Enhancing Robustness and Generalization Capability for Multimodal Recommender Systems via Sharpness-Aware Minimization

Jinfeng Xu, Zheyu Chen, Jinze Li, Shuo Yang, Wei Wang, *Senior Member, IEEE*, Xiping Hu, *Senior Member, IEEE*, Raymond Chi-Wing Wong, and Edith C. H. Ngai, *Senior Member, IEEE*

***Abstract*—Multimodal recommender systems utilize a variety of information types to model user preferences and item properties, aiding in the discovery of items that align with user interests. Rich multimodal information alleviates inherent challenges in recommendation systems, such as data sparsity and cold start problems. However, multimodal information further introduces challenges in terms of robustness and generalization capability. Regarding robustness, multimodal information magnifies the risks associated with information adjustment and inherent noise, posing severe challenges to the stability of recommendation models. For generalization capability, multimodal recommender systems are more complex and difficult to train, making it harder for models to handle data beyond the training set, posing significant challenges to model generalization capability. In this paper, we analyze the shortcomings of existing robustness and generalization capability enhancement strategies in the multimodal recommendation field. We propose a sharpness-aware minimization strategy focused on batch data (BSAM), which effectively enhances the robustness and generalization capability of multimodal recommender systems without requiring extensive hyper-parameter tuning. Furthermore, we introduce a mixed loss variant strategy (BSAM+), which accelerates convergence and achieves remarkable performance improvement. We provide rigorous theoretical proofs and conduct experiments with nine advanced models on five widely used datasets to validate the superiority of our strategies. Moreover, our strategies can be integrated with existing robust training and data augmentation strategies to achieve further improvement, providing a superior training paradigm for multimodal recommendations.**

***Index Terms*—Robustness, multimodal recommendation.**

Received 9 January 2025; revised 12 June 2025; accepted 25 August 2025. Date of publication 9 September 2025; date of current version 9 October 2025. This work was supported in part by the Hong Kong UGC General Research Fund under Grant 17203320 and Grant 17209822 and in part by the project grants from the HKU-SCF FinTech Academy. Recommended for acceptance by Y. Deldjoo. *(Corresponding author: Edith C. H. Ngai.)*

Jinfeng Xu, Jinze Li, Shuo Yang, and Edith C. H. Ngai are with the Department of Electrical and Electronic Engineering, University of Hong Kong, Pok Fu Lam, Hong Kong (e-mail: jinfeng@connect.hku.hk; lijinze-hku@connect.hku.hk; shuo.yang@connect.hku.hk; chngai@eee.hku.hk).

Zheyu Chen is with the Department of Electrical and Electronic Engineering, Hong Kong Polytechnic University, Hung Hom, Hong Kong (e-mail: zheyu.chen@connect.polyu.hk).

Wei Wang and Xiping Hu are with the Department of Engineering, Shenzhen MSU-BIT University, Shenzhen 518115, China, and also with the School of Medical Technology, Beijing Institute of Technology, Beijing 100811, China (e-mail: ehomewang@ieee.org; huxp@bit.edu.cn).

Raymond Chi-Wing Wong is with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong (e-mail: raywong@ust.hk).

Digital Object Identifier 10.1109/TKDE.2025.3604242

I. Introduction

Recommender systems are crucial for helping users sift through the vast array of options available online and for pinpointing suitable items or services. The integration of deep learning techniques [1], [2], [3], [4] into recommendation systems has become widespread, leveraging historical user-item interactions to accurately model user preferences and facilitate personalized recommendations. Recently, the advent of rich multimodal content [5], [6], [7], [8], [9], which includes text, images, and videos, has led to the development of multimodal recommender systems [10], [11] that address significant challenges such as data sparsity and the cold start problems. However, the inclusion of multimodal data introduces additional robustness and generalization capability challenges in recommender systems. For robustness, the integration of multimodal data amplifies risks related to *inherent noise* [12], [13] and *information adjustment* [14], [15], which significantly compromises the stability of recommendation models (Defined in Section II-D and validated in Section II-D). For generalization capability, multimodal recommender systems face increased complexity and training difficulties, which inherently impair the models' capacity to effectively process and adapt to data beyond the training dataset, thereby posing substantial challenges to the generalization capability of these systems (Defined in Section II-E and validated in Section II-E).

In recent years, meticulously designed *gradient update strategy* [16] achieved an effective enhancement for the robustness of models within the multimodal recommendation field. However, this strategy entails unaffordable hyper-parameter search costs and does not significantly aid in improving model generalization capability. Furthermore, the *Sharpness-Aware Minimization strategy* (SAM) [17], [18], [19], which enhances model robustness and generalization by smoothing the local minima of the loss landscape, has shown considerable success in the representation learning field. Nevertheless, the inherent sparsity of recommendation data raises concerns about the effectiveness of SAM, with multimodal information further exacerbating these limitations. Additionally, manually defined interval batches necessitate high hyper-parameter search costs for the SAM strategy. We argue that this is due to the significant discrepancies between batch and global data caused by the sparsity of recommendation data, leading to global data misguiding the local minima of batch data, thereby failing to achieve satisfactory results.

In this paper, we empirically and theoretically validate the limitations of existing studies. To reduce the high costs of hyper-parameter tuning and address the challenges posed by data sparsity on existing sharpness-aware minimization strategies, we introduce a tailored *Batch-focused Sharpness-Aware Minimization strategy* (BSAM). This strategy simultaneously enhances the robustness and generalization capability of multimodal recommender systems without extensive hyper-parameter tuning. Moreover, we propose a mixed loss variant, called BSAM+, which accelerates convergence and significantly boosts performance. We provide rigorous proofs and conduct experiments across various models and datasets to validate the superiority of our strategies. Furthermore, our strategies can be integrated with existing robust training and data augmentation techniques to further improve results, thereby creating an exemplary training paradigm for multimodal recommendations. In summary, our contributions are threefold: 1) We conduct empirical and theoretical analyses to identify the limitations of existing robustness and generalization capability enhancement strategies in the multimodal recommendation field. 2) We introduce a tailored *Batch-focused Sharpness-Aware Minimization strategy* (BSAM) that improves the robustness and generalization of multimodal recommendations without intensive hyper-parameter tuning. Additionally, we develop BSAM+, a mixed loss variant that accelerates convergence and enhances performance. 3) We provide extensive experiments and rigorous proofs to verify the effectiveness and efficiency of BSAM and BSAM+ strategies.

II. Preliminary

We provide a wealth of preliminaries to help readers understand the background and motivation for our research.

A. Multimodal Recommendation Task

Considering a set of users $\mathcal{U} = \{u_1, u_2, \ldots, u_{|\mathcal{U}|}\}$, and a set of items $\mathcal{I} = \{i_1, i_2, \ldots, i_{|\mathcal{I}|}\}$, each $u \in \mathcal{U}$ is associated with an item set $\mathcal{I}^u \in \mathcal{I}$, where each item $i \in \mathcal{I}^u$ has an observed interaction with user u. Moreover, each item $i \in \mathcal{I}$ has multimodal information, including *visual feature* $v_i \in \mathcal{V}$ and *textual feature* $t_i \in \mathcal{T}$. The interaction matrix is defined as $\mathbf{R} \in \{0, 1\}^{|\mathcal{U}| \times |\mathcal{I}|}$, where $\mathbf{R}_{u,i} = 1$ is an observed interaction and $\mathbf{R}_{u,i} = 0$ is unobserved interaction. Formally, given a multimodal recommendation model denoted as $f(\cdot)$, we define the predicted score $r_{u,i}$ which signifies the preference of user u for iten i as follow:

$$r_{u,i} = f(u, i, v_i, t_i, \mathcal{I}^u|\Theta), \tag{1}$$

where $\Theta \in \mathbb{R}^d$ is the model parameters, and d is the model dimensionality.

B. Loss Function

The loss function for multimodal recommender systems can be divided into two parts: *major task* and *auxiliary task*. For the major task, following previous studies [9], [20], [21], we adopt *Bayesian personalized ranking* (BPR) [22] loss. The core objective of BPR is to enhance the divergence in the predictive preference between positive and negative items within each user-item triplet $(u, p, n) \in \mathcal{O}$, where $\mathcal{O}$ signifies the collection of training data, the term positive item p pertains to an item with which the user u has interacted, and the negative item n is selected randomly from the pool of items without interaction with user u. For the auxiliary task, following previous studies [23], [24], [25], we adopt *InfoNCE* [26] as the loss function. This auxiliary task can be decoupled to *alignment and uniformity loss*. *Alignment loss* minimizes the distance between positive pairs. Meanwhile, *uniformity loss* maximizes the distance between uncorrelated negative pairs. The major task and the auxiliary task are complementary because they have similar goals. Therefore, a more comprehensive model can be established by combining these two tasks. Given a dataset $S \triangleq \cup_{k=1}^{n}\{((u_k, i_k, v_{i_k}, t_{i_k}, \mathcal{I}^{u_k}), \mathbf{R}_{u_k,i_k})\}$ i.i.d drawn from a data distribution D, where n is the total number of samples in this dataset. We restate the entire training set loss function as $\mathcal{L}_S(\Theta)$. For simplification, we use variables x_k and y_k to denote $(u_k, i_k, v_{i_k}, t_{i_k}, \mathcal{I}^{u_k})$ and $\mathbf{R}_{u_k,i_k}$, respectively. Then, we utilize the training loss $\mathcal{L}_S(\Theta) \triangleq \frac{1}{n}\sum_{k=1}^{n} l(f(x_k|\Theta), y_k)$ to optimize the model parameters Θ of multimodal recommender systems, where $l(\cdot,\cdot)$ is a per-data-point loss function. Therefore, the conventional gradient descent is applied to $\mathcal{L}_S(\Theta)$ with the current learnable parameters Θ_t during training as follows:

$$\Theta_{t+1} = \Theta_t - \gamma \nabla_\Theta \mathcal{L}_S(\Theta_t), \tag{2}$$

where γ is the learning rate. For each training iteration, given the batch training loss $\mathcal{L}_B(\Theta_t)$ with a batch training data B.

C. Flat Local Minima

The connection between the geometry of the loss landscape and model performance has been studied extensively from both theoretical and empirical perspectives [27], [28]. It has been found that *flat local minima* lead to better generalization capabilities than *sharp minima* in the sense that a *flat minimizer* is more robust when the test loss is shifted due to random perturbations [29], [30], [31].

D. Robustness for Multimodal Recommendation

Integrating multimodal information mitigates data sparsity and cold-start issues in recommender systems, yet simultaneously introduces vulnerabilities to input distribution shifts. As defined by [16], these shifts manifest as two distinct robustness risks: 1) *Inherent Noise Risk:* Primarily occurring during training, this risk stems from intrinsic imperfections within the raw multimodal data. Examples include low-quality item images or irrelevant/erroneous feature information [12], [13]. 2) *Information Adjustment Risk:* Primarily occurring in deployment, this risk arises from frequent, intentional modifications to multimodal content. Examples include merchants updating product descriptions for promotions or replacing images to enhance appeal [14], [15]. Both risks perturb the input distribution, confounding the multimodal recommendation and leading to inaccurate recommendations. Achieving multimodal recommendation robustness is therefore defined by the model's capability to effectively mitigate the detrimental effects induced by these specific distribution shifts.

