are obtained by applying LeakyReLU$(\cdot)$ and softmax to $\tilde{e}$ (Eqs. (4)–(5)); FR therefore explains how observable attributes shift the underlying attention score that drives NIG.

Proposition 1 formalizes this linkage: with a linear first layer, the pre-activation attention score admits an explicit feature-space decomposition with FR as coefficients.

PROPOSITION 1 **(Feature-level decomposition of attention score)**. *Let* $\mathbf{S}_{u,i}$ *and* $\mathbf{S}_{u,k}$ *denote the auxiliary attribute vectors for a focal user node* i *and a user-neighbor* k*, and let* $\tilde{e}^{u}_{k\to i}$ *be the user-side linear pre-activation score in Eq.* (3)*. With the linear embedding in Eq.* (2) *and the Feature Relevance vectors* FR^{u}_{self} *and* FR^{u}_{nb} *from Definition 2, the score can be written as*

$$\tilde{e}^{u}_{k\to i} = \mathrm{FR}^{u}_{self}{}^{\top}\mathbf{S}_{u,i} + \mathrm{FR}^{u}_{nb}{}^{\top}\mathbf{S}_{u,k}.$$

In particular, for any two user-neighbors k *and* k' *of* i*,*

$$\tilde{e}^{u}_{k\to i} - \tilde{e}^{u}_{k'\to i} = \mathrm{FR}^{u}_{nb}{}^{\top}(\mathbf{S}_{u,k} - \mathbf{S}_{u,k'}),$$

so the difference in attention score between k *and* k' *is a linear function of the difference in their attributes, with coefficients given by* FR^{u}_{nb}*. An analogous statement holds for the business-side score* $\tilde{e}^{b}_{l\to j}$ *with* FR^{b}_{self} *and* FR^{b}_{nb}*.*

This result makes attention auditable: positive (negative) entries in $\mathrm{FR}^{u}_{\mathrm{nb}}$ increase (decrease) a user-neighbor's score, holding other factors fixed (and analogously for $\mathrm{FR}^{b}_{\mathrm{nb}}$ on the business side).

We next aggregate neighbors' embedding for the focal node, which is a mapping from neighbor importance ($\{\alpha_u^{k\to i}\}$ and $\{\alpha_b^{l\to j}\}$) to node embedding:

$$\begin{aligned}\text{Aggregation for user } i\text{: } e(k\to i, k\in N_{u,i}) \to v(i) : \mathbf{H}_{u,i}^{(2)} &= \sum_{k\in N_{u,i}} \left(\alpha_u^{k\to i}\mathbf{H}_{u,k}^{(1)}\right),\\ \text{Aggregation for business } j\text{: } e(l\to j, l\in N_{b,j}) \to v(j) : \mathbf{H}_{b,j}^{(2)} &= \sum_{l\in N_{b,j}} \left(\alpha_b^{l\to j}\mathbf{H}_{b,l}^{(1)}\right),\end{aligned} \tag{6}$$

where $N_{u,i}$ and $N_{b,j}$ are the neighbor sets for nodes i and j on the user and business graphs, respectively. Nodes (k or l) with high neighbor importance ($\alpha_u^{k\to i}$ or $\alpha_b^{l\to j}$) contribute more to the embedding of the focal node (i or j).

Because the α's are non-negative and sum to one, Eq. (6) forms a convex combination of neighbors: the Laplacian regularization term encodes global homophily, while the attention weights (Eq. (5)) refine similarity locally by concentrating mass on ties whose roles/attributes are most informative for the current decision. Proposition 2 states this property explicitly.

PROPOSITION 2 **(Convex combination interpretation of NIG)**. *For each user or business node i, the MG-GAT aggregation in Eq. 6 represents the node embedding as a convex combination of its neighbors' embeddings, with weights given by the NIG. In particular, the embedding $H_i^{(2)}$ lies in the convex hull of $\{H_k^{(1)} : k \in \mathcal{N}_i\}$.*

Interpreting NIG as barycentric coordinates clarifies how each node representation is synthesized from its neighbors and how changes in neighbors or weights move the embedding within their convex hull.

Next, we feed $\mathbf{H}_{u,i}^{(2)}$, $\mathbf{H}_{b,j}^{(2)}$, $\mathbf{S}_{u,i}$, and $\mathbf{S}_{b,j}$ into separate dense layers:

$$\begin{aligned}\text{user } i\text{: } v\to v : \mathbf{H}_{u,i}^{(3)} &= \text{actv}_1\left(\mathbf{W}_u^{(2)}\mathbf{H}_{u,i}^{(2)} + \mathbf{W}_{us}^{(2)}\mathbf{S}_{u,i} + \mathbf{b}_{u,i}^{(1)}\right),\\ \text{business } j\text{: } v\to v : \mathbf{H}_{b,j}^{(3)} &= \text{actv}_1\left(\mathbf{W}_b^{(2)}\mathbf{H}_{b,j}^{(2)} + \mathbf{W}_{bs}^{(2)}\mathbf{S}_{b,j} + \mathbf{b}_{b,j}^{(1)}\right),\end{aligned} \tag{7}$$

where $\mathbf{W}_u^{(2)} \in \mathbb{R}^{d_u^{(1)}\times d_u^0}$, $\mathbf{W}_b^{(2)} \in \mathbb{R}^{d_b^{(1)}\times d_b^0}$, $\mathbf{b}_u^{(1)} \in \mathbb{R}^{d_u^{(1)}}$, and $\mathbf{b}_b^{(1)} \in \mathbb{R}^{d_b^{(1)}}$, and where $\mathbf{W}_{us}^{(2)} \in \mathbb{R}^{d_u^{(1)}\times s_u}$ and $\mathbf{W}_{bs}^{(2)} \in \mathbb{R}^{d_b^{(1)}\times s_b}$ are the learnable weights; where $d_u^{(1)}$ and $d_b^{(1)}$ are the sizes of the latent dimension; and where $\text{actv}_1(\cdot)$ is a nonlinear activation function (chosen as a hyperparameter).

To obtain the final embedding of user ($\mathbf{U}$) and business ($\mathbf{B}$), we perform the following:

$$\begin{aligned}\text{user } i\text{'s final embedding: } \mathbf{U}_i &= \text{actv}_2(\mathbf{W}_u^{(3)}\mathbf{H}_{u,i}^{(3)}) + \mathbf{H}_{u,i}^{(4)},\\ \text{business } j\text{'s final embedding: } \mathbf{B}_j &= \text{actv}_2(\mathbf{W}_b^{(3)}\mathbf{H}_{b,j}^{(3)}) + \mathbf{H}_{b,j}^{(4)},\end{aligned} \tag{8}$$

where $\mathbf{W}_u^{(3)} \in \mathbb{R}^{d_u^{(1)}\times k_f}$ and $\mathbf{W}_b^{(3)} \in \mathbb{R}^{d_b^{(1)}\times k_f}$ are the learnable weights to transform the intermediate latent embedding to the dimension of the final embedding (k_f); where $\mathbf{H}_{u,i}^{(4)} \in \mathbb{R}^{k_f}$ and $\mathbf{H}_{b,j}^{(4)} \in \mathbb{R}^{k_f}$ are learnable weights; and where $\text{actv}_2(\cdot)$ is a nonlinear activation function. Learning $\mathbf{H}_{u,i}^{(4)}$ and $\mathbf{H}_{b,j}^{(4)}$ bears some resemblance to the traditional graph-regularized matrix factorization method that adopts global smoothness regularization. The embedding is learned directly from the rating matrix using a global graph regularization term. Combining the two embeddings adds further expressive power to the final embeddings.

Finally, the prediction of user i's rating on business j can be formalized as

$$\hat{X}_{ij} = \text{norm}\big(\mathbf{U}_i\mathbf{B}_j^T + \mathbf{b}_{u,i}^{(x)} + \mathbf{b}_{b,j}^{(x)} + b^{(x)}\big), \tag{9}$$

where $\text{norm}(x) = (r_{\max} - r_{\min}) \cdot \text{sigmoid}(x) + r_{\min}$; and $(\cdot)^T$ is the transpose of the matrix. This step normalizes the rating into the correct range, with $r_{\max}$ and $r_{\min}$ being the maximum and minimum. The user-specific, business-specific, and global bias terms are denoted as $\{\mathbf{b}_u^{(x)}, \mathbf{b}_b^{(x)}, b^{(x)}\} \in \{\mathbb{R}^n, \mathbb{R}^m, \mathbb{R}\}$, respectively, which are standard terms in the RS literature.

Scope of interpretability. We adopt a model-mechanistic notion of interpretability: MG-GAT exposes the prediction-time evidence it uses for each recommendation—NIG, the neighbor weights in its own aggregation, and FR, a feature-level decomposition of the corresponding attention logits over observable attributes. We treat NIG/FR as auditable evidence signals for understanding and communicating the model's computation, rather than as universal or causal explanations.[3]

3.4. Model Training

We train by minimizing mean squared error with graph Laplacian regularization:

$$\mathcal{L} = ||\mathbf{\Omega}_{\text{training}} \circ (\mathbf{X} - \hat{\mathbf{X}})||_F^2 + \theta_1 L_{\text{reg}}, \tag{10}$$

where θ_1 controls the strength of graph regularization with $L_{\text{reg}} = \text{Tr}(\mathbf{H}_u^{(4)T}\tilde{\mathbf{L}}_u\mathbf{H}_u^{(4)}) + \text{Tr}(\mathbf{H}_b^{(4)T}\tilde{\mathbf{L}}_b\mathbf{H}_b^{(4)})$ and $\tilde{\mathbf{L}}_u = \mathbf{L}_u + \theta_2\mathbf{I}$, $\tilde{\mathbf{L}}_b = \mathbf{L}_b + \theta_2\mathbf{I}$. We omit Laplacian regularization on $\mathbf{H}_u^{(3)}$ and $\mathbf{H}_b^{(3)}$ because Eq. (6) provides local smoothing. We apply standard ℓ_2 regularization and optimize with Adam, tuning hyperparameters via Hyperopt.

Algorithm 1 summarizes the training-time update steps for $\mathbf{U}$, $\mathbf{B}$, and $\hat{\mathbf{X}}$ under Eq. (10). We keep MG-GAT shallow so its prediction-time evidence signals (NIG, FR) remain exposed and auditable. NIG identifies which neighbors matter; FR identifies which attributes make them matter. Together they define the evidence interface passed to Stage 2. These signals also act as continuous filters over ties and attributes (SDI-2: efficiency) and serve as control inputs for the Stage 2 constrained LLM generator (Section 4), so that explanations directly reflect the deployed model without a surrogate explainer.

4. Parameter-Grounded Explanation Generation with LLMs

In Stage 2 of our end-to-end RS (Fig. 2), we translate MG-GAT's prediction-time signals into user-facing natural-language rationales. We generate parameter-grounded explanations that depend only on MG-GAT's internal signals (NIG, FR), the recommended item's attributes, and the attributes of

[3] This scope choice aligns with ongoing debates on "attention as explanation" (e.g., Jain and Wallace 2019, Serrano and Smith 2019, Wiegreffe and Pinter 2019); our claims are limited to what MG-GAT exposes architecturally and do not assert causal mechanisms in user behavior or outcomes.

[3] In Algorithm 1, $N_{u,i}$ refers to the neighboring set of individual i within the network $\mathbf{G}_u$, capturing the relationships for all individuals in that network. Similarly, $N_{b,j}$ refers to the neighboring set of businesses j within the network $\mathbf{G}_b$.

Algorithm 1 Multi-Graph Attention Network (MG-GAT).

input $\mathbf{X}$, $\mathbf{S}_u$, $\mathbf{S}_b$, $\mathbf{G}_u$, $\mathbf{G}_b$, k_f, θ_1, θ_2, V (max iteration)
for $v = 1 : V$ **do**
  Update user embedding
  for $i = 1 : n$ **do**
    $\mathbf{H}_{u,i}^{(1)(v)} = \mathbf{W}_u^{(1)(v)} \mathbf{S}_{u,i}$
    for $k \in N_i^u$ **do**
      $\alpha_{k \to i}^{u(v)} = \text{softmax}_k\Big(\text{LeakyReLU}\big(\mathbf{a}_u^{(v)T}\big[\mathbf{H}_{u,i}^{(1)(v)} || \mathbf{H}_{u,k}^{(1)(v)}\big]\big)\Big)$
    $\mathbf{H}_{u,i}^{(2)(v)} = \sum_{k \in N_{u,i}} \big(\alpha_{k \to i}^{u(v)} \mathbf{H}_{u,k}^{(1)(v)}\big)$
    $\mathbf{H}_{u,i}^{(3)(v)} = \text{actv}_1\big(\mathbf{W}_u^{(2)(v)} \mathbf{H}_{u,i}^{(2)(v)} + \mathbf{W}_{us}^{(2)(v)} \mathbf{S}_{u,i} + \mathbf{b}_u^{(1)(v)}\big)$
    $\mathbf{U}_i^{(v)} = \text{actv}_2(\mathbf{W}_u^{(3)} \mathbf{H}_{u,i}^{(3)}) + \mathbf{H}_{u,i}^{(4)(v)}$
  Update business embedding
  for $j = 1 : m$ **do**
    $\mathbf{H}_{b,j}^{(1)(v)} = \mathbf{W}_b^{(1)(v)} \mathbf{S}_{b,j}$
    for $l \in N_j^b$ **do**
      $\alpha_{l \to j}^{b(v)} = \text{softmax}_l\Big(\text{LeakyReLU}\big(\mathbf{a}_b^{(v)T}\big[\mathbf{H}_{b,j}^{(1)(v)} || \mathbf{H}_{b,l}^{(1)(v)}\big]\big)\Big)$
    $\mathbf{H}_{b,j}^{(2)(v)} = \sum_{l \in N_{b,j}} \big(\alpha_{l \to j}^{b(v)} \mathbf{H}_{b,l}^{(1)(v)}\big)$
    $\mathbf{H}_{b,j}^{(3)(v)} = \text{actv}_1\big(\mathbf{W}_b^{(2)(v)} \mathbf{H}_{b,j}^{(2)(v)} + \mathbf{W}_{bs}^{(2)(v)} \mathbf{S}_{b,j} + \mathbf{b}_b^{(1)(v)}\big)$
    $\mathbf{B}_j^{(v)} = \text{actv}_2(\mathbf{W}_b^{(3)} \mathbf{H}_{b,j}^{(3)}) + \mathbf{H}_{b,j}^{(4)(v)}$
  $\hat{X}_{ij}^{(v)} = \text{norm}\big(\mathbf{U}_i^{(v)} \mathbf{B}_j^{(v)T} + \mathbf{b}_{u,i}^{(x)(v)} + \mathbf{b}_{b,j}^{(x)(v)} + b_x^{(v)}\big)$
output $\mathbf{U}^{(V)}, \mathbf{B}^{(V)}$, $\hat{\mathbf{X}}^{(V)} = \mathbf{U}^{(V)} \mathbf{B}^{(V)T}$

the selected neighbors, so each narrative is directly traceable to the deployed predictor rather than to a post-hoc surrogate.

Stage 2 in this section is structured as follows. First, we define parameter-grounded explanations as an evidence-contract property: the generator may only use MG-GAT's prediction-time evidence interface (NIG/FR) and the observed attributes of the recommended item and selected neighbors. Second, we operationalize this contract by instantiating a small neighbor support set (top-K by NIG) and a small attribute set (top-F by NIG-weighted, instance-level FR contributions). Third, we document a key failure mode of vanilla prompting in our setting (contextual misattribution). Finally, we present a constrained plan-and-critique procedure that separates evidence selection from surface realization and filters candidates using NIG-weighted critics, enabling auditing against the evidence contract.

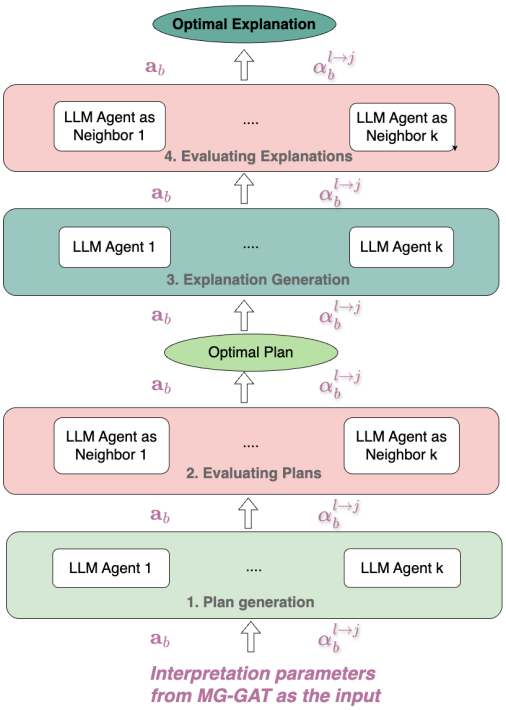


Figure 2 The MG-GAT-enabled explanation-generation process. It leverages the signals (FR and NIG) learned in Stage 1 to produce explanations.

Running example. Consider a focal recommendation (u, j) where the interface shows two previously liked restaurants and one recommended restaurant j (Fig. 2). Stage 1 yields prediction-time neighbor weights on the user and business graphs (NIG), from which we select a small support $S_\theta(u, j)$ by taking the top-K neighbors on each side. To determine which observable attributes make these neighbors matter, we convert the global FR coefficients into instance-level feature contributions (coefficient $\times$ observed feature value) and aggregate them using NIG weights to obtain a top-F feature set. Stage 2 then prompts the generator with only this support set and feature set (the evidence contract) and filters candidate plans/narratives so the final explanation cites only evidence that the model used at prediction time.

Definition 3 (Parameter-grounded explanation). Let θ denote the parameters of MG-GAT. For a user–business pair (u, j), let $S_\theta(u, j)$ denote the high-salience neighbor set used in prompting and let $X(S_\theta(u, j))$ collect their observed attributes. An explanation $E(u, j)$ is parameter-grounded if there exists a (possibly randomized) mechanism Φ such that

$$E(u,j) \sim \Phi\Big(\mathrm{NIG}^u_\theta(u,\cdot),\, \mathrm{NIG}^b_\theta(:,j),\, \mathrm{FR}^u_{\mathrm{self}},\, \mathrm{FR}^u_{\mathrm{nb}},\, \mathrm{FR}^b_{\mathrm{self}},\, \mathrm{FR}^b_{\mathrm{nb}},\, x_j,\, X(S_\theta(u,j))\Big),$$

i.e., the explanation is determined in distribution solely by MG-GAT's prediction-time NIG signals, the global FR coefficient vectors learned in Stage 1, and the observed attributes of the recommended item and the selected neighbors, given a fixed decoding configuration of the generator. □

Operationalizing the support set and feature set. For each focal pair (u, j), we instantiate the neighbor support set as

$$S_\theta(u,j) = S_K^u(u) \cup S_K^b(j), \qquad S_K^u(u) = \mathrm{TopK}\left(\alpha_u^{k\to u} : k \in N_u^u\right), \quad S_K^b(j) = \mathrm{TopK}\left(\alpha_b^{l\to j} : l \in N_j^b\right),$$

where α are the prediction-time NIG weights from Eq. (5). To select salient observable attributes, we convert FR (a coefficient vector) into an instance-level contribution score using the decomposition of the linear pre-activation score $\tilde{e}$ (Def. 2). For an edge $(k \to i)$, define the featurewise contribution vector

$$\mathbf{c}_{k\to i}^u = \mathrm{FR}_{\mathrm{self}}^u \odot x_i + \mathrm{FR}_{\mathrm{nb}}^u \odot x_k, \qquad \mathbf{c}_{l\to j}^b = \mathrm{FR}_{\mathrm{self}}^b \odot x_j + \mathrm{FR}_{\mathrm{nb}}^b \odot x_l,$$

where $\odot$ denotes elementwise multiplication and x denotes the observed attribute vector. We then aggregate absolute contributions across the selected support using NIG weights,

$$\mathrm{score}(f) = \sum_{k \in S_K^u(u)} \alpha_u^{k\to u} |c_{k\to u,f}^u| + \sum_{l \in S_K^b(j)} \alpha_b^{l\to j} |c_{l\to j,f}^b|,$$

and take the top-F attributes by $\mathrm{score}(f)$. This procedure ensures that Stage 2 uses only prediction-time evidence and that "top-F features" are defined as NIG-weighted, instance-specific attribute contributions rather than as raw coefficients.

4.1. Hallucination Problems of Vanilla LLMs in Explanation Generation

Even with simple prompts, LLMs can introduce unsupported statements into explanations. In ERS settings, this is especially problematic because an explanation is intended to clarify the system's actual basis for recommending an item and to build trust. In our setting, the dominant failure mode is contextual misattribution: when the prompt contains multiple similar restaurants (e.g., previously liked items plus the recommended item), the LLM may accidentally transfer attributes from one restaurant to another. This motivates support-limited prompting and explicit evidence constraints in Stage 2 (see also large-scale evidence on LLM factual errors in Tonmoy et al. 2024). We focus on contextual misattribution, defined below.

Definition 4 (Contextual Misattribution). Contextual misattribution occurs when characteristics of one item are mistakenly attributed to another within a similar context, leading to incorrect explanations. □

For example, when provided information about multiple similar businesses, an LLM might describe one business as offering "outdoor seating" based on features of a different business mentioned in the input, even if the first business does not have that feature. In our context, such errors can mislead users about what a recommended business offers and thereby undermine trust in the system.

To formally assess the LLM's performance, we evaluated an advanced language model (`GPT-4o`) in generating explanations for high-rated (4 stars or higher) restaurants in Ontario. Following Dai et al. (2023), we designed prompts that combined features of restaurants previously visited by

the user with those of the recommended restaurant. We selected diverse yet controlled categories, including Chinese (takeout, family-friendly), Italian (full bars, outdoor seating), upscale, American (business parking), and Spanish (good for groups, full bars), resulting in a dataset of 102 restaurants.[4]

With this setup, we observed a 24.24% contextual-misattribution rate in the explanations, highlighting the need to reduce unsupported statements in LLM outputs.[5] Under the same audit setup, our constrained MG-GAT-enabled plan-and-critique pipeline, introduced in the next section, reduces this contextual-misattribution rate to 9.8%.

4.2. MG-GAT-Enabled Automated Explanation Generation

Building on Def. 3, we implement Stage 2 as a constrained plan-and-critique procedure that separates (i) evidence selection and (ii) surface realization. We draw on plan-based prompting patterns (e.g., Tree-of-Thoughts (Yao et al. 2024)) as an implementation template, but our design contribution is to use high-salience neighbors and attributes selected by NIG/FR to define the evidence slots and to score and filter candidates so the final narrative is auditable under the evidence contract.

Two-layer contract (evidence vs. realization). We treat Stage 2 as a two-layer contract: (i) a deterministic evidence contract, where the support set (top-K neighbors $S_\theta(u, j)$ and top-F features $\mathcal{F}_j^*$) is fixed given the Stage 1 model and input; and (ii) a stochastic surface realization, where the LLM paraphrases this fixed evidence. To strengthen support-limitedness beyond a soft penalty, we require each candidate plan and explanation to output an `EVIDENCE_USED` line listing the neighbors/features it cites; candidates that cite out-of-support restaurants or attributes are rejected, and the evidence line is stripped before the final explanation is shown to users.

Why use an LLM. An immediate question is why use an LLM at all, rather than fixed human-written templates. We treat the LLM as a surface-realization module: the evidence it may cite is fixed by the NIG/FR support (the evidence contract), and the LLM only paraphrases that evidence under a fixed decoding configuration. As an engineering sanity check on surface form (not as a theoretical claim), we used Microsoft Azure AI Foundry's built-in coherence and fluency diagnostics and found that LLM-realized text scored higher than human-written templates on average (coherence: 4.33 vs. 2.39; fluency: 4.00 vs. 2.83, averaged across six vignette contexts). This demonstrates that, once evidential support is held fixed by the NIG/FR evidence contract, an

[4] We fix the temperature at 0.5 in this generation. This introduces stochasticity in surface realization, but preserves parameter grounding under a fixed decoding configuration; the evidential support is determined by NIG/FR.

[5] Three human annotators reviewed each generated explanation and coded whether it contained any statement that contradicted or was not supported by the factual information provided in the prompt (i.e., whether the explanation conflicted with the restaurant attributes and context shown to the model). Disagreements were resolved by discussion and adjudication. We report the contextual-misattribution rate as the fraction of explanations flagged as containing at least one such unsupported or contradictory statement.

LLM can improve readability/fluency relative to rigid templates without changing the underlying evidence used for the rationale.[6]

As shown in Fig. 2, we structure generation into two phases (plan search, then explanation generation), both scored by NIG-weighted critics. Here, MG-GAT provides only the weights used to aggregate critic scores across neighbors; the critics themselves are implemented as fixed evaluation prompts that check (i) support compliance and (ii) the six explanation-quality criteria given the provided evidence slots. Interpreting each neighbor as a "critic" (a fan of that neighbor restaurant) is a pragmatic proxy for local evidence relevance: if the recommendation is justified via similarity to a neighbor l, then a perspective aligned with l can assess whether the cited attributes plausibly transfer from l to j. Weighting critic scores by NIG makes this alignment explicit and auditable by giving more influence to the same neighbors that MG-GAT relied on more at prediction time (while also surfacing any locality-induced bias as part of the evidence contract). This design reduces information overload in prompts, steers the model away from contextual misattribution, and helps ensure that every explanation remains directly traceable to MG-GAT's prediction-time signals.

