EviSnap: Faithful Evidence-Cited Explanations for Cold-Start Cross-Domain Recommendation

Anonymous ACL submission

Abstract

Cold-start cross-domain recommender (CDR) systems predict a user's preferences in a target domain using only their source-domain behavior, yet existing CDR models either map opaque embeddings or rely on post-hoc or LLM-generated rationales that are hard to audit. We introduce **EviSnap**, a lightweight CDR framework whose predictions are explained by construction with evidence-cited, faithful rationales. EviSnap distills noisy reviews into compact facet cards using an LLM offline, pairing each facet with verbatim supporting sentences. It then induces a shared, domain-agnostic concept bank by clustering facet embeddings and computes user-positive, user-negative, and item-presence concept activations via evidence-weighted pooling. A single linear concept-to-concept map transfers users across domains, and a linear scoring head yields per-concept additive contributions, enabling exact score decompositions and counterfactual 'what-if' edits grounded in the cited sentences. Experiments on the Amazon Reviews dataset across six transfers among Books, Movies, and Music show that EviSnap consistently outperforms strong mapping and review-text baselines while passing deletion- and sufficiency-based tests for explanation faithfulness.

1 Introduction

Real-world recommender systems frequently face cold-start users: people with interaction history in one domain (e.g., Movies) but none in a target domain (e.g., Music or Books). Cold-start cross-domain recommendation (CDR) tackles this by transferring preferences from a source domain to a target domain (Fernández-Tobías et al., 2012; Khan et al., 2017; Zang et al., 2022). However, most CDR systems remain hard to audit. Mapping-based methods learn a transfer function between latent user embeddings and scale well (Man et al., 2017; Hu et al., 2018; Zhu et al., 2022), but the transferred signal is opaque: the model cannot clearly state what preferences were moved or why a target item is recommended. Review-aware models often improve accuracy by encoding text (Zheng et al., 2017; Tay et al., 2018), yet their explanations are typically post-hoc (e.g., attention/highlight rationales) and may not reflect the actual scoring function (Jain and Wallace, 2019; Wiegreffe and Pinter, 2019; Zhang et al., 2020). More recently, LLM-based recommenders can generate fluent justifications (Bao et al., 2023; Mysore et al., 2023; Zhu et al., 2024), but these can be costly at inference and are not guaranteed to be faithful or verifiable (Wu et al., 2024).

We argue that CDR needs an intermediate representation that is (i) shared across domains, (ii) grounded in verifiable evidence, and (iii) used directly by the predictor so that explanations can be tested rather than narrated (Lei et al., 2016; Ross et al., 2017; Rudin, 2019). To this end, we propose **EviSnap**, a lightweight CDR framework that produces faithful, evidence-cited explanations by construction. EviSnap operates in an evidence-grounded concept space built from reviews: an LLM distills raw reviews into compact facet cards (short, domain-agnostic facet phrases) and attaches verbatim supporting sentences. We embed facets from both domains and cluster them into a shared concept bank. For each user we compute separate positive and negative concept activations from praised vs. criticized evidence, and for each target item we compute concept presence activations from its evidence sentences. A single linear concept-to-concept map transfers users from the source concept space to the target space, and an additive linear scoring head produces ratings. Because the score is a sum of per-concept terms, EviSnap yields an exact score decomposition: the explanation is the model itself, paired with the highest-scoring user/item evidence sentences for each surfaced concept. The linear structure also enables transparent counterfactual edits ('what if this concept was

stronger?") with predictable changes in the score.

Our main contributions are:

- **Evidence-cited, domain-agnostic concept representation for CDR.** We introduce an offline facet-card pipeline with verbatim evidence and induce a shared concept bank across domains, yielding sentence-traceable user-positive, user-negative, and item-presence activations.
- **Transparent transfer and faithful explanations by construction.** EviSnap uses a single linear concept mapping and an additive linear scorer, so reported per-concept contributions reconstruct the prediction exactly.
- **Empirical gains with faithfulness diagnostics.** Experiments over the Amazon Reviews dataset (He and McAuley, 2016) among BOOKS, MOVIES, and MUSIC , EviSnap outperforms strong mapping and text-based baselines while passing deletion- and sufficiency-based tests of explanation faithfulness (Lei et al., 2016).

2 Problem Definition

We study *cross-domain recommendation (CDR)* for *cold-start users* across two domains: a source domain S and a target domain T. Let $\mathcal{U}$ be the user set and $\mathcal{I}_S, \mathcal{I}_T$ the item sets in S and T. We observe ratings r_{ui} for some user–item pairs, and we use review text as side information: $\mathcal{R}_S(u)$ denotes the set of source-domain reviews written by user u, and $\mathcal{R}_T(i)$ denotes the set of target-domain reviews written about item i.

We adopt a *user-level cold-start split*: users in training and test are disjoint. At inference time for a test user u, we assume no target-domain history is available, i.e., only $\mathcal{R}_S(u)$ is observed for the user, while item-side text $\mathcal{R}_T(i)$ is available for target items.