E. Generalization Capability

Many modern complex neural networks are prone to memorizing training data, thus leading to *overfitting* problems, especially with relatively small datasets. Therefore, it is important to ensure that the actual parameters chosen for models can perform well beyond the training set. It is worth noting that the *generalization gap* between the expected loss $\mathcal{L}_D(\Theta)$ and the training loss $\mathcal{L}_S(\Theta)$ represents the *generalization capability* of the model to generalize on unseen data. The expected loss $\mathcal{L}_D(\Theta)$ can be formally expressed as follows:

$$\mathcal{L}_S(\Theta) = \mathbb{E}_{(x_k,y_k)\sim D}\left[l\left(f(x_k|\Theta), y_k\right)\right]. \tag{3}$$

The empirical training loss $\mathcal{L}_S(\Theta)$ is defined as follow,

$$\mathcal{L}_D(\Theta) = \frac{1}{n}\sum_{k=1}^{n} l\left(f(x_k|\Theta), y_k\right). \tag{4}$$

The goal of multimodal recommendation is to recommend suitable items to users with whom they have never interacted before. Therefore, generalization capability is particularly important for multimodal recommendation models. However, due to the highly personalized nature of user preferences and the inherent data sparsity of recommendation systems, a significant performance gap often exists between the validation and test sets [32]. Furthermore, the performance of most existing models remains highly sensitive to random seeds [11]. Therefore, a meaningful and important way to evaluate the generalization capability of multimodal recommendations is to assess the stability of model performance under different random seeds (Validated in Section II-E).

III. Existing Flat Local Minima Methods

We theoretically and empirically analyze two existing methods, *Mirror Gradient* and *Sharpness-aware Minimization*.

A. Mirror Gradient (MG)

Previous work [16] proposes a two-phase *mirror gradient strategy* (MG) to enhance both the generalization capability and robustness of multimodal recommendation models by pursuing the *flat local minima* from the loss landscape perspective. For every β batches, where β is a hyper-parameter, it first utilizes conventional gradient by (2) to update the first $\beta - 1$ batches. Then, it employs a mirror training strategy to update Θ for the last batch B:

$$\begin{cases} \Theta_{t+1/2} = \Theta_t - \alpha_1\gamma\nabla_\Theta\mathcal{L}_B(\Theta_t), \\ \Theta_{t+1} = \Theta_{t+1/2} - \alpha_2\gamma\nabla_\Theta\mathcal{L}_B(\Theta_{t+1/2}), \end{cases} \tag{5}$$

where α_1 and α_2 are two positive scaling hyper-parameters.

Theorem 1: According to previous work [16], the training loss for the mirror training strategy in (5) is equal to a conventional training loss with an implicit regularization term:

$$\mathcal{L}_B^M(\Theta_{t+1}) = (\alpha_1 - \alpha_2)\mathcal{L}_B(\Theta_{t+1}) + \alpha_1\alpha_2\gamma\|\nabla_\Theta\mathcal{L}_B(\Theta_t)\|_2^2,$$

where the first term can be considered as a conventional gradient with scaling factor $\alpha_1 - \alpha_2$. The last term can be considered as an implicit regularization term.

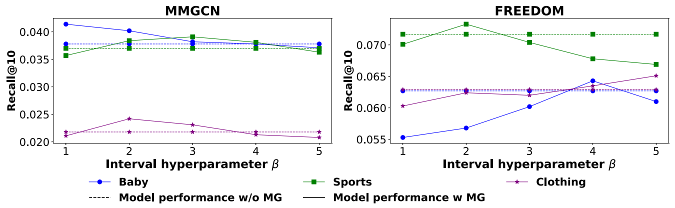


Fig. 1. Interval hyper-parameter β for MG. Optimal choices vary across models and datasets, and some sub-optimal parameter choices even lead to performance degradation, which brings inevitable hyper-parameter tuning costs.

However, there are four limitations for MG. *L1:* MG brings three extra hyper-parameters α_1, α_2, and β, which significantly increase the cost of finding the optimal settings. *L2:* Turning hyper-parameters α_1 and α_2 for implicit regularization term will inevitably affect the conventional gradient term. *L3:* Learning rate γ and the scaling factor $\alpha_1\alpha_2$ in the implicit regularization term are correlated, increasing the difficulty for hyper-parameters selection on different datasets. *L4:* Pre-defining interval β as a hyper-parameter is unreasonable and inefficient. We empirically investigate the impact of β in Fig. 1 to verify this limitation.

L3 and *L2* jointly aggravate *L1*. Specifically, during model parameter updating, the conventional gradient term is affected *linearly* by the learning rate γ, while the implicit regularization term is affected *quadratically* by the learning rate γ. However, both of these two terms are jointly affected by two additional hyper-parameters α_1 and α_2. This leads to the search range of the hyper-parameters of the model varying greatly at different learning rates. For example, we define $k = (\alpha_1 - \alpha_2)/(\alpha_1\alpha_2\gamma)$ to denote a well-designed influence rate between these two terms. If we hope to change the learning rate γ while keeping this influence rate k, numerous choices for α_1 and α_2 become available. Thus, it will extremely increase the cost of finding the optimal settings.

B. Sharpness-Aware Minimization (SAM)

Although MG contributed to exploring the generalization capability and robustness enhancement for multimodal recommendation models from a loss landscape perspective. However, it cannot be widely used due to the *unacceptable* model tuning parameter cost.

Sharpness-aware Minimization (SAM) [17], [18], [33] is a recently proposed training scheme that seeks flat minima by formulating a min-max problem and utilizing *adversarial weight perturbation* (AWP) to encourage parameters to sit in neighborhoods with uniformly low loss. Conventional optimization methods minimize $\mathcal{L}_S$ by stochastic gradient descent. SAM aims at additionally minimizing the worst-case sharpness of the training loss in a neighborhood defined by a ball around Θ, i.e., $\max_{\|\epsilon\|_p<\rho}\mathcal{L}_S(\Theta+\epsilon) - \mathcal{L}_S(\Theta)$, where $\rho \geq 0$ is a radius hyper-parameter and $p \in [1,\infty]$ (In practice, $p = 2$ is optimal settings in most cases). This leads to the overall SAM loss $\mathcal{L}_S^S$

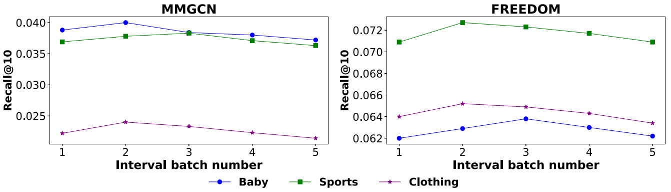


Fig. 2. Interval batch number study for SAM. Updating all batches with SAM will lead to suboptimal results.

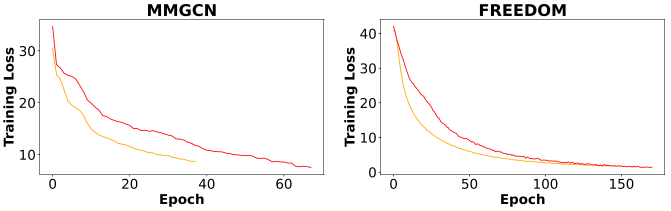


Fig. 3. Convergence speed for with or without SAM for MMGCN and FREEDOM models on the Baby dataset.

which is expressed as follows:

$$\mathcal{L}_S^S = \left[\max_{\|\epsilon\|_p \le \rho} \mathcal{L}_S(\Theta + \epsilon) - \mathcal{L}_S(\Theta)\right] + \mathcal{L}_S(\Theta) = \max_{\|\epsilon\|_p \le \rho} \mathcal{L}_S(\Theta + \epsilon). \tag{6}$$

To efficiently optimize $\mathcal{L}_S^S$, SAM first approximates (6) via first-order Taylor expansion and computes the *adversarial perturbation* ϵ_t in iteration t as follows:

$$\epsilon_t \approx \arg\max_{\|\epsilon\|_2 \le \rho} \epsilon^\top \nabla_\Theta \mathcal{L}_S(\Theta_t) = \rho \frac{\nabla_\Theta \mathcal{L}_S(\Theta_t)}{\|\nabla_\Theta \mathcal{L}_S(\Theta_t)\|_2}. \tag{7}$$

Subsequently, one can compute the gradient at the perturbed point $\Theta_t + \epsilon_t$. Then, for batch data B, it uses the updating step of the conventional SGD optimizer to update:

$$\Theta_{t+1} = \Theta_t - \gamma \nabla \mathcal{L}_B(\Theta_t + \epsilon_t). \tag{8}$$

Other base optimizers, e.g., Adam, Adagrad, and RMSprop, can also be used to update the model parameters in (8).

However, there are two limitations to SAM. *L1:* SAM updates are also not suitable to be applied to all batches in many cases (especially sparsity dataset), which is also observed by previous studies [34], [35]. As shown in Fig. 2, updating for all batches will lead to sub-optimal model performance in the multimodal recommendation field, which may be caused by the natural data sparsity problem in the recommendation field. *L2:* As shown in Fig. 3 illustrated, SAM leads to increased difficulty in the convergence of the model.

Inspired by previous work[1] [35], the reason behind *L1* may be that the updating for all batches increases the sharpness loss of the entire dataset, leading to inconsistencies with the subsequent sharpness minimization step which only uses the current batch data to minimize training loss value and its sharpness. Therefore, we propose a new variant of SAM called BSAM to remove the negative effect from the entire dataset loss and effectively reach flat local minima by minimizing training loss for each batch of data. Moreover, the reason behind *L2* may be that SAM can both minimize conventional major loss and minimize the sharpness of the loss landscape with a unified objective $\max_{\|\epsilon\|_p < \rho} \mathcal{L}_S(\Theta + \epsilon)$. Therefore, although the sharpness minimization loss can be adjusted via the perturbation range, while the balance between these two operations cannot be adjusted. To this end, we further propose an enhanced BSAM+ by adding a trade-off capability between these two operations. We detailed BSAM and BSAM+ in Section IV with theoretical analysis. Then, we provide extensive empirical evaluations in Section V.

IV. BSAM and BSAM+

In this section, we introduce our proposed BSAM framework and its variant, BSAM+, along with a detailed convergence analysis. BSAM is specifically designed to address the limitations of existing flat local minima methods, such as MG and SAM, in multimodal recommendation scenarios. While SAM successfully reduces the impractically large hyper-parameter search space of MG, BSAM not only inherits this advantage but also improves upon it by focusing on optimizing the sharpness of each batch during training, rather than prioritizing global sharpness as in SAM. This batch-focused approach effectively mitigates SAM's suboptimal performance on individual batches—a critical issue that is further exacerbated in recommendation systems due to severe data sparsity and the highly personalized nature of user preferences. Furthermore, we propose a variant, BSAM+, which incorporates a mixed loss that combines the SAM loss and the conventional loss. Additionally, by applying two separate losses to two different batches and combining them as a mixed loss, we effectively double the batch size, which further enhances the model's generalization ability.