Even with support-limited prompts, single-pass generation can drift because the model must decide both (i) what evidence to cite and (ii) how to narrate it. We therefore separate evidence selection (plan) from verbalization (narrative) and use NIG-weighted critics to select the best on-support plan and narrative under the six explanation-quality criteria. These critic scores are used only as an internal selection heuristic among grounded candidates; our reported evidence on explanation quality comes from the randomized human-subject experiment (Section 6), not from LLM self-scores.

First, we start with providing a Prompt 1 that introduces a customer's dining history. Let $\mathcal{H}_u$ denote the set of restaurants previously highly rated by user u. We use NIG to compute business-side neighbor weights $\alpha_b^{l\to j}$ and define $S_\theta(u,j) = \{l \in \mathcal{H}_u : l \text{ is among the top-}K \text{ business neighbors of } j \text{ under } \alpha_b^{l\to j}\}$ as the neighbor set exposed to the LLM. Additionally, each restaurant can have a long list of features, so we let $\mathcal{F}_j^*$ denote the top-F features selected by the NIG-weighted instance-level contribution score score(f) (global FR coefficients $\times$ observed feature values), and include only these in the prompt. Then, the general prompt is combined with Prompt 2 to generate a plan for explanations.

The full prompt templates (general, plan, evaluation, and explanation) are provided in Appendix C. Briefly, the general prompt includes only the top neighbors and features selected by

[6] For deployments requiring maximal controllability or minimal latency, we also provide a deterministic template mode that verbalizes the same NIG/FR evidence without any LLM calls (Appendix C).

NIG/FR; the plan prompt asks the LLM to propose an explanation plan; and the evaluation prompt scores each plan on the six explanation metrics using NIG-weighted critics.[7]

We evaluate each plan on six user-centric explainability metrics. We denote the scores for a plan i for business l (j's neighbor) as $\{S_e^{p,l}, S_f^{p,l}, S_p^{p,l}, S_s^{p,l}, S_t^{p,l}, S_r^{p,l}\}$, corresponding to effectiveness, efficiency, persuasiveness, satisfaction, transparency, and trust. We then compute the average score S_i^p for each plan i, weighted by NIG of agent l towards recommended business j:

$$S_i^p = \sum_{l \in S_\theta(u,j)} \alpha_b^{l \to j} \sum_{k \in \{e,f,p,s,t,r\}} w_k \, S_k^{p,l} \tag{11}$$

When a small calibration sample is available, we learn weights w_k on the $K = 6$ explanation dimensions by regressing an engagement outcome (e.g., adoption or future interest) on the standardized dimension scores, and use these w_k in Eqs. (11) and (14).[8] In our implementation, these weights are calibrated on an independent pilot study conducted prior to the main randomized experiment on August 2024 and then fixed for all explanations shown in the main experiment to avoid outcome leakage. This yields a weighted combination that tilts selection toward the targeted engagement objective. If no calibration data are available, we use a neutral prior with uniform weights $w_k = 1/K$, which corresponds to an equal-weight (convex) average across dimensions.

We then select the best plan as the plan with the highest average score.

$$\text{best plan} = \arg\max_i S_i^p. \tag{12}$$

In the second explanation-generation step, we generate multiple explanations based on the best plan from the calculations above, using Prompt 4. LLMs construct the explanation by focusing on the selected key features and neighboring restaurants. This structured approach allows the LLM to produce clear and accurate explanations, reducing hallucinations. The explanation prompt (Appendix C) then instantiates the best plan into candidate narratives, again constrained to the selected neighbors and features.

We then use Prompt 3 again to evaluate each of the explanations. For each explanation i, we gather the scores across the six user-centric metrics assigned by neighbor l of recommended business j. These scores are denoted as $\{S_e^{e,l}, S_f^{e,l}, S_p^{e,l}, S_s^{e,l}, S_t^{e,l}, S_r^{e,l}\}$ respectively. Next, we compute the average score S_i^e for each explanation i, weighted by the NIG between agent l (a neighbor of j) and the recommended business j:

$$S_i^e = \sum_{l \in S_\theta(u,j)} \alpha_b^{l \to j} \sum_{k \in \{e,f,p,s,t,r\}} w_k \, S_k^{e,l} \tag{13}$$

where $\alpha_b^{l \to j}$ is the NIG from business l to business j.

[7] We do not directly optimize the six scores as explicit rewards: in multi-objective settings, asking an LLM to "maximize" them is unstable and can induce ungrounded, off-policy text. Instead, we constrain generation to NIG/FR and use NIG-weighted critics to select among grounded candidates, implicitly balancing the criteria.

[8] We use coefficients estimated in an independent pilot calibration study (Table G3).

Finally, we select the best explanation according to S_i^e to obtain the final explanation:

$$\text{best explanation} = \arg\max_i S_i^e. \tag{14}$$

PROPOSITION 3 **(Parameter grounding of the Stage-2 explanations)**. *Assume Algorithm 2 is run with fixed MG-GAT parameters θ and a fixed decoding configuration for the LLM (e.g., fixed temperature and sampling scheme). Then there exists a (possibly randomized) mechanism Φ_θ such that, for every user–business pair (u, j), the explanation $E(u, j)$ produced by Algorithm 2 satisfies*

$$E(u,j) \sim \Phi_\theta\Big(\mathrm{NIG}^u_\theta(u,\cdot),\, \mathrm{NIG}^b_\theta(:,j),\, \mathrm{FR}^u_{self},\, \mathrm{FR}^u_{nb},\, \mathrm{FR}^b_{self},\, \mathrm{FR}^b_{nb},\, x_j,\, X(S_\theta(u,j))\Big),$$

and is therefore parameter-grounded in the sense of Definition 3.

Proof in Appendix D.3. Definition 3 and the plan-and-critique procedure formalize our Stage-2 pipeline: explanations are generated from MG-GAT's prediction-time signals (NIG, FR), the recommended item's attributes, and the selected neighbors' attributes under a fixed decoding scheme. We do not fit a separate surrogate explainer to approximate MG-GAT; instead, NIG/FR determine (i) which neighbors and attributes enter the prompt and (ii) how candidate plans and narratives are scored. This contrasts with post-hoc explainers (e.g., SHAP) and with unconstrained LLM prompting that can introduce unsupported content.

COROLLARY 1 **(Prediction–explanation distributional invariance)**. *Under the conditions of Proposition 3, if two user–business pairs (u, j) and (u', j') satisfy*

$$\Big(\mathrm{NIG}^u_\theta(u,\cdot),\, \mathrm{NIG}^b_\theta(:,j),\, x_j,\, X(S_\theta(u,j))\Big) = \Big(\mathrm{NIG}^u_\theta(u',\cdot),\, \mathrm{NIG}^b_\theta(:,j'),\, x_{j'},\, X(S_\theta(u',j'))\Big),$$

then the explanations have identical distributions:

$$E(u,j) \overset{d}{=} E(u',j').$$

Proof in Appendix D.4. This invariance property captures a basic form of prediction–explanation alignment: if MG-GAT assigns two recommendation instances effectively the same internal representation—in terms of which neighbors matter and which features are relevant—then the explanation layer will describe them in the same way. In other words, explanations depend only on what the model "believes" (via NIG/FR) and not on arbitrary idiosyncrasies of the LLM. Generic prompt-based explanations, by contrast, can vary across otherwise equivalent cases because there is no such invariance constraint.

Operationally, we instantiate ToT by treating NIG/FR as control signals and using NIG-weighted critics to select among candidate plans and narratives, so generation remains support-limited to the evidence used for prediction.

PROPOSITION 4 **(Support-limited prompting and selection)**. *Let $S_\theta(u, j)$ denote the high-salience neighbor set used in prompting and let $\mathcal{F}_j^*$ denote the set of top-F features selected by the*

NIG-weighted instance-level contribution score $\text{score}(f)$. *Under the prompting scheme described in Section 4.2, the planner and generator prompts expose the LLM only to entities in* $\{j\} \cup S_\theta(u, j)$ *and features in* $\mathcal{F}_j^*$. *The scoring-and-selection procedure in Algorithm 2 therefore discourages plans and narratives that justify the recommendation using items or features outside this support (in particular, items/features with that are not surfaced in the evidence slots). In our implementation, we further enforce this contract with an evidence checklist: candidates that cite out-of-support items/features are rejected before scoring and selection.*

Proof in Appendix D.5. This support limitation provides a structural form of on-path traceability: even though we use an LLM to verbalize the recommendation, the set of entities and attributes that can be cited as reasons is bounded by MG-GAT's own importance and relevance scores.

Empirically, this support limitation aligns with the reduction in contextual misattribution reported in Section 4.1, and conceptually it distinguishes our approach from unconstrained LLM-based explanations that may generate fluent but off-policy narratives.[9]

Algorithm 2 MG-GAT-Guided Explanation Generation using LLMs

input user u, recommended restaurant j, history $\mathcal{H}_u$, NIG, FR
Select neighbors $S_\theta(u, j)$ from $\mathcal{H}_u$ using top-K NIG weights $\alpha_b^{l \to j}$
Select features $\mathcal{F}_j^*$ using top-F NIG-weighted instance-level contribution scores $\text{score}(f)$
general_prompt ← formatted prompt with $S_\theta(u, j)$ and restaurant j using Prompt 1.
plans ← multiple plans generated by the LLM according to *general_prompt* using Prompt 2 (each includes an `EVIDENCE_USED` header).
Filter *plans* by the evidence contract: discard any plan that cites restaurants/features outside $\{j\} \cup S_\theta(u, j)$ and $\mathcal{F}_j^*$.
scored_plans ← score plan quality for all *plans* using Prompt 3, weighted by NIG using Eqs. (11)–(12)
best_plan ← plan with highest S_i^p (Eq. (12)).
explanations ← multiple explanations based on *general_prompt* and *best_plan* using Prompt 4 (each includes an `EVIDENCE_USED` header).
Filter *explanations* by the evidence contract; strip the `EVIDENCE_USED` header before display.
scored_explanations ← score explanation quality for all *explanations* using Prompt 3, weighted by NIG using Eqs. (13)–(14).
best_explanation ← explanation with highest S_i^e (Eq. (14)).
output *best_explanation*

[9] Appendix C.1 provides provider-agnostic Stage 2 call-budget accounting (template/fast/full modes) and deployment considerations, including standard mitigations such as on-demand explanation, caching, and batching.

5. Empirical Results

We evaluate MG-GAT on the Yelp Open Dataset in two regions to answer two questions: (RQ1) How does MG-GAT compare to strong matrix-completion and graph-based RSs in rating prediction and ranking? (RQ2) Which design components (multi-graph fusion, NIG, FR, and auxiliary/network signals) drive performance?

5.1. Data

We focus on two regions, Ontario (ON) in Canada and Pennsylvania (PA) in the US, and refer readers to Appendix E for full construction details and summary statistics.

Business data and multi-graph construction. We incorporate four types of auxiliary business information: basic business attributes and categories, location, check-in information (temporal popularity), and an LLM-derived perceptual similarity graph (Li et al. 2024). We construct the aggregated business network $\mathbf{G}_b$ using four k-nearest-neighbor components (geographical proximity, consumer co-visitation, shared categories, and perceptual similarity), with $k = 10$ per component to control density and computation.

User data and social network. We use Yelp user metadata (e.g., Elite status and compliments) together with friendship links to construct a user network $\mathbf{G}_u$. This results in 19 user attributes in ON and 33 in PA.

Implicit interaction features. To capture implicit feedback, we binarize observed interactions and extract low-rank implicit features via SVD; these features are included as auxiliary inputs and their influence is regularized toward zero if non-predictive (Appendix E).

5.2. Experimental Setting

The data are divided into training (2009-2016), validation (2017), and testing (2018) sets to align with practical setups for predicting future ratings. Table E2 in Appendix E provides the statistics.

Evaluation metrics. We evaluate both rating prediction and ranking quality using four metrics: RMSE (rating prediction), Spearman's rank-order correlation (ranking), Fraction of Concordant Pairs (FCP; ranking), and Bayesian Personalized Ranking (BPR; ranking) (Rendle et al. 2009, Koren and Sill 2013). RMSE is computed on observed test entries and aligns with the squared-error fidelity term in our objective; the ranking metrics are computed from predicted scores as detailed in Appendix F. For FCP and BPR, we restrict to users with at least two ratings in the test set.

Benchmarks. Our method is evaluated against fourteen benchmark models. The ability of these models to incorporate different sources of information varied, as we summarize in Table A2 in Appendix A. We chose two high-performing non-DL approaches. First, SVD++ (Koren 2008) is a robust benchmark renowned for superior performance compared to other frequently used non-DL methods. Second, GRALS (Rao et al. 2015) is a graph-regularized matrix completion method that can incorporate auxiliary and network information.

We evaluated twelve DL models: NNMF (Dziugaite and Roy 2015), sRGCNN (Monti et al. 2017), GC-MC (Berg et al. 2018), F-EAE (Hartford et al. 2018), GraphRec (Fan et al. 2019), NGCF (Wang et al. 2019b), IGMC (Zhang and Chen 2020), IFM (Pan et al. 2021), HGCL (Chen et al. 2023), LightGCL (Cai et al. 2023), KGAT (Shimizu et al. 2022), and DGAN (Wu et al. 2019). NNMF extends matrix factorization by adding dense layers to predict user-item interactions. NGCF and GraphRec leverage bipartite graphs for matrix completion, while sRGCNN and GC-MC, both GNN-based models, incorporate network data to enhance accuracy. Models like IFM, KGAT, and HGCL utilize auxiliary data to enrich recommendations. Semi-inductive models like F-EAE and IGMC achieve strong performance by focusing on local networks but lack interpretability. HGCL and LightGCL apply contrastive learning to refine embeddings and improve accuracy. Additionally, DGAN incorporates network data to capture complex user-item interactions.

5.3. Evaluation of Learning Performance

We compare the performance of MG-GAT with several benchmarks in Table 2. To interpret these results, Appendix A (Table A2) indicates which baselines use auxiliary attributes and/or network data; MG-GAT is among the few that use both. Our method is evaluated in two versions: interpretable and uninterpretable models, which differ by whether NIG has a linear or a nonlinear relationship with features (hence whether FR is interpretable or not). The uninterpretable MG-GAT outperforms all other methods across all four evaluation metrics.

Specifically, on the ON dataset, our method attains an RMSE of 1.130, better than DGAN's 1.203 and the interpretable MG-GAT's 1.210. On the PA dataset, our uninterpretable MG-GAT achieves an RMSE of 1.217, surpassing the next best methods, the interpretable MG-GAT and DGAN (Wu et al. 2019), which have RMSEs of 1.249 and 1.250, respectively. This corresponds to a relative RMSE improvement of approximately 6.5% in ON and 2.7% in PA compared to DGAN, the strongest benchmark among the others.

Among uninterpretable methods, the best-performing models are IGMC (Zhang and Chen 2020), GRALS (Rao et al. 2015), and HGCL (Chen et al. 2023). IGMC attains RMSEs of 1.268 (ON) and 1.305 (PA) but is still outperformed by MG-GAT. GRALS and HGCL both use auxiliary and network data but remain black-box: GRALS reaches 1.279 (ON) and 1.328 (PA), and HGCL 1.310 (ON) and 1.323 (PA). Compared with these uninterpretable baselines, MG-GAT achieves lower RMSE while retaining interpretability, underscoring the value of our approach to integrating auxiliary and network information.

Two other interpretable methods, IFM (Pan et al. 2021) and KGAT (Shimizu et al. 2022), are noteworthy because they expose built-in explanation cues while pursuing strong predictive performance. Both incorporate auxiliary information but lack network data integration. IFM achieves

RMSEs of 1.410 in ON and 1.490 in PA, while KGAT scores 1.383 in ON and 1.450 in PA. Although these models leverage auxiliary data and provide explanation signals, their performance is inferior to MG-GAT, highlighting the advantage of combining both auxiliary and network information.

Given that the performance difference between the interpretable and uninterpretable versions of MG-GAT is minor, approximately 6.6% in ON and 2.6% in PA, we argue for the importance of interpretability. Specifically, the RMSE difference between the interpretable and uninterpretable MG-GAT is 0.080 (1.210 vs. 1.130) in ON and 0.032 (1.249 vs. 1.217) in PA. Considering this modest trade-off, we advocate for the interpretable MG-GAT, as it provides significant benefits in understanding the model's decisions without substantially compromising predictive performance, and can be used to inform MG-GAT-based explanation generation (Section 4).

The trade-off between interpretability and predictive performance is well recognized in the IML literature (Arrieta et al. 2020) and is reflected in our analysis. Our results indicate that the performance difference between interpretable and black-box models is modest given the added value of interpretability. Rather than focusing solely on maximizing predictive power, we aim to deliver competitive performance with an interpretable RS architecture that exposes NIG and FR as prediction-time signals.

5.4. Ablation Studies

We evaluate how MG-GAT's performance depends on (i) multi-graph fusion and interpretable components (NIG, FR), (ii) network inputs, (iii) auxiliary features, and (iv) added nonlinearities/alternative attention.

5.4.1. Model Design Choices. Three design choices matter. Using uniform (rather than learned) graph weights increases RMSE to 1.280 (PA) and 1.230 (ON), implying that learning graph-fusion weights improves accuracy (about 2.5% in PA and 1.7% in ON). Removing FR increases RMSE to 1.305 (PA) and 1.254 (ON), and removing NIG increases RMSE to 1.303 (PA) and 1.257 (ON). Overall, learned fusion, FR, and NIG each contribute to predictive accuracy.

5.4.2. Value of Network Information. Network structure adds signal beyond the observed user–business interactions. Replacing user–user and business–business graphs with only the user–business interaction graph increases RMSE to 1.288 (PA) and 1.271 (ON). Degrading network quality also hurts: shuffling 50% of edges yields RMSE 1.283 (PA) and 1.265 (ON), and removing 50% of edges yields RMSE 1.293 (PA) and 1.270 (ON). Removing both network and auxiliary information produces the largest degradation (RMSE 1.405 in PA; 1.325 in ON), indicating that both sources are important.

Table 2 Performance evaluations on Yelp.

State	Method	**RMSE** (Rating)	**Spearman** (Ranking)	**FCP** (Ranking)	**BPR** (Ranking + Rating)
ON	SVD++ (Koren et al. 2009)	1.284 (9e-05)	0.362 (0.0001)	0.57 (0.0002)	0.511 (3e-05)
	GRALS (Rao et al. 2015)	1.279 (9e-05)	0.372 (0.0001)	0.59 (0.0001)	0.515 (2e-05)
	NNMF (Dziugaite and Roy 2015)	1.323 (9e-05)	0.277 (0.0001)	0.538 (0.0002)	0.496 (2e-05)
	sRGCNN (Monti et al. 2017)	1.372 (9e-05)	0.115 (0.0001)	0.538 (0.0001)	0.502 (2e-05)
	F-EAE (Hartford et al. 2018)	1.317 (0.0001)	0.302 (0.0001)	0.567 (0.0001)	0.508 (1e-05)
	GC-MC (Berg et al. 2018)	1.291 (0.0001)	0.368 (0.0001)	0.589 (0.0001)	0.514 (3e-05)
	GraphRec (Fan et al. 2019)	1.602 (9e-05)	0.109 (8e-05)	0.225 (0.0001)	0.205 (0.0001)
	NGCF (Wang et al. 2019a)	1.472 (0.0010)	0.023 (0.0010)	0.528 (0.0014)	0.493 (0.0002)
	DGAN (Wu et al. 2019)	1.203 (0.0001)	0.485 (0.0001)	0.610 (0.0001)	0.529 (0.0001)
	IGMC (Zhang and Chen 2020)	1.268 (0.0001)	0.383 (0.0001)	0.599 (0.00014)	0.520 (4e-05)
	IFM (Pan et al. 2021)	1.410 (9e-5)	0.175 (0.0002)	0.550 (0.0004)	0.513 (1e-5)
	KGAT (Shimizu et al. 2022)	1.383 (0.0001)	0.335 (0.0002)	0.576 (0.0002)	0.508 (2e-5)
	HGCL (Chen et al. 2023)	1.310 (0.0001)	0.305 (0.0002)	0.570 (0.0001)	0.512 (1e-05)
	LightGCL (Cai et al. 2023)	1.394 (0.0001)	0.321 (0.0004)	0.576 (0.0003)	0.504 (3e-5)
	MG-GAT [FR and NIG interpretable]	1.210 (0.0002)	0.480 (0.0002)	0.605 (0.0003)	0.530 (4e-05)
	MG-GAT [FR uninterpretable and NIG interpretable]	1.130 (0.0001)	0.493 (0.0001)	0.613 (0.0003)	0.531 (4e-05)
PA	SVD++ (Koren et al. 2009)	1.339 (0.0001)	0.356 (0.0002)	0.579 (0.0003)	0.512 (5e-05)
	GRALS (Rao et al. 2015)	1.328 (0.0002)	0.367 (0.0002)	0.564 (0.0003)	0.508 (5e-05)
	NNMF (Dziugaite and Roy 2015)	1.447 (0.0023)	0.044 (0.0021)	0.508 (0.0034)	0.500 (0.0001)
	sRGCNN (Monti et al. 2017)	1.424 (0.00014)	0.173 (0.00017)	0.544 (0.00024)	0.503 (3e-05)
	F-EAE (Hartford et al. 2018)	1.382 (0.0002)	0.336 (0.0002)	0.577 (0.0003)	0.510 (2e-05)
	GC-MC (Berg et al. 2018)	1.342 (0.0002)	0.367 (0.0002)	0.590 (0.0002)	0.514 (4e-05)
	GraphRec (Fan et al. 2019)	1.676 (0.0002)	0.107 (0.0001)	0.216 (0.0001)	0.203 (0.0001)
	NGCF (Wang et al. 2019a)	1.621 (0.0017)	0.022 (0.0015)	0.511 (0.0022)	0.496 (0.0003)
	DGAN (Wu et al. 2019)	1.250 (0.0001)	0.401 (0.0001)	0.602 (0.0001)	0.528 (0.0001)
	IGMC (Zhang and Chen 2020)	1.305 (0.00016)	0.399 (0.0002)	0.598 (0.0002)	0.520 (6e-05)
	IFM (Pan et al. 2021)	1.490 (9e-5)	0.120 (0.0010)	0.519 (0.0002)	0.483 (1e-5)
	KGAT (Shimizu et al. 2022)	1.450 (0.0011)	0.220 (0.0015)	0.518 (0.0014)	0.496 (0.0002)
	HGCL (Chen et al. 2023)	1.323 (0.0002)	0.372 (0.0002)	0.559 (0.0023)	0.502 (5e-05)
	LightGCL (Cai et al. 2023)	1.430 (0.0001)	0.255 (0.0005)	0.530 (0.0015)	0.495 (0.0002)
	MG-GAT [FR and NIG interpretable]	1.249 (9e-05)	0.405 (0.0001)	0.602 (0.00014)	0.520 (3e-05)
	MG-GAT [FR uninterpretable and NIG interpretable]	1.217 (0.0001)	0.430 (0.0001)	0.645 (0.0003)	0.551 (4e-05)

Note: See Table A2 in Appendix A.2 for each baseline's inputs. A lower RMSE, a higher FCP, a higher BPR, and a higher Spearman correlation correspond to better performance. We test the performance of the model using 1,000 bootstrap samples from the test set. FCP and BPR are evaluated only on users that have more than one rating in the test set, which consists of 37% and 30% of users in the test set, in ON and PA.

5.4.3. Value of Auxiliary Information. Auxiliary features materially improve prediction. Removing auxiliary information increases RMSE to 1.312 (PA) and 1.265 (ON). Using only implicit features (latent embeddings) yields RMSE 1.300 (PA) and 1.260 (ON), while using only the top ten predictive features yields RMSE 1.260 (PA) and 1.230 (ON), indicating that much of the predictive value is concentrated in a small subset of features. In contrast, using only the bottom ten (least predictive) features for NIG performs similarly to the implicit-only setting, suggesting that non-predictive features contribute little.

A practical diagnostic for feature quality is to compare a model that uses auxiliary features to one that omits them (or uses only implicit features). If performance is similar, auxiliary features likely add limited predictive value and should be excluded from interpretive explanations to avoid highlighting spurious relationships. More generally, feature screening is important: explanations are most useful when the surfaced features are demonstrably predictive and stable.