Our goal is to learn a predictor g_ϕ that estimates a cold-start user's preference for a target item:

$$\hat{r}_{ui} = g_\phi(u, i \mid \mathcal{R}_S(u), \mathcal{R}_T(i)),$$

where $u \in \mathcal{U},\ i \in \mathcal{I}_T$.

3 Framework

Given the cold-start CDR setting in Section 2, we instantiate g_ϕ with an interpretable, evidence-grounded model that predicts target-domain ratings using a shared concept space as shown in Figure 1. EviSnap (i) distills reviews into evidence-cited facet cards, (ii) induces a shared concept bank and computes user/item concept activations, and (iii) applies a simple linear transfer and linear scoring head. Because the final scorer is additive in concept features, the same quantities that produce $\hat{r}_{ui}$ also yield faithful, sentence-grounded explanations.

In this section, we describe each module and its training objective in turn.

3.1 Generative Facet Card Construction with LLMs

We first preprocess review text into compact, auditable facet cards. For each source-domain user u and each target-domain item i, we input the corresponding bundle of reviews to an LLM and obtain a single JSON object (Figure 2). Each card contains a small set of short, domain-agnostic facet phrases paired with verbatim supporting sentences from the input reviews. Facets are accompanied by a support count.

User cards include facet polarity ($+1$ liked, -1 dislike), while item cards use polarity 0 (items express properties only). We run this extraction offline once and treat facet phrases and evidence sentences as fixed inputs to EviSnap. The LLM is not used during model training or inference. This representation denoises long reviews while preserving traceability: every downstream concept activation can cite an original sentence.

Example. User facets may include *fast pacing* ($+1$) and *slow plot* (-1) with copied evidence sentences; an item may include *live energy* (0) with evidence such as "The live drums give the songs amazing energy."

3.2 Evidence–Grounded Concept Space

This stage builds a shared, interpretable feature space that supports both cross-domain transfer and faithful explanations. We want the model's internal variables to be (i) domain-agnostic so they align across S and T, (ii) evidence-grounded so each activation can be traced to specific sentences, and (iii) simple enough that the final scoring function can decompose cleanly into per-concept contributions. We achieve this by inducing a concept bank from facet phrases and computing user/item concept activations by pooling sentence-to-concept evidence scores.

Concept bank. We induce concepts by clustering facet phrases so that synonymous facets collapse into a single dimension and the same concept labels

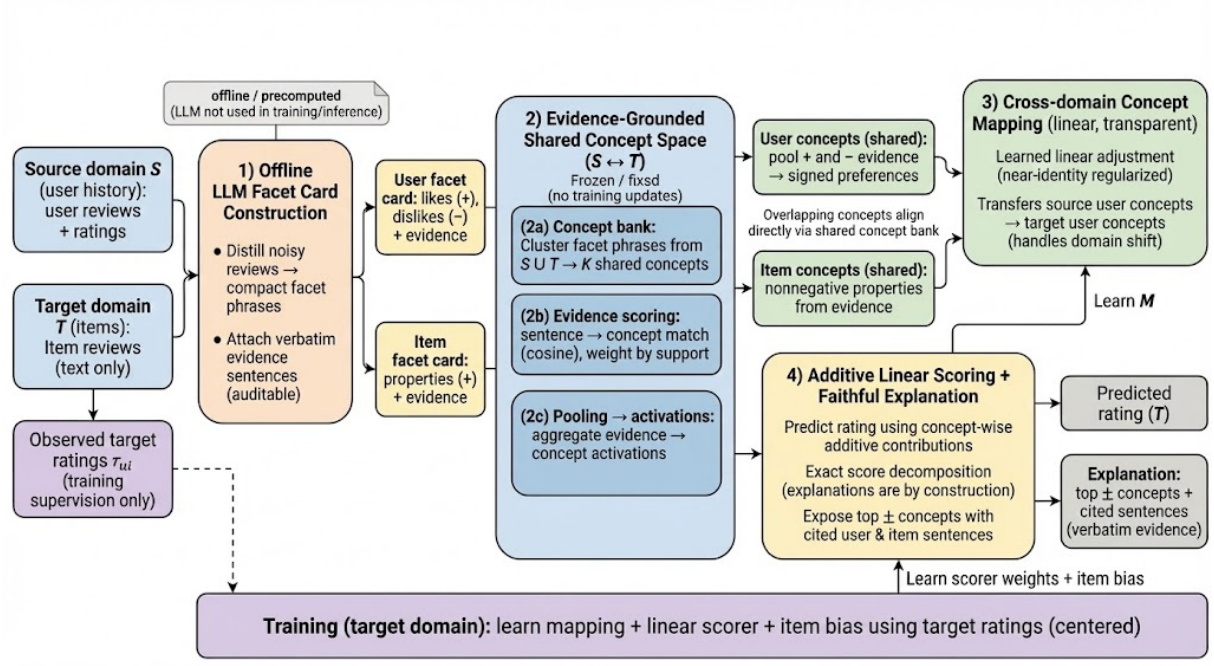

Figure 1: EviSnap overview: offline LLM facet cards with verbatim evidence, a frozen shared concept bank and activations, a near-identity linear transfer map $\mathbf{M}$, additive scoring with exact, evidence-cited per-concept contributions.