A. BSAM

The sharpness loss of the entire dataset will lead to inconsistencies with each sharpness minimization step which only uses the current batch data to minimize the training loss value and its sharpness. To address this problem, we restrict adversarial perturbation ϵ to remove the negative influence for the sharpness loss of the entire dataset within each batch sharpness minimization. Formally, for batch B, after we approximate via first-order Taylor expansion, the restricted adversarial perturbation $\hat{\epsilon}_t$ in iteration t is as follows:

$$\hat{\epsilon}_t = \rho \frac{\nabla_\Theta \mathcal{L}_B(\Theta_t) - \mathrm{d}(\nabla_\Theta \mathcal{L}_B(\Theta_t), \nabla_\Theta \mathcal{L}_S(\Theta_t)) \nabla_\Theta \mathcal{L}_S(\Theta_t)}{\|\nabla_\Theta \mathcal{L}_B(\Theta_t) - \mathrm{d}(\nabla_\Theta \mathcal{L}_B(\Theta_t), \nabla_\Theta \mathcal{L}_S(\Theta_t)) \nabla_\Theta \mathcal{L}_S(\Theta_t)\|_2}, \tag{9}$$

where $\mathrm{d}(\cdot)$ denotes the distance function, and this paper adopts Manhattan Distance due to its universality (Any suitable alternative can replace it). For the batch gradient $\nabla_\Theta \mathcal{L}_B(\Theta_t)$,

[1] Note that FSAM [35] is effective in the computer vision field, but performs poorly in the recommendation scenario. This is because different batches in the recommendation scenario may contain completely different users. Recommendations pay close attention to the personalization of users, which is different from the computer vision field.

it is also computed in conventional SAM and thus does not bring extra computation overhead. However, BSAM needs to compute the full gradient $\nabla_\Theta \mathcal{L}_S(\Theta_t)$, which is computed on the whole dataset, and thus is computationally prohibitive in practice. To address this problem, we follow previous work [35] to estimate $\nabla_\Theta \mathcal{L}_S(\Theta_t)$ using the *Exponential Moving Average* (EMA), which accurately calculates the historical small batch gradient:

$$EMA_t = (1-\lambda)\nabla\mathcal{L}_B(\Theta_t) + \lambda EMA_{t-1}. \tag{10}$$

We state that EMA_t is an excellent estimation for $\nabla_\Theta \mathcal{L}_S(\Theta_t)$ as shown in Theorem 2 proved in the Appendix, available online.

Assumptions: We first state some standard assumptions in stochastic optimization [34], [36], [37] that will be used in our theoretical analysis:

Assumption 1. (β-Smoothness): Assume the loss function $\mathcal{L}_S$ $\mathbb{R}^d \mapsto \mathbb{R}$ to be β-*smooth*. There exists $\beta > 0$ such that:

$$\|\nabla\mathcal{L}_S(\Theta_a) - \nabla\mathcal{L}_S(\Theta_b)\|_2 \le \beta\|\Theta_a - \Theta_b\|_2, \forall\Theta_a, \Theta_b \in \mathbb{R}^d.$$

Assumption 2. (Bounded variance): There exists a constant $M > 0$ for any data batch B such that:

$$\mathbb{E}[\|\nabla\mathcal{L}_B(\Theta) - \nabla\mathcal{L}_S(\Theta)\|_2^2] \le M, \quad \forall\Theta \in \mathbb{R}^d.$$

Assumption 3. (Bounded gradient): There exists $G > 0$ for any data batch B such that:

$$\mathbb{E}[\|\nabla\mathcal{L}_B(\Theta)\|_2] \le G, \quad \forall\Theta \in \mathbb{R}^d.$$

Theorem 2: Based on Assumptions 1–3, assume that SAM uses SGD as an optimizer with a learning rate γ to update the model parameter. Setting $\lambda = 1 - C\gamma^{2/3}$, after $T > C'\gamma^{-2/3}$ training iterations, with probability $1-\delta$, we have:

$$\Phi_T = \|EMA_T - \nabla\mathcal{L}_S(\Theta_T)\|_2 \le \mathcal{O}\left(\gamma^{\frac{1}{3}}\beta^{\frac{1}{3}}G^{\frac{1}{3}}M^{\frac{1}{3}}\log\left(\frac{1}{\delta}\right)\right),$$

where C and C' are two universal constants.

We estimate that the error bound Φ_t between $\nabla\mathcal{L}_S(\Theta_t)$ and its EMA_T is at the order of $\mathcal{O}(\gamma^{1/3})$. On the non-convex problem, learning rate γ is often set as $\mathcal{O}(1/\sqrt{T})$ to ensure convergence. We provide a detailed convergence analysis in Section IV-C. Therefore, $\Phi_t = \mathcal{O}(\gamma^{1/6})$ is so small since the training iteration T is often large. Thus, EMA_t is an excellent estimation for $\nabla_\Theta\mathcal{L}_S(\Theta_t)$.

Therefore, for batch B, the restricted adversarial perturbation $\hat{\epsilon}_t$ in (9) is approximated as:

$$\hat{\epsilon}_t \approx \rho\frac{\nabla_\Theta\mathcal{L}_B(\Theta_t) - \mathrm{d}(\nabla_\Theta\mathcal{L}_B(\Theta_t), EMA_t)\cdot EMA_t}{\|\nabla_\Theta\mathcal{L}_B(\Theta_t) - \mathrm{d}(\nabla_\Theta\mathcal{L}_B(\Theta_t), EMA_t)\cdot EMA_t\|_2}. \tag{11}$$

Actually, BSAM indeed computes the adversarial perturbation by maximizing the current batch loss while minimizing the loss on the entire dataset, we can write the loss function $\mathcal{L}_B^B$ for batch B:

$$\mathcal{L}_B^B = \mathcal{L}_B(\Theta+\hat{\epsilon}) \text{ s.t. } \hat{\epsilon} = \arg\max_{\|\epsilon\|_2\le\rho} \mathcal{L}_B(\Theta+\epsilon) - \mathcal{T}\cdot\mathcal{L}_S(\Theta+\epsilon), \tag{12}$$

where $\mathcal{T} = \mathrm{d}(\mathcal{L}_B(\Theta+\epsilon), \mathcal{L}_S(\Theta+\epsilon))$. We further analyze the convergence of BSAM under a non-convex setting in Section IV-C to verify that BSAM shares the same convergence speed as SAM. Moreover, we highlight the difference between our BSAM and FSAM: 1) FSAM [35] aims to completely eliminate the full gradient component and retain only the batch-related component. However, due to the personalized differences between different users in different batches in the recommendation scenario, it is difficult to achieve satisfactory results. Our BSAM restricts the adversarial perturbation ϵ while focusing on the batch to eliminate the negative impact on the sharpness loss of the entire dataset in each batch sharpness minimization. 2) FSAM uses a constant to simplify the approximate full gradient component for each batch, making it impossible to achieve satisfactory results in the recommendation scenario due to the personalized differences between different users in different batches. Note that Section V empirically verifies the superiority of BSAM in recommendation scenarios.

B. BSAM+ (Mixed Loss Strategy)

From our Theorem 3 in Section IV-C, we know that a large perturbation radius ρ might reach a flatter local minima, but leads to increased difficulty in the convergence process. To achieve good generalization ability brought by a large perturbation radius while enjoying a good convergence property, we propose a BSAM+, which mixes the conventional loss with BSAM loss, inspired by [38].[2] To enhance the optimization process, we utilize two distinct batches of data, namely B_1 and B_2, for the two gradient steps involved. It is worth noting that this effectively doubles the virtual batch size. Formally, the optimization process for BSAM+ is expressed as:

$$\Theta_{t+1} = \Theta_t - \gamma\left[\kappa\nabla\mathcal{L}_{B_1}(\Theta_t+\hat{\epsilon}) + (1-\kappa)\nabla\mathcal{L}_{B_2}(\Theta_t)\right], \tag{13}$$

where $\kappa \in [0, 1]$ is a pre-defined balance hyper-parameter. Intuitively, the two loss terms in our BSAM+ objective are complementary to each other. BSAM loss provides a smoothed landscape to find a flat local minima, while the conventional loss helps recover the necessary local information and better locates the minima that contribute to high performance. It is worth noting that this mixed loss function can also be plugged into other variants of SAM. We further provide a theoretical evaluation from the convergence analysis perspective in Section IV-C and the empirical perspective in Section V.

C. Analysis

We analyze the convergence properties of SAM, BSAM, SAM+, and BSAM+ under the non-convex setting.

Theorem 3: Based on Assumption 1–2 and assume that SAM and BSAM use SGD as an optimizer with a learning rate γ to update the model parameter. Fixing learning rate $\gamma = \frac{\gamma_0}{\sqrt{T}} \le \frac{1}{\beta}$

[2] GNP [38] only utilizes a single batch to adjust weights for main loss and SAM loss. We innovatively mixed loss strategy to combine two different batches. Compared with GNP, our strategy can further make up for the full gradient information that BSAM may lose and speed up the convergence of the model except to adjust weights for main loss and SAM loss.

and the perturbation radius $\rho = \frac{\rho_0}{t}$, we have:

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla\mathcal{L}_S(\Theta_t)\|^2 \le \frac{2\Delta}{\gamma_0\sqrt{T}} + \frac{(2M + \rho_0^2\beta^2)\beta\gamma_0}{\sqrt{T}} + \frac{\rho_0^2\beta^2\log T}{\sqrt{T}},$$

where $\Delta = \mathbb{E}[\mathcal{L}_S(\Theta_0) - \mathcal{L}_S(\Theta^*)]$ with an optimal solution Θ^*.

Theorem 4: Based on Assumption 1–2 and assume that SAM+ and BSAM+ use SGD as an optimizer with a learning rate γ to update the model parameter. Fixing learning rate $\$\gamma = \frac{\gamma_0}{\sqrt{T}} \le \frac{1}{\beta}$ and the perturbation radius $\rho = \frac{\rho_0}{t}$, we have:

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla\mathcal{L}_S(\Theta_t)\|^2 \le \frac{2\Delta}{\gamma_0\sqrt{T}} + \frac{(2M(2\kappa^2 - 2\kappa + 1) + \kappa^2\rho_0^2\beta^2)\beta\gamma_0}{\sqrt{T}} + \frac{\kappa^2\rho_0^2\beta^2\log T}{\sqrt{T}},$$

where $\Delta = \mathbb{E}[\mathcal{L}_S(\Theta_0) - \mathcal{L}_S(\Theta^*)]$ with an optimal solution Θ^*.

Theorem 3 is widely proved by previous studies [34], [35]. Therefore, we provide a proof of Theorem 4 in the Appendix, available online:

Based on Theorem 3–4, we have the following observations:

- Theorem 3 shows the convergence rate of BSAM is $\mathcal{O}(\log T/\sqrt{T})$, which shares the same convergence speed as SAM. But, BSAM achieves better generalization capability and robustness for multimodal recommender systems than SAM as shown in Section V.
- Compared to the convergence properties of SAM and BSAM as stated in Theorem 3, SAM+ and BSAM+ as stated in Theorem 4 offer immediate improvements in two aspects. 1) It can adjust the convergence rate by a factor κ^2. 2) SAM+ and BSAM+ enable using two different data batches to compute the two gradient steps. This effectively doubles the batch size for each training iteration.