5.4.4. Predictive Accuracy vs. Interpretability. Adding nonlinearity can improve RMSE but reduces transparency. Adding one nonlinear layer to the uniformly weighted model reduces RMSE from 1.280 to 1.270 in PA and from 1.230 to 1.210 in ON (about 0.8% and 1.6% gains). With learned graph weighting and three nonlinear layers, RMSE reaches 1.217 (PA) and 1.130

Table 3 Ablation Studies for Different Configurations for ON and PA

State	Category	Configuration	**RMSE** (Rating)	**FCP** (Ranking)	**BPR** (Ranking + Rating)	**Spearman** (Ranking)
PA	Main Model	Graph Weighting (Interpretable)	1.249 (9e-05)	0.602 (0.0001)	0.520 (3e-05)	0.405 (0.0002)
	Model Design	Uniform Graph Weighting	1.280 (0.0001)	0.602 (0.0001)	0.519 (3e-05)	0.400 (0.0002)
		NIG Removed	1.303 (0.0001)	0.597 (0.0002)	0.518 (4e-05)	0.395 (0.0002)
		FR Removed	1.305 (0.0002)	0.596 (0.0002)	0.518 (4e-05)	0.389 (0.0002)
	Value of Network Information	User-Business Network as Graph	1.288 (0.0002)	0.600 (0.0003)	0.514 (3e-05)	0.403 (0.0002)
		Shuffled 50% Networks	1.283 (0.0002)	0.605 (0.0003)	0.520 (4e-05)	0.408 (0.0002)
		Missing 50% Networks	1.293 (0.0001)	0.601 (0.0002)	0.519 (4e-5)	0.399 (0.0002)
		No Networks or Auxiliary Information	1.405 (0.0001)	0.574 (0.0002)	0.507 (1e-05)	0.332 (0.0002)
	Value of Auxiliary Information	No Auxiliary Information	1.312 (0.0002)	0.586 (0.0002)	0.516 (4e-05)	0.384 (0.0002)
		Implicit Features Only for NIG	1.280 (0.0001)	0.601 (0.0002)	0.519 (3e-05)	0.395 (0.0001)
		Top Ten Predictive Features for NIG	1.260 (0.0001)	0.600 (0.0001)	0.518 (2e-05)	0.403 (0.0004)
		Bottom Ten Predictive Features for NIG	1.280 (0.0001)	0.600 (0.0002)	0.519 (4e-05)	0.395 (0.0002)
	Interpretability vs. Predictability	Uniform Weighting+ One Nonlinear	1.270 (0.0002)	0.611 (0.0001)	0.520 (2e-05)	0.405 (0.0001)
		Graph Weighting + One Nonlinear	1.250 (0.0002)	0.621 (0.0003)	0.528 (2e-05)	0.410 (0.0002)
		Uniform Weighting + Three Nonlinear	1.240 (0.0001)	0.623 (0.0002)	0.532 (2e-05)	0.415 (0.0002)
		Graph Weighting + Three Nonlinear	1.217 (0.0001)	0.645 (0.0003)	0.551 (4e-05)	0.430 (0.0001)
		Dynamic Attention	1.240 (0.0001)	0.605 (0.0003)	0.527 (2e-5)	0.411 (0.0002)
ON	Main Model	Graph Weighting (Interpretable)	1.210 (0.0002)	0.605 (0.0003)	0.530 (4e-05)	0.480 (0.0002)
	Model Design	Uniform Graph Weighting	1.230 (0.0003)	0.594 (0.0001)	0.519 (2e-05)	0.493 (0.0003)
		NIG Removed	1.257 (9e-05)	0.592 (0.0001)	0.518 (3e-05)	0.390 (0.0001)
		FR Removed	1.254 (0.0002)	0.589 (0.0001)	0.517 (3e-5)	0.398 (0.0001)
	Value of Network Information	User-Business Network as Graph	1.271 (0.0001)	0.575 (0.0002)	0.509 (4e-05)	0.459 (0.0001)
		Shuffled 50% Networks	1.265 (8e-05)	0.583 (0.0001)	0.516 (2e-05)	0.377 (0.0001)
		Missing 50% Networks	1.270 (8e-5)	0.585 (0.0001)	0.516 (1e-5)	0.383 (0.0001)
		No Networks or Auxiliary Information	1.325 (8e-05)	0.575 (0.0001)	0.510 (1e-05)	0.344 (0.0001)
	Value of Auxiliary Information	No Auxiliary Information	1.265 (9e-5)	0.585 (0.0001)	0.516 (3e-5)	0.382 (0.0001)
		Implicit Features Only for NIG	1.245 (0.0001)	0.592 (0.0002)	0.518 (4e-05)	0.420 (0.0002)
		Top Ten Predictive Features for NIG	1.230 (0.0001)	0.599 (0.0002)	0.520 (3e-05)	0.475 (0.0001)
		Bottom Ten Predictive Features for NIG	1.244 (0.0001)	0.592 (0.0002)	0.518 (4e-05)	0.421 (0.0002)
	Interpretability vs. Predictability	Uniform Weighting + One Nonlinear	1.210 (0.0001)	0.602 (0.0002)	0.525 (2e-05)	0.498 (0.0004)
		Graph Weighting + One Nonlinear	1.220 (0.0002)	0.610 (0.0003)	0.532 (2e-05)	0.460 (0.0002)
		Uniform Weighting + Three Nonlinear	1.170 (0.0001)	0.618 (0.0001)	0.528 (3e-05)	0.495 (0.0002)
		Graph Weighting + Three Nonlinear	1.130 (0.0001)	0.613 (0.0003)	0.531 (4e-05)	0.493 (0.0001)
		Dynamic Attention	1.210 (0.0001)	0.606 (0.0002)	0.531 (3e-5)	0.480 (0.0001)

(ON), improving by 2.6% (PA) and 6.6% (ON) over the main interpretable model. However, these nonlinearities break the direct feature-level mapping from auxiliary attributes to NIG/FR, making the model harder to audit.

We also test dynamic attention (Brody et al. 2022). Dynamic attention yields RMSE 1.240 (PA) and 1.210 (ON): a marginal gain in PA and no gain in ON, while further reducing interpretability due to more complex transformations.

6. Experiment on Parameter-Grounded Explanations

In this section, we evaluate whether the parameter-grounded explanations improve user responses to recommendations relative to alternatives. We conduct a between-subjects randomized controlled experiment that emulates a Yelp search scenario and compares five conditions (no explanation; relevant-item; social; SHAP+LLM; and MG-GAT+LLM). We measure recommendation acceptance and engagement (perceived relevance and future interest) and perceived explanation quality along six explanation-quality criteria (Tintarev and Masthoff 2012a).

We frame this human-subject experiment as preference-conditioned scenarios rather than longitudinal personalization. Because the experimental conditions differ in both the *evidence interface* (what information is surfaced/selected) and, for some baselines, the *surface realization* mechanism,

we interpret treatment effects as comparisons of end-to-end explanation strategies rather than as isolating a single component in isolation.

6.1. Design

Our experimental design emulates a typical Yelp usage scenario in which a user evaluates restaurant recommendations. Participants answered on behalf of their closest friend rather than themselves. This third-person framing follows the projective questioning tradition in consumer and survey research for vignette-based judgments (Fisher 1993, Haire 1950, King and Bruner 2000). We adopt this *third-person technique* (indirect/projective questioning) to reduce self-presentation concerns and socially desirable responding in self-reports and to help participants apply a specified preference profile consistently in a hypothetical scenario (Fisher 1993, Krumpal 2013, Tourangeau and Yan 2007, Aguinis and Bradley 2014). This design choice is distinct from the communication-literature *Third-Person Effect* (self–other persuasion gaps) (Davison 1983, Perloff 1993, Sun et al. 2008); we therefore do not interpret the results as evidence about differential susceptibility to persuasion. Because the friend-referent framing is held constant across all conditions, our primary inferences are comparative across explanation methods under a common vignette frame. As is standard for vignette experiments, we interpret the outcomes as perceived usefulness and engagement intentions in this controlled scenario; examining behavioral impacts in first-person, repeated-use settings is an important direction for future work.

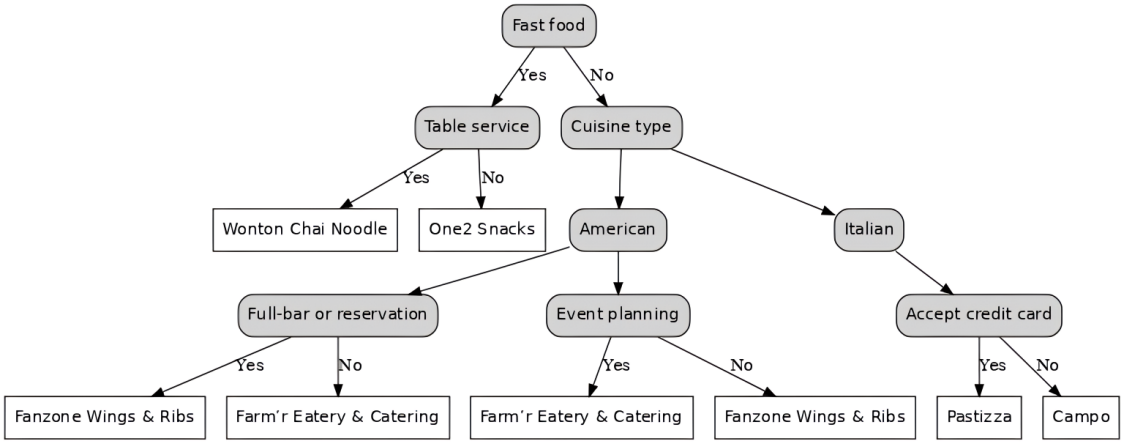


Figure 3 Preference-conditioned experiment design showing how survey questions lead to final recommendations.

Experiment Setup. Using a short branching questionnaire over key preferences (cuisine, fast food versus non–fast food, and service/amenity attributes; Fig. 3), each participant was assigned to a realistic recommendation scenario consisting of two previously liked restaurants (used as context and referenced in the explanations) and one focal recommendation produced by MG-GAT

for a matched user in the Yelp data. Participants then viewed a Yelp-style search result with the recommended restaurant highlighted. Conditional on the elicited preference path (Fig. 3), we then randomized participants into one of the five explanation conditions below. Full branching logic and vignette wordings are provided in Appendix G.

6.1.1. Experiment Conditions . The experiment comprises five conditions: one control and four explanation treatments. The treatment and control interfaces are included in Appendix G.2.[10]

In the control condition ("No Explanation"), the recommendation is presented without any contextual information about the algorithmic decision-making process.

Treatment Condition I is the "Relevant Business Explanation" condition, also referred to as a "relevant item" explanation in the ERS literature (Zhang et al. 2020). Such explanations are typically generated using item-based collaborative filtering, which identifies items similar to the recommendation based on user–item interactions. This type of explanation is common on platforms like Amazon and Netflix (see Appendix H).

In Treatment Condition II (the "Park et al. (2017) Explanation" condition), the explanation is adapted from Park et al. (2017) and represents a "social explanation" based on user networks. Park et al. (2017) is closely related to our setting because it is validated via human experiments and leverages both rating and social information for ERS. In this condition, the recommendation is explained through the preferences of the user's friends (i.e., social ties), instantiated in the same interface format and with the same LLM-based surface realization as the other explanation conditions so that the primary difference is the evidence slot.

In Treatment Condition III (the "SHAP Explanation" condition), we first train a supervised rating predictor on the experimental subset (attributes only) and compute SHAP (SHapley Additive exPlanations) values for the focal recommendation. We then identified the top three features with the largest positive SHAP contributions for that item and provided the LLM with these features (as the evidence slot) together with the two previously liked restaurants shown in the scenario, using the same decoding configuration as the other explanation conditions.

In Treatment Condition IV (the "MG-GAT Explanation" condition), the explanation is generated by integrating the LLM with MG-GAT using the method developed in Section 4.

Each explanation text is pre-generated using `GPT-4o`. Word counts are comparable across Treatment Conditions I–IV (60, 66, 68, and 67 words, respectively), helping ensure that differences are not driven by superficial formatting. All LLM-based explanation treatments use the same LLM and fixed decoding configuration; the treatments primarily differ in the evidence interface (what information is surfaced/selected) and, for the MG-GAT condition, the constrained plan-and-critique pipeline used to populate and validate the evidence slots.

[10] Explanations for all preference paths are presented in Figures G1–G6 of Appendix G.2. For the remainder of this section, we use this path as the illustrative example and direct readers to Appendix G.2 for the other paths.

6.1.2. Evaluation Metrics. After their interaction with the recommendation page, participants were presented with a series of statements assessing their acceptance behavior, trust in the RS, and the perceived utility of the explanations. Participants indicated their level of agreement on a 7-point Likert scale, allowing us to quantify engagement and responses to the recommendations.

We first measure Perceived Relevance (PR) and Future Interest (FI). Specifically, PR is measured by "*The recommended item, `[recommended restaurant]`, matched your friend's interests.*" FI is measured by "*Your friend would feel more inclined to try other recommended businesses provided by the same platform in the future.*" Second, we evaluate the explanation quality to the treatment conditions (excluding the control). These questions assess explanation quality along our design dimensions (effectiveness, efficiency, persuasiveness, satisfaction, transparency, and trust).[11]

Consistent with prior ERS work, these measures are collected as single-item perceived judgments; we therefore interpret them as perceived trust/transparency/etc. under the vignette frame rather than as fully validated multi-item psychological scales.

6.1.3. Participant Recruitment and Experimental Procedure. We recruited a total of $380 \times 5 = 1,900$ participants from CloudResearch (`www.cloudresearch.com`) and randomly assigned them to one of five conditions.[12] An initial screening question ensured that only participants who had previously used Yelp (or a similar platform) were included, so that respondents were familiar with the interface and context. We also removed participants who failed attention check questions, leaving a total of 1363 participants (324, 288, 266, 233 and 252 in the five conditions, respectively). [13]

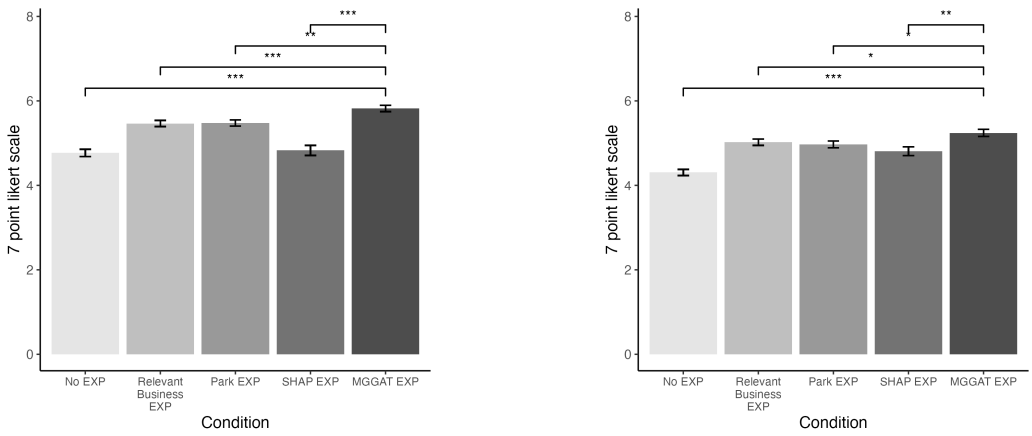
6.2. Main Results

We conducted balance checks to confirm that our experimental conditions were comparable across key covariates. Specifically, we used chi-squared tests for gender and race and one-way ANOVA for age, inclination to explore new restaurants, and years of Yelp use; in all cases, we found no significant differences across conditions. This suggests that observed effects can be attributed to

[11] The full list of questions can be found in Table G1 of Appendix G. These statements were developed based on relevant questions used in Tintarev and Masthoff (2012a).

[12] This study was reviewed and approved by the Institutional Review Board at the author's university (anonymized for review purposes).

[13] To ensure that subjects across all conditions had a similar openness to recommendations and similar use experience in the past on Yelp, we posed two additional questions at the end of the questionnaire. The first asked participants to what extent they agreed with the statement, "*I like to explore new restaurants.*" The second asked, "*How many years have you used a crowdsourced local business review platform, such as Yelp or Google Reviews?*" Moreover, to validate our experimental manipulation, subjects were asked the following question as a manipulation check: "*The platform provided a detailed explanation of why `[recommended restaurant]` was recommended.*" This check verifies that participants noticed the presence of an explanation. Lastly, we collected demographic information, including age, gender, and race. To encourage attention on open-ended items, we required participants to spend at least one minute before submitting their textual responses.



(a) Perceived Relevance (b) Future Interest

Figure 4 Evaluation of the RSs. Perceived Relevance is measured by: "The recommended item matched your friend's interests." Future Interest is measured by: "Your friend would feel more inclined to try other recommended businesses provided by the same platform in the future."

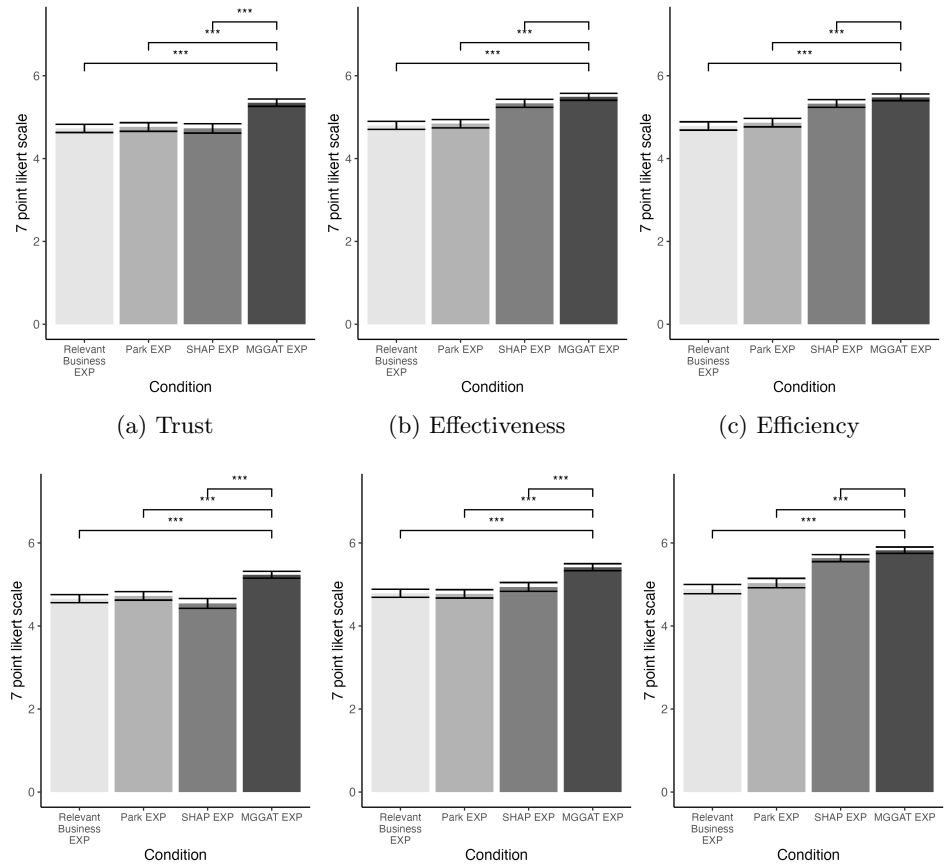
Note: The assignment of stars is determined by the p-value obtained from a statistical comparison between two conditions. Here, three stars are assigned for a p-value < 0.001, two stars for a p-value < 0.01, and one star for a p-value < 0.05.

variation in the explanation treatments rather than pre-existing group differences. In contrast, the manipulation-check items differed significantly across conditions (ANOVA with Tukey HSD post hoc tests), indicating that participants' perceptions of the explanations varied as intended.

Figure 4 reports Perceived Relevance (PR) and Future Interest (FI). Across both measures, MG-GAT-enabled explanation yields the highest engagement (PR: $M = 5.82, SD = 1.19$; FI: $M = 5.24, SD = 1.33$) and significantly exceeds the no-explanation control and each baseline explanation condition. Full statistics and additional outcomes are reported in Appendix G (Table G1). Fig. 5 shows that MG-GAT explanations also achieve the highest mean scores on all six explanation-quality criteria. Pairwise comparisons indicate that, relative to SHAP+LLM, MG-GAT-enabled explanation significantly improves trust, persuasiveness, and satisfaction, while differences on transparency, effectiveness, and efficiency are not statistically distinguishable (Table G1).

6.3. Mechanisms: Exploratory Mediation Analysis of Explanation Quality

We explore potential pathways via an exploratory mediation analysis that relates treatment assignment to engagement outcomes (PR and FI) through participants' perceived explanation-quality ratings. Specifically, we estimate mediation-style models using the six explanation-quality dimensions (Table A1 in Appendix A.1) as post-treatment variables (see Table G2 in Appendix G). Across comparisons, MG-GAT-enabled explanation yields higher mean ratings on all six dimensions, and including these ratings in outcome regressions attenuates the estimated treatment coefficients. This pattern is consistent with perceived explanation quality as a plausible pathway linking explanation design to user engagement. We emphasize, however, that this analysis is descriptive with respect to mechanism and does not by itself establish a causal mediator→outcome relationship.



(a) Trust (b) Effectiveness (c) Efficiency

(d) Persuasiveness (e) Satisfaction (f) Transparency

Figure 5 User-centric evaluations on the explanations (Tintarev and Masthoff 2012a).

Note: The assignment of stars is determined by the p-value obtained from a statistical comparison between two conditions. Here, three stars are assigned for a p-value < 0.001, two stars for a p-value < 0.01, and one star for a p-value < 0.05.

7. Conclusion

We develop and evaluate a two-stage, explanation-integrated recommender that couples prediction and explanation via shared, model-derived signals. In Stage 1, MG-GAT fuses user–user and business–business networks with auxiliary attributes and exposes two prediction-time interpretation signals: NIG, which identifies influential neighbors for a recommendation, and FR, which traces this influence to observable attributes. We formally characterize these signals and show how they provide a parameter-grounded interface between model inference and explanation. In Stage 2, we use NIG/FR as control inputs to a constrained plan-and-critique LLM generator, restricting explanations to a small support of high-salience neighbors and attributes. Empirically, MG-GAT matches or exceeds strong graph- and KG-based baselines in recommendation accuracy, while the constrained explanation pipeline reduces contextual misattribution relative to naïve prompting and improves user acceptance, satisfaction, and perceived trust in a randomized experiment.

Portability Beyond MG-GAT. Parameter-grounded explanation is a *conditional* design principle: it is portable to other recommenders only when the deployed predictor exposes (i) a discrete,

instance-level evidence interface (e.g., top-K history items, neighbors, edges, or paths), (ii) an auditable mapping from that evidence to *observable* attributes (either natively or via a constrained approximation whose faithfulness can be assessed), and (iii) an explicit generation contract that restricts the explanation layer to this interface under a fixed decoding configuration. MG-GAT instantiates these conditions via NIG (tie-level salience) and FR (feature-level decomposition), yielding a native interface in feature space.

These conditions do *not* hold automatically for all attention-based models. For sequential recommenders, attention over a user's history can define an NIG-like top-K evidence set, but a feature-level rationale requires either a native decomposition or a validated constrained approximation over observable metadata (e.g., categories or attributes). For other GNN recommenders, edge- or path-influence scores can serve as the discrete evidence interface; when a model lacks a native feature-space decomposition, any surrogate attribution used to populate the interface should be treated as a limitation rather than as model-intrinsic evidence. Accordingly, we view portability as a promising direction for future work, but we do not evaluate cross-architecture generalization in this paper.

Our work is not without limitations; future work can (i) evaluate true user control in an interactive interface, (ii) use fully crossed factorial designs that disentangle evidence sources from surface realization (e.g., same content with different phrasings, and same phrasing with different content sources), (iii) test long-horizon behavioral impacts in field deployments, and (iv) examine whether and when the parameter-grounded interface transfers to other recommender architectures beyond MG-GAT.

References

Abbasi A, Parsons J, Pant G, Sheng ORL, Sarker S (2024) Pathways for design research on artificial intelligence. *Information Systems Research* .

Adomavicius G, Tuzhilin A (2005) Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering* 17(6):734–749.

Aguinis H, Bradley KJ (2014) Best practice recommendations for designing and implementing experimental vignette methodology studies. *Organizational Research Methods* 17(4):351–371.

Arrieta AB, Díaz-Rodríguez N, Del Ser J, Bennetot A, Tabik S, Barbado A, García S, Gil-López S, Molina D, Benjamins R, et al. (2020) Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion* 58:82–115.

Bapna R, Gupta A, Rice S, Sundararajan A (2017) Trust and the strength of ties in online social networks: An exploratory field experiment. *MIS Quarterly* 41(1).

Bauer K, von Zahn M, Hinz O (2023) Expl (ai) ned: The impact of explainable artificial intelligence on users' information processing. *Information Systems Research* .

Bauman K, Liu B, Tuzhilin A (2017) Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews. *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 717–725.

Bauman K, Tuzhilin A (2018) Recommending remedial learning materials to students by filling their knowledge gaps. *MIS Quarterly* 42(1):313–A7.

Bauman K, Tuzhilin A (2022) Know thy context: parsing contextual information from user reviews for recommendation purposes. *Information Systems Research* 33(1):179–202.

Bauman K, Tuzhilin A, Unger M (2024) HyperCARS: Using hyperbolic embeddings for generating hierarchical contextual situations in context-aware recommender systems. *Information Systems Research* .

Berg Rvd, Kipf TN, Welling M (2018) Graph convolutional matrix completion. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* .

Biddle BJ (1986) Recent developments in role theory. *Annual review of sociology* 12(1):67–92.

Bobadilla J, Ortega F, Hernando A, Gutiérrez A (2013) Recommender systems survey. *Knowledge-based systems* 46:109–132.

Brody S, Alon U, Yahav E (2022) How attentive are graph attention networks? *International Conference on Learning Representations*.

Cai D, He X, Han J, Huang TS (2010) Graph regularized nonnegative matrix factorization for data representation. *IEEE transactions on pattern analysis and machine intelligence* 33(8):1548–1560.

Cai X, Huang C, Xia L, Ren X (2023) Lightgcl: Simple yet effective graph contrastive learning for recommendation. *International Conference on Learning Representations*.

Chen C, Zhang M, Liu Y, Ma S (2018a) Neural attentional rating regression with review-level explanations. *Proceedings of the 2018 World Wide Web Conference*, 1583–1592.

Chen CM, Wang CJ, Tsai MF, Yang YH (2019) Collaborative similarity embedding for recommender systems. 2637–2643.

Chen J, Liu Z, Huang X, Wu C, Liu Q, Jiang G, Pu Y, Lei Y, Chen X, et al. (2024) When large language models meet personalization: Perspectives of challenges and opportunities. *World Wide Web* 27(4):42.