can be reused across domains. We embed every distinct facet phrase once using a frozen sentence encoder $f(\cdot)$:

$$\mathbf{e} = f(\text{facet phrase}) \in \mathbb{R}^d$$

We then cluster all facet embeddings with k–means into K clusters. The cluster centroids are our *concept prototypes*

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}_1^\top \\ \vdots \\ \mathbf{d}_K^\top \end{bmatrix} \in \mathbb{R}^{K \times d}, \quad \|\mathbf{d}_k\|_2 = 1 \text{ for all } k$$

Each concept k is labeled by the facet phrase in its cluster whose embedding is closest to $\mathbf{d}_k$. These labels (e.g., "*live energy*", "*vocal clarity*") appear in our explanations.

Sentence–level evidence scores. We score concepts at the sentence level so that explanations can cite the single strongest supporting sentence (while still accounting for multiple mentions).

For any entity e (user or item), we have a small set of evidence sentences

$$\mathcal{S}(e) = \{(s_j, w_j)\}_{j=1}^{N_e},$$

where s_j is an evidence sentence and w_j is a non–negative weight. We use

$$w_j = \log\big(1 + \text{support count}(s_j)\big)$$

so that facets mentioned in more reviews carry slightly more weight.

We embed each sentence,

$$\mathbf{h}_j = f(s_j) \in \mathbb{R}^d, \qquad \|\mathbf{h}_j\|_2 = 1,$$

and compute its alignment with each concept prototype $\mathbf{d}_k$ using cosine similarity followed by ReLU:

$$\alpha_{jk} = \max\left(0,\ \mathbf{h}_j^\top \mathbf{d}_k\right)$$

Here $\alpha_{jk} \in [0, 1]$ measures how strongly sentence s_j supports concept k. Negative or unrelated evidence is clipped to 0.

Pooling over evidence. This pooling choice lets us produce explanations by pointing to the highest-scoring evidence sentence for each concept, while still rewarding concepts supported by multiple reviews.

To obtain a fixed-length concept vector from a variable number of evidence sentences, we aggregate sentence-level alignments into one score per concept. We use weighted log–sum–exp pooling as a smooth max that highlights the strongest supporting sentence without discarding additional supporting evidence:

$$s_k(e) = \frac{1}{\alpha} \log \sum_j \tilde{w}_j \, \exp\big(\alpha \, \alpha_{jk}\big), \quad \tilde{w}_j = \frac{w_j}{\sum_j w_j},$$

(a) System Prompt: Facet Extractor

Role. You are an information extractor for the Amazon Reviews 2014 dataset.

Task. Your task is to turn a bundle of reviews into a small set of domain-agnostic "facets" (stable preference or property phrases) with verbatim evidence sentences copied from the input.

Rules.

- Output JSON only, with keys: `{"meta": {...}, "facets": [ {"facet","polarity","support_count", "evidence":[{"review_id","rating", "unix_time","sentence"}]} ]}`.
- No chain-of-thought. Do not explain your reasoning.
- Facet: $\leq$4 words, lower-case, domain-agnostic. Avoid named entities, brands, plot spoilers, shipping/seller issues.
- Evidence sentences must be copied verbatim from the input; do not paraphrase.
- Merge duplicates/synonyms; prefer facets supported by multiple reviews.
- Polarity: users = +1 like / -1 avoid; items = 0 (items just have facets).

(b) User Prompt Template

Task. You will receive a JSONL bundle of reviews for a single user from the Amazon 2014 dataset (source domain = `{source_domain}`). Goal: extract 6–10 concise preference facets with polarity (+1 like, -1 avoid). Use only the provided text. Output a single JSON object.

Meta field.

Fill `"meta"` as:
`{"mode":"user", "entity_id":"{reviewer_id}", "domain":"{source_domain}"}`

Input format.

```
INPUT (JSONL begins)
{jsonl_bundle}
INPUT END
```

Output requirement.

Produce ONLY the JSON object.

(c) Item Prompt Template

Task. You will receive a JSONL bundle of reviews for a single item from the Amazon 2014 dataset (target domain = `{target_domain}`). Goal: extract 6–12 item facets (polarity = 0). Use only the provided text. Output a single JSON object.

Meta field.

Fill `"meta"` as:
`{"mode":"item", "entity_id":"{asin}", "domain":"{target_domain}"}`

Input format.

```
INPUT (JSONL begins)
{jsonl_bundle}
INPUT END
```

Output requirement.