Note that in SAM+ and BSAM+, the two batches' losses are separable and can be calculated independently and in parallel. This parallel computing capability halves the training time, making it a highly efficient approach for large-scale problems. We summarize the algorithmic steps of BSAM and BSAM+ with SGD as the base optimizer in Algorithms 1.

Algorithm 1: BSAM+ Algorithm.

```
Input: Training dataset $S \triangleq \cup_{k=1}^{n}\{(x_k, y_k))\}$,
       perturbation radius $\rho$, learning rate $\gamma$,
       momentum factor $\lambda$, similarity constant $c$,
       balance factor $\kappa$
Output: Trained weight $\Theta$
Initialize $\Theta$, $t \leftarrow 0$, $EMA_{t-1} = 0$;
while not converged do
    if BSAM then
        // BSAM
        Sample a batch data $B$ from $S$;
        $EMA_t = (1-\lambda)\nabla\mathcal{L}_B(\Theta_t) + \lambda EMA_{t-1}$;
        Compute adversarial perturbation:
          $\hat{\epsilon}_t \approx \rho\frac{\nabla_\Theta\mathcal{L}_B(\Theta_t) - \mathrm{d}(\mathcal{L}_{B_1}(\Theta_t), EMA_t)\cdot EMA_t}{\|\nabla_\Theta\mathcal{L}_B(\Theta_t) - \mathrm{d}(\mathcal{L}_{B_1}(\Theta_t), EMA_t)\cdot EMA_t\|_2}$;
        Update $\Theta$ using gradient descent:
        $\Theta_{t+1} \leftarrow \Theta_t - \gamma\nabla\mathcal{L}_B(\Theta_t + \hat{\epsilon}_t)$;
    end
    else
        // BSAM+
        Sample two batches data $B_1$ and $B_2$ from $S$;
        $EMA_t = (1-\lambda)\nabla\mathcal{L}_{B_1}(\Theta_t) + \lambda EMA_{t-1}$;
        Compute adversarial perturbation:
          $\hat{\epsilon}_t \approx \rho\frac{\nabla_\Theta\mathcal{L}_{B_1}(\Theta_t) - \mathrm{d}(\mathcal{L}_{B_1}(\Theta_t), EMA_t)\cdot EMA_t}{\|\nabla_\Theta\mathcal{L}_{B_1}(\Theta_t) - \mathrm{d}(\mathcal{L}_{B_1}(\Theta_t), EMA_t)\cdot EMA_t\|_2}$;
        Update $\Theta$ using gradient descent:
        $\Theta_{t+1} =$
          $\Theta_t - \gamma[\kappa\nabla\mathcal{L}_{B_1}(\Theta_t + \hat{\epsilon}) + (1-\kappa)\nabla\mathcal{L}_{B_2}(\Theta_t)]$;
    end
    $t \leftarrow t + 1$;
end
return $\Theta_t$
```

V. Experiment

We conduct extensive experiments on some widely used real-world datasets. Experiment results can answer the following questions:

RQ1: Can BSAM and BSAM+ enhance the performance of multimodal recommender systems?

RQ2: Can BSAM and BSAM+ enhance the performance of non-multimodal recommender systems?

RQ3: Can BSAM and BSAM+ mitigate inherent noise and information adjustment risks?

RQ4: Can BSAM and BSAM+ improve to generalization capability of multimodal recommender systems?

RQ5: Are BSAM and BSAM+ superior to flat local minima methods?

RQ6: Can BSAM and BSAM+ be compatible with various optimizers?

RQ7: Can BSAM and BSAM+ be compatible with robust training and data augmentation strategies?

RQ8: Can BSAM and BSAM+ be compatible with various batch sizes?

RQ9: Can our mixed loss apply to other SAM methods?

RQ10: Can BSAM and BSAM+ affect the convergence properties?

RQ11: Does BSAM and BSAM+ enable multimodal models to approach flatter local minima?

RQ12: How sensitive are BSAM and BSAM+ under the perturbation of hyper-parameters?

A. Experimental Settings

1) Datasets: The experiments are conducted on five real-world datasets: Baby, Sports, Clothing, Pet, and Office from Amazon [39]. All the datasets comprise textual and visual features in the form of item descriptions and images. Our data preprocessing methodology follows the approach outlined in MMRec [40]. Table I shows the statistics of these datasets.

2) Evaluation Protocols: To evaluate the performance fairly, we adopt two widely used metrics: Recall@K (R@K) and

TABLE I
STATISTICS OF THE FIVE EVALUATION DATASETS

Datasets	#Users	#Items	#Interactions	Sparsity
Baby	19,445	7,050	160,792	99.88%
Sports	35,598	18,357	296,337	99.95%
Clothing	39,387	23,033	278,677	99.97%
Pet	19,856	8,510	157,836	99.91%
Office	4,905	2,420	53,258	99.55%

NDCG@K (N@K). We report the average metrics of all users in the test dataset under both K = 5 and K = 10. We follow the popular evaluation setting [9], [25], [41] with a random data splitting 8:1:1 for training, validation, and testing.

3) Baselines: We extensively examine the performance of our BSAM and BSAM+ across a variety of multimodal recommendation models, including MMGCN [6], DualGNN [42], SLMRec [23], FREEDOM [9], DRAGON [43], and LGMRec [41]. Besides, we further evaluate the performance of our BSAM and BSAM+ across non-multimodal models, including NGCF [44], LightGCN [20], and LayerGCN [45]. Additionally, we compare our BSAM and BSAM+ with MG [16], SAM [18], and ASAM [17] to verify the effectiveness of our approach. Moreover, we analyze our mixed loss strategy by designing mixed loss variants SAM+ and ASAM + for SAM and ASAM, respectively. Finally, we test the compatibility of our BSAM+ with the adversarial training strategy (AMR [12]) and LLM-based data augmentation strategy (GPT-4v [46]).

4) Implementation Details: We retain the standard settings for all baselines and fix batch size B by 2048. For our BSAM and BSAM+, we apply a grid search on hyper-parameters λ and κ in {0.2, 0.4, 0.6, 0.8}, and the perturbation radius ρ in {0.05, 0.10, 0.15, 0.20}. The common optimizer is Adam [47] and all training and evaluation of all models are conducted on RTX3090 GPU. For the GPT-4v data augmentation strategy, we utilize GPT-4V(ision) [46] to augment the raw text description via the items' image for Baby and Office datasets to improve the correlation between textual and visual modalities. *'gpt-4-vision-preview'* serves as the chosen LLM model, we design the following prompt: *'[V] This is the description of an item and the corresponding picture, please combine the picture to improve the quality of the description in one paragraph. The description is as follows: [T].'*, where $[V]$ is the raw image for each item and $[T]$ is the raw text description for each item.

B. Overall Performance (RQ1 - RQ2)

We evaluate the effectiveness of our BSAM and BSAM+ on various models for both multimodal and non-multimodal recommendation scenarios. From Table II, we find the following observations:

Observation1: BSAM and BSAM+ effectively enhance the performance of various multimodal recommendation models. As Table II shows, we conduct extensive experiments of BSAM and BSAM+ across six multimodal recommendation models using five distinct public datasets. Experiment results show that BSAM and BSAM+ achieve impressive improvements on all baselines across all evaluation metrics. BSAM+ consistently achieves superior results to BSAM. To summarize, the experimental results validate our motivation to rely on the general consensus in representation learning that user preferences within the range of flat local minima can greatly avoid the inherent noise risk in multimodal information.

Observation2: BSAM and BSAM+ effectively enhance the performance of various non-multimodal recommendation models. As Table II shows, we evaluate extensive experiments of BSAM and BSAM+ across three non-multimodal recommendation models using five distinct public datasets. Experiment results show that BSAM and BSAM+ still achieve notable improvements on all baselines across all evaluation metrics. This phenomenon from BSAM and BSAM+ can effectively lead to a flat local minima loss landscape to improve model robustness in the non-multimodal scenario.

C. Robustness (RQ3)

We further evaluate the robustness of BSAM and BSAM+ by explicitly simulating the two risks (inherent noise risk and information adjustment risk) mentioned in Section II-D in the inference phase.

Risk1: Inherent noise. Following previous work [16], [48] settings, we inject Gaussian noise ϵ into learnable item embedding layers for each model. This step aims to simulate the presence of inherent noise in multimodal recommendations.

Risk2: Information adjustment. We build an adjustment version for each dataset to simulate information adjustment in multimodal recommender systems. Specifically, we first replace the texts of the original dataset with captions generated by images using GPT-4v. Then, we use GPT-4v to summarize captions and add them into images to replace the images of the original dataset. Both these two processes occur randomly with a probability of one percent.

We conduct extensive experiments of BSAM and BSAM+ across three multimodal recommendation models using two distinct public datasets for varying noise and adjustment levels. For inherent noise, we build the following levels: L_1: injected Gaussian noise $\epsilon \sim \mathcal{N}(0, 10^{-6})$, L_2: injected Gaussian noise $\epsilon \sim \mathcal{N}(0, 10^{-5})$, L_3: injected Gaussian noise $\epsilon \sim \mathcal{N}(0, 10^{-4})$. For information adjustment, we build the following levels: L_1 randomly adjustment occurs with a probability of 1%, L_2 randomly adjustment occurs with a probability of 3%, and L_3 randomly adjustment occurs with a probability of 5%. The results (reporting the average of results over 5 runs) in Table III show that both BSAM and BSAM+ effectively mitigate the performance degradation caused by inherent noise and information adjustment, thereby enhancing the robustness of multimodal recommendation models.