Chen M, Huang C, Xia L, Wei W, Xu Y (2023) Heterogeneous graph contrastive learning for recommendation. *Proceedings of ACM International Conference on Web Search and Data Mining*, 544–552.

Chen X, Xu H, Zhang Y, Tang J, Cao Y, Qin Z (2018b) Sequential recommendation with user memory networks. *Proceedings of ACM International Conference on Web Search and Data Mining*, 108–116.

Crandall DJ, Backstrom L, Cosley D, Suri S, Huttenlocher D, Kleinberg J (2010) Inferring social ties from geographic coincidences. *Proceedings of the National Academy of Sciences* 107(52):22436–22441.

Dai S, Shao N, Zhao H, Yu W, Si Z, Xu C, Sun Z, Zhang X, Xu J (2023) Uncovering chatgpt's capabilities in recommender systems. *Proceedings of the 17th ACM Conference on Recommender Systems*, 1126–1132.

Davison WP (1983) The third-person effect in communication. *Public Opinion Quarterly* 47(1):1–15, URL `http://dx.doi.org/10.1086/268763`.

Dziugaite GK, Roy DM (2015) Neural network matrix factorization. *arXiv preprint arXiv:1511.06443* .

Fan W, Ma Y, Li Q, He Y, Zhao E, Tang J, Yin D (2019) Graph neural networks for social recommendation. *World Wide Web*, 417–426.

Fisher RJ (1993) Social desirability bias and the validity of indirect questioning. *Journal of consumer research* 20(2):303–315.

Gao J, Wang X, Wang Y, Xie X (2019) Explainable recommendation through attentive multi-view learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3622–3629.

Gao S, Wang Y, Fang J, Chen L, Han P, Shang S (2024) Dre: Generating recommendation explanations by aligning large language models at data-level. *arXiv preprint arXiv:2404.06311* .

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

Gedikli F, Jannach D, Ge M (2014) How should i explain? a comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies* 72(4):367–382.

Geng S, Liu S, Fu Z, Ge Y, Zhang Y (2022) Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). *Proceedings of the 16th ACM Conference on Recommender Systems*, 299–315.

Geng S, Tan J, Liu S, Fu Z, Zhang Y (2023) Vip5: Towards multimodal foundation models for recommendation. *arXiv preprint arXiv:2305.14302* .

Granovetter MS (1977) The strength of weak ties. *Social networks*, 347–367 (Elsevier).

Grieser W, LeSage J, Zekhnini M (2021) Industry networks and the geography of firm behavior. *Management Science* .

Guidotti R, Monreale A, Ruggieri S, Turini F, Giannotti F, Pedreschi D (2018) A survey of methods for explaining black box models. *ACM Computing Surveys* 51(5).

Hada DV, Shevade SK (2021) Rexplug: Explainable recommendation using plug-and-play language model. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 81–91.

Haire M (1950) Projective techniques in marketing research. *Journal of marketing* 14(5):649–656.

Hartford J, Graham DR, Leyton-Brown K, Ravanbakhsh S (2018) Deep models of interactions across sets. *International Conference on Machine Learning (ICML)* .

He X, Deng K, Wang X, Li Y, Zhang Y, Wang M (2020) LightGCN: Simplifying and powering graph convolution network for recommendation. *Proceedings of International ACM SIGIR conference on research and development in Information Retrieval*, 639–648.

Herlocker JL, Konstan JA, Riedl J (2000) Explaining collaborative filtering recommendations. *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, 241–250.

Herm LV (2023) Impact of explainable ai on cognitive load: Insights from an empirical study. *European Conference on Information Systems (ECIS)* (Kristiansand, Norway).

Hollender N, Hofmann C, Deneke M, Schmitz B (2010) Integrating cognitive load theory and concepts of human–computer interaction. *Computers in Human Behavior* 26(6):1278–1288.

Huang X, Fang Q, Qian S, Sang J, Li Y, Xu C (2019) Explainable interaction-driven user modeling over knowledge graph for sequential recommendation. *Proceedings of ACM International Conference on Multimedia*.

Jain S, Wallace BC (2019) Attention is not explanation. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

King MF, Bruner GC (2000) Social desirability bias: A neglected aspect of validity testing. *Psychology & Marketing* 17(2):79–103.

Koren Y (2008) Factorization meets the neighborhood: a multifaceted collaborative filtering model. *Proceedings of the ACM SIGKDD International Conference on Knowledge discovery and data mining*, 426–434.

Koren Y, Bell R, Volinsky C (2009) Matrix factorization techniques for recommender systems. *Computer* 42(8):30–37.

Koren Y, Sill J (2013) Collaborative filtering on ordinal user feedback. *International joint conference on artificial intelligence*.

Krumpal I (2013) Determinants of social desirability bias in sensitive surveys: A literature review. *Quality & Quantity* 47(4):2025–2047, URL `http://dx.doi.org/10.1007/s11135-011-9640-9`.

Kunkel J, Donkers T, Michael L, Barbu CM, Ziegler J (2019) Let me explain: Impact of personal and impersonal explanations on trust in recommender systems. 1–12.

Lee D, Cheng Z, Mao C, Manzoor E (2025) Guided diverse concept miner (gdcm): Uncovering relevant constructs for managerial insights from text. *Information systems research* 36(1):370–393.

Lee K, Ram S (2024) Explainable deep learning for false information identification: An argumentation theory approach. *Information Systems Research* 35(2):890–907.

Lei Y, Lian J, Yao J, Huang X, Lian D, Xie X (2024) Recexplainer: Aligning large language models for explaining recommendation models. *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1530–1541.

Leng Y, Dong X, Pentland A (2020) Learning quadratic games on networks. *International Conference on Machine Learning (ICML)* .

Li J, Zhang W, Wang T, Xiong G, Lu A, Medioni G (2023a) Gpt4rec: A generative framework for personalized recommendation and user interests interpretation. *arXiv preprint arXiv:2304.03879* .

Li L, Chen L, Dong R (2021) CAESAR: context-aware explanation based on supervised attention for service recommendations. *Journal of Intelligent Information Systems* (1):147–170.

Li L, Zhang Y, Chen L (2023b) Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems* 41(4):1–26.

Li L, Zhang Y, Chen L (2023c) Prompt distillation for efficient llm-based recommendation. *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 1348–1357.

Li P, Castelo N, Katona Z, Sarvary M (2024) Frontiers: Determining the validity of large language models for automated perceptual analysis. *Marketing Science* 43(2):254–266.

Li WJ, Yeung DY (2009) Relation regularized matrix factorization. *21ST International Joint Conference on Artificial Intelligence (IJCAI-09)*, 1126.

Li Z, Fang X, Bai X, Sheng ORL (2017) Utility-based link recommendation for online social networks. *Management Science* 63(6):1938–1952.

Liu J, Liu C, Zhou P, Lv R, Zhou K, Zhang Y (2023) Is chatgpt a good recommender? a preliminary study. *arXiv preprint arXiv:2304.10149* .

Lu H, Ma W, Wang Y, Zhang M, Wang X, Liu Y, Chua TS (2023) User perception of recommendation explanation: Are your explanations what users need? *ACM Transactions on Information Systems* 41(2):1–31.

Lundberg SM, Lee SI (2017) A unified approach to interpreting model predictions. *Advances in neural information processing systems* 30.

McInerney J, Lacker B, Hansen S, Higley K, Bouchard H, Gruson A, Mehrotra R (2018) Explore, exploit, and explain: personalizing explainable recommendations with bandits. *Proceedings of the 12th ACM Conference on Recommender Systems*, 31–39, RecSys '18 (Association for Computing Machinery).

McPherson M, Smith-Lovin L, Cook JM (2001) Birds of a feather: Homophily in social networks. *Annual review of sociology* 415–444.

Monti F, Bronstein M, Bresson X (2017) Geometric matrix completion with recurrent multi-graph neural networks. *Advances in Neural Information Processing Systems*, 3697–3707.

Ni J, Li J, McAuley J (2019) Justifying recommendations using distantly-labeled reviews and fine-grained aspects. *Proceedings on empirical methods in natural language processing (EMNLP-IJCNLP)*, 188–197.

Nunes I, Jannach D (2017) A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction* 27:393–444.

Pan D, Li X, Li X, Zhu D (2021) Explainable recommendation via interpretable feature mapping and evaluation of explainability. *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence* 2690–2696.

Pancras J, Sriram S, Kumar V (2012) Empirical investigation of retail expansion and cannibalization in a dynamic environment. *Management Science* 58(11):2001–2018.

Park H, Jeon H, Kim J, Ahn B, Kang U (2017) Uniwalk: Explainable and accurate recommendation for rating and network data. *arXiv preprint arXiv:1710.07134* .

Peng Y, Chen H, Lin CS, Huang G, Hu J, Guo H, Kong B, Hu S, Wu X, Wang X (2024) Uncertainty-aware explainable recommendation with large language models. *2024 International Joint Conference on Neural Networks (IJCNN)*, 1–8 (IEEE).

Perloff RM (1993) Third-person effect research 1983–1992: A review and synthesis. *International Journal of Public Opinion Research* 5(2):167–184, URL `http://dx.doi.org/10.1093/ijpor/5.2.167`.

Ragodos R, Wang T, Feng L, et al. (2024) From model explanation to data misinterpretation: Uncovering the pitfalls of post hoc explainers in business research. *arXiv preprint arXiv:2408.16987* .

Rahdari B, Ding H, Fan Z, Ma Y, Chen Z, Deoras A, Kveton B (2024) Logic-scaffolding: Personalized aspect-instructed recommendation explanation generation using llms. *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 1078–1081.

Rao N, Yu HF, Ravikumar PK, Dhillon IS (2015) Collaborative filtering with graph information: Consistency and scalable methods. *Advances in Neural Information Processing Systems (NIPS)*.

Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L (2009) Bpr: Bayesian personalized ranking from implicit feedback. *Proceedings of Conference on Uncertainty in Artificial Intelligence*, 452–461.

Rendle S, Zhang L, Koren Y (2019) On the difficulty of evaluating baselines: A study on recommender systems. *arXiv preprint arXiv:1905.01395* .

Rudin C (2019) Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1(5):206–215.

Seo S, Huang J, Yang H, Liu Y (2017) Interpretable convolutional neural networks with dual local and global attention for review rating prediction. *Proceedings of ACM Conference on Recommender Systems*, 297–305, RecSys '17 (New York, NY, USA: Association for Computing Machinery).

Serrano S, Smith NA (2019) Is attention interpretable? *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Sharma K, Lee YC, Nambi S, Salian A, Shah S, Kim SW, Kumar S (2024) A survey of graph neural networks for social recommender systems. *ACM Computing Surveys* 56(10):1–34.

Shimizu R, Matsutani M, Goto M (2022) An explainable recommendation framework based on an improved knowledge graph attention network with massive volumes of side information. *Knowledge-Based Systems* 239:107970.

Sun P, Wu L, Zhang K, Fu Y, Hong R, Wang M (2020) Dual learning for explainable recommendation: Towards unifying user preference prediction and review generation. 837–847.

Sun Y, Pan Z, Shen L (2008) Understanding the third-person perception: Evidence from a meta-analysis. *Journal of Communication* 58(2):280–300.

Tintarev N, Masthoff J (2010) Designing and evaluating explanations for recommender systems. *Recommender systems handbook*, 479–510 (Springer).

Tintarev N, Masthoff J (2012a) Beyond explaining single item recommendations. *Recommender Systems Handbook*, 711–756 (Springer).

Tintarev N, Masthoff J (2012b) Evaluating the effectiveness of explanations for recommender systems: Methodological issues and empirical studies on the impact of personalization. *User Modeling and User-Adapted Interaction* 22:399–439.

Tonmoy S, Zaman S, Jain V, Rani A, Rawte V, Chadha A, Das A (2024) A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313* .

Tourangeau R, Yan T (2007) Sensitive questions in surveys. *Psychological Bulletin* 133(5):859–883, URL `http://dx.doi.org/10.1037/0033-2909.133.5.859`.

Veličković P, Cucurull G, Casanova A, Romero A, Lio P, Bengio Y (2018) Graph attention networks. *International Conference on Learning Representations (ICLR)*.

Wang N, Wang H, Jia Y, Yin Y (2018a) Explainable recommendation via multi-task learning in opinionated text data. *The 41st international ACM SIGIR conference on research & development in information retrieval*, 165–174.

Wang S, Hu L, Wang Y, He X, Sheng QZ, Orgun MA, Cao L, Ricci F, Yu PS (2021) Graph learning based recommender systems: A review. *International Joint Conference on Artificial Intelligence (IJCAI)* .

Wang W, Benbasat I (2007) Recommendation agents for electronic commerce: Effects of explanation facilities on trusting beliefs. *Journal of Management Information Systems* 23(4):217–246.

Wang X, He X, Cao Y, Liu M, Chua TS (2019a) KGAT: Knowledge graph attention network for recommendation. *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 950–958.

Wang X, He X, Feng F, Nie L, Chua TS (2018b) Tem: Tree-enhanced embedding model for explainable recommendation. *Proceedings of the 2018 World Wide Web Conference*, 1543–1552.

Wang X, He X, Wang M, Feng F, Chua TS (2019b) Neural graph collaborative filtering. *Proceedings of ACM SIGIR conference on Research and development in Information Retrieval*, 165–174.

Wang Y, Jiang Z, Chen Z, Yang F, Zhou Y, Cho E, Fan X, Lu Y, Huang X, Yang Y (2024) Recmind: Large language model powered agent for recommendation 4351–4364.

Wiegreffe S, Pinter Y (2019) Attention is not not explanation. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*.

Wu L, Zheng Z, Qiu Z, Wang H, Gu H, Shen T, Qin C, Zhu C, Zhu H, Liu Q, et al. (2024) A survey on large language models for recommendation. *World Wide Web* 27(5):60.

Wu Q, Zhang H, Gao X, He P, Weng P, Gao H, Chen G (2019) Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. *World Wide Web*, 2091–2102.

Xian Y, Fu Z, Zhao H, Ge Y, Chen X, Huang Q, Geng S, Qin Z, De Melo G, Muthukrishnan S, et al. (2020) Cafe: Coarse-to-fine neural symbolic reasoning for explainable recommendation. *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 1645–1654.

Yao S, Yu D, Zhao J, Shafran I, Griffiths T, Cao Y, Narasimhan K (2024) Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems* 36.

Zhang J, Curley SP (2018) Exploring explanation effects on consumers' trust in online recommender agents. *International Journal of Human–Computer Interaction* 34(5):421–432.

Zhang M, Chen Y (2020) Inductive matrix completion based on graph neural networks. *International Conference on Learning Representations (ICLR)* .

Zhang Y, Chen X, et al. (2020) Explainable recommendation: A survey and new perspectives. *Foundations and Trends® in Information Retrieval* 14(1):1–101.

Zhang Y, Lai G, Zhang M, Zhang Y, Liu Y, Ma S (2014) Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, 83–92.

Zhou T, Wang Y, Yan L, Tan Y (2023) Spoiled for choice? personalized recommendation for healthcare decisions: A multiarmed bandit approach. *Information Systems Research* 34(4):1493–1512.

Zhu Y, Xian Y, Fu Z, De Melo G, Zhang Y (2021) Faithfully explainable recommendation via neural logic reasoning. *arXiv preprint arXiv:2104.07869* .

Acknowledgments

This research was supported by the Marketing Science Institute.

Interpretable Recommendations and Parameter-Grounded Explanations with Multi-Graph Attention

Appendix Contents

Appendix A. Literature Comparisons

A.1. Evaluation Quality Metrics

Table A1 summarizes the six explanation quality metrics introduced by Tintarev and Masthoff (2012a,b). We exclude scrutability because its assessment requires an interactive system in which users can modify inferred preferences and evaluate whether the system updates its recommendations in response, which is outside the scope of our experimental design.

Metric	Explanation
Effectiveness	This explanation helps me determine how well I like this business.
Efficiency	This explanation helps me decide faster if I will like this business.
Persuasiveness	This explanation makes me want to visit this business.
Satisfaction	This explanation would improve how easy it is to pick a recommendation.
Transparency	This explanation helps me understand what the recommendation is based on.
Trust	This explanation helps me trust the recommendation.

Table A1 Evaluation Metrics for Explanations Proposed by Tintarev and Masthoff (2012b)

A.2. Comparison with Benchmarks

We summarize the input data and the characteristics of different benchmarks in Table A2.

	Data		Method characteristics	
	Auxiliary	**Network**	**Interpretability**	**Cold-start (only auxiliary information is available)**
SVD++ (Koren 2008)	N	N	N	N
NNMF (Dziugaite and Roy 2015)	N	N	N	N
F-EAE (Hartford et al. 2018)	N	N	N	N
IGMC (Zhang and Chen 2020)	N	N	N	N
sRGCNN (Monti et al. 2017)	N	Y	N	N
GraphRec (Fan et al. 2019)	N	Y	N	N
DGAN (Wu et al. 2019)	Y	Y	Y	N
NGCF (Wang et al. 2019b)	N	Y	N	N
GC-MC (Berg et al. 2018)	Y	N	N	Y
GRALS (Rao et al. 2015)	Y	Y	N	Y
IFM (Pan et al. 2021)	Y	N	Y	Y
KGAT (Shimizu et al. 2022)	Y	N	Y	Y
HGCL (Chen et al. 2023)	Y	Y	N	Y
LightGCL (Cai et al. 2023)	N	N	N	N
MG-GAT	Y	Y	Y	Y

Table A2 Input data and characteristics of different benchmarks. Note: In this table, "Network" indicates explicit user–user or business–business graphs beyond the user–item interaction graph; knowledge-graph/entity relations used by KG-based recommenders are counted under "Auxiliary".

A.3. Comparisons with the ERS Literature

We compare our paper with the ones in the ERS literature in Table A3.

Paper	Large language model	User study	Number of Subjects	Category	Network data	Auxiliary information	Metrics
Rahdari et al. (2024)	Yes	Human labeler	None	LLM	No	Yes	No
Liu et al. (2023)	Yes	Human labeler	3	LLM	No	Yes	BLEU; ROUGE
Li et al. (2023b)	Yes	Case study	None	LLM	No	Yes	BLEU; ROUGE Incorrect classification Correct classification, insufficient explanation;
Lei et al. (2024)	Yes	Case study	3	LLM	No	Yes	Correct classification acceptable explanation; Correct classification, satisfying explanation
Li et al. (2023a)	Yes	Case study	None	LLM	No	Yes	No BLEU; ROUGE
Wang et al. (2024)	Yes	No	None	LLM	No	No	BLEU; ROUGE
Peng et al. (2024)	Yes	No	None	LLM	No	Yes	BLEU; ROUGE
Gao et al. (2023)	Yes	Case study	None	LLM	No	No	No
Geng et al. (2022)	Yes	No	None	LLM	No	No	BLEU; ROUGE
Hada and Shevade (2021)	Yes	No	None	LLM	No	No	BLEU
Ni et al. (2019)	Yes	Controlled experiment	≥ 3	LLM	No	No	BLEU; human (Relevance; Informativeness; Diversity)
Geng et al. (2023)	Yes	No	None	LLM	No	Yes	BLEU; ROUGE
Gao et al. (2024)	Yes	Case study	None	LLM	No	No	Aspect score; Rating score
Lu et al. (2023)	No	Controlled experiment	39	NA	No	No	BLEU; ROUGE
Shimizu et al. (2022)	No	No	None	Knowledge graph paths	No	Yes	No
Li et al. (2021)	No	Case study; Human labeler	10	Pre-defined templates	No	No	No
Zhu et al. (2021)	No	Controlled experiment	20	Knowledge graph paths	No	No	No
Pan et al. (2021)	No	No	None	Item features	No	Yes	General preferences; Specific preferences
Sun et al. (2020)	No	Case study	None	Knowledge graph paths	No	No	BLEU; ROUGE
Huang et al. (2019)	No	No	None	Knowledge graph paths	Yes	Yes	No
Xian et al. (2020)	No	Case study	None	Knowledge graph paths	Yes	Yes	No
Wang et al. (2018b)	No	No	None	Item features	No	Yes	No
McInerney et al. (2018)	Yes	Controlled experiment	None	Item features	No	Yes	No
Chen et al. (2018a)	No	No	None	Ranked text	No	No	No
Chen et al. (2018b)	No	No	None	Deep learning	No	No	No
Seo et al. (2017)	No	No	None	Deep learning	No	No	No
Park et al. (2017)	No	No	None	Graph-based social explanation	Yes	No	No
Our paper	Yes	Controlled experiment	1363	LLM guided by model parameters	Yes	Yes	Perceived relevance; Future interest; Confidence/Trust; Transparency; Effectiveness; Efficiency; Persuasiveness; Satisfaction

Table A3 Paper Comparisons in the Explainable Recommender Systems Literature

A.4. Comparative Analysis with Attention- and KG-based recommenders

Table A4 situates MG-GAT relative to representative attention-based and knowledge-graph (KG) recommenders along four explanation-relevant dimensions that are central to our framing in Section 2. First, we distinguish the underlying relational structure each method operates on (single homogeneous graph, multi-graph/user–item plus social graph, or an explicit KG). Second, we record what is actually exposed as an explanation signal (e.g., attention weights over latent states, explicit KG paths, or text salience). Third, we indicate whether the method provides an explicit feature-level decomposition of attention or importance scores into observable attributes (as opposed to weights over latent embeddings). Fourth, we indicate whether any user-facing explanation text is *parameter-grounded* in prediction-time model signals, as defined in Section 4.

The goal of this table is not to exhaustively catalogue all variants, but to clarify how MG-GAT combines several known components into an interpretable interface: it exposes NIG as tie-level salience and FR as an attribute-level decomposition of neighbor importance, and it reuses these prediction-time signals as control inputs for Stage 2 explanation generation. This comparison motivates why NIG/FR can serve as auditable signals for both model interpretation and explanation generation.

Table A4 Comparison of representative attention- and KG-based recommenders with MG-GAT along explanation-relevant dimensions.

Method family (examples)	**Graph structure**	**Exposed explanation signal**	**Feature-level decomposition?**	**Parameter-grounded narrative?**
GAT-style attention over a single graph (Veličković et al. 2018)	single homogeneous graph	attention weights over latent node states	No	No
Dual-/social-side graph attention recommenders (Wu et al. 2019, Fan et al. 2019)	user–item + social-side graph	latent attention weights (typically over embeddings)	No	No
KG attention recommenders (e.g., KGAT and variants) (Wang et al. 2019a, Shimizu et al. 2022)	knowledge graph	attention weights / path scores over KG embeddings (often visualized)	No	No
Path-based KG explanation methods (Huang et al. 2019, Xian et al. 2020, Pan et al. 2021)	knowledge graph	explicit multi-hop reasoning paths	No (path-level, not feature-level)	No (paths/visualizations rather than parameter-grounded text)
Review-/text-attention explainers (Chen et al. 2018a, Li et al. 2021)	user–item interactions + text	salient words, reviews, or aspects	No (text salience, not attention-logit decomposition)	No (no explicit grounding in prediction-time parameters)
MG-GAT (this paper)	user–user + business–business graphs (multi-graph)	NIG (tie salience) + FR (attribute contributions)	**Yes** (FR linear decomposition)	**Yes** (Stage 2 narratives grounded in NIG/FR)

A.5. Compare the Attention Mechanisms With Relevant Methods in the RS Literature

Our method is similar in some respects to graph regularized matrix factorization (GRMF) and also shares similarities with regression and graph convolutional network (GCN). We discuss here the similarities and the major differences.

Graph Regularized Matrix Factorization. Our framework extends GRMF and similarly maps neighboring users and nearby businesses in a latent space (Cai et al. 2010). However, the major difference is that our framework integrates auxiliary information on nodes (i.e., users and business) and the connections between nodes in a selective fashion. We use auxiliary information to remove noisy links and perform localized aggregation on embeddings (i.e., not all network neighbors are equally predictive). Our architecture enforces local smoothness on the embeddings; hence, our method is more realistic and flexible (due to the reasoning provided in Section 3.1).

Linear Regression. The learning of FR (Def. 2) bears conceptual similarity to linear regression; in particular, the former can be viewed as a regression where the "dependent variable" is the neighbor importance from the neural network architecture. However, a simple regression cannot be performed since the dependent variable is not observed. In this case, we use the final rating to guide learning of the neighbor-importance signal and impose a linear structure, similar to linear regression, to maintain explainability.

Graph Convolutional Network. We compare the network aggregation component of our method with a GCN, one of the most popular geometric deep learning layers. GCN is developed from a spectral graph filtering perspective, where the layer-wise propagation matrix comes essentially from a degree-one polynomial of the graph Laplacian matrix. This approach makes the smoothing or regularization in GCN a global operation due to the global nature of the graph Fourier transform. In comparison, GAT can be interpreted more as a local smoothing framework, offering more flexibility. In GCN, the weight matrix used for layer-wise propagation is $\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{G}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$, where $\tilde{\mathbf{G}} = \mathbf{G} + \mathbf{I}$ and $\mathbf{I}$ is an identity matrix and $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{G}}_{ij}$. The adjacency matrix of the graph, $\mathbf{G}$, is fixed and non-adaptive. In the case of an unweighted graph (where all edges have unitary weight), GCN leads to an equal weighting of all neighbors for each node. In comparison, an attention mechanism allows for adapting the weights of the neighbors to the focal node, which results in a more flexible and complex neighborhood structure that has higher predictive power. Our study's attention mechanism leads to performance and interpretability advantages. In addition, GCN is an inherently transductive framework, while GAT can be used in cold-start settings to make predictions for new nodes (users or businesses) based on the auxiliary information.