Produce ONLY the JSON object.

Figure 2: LLM prompts used in our facet-extraction pipeline for the Amazon Reviews 2014 dataset: (a) system prompt defining the facet extraction role and JSON schema; (b) user prompt for user-level preference facets; and (c) item prompt for item-level facets.

where $\alpha > 0$ is a temperature. When α is large, this behaves like a soft maximum over sentences.

User and item concept vectors. We represent preference polarity explicitly for users (likes vs. dislikes) because it affects whether a concept should increase or decrease a target-item score, whereas items only express presence of properties.

For users, facet cards differentiate positive and negative opinions. We pool them separately:

- $\mathbf{U}^+(u) \in \mathbb{R}^K$ from positive evidence sentences (facets the user likes),
- $\mathbf{U}^-(u) \in \mathbb{R}^K$ from negative evidence sentences (facets the user dislikes).

We then form a *signed* source–domain user vector:

$$\mathbf{a}_S(u) = \mathbf{U}^+(u) - \mathbf{U}^-(u)$$

Each entry is high and positive if the user repeatedly praises that concept, and negative if they repeatedly criticize it.

For items in the target domain, facets describe item attributes rather than preferences. Therefore item facets have no polarity (we set polarity to 0) and we pool all item evidence sentences into a nonnegative concept vector:

$$\mathbf{b}(i) \in \mathbb{R}^K_{\geq 0},$$

which encodes which concepts the item offers and how strongly.

3.3 Cross–Domain Concept Mapping

Cold-start users have no history in the target domain, so we must transfer their preferences from S into a target-domain concept representation that can be compared with target items. We choose a single linear map for transparency: its weights directly show which source concepts contribute to which target concepts, making the transfer step itself inspectable rather than a black-box embedding shift.

We now map each user's source domain concept vector to a target domain vector. We adopt a simple linear map

$$\mathbf{a}_T(u) = \mathbf{M}\,\mathbf{a}_S(u), \qquad \mathbf{M} \in \mathbb{R}^{K\times K}$$

Matrix $\mathbf{M}$ is learned from data and initialized to the identity. Intuitively, $\mathbf{M}$ says how source concepts (e.g., "*fast–paced action*") translate into target concepts (e.g., "*live energy*").

To keep $\mathbf{M}$ easy to interpret and to avoid overfitting in the cold–start setting, we regularize it toward the identity:

$$\|\mathbf{M} - \mathbf{I}\|_F^2,$$

where $\mathbf{I}$ is the $K \times K$ identity matrix and $\| \cdot \|_F$ denotes the Frobenius norm.

3.4 Linear Rating Prediction

Given mapped user concepts $\mathbf{a}_T(u)$ and item concepts $\mathbf{b}(i)$, we need a scoring function that is (i) accurate enough to model user-item matching, but also (ii) additively decomposable so explanations can be faithful. We therefore use a linear head over per-concept features: an interaction term captures match (user preference aligned with item property), while marginal terms capture user and item specific tendencies. This keeps inference lightweight and ensures each concept's effect on the score can be computed exactly.

We construct a feature vector using an element wise interaction and optional marginal terms:

$$\begin{aligned} \mathbf{x}^{(\text{int})}(u,i) &= \mathbf{a}_T(u) \odot \mathbf{b}(i) \quad \in \mathbb{R}^K, \\ \mathbf{x}^{(u)}(u,i) &= \mathbf{a}_T(u) \in \mathbb{R}^K, \\ \mathbf{x}^{(i)}(u,i) &= \mathbf{b}(i) \in \mathbb{R}^K, \end{aligned}$$

where $\odot$ denotes element–wise multiplication. We then concatenate them:

$$\mathbf{z}(u,i) = \left[\, \mathbf{x}^{(\text{int})}(u,i) \;\middle|\; \mathbf{x}^{(u)}(u,i) \;\middle|\; \mathbf{x}^{(i)}(u,i) \, \right],$$

where $\mathbf{z}(u,i) \in \mathbb{R}^{3K}$.

A single linear layer computes a *centered* rating score:

$$y_c(u,i) = \mathbf{w}^\top \mathbf{z}(u,i) + b_i,$$

where $\mathbf{w} \in \mathbb{R}^{3K}$ are the head weights and $b_i \in \mathbb{R}$ is an item bias parameter (one scalar per item). The final predicted rating adds back the target–domain mean rating μ_T:

$$\hat{r}_{ui} = \mu_T + y_c(u,i)$$

In practice, we compute the mean rating μ_T over the training portion of the target domain and train on *centered* labels $r_{ui} - \mu_T$. At evaluation time, we clamp $\hat{r}_{ui}$ to the valid rating range (e.g., $[1, 5]$).