Furthermore, we conduct in-depth robustness experiments under controlled noise injection settings to validate the resilience of BSAM and BSAM+. Specifically, we adhere to the three levels of inherent noise injection and directly inject noise into the raw modality features. The results (reporting the average of results over 5 runs) in Table IV demonstrate that BSAM and

TABLE II
PERFORMANCE COMPARISON OF BASELINES WITH OR WITHOUT BSAM AND BSAM+ ON ALL DATASETS IN TERMS OF RECALL@K (R@K) AND NDCG@K (N@K)

Datasets	Baby				Sports				Clothing				Pet				Office			
Metrics	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10
non-multimodal recommendation models																				
NGCF	0.0253	0.0427	0.0166	0.0223	0.0290	0.0467	0.0188	0.0246	0.0166	0.0271	0.0108	0.0142	0.0472	0.0729	0.0306	0.0391	0.0517	0.0763	0.0339	0.0424
+BSAM	0.0271	0.0464	0.0180	0.0242	0.0308	0.0499	0.0196	0.0262	0.0180	0.0292	0.0116	0.0152	0.0509	0.0778	0.0329	0.0411	0.0562	0.0823	0.0368	0.0457
+BSAM+	**0.0277***	**0.0466***	**0.0182***	**0.0243***	**0.0316***	**0.0505***	**0.0204***	**0.0266***	**0.0181***	**0.0294***	**0.0117***	**0.0154***	**0.0513***	**0.0785***	**0.0333***	**0.0419***	**0.0567***	**0.0831***	**0.0371***	**0.0462***
Improv.	9.49%	9.13%	9.64%	8.97%	8.97%	8.14%	8.51%	8.13%	9.04%	8.49%	8.33%	8.45%	8.69%	7.68%	8.82%	7.16%	9.67%	8.91%	9.44%	8.96%
LightGCN	0.0298	0.0479	0.0199	0.0257	0.0370	0.0569	0.0248	0.0311	0.0222	0.0340	0.0150	0.0188	0.0591	0.0891	0.0402	0.0500	0.0489	0.0791	0.0355	0.0459
+BSAM	0.0324	0.0515	0.0216	0.0274	0.0394	0.0603	0.0264	0.0327	0.0239	0.0362	0.0162	0.0200	0.0621	0.0936	0.0427	0.0529	0.0513	0.0820	0.0374	0.0486
+BSAM+	**0.0327***	**0.0524***	**0.0218***	**0.0280***	**0.0400***	**0.0610***	**0.0268***	**0.0333***	**0.0243***	**0.0370***	**0.0166***	**0.0204***	**0.0633***	**0.0948***	**0.0432***	**0.0533***	**0.0524***	**0.0843***	**0.0381***	**0.0490***
Improv.	9.73%	9.39%	9.55%	8.95%	8.11%	7.21%	8.06%	7.07%	9.46%	8.82%	10.67%	8.51%	7.11%	6.40%	7.46%	6.60%	7.16%	6.57%	7.32%	6.75%
LayerGCN	0.0319	0.0529	0.0213	0.0281	0.0379	0.0594	0.0254	0.0323	0.0235	0.0371	0.0156	0.0200	0.0583	0.0894	0.0393	0.0494	0.0538	0.0825	0.0387	0.0486
+BSAM	0.0340	0.0551	0.0227	0.0298	0.0413	0.0634	0.0274	0.0343	0.0251	0.0399	0.0167	0.0212	0.0626	0.0941	0.0428	0.0531	0.0573	0.0860	0.0407	0.0504
+BSAM+	**0.0345***	**0.0568***	**0.0232***	**0.0303***	**0.0416***	**0.0649***	**0.0277***	**0.0349***	**0.0258***	**0.0404***	**0.0171***	**0.0218***	**0.0633***	**0.0951***	**0.0430***	**0.0535***	**0.0588***	**0.0878***	**0.0419***	**0.0519***
Improv.	8.15%	7.37%	8.92%	7.83%	9.76%	9.26%	9.06%	8.05%	9.79%	8.89%	9.62%	9.00%	8.58%	6.38%	9.41%	8.30%	9.29%	6.42%	8.27%	6.79%
multimodal recommendation models																				
MMGCN	0.0240	0.0378	0.0160	0.0200	0.0216	0.0370	0.0143	0.0193	0.0130	0.0218	0.0088	0.0110	0.0378	0.0619	0.0251	0.0329	0.0318	0.0560	0.0223	0.0305
+BSAM	0.0273	0.0420	0.0179	0.0221	0.0238	0.0404	0.0155	0.0211	0.0147	0.0242	0.0097	0.0124	0.0412	0.0680	0.0274	0.0363	0.0365	0.0639	0.0246	0.0337
+BSAM+	**0.0277***	**0.0434***	**0.0181***	**0.0229***	**0.0243***	**0.0409***	**0.0160***	**0.0214***	**0.0151***	**0.0248***	**0.0102***	**0.0128***	**0.0422***	**0.0693***	**0.0282***	**0.0370***	**0.0371***	**0.0651***	**0.0253***	**0.0343***
Improv.	15.42%	14.15%	13.13%	14.50%	12.50%	10.54%	11.89%	10.88%	16.15%	13.76%	15.91%	16.36%	11.64%	11.95%	12.35%	12.46%	16.67%	16.25%	13.45%	12.46%
DualGNN	0.0322	0.0448	0.0216	0.0240	0.0374	0.0568	0.0253	0.0310	0.0277	0.0454	0.0185	0.0241	0.0578	0.0902	0.0396	0.0503	0.0550	0.0873	0.0369	0.0477
+BSAM	0.0353	0.0497	0.0228	0.0258	0.0406	0.0616	0.0274	0.0337	0.0302	0.0499	0.0200	0.0260	0.0617	0.0970	0.0427	0.0540	0.0581	0.0923	0.0389	0.0509
+BSAM+	**0.0359***	**0.0505***	**0.0232***	**0.0261***	**0.0413***	**0.0623***	**0.0281***	**0.0342***	**0.0310***	**0.0508***	**0.0207***	**0.0268***	**0.0626***	**0.0979***	**0.0433***	**0.0547***	**0.0588***	**0.0926***	**0.0394***	**0.0514***
Improv.	11.49%	12.72%	7.41%	8.75%	10.43%	9.68%	11.07%	10.32%	11.91%	12.11%	11.89%	11.20%	8.30%	8.54%	9.34%	8.75%	6.91%	6.07%	6.78%	7.76%
SLMRec	0.0343	0.0529	0.0226	0.0290	0.0429	0.0663	0.0288	0.0365	0.0292	0.0452	0.0196	0.0247	0.0618	0.0976	0.0416	0.0533	0.0461	0.0761	0.0314	0.0414
+BSAM	0.0377	0.0575	0.0247	0.0318	0.0453	0.0704	0.0305	0.0384	0.0317	0.0489	0.0211	0.0266	0.0667	0.1033	0.0444	0.0560	0.0501	0.0822	0.0353	0.0452
+BSAM+	**0.0385***	**0.0588***	**0.0255***	**0.0324***	**0.0461***	**0.0710***	**0.0310***	**0.0391***	**0.0322***	**0.0495***	**0.0217***	**0.0271***	**0.0674***	**0.1048***	**0.0450***	**0.0571***	**0.0514***	**0.0837***	**0.0351***	**0.0457***
Improv.	12.24%	11.15%	12.83%	11.72%	7.46%	7.09%	7.64%	7.12%	10.27%	9.51%	10.71%	9.72%	9.06%	7.38%	8.17%	7.13%	11.50%	9.99%	12.42%	10.39%
FREEDOM	0.0374	0.0627	0.0243	0.0330	0.0446	0.0717	0.0291	0.0385	0.0388	0.0629	0.0257	0.0341	0.0705	0.1086	0.0472	0.0595	0.0563	0.0952	0.0389	0.0518
+BSAM	0.0397	0.0650	0.0259	0.0347	0.0468	0.0747	0.0306	0.0399	0.0403	0.0655	0.0271	0.0353	0.0740	0.1127	0.0493	0.0614	0.0601	0.0994	0.0407	0.0537
+BSAM+	**0.0402***	**0.0661***	**0.0264***	**0.0354***	**0.0476***	**0.0751***	**0.0312***	**0.0404***	**0.0409***	**0.0660***	**0.0273***	**0.0359***	**0.0744***	**0.1133***	**0.0500***	**0.0621***	**0.0606***	**0.1004***	**0.0414***	**0.0544***
Improv.	7.49%	5.42%	8.64%	7.27%	6.73%	4.74%	7.22%	4.94%	5.41%	4.92%	6.23%	5.28%	5.53%	4.33%	5.93%	4.37%	7.64%	5.46%	6.43%	5.02%
DRAGON	0.0380	0.0662	0.0249	0.0345	0.0449	0.0752	0.0296	0.0413	0.0401	0.0671	0.0270	0.0365	0.0747	0.1151	0.0500	0.0630	0.0629	0.1014	0.0414	0.0557
+BSAM	0.0394	0.0685	0.0257	0.0358	0.0466	0.0775	0.0306	0.0426	0.0431	0.0708	0.0287	0.0382	0.0779	0.1193	0.0523	0.0656	0.0658	0.1050	0.0432	0.0580
+BSAM+	**0.0399***	**0.0691***	**0.0264***	**0.0362***	**0.0472***	**0.0783***	**0.0310***	**0.0432***	**0.0435***	**0.0717***	**0.0291***	**0.0387***	**0.0788***	**0.1203***	**0.0528***	**0.0661***	**0.0665***	**0.1064***	**0.0438***	**0.0588***
Improv.	5.00%	4.38%	6.02%	4.93%	4.68%	4.12%	4.73%	4.60%	8.48%	6.86%	7.78%	6.03%	5.49%	4.52%	5.60%	4.92%	5.72%	4.93%	5.80%	5.57%
LGMRec	0.0381	0.0647	0.0256	0.0333	0.0451	0.0719	0.0298	0.0387	0.0359	0.0555	0.0239	0.0302	0.0702	0.1057	0.0469	0.0584	0.0600	0.0959	0.0393	0.0514
+BSAM	0.0401	0.0676	0.0268	0.0348	0.0471	0.0747	0.0310	0.0400	0.0384	0.0589	0.0255	0.0321	0.0738	0.1107	0.0489	0.0609	0.0629	0.1001	0.0412	0.0539
+BSAM+	**0.0408***	**0.0684***	**0.0273***	**0.0352***	**0.0478***	**0.0756***	**0.0317***	**0.0406***	**0.0393***	**0.0600***	**0.0261***	**0.0327***	**0.0745***	**0.1121***	**0.0496***	**0.0615***	**0.0637***	**0.1015***	**0.0418***	**0.0545***
Improv.	7.09%	5.72%	6.64%	5.71%	5.99%	5.15%	6.38%	4.91%	9.47%	8.11%	9.21%	8.28%	6.13%	6.05%	5.76%	5.31%	6.17%	5.84%	6.36%	6.03%

* indicates the improvement is statistically significant where the p-value is less than 0.01.

