LEARNING ROBUST RECOMMENDERS FROM NOISY IMPLICIT FEEDBACK VIA GMM-WEIGHTED BAYESIAN TRANSITION MATRIX

Anonymous authors
Paper under double-blind review

ABSTRACT

Label noise is a central challenge in learning from implicit feedback for recommendation. Conventional approaches discard noisy examples for robustness, but this sacrifices data efficiency. Unlike filtering approaches, Bayes-label transition matrix (BLTM) based methods keep all data, but their transition matrix estimates are skewed in practice. To reduce this skew, we introduce GMM-weighted Bayes-label Transition Matrix (RGBT), which augments BLTM with GMM-based instance weights. A GMM assigns each instance a reliability score, and these scores calibrate the BLTM to reduce bias. We show theoretically that RGBT retains all samples, yields consistent estimates, and provably reduces variance compared to CLTM, generalizing to any method with deterministic labels and non-negative weights. Experiments on real and synthetic datasets show that RGBT handles noisy samples more effectively than sample-selection methods, and calibrates the transition matrix more accurately than existing approaches.

1 INTRODUCTION

Recommender systems have advanced considerably in recent years Liu et al. (2024b); Wang et al. (2023b); Li et al. (2024b); Yin et al. (2025). Most of these systems rely on implicit feedback clicks, purchases, and similar signals to infer user preferences, assuming these behaviors reflect interests Bai et al. (2024); Cai et al. (2024); Sun et al. (2024); Liu et al. (2024a); Wang et al. (2024); Krauth et al. (2025). However, notable exceptions exist. For instance, a user might purchase an item only to return it later, or click on a video but exit almost immediately. In such cases, the interactions do not reflect genuine user interest, and are often referred to as noisy interactions in the literature Wang et al. (2021; 2022); Gao et al. (2022); Wang et al. (2023c); He et al. (2024).

Existing denoising methods typically address noisy implicit feedback by directly discarding unreliable samples to ensure robustness, yet this strategy inevitably leads to suboptimal data utilization Wang et al. (2021; 2022); Gao et al. (2022); Wang et al. (2023c); He et al. (2024). For instance, ADT Wang et al. (2021) adaptively prunes or down-weights high-loss interactions; DeCA Wang et al. (2022) aligns model predictions while maximizing data likelihood by leveraging cross-model agreement; SGDL Gao et al. (2022) employs a self-guided meta-learning strategy that collects reliable signals early in training to guide subsequent denoising; BOD Wang et al. (2023c) formulates the denoising task as a bi-level optimization problem efficiently solved via a weight generator; and DCF He et al. (2024) implements a dual-correction mechanism through stabilized sample dropping and progressive label refinement. While these methods have demonstrated certain effectiveness in practice, they either fail to adequately address inherent biases in noise distribution modeling or underutilize the informational value present in noisy samples. More critically, the majority of these approaches are heuristic in nature, lacking the theoretical foundations to guarantee that the denoised model converges to the optimal cleaner.

Estimating the noise transition matrix is a key challenge in learning from noisy implicit feedback. BLTM-based methods Yang et al. (2022) can use all data in theory, but their estimates are often biased in practice. Prior work has explored various strategies, including consistency modeling Ye & Lu (2024), matrix factorization Yao et al. (2020), anchor-free estimation Xia et al. (2019), volume minimization Li et al. (2021), cycle-consistency regularization Cheng et al. (2022b), instance-dependent

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

estimation Wang et al. (2025), and self-supervised learning Lin et al. (2025). Yet none work well with complex real-world noise. The resulting bias can then hurt the final recommendation model.

To address these issues, we propose RGBT, which integrates GMM-based weighting with transition matrix correction. Our main contributions are as follows.

1) We develop a robust recommendation framework that effectively leverages noisy samples through Bayesian label transition matrix (BLTM) modeling, while achieving more accurate bias correction in transition matrix estimation via Gaussian Mixture Model (GMM)-based calibration.

2) We prove theoretically that RGBT uses all data, yields consistent estimates, and has lower variance than existing transition-matrix methods, extending beyond GMM-based calibration to any method with deterministic labels and non-negative weights.

3) Experiments on real and synthetic datasets show that RGBT handles noise better than sample-selection baselines and calibrates the transition matrix more accurately than existing approaches.

2 PRELIMINARIES

Consider a recommendation system with m users and n items. Let $Y \in \{1, \dots, K\}^{m \times n}$ be the clean user–item rating matrix, where y_{ui} is the true rating of user u for item i. In practice, we only observe a noisy version $\widetilde{Y} \in \{1, \dots, K\}^{m \times n}$, in which each entry is flipped from its true value to another rating with probabilities that depend on the instance.

We learn user embeddings $\mathbf{p}_u \in \mathbb{R}^d$ and item embeddings $\mathbf{q}_i \in \mathbb{R}^d$. For each user-item pair (u, i), we form a feature vector $\mathbf{x}_{ui} = [\mathbf{p}_u; \mathbf{q}_i] \in \mathbb{R}^{2d}$. The observed label $\tilde{y}_{ui} \in \{1, \dots, K\}$ is noisy. This yields a K-class classification problem with noisy labels.

For robustness, we keep only high-confidence predictions as distilled samples. Let $\hat{\boldsymbol{\eta}}(\mathbf{x}_{ui}) \in [0, 1]^K$ be the estimated class probability vector. Given $\rho_{\max} \in [0, 1]$ as the maximum noise rate threshold, the distilled label is defined as:

$$y^*_{ui} = \begin{cases} \arg\max_k \hat{\eta}_k(\mathbf{x}_{ui}) & \text{if } \max_k \hat{\eta}_k(\mathbf{x}_{ui}) > \frac{1+\rho_{\max}}{2}, \\ \varnothing & \text{otherwise.} \end{cases} \tag{1}$$

The resulting set $\mathcal{D}_{\text{distilled}} = \{(\mathbf{x}_i, \tilde{y}_i, y^*_i)\}_{i=1}^M$ keeps only predictions above this threshold, where M is the number of retained instances and y^*_i is the inferred Bayes optimal label.

Under the Instance-Dependent Noise (IDN) assumption, the noise transition matrix depends on both the true label and the instance features. We define $\mathbf{T}(\mathbf{x}) \in [0, 1]^{K \times K}$ with entries

$$T_{ij}(\mathbf{x}) = \mathbb{P}(\widetilde{Y} = j \mid Y = i, \mathbf{X} = \mathbf{x}), \quad i, j \in \{1, \dots, K\}, \tag{2}$$

where for each i and $\mathbf{x}$, $T_{ij}(\mathbf{x}) \geq 0$ and $\sum_{j=1}^K T_{ij}(\mathbf{x}) = 1$.

With the distilled samples, we define the Bayes-label transition matrix (BLTM) under IDN as

$$T^*_{ij}(\mathbf{x}) = \mathbb{P}(\widetilde{Y} = j \mid Y^* = i, \mathbf{X} = \mathbf{x}), \tag{3}$$

where $Y^* = \arg\max_y \mathbb{P}(Y = y \mid \mathbf{x})$ is the Bayes optimal label. This matrix is estimated by a transition network with parameters θ:

$$\hat{T}^*_{ij}(\mathbf{x}; \theta) = \mathbb{P}(\widetilde{Y} = j \mid Y^* = i, \mathbf{X} = \mathbf{x}; \theta), \quad \mathbf{x} \in \mathcal{D}_{\text{distilled}}. \tag{4}$$

A GMM is used to assess reliability. With C components and parameters $\Theta = \{(\pi_c, \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)\}_{c=1}^C$, the weight for a feature vector $\mathbf{x}$ is

$$w(\mathbf{x}) = \frac{\pi_r \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r)}{\sum_{c=1}^C \pi_c \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)}, \tag{5}$$

where component r is selected as the reliable one (e.g., with the largest $\|\boldsymbol{\mu}_c\|^2$). These weights then calibrate the transition matrix.

We learn a classifier $f : \mathcal{X} \to \{1, \dots, K\}$ from noisy data. The goal is to minimize the expected risk $\mathbb{E}[\mathbb{I}(f(\mathbf{X}) \neq Y)]$ with respect to the clean distribution, using distilled samples and reliability-weighted BLTM estimation.

3 METHOD

3.1 ROBUST GMM-WEIGHTED BAYES-LABEL TRANSITION FRAMEWORK

Label noise makes it hard to learn from implicit feedback in recommender systems. RGBT handles this with three components: BLTM for data utilization via distilled samples, GMM for reliability, and GMM-calibrated transition matrix estimation.

3.1.1 BLTM WITH DISTILLED SAMPLES FOR ENHANCED DATA UTILIZATION

Traditional denoising methods discard noisy instances and sacrifice data efficiency. We avoid this with the Bayes-label transition matrix (BLTM) using distilled samples. Distilled examples $(\mathbf{x}, \tilde{y}, y^*)$ are obtained by keeping high-confidence predictions. Let $\hat{\boldsymbol{\eta}}(\mathbf{x}) \in [0,1]^K$ be the estimated class probability vector and $\rho_{\max}$ the maximum noise rate threshold. The distilled label is

$$y^* = \begin{cases} \arg\max_k \hat{\eta}_k(\mathbf{x}) & \text{if } \max_k \hat{\eta}_k(\mathbf{x}) > \frac{1+\rho_{\max}}{2}, \\ \varnothing & \text{otherwise.} \end{cases} \tag{6}$$

Keeping only predictions above this threshold yields $\mathcal{D}_{\text{distilled}} = \{(\mathbf{x}_i, \tilde{y}_i, y_i^*)\}_{i=1}^M$.