Per–concept contributions (for explanation). The head weights $\mathbf{w}$ can be seen as three blocks:

$$\mathbf{w} = \left[\, \mathbf{w}^{(\text{int})} \;\middle|\; \mathbf{w}^{(u)} \;\middle|\; \mathbf{w}^{(i)} \, \right],$$

each of length K. The centered score can then be written as a sum over concepts:

$$\begin{aligned} y_c(u,i) = \sum_{k=1}^{K} \Big(& w_k^{(\text{int})} \, a_{T,k}(u) \, b_k(i) + w_k^{(u)} \, a_{T,k}(u) \\ & + \, w_k^{(i)} \, b_k(i) \Big) + b_i \end{aligned}$$

For explanation, we define the per–concept contribution

$$\begin{aligned} \text{contrib}_k(u,i) = & \, w_k^{(\text{int})} \, a_{T,k}(u) \, b_k(i) \\ & + w_k^{(u)} \, a_{T,k}(u) + w_k^{(i)} \, b_k(i), \end{aligned}$$

and we display the top few positive and negative contrib_k together with the concept label and the best–matching user and item evidence sentences. Crucially, the explanation is *faithful by construction*: there is no separate explanation module—$\text{contrib}_k(u,i)$ is defined to be exactly the k-th additive term in the model's scoring function. Therefore the centered score decomposes exactly as

$$y_c(u,i) = \sum_k \text{contrib}_k(u,i) + b_i,$$

so the reported contributions (plus the item bias b_i and the mean μ_T) reconstruct the predicted rating exactly.

3.5 Training

We train only the transfer and scoring parameters: the cross-domain map $\mathbf{M}$, the linear head weights $\mathbf{w}$, and item biases b_i, so that predicted target-domain ratings fit observed ratings while keeping transfer interpretable and stable. We center ratings to factor out the global target-domain mean, and we apply light regularization to (i) keep $\mathbf{M}$ near-identity to reduce overfitting and preserve concept semantics, and (ii) prevent item biases from dominating the explanation.

Let $\mathcal{D}_{\text{train}} \subset \mathcal{U} \times \mathcal{I}_T$ denote the training set of user–item pairs in the target domain, with observed ratings r_{ui}. We use a user–level cold–start split: users in training and test do not overlap.

We first compute the mean target–domain rating over training data,

$$\mu_T = \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{(u,i) \in \mathcal{D}_{\text{train}}} r_{ui}$$

We then train the model to predict the *centered* rating $r_{ui} - \mu_T$. The main loss term is mean squared error:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{(u,i) \in \mathcal{D}_{\text{train}}} \big(y_c(u,i) - (r_{ui} - \mu_T) \big)^2$$

We add light regularization to stabilize the mapping and keep biases small:

$$\mathcal{L}_{\text{reg}} = \lambda_M \|\mathbf{M} - \mathbf{I}\|_F^2 + \lambda_b \sum_i b_i^2,$$

where λ_M and λ_b are small hyperparameters.

The final training objective is

$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{reg}}$$

We optimize $\mathcal{L}$ with mini–batch stochastic gradient descent (AdamW in our implementation). Because all operations are linear in the learnable parameters (except for the fixed encoder and concept scoring), training is stable and efficient.

4 Experimental Evaluation

We evaluate EviSnap in the cold-start cross-domain recommendation setting on review-driven transfers between MOVIES, BOOKS, and MUSIC. This section describes the dataset construction and evaluation scenarios, and then introduces the baselines used for comparison.

4.1 Dataset and Transfer Scenarios

We use the Amazon Reviews 2014 dataset (He and McAuley, 2016) and consider the BOOKS, MOVIES, and MUSIC domains. Following common practice in cross-domain recommendation, we evaluate all six directed transfers $S \to T$ among these domains.

For each $S \to T$, we restrict to overlapping users who have source-domain text $\mathcal{R}_S(u)$ and at least one observed rating on a target-domain item. We then create a user-level cold-start split by randomly assigning 80% of these users to training and 20% to testing (users are disjoint across splits). At test time, the model observes only $\mathcal{R}_S(u)$ for the user (no target-domain user history), while target-item text $\mathcal{R}_T(i)$ is available for $i \in \mathcal{I}_T$; test ratings are used only for evaluation.

To prevent leakage through item-side text, when constructing $\mathcal{R}_T(i)$ (and extracting target-item facet cards) we exclude all target-domain reviews authored by held-out users, so the model never observes target-domain review text from test users.

4.2 Experimental Setting

We use precomputed facet cards for source-domain users and target-domain items, constructed per $S \to T$ task using the split and leakage prevention protocol in Section 4.1, and treat them as fixed inputs (no LLM calls during model training/evaluation). Facet phrases and evidence sentences are embedded with a frozen sentence encoder, clustered with k-means to form a shared K-concept bank, and pooled into user/item concept activations via evidence-weighted log-sum-exp pooling over ReLU cosine similarities. We train only the linear concept map $\mathbf{M}$ and the additive linear head by minimizing MSE on centered target-domain ratings, regularizing $\mathbf{M}$ toward identity and applying a small ℓ_2 penalty on item biases. We use a single hyperparameter setting for all transfers and do not perform hyperparameter tuning. We ran each scenario 5 times and took the average.