TABLE III
THE TERMS "N" AND "A" REFER TO INHERENT NOISE AND INFORMATION ADJUSTMENT RISKS, RESPECTIVELY

Dataset	Model	Original	N L_1	Decr.	N L_2	Decr.	N L_3	Decr.	Original	A L_1	Decr.	A L_2	Decr.	A L_3	Decr.
Baby	MMGCN	0.0378	0.0310	17.99%	0.0301	20.37%	0.0268	29.10%	0.0378	0.0349	7.67%	0.0341	9.79%	0.0329	13.00%
	+BSAM	0.0420	0.0383	8.81%	0.0378	10.00%	0.0366	12.86%	0.0420	0.0407	3.10%	0.0402	4.29%	0.0395	5.95%
	+BSAM+	0.0434	0.0414	**4.61%**	0.0410	**5.53%**	0.0404	**6.91%**	0.0434	0.0428	**1.38%**	0.0422	**2.76%**	0.0418	**3.69%**
	FREEDOM	0.0627	0.0596	4.94%	0.0586	6.54%	0.0570	9.09%	0.0627	0.0596	4.94%	0.0591	5.74%	0.0584	6.86%
	+BSAM	0.0650	0.0635	2.31%	0.0630	3.08%	0.0622	4.31%	0.0650	0.0637	2.00%	0.0631	2.92%	0.0628	3.38%
	+BSAM+	0.0661	0.0652	**1.36%**	0.0648	**1.97%**	0.0643	**2.72%**	0.0661	0.0654	**1.06%**	0.0650	**1.66%**	0.0644	**2.57%**
	LGMRec	0.0647	0.0604	6.65%	0.0595	8.04%	0.0583	9.89%	0.0647	0.0620	4.17%	0.0610	5.72%	0.0602	6.96%
	+BSAM	0.0676	0.0653	3.40%	0.0644	4.73%	0.0637	5.77%	0.0676	0.0655	3.11%	0.0649	3.99%	0.0643	4.88%
	+BSAM+	0.0684	0.0669	**2.19%**	0.0662	**3.22%**	0.0657	**3.95%**	0.0684	0.0674	**1.46%**	0.0670	**2.05%**	0.0665	**2.78%**
Sports	MMGCN	0.0370	0.0326	11.89%	0.0303	18.11%	0.0282	23.78%	0.0370	0.0348	5.95%	0.0332	10.27%	0.0326	11.89%
	+BSAM	0.0404	0.0378	6.44%	0.0363	10.15%	0.0352	12.87%	0.0404	0.0388	3.96%	0.0379	6.19%	0.0374	7.43%
	+BSAM+	0.0409	0.0396	**3.18%**	0.0383	**6.36%**	0.0377	**7.82%**	0.0409	0.0400	**2.20%**	0.0392	**4.16%**	0.0388	**5.13%**
	FREEDOM	0.0717	0.0692	3.49%	0.0683	4.74%	0.0677	5.58%	0.0717	0.0694	3.21%	0.0686	4.32%	0.0681	5.02%
	+BSAM	0.0747	0.0723	3.21%	0.0718	3.88%	0.0713	4.55%	0.0747	0.0731	2.14%	0.0723	3.21%	0.0718	3.88%
	+BSAM+	0.0751	0.0737	**1.86%**	0.0732	**2.53%**	0.0728	**3.06%**	0.0751	0.0736	**2.00%**	0.0731	**2.66%**	0.0727	**3.20%**
	LGMRec	0.0719	0.0683	5.00%	0.0678	5.70%	0.0671	6.68%	0.0719	0.0690	4.03%	0.0678	5.70%	0.0671	6.68%
	+BSAM	0.0747	0.0725	2.95%	0.0721	3.48%	0.0717	4.02%	0.0747	0.0728	2.54%	0.0721	3.48%	0.0716	4.15%
	+BSAM+	0.0756	0.0741	**1.98%**	0.0736	**2.65%**	0.0732	**3.17%**	0.0756	0.0744	**1.59%**	0.0737	**2.51%**	0.0731	**3.31%**

"Decr." denotes the relative decrease compared to the original result after simulating risk. We report the results on the Recall@10 metric.

BSAM+ still exhibit satisfactory robustness enhancement under controlled noise injection settings.

D. Generalization Capability (RQ4)

Model generalization capability refers to the ability of the model to perform on unseen data beyond the training set. However, due to the data sparsity problem in recommender systems, a direct measurement of the performance gap between the training and testing sets is unconvincing [11], [32]. Therefore, we design a tailored evaluation strategy to test the generalization capability of recommendation systems. Specifically, we randomly split the dataset 10 times based on timestamps to make the test and training sets different each time. For the fairness of the comparison, we fixed 10 random seeds 996-1005, and reported the average of the results. We conduct extensive experiments of BSAM across six multimodal recommendation models using two public datasets and plot the results as a violin plot. Fig. 4 illustrates our BSAM significantly enhances the generalization

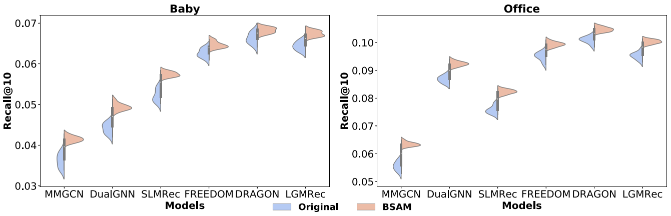


Fig. 4. Generalization capability study in terms of Recall@10.

TABLE IV
"DECR." DENOTES THE RELATIVE DECREASE COMPARED TO THE ORIGINAL RESULT AFTER SIMULATING RISK

Model	Original	L_1	Decr.	L_2	Decr.	L_3	Decr.
Baby							
MMGCN	0.0378	0.0355	6.08%	0.0348	7.94%	0.0334	11.64%
+BSAM	0.0420	0.0410	2.38%	0.0404	3.81%	0.0398	5.24%
+BSAM+	0.0434	0.0428	**1.38%**	0.0423	**2.53%**	0.0416	**4.15%**
FREEDOM	0.0627	0.0597	4.78%	0.0592	5.58%	0.0581	7.34%
+BSAM	0.0650	0.0639	1.69%	0.0633	2.62%	0.0627	3.54%
+BSAM+	0.0661	0.0655	**0.91%**	0.0650	**1.66%**	0.0641	**3.03%**
LGMRec	0.0647	0.0622	3.86%	0.0611	5.56%	0.0602	6.96%
+BSAM	0.0676	0.0657	2.81%	0.0651	3.70%	0.0644	4.73%
+BSAM+	0.0684	0.0674	**1.46%**	0.0670	**2.05%**	0.0663	**3.07%**
Sports							
MMGCN	0.0370	0.0352	4.86%	0.0335	9.46%	0.0324	12.43%
+BSAM	0.0404	0.0390	3.47%	0.0381	5.69%	0.0375	7.18%
+BSAM+	0.0409	0.0402	**1.71%**	0.0394	**3.67%**	0.0389	**4.89%**
FREEDOM	0.0717	0.0696	2.93%	0.0686	4.32%	0.0679	5.30%
+BSAM	0.0747	0.0732	2.01%	0.0724	3.08%	0.0718	3.88%
+BSAM+	0.0751	0.0737	**1.86%**	0.0731	**2.66%**	0.0728	**3.06%**
LGMRec	0.0719	0.0692	3.76%	0.0679	5.56%	0.0670	6.82%
+BSAM	0.0747	0.0729	2.41%	0.0722	3.35%	0.0717	4.02%
+BSAM+	0.0756	0.0744	**1.59%**	0.0737	**2.51%**	0.0732	**3.17%**

We report the results on the Recall@10 metric.

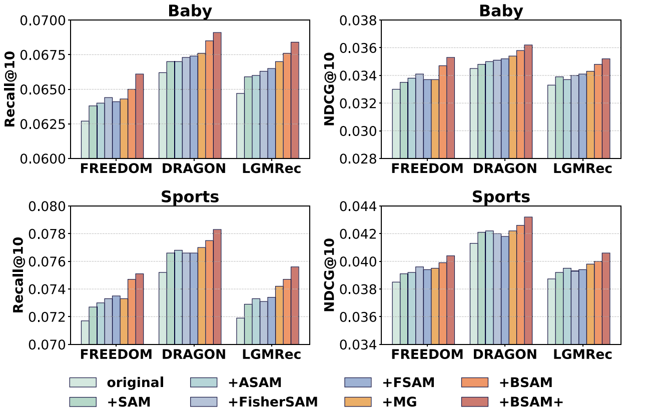


Fig. 5. Performance comparison with other flat local minima methods on Baby and Sports datasets.

capability for all multimodal recommendation models (BSAM not only can improve model performance, but also can reduce the performance difference of the model under different seeds).

E. In-Depth Analysis (RQ5 - RQ8)

1) Comparing With Flat Local Minima Methods (RQ5): The connection between the shape of loss landscape and model performance has been studied extensively, and many flat local minima methods [17], [18], [19] receive broader attention in representation learning. In detail, we apply SAM [18], ASAM [17], FisherSAM [49], and FSAM [35] for multimodal recommendation model training. Besides, MG [16] is listed as a baseline, which is a local flat minima method designed specifically for multimodal recommendations. To sum up, we compare these methods with our BSAM and BSAM+ in the multimodal recommendations scenario. We conduct extensive experiments of BSAM and BSAM+ across three multimodal recommendation models using two public datasets. As Fig. 5 shows, BSAM and BSAM+ outperform all baselines under two metrics. We have two findings, 1) BSAM and BSAM+ significantly outperform all SAM-based methods, which suggests that in the multimodal recommendation domain, we should pay more attention to the local flat minima of each batch and reduce the interference caused by the loss landscape of the entire dataset. We attribute this phenomenon to the data sparsity problem in the recommendation domain. 2) BSAM and BSAM+ outperform MG, which proves that the SAM approach is still powerful in the recommendation domain. We state that SAM can more intuitively perceive and smooth the sharpness of the loss landscape compared to the loss regularization in MG.

2) Compatibility With Various Optimizers (RQ6): One of the advantages of the SAM-based method is its adaptability to various optimizers. Therefore, we further investigated whether BSAM and BSAM+ can achieve desirable results under different optimizers. We selected Adam [47], RMSprop [50], and Adagrad [51] as the optimizers for our study (Note that SGD [52], [53] was excluded due to its generally poor performance in the multimodal recommendation scenario). We conducted extensive experiments with BSAM and BSAM+ across two multimodal recommendation models using two public datasets. Table VI shows that BSAM and BSAM+ consistently deliver a noticeable improvement with various optimizers.

3) Compatibility With Robust Training and Data Augmentation Strategies (RQ7): Existing studies enhance the robustness of multimodal recommendations by adversarial training strategy [12], [14], [15], [54] and data augmentation method [55], [56], [57]. Therefore, we further evaluate the compatibility of our BSAM+ with the adversarial training strategy (AMR [12]) and LLM-based data augmentation strategy (GPT-4v [46]). We conducted extensive experiments across two multimodal recommendation models using four public datasets. Table VII shows that combining BSAM+ with both AMR and GPT-4v can further improve model performance. Additionally, GPT-4v generally outperforms AMR on all datasets except Office, which we attribute to GPT-4v's ability to reduce the inherent gap between visual and textual information of items. Notably, simultaneously using both AMR and GPT-4v achieves more satisfactory performance than adopting either one alone.

4) Different Batch Size (RQ8): One of the main motivations for this paper is that in the recommendation field, where data is naturally sparse, global sharpness-aware minimization does not contribute to generalization and can have a detrimental effect. Consequently, we improve generalization performance by minimizing the sharpness in the loss landscape of batch data. We further conducted experiments to investigate the impact of batch size on BSAM and BSAM+ using the three most recent models (FREEDOM, DRAGON, and LGMRec) across four public datasets. In Table V, we compare the performance under batch sizes from $\{1024, 2048, 4096\}$ while keeping other hyper-parameters unchanged. We found that both BSAM and BSAM+ achieve superior results at various batch sizes.