Under the instance-dependent noise assumption, the BLTM is defined as:

$$T_{ij}^*(\mathbf{x}) = \mathbb{P}(\tilde{Y} = j \mid Y^* = i, \mathbf{X} = \mathbf{x}), \quad i, j \in \{1, \ldots, K\}, \tag{7}$$

where $Y^* = \arg\max_y \mathbb{P}(Y = y \mid \mathbf{x})$ is the Bayes optimal label, $\tilde{Y}$ is the noisy label, and $\mathbf{X}$ is the feature variable.

We train a transition network parameterized by θ to estimate the BLTM using distilled samples:

$$\hat{T}_{ij}^*(\mathbf{x}; \theta) = \mathbb{P}(\tilde{Y} = j \mid Y^* = i, \mathbf{X} = \mathbf{x}; \theta) \quad \text{for } \mathbf{x} \in \mathcal{D}_{\text{distilled}}. \tag{8}$$

Here $\hat{T}^*(\mathbf{x}; \theta) \in [0,1]^{K \times K}$ is the estimated BLTM, where each row sums to 1.

The base learning objective with distilled samples is:

$$\mathcal{L}_{\text{BLTM}}(\theta) = -\frac{1}{M} \sum_{(\mathbf{x}, \tilde{y}, y^*) \in \mathcal{D}_{\text{distilled}}} \tilde{\boldsymbol{y}}^\top \log \left(\boldsymbol{y}^{*\top} \hat{T}^*(\mathbf{x}; \theta) \right), \tag{9}$$

where $\mathcal{L}_{\text{BLTM}}$ denotes the BLTM loss, $\tilde{\boldsymbol{y}}$ is the one-hot encoding of the observed noisy label $\tilde{y}$, and $\boldsymbol{y}^*$ is the one-hot encoding of the distilled Bayes optimal label y^*.

3.1.2 GMM-BASED RELIABILITY EXTRACTION

BLTM improves data utilization via distilled samples, but sample quality varies. A GMM provides reliability weights for calibration.

For each user-item pair (u, v) in the distilled set $\mathcal{D}_{\text{distilled}}$, we compute the prediction score $\hat{y}_{uv}$ and construct reliability features. Let $\mathbf{p}_u \in \mathbb{R}^d$ and $\mathbf{q}_v \in \mathbb{R}^d$ be user and item embeddings respectively, with d denoting the embedding dimension. The prediction score is:

$$\hat{y}_{uv} = \mathbf{p}_u^\top \mathbf{q}_v. \tag{10}$$

The co-occurrence feature c_{uv} captures neighborhood similarity:

$$c_{uv} = \sum_{u' \in \mathcal{U}_v} \mathbb{I}(|\mathcal{I}_u \cap \mathcal{I}_{u'}| > 1), \tag{11}$$

where $\mathcal{U}_v = \{u : (u, v) \in \mathcal{D}_{\text{train}}\}$ is the set of users who interacted with item v, $\mathcal{I}_u = \{v : (u, v) \in \mathcal{D}_{\text{train}}\}$ is the item set of user u, and $\mathbb{I}(\cdot)$ is the indicator function.

The reliability feature vector is then $\mathbf{x}_{uv} = (\hat{y}_{uv}, c_{uv}) \in \mathbb{R}^2$.

A two-component Gaussian Mixture Model (GMM) models the distribution of reliability features:

$$p(\mathbf{x}|\Theta) = \sum_{k=1}^{2} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \tag{12}$$

where $\Theta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^2$ are GMM parameters: $\pi_k \in [0,1]$ (mixing coefficients with $\sum_k \pi_k = 1$), $\boldsymbol{\mu}_k \in \mathbb{R}^2$ (mean vectors), and $\boldsymbol{\Sigma}_k \in \mathbb{R}^{2\times 2}$ (covariance matrices).

From the fitted GMM, we derive instance-specific reliability weights for distilled samples:

$$w_{uv} = \frac{\pi_{k^*}\mathcal{N}(\mathbf{x}_{uv}|\boldsymbol{\mu}_{k^*}, \boldsymbol{\Sigma}_{k^*})}{\sum_{k=1}^2 \pi_k \mathcal{N}(\mathbf{x}_{uv}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}, \tag{13}$$

where $k^* = \arg\max_{k\in\{1,2\}} \|\boldsymbol{\mu}_k\|^2$ identifies the component with larger norm (assumed to represent reliable samples), and $w_{uv} \in [0,1]$ is the reliability weight for the user-item pair (u, v).

3.1.3 GMM-Calibrated BLTM Estimation and Classification

The final component integrates reliability weights to calibrate BLTM estimation and enhance classification. We reformulate the transition matrix learning with reliability-weighted distilled samples:

$$\mathcal{L}_{\text{calibrated}}(\mathbf{w}) = -\frac{1}{M} \sum_{(\mathbf{x},\tilde{y},y^*)\in\mathcal{D}_{\text{distilled}}} w(\mathbf{x}) \cdot \boldsymbol{y}^{*\top} \log\left(f(\mathbf{x};\mathbf{w})\right), \tag{14}$$

where $\mathcal{L}_{\text{calibrated}}(\mathbf{w})$ is the calibrated loss, $M = |\mathcal{D}_{\text{distilled}}|$ is the number of distilled samples, $w(\mathbf{x}) \in [0,1]$ is the reliability weight from GMM, $\boldsymbol{y}^* \in \{0,1\}^K$ is the one-hot encoding of the distilled Bayes optimal label y^*, and $f(\mathbf{x};\mathbf{w}) : \mathbb{R}^p \to \mathbb{R}^K$ is the classifier that outputs the estimated clean class posterior distribution.

For the final classification, we employ the calibrated BLTM for noise correction. The classifier $f(\mathbf{x};\mathbf{w})$ with parameters $\mathbf{w}$ outputs the estimated clean class posterior distribution. The classification loss is:

$$\mathcal{L}_{\text{class}}(\mathbf{w},\theta) = -\frac{1}{N} \sum_{(\mathbf{x},\tilde{y})\in\mathcal{D}} \tilde{\boldsymbol{y}}^{\top} \log\left(\hat{T}^*(\mathbf{x};\theta)\, f(\mathbf{x};\mathbf{w})\right), \tag{15}$$

where $\mathcal{L}_{\text{class}}(\mathbf{w},\theta)$ is the classification loss, $N = |\mathcal{D}|$ is the total number of training samples, $\mathcal{D}$ is the complete training dataset, $\tilde{\boldsymbol{y}} \in \{0,1\}^K$ is the one-hot encoding of the observed noisy label $\tilde{y}$, and the product $\hat{T}^*(\mathbf{x};\theta)\, f(\mathbf{x};\mathbf{w}) \in \mathbb{R}^K$ yields the predicted noisy class distribution.

The complete RGBT framework jointly optimizes:

$$\mathcal{L}_{\text{RGBT}}(\mathbf{w},\theta) = \mathcal{L}_{\text{class}}(\mathbf{w},\theta) + \lambda \mathcal{L}_{\text{calibrated}}(\mathbf{w}), \tag{16}$$

where $\lambda > 0$ is a balancing hyperparameter that controls the relative weight of the classification and transition matrix learning objectives.

We summarize the RGBT training procedure in Algorithm 1.

3.2 Theoretical Analysis

3.2.1 Noise Model and Bayes Optimality

We work with bounded instance-dependent label noise. Let $\eta_y(\mathbf{x}) = \mathbb{P}(Y = y \mid \mathbf{X} = \mathbf{x})$ be the clean class-posterior, and $\tilde{\eta}_y(\mathbf{x}) = \mathbb{P}(\widetilde{Y} = y \mid \mathbf{X} = \mathbf{x})$ the noisy version. The relation between the two is governed by a transition matrix $T_{y'y}(\mathbf{x}) = \mathbb{P}(\widetilde{Y} = y \mid Y = y', \mathbf{X} = \mathbf{x})$:

$$\tilde{\eta}_y(\mathbf{x}) = \sum_{y'\in\mathcal{Y}} T_{y'y}(\mathbf{x})\eta_{y'}(\mathbf{x}). \tag{17}$$

Under this model, the Bayes optimal classifier is unchanged by the noise. We state this formally below.

Theorem 1 (Bayes optimality under noise) *For any* $\mathbf{x} \in \operatorname{supp}(P_D)$, $g^*_{D^*}(\mathbf{x}) = g^*_D(\mathbf{x}) = \arg\max_y P_D(Y = y|\mathbf{x})$.

Algorithm 1: Robust GMM-Weighted Bayes-Label Transition Training

Input: Noisy dataset $\mathcal{D}$, noise rate $\rho_{\max}$, learning rates $\eta_{\mathbf{w}}, \eta_\theta$, refresh interval t_{ref}
Output: Classifier $f(\cdot; \mathbf{w})$, transition estimator $\hat{T}^*(\cdot; \theta)$
Initialize $\mathbf{w}, \theta$; $\mathcal{D}_{\text{dist}} \leftarrow \emptyset$; $t \leftarrow 0$;
while *not converged* **do**
  if $t \bmod t_{ref} = 0$ *or* $\mathcal{D}_{dist} = \emptyset$ **then**
    Extract distilled examples $(\mathbf{x}, \tilde{y}, y^*)$ from $\mathcal{D}$ where $\max f(\mathbf{x}; \mathbf{w}) > (1 + \rho_{\max})/2$;
    Fit GMM and compute reliability weights $w(\mathbf{x})$ on $\mathcal{D}_{\text{dist}}$;
  end
  Update $\mathbf{w}$ with $\mathcal{L}_{\text{calibrated}}$ using $\mathcal{D}_{\text{dist}}$ weighted by $w(\mathbf{x})$;
  Update θ and $\mathbf{w}$ with $\mathcal{L}_{\text{class}}$ on $\mathcal{D}$;
  $t \leftarrow t + 1$;
end
return $f(\cdot; \mathbf{w}), \hat{T}^*(\cdot; \theta)$

3.2.2 Distillation: Extracting Reliable Labels

Directly estimating the transition matrix from noisy data is difficult because the clean labels are unobserved. To get around this, we first extract a set of high-confidence examples.