More specifically, attending over only observed networks rather than all nodes has the following advantages:

- Graph Structure as Implicit Priors: The graph structures in attention networks serve as implicit priors by encoding relational information and representations of neighboring nodes. This graph structure imposes the following constraints on the parameter learning process:
  - —Locality Prior: The graph structure used in GAT inherently assumes that the information needed for a node's representation can be found in its local neighborhood. This locality prior means that the model relies on the immediate or a few-hop neighbors of a node to aggregate information, which makes the model more focused on relevant information encoded in the network.

—Homophily and Smoothness Prior: GAT implicitly assumes that connected nodes in observed networks are likely to share similar features or belong to the same class. This smoothness constraint encourages neighboring nodes to have similar representations. During the message-passing process, node features are aggregated and updated based on their neighbors, ensuring that the model captures meaningful relationships.

—Structural Role Prior: GAT implicitly imposes priors on the structural roles of nodes based on their position and connections within the graph. Nodes with similar structural roles (e.g., hubs, bridges) tend to have similar representations, regardless of their feature similarities. This helps the model generalize across different parts of the network by learning roles rather than just node-level features.

- Avoiding Overfitting: Including all users and businesses as neighbors without any selection criterion can lead to overfitting, especially when the dataset contains a large number of irrelevant or noisy connections. By focusing on neighbors identified through meaningful networks (e.g., friendship and business networks), GAT helps reduce the complexity of the model and prevents it from learning from noise. This selective neighbor inclusion acts as a natural regularizer, ensuring that the model learns from relevant interactions rather than overfitting all possible connections.

Appendix B. Key Notations

We summarize the key notations used in this paper in Table B1. For notational consistency, we use the lower-case, bold-lower-case, bold-capital case to represent scalar, vector, and matrix, respectively. We use subscript or superscript u and b to represent **u**sers and **b**usinesses, respectively.

Table B1 Key notations

Notations	Definitions and Descriptions
$\mathbf{X} \in \mathbb{R}^{n \times m}$	User-business rating matrix with n users and m businesses
$\hat{\mathbf{X}} \in \mathbb{R}^{n \times m}$	Predicted user-business rating matrix
$\mathbf{U} \in \mathbb{R}^{n \times k_f}, \mathbf{B} \in \mathbb{R}^{m \times k_f}$	Inferred latent user and business embedding with dimension k_f
$\mathbf{S}^{(u)} \in \mathbb{R}^{n \times s_u}, \mathbf{S}^{(b)} \in \mathbb{R}^{m \times s_b}$	Auxiliary information about users and businesses with dimensions s_u and s_b
$\mathbf{G}_u, \mathbf{G}_b$	User friendship network and business network
$\mathbf{L}_u, \mathbf{L}_b$	Graph Laplacian of the user friendship network and business network
N_i^u, N_j^b	Neighbor set of user i or business j on the user or business graph
$\alpha_u^{k \to i}, \alpha_b^{l \to j}$	Neighbor importance for user and business
NIG_θ	Neighbor Importance Graph: attention weights over $\mathbf{G}_u$ and $\mathbf{G}_b$ used for prediction and explanations
FR_θ	Feature Relevance coefficient vectors $\{\text{FR}^u_{\text{self}}, \text{FR}^u_{\text{nb}}, \text{FR}^b_{\text{self}}, \text{FR}^b_{\text{nb}}\}$ decomposing attention logits over observable attributes
k_f	Dimension of latent user/business embeddings
K, F	Number of high-salience neighbors (top-K) and features (top-F) exposed to Stage 2
$S_\theta(u, j)$	High-salience neighbor set for user u and recommended item j selected via NIG_θ
$\text{score}(f)$	NIG-weighted instance-level contribution score used to select salient features for Stage 2
$\mathcal{F}_j^*$	High-salience feature set for business j selected by the top-F $\text{score}(f)$ values
$\mathbf{a}_{u,\text{self}}$ $(\mathbf{a}_{u,\text{nb}}), \mathbf{a}_{b,\text{self}}$ $(\mathbf{a}_{b,\text{nb}})$	Coefficients to compute feature relevance of focal (neighbor) nodes
$\Omega_{\text{training}} \in \mathbb{R}^{n \times m}$ and $\Omega_{\text{test}} \in \mathbb{R}^{n \times m}$	Indicator matrix representing entries for the training and test set
$\mathcal{R}(\cdot)$	Regularization term
$\mathcal{L}$	Loss function
$\mathbf{W}_u^{(q)}$ and $\mathbf{W}_b^{(q)} \forall q \in \{1, 2, 3\}, \mathbf{W}_{us}^{(2)}, \mathbf{W}_{bs}^{(2)}$	Learnable weights in the deep learning layer
$\mathbf{H}_u^{(q)}$ and $\mathbf{H}_b^{(q)}$ $\forall q \in \{1, 2, 3, 4\}$	Intermediate embeddings for users and businesses
$\mathbf{b}_u^{(1)} \in \mathbb{R}^{d_u^1}, \mathbf{b}_u^{(x)} \in \mathbb{R}^n, \mathbf{b}_b^{(1)} \in \mathbb{R}^{d_b^1}, \mathbf{b}_b^{(x)} \in \mathbb{R}^m, b_x \in \mathbb{R}$	Learnable bias term, where d_1^u and d_1^b are the size of the latent dimension

Appendix C. Stage-2 Implementation Details

C.1. Cost and Deployment Considerations

The main deployment overhead of our two-stage design is Stage 2 generation, not Stage 1 scoring. Stage 1 inference is comparable to other GNN recommenders and can be batched and cached; Stage 2 invokes an LLM multiple times (Algorithm 2). Because LLM latency varies by provider, hardware, and load, we report the stable quantities that determine runtime and cost: the number of model calls and the (approximate) output-length targets, and we emphasize standard mitigations such as on-demand explanation, caching, and batching.

Deployment configurations. Depending on cost/latency constraints, Stage 2 can be instantiated at different fidelity levels while preserving the same evidence contract (top-K neighbors and top-F features from NIG/FR):

- **Template mode (zero-cost fallback).** Deterministic templates verbalize NIG/FR without LLM calls.
- **Fast mode.** A single constrained generation call produces an explanation with the `EVIDENCE_USED` checklist; no plan search and no critic scoring. This reduces latency but provides weaker filtering against drift than full plan-and-critique.
- **Full mode.** Algorithm 2 performs plan-and-critique search with NIG-weighted critics to select among multiple candidates, improving controllability and reducing contextual misattribution at the cost of additional calls.

Table C1 Stage 2 cost accounting by deployment mode (per recommendation). Let $K = |S_\theta(u, j)|$ be the number of neighbors surfaced and let N_p/N_e be the number of candidate plans/explanations.

Mode	# LLM calls	Notes
Template mode	0	Deterministic surface realization; strongest cost/latency control.
Fast mode	1	One constrained explanation; evidence checklist; no critics.
Full mode	$1 + KN_p + N_e + KN_e$	Plan-and-critique with NIG-weighted critic scoring and selection (Algorithm 2).

Table C2 Illustrative Stage 2 call budget (per recommendation) under Algorithm 2. Let $K = |S_\theta(u, j)|$ be the number of neighbors surfaced and let N_p/N_e be the number of candidate plans/explanations generated.

Component	# LLM calls	Output-length target (approx.)
Plan generation	1	Short plan paragraph
Plan evaluation (critics)	$K \cdot N_p$	One paragraph ending with six metric scores
Explanation generation	N_e	2–3 sentences; ∼60–68 words (∼80–95 tokens)
Explanation evaluation (critics)	$K \cdot N_e$	One paragraph ending with six metric scores

C.2. Prompt Templates

For reproducibility, we provide the exact prompt templates used in Stage 2. These templates are parameter-grounded: all neighbor and feature content is selected via NIG/FR, and no additional context is injected. In addition, we include an `EVIDENCE_USED` checklist line in the plan and explanation prompts to enable a lightweight, deterministic support check; this line is removed before user-facing display.

Prompt 1 (General Prompt Provided to All LLM Agents)

Here is the dining history of a customer, including only the restaurants they have highly rated:

- *Neighbor $k(\in\{1,2,3\})$: {restaurant name} described with highly relevant features $\{F_{N_k}$ by FR$\}$.*

Based on this user's visiting history, the machine learning algorithm recommends the following restaurant:

- *Focus business: {restaurant name} with relevant features $\{F_f$ according to FR$\}$.*

Prompt 2 (Plan Generation)

You are a restaurant recommender system. {Provide the general prompt in Prompt 1.}

Step 1 (evidence checklist). *Output exactly one line in the form:*

`EVIDENCE_USED: neighbors=[Neighbor 1, Neighbor 2, Neighbor 3]; features=[...]`

List only neighbors and feature names that appear in the prompt. Do not introduce any restaurant or feature that is not provided above.

Step 2 (plan). *Write a short plan (one paragraph) for the explanation that will be shown to the consumer, linking the recommendation to the visited restaurants using only the listed evidence.*

Prompt 3 (Evaluation Prompt)

Consider you are a fan of {name of neighboring restaurants}. {Provide the general prompt in Prompt 1.} Please evaluate the explanation (plan) for the recommendation based on the following criteria: If the plan/explanation mentions any restaurant or attribute that does not appear in the prompt, treat it as unsupported and assign very low scores (especially transparency and trust). Provide a brief analysis and end the paragraph with exactly this sentence with the ratings: "Thus, effectiveness score is {s}, efficiency score is {s}, persuasiveness score is {s}, satisfaction score is {s}, transparency score is {s}, and trust score is {s}."

Prompt 4 (Explanation Generation)

You are a restaurant recommender system. {Provide the general prompt in Prompt 1.}

Step 1 (evidence checklist; internal). *Output exactly one line in the form:*

`EVIDENCE_USED: neighbors=[Neighbor 1, Neighbor 2, Neighbor 3]; features=[...]`

List only neighbors and feature names that appear in the prompt. Do not introduce any restaurant or feature that is not provided above.

Step 2 (user-facing text). *Then generate a paragraph of explanation (in K sentences) for the recommended restaurant to be shown to the consumer so that the consumer can understand the reasons behind the recommendation (relate to the customer's visited restaurants). The* `EVIDENCE_USED` *line will be removed before display.*

Appendix D. Proofs

D.1. Proof for Proposition 1

Proof for Proposition 1 From Eq. (2), the first-layer embedding is linear in the observed attributes: $H^{(1)} = W^{(1)}x$. Writing the attention logit in Eq. (3) as the sum of a focal term and a neighbor term and substituting $H^{(1)}$ yields

$$e_{k\to i} = a_{\text{self}}^{\top} H_i^{(1)} + a_{\text{nb}}^{\top} H_k^{(1)} = a_{\text{self}}^{\top} W^{(1)} x_i + a_{\text{nb}}^{\top} W^{(1)} x_k.$$

By Definition 2, $\text{FR}_{\text{self}} = (a_{\text{self}}^{\top} W^{(1)})^{\top}$ and $\text{FR}_{\text{nb}} = (a_{\text{nb}}^{\top} W^{(1)})^{\top}$, so

$$e_{k\to i} = \text{FR}_{\text{self}}^{\top} x_i + \text{FR}_{\text{nb}}^{\top} x_k.$$

Taking the difference between neighbors k and k' cancels the focal term and gives

$$e_{k\to i} - e_{k'\to i} = \text{FR}_{\text{nb}}^{\top}(x_k - x_{k'}),$$

as claimed. □

D.2. Proof for Proposition 2

Proof for Proposition 2 By construction, the attention weights $\alpha_{k\to i}$ used in Eq. (4) are obtained by applying a softmax to the logits $e_{k\to i}$, so $\alpha_{k\to i} \geq 0$ and $\sum_{k\in\mathcal{N}_i} \alpha_{k\to i} = 1$ for each i. Eq. (4) sets $H_i^{(2)} = \sum_{k\in\mathcal{N}_i} \alpha_{k\to i} H_k^{(1)}$, which is a convex combination of the neighbor embeddings. Therefore $H_i^{(2)}$ lies in the convex hull of $\{H_k^{(1)} : k \in \mathcal{N}_i\}$. □

D.3. Proof for Proposition 3

Proof for Proposition 3 Algorithm 2 takes as input MG-GAT's prediction-time NIG signals, the global FR coefficient vectors $\{\text{FR}_{\text{self}}^{u}, \text{FR}_{\text{nb}}^{u}, \text{FR}_{\text{self}}^{b}, \text{FR}_{\text{nb}}^{b}\}$, and the observed attributes of the recommended business x_j and the selected neighbors $X(S_\theta(u,j))$. The plan-selection step deterministically chooses a subset of neighbors and features from these inputs (using the NIG-weighted instance-level contribution score $\text{score}(f)$), and the prompt-construction step includes only this selected information. Critics then score candidate plans and explanations under a fixed decoding configuration. The only randomness arises from the LLM's internal sampling given this fixed configuration. Therefore, there exists a (possibly randomized) mechanism Φ_θ mapping $\big(\text{NIG}_\theta^u(u,\cdot), \text{NIG}_\theta^b(:,j), \text{FR}_{\text{self}}^{u}, \text{FR}_{\text{nb}}^{u}, \text{FR}_{\text{self}}^{b}, \text{FR}_{\text{nb}}^{b}, x_j, X(S_\theta(u,j))\big)$ to the distribution of $E(u,j)$, so $E(u,j)$ is parameter-grounded by Definition 3. □

D.4. Proof for Corollary 1

Proof for Corollary 1 Because θ is fixed, the global FR coefficient vectors $\{\text{FR}_{\text{self}}^{u}, \text{FR}_{\text{nb}}^{u}, \text{FR}_{\text{self}}^{b}, \text{FR}_{\text{nb}}^{b}\}$ are identical across all recommendation instances. Therefore, if the tuples stated in Corollary 1 coincide for (u,j) and (u',j'), then the full input tuples to Φ_θ in Proposition 3 also coincide. By Definition 3 and Proposition 3 we have

$$E(u,j) \sim \Phi_\theta(\cdot) \quad \text{and} \quad E(u',j') \sim \Phi_\theta(\cdot)$$

with the same argument. Therefore $E(u,j)$ and $E(u',j')$ have the same distribution, i.e., $E(u,j) \overset{d}{=} E(u',j')$.

D.5. Proof for Proposition 4

Proof for Proposition 4 Algorithm 2 takes as input the recommended business j, the selected neighbor set $S_\theta(u, j)$ (from NIG), and the selected feature set $\mathcal{F}_j^*$ (top-F by the NIG-weighted instance-level contribution score score(f)). The prompts constructed for planning and explanation include only these entities and attributes. Moreover, we enforce the evidence contract explicitly: any candidate plan or explanation that cites restaurants or features outside $\{j\} \cup S_\theta(u, j)$ and $\mathcal{F}_j^*$ is rejected before scoring/selection. Therefore, explanations that survive selection are support-limited to $\{j\} \cup S_\theta(u, j)$ and $\mathcal{F}_j^*$, establishing the claimed support limitation. □

Appendix E. Summary Statistics of the Yelp Data

E.1. Analytic sample construction

We construct the Ontario (ON) and Pennsylvania (PA) datasets from the Yelp Open Dataset using the following reproducible filtering pipeline:

1. **Region filter.** Keep businesses whose `state` field equals `ON` (Canada) or `PA` (U.S.).
2. **Restaurant filter.** Keep businesses whose `categories` field contains the tag `Restaurants` (including multi-category businesses).
3. **Rating extraction.** Keep all user–business reviews with an observed star rating and a valid timestamp; if multiple reviews exist for the same $(user, business)$ pair, we keep the most recent review to avoid leakage across the time split.
4. **Entity inclusion.** Retain all users and businesses that appear at least once after the above steps (i.e., no additional minimum-interaction/k-core filtering).
5. **Auxiliary-information encoding.** Business attributes, categories, and hours are one-hot encoded (multi-label allowed). Missing attribute values are treated as "unknown/not present" and encoded as zeros, and missing check-in vectors are encoded as all zeros, so we do not drop businesses solely due to sparse auxiliary information.

The resulting analytic samples contain 706,998 (ON) and 260,350 (PA) observed ratings, summarized in Table E1.

E.2. Data Descriptions

We apply MG-GAT to the publicly available Yelp dataset (from 2009 to 2018), an online review platform where users may rate and post reviews on businesses (e.g., restaurants).[14] Yelp shares data from eleven states and provinces; among them, we chose two representative locations in two countries: Ontario (ON) in Canada and Pennsylvania (PA) in the United States. These regions encompass 135,173 and 76,865 users, and 32,393 and 10,966 businesses, respectively. We summarize the statistics in Table E1 and in Figure E1.

Business. Yelp's business information comprises self-uploaded data from business owners and user reviews. Their diversity and high dimensionality make the data ideal for evaluating the effectiveness of the proposed method. The model uses three types of auxiliary information about the businesses: basic information (attributes, categories, and operation hours); location data; and check-in information reflecting temporal popularity. We present the summary statistics of business features in Table E5–E6.

1. The basic information that Yelp collects consists of three elements: business attributes (amenities), business categories, and operation hours. The attribute information collected in Canada is different from that collected in the United States: It includes 84 attributes in ON and 93 attributes in PA. Business attributes include parking options, WiFi availability, and takeout service. Because most of the attributes are categorical variables, we adopt one-hot (i.e., one-of-K) encoding to indicate whether a business has a particular business attribute. The business category data (e.g., Mexican, burgers) include 953 and 946 categories in ON and

[14] The data was accessed via `https://www.yelp.com/dataset/`.

PA, respectively. In this second type of information, each business may belong to multiple categories. Thus, we similarly adopt the one-hot encoding on business categories. Meanwhile, the operation hours contain information about when businesses open and close on each day of the week.

2. The location data, in latitude and longitude, provides spatial information on businesses (Figure E2).

3. The check-in information from users allows us to analyze the temporal patterns of the popularity of the businesses. We aggregate the check-ins into 144 hourly bins in a week to obtain one vector for each business.

We construct a business network $\mathbf{G}_b$ based on four key conditions among its members: (1) geographical proximity, (2) consumer co-visitations, (3) shared business category, and (4) perceptual similarity derived from an LLM-based perceptual map.

Geographical proximity captures spatial dependencies between businesses, as recognized in the spatial econometrics literature. These dependencies may arise from positive demand spillovers, spatial competition, and cannibalization effects (Pancras et al. 2012). Co-visitation data provide additional predictive value, reflecting economic ties based on consumer behavior. Businesses in the same category or with similar attributes affect each other through shared consumer markets and competition, a principle adopted in prior studies (Grieser et al. 2021). Lastly, the LLM-based perceptual map approximates similarities in consumer perceptions and brand positioning, enhancing the network by identifying perceptually related businesses (Li et al. 2024)

Because the network created through these four mechanisms is dense, we construct a k-nearest neighbor graph for each mechanism to optimize memory and computational efficiency. A common approach is to set $k \sim \log(m)$, where m is the total number of businesses. Here, k is set at ten for each edge type (geographically close, consumer co-visits, business category, and perceptual map), ensuring no more than 40 edges per business. The resulting combination forms the business network $\mathbf{G}_b$.

User. User information can be categorized into two types: the basic metadata and friendships on Yelp. The basic metadata on users includes whether they are elite reviewers in a certain year and detailed compliments about their work.[15] This auxiliary information leads to 19 attributes for each user in ON and 33 attributes for each user in PA. Yelp also provides a list of each Yelp user's friends. With this information, we can build a friendship network, where a connection on the network indicates that the two users are friends.

Implicit features. Our model incorporates both explicit and implicit user feedback. Explicit feedback is derived from user ratings, while implicit feedback captures users' interactions with businesses. Implicit feedback assumes that users demonstrate stronger preferences for businesses they have engaged with. This assumption has been validated in several studies (Koren 2008, Chen et al. 2019) and through the Netflix competition (Rendle et al. 2019), where implicit feedback was shown to be highly predictive in RSs.

To generate implicit features, we binarize continuous user ratings to indicate whether a user-business interaction has occurred. To reduce complexity, we then apply singular value decomposition (SVD) on the binarized rating matrix. The resulting implicit feedback is represented as $\mathbf{X}_{\text{bina}} = \mathbf{U}^{(0)}\Sigma\mathbf{B}^{(0)}$, where $\mathbf{U}^{(0)} \in$

[15] The compliments include the number of "useful," "funny," and "cool" reviews; the number of fans; the number of compliments that describe reviews as being "hot," "cute," "plain," "cool," "funny," or "good writer"; and the number of compliments on the users' profile, lists, notes, photos, and other information.

$\mathbb{R}^{n\times k_i}$, $\Sigma \in \mathbb{R}^{k_i\times k_i}$, and $\mathbf{B}^{(0)} \in \mathbb{R}^{k_i\times m}$. Here, k_i represents the dimension of the implicit features and is treated as a hyperparameter.

Specifically, we have binarized the continuous user ratings by setting a threshold where any rating greater than zero (i.e., there exists a rating) is converted to a value of one ($X_{uv} > 0 \Rightarrow X_{\text{bina},uv} = 1$), and a rating of zero remains zero ($X_{uv} = 0 \Rightarrow X_{\text{bina},uv} = 0$). This binary feature therefore captures whether any interaction between the user and business occurred, regardless of the specific rating value.

The user and business implicit features are then calculated as $\mathbf{S}u, \text{imp} = \mathbf{U}^{(0)}\Sigma^{\frac{1}{2}} \in \mathbb{R}^{n\times k_i}$ and $\mathbf{S}_{b,\text{imp}} = \mathbf{B}^{(0)T}\Sigma^{\frac{1}{2}} \in \mathbb{R}^{m\times k_i}$, respectively. These implicit features are combined with explicit features to form the auxiliary information, $\mathbf{S}_u$ for users and $\mathbf{S}_b$ for businesses.

If the implicit features are found to be non-predictive, the parameters governing their influence—specifically, the dimensions in $\mathbf{W}_b^{(1)}$, $\mathbf{W}_u^{(1)}$, $\mathbf{W}_{bs}^{(2)}$, and $\mathbf{W}_{us}^{(2)}$—are regularized toward zero.

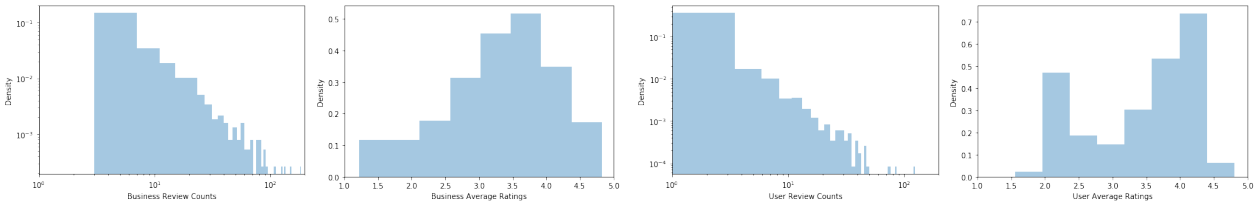
E.3. Experimental Setting

We split the rating data into training, validation, and test sets based on time. The data between 2009 and 2016 are used for training, the 2017 data are used for validation, and the 2018 data are used for testing. This sample split aligns with practical RSs to predict "future" ratings. We present the statistics for the three subsets in Table E2.

Table E1 Summary statistics of the data for Ontario and Pennsylvania.

Statistic	Ontario	Pennsylvania
Rating count	706,998	260,350
User count	135,173	76,865
Business count	32,393	10,966
Average rating (std. dev.)	3.556 (1.334)	3.728 (1.384)
Ratings per user (std. dev.)	5.230 (21.007)	3.387 (12.140)
Ratings per business (std. dev.)	21.826 (47.221)	23.742 (56.371)
Average degree per user (std. dev.)	7.077 (34.738)	5.557 (29.906)
Average degree per business	40	40

Note: The business network aggregates four k-NN components ($k = 10$ each): a spatial graph (nearest neighbors), a co-visitation graph (frequently co-visited businesses), a same-category graph, and a perceptual-similarity graph (LLM-assisted). This construction yields a fixed degree of 40 per business in the aggregated business network.

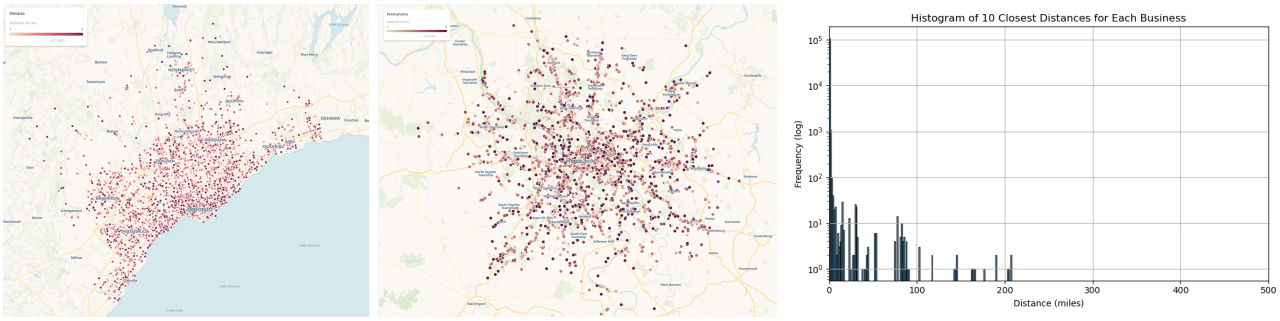


(a) business review counts (b) business average ratings (c) user review counts (d) user average ratings

Figure E1 Distributions of review counts and average ratings for businesses and users.