5) Mixed Loss for SAM Methods (RQ9): We further evaluate whether our mixed loss strategy can be applied to varying SAM methods. Therefore, we design the SAM+, ASAM+, and FisherSAM+ variants, + means using our mixed loss strategy (detailed in Section IV-B). We conduct experiments across three multimodal recommendation models using two datasets. Table VIII

TABLE V
PERFORMANCE COMPARISON OF BASELINES WITH OR WITHOUT BSAM AND BSAM+ ON FOUR DATASETS UNDER VARIOUS BATCH SIZES

Datasets		Baby				Sports				Clothing				Pet			
Batch Size	Variants	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10
1024	FREEDOM	0.0370	0.0626	0.0241	0.0328	0.0444	0.0715	0.0289	0.0383	0.0386	0.0627	0.0255	0.0339	0.0703	0.1087	0.0471	0.0597
	+BSAM	0.0395	0.0653	0.0259	0.0349	0.0467	0.0746	0.0308	0.0401	0.0404	0.0652	0.0267	0.0354	0.0738	0.1125	0.0492	0.0613
	+BSAM+	**0.0397***	**0.0659***	**0.0260***	**0.0350***	**0.0469***	**0.0748***	**0.0310***	**0.0402***	**0.0407***	**0.0654***	**0.0269***	**0.0357***	**0.0739***	**0.1130***	**0.0495***	**0.0616***
	Improv.	7.30%	5.27%	7.88%	6.71%	5.63%	4.62%	7.27%	4.96%	5.44%	4.31%	5.49%	5.31%	5.12%	3.96%	5.10%	3.18%
	DRAGON	0.0376	0.0659	0.0246	0.0342	0.0446	0.0749	0.0293	0.0410	0.0399	0.0669	0.0268	0.0363	0.0745	0.1150	0.0496	0.0628
	+BSAM	0.0392	0.0681	0.0255	0.0356	0.0464	0.0773	0.0303	0.0424	0.0429	0.0706	0.0288	0.0383	0.0779	0.1191	0.0520	0.0655
	+BSAM+	**0.0394***	**0.0686***	**0.0262***	**0.0360***	**0.0469***	**0.0781***	**0.0308***	**0.0428***	**0.0432***	**0.0717***	**0.0290***	**0.0386***	**0.0786***	**0.1201***	**0.0526***	**0.0659***
	Improv.	4.79%	4.10%	6.50%	5.26%	5.16%	4.27%	5.12%	4.39%	8.27%	7.17%	8.21%	6.34%	5.50%	4.43%	6.05%	4.94%
	LGMRec	0.0371	0.0635	0.0249	0.0325	0.0444	0.0710	0.0292	0.0380	0.0352	0.0547	0.0233	0.0294	0.0695	0.1049	0.0463	0.0577
	+BSAM	0.0388	0.0666	0.0261	0.0340	0.0460	0.0738	0.0305	0.0392	0.0380	0.0580	0.0248	0.0314	0.0729	0.1101	0.0482	0.0603
	+BSAM+	**0.0398***	**0.0670***	**0.0265***	**0.0342***	**0.0470***	**0.0744***	**0.0310***	**0.0399***	**0.0383***	**0.0593***	**0.0252***	**0.0318***	**0.0736***	**0.1112***	**0.0488***	**0.0608***
	Improv.	7.28%	5.51%	6.43%	5.23%	5.86%	4.79%	6.16%	5.00%	8.81%	8.41%	8.15%	8.16%	5.90%	6.00%	5.40%	5.37%
2048	FREEDOM	0.0374	0.0627	0.0243	0.0330	0.0446	0.0717	0.0291	0.0385	0.0388	0.0629	0.0257	0.0341	0.0705	0.1086	0.0472	0.0595
	+BSAM	0.0397	0.0650	0.0259	0.0347	0.0468	0.0747	0.0306	0.0399	0.0403	0.0655	0.0271	0.0353	0.0740	0.1127	0.0493	0.0614
	+BSAM+	**0.0402***	**0.0661***	**0.0264***	**0.0354***	**0.0476***	**0.0751***	**0.0312***	**0.0404***	**0.0409***	**0.0660***	**0.0273***	**0.0359***	**0.0744***	**0.1133***	**0.0500***	**0.0621***
	Improv.	7.49%	5.42%	8.64%	7.27%	6.73%	4.74%	7.22%	4.94%	5.41%	4.92%	6.23%	5.28%	5.53%	4.33%	5.93%	4.37%
	DRAGON	0.0380	0.0662	0.0249	0.0345	0.0449	0.0752	0.0296	0.0413	0.0401	0.0671	0.0270	0.0365	0.0747	0.1151	0.0500	0.0630
	+BSAM	0.0394	0.0685	0.0257	0.0358	0.0466	0.0775	0.0306	0.0426	0.0431	0.0708	0.0287	0.0382	0.0779	0.1193	0.0523	0.0656
	+BSAM+	**0.0399***	**0.0691***	**0.0264***	**0.0362***	**0.0472***	**0.0783***	**0.0310***	**0.0432***	**0.0435***	**0.0717***	**0.0291***	**0.0387***	**0.0788***	**0.1203***	**0.0528***	**0.0661***
	Improv.	5.00%	4.38%	6.02%	4.93%	4.68%	4.12%	4.73%	4.60%	8.48%	6.86%	7.78%	6.03%	5.49%	4.52%	5.60%	4.92%
	LGMRec	0.0381	0.0647	0.0256	0.0333	0.0451	0.0719	0.0298	0.0387	0.0359	0.0555	0.0239	0.0302	0.0702	0.1057	0.0469	0.0584
	+BSAM	0.0401	0.0676	0.0268	0.0348	0.0471	0.0747	0.0310	0.0400	0.0384	0.0589	0.0255	0.0321	0.0738	0.1107	0.0489	0.0609
	+BSAM+	**0.0408***	**0.0684***	**0.0273***	**0.0352***	**0.0478***	**0.0756***	**0.0317***	**0.0406***	**0.0393***	**0.0600***	**0.0261***	**0.0327***	**0.0745***	**0.1121***	**0.0496***	**0.0615***
	Improv.	7.09%	5.72%	6.64%	5.71%	5.99%	5.15%	6.38%	4.91%	9.47%	8.11%	9.21%	8.28%	6.13%	6.05%	5.76%	5.31%
4096	FREEDOM	0.0389	0.0630	0.0260	0.0337	0.0450	0.0722	0.0295	0.0390	0.0392	0.0635	0.0259	0.0345	0.0708	0.1090	0.0474	0.0601
	+BSAM	0.0410	0.0656	0.0271	0.0350	0.0469	0.0751	0.0312	0.0404	0.0408	0.0659	0.0272	0.0357	0.0743	0.1133	0.0498	0.0618
	+BSAM+	**0.0413***	**0.0665***	**0.0278***	**0.0359***	**0.0479***	**0.0754***	**0.0316***	**0.0410***	**0.0412***	**0.0665***	**0.0274***	**0.0361***	**0.0747***	**0.1137***	**0.0501***	**0.0625***
	Improv.	6.17%	5.56%	6.92%	6.53%	6.44%	4.43%	7.12%	5.13%	5.10%	4.72%	5.79%	4.64%	5.51%	4.31%	5.70%	3.99%
	DRAGON	0.0385	0.0664	0.0252	0.0347	0.0451	0.0753	0.0297	0.0416	0.0403	0.0674	0.0272	0.0368	0.0748	0.1153	0.0502	0.0633
	+BSAM	0.0398	0.0687	0.0259	0.0360	0.0468	0.0778	0.0309	0.0428	0.0434	0.0710	0.0289	0.0383	0.0781	0.1194	0.0525	0.0658
	+BSAM+	**0.0400***	**0.0692***	**0.0266***	**0.0363***	**0.0474***	**0.0785***	**0.0311***	**0.0433***	**0.0437***	**0.0718***	**0.0293***	**0.0388***	**0.0790***	**0.1204***	**0.0530***	**0.0661***
	Improv.	3.90%	4.22%	5.56%	4.61%	5.10%	4.25%	4.71%	4.09%	8.44%	6.53%	7.72%	5.43%	5.61%	4.42%	5.58%	4.42%
	LGMRec	0.0382	0.0649	0.0258	0.0336	0.0452	0.0721	0.0300	0.0388	0.0361	0.0557	0.0241	0.0303	0.0700	0.1061	0.0472	0.0582
	+BSAM	0.0400	0.0674	0.0267	0.0349	0.0473	0.0747	0.0312	0.0399	0.0385	0.0587	0.0256	0.0323	0.0737	0.1110	0.0490	0.0611
	+BSAM+	**0.0406***	**0.0680***	**0.0274***	**0.0354***	**0.0480***	**0.0755***	**0.0318***	**0.0404***	**0.0392***	**0.0598***	**0.0260***	**0.0328***	**0.0743***	**0.1120***	**0.0497***	**0.0614***
	Improv.	6.28%	4.78%	6.20%	5.36%	6.19%	4.72%	6.00%	4.12%	8.59%	7.36%	7.88%	8.25%	6.14%	5.56%	5.30%	5.50%

The superscript * indicates the improvement is statistically significant where the p-value is less than 0.01.

TABLE VI
PERFORMANCE COMPARISON FOR DIFFERENT OPTIMIZERS ON TWO DATASETS IN TERMS OF RECALL@K (R@K) AND NDCG@K (N@K)

Optimizer	Datasets	Baby				Sports			
	Metrics	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10
Adam	FREEDOM	0.0374	0.0627	0.0243	0.0330	0.0446	0.0717	0.0291	0.0385
	+BSAM	0.0397	0.0650	0.0259	0.0347	0.0468	0.0747	0.0306	0.0399
	+BSAM+	**0.0402***	**0.0661***	**0.0264***	**0.0354***	**0.0476***	**0.0751***	**0.0312***	**0.0404***
	Improv.	7.49%	5.42%	8.64%	7.27%	6.73%	4.74%	7.22%	4.94%
RMSprop	FREEDOM	0.0366	0.0614	0.0238	0.0323	0.0441	0.0709	0.0288	0.0380
	+BSAM	0.0378	0.0630	0.0245	0.0333	0.0460	0.0731	0.0301	0.0393
	+BSAM+	**0.0383***	**0.0638***	**0.0249***	**0.0337***	**0.0464***	**0.0733***	**0.0305***	**0.0395***
	Improv.	4.64%	3.91%	4.62%	4.33%	5.22%	3.39%	5.90%	3.95%
Adagrad	FREEDOM	0.0332	0.0461	0.0219	0.0247	0.0387	0.0583	0.0259	0.0321
	+BSAM	0.0343	0.0475	0.0227	**0.0255***	0.0400	**0.0602***	0.0266	0.0328
	+BSAM+	**0.0348***	**0.0477***	**0.0229***	0.0254	**0.0401***	**0.0602***	**0.0268***	**0.0329***
	Improv.	4.82%	3.47%	4.57%	3.24%	3.62%	3.26%	3.47%	2.49%
Optimizer	Datasets	Baby				Sports			
	Metrics	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10
Adam	LGMRec	0.0381	0.0647	0.0256	0.0333	0.0451	0.0719	0.0298	0.0387
	+BSAM	0.0401	0.0676	0.0268	0.0348	0.0471	0.0747	0.0310	0.0400
	+BSAM+	**0.0408***	**0.0684***	**0.0273***	**0.0352***	**0.0478***	**0.0756***	**0.0317***	**0.0406***
	Improv.	7.09%	5.72%	6.64%	5.71%	5.99%	5.15%	6.38%	4.91%
RMSprop	LGMRec	0.0377	0.0650	0.0258	0.0337	0.0433	0.0701	0.0291	0.0378
	+BSAM	0.0390	0.0669	0.0262	0.0341	0.0450	0.0727	0.0301	0.0388
	+BSAM+	**0.0391***	**0.0671***	**0.0264***	**0.0345***	**0.0452***	**0.0731***	**0.0302***	**0.0390***
	Improv.	3.71%	3.23%	2.33%	2.37%	4.39%	4.28%	3.78%	3.17%
Adagrad	LGMRec	0.0330	0.0554	0.0210	0.0261	0.0400	0.0620	0.0268	0.0339
	+BSAM	0.0349	**0.0569***	0.0215	**0.0267***	0.0417	0.0641	0.0278	0.0347
	+BSAM+	**0.0352***	**0.0569***	**0.0218***	0.0265	**0.0424***	**0.0649***	**0.0280***	**0.0355***
	Improv.	6.67%	2.71%	3.81%	2.30%	6.00%	4.68%	4.48%	4.72%

* means the statistical significance for $p < 0.01$.

demonstrates that our mixed loss strategy can improve the effectiveness of the SAM method for multimodal recommender systems. However, it is worth mentioning that these variants still do not perform as well as BSAM+. We attribute this to the large difference between the local and global landscapes due to the sparsity of the recommender system. Fortunately, our BSAM effectively improves the model's robustness by reducing the impact of the global landscape.