We keep predictions that pass a confidence threshold as distilled labels:

$$y^* = \begin{cases} \arg\max_k f_k(\mathbf{x}; \mathbf{w}) & \text{if } \max_k f_k(\mathbf{x}; \mathbf{w}) > \frac{1+\rho_{\max}}{2}, \\ \varnothing & \text{otherwise.} \end{cases} \tag{18}$$

The threshold is chosen so that any retained prediction is better than random, as stated below.

Theorem 2 (Distillation condition) *If* $\max_k f_k(\mathbf{x}; \mathbf{w}) > (1 + \rho_{\max})/2$*, then* $\eta_{y^*}(\mathbf{x}) > 1/2$.

The proof follows from the noise bound $\rho_{\max} < 1/2$ and is deferred to Appendix A.

These distilled labels are not all equally reliable. We use a GMM to assign each sample a weight $w(\mathbf{x})$, which reflects its estimated reliability. The calibrated learning objective is then

$$\mathcal{L}_{\text{calibrated}}(\mathbf{w}) = -\frac{1}{M} \sum_{(\mathbf{x}, \tilde{y}, y^*) \in \mathcal{D}_{\text{distilled}}} w(\mathbf{x})\, \boldsymbol{y}^{*\top} \log f(\mathbf{x}; \mathbf{w}). \tag{19}$$

Minimizing this loss drives the classifier toward the clean posterior, as the next theorem shows.

Theorem 3 (Classifier consistency) *As* $M \to \infty$ *and* $\rho_{\max} \to 0$*, the minimizer of* $\mathcal{L}_{calibrated}$ *in equation 19 satisfies* $f(\mathbf{x}; \hat{\mathbf{w}}) \xrightarrow{p} \eta(\mathbf{x})$.

3.2.3 Learning the Transition Matrix

With the classifier approaching $\eta(\mathbf{x})$, we can estimate the transition matrix. We do this by minimizing

$$\mathcal{L}_{\text{class}}(\mathbf{w}, \theta) = -\frac{1}{N} \sum_{(\mathbf{x}, \tilde{y}) \in \mathcal{D}} \tilde{\boldsymbol{y}}^\top \log \left(\hat{T}^*(\mathbf{x}; \theta) f(\mathbf{x}; \mathbf{w}) \right). \tag{20}$$

The intuition is straightforward: $\hat{T}^* f$ should match the observed noisy posterior $\tilde{\eta}$. Since $f \to \eta$ and $\tilde{\eta} = T^* \eta$, consistency follows.

Theorem 4 (Transition matrix consistency) *Under* $\mathcal{L}_{RGBT} = \mathcal{L}_{class} + \lambda \mathcal{L}_{calibrated}$, $\hat{T}^*(\mathbf{x}; \hat{\theta}) \xrightarrow{p} T^*(\mathbf{x})$.

3.2.4 Variance Reduction

We use distilled labels instead of the plug-in estimates from a noisy-label classifier, as conventional CLTM does. The difference is variance: distilled labels are deterministic given $\mathbf{x}$, whereas plug-in estimates are random.

Theorem 5 (Variance reduction) *Under bounded noise with* $\rho_{\max} < 1/2$*,*

$$\lim_{n\to\infty} n \cdot \mathrm{Var}[\hat{T}_{ij}^{BLTM}(\mathbf{x})] \leq \lim_{n\to\infty} n \cdot \mathrm{Var}[\hat{T}_{ij}^{CLTM}(\mathbf{x})].$$

Appendix A provides the full derivations, the proof of Theorem 5, a finite-sample bound (Corollary 1), and a general condition extending the result beyond GMM weighting (Corollary 2).

4 EXPERIMENTS

We evaluate RGBT on six recommendation backbones and three real-world datasets to answer four questions. RQ1: How does RGBT compare with standard training and denoising methods? RQ2: How accurately does our calibration method estimate transition matrices under synthetic flip noise compared with existing matrix-based approaches, and how does this accuracy translate to debiasing performance? RQ3: How sensitive is RGBT to its calibration and distillation hyperparameters? RQ4: How do reliability weighting and transition correction contribute to recommendation performance?

4.1 EXPERIMENTAL SETTINGS

4.1.1 DATASETS

We use three public datasets: Adressa for news reading, MovieLens for movie ratings, and Yelp for business recommendation (Gao et al., 2022; Wang et al., 2021; He et al., 2024; Wang et al., 2022). Following the DCF preprocessing protocol (He et al., 2024), the recommendation test sets retain interactions with dwell times of at least 10 seconds for Adressa, ratings of five for MovieLens, and ratings higher than three for Yelp. The training and validation sets remain noisy and unfiltered.

4.1.2 NOISE SIMULATION FOR RQ2

Following RRFN (Ye & Lu, 2024), we inject two types of rating flip noise into clean MovieLens data. For a clean rating y among K classes, symmetric flip noise assigns the noisy label $\tilde{y} = k$ with probability $\eta/(K-1)$ for each $k \neq y$, whereas pairwise flip noise assigns $\tilde{y} = (y+1) \bmod K$ with probability η, using zero-based class indices. In both settings, the rating remains unchanged with probability $1-\eta$. We use noise rates $\eta \in \{0.1, 0.2, 0.3, 0.4\}$.

4.1.3 EVALUATION PROTOCOLS

Following the protocol established in prior research, we use the training, validation, and clean test splits with an 8:1:1 ratio for RQ1 and RQ3 (Wang et al., 2021; 2022; Gao et al., 2022; He et al., 2024). For performance evaluation, we adopt the two widely used ranking metrics common in denoising recommendation literature: NDCG@K and Recall@K. Higher scores on both metrics indicate better model performance. Table 1 reports results for $K = 10$ and $K = 50$ across the three datasets.

To quantify noise estimation accuracy in RQ2, we average the estimated instance-dependent transition matrices over the validation interactions and compute the entrywise L_1 discrepancy from the known transition matrix (Cheng et al., 2022a; Lin et al., 2025).

4.1.4 BASELINES

We evaluate RGBT on six backbones: GMF and NeuMF (He & Chua, 2017), NGCF (Wang et al., 2019b), LightGCN (He et al., 2020), XSimGCL (Yu et al., 2024), and CDAE (Wu et al., 2016). For recommendation performance, we compare against Normal (BCE training), WBPR (Gantner et al., 2012), WRMF (Hu et al., 2008), T-CE (Wang et al., 2021), DeCA (Wang et al., 2022), and SGDL (Gao et al., 2022). We adapt TCL (Huang et al., 2023) and C2D (Zheltonozhskii et al., 2022) to recommendation. For transition matrix estimation in RQ2, we additionally consider CCR (Cheng et al., 2022b), Dual-T (Yao et al., 2020), T-Revision (Xia et al., 2019), VolMinNet (Li et al., 2021), RRFN (Ye & Lu, 2024), and CONL (Lin et al., 2025).

4.2 PERFORMANCE COMPARISON (RQ1)

To answer RQ1, Table 1 compares RGBT with baseline models across datasets; additional WRMF, T-CE, and BLTM results are in Appendix B. On MovieLens and Yelp, RGBT improves over Normal on all four reported metrics for all six backbones. On Adressa, the gains depend on the backbone

and metric: NGCF and LightGCN show substantial improvements in Recall@50, while GMF does not consistently outperform the strongest baselines.

Table 1: Performance comparison on six backbones and three datasets. Higher is better. Bold and underlining indicate the best and second-best scores among the methods shown, including ties.