4.3 Baseline Methods

We compare against five cross-domain recommendation baselines that cover both mapping-based transfer and review-text models:

- **EMCDR** (Man et al., 2017): learns latent representations in each domain and trains a mapping function to transfer user representations from S to T.
- **PTUPCDR** (Zhu et al., 2022): It uses a meta-network to generate personalized transfer (bridge) functions for different users.
- **MACDR** (Wang et al., 2024): It employs a prototype-enhanced mixture-of-experts transfer mechanism and distribution alignment to improve robustness under sparse supervision.
- **DeepCoNN+** (Zheng et al., 2017): It is a text-based recommender that models users and items from review text; we use its cross-domain variant with an added mapping layer for transfer.
- **HeroGraph** (Cui et al., 2020): It obtains cross-domain information by a shared graph, which is constructed by collecting users' and items' information from multiple domains.

4.4 Main Results: Cold-Start Cross-Domain Recommendation

Table 1 reports MAE/RMSE on six directed transfers among BOOKS, MOVIES, and MUSIC under the user-level cold-start split. EviSnap achieves the best performance on five of six transfer directions, and is second-best on the remaining MUSIC→MOVIES setting, where HeroGraph is strongest (MAE 0.8020, RMSE 1.0880). Averaged

Table 1: Cross-domain recommendation results. Best and second best are in **bold** and underlined, respectively.

Method	Metric	Books → Music	Books → Movies	Movies → Music	Movies → Books	Music → Movies	Music → Books
EMCDR	MAE	1.2894	0.9701	1.1073	0.9834	0.9860	1.1730
	RMSE	1.5811	1.2461	1.3679	1.2427	1.2856	1.4954
PTUPCDR	MAE	1.0473	0.9453	0.9384	0.9278	0.9642	1.0809
	RMSE	1.3794	1.2326	1.2562	1.2073	1.2760	1.4276
MACDR	MAE	0.8397	0.8987	0.8757	0.8790	0.9038	0.8579
	RMSE	1.1042	1.1388	1.1356	1.1376	1.1564	1.0921
HeroGraph	MAE	0.8150	0.8610	0.7980	0.8670	**0.8020**	0.8860
	RMSE	1.0260	1.1180	1.1010	1.1330	**1.0880**	1.1210
DeepCoNN+	MAE	0.8064	0.8462	0.8413	0.7980	0.9254	0.8548
	RMSE	1.0514	1.0919	1.0953	1.0180	1.1777	1.0736
EviSnap	MAE	**0.7882**	**0.8205**	**0.7768**	**0.7916**	0.8990	**0.8298**
	RMSE	**1.0243**	**1.0696**	**1.0438**	**1.0056**	1.1427	**1.0446**

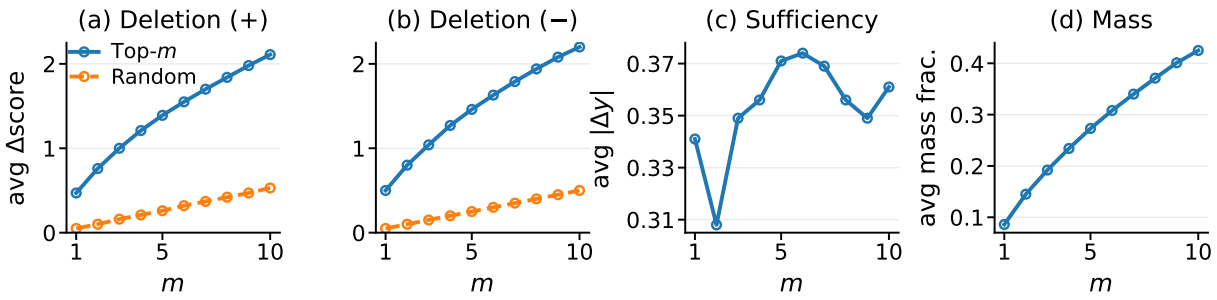

Figure 3: Faithfulness diagnostics on MUSIC→MOVIES. (a) positive deletion vs. random, (b) negative deletion vs. random, (c) sufficiency ($|y_{\text{full}} - y_m|$), (d) contribution mass.

over transfers, EviSnap improves over the strongest review-text baseline DeepCoNN+ from 0.845 to 0.818 MAE and from 1.085 to 1.055 RMSE (relative 3.3% and 2.7%), and yields larger gains over the best mapping-based baseline MACDR (6.6% MAE, 6.4% RMSE). The largest improvements occur on MOVIES→MUSIC, where EviSnap reduces MAE from 0.8413 to 0.7768 and RMSE from 1.0953 to 1.0438. Overall, these results suggest that transferring users through an evidence-grounded concept space is competitive for cold-start CDR while maintaining an additive structure that directly supports faithful, sentence-grounded explanations.