The location information, in terms of latitude and longitude, is shown in Figures E2a and E2b. Additionally, Figure E2c presents the spatial distances for the top 10 closest businesses.

We present the statistics about the training, validation, and test sets of the datasets in Table E2.



(a) Ontario (b) Pennsylvania (c) Distance distribution for top 10 closest businesses

Figure E2 **Spatial distributions of businesses in (a) and (b); the color code represents the average ratings of businesses. The spatial distributions between businesses and their ten closest businesses are shown in (c).**

Table E2 **Statistics about the training, validation and test set**

	Time period	Metrics	Ontario	Pennsylvania
Training	2009 - 2016	Average user rating	3.468 (1.373)	3.641 (1.403)
		Average business rating	3.419 (0.995)	3.604 (1.049)
Validation	2017	Average user rating	3.463 (1.479)	3.665 (1.501)
		Average business rating	3.401 (1.266)	3.614 (1.285)
Test	2018	Average user rating	3.469 (1.502)	3.667 (1.535)
		Average business rating	3.395 (1.314)	3.616 (1.373)

E.4. Detailed Summary Statistics on User and Business Auxiliary Information

User Data We present the auxiliary information after min-max normalization for ON in Table E3 and for PA in Table E4.

Business Data We present summary statistics for the business auxiliary information in Table E5.

Our model incorporates both explicit and implicit user feedback. Explicit feedback is derived from user ratings, while implicit feedback captures users' interactions with businesses. Implicit feedback assumes that users demonstrate stronger preferences for businesses they have engaged with. This assumption has been validated in several studies (Koren 2008, Chen et al. 2019) and through the Netflix competition (Rendle et al. 2019), where implicit feedback was shown to be highly predictive in recommendation systems.

To generate implicit features, we binarize continuous user ratings to indicate whether a user-business interaction has occurred. To reduce complexity, we then apply singular value decomposition (SVD) on the binarized rating matrix. The resulting implicit feedback is represented as $\mathbf{X}_{\text{bina}} = \mathbf{U}^{(0)}\Sigma\mathbf{B}^{(0)}$, where $\mathbf{U}^{(0)} \in \mathbb{R}^{n\times k_i}$, $\Sigma \in \mathbb{R}^{k_i\times k_i}$, and $\mathbf{B}^{(0)} \in \mathbb{R}^{k_i\times m}$. Here, k_i represents the dimension of the implicit features and is treated as a hyperparameter.

The user and business implicit features are then calculated as $\mathbf{S}u, \text{imp} = \mathbf{U}^{(0)}\Sigma^{\frac{1}{2}} \in \mathbb{R}^{n\times k_i}$ and $\mathbf{S}_{b,\text{imp}} = \mathbf{B}^{(0)T}\Sigma^{\frac{1}{2}} \in \mathbb{R}^{m\times k_i}$, respectively. These implicit features are combined with explicit features to form the auxiliary information, $\mathbf{S}_u$ for users and $\mathbf{S}_b$ for businesses.

If the implicit features are found to be non-predictive, the parameters governing their influence—specifically, the dimensions in $\mathbf{W}_b^{(1)}$, $\mathbf{W}_u^{(1)}$, $\mathbf{W}_{bs}^{(2)}$, and $\mathbf{W}_{us}^{(2)}$—are regularized toward zero.

Table E3 User Auxiliary Information After Min-Max Normalization (ON)

	Mean	Standard Deviation	25th Percentile	Median	75th Percentile
compliments: cool	0.000202	0.004825	0	0	0.000000
compliments: cute	0.000111	0.004247	0	0	0.000000
compliments: funny	0.000202	0.004825	0	0	0.000000
compliments: hot	0.000134	0.004046	0	0	0.000000
compliments: list	0.000068	0.003512	0	0	0.000000
compliments: more	0.000137	0.003703	0	0	0.000000
compliments: note	0.000461	0.008414	0	0	0.000000
compliments: photos	0.000118	0.004614	0	0	0.000000
compliments: plain	0.000441	0.009544	0	0	0.000000
compliments: profile	0.000064	0.003610	0	0	0.000000
compliments: writer	0.000271	0.005737	0	0	0.000000
votes: cool	0.000177	0.005336	0	0	0.000000
votes: funny	0.000225	0.006593	0	0	0.000000
votes: useful	0.000230	0.005714	0	0	0.000011
profile: fans	0.000860	0.009534	0	0	0.000000
profile: yelping_since_year	0.687068	0.174316	0.571429	0.714286	0.785714
profile: yelping_since_month	0.495953	0.310605	0.181818	0.545455	0.727273
profile: yelping_since_day	0.492608	0.294071	0.233333	0.500000	0.733333
profile: elite_2005	0.000089	0.009422	0	0	0.000000
profile: elite_2006	0.000577	0.024015	0	0	0.000000
profile: elite_2007	0.001339	0.036568	0	0	0.000000
profile: elite_2008	0.002367	0.048598	0	0	0.000000
profile: elite_2009	0.005230	0.072132	0	0	0.000000
profile: elite_2010	0.009092	0.094918	0	0	0.000000
profile: elite_2011	0.011822	0.108084	0	0	0.000000
profile: elite_2012	0.016668	0.128023	0	0	0.000000
profile: elite_2013	0.020159	0.140546	0	0	0.000000
profile: elite_2014	0.021521	0.145112	0	0	0.000000
profile: elite_2015	0.029022	0.167869	0	0	0.000000
profile: elite_2016	0.036316	0.187077	0	0	0.000000
profile: elite_2017	0.041347	0.199092	0	0	0.000000
profile: elite_2018	0.036094	0.186526	0	0	0.000000
profile: elite_None	0.933234	0.249618	1	1	1.000000

Table E4 User Auxiliary Information After Min-Max Normalization (PA)

	Mean	Standard Deviation	25th Percentile	Median	75th Percentile
compliments: cool	0.000190	0.005267	0	0	0.000000
compliments: cute	0.000117	0.004852	0	0	0.000000
compliments: funny	0.000190	0.005267	0	0	0.000000
compliments: hot	0.000235	0.006187	0	0	0.000000
compliments: list	0.000086	0.004366	0	0	0.000000
compliments: more	0.000168	0.004459	0	0	0.000000
compliments: note	0.000269	0.005434	0	0	0.000000
compliments: photos	0.000047	0.003772	0	0	0.000000
compliments: plain	0.000137	0.004489	0	0	0.000000
compliments: profile	0.000074	0.004027	0	0	0.000000
compliments: writer	0.000190	0.004947	0	0	0.000000
votes: cool	0.000159	0.005399	0	0	0.000000
votes: funny	0.000113	0.004547	0	0	0.000004
votes: useful	0.000216	0.005802	0	0	0.000012
fans	0.001154	0.012567	0	0	0.000000
yelping_since_year	0.659409	0.181769	0.500000	0.642857	0.785714
yelping_since_month	0.489904	0.305592	0.181818	0.454545	0.727273
yelping_since_day	0.492969	0.293464	0.233333	0.500000	0.733333
elite_2005	0.000182	0.013495	0	0	0.000000
elite_2006	0.000872	0.029511	0	0	0.000000
elite_2007	0.002433	0.049264	0	0	0.000000
elite_2008	0.003812	0.061623	0	0	0.000000
elite_2009	0.007038	0.083600	0	0	0.000000
elite_2010	0.011267	0.105546	0	0	0.000000
elite_2011	0.015469	0.123409	0	0	0.000000
elite_2012	0.021909	0.146387	0	0	0.000000
elite_2013	0.024862	0.155706	0	0	0.000000
elite_2014	0.026410	0.160353	0	0	0.000000
elite_2015	0.033110	0.178926	0	0	0.000000
elite_2016	0.038366	0.192081	0	0	0.000000
elite_2017	0.042465	0.201648	0	0	0.000000
elite_2018	0.038353	0.192049	0	0	0.000000
elite_None	0.926520	0.260925	1	1	1.000000

We summarize the statistics of the ratings for ON and PA in Table E1. We show the distributions of the number of reviews of each business and user in Figures E1a and E1c, respectively, both of which follow a power-law distribution. The distributions of the average ratings of businesses and users are shown in Figures E1b and E1d.

Table E5 Business Auxiliary Information (Attributes)

	ON Sum	ON Mean	PA Sum	PA Mean
attributes: AcceptsInsurance	745	0.022999	303	0.027631
attributes: AgesAllowed: 19plus	59	0.001821	NaN	NaN
attributes: Alcohol: beer and wine	1615	0.049856	154	0.014043
attributes: Alcohol: full bar	4711	0.145433	1415	0.129035
attributes: Alcohol: none	4840	0.149415	1692	0.154295
attributes: Ambience: casual	4989	0.154015	1351	0.123199
attributes: Ambience: classy	198	0.006112	63	0.005745
attributes: Ambience: divey	NaN	NaN	142	0.012949
attributes: Ambience: hipster	291	0.008983	74	0.006748
attributes: Ambience: intimate	195	0.006020	63	0.005745
attributes: Ambience: romantic	144	0.004445	NaN	NaN
attributes: Ambience: trendy	476	0.014695	157	0.014317
attributes: Ambience: upscale	90	0.002778	NaN	NaN
attributes: BestNights: Friday	897	0.027691	343	0.031278
attributes: BestNights: Monday	136	0.004198	NaN	NaN
attributes: BestNights: Saturday	904	0.027907	351	0.032008
attributes: BestNights: Sunday	215	0.006637	82	0.007478
attributes: BestNights: Thursday	558	0.017226	202	0.018421
attributes: BestNights: Tuesday	112	0.003458	55	0.005016
attributes: BestNights: Wednesday	199	0.006143	96	0.008754
attributes: BikeParking	13816	0.426512	3857	0.351724
attributes: BusinessAcceptsCreditCards	17896	0.552465	8240	0.751413
attributes: BusinessParking: garage	875	0.027012	307	0.027996
attributes: BusinessParking: lot	5386	0.166270	2009	0.183203
attributes: BusinessParking: street	6708	0.207082	2041	0.186121
attributes: BusinessParking: valet	131	0.004044	79	0.007204
attributes: BusinessParking: validated	151	0.004662	NaN	NaN
attributes: ByAppointmentOnly	2077	0.064119	909	0.082893
attributes: Caters	4889	0.150928	1377	0.125570
attributes: CoatCheck	365	0.011268	63	0.005745
attributes: DogsAllowed	718	0.022165	230	0.020974
attributes: DriveThru	203	0.006267	122	0.011125
attributes: GoodForDancing	379	0.011700	138	0.012584
attributes: GoodForKids	11685	0.360726	3314	0.302207
attributes: GoodForMeal: breakfast	617	0.019047	254	0.023163
attributes: GoodForMeal: brunch	982	0.030315	227	0.020700
attributes: GoodForMeal: dessert	375	0.011577	119	0.010852
attributes: GoodForMeal: dinner	4060	0.125336	1226	0.111800
attributes: GoodForMeal: latenight	536	0.016547	182	0.016597
attributes: GoodForMeal: lunch	4504	0.139042	1385	0.126299
attributes: HairSpecializesIn: coloring	221	0.006822	68	0.006201
attributes: HairSpecializesIn: curly	196	0.006051	65	0.005927
attributes: HairSpecializesIn: extensions	145	0.004476	61	0.005563
attributes: HairSpecializesIn: kids	132	0.004075	NaN	NaN
attributes: HairSpecializesIn: perms	130	0.004013	NaN	NaN
attributes: HappyHour	883	0.027259	626	0.057086
attributes: HasTV	5588	0.172506	1820	0.165968
attributes: Music: background music	619	0.019109	120	0.010943
attributes: Music: dj	358	0.011052	95	0.008663
attributes: Music: jukebox	96	0.002964	129	0.011764
attributes: Music: karaoke	53	0.001636	NaN	NaN
attributes: Music: live	297	0.009169	99	0.009028
attributes: NoiseLevel: loud	1147	0.035409	295	0.026901
attributes: NoiseLevel: quiet	2608	0.080511	636	0.057997
attributes: NoiseLevel: very loud	438	0.013521	112	0.010213
attributes: OutdoorSeating	4054	0.125150	1253	0.114262
attributes: RestaurantsAttire: casual	11331	0.349798	3136	0.285975
attributes: RestaurantsAttire: dressy	374	0.011546	85	0.007751
attributes: RestaurantsDelivery	3971	0.122588	1028	0.093744
attributes: RestaurantsGoodForGroups	10741	0.331584	3013	0.274758
attributes: RestaurantsPriceRange2: 1	5897	0.182046	2397	0.218585
attributes: RestaurantsPriceRange2: 2	13849	0.427531	3639	0.331844
attributes: RestaurantsPriceRange2: 3	2777	0.085728	616	0.056174
attributes: RestaurantsPriceRange2: 4	499	0.015405	104	0.009484
attributes: RestaurantsReservations	6270	0.193560	1071	0.097666
attributes: RestaurantsTableService	7090	0.218874	1863	0.169889
attributes: RestaurantsTakeOut	13920	0.429722	3889	0.354642
attributes: Smoking: no	833	0.025715	317	0.028908
attributes: Smoking: outdoor	703	0.021702	140	0.012767
attributes: Smoking: yes	NaN	NaN	141	0.012858
attributes: WheelchairAccessible	6500	0.200661	2550	0.232537
attributes: WiFi: free	5533	0.170809	1593	0.145267
attributes: WiFi: no	6135	0.189393	1593	0.145267
attributes: WiFi: paid	89	0.002748	NaN	NaN

Table E6 Business Auxiliary Information (Categories)

	ON Sum	ON Mean	PA Sum	PA Mean
categories: Restaurants: Afghan	82	0.002531	NaN	NaN
categories: Restaurants: African	58	0.001791	NaN	NaN
categories: Restaurants: American (New)	226	0.006977	495	0.045140
categories: Restaurants: American (Traditional)	846	0.026117	674	0.061463
categories: Restaurants: Asian Fusion	516	0.015929	75	0.006839
categories: Restaurants: Barbeque	380	0.011731	102	0.009301
categories: Restaurants: Breakfast & Brunch	1100	0.033958	339	0.030914
categories: Restaurants: British	75	0.002315	NaN	NaN
categories: Restaurants: Buffets	137	0.004229	NaN	NaN
categories: Restaurants: Burgers	1011	0.031210	342	0.031187
categories: Restaurants: Cafes	930	0.028710	182	0.016597
categories: Restaurants: Cajun/Creole	70	0.002161	NaN	NaN
categories: Restaurants: Canadian (New)	1108	0.034205	NaN	NaN
categories: Restaurants: Caribbean	361	0.011144	NaN	NaN
categories: Restaurants: Chicken Shop	111	0.003427	NaN	NaN
categories: Restaurants: Chicken Wings	598	0.018461	200	0.018238
categories: Restaurants: Chinese	1562	0.048220	240	0.021886
categories: Restaurants: Chinese: Dim Sum	163	0.005032	NaN	NaN
categories: Restaurants: Comfort Food	309	0.009539	NaN	NaN
categories: Restaurants: Creperies	67	0.002068	NaN	NaN
categories: Restaurants: Delis	242	0.007471	141	0.012858
categories: Restaurants: Diners	331	0.010218	158	0.014408
categories: Restaurants: Fast Food	1120	0.034575	343	0.031278
categories: Restaurants: Filipino	86	0.002655	NaN	NaN
categories: Restaurants: Fish & Chips	144	0.004445	NaN	NaN
categories: Restaurants: Food Court	82	0.002531	NaN	NaN
categories: Restaurants: Food Stands	57	0.001760	NaN	NaN
categories: Restaurants: French	212	0.006545	NaN	NaN
categories: Restaurants: Gastropubs	146	0.004507	NaN	NaN
categories: Restaurants: Gluten-Free	185	0.005711	NaN	NaN
categories: Restaurants: Greek	301	0.009292	58	0.005289
categories: Restaurants: Halal	326	0.010064	NaN	NaN
categories: Restaurants: Hot Dogs	98	0.003025	55	0.005016
categories: Restaurants: Hot Pot	61	0.001883	NaN	NaN
categories: Restaurants: Indian	704	0.021733	56	0.005107
categories: Restaurants: Irish	51	0.001574	NaN	NaN
categories: Restaurants: Italian	1146	0.035378	444	0.040489
categories: Restaurants: Japanese	950	0.029327	90	0.008207
categories: Restaurants: Japanese: Ramen	73	0.002254	NaN	NaN
categories: Restaurants: Korean	414	0.012781	NaN	NaN
categories: Restaurants: Latin American	113	0.003488	NaN	NaN
categories: Restaurants: Mediterranean	591	0.018245	90	0.008207
categories: Restaurants: Mediterranean: Falafel	60	0.001852	NaN	NaN
categories: Restaurants: Mexican	401	0.012379	185	0.016870
categories: Restaurants: Middle Eastern	553	0.017072	51	0.004651
categories: Restaurants: Middle Eastern: Lebanese	53	0.001636	NaN	NaN
categories: Restaurants: Modern European	85	0.002624	NaN	NaN
categories: Restaurants: Noodles	124	0.003828	NaN	NaN
categories: Restaurants: Pakistani	195	0.006020	NaN	NaN
categories: Restaurants: Persian/Iranian	97	0.002994	NaN	NaN
categories: Restaurants: Pizza	1168	0.036057	776	0.070764
categories: Restaurants: Portuguese	133	0.004106	NaN	NaN
categories: Restaurants: Poutineries	63	0.001945	NaN	NaN
categories: Restaurants: Salad	333	0.010280	222	0.020244
categories: Restaurants: Sandwiches	1197	0.036952	596	0.054350
categories: Restaurants: Seafood	538	0.016609	150	0.013679
categories: Restaurants: Soup	147	0.004538	70	0.006383
categories: Restaurants: Southern	60	0.001852	NaN	NaN
categories: Restaurants: Steakhouses	268	0.008273	90	0.008207
categories: Restaurants: Sushi Bars	641	0.019788	102	0.009301
categories: Restaurants: Taiwanese	104	0.003211	NaN	NaN
categories: Restaurants: Tapas Bars	100	0.003087	NaN	NaN
categories: Restaurants: Tapas/Small Plates	103	0.003180	NaN	NaN
categories: Restaurants: Tex-Mex	78	0.002408	NaN	NaN
categories: Restaurants: Thai	543	0.016763	78	0.007113
categories: Restaurants: Turkish	57	0.001760	NaN	NaN
categories: Restaurants: Vegan	205	0.006329	NaN	NaN
categories: Restaurants: Vegetarian	267	0.008243	58	0.005289
categories: Restaurants: Vietnamese	417	0.012873	NaN	NaN

Appendix F. Performance Evaluation Metrics

The RMSE is computed as $\sqrt{\frac{||\mathbf{\Omega}_{\text{test}} \circ (\mathbf{X} - \hat{\mathbf{X}})||_2^2}{||\mathbf{\Omega}_{\text{test}}||_1}}$, where $\mathbf{\Omega}_{\text{test}}$ is the indicator matrix, with one for the entries in the test set and zero otherwise, and where $||\cdot||_1$ represents the entry-wise ℓ_1 norm.

In addition to the RMSE, we use three ranking-based metrics: Spearman's rank-order correlation (Spearman's correlation), Bayesian personalized ranking (BPR) (Rendle et al. 2009), and Fraction of Concordant Pairs (FCP) (Koren and Sill 2013). These ranking-based metrics are defined on a rating pair level, instead of on a single-rating level. Spearman's correlation is defined by comparing the predicted and the actual rankings for all predictions. In contrast, BPR and FCP are user-level pairwise ranking metrics, which compare the ranking pairs for each user. Therefore, these two metrics are evaluated only on users who have more than one rating in the test set (30% of users in PA and 37% of users in ON). BPR is a pairwise personalized ranking loss derived from the maximum posterior estimator, which is computed as $\text{BPR} = e^{\frac{1}{|D_{\text{test}}|} \sum_{i,j,j' \in D_{\text{test}}} \ln \sigma (\hat{X}_{ij} - \hat{X}_{ij'})}$, where D_{test} consists of pairs in the test set for which $X_{ij} \geq X_{ij'}$, for each user i; and $\sigma(\cdot)$ represents the sigmoid function. The BPR represents the geometric mean of the data likelihood, and a higher BPR corresponds to a better personalized ranking. FCP measures the correctly ranked business pairs for each user in the RS. The number of correctly ranked (concordant) business pairs for each user, based on predicted ratings, is $n_c^i = |\{(j,j')|\hat{X}_{ij} \leq \hat{X}_{ij'} \text{ and } X_{ij} \leq X_{ij'}\}|$. The number of discordant pairs for user i is similarly defined as $n_d^i = |\{(j,j')|\hat{X}_{ij} > \hat{X}_{ij'} \text{ and } X_{ij} \leq X_{ij'}\}|$. FCP is defined as $\frac{\sum_{i=1}^n n_c^i}{\sum_{i=1}^n n_c^i + \sum_{i=1}^n n_d^i}$, and a higher value represents more concordant pairs.

A number of CS studies use top-N metrics to evaluate the efficacy of RSs. Here, top-N metrics determine whether the recommended list contains the top-N items (such as recall@N, precision@N, and nDCG) from the groundtruth list for each user. In evaluating these top-N related metrics, many of these studies sample the test set to include only users that provide a minimum number of ratings—for example, choosing a threshold of ten (Chen et al. 2019). However, because of data sparsity issues, using top-N metrics also results in excluding a significant number of users from our data. For instance, in the Yelp dataset, the distribution of users rating a particular number of items is as follows: ON [1 (63%), 2 (15%), 3 (7%), 4 (4%), 5 (2%), 6+ (8%)]; PA [1 (70%), 2 (14%), 3 (5%), 4 (3%), 5 (2%), 6 and above (6%)]. This issue may elucidate why the management studies referenced above do not use this metric. Moreover, setting a value for N in the prediction task can seem arbitrary, but it also can markedly influence the results, depending on the distribution of ratings. If we choose to use a top-N metric, we are faced with two choices: We either must select a tiny value for N, rendering the evaluation task overly simplistic and less informative, or we must exclude users who have not visited a certain number of stores, thereby introducing a systematic bias into the sample. Neither option is suitable for our evaluation.

Appendix G. Experiment Setup

This section complements Section 6 by documenting the randomized controlled experiment in greater detail. We first present the user interface for each experimental condition, then report omnibus and pairwise statistics for every questionnaire item, assess randomization balance across demographic and behavioral covariates, and summarize mediation models linking explanation-quality dimensions (transparency, trust, effectiveness, efficiency, persuasiveness, and satisfaction) to perceived relevance and future interest. Together, these materials are intended to facilitate replication and to make explicit how the experimental evidence supports our claims about explanation quality and user acceptance.

G.1. Branching Questionnaire and Scenario Assignment

Participants completed a short branching questionnaire to elicit key preferences for their closest friend. We first asked whether the friend typically prefers fast food. Fast-food preferences routed respondents to one of two focal fast-food restaurants (*Wonton Chai Noodle* or *One2 Snacks*), with a follow-up on table service. If the friend did not prefer fast food, respondents chose between American and Italian cuisine to keep the branching structure tractable and aligned with Yelp coverage. For American cuisine, one of three follow-up questions (full bar, reservations, or event planning services) was randomly sampled to select one of two focal American restaurants (*Fanzone Wings & Ribs* or *Farm'r Eatery & Catering*); for Italian cuisine, a credit-card acceptance question routed to *Pastizza* or *Campo*. For each path, participants were shown two previously liked restaurants consistent with the elicited preferences, followed by a focal recommendation produced by MG-GAT for a matched user in Yelp.

G.2. On-Screen Text for Illustrative Path

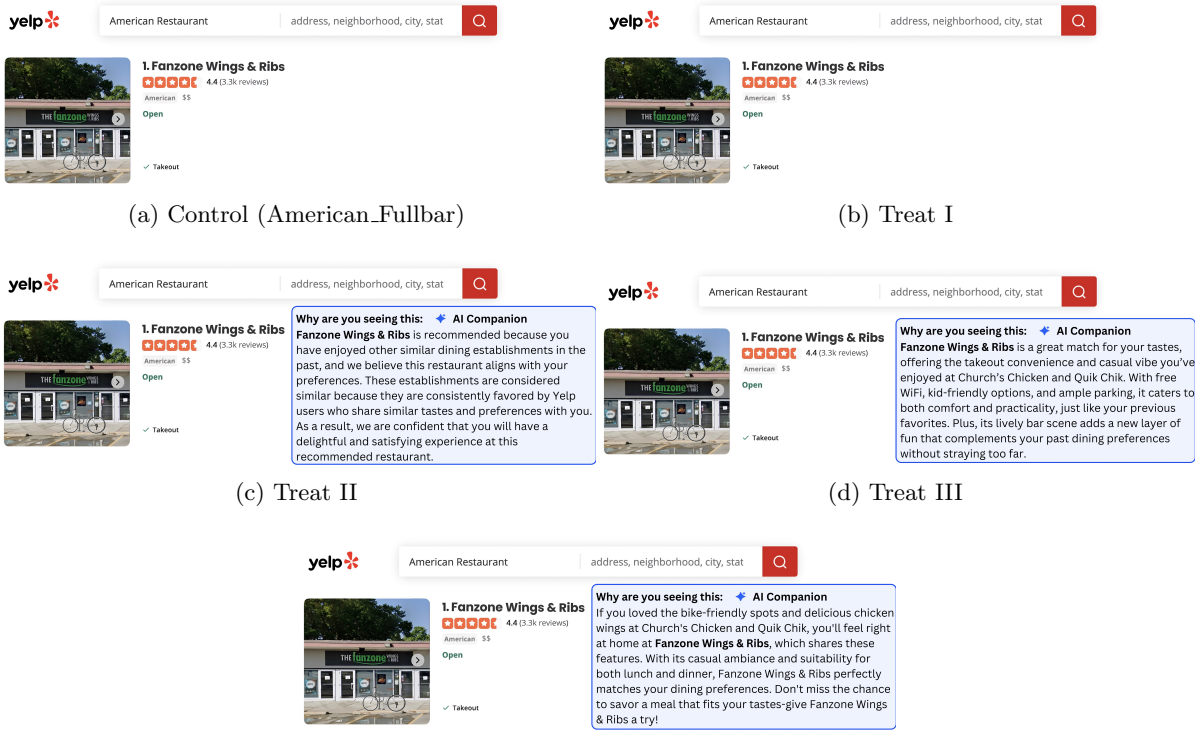
For reproducibility, we record the exact on-screen text shown in the illustrative American/full-bar path (Fanzone Wings & Ribs). Screenshots for all conditions are in Figures G1–G6.