Dataset		Adressa				MovieLens				Yelp			
Backbone	Method	R@10	R@50	N@10	N@50	R@10	R@50	N@10	N@50	R@10	R@50	N@10	N@50
GMF	Normal	**0.1881**	0.2544	0.1137	0.1314	0.0769	0.1987	0.0595	0.0963	0.0261	0.0859	0.0193	0.0362
	WBPR	0.1708	0.2418	0.1019	0.1209	0.0696	0.1918	0.0552	0.0921	0.0205	0.0708	0.0148	0.0290
	DeCA	0.1511	0.2541	0.0976	0.1263	0.0765	0.1837	0.0568	0.0901	0.0241	0.0759	0.0179	0.0327
	SGDL	0.1870	**0.2653**	**0.1139**	**0.1347**	0.0779	0.1991	0.0654	0.1021	0.0227	0.0754	0.0166	0.0316
	TCL	0.1788	0.2556	0.1089	0.1293	0.0399	0.1663	0.0347	0.0722	0.0283	0.0890	0.0205	0.0377
	C2D	0.1639	0.2612	0.1043	0.1301	0.0759	0.1835	0.0574	0.0907	0.0282	0.0911	0.0205	0.0383
	RGBT (Ours)	0.1866	0.2546	0.1138	0.1320	**0.0836**	**0.2124**	**0.0664**	**0.1069**	**0.0302**	**0.1068**	**0.0210**	**0.0427**
NeuMF	Normal	0.2856	0.3628	**0.1790**	**0.1995**	0.0644	0.1892	0.0554	0.0927	0.0231	0.0773	0.0163	0.0317
	WBPR	0.1969	0.3654	0.1300	0.1769	0.0665	0.1996	0.0538	0.0937	0.0185	0.0697	0.0130	0.0274
	DeCA	0.2366	0.3539	0.1537	0.1864	0.0516	0.1605	0.0416	0.0757	0.0205	0.0714	0.0147	0.0290
	SGDL	0.1805	0.2602	0.1107	0.1317	0.0745	0.2033	0.0602	0.0987	0.0270	0.0832	**0.0200**	0.0359
	TCL	0.2832	0.3611	0.1734	0.1943	0.0548	0.1843	0.0445	0.0852	0.0164	0.0538	0.0113	0.0220
	C2D	0.2865	0.3630	0.1749	0.1954	0.0725	0.1859	0.0585	0.0925	0.0157	0.0535	0.0112	0.0220
	RGBT (Ours)	**0.2890**	**0.3686**	0.1756	0.1964	**0.0879**	**0.2151**	**0.0745**	**0.1130**	**0.0272**	**0.0987**	0.0192	**0.0394**
NGCF	Normal	0.1979	0.2603	0.1215	0.1384	0.0646	0.1864	0.0525	0.0886	0.0320	0.0958	0.0229	0.0410
	WBPR	0.1799	0.2458	0.1109	0.1285	0.0688	0.1880	0.0538	0.0906	0.0303	0.0930	0.0219	0.0395
	DeCA	0.1747	0.2328	0.1057	0.1214	0.0403	0.1044	0.0390	0.0587	0.0212	0.0664	0.0156	0.0285
	SGDL	0.1924	0.2546	0.1137	0.1305	0.0769	0.1975	0.0590	0.0949	0.0272	0.0858	0.0195	0.0360
	TCL	0.2738	0.3582	0.1694	0.1933	0.0797	0.1936	0.0645	0.0997	0.0278	0.0871	0.0203	0.0369
	C2D	0.2749	0.3588	0.1688	0.1922	0.0800	0.1902	0.0638	0.0975	0.0235	0.0736	0.0172	0.0314
	RGBT (Ours)	**0.2755**	**0.3680**	**0.1711**	**0.1960**	**0.0847**	**0.2097**	**0.0733**	**0.1118**	**0.0328**	**0.1069**	**0.0233**	**0.0444**
LightGCN	Normal	0.1971	0.2604	0.1204	0.1377	0.0648	0.1857	0.0530	0.0885	0.0321	0.0958	0.0231	0.0411
	WBPR	0.1898	0.2543	0.1164	0.1339	0.0673	0.1911	0.0530	0.0906	0.0306	0.0935	0.0220	0.0397
	DeCA	0.1967	0.2541	0.1218	0.1373	0.0590	0.1698	0.0460	0.0810	0.0214	0.0668	0.0156	0.0286
	SGDL	0.1896	0.2565	0.1123	0.1304	0.0748	0.1981	0.0581	0.0949	0.0290	0.0889	0.0209	0.0381
	TCL	0.1997	0.2601	0.1232	0.1396	0.0738	0.1746	0.0600	0.0910	0.0336	0.1020	0.0244	0.0437
	C2D	0.1893	0.2598	0.1176	0.1366	0.0727	0.1715	0.0594	0.0891	0.0312	0.0926	0.0223	0.0397
	RGBT (Ours)	**0.2562**	**0.4177**	**0.1602**	**0.2002**	**0.0880**	**0.2039**	**0.0767**	**0.1115**	**0.0411**	**0.1275**	**0.0301**	**0.0543**
XSimGCL	Normal	0.1071	0.1851	0.0649	0.0847	0.0231	0.0677	0.0149	0.0287	0.0584	0.1668	0.0435	0.0739
	WBPR	0.1085	0.1877	0.0665	0.0865	0.0253	0.0685	0.0168	0.0309	0.0547	0.1550	0.0410	0.0693
	DeCA	0.1244	0.1648	0.0766	0.0874	0.0488	0.1784	0.0422	0.0796	0.0403	0.1160	0.0306	0.0521
	SGDL	0.1069	0.1867	0.0651	0.0853	0.0211	0.0641	0.0136	0.0275	0.0594	0.1672	0.0440	0.0742
	TCL	0.0871	0.1769	0.0554	0.0781	0.0353	0.0985	0.0257	0.0454	0.0605	0.1721	0.0448	0.0760
	C2D	0.1118	0.1844	0.0681	0.0864	0.0296	0.0848	0.0211	0.0382	**0.0611**	0.1751	**0.0454**	**0.0773**
	RGBT (Ours)	**0.2310**	**0.3118**	**0.1547**	**0.1766**	**0.0882**	**0.2146**	**0.0764**	**0.1150**	0.0598	**0.1761**	0.0436	0.0767
CDAE	Normal	0.2708	0.3656	0.1575	0.1833	0.0761	0.1990	0.0626	0.0996	0.0296	0.0981	0.0219	0.0413
	WBPR	0.2520	0.3616	0.1593	0.1905	0.0740	0.1906	0.0601	0.0960	0.0187	0.0619	0.0136	0.0261
	DeCA	0.2354	0.3576	0.1561	0.1908	0.0438	0.1527	0.0368	0.0715	0.0207	0.0592	0.0153	0.0263
	SGDL	0.2484	0.3735	0.1525	0.1860	0.0664	0.1951	0.0589	0.0971	0.0249	0.0809	0.0179	0.0339
	TCL	0.1861	0.4407	0.1093	0.1760	0.0677	0.2095	0.0541	0.0965	0.0268	0.0861	0.0195	0.0365
	C2D	0.2816	**0.4895**	0.1750	**0.2228**	0.0682	0.2106	0.0628	0.1049	0.0247	0.0826	0.0182	0.0348
	RGBT (Ours)	**0.2886**	0.3726	**0.1761**	0.1986	**0.0902**	**0.2178**	**0.0751**	**0.1130**	**0.0421**	**0.1347**	**0.0305**	**0.0567**

4.3 Matrix Estimation and Debiasing (RQ2)

To answer RQ2, we strictly evaluate the estimation accuracy of the transition matrix using the synthetic flip noise settings described in Section 4.1.2. We measure the L_1 norm discrepancy between the estimated transition matrix $\hat{T}$ and the ground-truth transition matrix T on the MovieLens dataset.

Table 2: Matrix estimation error (mean ± standard deviation) over four backbones and four noise rates. Bold marks the lowest mean shown.

Method	Pairflip	Symmetric
CCR	2.4474 ± 1.1558	2.2820 ± 1.1471
DUAL_T	2.5983 ± 0.9354	2.2781 ± 0.9203
T_REVISION	5.0203 ± 1.0717	4.6194 ± 1.2125
VOLMINNET	6.2523 ± 0.4488	5.4986 ± 1.1517
RRFN	2.2616 ± 1.0923	2.1965 ± 1.0764
CONL	3.4056 ± 0.4232	2.3971 ± 0.7384
RGBT (Ours)	**2.2240 ± 0.9949**	**1.1050 ± 0.6192**

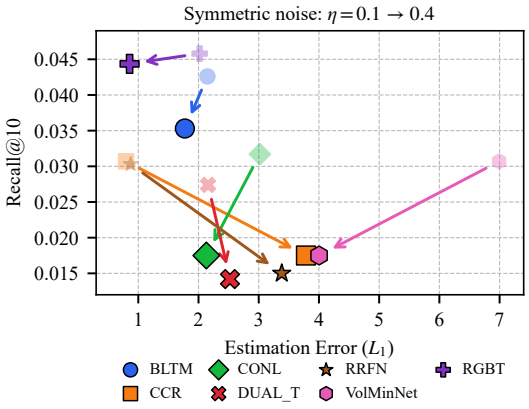


Figure 1: Performance shift under symmetric noise from $\eta = 0.1$ to $\eta = 0.4$.

Table 3: Ablation study on ML-100k. Bold indicates the best performance, including ties.

Model	Variant	Recall		NDCG	
		@10	@50	@10	@50
NeuMF-end	Base	0.0811	0.2099	0.0698	0.1091
	w/o BLTM	0.0824	0.2061	0.0710	0.1088
	w/o GMM	0.0868	**0.2155**	0.0741	0.1125
	Full (Ours)	**0.0879**	0.2151	**0.0745**	**0.1130**
GMF	Base	0.0782	0.2046	0.0643	0.1041
	w/o BLTM	0.0762	0.2053	0.0622	0.1026
	w/o GMM	0.0806	0.2063	**0.0671**	0.1058
	Full (Ours)	**0.0836**	**0.2124**	0.0664	**0.1069**

RGBT achieves the lowest average L_1 error among the methods in Table 2 under both Pairflip noise (2.2240) and Symmetric noise (1.1050).
In Appendix B, Table B4, RGBT achieves the lowest mean error at Pairflip rates of 0.2 and 0.3, and at Symmetric rates of 0.2, 0.3, and 0.4. These results support effective estimation across several noise settings, but do not establish uniform superiority across all settings.
Figure 1 shows that, under Symmetric noise, mean Recall@10 changes from 0.0458 to 0.0444 as η increases from 0.1 to 0.4, while mean matrix error decreases from 2.0184 to 0.8554.

Downstream recommendation also depends on the backbone and noise level. Under Pairflip noise, LightGCN obtains Recall@10 of 0.0568 at $\eta = 0.1$ and 0.0556 at $\eta = 0.4$, whereas NeuMF decreases from 0.0103 to 0.0029. Favorable matrix estimation results therefore do not imply uniform robustness of downstream recommendation across backbones.

4.4 Parameter Sensitivity (RQ3)

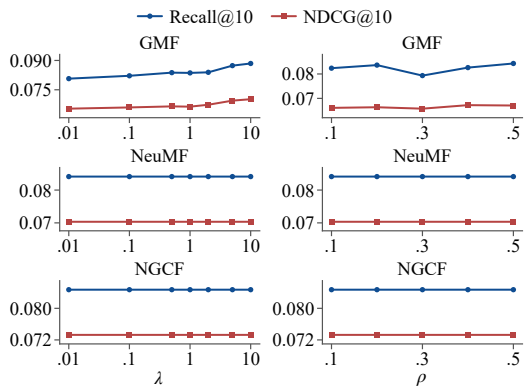


Figure 2: Sensitivity to λ (left, log scale) and ρ (right) on ML-100k.