4.5 Ablation: Linear-Head Feature Blocks

Our scorer is designed to be both accurate and decomposable into per-concept effects. To isolate whether gains come primarily from *concept-level matching* or from *marginal* user/item signals, we ablate which concept-feature blocks are fed into the linear head, while keeping the concept bank, pooling, mapping $\mathbf{M}$, item bias, and training objective fixed.

Given mapped user concepts $\mathbf{a}_T(u) \in \mathbb{R}^K$ and item concepts $\mathbf{b}(i) \in \mathbb{R}^K_{\geq 0}$, we define three blocks:

$$\mathbf{x}^{(\text{int})}(u,i) = \mathbf{a}_T(u) \odot \mathbf{b}(i),$$
$$\mathbf{x}^{(u)}(u,i) = \mathbf{a}_T(u),$$
$$\mathbf{x}^{(i)}(u,i) = \mathbf{b}(i)$$

We then form the head input using binary switches $\delta_u, \delta_i \in \{0,1\}$:

$$\mathbf{z}(u,i) = \left[\, \mathbf{x}^{(\text{int})}(u,i) \;\middle|\; \delta_u\, \mathbf{x}^{(u)}(u,i) \;\middle|\; \delta_i\, \mathbf{x}^{(i)}(u,i) \,\right]$$

This yields four variants: **IntOnly** ($\delta_u{=}0, \delta_i{=}0$), **Int+User** $(1,0)$, **Int+Item** $(0,1)$, and **Full** $(1,1)$.

On MUSIC to MOVIES (other transfer directions show similar trends), **Full** performs best, reducing error relative to **IntOnly** by 0.0157 MAE and

Variant	Blocks in $\mathbf{z}(u,i)$			MAE↓	RMSE↓
	$\mathbf{x}^{(\text{int})}$	$\mathbf{x}^{(u)}$	$\mathbf{x}^{(i)}$		
IntOnly	1	0	0	0.9170	1.1522
Int+User	1	1	0	0.9155	1.1509
Int+Item	1	0	1	<u>0.9079</u>	<u>1.1504</u>
Full	1	1	1	**0.9013**	**1.1468**

Table 2: Linear-head block ablation on Music to Movies (K=128). Best/second-best are in **bold**/<u>underline</u>.

0.0054 RMSE, indicating that marginal blocks provide additional signal beyond pure element-wise matching.

4.6 Faithfulness Diagnostics

Because $y_c(u,i)$ is additive in concept terms (Section 3), we can test faithfulness via direct concept-space interventions. On MUSIC to MOVIES (other transfers have similar behaviors), we report (i) *deletion*: ablate the top-m positive/negative concepts ranked by contrib_k and measure the resulting score change, using random deletions as a control; and (ii) *sufficiency*: keep only the top-m concepts by $|\text{contrib}_k|$ and compute the residual $|y_c - y_c^{(m)}|$. Figure 3 shows that top-m deletion perturbs predictions far more than random, while a small set of top concepts largely reconstructs y_c, supporting that the surfaced concepts are the model's true drivers and that decisions are distributed across multiple facets.

4.7 Qualitative Analysis

Table 3 illustrates a cold-start MOVIES to MUSIC explanation for user `ALLHLOG4NLA0A` and item `B0007SMCWY`. Each row corresponds to a concept k in the shared concept bank. The reported score is the exact additive term $\text{contrib}_k(u,i)$ in our linear head, so positive (negative) values raise (lower) the predicted rating by that amount. For each concept, we cite the highest-alignment verbatim sentence from the user's MOVIES reviews and the item's MUSIC reviews, making the transfer auditable at the sentence level. In this case, the prediction is supported by aligned evidence for *musicianship/deep cuts* (the user values technical skill. the album provides many lesser-known performances), as well as complementary signals of *great value* and *nostalgia*. For readability, we show only the largest-magnitude contributions.

5 Related Work

Cold-start cross-domain recommendation (CDR) transfers a user's signal from a source to a target domain, typically by learning domain-specific representations and a mapping/bridge function (e.g., EMCDR/CoNet and later personalized or prototype/mixture-based transfer) to cope with sparse supervision (Zang et al., 2022; Man et al., 2017; Hu et al., 2018; Zhu et al., 2022; Wang et al., 2024). Reviews provide fine-grained signals via aspect-style models and neural review encoders (Zheng et al., 2017; Tay et al., 2018), but many "explanations" in review-aware recommenders are post-hoc (e.g., attention/highlights) and need not reflect the true scoring rule (Jain and Wallace, 2019; Wiegreffe and Pinter, 2019); similarly, LLM-generated justifications can be fluent yet weakly coupled to the predictor and hard to audit (Bao et al., 2023; Wu et al., 2024). EviSnap follows interpretability-by-construction (Rudin, 2019): it distills reviews into facet cards with verbatim evidence and predicts in an additive concept space with a linear transfer, yielding exact per-concept contributions that can be directly validated via deletion/sufficiency interventions (Lei et al., 2016; Ross et al., 2017).