- **Context blurb (before the recommendation).** "Imagine that your friend has visited the following two restaurants, both of which he/she rated 4 out of 5. Each restaurant offers chicken wings, convenient bike parking, and a casual atmosphere, making them suitable for a relaxed meal. Whether for lunch or dinner, both establishments are equally ideal."
- **Treatment I (Relevant Business).** "Fanzone Wings & Ribs is recommended because it closely resembles two restaurants that you already enjoy and have expressed a liking for: Church's Chicken and Quik Chik. The similarities between these restaurants and Fanzone Wings & Ribs provide a strong basis for believing you would likely appreciate Fanzone's offerings as well, making it a suitable recommendation based on your preferences."
- **Treatment II (Park et al. (2017) Social).** "Fanzone Wings & Ribs is recommended because you have enjoyed other similar dining establishments in the past, and we believe this restaurant aligns with your preferences. These establishments are considered similar because they are consistently favored by your friends or people you follow who share similar tastes and preferences with you. As a result, we are confident that you will have a delightful and satisfying experience at this recommended restaurant."

- **Treatment III (SHAP features).** "Fanzone Wings & Ribs is a great match for your tastes, offering the takeout convenience and casual vibe you've enjoyed at Church's Chicken and Quik Chik. With free WiFi, kid-friendly options, and ample parking, it caters to both comfort and practicality, just like your previous favorites. Plus, its lively bar scene adds a new layer of fun that complements your past dining preferences without straying too far."
- **Treatment IV (MG-GAT parameter-grounded).** "If you loved the bike-friendly spots and delicious chicken wings at Church's Chicken and Quik Chik, you'll feel right at home at Fanzone Wings & Ribs, which shares these features. With its casual ambiance and suitability for both lunch and dinner, Fanzone Wings & Ribs perfectly matches your dining preferences. Overall, these shared features make Fanzone Wings & Ribs a strong match for your dining preferences."

G.3. Interface of the Experiment Conditions

Figures G1–G6 present the interfaces for all experiment conditions grouped by restaurant.



(a) Control (American_Fullbar) (b) Treat I

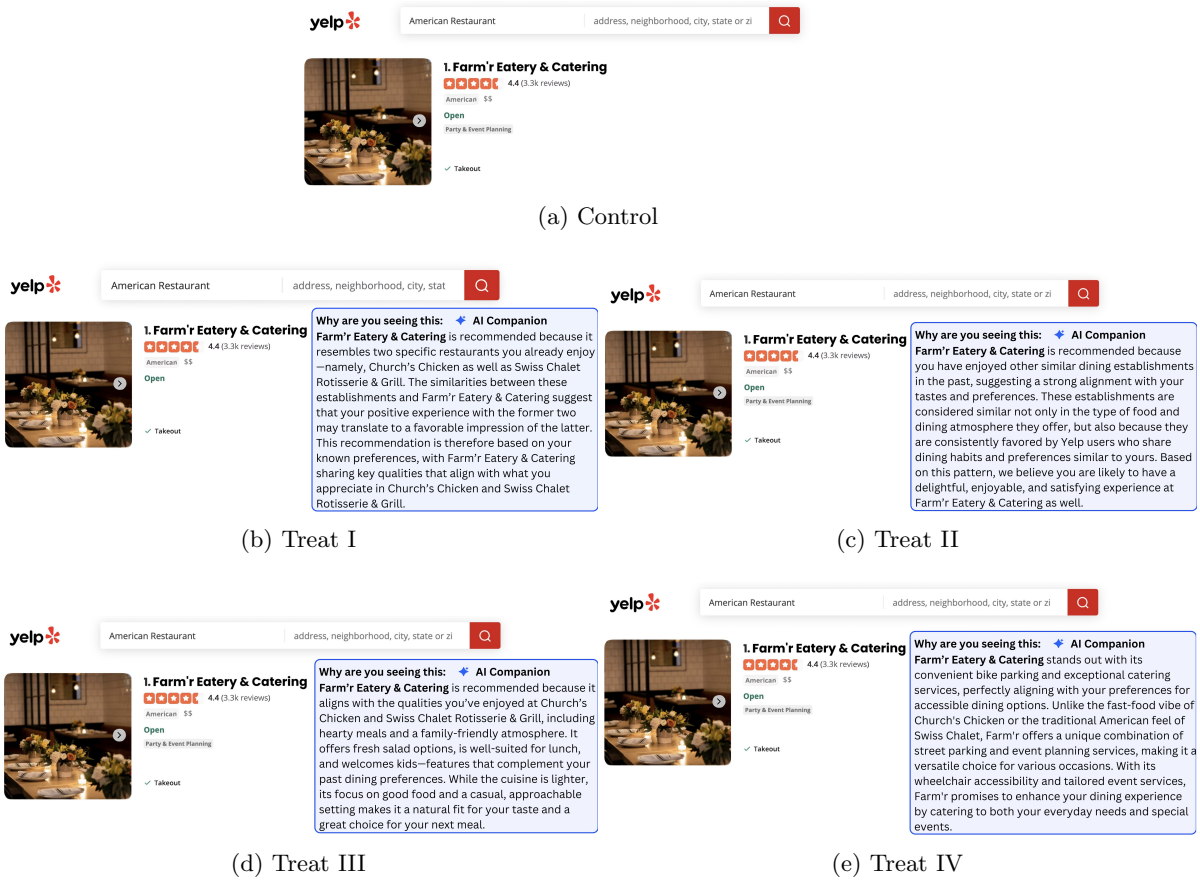
(c) Treat II (d) Treat III

(e) Treat IV

Figure G1 Interfaces for Fanzone Wings & Ribs (American_Fullbar)

G.4. Statistical Analysis of All Questions

We next present a detailed analysis of our experimental results in Table G1 across five experimental conditions: No Explanation (No EXP), Relevant Business Explanation (Relevant Business EXP), Park et al. (2017) Explanation (Park EXP), SHAP Explanation (SHAP EXP), and MGGAT Explanation (MGGAT EXP).



(a) Control

(b) Treat I

(c) Treat II

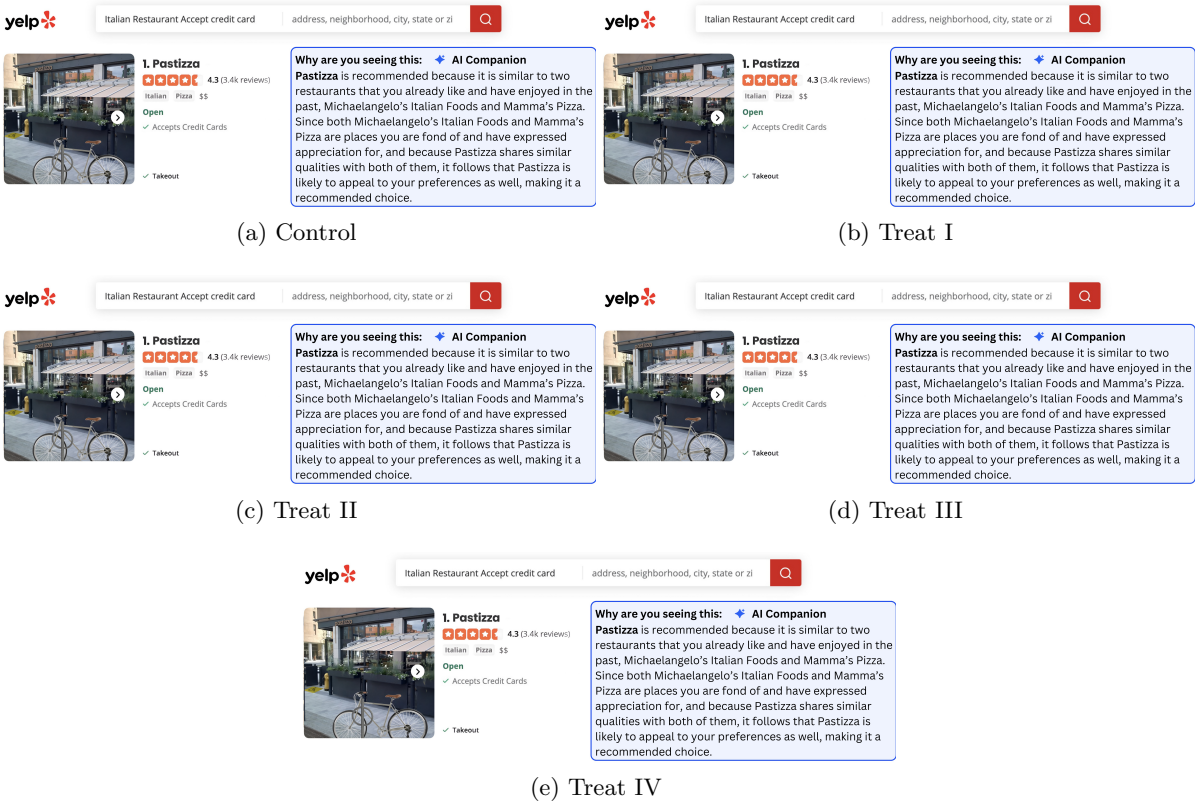
(d) Treat III

(e) Treat IV

Figure G2 Interfaces for Farm'r Eatery & Catering (American_noFullbar/Reservation)

Each condition's performance was evaluated across several questions using the mean score and standard deviation (SD) on a 7-point Likert scale. An Analysis of Variance (ANOVA) was conducted to assess the statistical significance of differences in mean scores among these conditions, followed by pairwise comparisons.

- Match with User Interests, Engagement, and Satisfaction: For queries pertaining to the match between the recommended item and user interests, as well as the probability of users engaging with the recommendation, the MGGAT-Based Explanation condition achieved the highest mean ratings across the five conditions. This pertains to the statements, "The recommended item, 'Fanzone Wings & Ribs,' matched your friend's interests," "Your friend would click on the recommendation, 'Fanzone Wings & Ribs,'" and "Your friend would be satisfied with this recommendation." The ANOVA test revealed a significant difference across conditions for all statements (with $p < 0.001$).
- Understanding of Recommendation System and Confidence in Its Effectiveness: The MGGAT-Based Explanation condition also achieved the highest mean ratings in statements related to understanding the recommendation system and the user's confidence in its effectiveness. These statements include: "This explanation helps your friend to understand what recommendation is based on," "I believe my friend has gained insight into why 'Fanzone Wings & Ribs' was recommended to him/her," and "This explanation increases your friend's confidence in the recommender system." Once again, the ANOVA



(a) Control (b) Treat I

(c) Treat II (d) Treat III

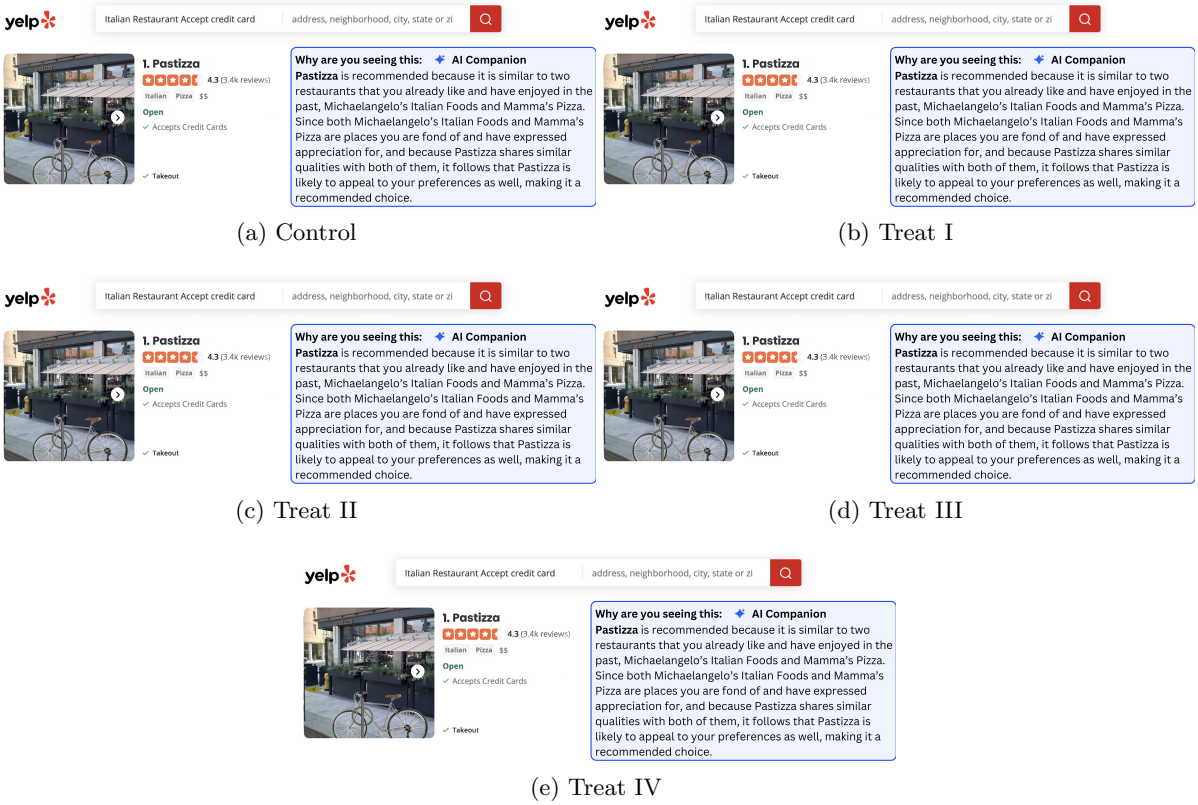
(e) Treat IV

Figure G3 Interfaces for Pastizza (Italian_CreditCards)

results demonstrated significant variances across the conditions for all statements (with p-values less than 0.001).

- Usefulness of Explanations and Comfort in Relying on Them: For items assessing usefulness, decision efficiency, persuasiveness, and ease of relying on explanations, the MGGAT-Based Explanation condition again achieved the highest mean ratings. ANOVA indicates significant differences across conditions for these statements ($p < 0.001$).
- Future Engagement and Willingness to Explore: Regarding the statement on future engagement, "Your friend would feel more inclined to try other recommended businesses provided by the same platform in the future," the MGGAT-Based Explanation condition displayed a significantly higher mean score (p=0.005).

In summary, the MGGAT Explanation condition achieves the highest mean ratings across engagement and explanation-quality measures. Pairwise comparisons show that, relative to the SHAP Explanation condition, the MGGAT condition significantly improves perceived relevance, future interest, trust, persuasiveness, and satisfaction, while differences on transparency, efficiency, and effectiveness are not statistically distinguishable (Table G1).



(a) Control

(b) Treat I

(c) Treat II

(d) Treat III

(e) Treat IV

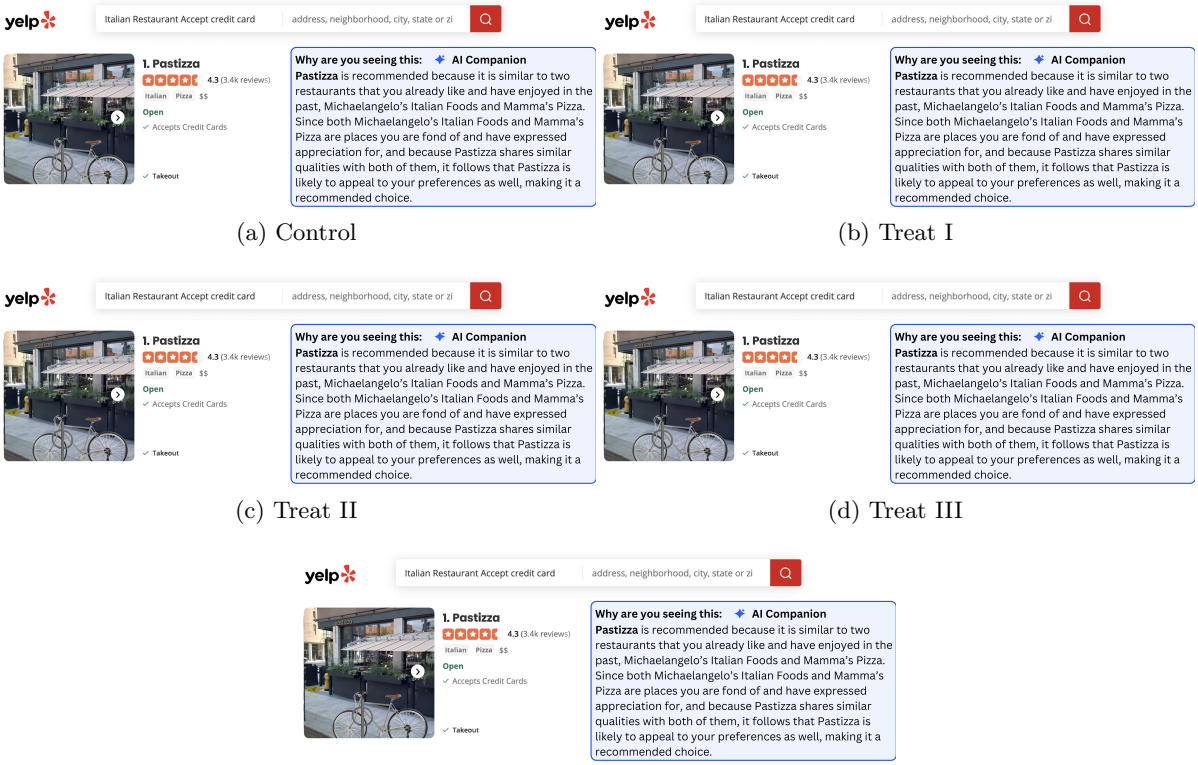
Figure G4 Interfaces for Campo (Italian_noCreditCards)

G.5. Balance Check

We first affirmed balance across the five conditions through a chi-squared test for gender and race. The outcomes of these tests were not statistically significant, suggesting no differences in the distribution of these demographic variables across the groups. Consequently, this indicates that the groups were demographically balanced, and any variations in responses cannot be ascribed to these demographic factors. To further ensure the equivalence of the groups, we conducted a one-way ANOVA for age, propensity to explore new restaurants, and years of Yelp usage. The results, being non-significant, suggest no statistically meaningful disparities among the groups concerning these variables. Therefore, any differences in users' perception of the recommendations cannot be ascribed to their age, willingness to explore new restaurants, or familiarity with Yelp. Furthermore, we employed the Tukey multiple comparisons of means to bolster the above findings. All comparisons turned out to be non-significant, providing robust evidence of no substantial differences among the conditions in terms of the variables we scrutinized.

G.6. Mediation Analysis: How Explanations Mediate Users' Perceived Relevance and Future Interest in the Recommendation

We conducted a mediation analysis to assess the underlying mechanisms through which different explanations affect Perceived Relevance and Future Interest. Specifically, we conducted three separate regression models to explore the mediation effects:



(a) Control (b) Treat I

(c) Treat II (d) Treat III

(e) Treat IV

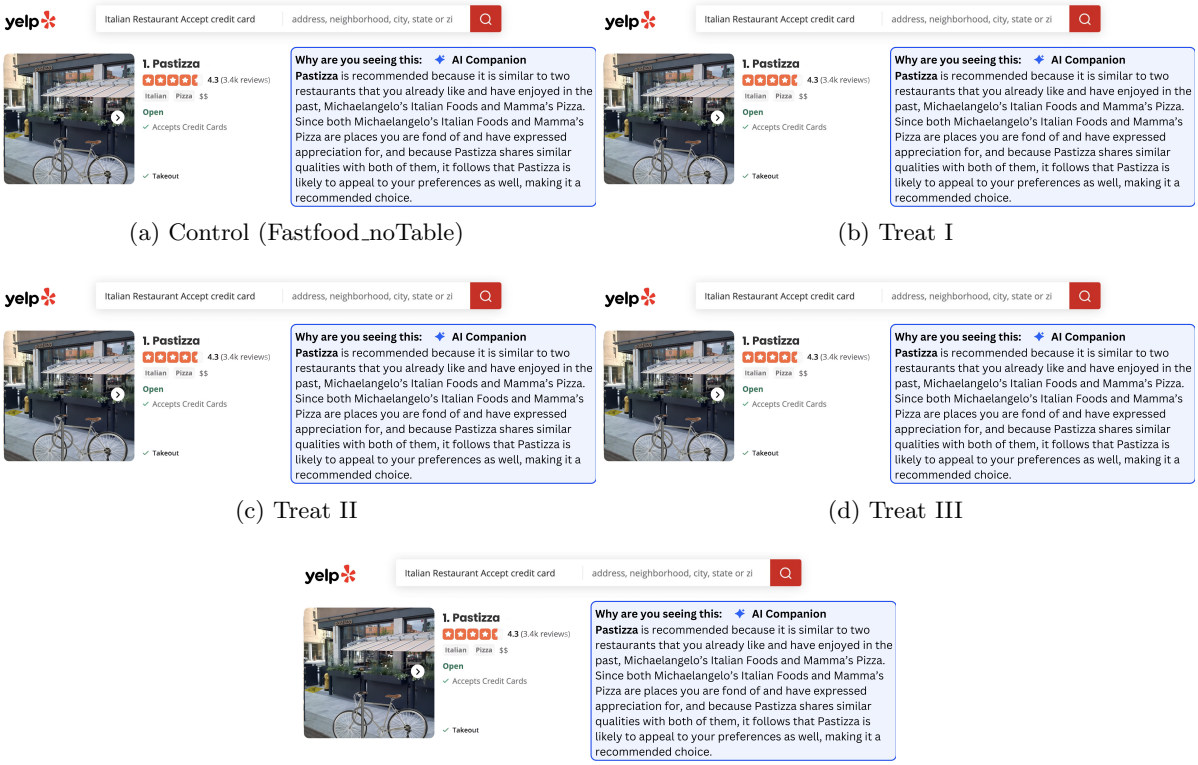
Figure G5 Interfaces for Wonton Chai Noodle (Fastfood_Table)

Model 1: X → Y (Total Effect): Regress the dependent variable (Y) on the independent variable (X). This step estimates the total effect of X on Y. Here, the independent variables correspond to the experimental condition indicators, namely the Relevant Business Explanation condition, the Park et al. (2017) Explanation condition, the MG-GAT condition, and the SHAP Explanation condition. The dependent variables in the analysis are Perceived Relevance and Future Interest. Regression results are displayed in the first column of Table G2. The MG-GAT Explanation condition serves as the reference group for the baseline intercept term. Panels (1) to (6) present results for Perceived Relevance, while panels (8) to (13) provide results for Future Interest.

Model 2: X → M: Regress the mediator (M) on the independent variable (X). This step estimates the effect of X on M. We consider six mediators: transparency, trust, effectiveness, efficiency, persuasiveness, and satisfaction. Results are presented in the second column of Tables G2.

Model 3: X + M → Y: Regress the dependent variable (Y) on both the independent variable (X) and the mediator (M). This will estimate the effect of M on Y and the direct effect of X on Y. We report how the estimated treatment coefficients change when each explanation-quality dimension is included; we interpret coefficient attenuation patterns as exploratory evidence consistent with mediation-style pathways. The results are presented in the third column of Tables G2.

Across dimensions, including the mediator attenuates the treatment coefficients; we report these patterns as exploratory associations rather than claims of "full mediation."