We investigate the calibrated classifier loss weight λ and the distillation parameter ρ on ML-100k (Figure 2). The confidence threshold is $(1 + \rho)/2$.
For GMF, Recall@10 and NDCG@10 generally increase toward larger calibration weights within the tested range, with the highest reported values at $\lambda = 10$.
The response to ρ is nonmonotonic: GMF's Recall@10 is highest at $\rho = 0.5$, whereas NDCG@10 is highest at $\rho = 0.4$. NeuMF and NGCF show flat curves over both displayed ranges. Thus, the preferred setting depends on the backbone and metric.

4.5 Ablation Study (RQ4)

We compare the full RGBT against Base, which uses standard classifier training, and two variants: (1) *w/o GMM*, which removes reliability weighting; and (2) *w/o BLTM*, which removes transition correction.

The full RGBT outperforms Base on all reported metrics for both backbones (Table 3).

5 Related Work

5.1 Denoising in Recommender Systems

Early work on recommender systems relied on matrix factorization and graph models Koren et al. (2009); Hu et al. (2008); Wu et al. (2016); He et al. (2020); Wang et al. (2019a); Yang et al. (2018). Recent efforts have focused on denoising, with approaches based on pruning Wang et al. (2021); Tian et al. (2022), GMMs Sugiyama et al. (2025), meta-learning Gao et al. (2022); Wang et al. (2023c); Ge et al. (2023), and diffusion Wang et al. (2023a); Zhao et al. (2024). Other work has addressed exposure bias Li et al. (2024a); Cao et al. (2024); He et al. (2024), contextual noise Kim et al. (2014); Lu et al. (2018), and negative sampling Ding et al. (2019a); Wang et al. (2020); Chen et al. (2019). Most denoising methods discard noisy examples, which can reduce data efficiency.

5.2 Denoising in Statistical Methods

Theoretical work includes PAC learning Angluin & Laird (1988), consistent loss correction Liu & Tao (2015), Bayesian ranking Rendle et al. (2012); Gantner et al. (2012); Ding et al. (2019b; 2018), and matrix factorization He et al. (2016). Transition matrix methods have been explored via volume minimization Li et al. (2021), part-dependent Xia et al. (2020b) and dual-T Yao et al. (2020) approximations, instance-dependent models Berthon et al. (2021); Lin et al. (2025); Wang et al. (2025); Menon et al. (2018); Zhang et al. (2024), Bayesian correction Yu & Qin (2020), transition revision Xia et al. (2019), semi-supervised separation Li et al. (2020), bounded noise Cheng et al. (2020), causal learning Yao et al. (2021), Bayes-optimal estimation Yang et al. (2022; 2023), and cycle consistency Cheng et al. (2022b). BLTM-based methods use all data but are often biased.

5.3 Denoising in Machine Learning

Machine learning has produced various noise-robust classification methods, from SVM Natarajan et al. (2013) and PU learning Du Plessis et al. (2014) to loss correction Patrini et al. (2017) and robust training Yuan et al. (2018); Shen & Sanghavi (2019); Xia et al. (2020a); Nguyen et al. (2020); Bai et al. (2022); Zhu et al. (2021); Bai et al. (2021); Wang et al. (2019b). Benchmarks Wei et al. (2022) and causal studies Yao et al. (2023) have also been developed. Theoretical guarantees are limited.

6 Conclusion

We propose RGBT, a framework that uses GMM-based reliability weights to calibrate the Bayes-label transition matrix for recommendation with instance-dependent noise. The GMM assigns each instance a reliability score, and these scores adjust the transition matrix to reduce bias while using all available data. We show theoretically that RGBT uses all data, gives consistent estimates, and reduces variance compared to existing methods, extending beyond GMM-based calibration to any method with deterministic labels and non-negative weights. Experiments on real and synthetic data confirm that RGBT handles noise better than sample-selection baselines and calibrates the transition matrix more accurately than existing approaches. This work suggests that combining reliability weighting with transition matrix estimation is a promising direction for robust recommendation. Future work could extend this framework by incorporating causal modeling of noise feedback.

LLM USAGE DISCLOSURE

We did not use any large language model (LLM) for research tasks such as ideation, experimental design, mathematical reasoning, or code implementation. Generative AI tools were used only for minor language polishing and grammar correction during the paper writing process. All technical content, claims, and artifacts are the sole work of the authors, who take full responsibility for the final paper.

REFERENCES

Dana Angluin and Philip Laird. Learning from noisy examples. *Machine learning*, 2(4):343–370, 1988.

Haoyue Bai, Min Hou, Le Wu, Yonghui Yang, Kun Zhang, Richang Hong, and Meng Wang. Unified representation learning for discrete attribute enhanced completely cold-start recommendation. *IEEE Transactions on Big Data*, 2024.

Yingbin Bai, Erkun Yang, Bo Han, Yanhua Yang, Jiatong Li, Yinian Mao, Gang Niu, and Tongliang Liu. Understanding and improving early stopping for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34:24392–24403, 2021.

Yingbin Bai, Erkun Yang, Zhaoqing Wang, Yuxuan Du, Bo Han, Cheng Deng, Dadong Wang, and Tongliang Liu. Rsa: reducing semantic shift from aggressive augmentations for self-supervised learning. *Advances in Neural Information Processing Systems*, 35:21128–21141, 2022.

Antonin Berthon, Bo Han, Gang Niu, Tongliang Liu, and Masashi Sugiyama. Confidence scores make instance-dependent label-noise learning possible. In *International conference on machine learning*, pp. 825–836. PMLR, 2021.

Miaomiao Cai, Min Hou, Lei Chen, Le Wu, Haoyue Bai, Yong Li, and Meng Wang. Mitigating recommendation biases via group-alignment and global-uniformity in representation learning. *ACM Transactions on Intelligent Systems and Technology*, 15(5):1–27, 2024.

Tianwei Cao, Qianqian Xu, Zhiyong Yang, Zhanyu Ma, and Qingming Huang. Practically unbiased pairwise loss for recommendation with implicit feedback. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Chong Chen, Min Zhang, Chenyang Wang, Weizhi Ma, Minming Li, Yiqun Liu, and Shaoping Ma. An efficient adaptive transfer neural network for social-aware recommendation. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pp. 225–234, 2019.

De Cheng, Tongliang Liu, Yixiong Ning, Nannan Wang, Bo Han, Gang Niu, Xinbo Gao, and Masashi Sugiyama. Instance-dependent label-noise learning with manifold-regularized transition matrix estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16630–16639, 2022a.

De Cheng, Yixiong Ning, Nannan Wang, Xinbo Gao, Heng Yang, Yuxuan Du, Bo Han, and Tongliang Liu. Class-dependent label-noise learning with cycle-consistency regularization. *Advances in Neural Information Processing Systems*, 35:11104–11116, 2022b.

Jiacheng Cheng, Tongliang Liu, Kotagiri Ramamohanarao, and Dacheng Tao. Learning with bounded instance and label-dependent label noise. In *International conference on machine learning*, pp. 1789–1799. PMLR, 2020.

Jingtao Ding, Fuli Feng, Xiangnan He, Guanghui Yu, Yong Li, and Depeng Jin. An improved sampler for bayesian personalized ranking by leveraging view data. In *Companion proceedings of the the web conference 2018*, pp. 13–14, 2018.

Jingtao Ding, Yuhan Quan, Xiangnan He, Yong Li, and Depeng Jin. Reinforced negative sampling for recommendation with exposure data. In *IJCAI*, pp. 2230–2236. Macao, 2019a.

Jingtao Ding, Guanghui Yu, Xiangnan He, Fuli Feng, Yong Li, and Depeng Jin. Sampler design for bayesian personalized ranking by leveraging view data. *IEEE transactions on knowledge and data engineering*, 33(2):667–681, 2019b.

Marthinus C Du Plessis, Gang Niu, and Masashi Sugiyama. Analysis of learning from positive and unlabeled data. *Advances in neural information processing systems*, 27, 2014.

Zeno Gantner, Lucas Drumond, Christoph Freudenthaler, and Lars Schmidt-Thieme. Personalized ranking for non-uniformly sampled items. In *Proceedings of KDD cup 2011*, pp. 231–247. PMLR, 2012.

Yunjun Gao, Yuntao Du, Yujia Hu, Lu Chen, Xinjun Zhu, Ziquan Fang, and Baihua Zheng. Self-guided learning to denoise for robust recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pp. 1412–1422, 2022.

Yingqiang Ge, Mostafa Rahmani, Athirai A Irissappane, Jose Sepulveda, James Caverlee, and Fei Wang. Automated data denoising for recommendation. *CoRR*, 2023.

Xiangnan He and Tat-Seng Chua. Neural factorization machines for sparse predictive analytics. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 355–364, 2017.

Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. Fast matrix factorization for online recommendation with implicit feedback. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 549–558, 2016.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 639–648, 2020.

Zhuangzhuang He, Yifan Wang, Yonghui Yang, Peijie Sun, Le Wu, Haoyue Bai, Jinqi Gong, Richang Hong, and Min Zhang. Double correction framework for denoising recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1062–1072, 2024.

Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE international conference on data mining*, pp. 263–272. Ieee, 2008.

Zhizhong Huang, Junping Zhang, and Hongming Shan. Twin contrastive learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Youngho Kim, Ahmed Hassan, Ryen W White, and Imed Zitouni. Modeling dwell time to predict click-level satisfaction. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pp. 193–202, 2014.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Karl Krauth, Yixin Wang, and Michael Jordan. Breaking feedback loops in recommender systems with causal inference. *ACM Transactions on Recommender Systems*, 4(1):1–20, 2025.