Table 3: One MOVIES→MUSIC explanation

Concept (contrib)	Cited evidence (user → item)
musicianship/deep cuts (+0.45)	*User:* A MUST FOR EVERY MUSICIAN! *Item:* The 38 tracks are non-sequential and they include many lesser-known performances – both originals and covers – in addition to most of the well-known hits,
great value (+0.32)	*User:* This is worth every singe penny. *Item:* But the price can't be beat: the set is currently available new from Amazon Marketplace vendors for around \$5 (plus shipping).
nostalgia (+0.28)	*User:* It brings me back to the 80s when you bought an Lp and the WHOLE LP was very good. *Item:* I am a hugh Al Green fan and this set brings back a lot of memories and still makes me want to dance and grove.

6 Conclusion

We propose EviSnap, a lightweight cold-start cross-domain recommender that transfers users through an evidence-grounded concept space derived from reviews. An offline LLM distills reviews into facet cards with verbatim evidence, and a shared concept bank with a linear map and additive head yields ratings with exact, evidence-cited per-concept contributions. Across six Amazon Reviews transfers among BOOKS, MOVIES, and MUSIC, EviSnap outperforms SOTA baselines, and shows the faithfulness of the surfaced concepts transfers.

7 Limitations

EviSnap relies on an offline LLM step to distill reviews into facet cards. While evidence sentences are verbatim, the extracted facet phrases and user polarity labels may be sensitive to the specific LLM and prompts and may inherit noise or biases from the model and data. Our approach also assumes sufficient review text for both source-domain users and target-domain items. Performance and explanation quality may degrade in text-sparse settings where users/items have few or no reviews. Finally, interpretability is enabled by an unsupervised k-means concept bank and a linear mapping/additive head. Concept quality can depend on the encoder and the choice of K, and the linear form may miss higher-order interactions that could benefit some transfers.

References

Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2023. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM conference on recommender systems*, pages 1007–1014.

Qiang Cui, Tao Wei, Yafeng Zhang, and Qing Zhang. 2020. Herograph: A heterogeneous graph framework for multi-target cross-domain recommendation. In *ORSUM@ RecSys*.

Ignacio Fernández-Tobías, Iván Cantador, Marius Kaminskas, and Francesco Ricci. 2012. Cross-domain recommender systems: A survey of the state of the art. In *Spanish conference on information retrieval*, volume 24. sn.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *proceedings of the 25th international conference on world wide web*, pages 507–517.

Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. Conet: Collaborative cross networks for cross-domain recommendation. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 667–676.

Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. *arXiv preprint arXiv:1902.10186*.

Muhammad Murad Khan, Roliana Ibrahim, and Imran Ghani. 2017. Cross domain recommender systems: A systematic literature review. *ACM Computing Surveys (CSUR)*, 50(3):1–34.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117.

Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. 2017. Cross-domain recommendation: An embedding and mapping approach. In *IJCAI*, volume 17, pages 2464–2470.

Sheshera Mysore, Andrew McCallum, and Hamed Zamani. 2023. Large language model augmented narrative driven recommendations. In *Proceedings of the 17th ACM conference on recommender systems*, pages 777–783.

Andrew Slavin Ross, Michael C Hughes, and Finale Doshi-Velez. 2017. Right for the right reasons: Training differentiable models by constraining their explanations. In *IJCAI*.

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

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2018. Multi-pointer co-attention networks for recommendation. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2309–2318.

Zihan Wang, Yonghui Yang, Le Wu, Richang Hong, and Meng Wang. 2024. Making non-overlapping matters: An unsupervised alignment enhanced cross-domain cold-start recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 11–20.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, and 1 others. 2024. A survey on large language models for recommendation. *World Wide Web*, 27(5):60.

Tianzi Zang, Yanmin Zhu, Haobing Liu, Ruohan Zhang, and Jiadi Yu. 2022. A survey on cross-domain recommendation: taxonomies, methods, and future directions. *ACM Transactions on Information Systems*, 41(2):1–39.

Yongfeng Zhang, Xu Chen, and 1 others. 2020. Explainable recommendation: A survey and new perspectives. *Foundations and Trends® in Information Retrieval*, 14(1):1–101.

Lei Zheng, Vahid Noroozi, and Philip S Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the tenth ACM international conference on web search and data mining*, pages 425–434.

Yaochen Zhu, Liang Wu, Qi Guo, Liangjie Hong, and Jundong Li. 2024. Collaborative large language model for recommender systems. In *Proceedings of the ACM Web Conference 2024*, pages 3162–3172.

Yongchun Zhu, Zhenwei Tang, Yudan Liu, Fuzhen Zhuang, Ruobing Xie, Xu Zhang, Leyu Lin, and Qing He. 2022. Personalized transfer of user preferences for cross-domain recommendation. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, pages 1507–1515.