(a) Control (Fastfood_noTable)

(b) Treat I

(c) Treat II

(d) Treat III

(e) Treat IV

Figure G6 Interfaces for One2 Snacks (Fastfood_noTable)

Table G1 Participants' ratings on their satisfaction, explanation effectiveness, and influence on their behavior

	No EXP		RB EXP		Park EXP		SHAP EXP		MGGAT EXP		*ANOVA p-value*	MGGAT vs. No EXP	MGGAT vs. RB	MGGAT vs. Park	MGGAT vs. SHAP
Questions	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**					
I believe my friend have gained insight into why [recommended restaurant] was recommended to him/her.	3.30	1.58	4.93	1.50	5.13	1.51	5.48	1.31	5.71	1.12	<.001	<.001	<.001	<.001	<.05
Your friend would click on the recommendation, [recommended restaurant]	4.45	1.50	5.14	1.40	5.21	1.34	4.85	1.66	5.50	1.33	<.001	<.001	<.01	<.05	<.001
Your friend would feel more inclined to try other recommended businesses provided by the same platform in the future.	4.31	1.32	5.02	1.27	4.97	1.34	4.81	1.58	5.24	1.33	<.001	<.001	<.05	<.05	<.01
The recommended item, [recommended restaurant], matched your friend's interests.	4.77	1.55	5.47	1.25	5.48	1.19	4.83	1.82	5.82	1.19	<.001	<.001	<.001	<.01	<.001
Your friend would be satisfied with this recommendation.	4.48	1.52	5.14	1.44	5.11	1.49	4.80	1.77	5.52	1.36	<.001	<.001	<.01	<.001	<.001
The method behind the recommendation system is competent and effective.	4.26	1.49	4.88	1.62	4.95	1.63	4.70	1.75	5.42	1.31	<.001	<.001	<.001	<.001	<.001
This explanation helps your friend to understand what recommendation is based on. (Transparency)	–	–	4.89	1.87	5.03	1.83	5.64	1.31	5.83	1.20	<.001	–	<.001	<.001	0.097
This explanation increases your friend's confidence in the recommender system. (Trust)	–	–	4.73	1.68	4.76	1.72	4.73	1.72	5.35	1.40	<.001	–	<.001	<.001	<.001
This explanation helps your friend decide efficiently whether to visit [recommended restaurant]. (Efficiency)	–	–	4.79	1.70	4.87	1.69	5.33	1.42	5.48	1.31	<.001	–	<.001	<.001	0.229
This explanation persuades your friend to visit [recommended restaurant]. (Persuasiveness)	–	–	4.66	1.66	4.73	1.67	4.55	1.81	5.23	1.30	<.001	–	<.001	<.001	<.001
This explanation is effective in helping your friend evaluate if he/she should visit [recommended restaurant]. (Effectiveness)	–	–	4.80	1.64	4.84	1.64	5.33	1.46	5.49	1.34	<.001	–	<.001	<.001	0.219
This explanation would improve how easy it is to pick a recommendation. (Satisfaction)	–	–	4.79	1.67	4.77	1.65	4.94	1.62	5.42	1.34	<.001	–	<.001	<.001	<.001

Note: RB stands for Relevant Business.

Table G2 Mediation Analysis Results

(1)	Y (Relevance)			M (Transparency)			Y (Relevance)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.356	0.118	0.003	-0.937	0.138	< 0.001	0.128	0.096	0.184
Park EXP	-0.344	0.120	0.004	-0.792	0.141	< 0.001	0.065	0.098	0.503
SHAP EXP	-0.993	0.125	< 0.001	-0.190	0.145	0.191	-0.895	0.099	< 0.001
M (Transparency)	—	—	—	—	—	—	0.517	0.021	< 0.001
Constant	5.821	0.086	< 0.001	5.825	0.101	< 0.001	2.809	0.142	< 0.001

(2)	Y (Relevance)			M (Trust)			Y (Relevance)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.356	0.118	0.003	-0.620	0.141	< 0.001	0.014	0.084	0.868
Park EXP	-0.344	0.120	0.004	-0.586	0.144	< 0.001	0.006	0.085	0.946
SHAP EXP	-0.993	0.125	< 0.001	-0.620	0.149	< 0.001	-0.623	0.088	< 0.001
M (Trust)	—	—	—	—	—	—	0.597	0.018	< 0.001
Constant	5.821	0.086	< 0.001	5.349	0.103	< 0.001	2.629	0.115	< 0.001

(3)	Y (Relevance)			M (Effectiveness)			Y (Relevance)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.356	0.118	0.003	-0.690	0.132	< 0.001	0.046	0.091	0.616
Park EXP	-0.344	0.120	0.004	-0.650	0.135	< 0.001	0.034	0.092	0.710
SHAP EXP	-0.993	0.125	< 0.001	-0.157	0.139	0.259	-0.902	0.095	< 0.001
M (Effectiveness)	—	—	—	—	—	—	0.582	0.021	< 0.001
Constant	5.821	0.086	< 0.001	5.492	0.097	< 0.001	2.624	0.133	< 0.001

(4)	Y (Relevance)			M (Efficiency)			Y (Relevance)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.356	0.118	0.003	-0.692	0.133	< 0.001	0.038	0.092	0.676
Park EXP	-0.344	0.120	0.004	-0.612	0.136	< 0.001	0.005	0.093	0.960
SHAP EXP	-0.993	0.125	< 0.001	-0.150	0.141	0.287	-0.908	0.095	< 0.001
M (Efficiency)	—	—	—	—	—	—	0.570	0.021	< 0.001
Constant	5.821	0.086	< 0.001	5.480	0.097	< 0.001	2.698	0.133	< 0.001

(5)	Y (Relevance)			M (Persuasiveness)			Y (Relevance)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.356	0.118	0.003	-0.574	0.140	< 0.001	-0.009	0.083	0.915
Park EXP	-0.344	0.120	0.004	-0.509	0.142	< 0.001	-0.037	0.085	0.667
SHAP EXP	-0.993	0.125	< 0.001	-0.689	0.147	< 0.001	-0.577	0.088	< 0.001
M (Persuasiveness)	—	—	—	—	—	—	0.605	0.018	< 0.001
Constant	5.821	0.086	< 0.001	5.234	0.102	< 0.001	2.657	0.114	< 0.001

(6)	Y (Relevance)			M (Satisfaction)			Y (Relevance)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.356	0.118	0.003	-0.628	0.136	< 0.001	-0.006	0.092	0.948
Park EXP	-0.344	0.120	0.004	-0.642	0.139	< 0.001	0.014	0.093	0.882
SHAP EXP	-0.993	0.125	< 0.001	-0.477	0.144	< 0.001	-0.727	0.096	< 0.001
M (Satisfaction)	—	—	—	—	—	—	0.557	0.021	< 0.001
Constant	5.821	0.086	< 0.001	5.417	0.100	< 0.001	2.803	0.130	< 0.001

(7)	Y (Future Interest)			M (Transparency)			Y (Future Interest)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.221	0.119	0.062	-0.937	0.138	< 0.001	0.247	0.099	0.012
Park EXP	-0.272	0.121	0.025	-0.792	0.141	< 0.001	0.123	0.100	0.216
SHAP EXP	-0.435	0.125	< 0.001	-0.190	0.145	0.191	-0.340	0.102	< 0.001
M (Transparency)	—	—	—	—	—	—	0.500	0.022	< 0.001
Constant	5.242	0.087	< 0.001	5.825	0.101	< 0.001	2.331	0.145	< 0.001

(8)	Y (Future Interest)			M (Trust)			Y (Future Interest)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.221	0.119	0.062	-0.620	0.141	< 0.001	0.138	0.087	0.113
Park EXP	-0.272	0.121	0.025	-0.586	0.144	< 0.001	0.067	0.088	0.448
SHAP EXP	-0.435	0.125	< 0.001	-0.620	0.149	< 0.001	-0.077	0.091	0.402
M (Trust)	—	—	—	—	—	—	0.579	0.019	< 0.001
Constant	5.242	0.087	< 0.001	5.349	0.103	< 0.001	2.146	0.119	< 0.001

(9)	Y (Future Interest)			M (Effectiveness)			Y (Future Interest)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.221	0.119	0.062	-0.690	0.132	< 0.001	0.176	0.092	0.057
Park EXP	-0.272	0.121	0.025	-0.650	0.135	< 0.001	0.102	0.094	0.277
SHAP EXP	-0.435	0.125	< 0.001	-0.157	0.139	0.259	-0.345	0.096	< 0.001
M (Effectiveness)	—	—	—	—	—	—	0.576	0.021	< 0.001
Constant	5.242	0.087	< 0.001	5.492	0.097	< 0.001	2.081	0.135	< 0.001

(10)	Y (Future Interest)			M (Efficiency)			Y (Future Interest)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.221	0.119	0.062	-0.692	0.133	< 0.001	0.170	0.093	0.066
Park EXP	-0.272	0.121	0.025	-0.612	0.136	< 0.001	0.074	0.094	0.432
SHAP EXP	-0.435	0.125	< 0.001	-0.150	0.141	0.287	-0.350	0.096	< 0.001
M (Efficiency)	—	—	—	—	—	—	0.566	0.021	< 0.001
Constant	5.242	0.087	< 0.001	5.480	0.097	< 0.001	2.141	0.135	< 0.001

(11)	Y (Future Interest)			M (Persuasiveness)			Y (Future Interest)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.221	0.119	0.062	-0.574	0.140	< 0.001	0.100	0.090	0.267
Park EXP	-0.272	0.121	0.025	-0.509	0.142	< 0.001	0.012	0.092	0.893
SHAP EXP	-0.435	0.125	< 0.001	-0.689	0.147	< 0.001	-0.050	0.095	0.601
M (Persuasiveness)	—	—	—	—	—	—	0.559	0.020	< 0.001
Constant	5.242	0.087	< 0.001	5.234	0.102	< 0.001	2.314	0.123	< 0.001

(12)	Y (Future Interest)			M (Satisfaction)			Y (Future Interest)		
	Coeff.	**SE**	**p**	**Coeff.**	**SE**	**p**	**Coeff.**	**SE**	**p**
Relevant Business EXP	-0.221	0.119	0.062	-0.628	0.136	< 0.001	0.128	0.092	0.165
Park EXP	-0.272	0.121	0.025	-0.642	0.139	< 0.001	0.085	0.094	0.368
SHAP EXP	-0.435	0.125	< 0.001	-0.477	0.144	< 0.001	-0.170	0.097	0.079
M (Satisfaction)	—	—	—	—	—	—	0.556	0.021	< 0.001
Constant	5.242	0.087	< 0.001	5.417	0.100	< 0.001	2.232	0.131	< 0.001

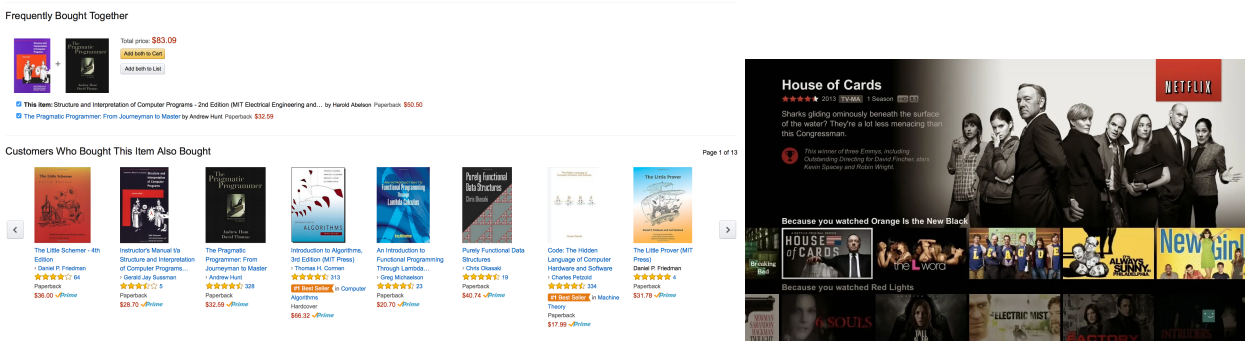
Table G3 Regression Results: Impact of Explanation Methods on User Responses

	Willingness to Click				Future Interest			
	Rel. Bus.	Park	MGGAT	Combined	Rel. Bus.	Park	MGGAT	Combined
Intercept	2.270***	2.584***	1.227**	2.151***	1.344***	1.765***	1.717***	1.634***
	(0.406)	(0.372)	(0.469)	(0.231)	(0.286)	(0.291)	(0.411)	(0.182)
Transparency	0.231*	0.108	0.334***	0.202***	0.042	0.084	0.079	0.071
	(0.093)	(0.079)	(0.085)	(0.049)	(0.065)	(0.062)	(0.075)	(0.038)
Trust	0.136	0.269**	0.181*	0.189***	0.253***	0.190**	0.224**	0.224***
	(0.094)	(0.082)	(0.077)	(0.048)	(0.067)	(0.064)	(0.067)	(0.038)
Effectiveness	-0.022	-0.210	0.130	-0.049	0.086	0.007	0.124	0.082
	(0.110)	(0.120)	(0.118)	(0.066)	(0.077)	(0.094)	(0.103)	(0.052)
Efficiency	0.237*	0.344***	0.055	0.229***	0.043	0.051	-0.151	-0.017
	(0.097)	(0.089)	(0.090)	(0.053)	(0.068)	(0.069)	(0.079)	(0.042)
Persuasiveness	-0.002	0.070	0.188*	0.091	0.126	0.197*	0.006	0.110*
	(0.104)	(0.099)	(0.089)	(0.056)	(0.073)	(0.078)	(0.078)	(0.044)
Satisfaction	-0.050	-0.094	-0.165	-0.097	0.154*	0.106	0.358***	0.185***
	(0.101)	(0.097)	(0.113)	(0.059)	(0.071)	(0.076)	(0.099)	(0.047)
$\mathbf{R^2}$	0.188	0.207	0.246	0.208	0.383	0.312	0.245	0.307

Note: Standard errors are reported in parentheses. Rel. Bus. = Relevant Business explanation; Combined = Combined explanation approach. Significance levels: * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$.

Appendix H. Explanations Associated with the RS in Practice

Given the benefits of ERS, platforms like Amazon (Figure H1a) and Netflix (Figure H1b) have started providing explanations. Both platforms build upon the similarity of products consumed by similar consumers.



(a) Amazon (b) Netflix

Figure H1 Real-world interpretable RSs: examples on Amazon and Netflix.

Appendix I. Ablation Studies: Interpretability Versus Predictability Trade-offs

We evaluate the trade-off between interpretability and predictive performance. We conducted two variations by incorporating non-linear transformations into our model: (1) using a single sigmoid activation function to introduce non-linearity in the first layer and (2) employing three non-linear layers, with the first two layers using ReLU activation functions and the final layer using a sigmoid activation function. The formulation of the two setups is as follows. For the first model using a single nonlinear (sigmoid) layer:

$$\text{user } i\text{: } \mathbf{H}_{u,i}^{(1)} = \sigma\left(\mathbf{W}_u^{(1)}\mathbf{S}_{u,i}\right); \text{business } j\text{: } \mathbf{H}_{b,j}^{(1)} = \sigma\left(\mathbf{W}_b^{(1)}\mathbf{S}_{b,j}\right), \tag{15}$$

where σ denotes the sigmoid activation function. We can further increase the predictability by chaining more nonlinear layers (at the cost of an even lower interpretability):

$$\begin{aligned} \text{user } i\text{: } \mathbf{H}_{u,i}^{(1)} &= \sigma\left(\mathbf{W}_u^{(3)}\text{ReLU}\left(\mathbf{W}_u^{(2)}\text{ReLU}\left(\mathbf{W}_u^{(1)}\mathbf{S}_{u,i}\right)\right)\right), \\ \text{business } j\text{: } \mathbf{H}_{b,j}^{(1)} &= \sigma\left(\mathbf{W}_b^{(3)}\text{ReLU}\left(\mathbf{W}_b^{(2)}\text{ReLU}\left(\mathbf{W}_b^{(1)}\mathbf{S}_{b,j}\right)\right)\right), \end{aligned} \tag{16}$$

where σ denotes the sigmoid activation function and ReLU denotes the ReLU activation function.

Appendix J. Feature Relevance

In this section, we offer substantive insights derived from our work on *Feature Relevance* (FR), as defined in Def. 2, which measures the contribution of nodal features to neighbor importance. Features are weighted according to their predictive powers for the underlying business and user relationships in the latent space. These predictive relationships, in turn, affect ratings.

Figure J1a and Figure J1d show the feature relevance (as a sum of the self and neighbor relevance) of all business auxiliary information for ON and PA. We observe that some common features are relevant for both PA and ON in finding important neighbors, including operation hours on different days of the week, business category (e.g., restaurants), and business attributes (e.g., attire, bike parking, WiFi, and offering takeout). Operation hours reveal whether they attract similar consumers; hence, similar operation hours further reflect their similarities. For instance, many high-end restaurants are open only for dinner and have very high ratings. In contrast, many fast-food restaurants are open 24 hours a day, but their ratings are low. Moreover, operation hours are the most-often completed features among the auxiliary information; hence, they explain more variations. Meanwhile, unique features exist, which shows the spatial heterogeneity of feature relevance across different regions.

For ON, we observe that operation hours on different days of the week, as well as some attributes and categories, are more important than others. Operation hours are more relevant to rating prediction, reflecting the nature of Yelp as a local recommendation platform. In comparison, check-ins and implicit features are less relevant. Operation hours reveal whether businesses attract similar consumers; hence, similar operation hours further reflect business similarities. For instance, many high-end restaurants are open only for dinner and have very high ratings. In contrast, many fast-food restaurants are open 24 hours a day, but their ratings are low. Moreover, operation hours are the most-often completed feature in the auxiliary information; hence, they explain more variations.

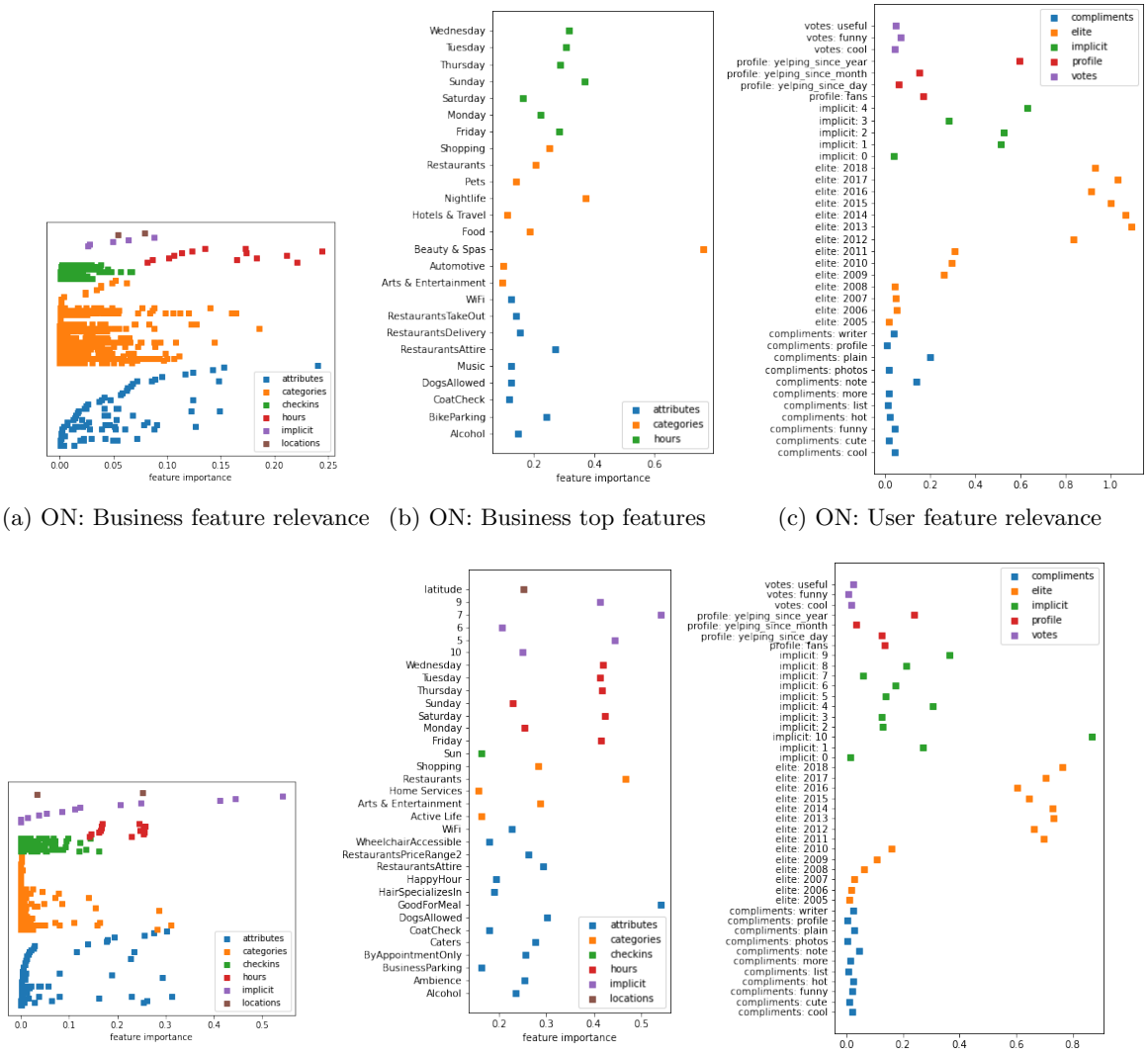
For PA, good for groups, reservations, table service, good for kids, and alcohol are important. These observations show two insights: (1) If Yelp is going to launch the service in a new region, it should first collect the shared features, as mentioned; and (2) because of the state-specific (i.e., spatial) heterogeneity, Yelp may want to learn state-specific feature relevance patterns with MG-GAT after some data have been collected.

Figure J1b and Figure J1e display the top features for ON and PA, respectively. We observe that some common features are important for both regions in finding important neighbors, including operation hours on different days of the week, business category (e.g., restaurants), and business attributes (e.g., attire, bike parking, WiFi, and offering takeout). Meanwhile, unique features also exist for each region, showing the spatial heterogeneity of feature relevance across regions. For example, more business categories are predictive in ON (e.g., beauty & spas); in addition, Friday as the best nights, delivery, and dogs allowed also are important for ON. For PA, good for groups, reservations, table service, good for kids, and alcohol are important. These observations show two insights: (1) If Yelp is going to launch the service in a new region, it should first collect the shared features, as mentioned; and (2) because of the state-specific (i.e., spatial)

heterogeneity, Yelp may want to learn state-specific feature relevance patterns with MG-GAT after some data have been collected.

We perform a similar analysis on user auxiliary information, as shown in Figure J1c and Figure J1f. As expected, elite status contributes more to neighbor importance. Interestingly, the elite status of users is more prominent in both regions after 2012, when Yelp became a public company. In addition, the compliments users receive on their comments (e.g., funny and cool) are not as important as elite status. The reason may be that Yelp already takes compliments into account when awarding elite status. This result suggests that Yelp should recommend new businesses to elite users first to help speed up the learning performance of new businesses. Elite users' ratings are more predictive of others' ratings. Implicit features (colored in green) are more relevant to user neighbor importance than to business neighbor importance. The reason may be that business auxiliary information is richer and can explain more variations than implicit features.[16]

[16] In our dataset, omitting the implicit features does not significantly affect predictive performance. This suggests that the variation explained by these implicit features can be captured by other information in the data.



(a) ON: Business feature relevance (b) ON: Business top features (c) ON: User feature relevance

(d) PA: Feature relevance (e) PA: Top features (f) PA: User feature relevance

Figure J1 Feature relevance for ON and PA.

Note: Business feature relevance contributing to neighbor importance is shown in (a) and (b), respectively, for all features and the top features. User feature relevance contributing to neighbor importance is shown in (c). Larger weights (along the x-axis) correspond to more relevant features in computing the neighbor importance.

Appendix K. Data Collection and Model Integration

K.1. Experimental Data Collection

We conducted a controlled between-subjects experiment to evaluate the effectiveness of different explanation methods in recommendation systems. The design used a vignette-based Yelp scenario, and participants evaluated recommendations on behalf of their closest friend (a third-person/projective framing used for hypothetical judgments) (Fisher 1993, Haire 1950, King and Bruner 2000). We adopt this *third-person technique* (indirect/projective questioning) to (i) reduce self-presentation and socially desirable responding in self-reports and (ii) help participants apply a specified preference profile consistently in a hypothetical scenario (Fisher 1993, Tourangeau and Yan 2007, Krumpal 2013, Aguinis and Bradley 2014). This design choice is distinct from the communication-literature *Third-Person Effect* (self–other persuasion gaps); we therefore do not interpret the results as evidence about differential susceptibility to persuasion. Because the friend-referent framing is held constant across all conditions, our primary inferences are comparative across explanation methods under a common vignette frame. As is standard for vignette experiments, we interpret the outcomes as perceived usefulness and stated engagement intentions in this controlled scenario; examining behavioral impacts in first-person, repeated-use settings is an important direction for future work.

We recruited 1,900 participants through CloudResearch with screening for Yelp familiarity. After removing participants who failed attention checks, our final sample comprised 1,363 participants distributed across five conditions: No Explanation (control), Relevant Business Explanation, Park et al. (2017) social explanation, SHAP Explanation, and MG-GAT Explanation (see Table G1 for condition-level outcomes).

Each participant viewed a vignette with two previously liked restaurants and one focal recommendation. All explanation texts were generated by GPT-4o with comparable length and a neutral tone across treatments. Participants rated Perceived Relevance, Future Interest, and six explanation-quality dimensions on 7-point Likert scales.

K.2. Pilot calibration of explanation-dimension weights

Designers may wish to prioritize certain explanation dimensions depending on the downstream objective (e.g., immediate willingness to click or future interest). When a small calibration sample is available, we illustrate a simple pilot calibration procedure that maps the six standardized explanation-dimension scores to an engagement outcome via regression, and then uses the resulting coefficients as weights to scalarize multi-dimensional plan/explanation scores in Eqs. (11) and (14).

Concretely, we estimate coefficients for transparency, trust, effectiveness, efficiency, persuasiveness, and satisfaction on an independent pilot sample collected prior to the main randomized experiment. We then normalize these coefficients to obtain weights w_k and fix them prior to running the main experiment. This calibration is performed outside the main evaluation sample to avoid outcome leakage. The weighted score used for selection is:

$$\begin{aligned} Q_{\text{explanation}} = {} & \beta_{\text{transparency}} S_{\text{transparency}} + \beta_{\text{trust}} S_{\text{trust}} + \beta_{\text{effectiveness}} S_{\text{effectiveness}} \\ & + \beta_{\text{efficiency}} S_{\text{efficiency}} + \beta_{\text{persuasiveness}} S_{\text{persuasiveness}} + \beta_{\text{satisfaction}} S_{\text{satisfaction}}. \end{aligned}$$

We treat this as a practical scalarization for selecting among already grounded candidates; it does not change MG-GAT's Stage 1 training.