Haoxuan Li, Chunyuan Zheng, Wenjie Wang, Hao Wang, Fuli Feng, and Xiao-Hua Zhou. Debiased recommendation with noisy feedback. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1576–1586, 2024a.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *International Conference on Learning Representations*, 2020.

Xuefeng Li, Tongliang Liu, Bo Han, Gang Niu, and Masashi Sugiyama. Provably end-to-end label-noise learning without anchor points. In *International conference on machine learning*, pp. 6403–6413. PMLR, 2021.

Yang Li, Kangbo Liu, Ranjan Satapathy, Suhang Wang, and Erik Cambria. Recent developments in recommender systems: A survey. *IEEE Computational Intelligence Magazine*, 19(2):78–95, 2024b.

Yexiong Lin, Yu Yao, Zhaoqing Wang, Xu Shen, Jun Yu, Bo Han, and Tongliang Liu. Improving the instance-dependent transition matrix estimation by exploiting self-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Hanyang Liu, Yong Wang, Zhiqiang Zhang, Jiangzhou Deng, Chao Chen, and Leo Yu Zhang. Matrix factorization recommender based on adaptive gaussian differential privacy for implicit feedback. *Information Processing & Management*, 61(4):103720, 2024a.

Qidong Liu, Jiaxi Hu, Yutian Xiao, Xiangyu Zhao, Jingtong Gao, Wanyu Wang, Qing Li, and Jiliang Tang. Multimodal recommender systems: A survey. *ACM Computing Surveys*, 57(2):1–17, 2024b.

Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2015.

Hongyu Lu, Min Zhang, and Shaoping Ma. Between clicks and satisfaction: Study on multi-phase user preferences and satisfaction for online news reading. In *The 41st international acm sigir conference on research & development in information retrieval*, pp. 435–444, 2018.

Aditya Krishna Menon, Brendan Van Rooyen, and Nagarajan Natarajan. Learning from binary labels with instance-dependent noise. *Machine Learning*, 107(8):1561–1595, 2018.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

Duc Tam Nguyen, Chaithanya Kumar Mummadi, Thi Phuong Nhung Ngo, Thi Hoai Phuong Nguyen, Laura Beggel, and Thomas Brox. Self: Learning to filter noisy labels with self-ensembling. In *International Conference on Learning Representations*, 2020.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1944–1952, 2017.

Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*, 2012.

Yanyao Shen and Sujay Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In *International conference on machine learning*, pp. 5739–5748. PMLR, 2019.

Takuto Sugiyama, Soh Yoshida, and Mitsuji Muneyasu. Joint modeling of prediction and behavioral patterns for reliable recommendation with implicit feedback. *IEEE Access*, 2025.

Peijie Sun, Yifan Wang, Min Zhang, Chuhan Wu, Yan Fang, Hong Zhu, Yuan Fang, and Meng Wang. Collaborative-enhanced prediction of spending on newly downloaded mobile games under consumption uncertainty. In *Companion Proceedings of the ACM Web Conference 2024*, pp. 10–19, 2024.

Changxin Tian, Yuexiang Xie, Yaliang Li, Nan Yang, and Wayne Xin Zhao. Learning to denoise unreliable interactions for graph collaborative filtering. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pp. 122–132, 2022.

Wenjie Wang, Fuli Feng, Xiangnan He, Liqiang Nie, and Tat-Seng Chua. Denoising implicit feedback for recommendation. In *Proceedings of the 14th ACM international conference on web search and data mining*, pp. 373–381, 2021.

Wenjie Wang, Yiyan Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. Diffusion recommender model. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*, pp. 832–841, 2023a.

Wenjie Wang, Yang Zhang, Haoxuan Li, Peng Wu, Fuli Feng, and Xiangnan He. Causal recommendation: Progresses and future directions. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*, pp. 3432–3435, 2023b.

Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 950–958, 2019a.

Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pp. 165–174, 2019b.

Xiang Wang, Yaokun Xu, Xiangnan He, Yixin Cao, Meng Wang, and Tat-Seng Chua. Reinforced negative sampling over knowledge graph for recommendation. In *Proceedings of the web conference 2020*, pp. 99–109, 2020.

Yifan Wang, Peijie Sun, Weizhi Ma, Min Zhang, Yuan Zhang, Peng Jiang, and Shaoping Ma. Intersectional two-sided fairness in recommendation. In *Proceedings of the ACM Web Conference 2024*, pp. 3609–3620, 2024.

Yu Wang, Xin Xin, Zaiqiao Meng, Joemon M Jose, Fuli Feng, and Xiangnan He. Learning robust recommenders through cross-model agreement. In *Proceedings of the ACM web conference 2022*, pp. 2015–2025, 2022.

Yuan Wang, Huaxin Pang, Ying Qin, Shikui Wei, and Yao Zhao. Adaptive estimation of instance-dependent noise transition matrix for learning with instance-dependent label noise. *Neural Networks*, 188:107464, 2025.

Zongwei Wang, Min Gao, Wentao Li, Junliang Yu, Linxin Guo, and Hongzhi Yin. Efficient bi-level optimization for recommendation denoising. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 2502–2511, 2023c.

Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. In *International Conference on Learning Representations*, 2022.

Yao Wu, Christopher DuBois, Alice X Zheng, and Martin Ester. Collaborative denoising auto-encoders for top-n recommender systems. In *Proceedings of the ninth ACM international conference on web search and data mining*, pp. 153–162, 2016.

Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. Are anchor points really indispensable in label-noise learning? *Advances in neural information processing systems*, 32, 2019.

Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In *International conference on learning representations*, 2020a.

Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. *Advances in neural information processing systems*, 33:7597–7610, 2020b.

Jheng-Hong Yang, Chih-Ming Chen, Chuan-Ju Wang, and Ming-Feng Tsai. Hop-rec: high-order proximity for implicit recommendation. In *Proceedings of the 12th ACM conference on recommender systems*, pp. 140–144, 2018.

Shuo Yang, Erkun Yang, Bo Han, Yang Liu, Min Xu, Gang Niu, and Tongliang Liu. Estimating instance-dependent bayes-label transition matrix using a deep neural network. In *International Conference on Machine Learning*, pp. 25302–25312. PMLR, 2022.

Shuo Yang, Songhua Wu, Erkun Yang, Bo Han, Yang Liu, Min Xu, Gang Niu, and Tongliang Liu. A parametrical model for instance-dependent label noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):14055–14068, 2023.

Yu Yao, Tongliang Liu, Bo Han, Mingming Gong, Jiankang Deng, Gang Niu, and Masashi Sugiyama. Dual t: Reducing estimation error for transition matrix in label-noise learning. *Advances in neural information processing systems*, 33:7260–7271, 2020.

Yu Yao, Tongliang Liu, Mingming Gong, Bo Han, Gang Niu, and Kun Zhang. Instance-dependent label-noise learning under a structural causal model. *Advances in Neural Information Processing Systems*, 34:4409–4420, 2021.

Yu Yao, Mingming Gong, Yuxuan Du, Jun Yu, Bo Han, Kun Zhang, and Tongliang Liu. Which is better for learning with noisy labels: the semi-supervised method or modeling label noise? In *International conference on machine learning*, pp. 39660–39673. PMLR, 2023.

Shanshan Ye and Jie Lu. Robust recommender systems with rating flip noise. *ACM Transactions on Intelligent Systems and Technology*, 16(1):1–19, 2024.

Hongzhi Yin, Liang Qu, Tong Chen, Wei Yuan, Ruiqi Zheng, Jing Long, Xin Xia, Yuhui Shi, and Chengqi Zhang. On-device recommender systems: A comprehensive survey. *Data Science and Engineering*, pp. 1–30, 2025.

Junliang Yu, Xin Xia, Tong Chen, Lizhen Cui, Quoc Viet Hung Nguyen, and Hongzhi Yin. XSimGCL: Towards extremely simple graph contrastive learning for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 36(2):913–926, 2024. doi: 10.1109/TKDE.2023.3288135.

Wenhui Yu and Zheng Qin. Sampler design for implicit feedback data by noisy-label robust learning. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pp. 861–870, 2020.

Fajie Yuan, Xin Xin, Xiangnan He, Guibing Guo, Weinan Zhang, Chua Tat-Seng, and Joemon M Jose. fbgd: Learning embeddings from positive unlabeled data with bgd. 2018.

Ruiheng Zhang, Zhe Cao, Shuo Yang, Lingyu Si, Haoyang Sun, Lixin Xu, and Fuchun Sun. Cognition-driven structural prior for instance-dependent label transition matrix estimation. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

Jujia Zhao, Wang Wenjie, Yiyan Xu, Teng Sun, Fuli Feng, and Tat-Seng Chua. Denoising diffusion recommender model. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1370–1379, 2024.

Evgenii Zheltonozhskii, Chaim Baskin, Avi Mendelson, Alex M. Bronstein, and Or Litany. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1657–1667, 2022.

Jianing Zhu, Jingfeng Zhang, Bo Han, Tongliang Liu, Gang Niu, Hongxia Yang, Mohan Kankanhalli, and Masashi Sugiyama. Understanding the interaction of adversarial training with noisy labels. *arXiv preprint arXiv:2102.03482*, 2021.

A Theoretical Derivations

This appendix contains the derivations and proofs for the results in Section 3.2. We first derive the CLTM and BLTM estimators from both the calibrated loss and the classification loss, then prove Theorem 5, and finally state the finite-sample bound and a general condition under which the same variance-reduction argument extends beyond GMM weighting.

A.1 CLTM and BLTM Estimators

Our derivation proceeds in two steps. The calibrated loss $\mathcal{L}_{\text{calibrated}}$ first drives the classifier toward the clean posterior $\eta(\mathbf{x})$. The classification loss $\mathcal{L}_{\text{class}}$ then estimates the transition matrix so that $\widehat{T}^* f$ matches the observed noisy posterior $\widetilde{\eta}$. Both steps are needed: without the first, f would remain at the noisy posterior; without the second, the transition matrix would not be identified.

A.1.1 CALIBRATING THE CLASSIFIER

We first extract high-confidence distilled labels Y^* by thresholding, as in equation 18, and train the classifier with the calibrated loss

$$\mathcal{L}_{\text{calibrated}}(\mathbf{w}) = -\frac{1}{M} \sum_{(\mathbf{x},\widetilde{y},y^*)\in\mathcal{D}_{\text{distilled}}} w(\mathbf{x})\, \boldsymbol{y}^{*\top} \log f(\mathbf{x};\mathbf{w}). \tag{21}$$

The distilled labels Y^* are deterministic given $\mathbf{x}$, so by Theorem 3, $f(\mathbf{x};\mathbf{w}) \to \eta(\mathbf{x})$. That is, $\mathcal{L}_{\text{calibrated}}$ turns the classifier from a noisy posterior estimator into a clean posterior estimator.

A.1.2 ESTIMATING THE TRANSITION MATRIX

We begin from the MLE for the transition matrix entry $T_{ij}(\mathbf{x}) = P(\widetilde{Y} = j \mid Y = i, \mathbf{X} = \mathbf{x})$. Given n i.i.d. samples, the negative log-likelihood is

$$-\log \mathcal{L}_{\text{MLE}}(T) = -\sum_{k=1}^{n}\sum_{i=1}^{K}\sum_{j=1}^{K} \mathbb{I}(Y_k = i)\mathbb{I}(\widetilde{Y}_k = j) \log T_{ij}(\mathbf{x}). \tag{22}$$

The true labels Y are unobserved, so we must replace them with estimates. The choice of estimates is what differentiates CLTM from our BLTM.

Equivalently, the same estimator can be derived from the classification loss

$$\mathcal{L}_{\text{class}}(\mathbf{w},\theta) = -\frac{1}{N} \sum_{(\mathbf{x},\widetilde{y})\in\mathcal{D}} \widetilde{\boldsymbol{y}}^{\top} \log\Big(\widehat{T}^*(\mathbf{x};\theta) f(\mathbf{x};\mathbf{w})\Big), \tag{23}$$

which encourages $\widehat{T}^* f$ to match the observed noisy posterior $\widetilde{\eta}$. Taking the first-order condition of equation 23 with respect to the ij-th entry of $\widehat{T}^*$ and normalizing the rows to sum to one yields

$$\widehat{T}_{ij}(\mathbf{x}) = \frac{\sum_{k=1}^{n} c_k\, \mathbb{I}(\widehat{Y}_k = i)\mathbb{I}(\widetilde{Y}_k = j)}{\sum_{k=1}^{n} c_k\, \mathbb{I}(\widehat{Y}_k = i)}, \tag{24}$$

where c_k is the weight assigned to sample k and $\widehat{Y}_k$ is the label used in place of the unobserved Y_k. CLTM and BLTM now correspond to two different choices of $(\widehat{Y}_k, c_k)$.

CLTM estimator. CLTM trains the classifier with standard cross-entropy on noisy labels:

$$\mathcal{L}_{\text{CE}}(\mathbf{w}) = -\frac{1}{N} \sum_{(\mathbf{x},\widetilde{y})\in\mathcal{D}} \widetilde{\boldsymbol{y}}^{\top} \log f(\mathbf{x};\mathbf{w}), \tag{25}$$

so f approximates the noisy posterior $\widetilde{\eta}(\mathbf{x})$. It then uses the random plug-in labels $\widehat{Y}_k = \arg\max f(\mathbf{x}_k;\mathbf{w})$ with uniform weights $c_k = 1$. Substituting into equation 24 gives

$$\widehat{T}^{\text{CLTM}}_{ij}(\mathbf{x}) = \frac{\sum_{k=1}^{n} \mathbb{I}(\widehat{Y}_k = i)\mathbb{I}(\widetilde{Y}_k = j)}{\sum_{k=1}^{n} \mathbb{I}(\widehat{Y}_k = i)}. \tag{26}$$

BLTM estimator. BLTM uses the calibrated classifier from Section A, so that $f(\mathbf{x};\mathbf{w}) \to \eta(\mathbf{x})$. It then uses the deterministic distilled labels $\widehat{Y}_k = Y^*_k$ with reliability weights $c_k = w_k$. Substituting into equation 24 gives

$$\widehat{T}^{\text{BLTM}}_{ij}(\mathbf{x}) = \frac{\sum_{k=1}^{n} w_k\, \mathbb{I}(Y^*_k = i)\mathbb{I}(\widetilde{Y}_k = j)}{\sum_{k=1}^{n} w_k\, \mathbb{I}(Y^*_k = i)}. \tag{27}$$

Thus CLTM and BLTM are the plug-in and distilled solutions of the same classification objective, and the role of $\mathcal{L}_{\text{calibrated}}$ is to ensure that the labels used in the distilled solution come from the clean posterior rather than the noisy one.

A.2 Proof of Theorem 5

We now compare the variances of the two estimators above.

For the CLTM estimator equation 26, the law of total variance gives

$$\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{CLTM}}] = \mathrm{Var}\big[\mathbb{E}[\widehat{T}_{ij}^{\mathrm{CLTM}} \mid \widehat{Y}]\big] + \mathbb{E}\big[\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{CLTM}} \mid \widehat{Y}]\big]. \tag{28}$$

The first term is positive when $\mathrm{Var}[\widehat{Y} \mid \mathbf{x}] > 0$; it reflects the variance introduced by using random plug-in labels.

For the BLTM estimator equation 27, the distilled labels Y^* are deterministic given $\mathbf{x}$, so the analogous decomposition is

$$\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{BLTM}}] = \mathbb{E}\big[\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{BLTM}} \mid Y^*]\big], \tag{29}$$

with $\mathrm{Var}[\mathbb{E}[\widehat{T}_{ij}^{\mathrm{BLTM}} \mid Y^*]] = 0$.

Subtracting equation 29 from equation 28 yields

$$\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{CLTM}}] - \mathrm{Var}[\widehat{T}_{ij}^{\mathrm{BLTM}}] = \mathrm{Var}\big[\mathbb{E}[\widehat{T}_{ij}^{\mathrm{CLTM}} \mid \widehat{Y}]\big] \geq 0. \tag{30}$$

Therefore $\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{BLTM}}] \leq \mathrm{Var}[\widehat{T}_{ij}^{\mathrm{CLTM}}]$, proving the theorem.

A.3 Finite-Sample Bound

Corollary 1 (Finite-sample bound) *With probability at least* $1 - \delta$*,*

$$|\widehat{T}_{ij}^{BLTM}(\mathbf{x}) - T_{ij}^*(\mathbf{x})| \leq \sqrt{\frac{V_{BLTM}(\mathbf{x}) \log(2/\delta)}{2n_{eff}(\mathbf{x})}} + \mathcal{O}\left(\frac{1}{n_{eff}(\mathbf{x})}\right), \tag{31}$$

where $n_{eff}(\mathbf{x}) = \frac{(\sum w_k)^2}{\sum w_k^2}$.

A.4 A General Condition for Variance Reduction

Theorem 5 compares BLTM and CLTM under the specific construction of Section 3.2, in which distilled labels are obtained by confidence thresholding and reliability weights are produced by a GMM. We now state a more general condition under which the same variance-reduction argument continues to hold for other label-extraction and weighting schemes.

Corollary 2 (General variance reduction) *Let* $\widehat{T}_{ij}^{\mathrm{CLTM}}$ *be the CLTM estimator using random plug-in labels* $\widehat{Y} = \arg\max_k f_k(\mathbf{x}; \mathbf{w})$*, and let* $\widehat{T}_{ij}^{\mathrm{ALT}}$ *be a transition-matrix estimator obtained by replacing* $\widehat{Y}$ *with a label* Y^* *produced by some alternative procedure. Suppose that, given* $\mathbf{x}$*, the label* $Y^*(\mathbf{x})$ *is deterministic so that* $\mathrm{Var}[Y^* \mid \mathbf{x}] = 0$*, that the weighting scheme assigns* $w_k \geq 0$ *for every retained sample with at least one positive weight, and that the alternative estimator takes the plug-in form*

$$\widehat{T}_{ij}^{\mathrm{ALT}}(\mathbf{x}) = \frac{\sum_{k=1}^n w_k \, \mathbb{I}(Y_k^* = i) \, \mathbb{I}(\widetilde{Y}_k = j)}{\sum_{k=1}^n w_k \, \mathbb{I}(Y_k^* = i)}.$$

Then

$$\mathrm{Var}\left[\widehat{T}_{ij}^{\mathrm{ALT}}(\mathbf{x})\right] \leq \mathrm{Var}\left[\widehat{T}_{ij}^{\mathrm{CLTM}}(\mathbf{x})\right].$$

Proof 1 *By the law of total variance,*

$$\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{CLTM}}] = \mathrm{Var}\left[\mathbb{E}[\widehat{T}_{ij}^{\mathrm{CLTM}} \mid \widehat{Y}]\right] + \mathbb{E}\left[\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{CLTM}} \mid \widehat{Y}]\right],$$

where the first term is non-negative whenever $\mathrm{Var}[\widehat{Y} \mid \mathbf{x}] > 0$*. The determinism of* Y^* *given* $\mathbf{x}$ *implies that*

$$\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{ALT}}] = \mathbb{E}\left[\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{ALT}} \mid Y^*]\right],$$

and the corresponding first term vanishes: $\mathrm{Var}[\mathbb{E}[\widehat{T}_{ij}^{\mathrm{ALT}} \mid Y^*]] = 0$. *Subtracting the two decompositions yields*

$$\mathrm{Var}[\widehat{T}_{ij}^{\mathrm{CLTM}}] - \mathrm{Var}[\widehat{T}_{ij}^{\mathrm{ALT}}] = \mathrm{Var}\Big[\mathbb{E}[\widehat{T}_{ij}^{\mathrm{CLTM}} \mid \widehat{Y}]\Big] \geq 0,$$

which proves the claim.

B Additional Experimental Results

We report additional results for WRMF and T-CE, together with a comparison between BLTM and RGBT. BLTM (Yang et al., 2022) is a simplified special case of RGBT. Their comparison evaluates the complete training configurations. Each table ranks only the methods shown: bold and underlining mark the best and second-best recommendation scores; bold marks the lowest mean estimation error. Ties are retained.

Table B1: Additional recommendation results on six backbones. Higher is better.

Method	Adressa				MovieLens				Yelp			
	R@10	R@50	N@10	N@50	R@10	R@50	N@10	N@50	R@10	R@50	N@10	N@50
GMF												
WRMF	0.1893	0.2547	0.1156	0.1332	0.0689	0.1898	0.0527	0.0888	0.0195	0.0669	0.0138	0.0273
T-CE	**0.1917**	**0.2564**	**0.1171**	**0.1345**	0.0814	0.1900	0.0621	0.0960	0.0246	0.0775	0.0181	0.0331
BLTM	0.1871	0.2126	0.1119	0.1196	0.0800	0.2022	**0.0668**	0.1037	0.0261	0.0440	0.0193	0.0250
RGBT (Ours)	0.1866	0.2546	0.1138	0.1320	**0.0836**	**0.2124**	0.0664	**0.1069**	**0.0302**	**0.1068**	**0.0210**	**0.0427**
NeuMF												
WRMF	0.2830	0.3581	**0.1766**	**0.1970**	0.0713	0.1992	0.0590	0.0963	0.0170	0.0616	0.0119	0.0247
T-CE	0.2396	0.3570	0.1540	0.1856	0.0719	0.1925	0.0585	0.0941	0.0187	0.0613	0.0145	0.0267
BLTM	0.2830	0.3103	0.1738	0.1822	0.0466	0.1628	0.0375	0.0731	0.0230	0.0390	0.0163	0.0214
RGBT (Ours)	**0.2890**	**0.3686**	0.1756	0.1964	**0.0879**	**0.2151**	**0.0745**	**0.1130**	**0.0272**	**0.0987**	**0.0192**	**0.0394**
NGCF												
WRMF	0.1970	0.2550	0.1209	0.1366	0.0641	0.1919	0.0484	0.0867	0.0227	0.0724	0.0167	0.0307
T-CE	0.1974	0.2611	0.1208	0.1378	0.0628	0.1858	0.0524	0.0889	0.0293	0.0896	0.0213	0.0384
BLTM	**0.2800**	0.3110	**0.1726**	0.1820	0.0833	0.1952	0.0641	0.0987	0.0324	0.0546	**0.0234**	0.0304
RGBT (Ours)	0.2755	**0.3680**	0.1711	**0.1960**	**0.0847**	**0.2097**	**0.0733**	**0.1118**	**0.0328**	**0.1069**	0.0233	**0.0444**
LightGCN												
WRMF	0.1938	0.2514	0.1197	0.1355	0.0623	0.1934	0.0487	0.0880	0.0226	0.0724	0.0167	0.0307
T-CE	0.1970	0.2595	0.1205	0.1375	0.0629	0.1844	0.0512	0.0878	0.0292	0.0891	0.0211	0.0380
BLTM	0.1990	0.2236	0.1202	0.1275	0.0741	0.2011	0.0592	0.0964	0.0309	0.0504	0.0220	0.0281
RGBT (Ours)	**0.2562**	**0.4177**	**0.1602**	**0.2002**	**0.0880**	**0.2039**	**0.0767**	**0.1115**	**0.0411**	**0.1275**	**0.0301**	**0.0543**
XSimGCL												
WRMF	0.0349	0.0749	0.0212	0.0318	0.0229	0.0651	0.0144	0.0277	0.0585	0.1668	0.0436	0.0740
T-CE	0.1080	0.1859	0.0653	0.0852	0.0224	0.0631	0.0135	0.0264	0.0592	0.1672	**0.0439**	0.0742
BLTM	0.1081	0.1868	0.0655	0.0854	0.0222	0.0640	0.0138	0.0269	0.0586	0.1670	0.0437	0.0741
RGBT (Ours)	**0.2310**	**0.3118**	**0.1547**	**0.1766**	**0.0882**	**0.2146**	**0.0764**	**0.1150**	**0.0598**	**0.1761**	0.0436	**0.0767**
CDAE												
WRMF	0.2856	0.3589	0.1730	0.1925	0.0725	0.2006	0.0581	0.0967	0.0134	0.0475	0.0094	0.0195
T-CE	0.2350	0.3733	0.1422	0.1797	0.0775	0.1953	0.0621	0.0974	0.0163	0.0565	0.0117	0.0233
BLTM	0.2838	**0.3775**	0.1748	**0.1986**	0.0762	0.2017	0.0620	0.0997	0.0261	0.0813	0.0189	0.0347
RGBT (Ours)	**0.2886**	0.3726	**0.1761**	**0.1986**	**0.0902**	**0.2178**	**0.0751**	**0.1130**	**0.0421**	**0.1347**	**0.0305**	**0.0567**

Table B2: BLTM and RGBT: average matrix estimation error. Lower is better. Each summary covers four backbones and four noise rates; $\pm$ denotes the standard deviation across these 16 settings.

Method	Pairflip Noise Avg	Symmetric Noise Avg
BLTM	2.5335 ± 0.5120	1.4914 ± 0.5512
RGBT (Ours)	$\mathbf{2.2240 \pm 0.9949}$	$\mathbf{1.1050 \pm 0.6192}$

Table B3: BLTM and RGBT: matrix estimation error by noise rate. Each entry summarizes four backbones. Lower is better.

Method	0.1	0.2	0.3	0.4
		Pairflip Noise		
BLTM	2.0672 ± 0.6215	2.3766 ± 0.2798	2.5374 ± 0.0988	$\mathbf{3.1530 \pm 0.0438}$
RGBT (Ours)	$\mathbf{1.1662 \pm 0.1441}$	$\mathbf{1.5711 \pm 0.0523}$	$\mathbf{2.4984 \pm 0.0831}$	3.6605 ± 0.0831
		Symmetric Noise		
BLTM	2.1484 ± 0.0155	$\mathbf{0.7932 \pm 0.2412}$	1.2503 ± 0.1033	1.7737 ± 0.2013
RGBT (Ours)	$\mathbf{2.0184 \pm 0.2235}$	1.0498 ± 0.2790	$\mathbf{0.4963 \pm 0.1962}$	$\mathbf{0.8554 \pm 0.2390}$

Table B4: Matrix estimation error at varying noise levels. Each entry summarizes four backbones. Lower is better. Bold marks the lowest mean among the methods shown.

Method	0.1	0.2	0.3	0.4
		Pairflip Noise		
CCR	$\mathbf{0.9449 \pm 0.0133}$	1.9492 ± 0.0068	2.9463 ± 0.0142	3.9490 ± 0.0079
DUAL_T	2.0026 ± 1.4695	2.3690 ± 0.9008	2.7529 ± 0.3113	$\mathbf{3.2687 \pm 0.3649}$
T_REVISION	5.7242 ± 1.0773	5.1809 ± 1.0508	4.3996 ± 1.2132	4.7766 ± 0.8477
VOLMINNET	7.0047 ± 0.0111	6.0098 ± 0.0074	5.9977 ± 0.0087	5.9971 ± 0.0132
RRFN	1.0356 ± 0.1623	1.7048 ± 0.4040	2.6963 ± 0.4963	3.6098 ± 0.6610
CONL	3.5564 ± 0.6715	3.2382 ± 0.5243	3.2941 ± 0.1784	3.5336 ± 0.1781
RGBT (Ours)	1.1662 ± 0.1441	$\mathbf{1.5711 \pm 0.0523}$	$\mathbf{2.4984 \pm 0.0831}$	3.6605 ± 0.0831
		Symmetric Noise		
CCR	$\mathbf{0.7972 \pm 0.0250}$	1.7921 ± 0.0275	2.7542 ± 0.1001	3.7844 ± 0.0380
DUAL_T	2.1589 ± 1.6035	2.1689 ± 1.0143	2.2622 ± 0.3763	2.5225 ± 0.6148
T_REVISION	5.8187 ± 1.1243	5.0264 ± 0.9347	4.1310 ± 0.6828	3.5014 ± 0.7892
VOLMINNET	6.9933 ± 0.0046	6.0000 ± 0.0087	4.9986 ± 0.0174	4.0026 ± 0.0134
RRFN	0.8730 ± 0.1592	1.8707 ± 0.1492	2.6599 ± 0.4166	3.3823 ± 0.9661
CONL	3.0185 ± 0.9230	2.5017 ± 0.5508	1.9383 ± 0.5774	2.1299 ± 0.5762
RGBT (Ours)	2.0184 ± 0.2235	$\mathbf{1.0498 \pm 0.2790}$	$\mathbf{0.4963 \pm 0.1962}$	$\mathbf{0.8554 \pm 0.2390}$

RGBT has lower average matrix error than BLTM under both noise types. BLTM retains lower error at Pairflip rate 0.4 and Symmetric rate 0.2.