TABLE VII
PERFORMANCE COMPARISON FOR ROBUST STRATEGIES ON FOUR DATASETS IN TERMS OF RECALL@5 (R@5) AND NDCG@5 (N@5)

Model	Model	Baby		Clothing		Pet		Office	
	Metrics	R@5	N@5	R@5	N@5	R@5	N@5	R@5	N@5
MMGCN	origin	0.0240	0.0160	0.0130	0.0110	0.0378	0.0251	0.0318	0.0223
	+B	0.0277	0.0181	0.0151	0.0102	0.0422	0.0282	0.0371	0.0253
	+B+A	0.0279	0.0182	0.0155	0.0102	0.0424	0.0283	0.0377	**0.0259**
	+B+G	0.0288	0.0188	0.0161	0.0107	0.0425	0.0286	0.0374	0.0255
	+B+A+G	**0.0289**	**0.0190**	**0.0163**	**0.0109**	**0.0430**	**0.028**8	**0.0378**	0.0258
LGMRec	origin	0.0381	0.0256	0.0359	0.0239	0.0702	0.0469	0.0600	0.0393
	+B	0.0408	0.0273	0.0393	0.0261	0.0745	0.0496	0.0637	0.0418
	+B+A	0.0404	0.0270	0.0399	0.0266	0.0748	0.0500	0.0648	0.0427
	+B+G	0.0428	0.0289	0.0405	0.0270	0.0750	0.0501	0.0640	0.0422
	+B+A+G	**0.0433**	**0.0291**	**0.0410**	**0.0276**	**0.0755**	**0.0504**	**0.0652**	**0.0431**

+B, +A, and +G denote +BSAM+, +AMR, and +GPT- 4v, respectively.

F. Convergence Speed (RQ10)

To analyze the impact of BSAM and BSAM+ on convergence speed, we visualize the training loss for three multimodal baselines on the Baby dataset. Following the previous training settings [9], [24], we set a maximum of 1000 epochs with a 20-epoch early stopping strategy. Fig. 7 reveals that while the BSAM strategy initially impedes convergence, the BSAM+ variant effectively addresses this issue and enhances convergence rates in multimodal recommendation models.

G. Visualization (RQ11)

To substantiate the assertion that both BSAM and BSAM+ contribute to guiding the model towards flatter local minima, we conducted visual analyses of the training loss landscapes on the Baby dataset, both with and without the integration of

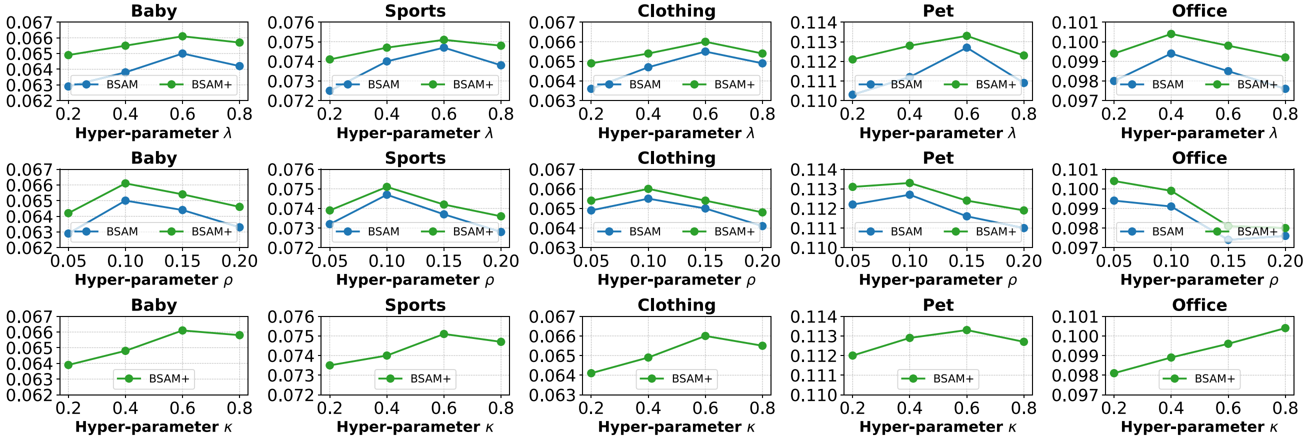


Fig. 6. Hyper-parameter study of BSAM and BSAM+, in terms of Recall@10 for FREEDOM model on all five datasets.

TABLE VIII
PERFORMANCE COMPARISON OF OUR MIXED LOSS STRATEGY IN TERMS OF RECALL@K (R@K), AND NDCG@K (N@K)

Model		MMGCN		FREEDOM		DRAGON	
Dataset	Variants	R@10	N@10	R@10	N@10	R@10	N@10
Baby	-	0.0378	0.0200	0.0627	0.0330	0.0662	0.0345
	SAM	0.0400	0.0207	0.0638	0.0335	0.0670	0.0348
	SAM+	**0.0408**	**0.0211**	**0.0640**	**0.0336**	**0.0673**	**0.0351**
	ASAM	0.0410	0.0215	0.0640	0.0337	0.0670	0.0349
	ASAM+	**0.0413**	**0.0218**	**0.0642**	**0.0338**	**0.0671**	**0.0350**
	FisherSAM	0.0412	0.0216	0.0644	0.0341	0.0673	0.0351
	FisherSAM+	**0.0419**	**0.0220**	**0.0649**	**0.0343**	**0.0678**	**0.0355**
Sports	-	0.0370	0.0193	0.0717	0.0385	0.0752	0.0413
	SAM	0.0383	0.0199	0.0727	0.0391	0.0766	0.0421
	SAM+	**0.0390**	**0.0203**	**0.0733**	**0.0394**	**0.0771**	**0.0424**
	ASAM	0.0385	0.0204	0.0730	0.0392	0.0768	0.0422
	ASAM+	**0.0388**	**0.0206**	**0.0731**	**0.0394**	**0.0771**	**0.0424**
	FisherSAM	0.0385	0.0201	0.0733	0.0396	0.0766	0.0420
	FisherSAM+	**0.0392**	**0.0208**	**0.0738**	**0.0400**	**0.0771**	**0.0425**

BSAM and BSAM+ in the most recent models (FREEDOM, DRAGON, and LGMRec). These models represent a more complex challenge compared to their predecessors. Following the approach delineated by [58], we recorded the parameters of the trained models as p. We then uniformly sampled 20 values each for m and n from the interval $[-100, 100]$, and initialized noise variables n_1 and n_2 from the standard normal distribution, ensuring that n_1 and n_2 conformed to the dimensionality of p. Subsequently, the model parameters were modified to $(p + mn_1 + nn_2)$, and the corresponding loss values were computed. This method enabled us to construct training loss landscapes, as depicted in Fig. 8, thereby providing a visual confirmation of the models' convergence behaviors towards flatter minima.

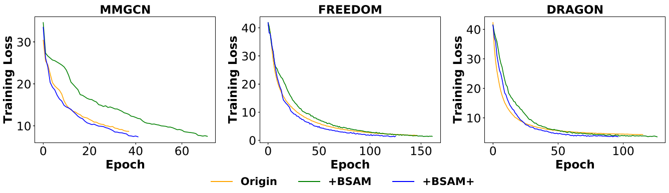
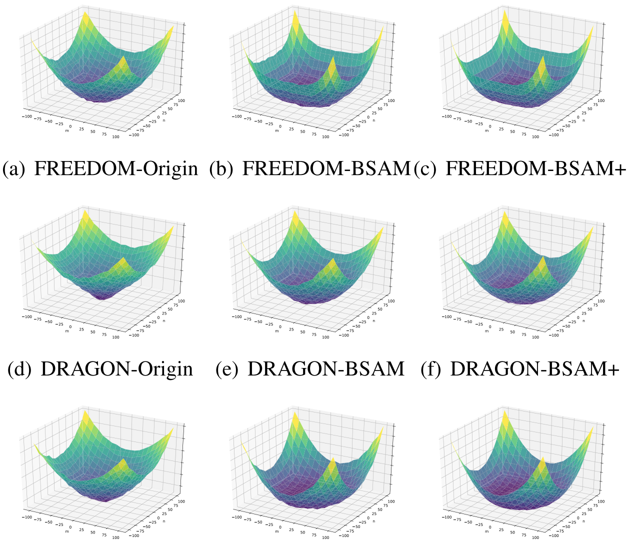


Fig. 7. Convergence study on the Baby dataset.


(a) FREEDOM-Origin (b) FREEDOM-BSAM (c) FREEDOM-BSAM+

(d) DRAGON-Origin (e) DRAGON-BSAM (f) DRAGON-BSAM+

(g) LGMRec-Origin (h) LGMRec-BSAM (i) LGMRec-BSAM+

Fig. 8. Visualization of local minima. Training loss landscapes of FREEDOM, DRAGON, and LGMRec on Baby trained with or without BSAM and BSAM+.

H. Parameter Study (RQ12)

We evaluate the impact of key hyper-parameters on BSAM and BSAM+ performance on all five datasets in terms of Recall@10. From Fig. 6, we have the following observations:

Hyper-parameter λ for BSAM and BSAM+: The optimal hyper-parameter λ is 0.6 for all datasets, except 0.4 for Office. Note that BSAM+ is more stable than BSAM under different λ, which we attribute to the fact that our hybrid loss strategy can effectively balance the global loss landscape with the local loss landscape.

Hyper-parameter ρ for BSAM and BSAM+: The optimal hyper-parameter ρ is determined to be 0.10 across all datasets, except 0.05 for Office. A larger value of ρ facilitates the minimization of the loss of landscape sharpness over a wider area. However, too large ρ may cross the valley [34].

Hyper-parameter κ for BSAM+: The optimal hyper-parameter κ is 0.6 for all datasets, except 0.8 for Office. This suggests that in a mixed loss strategy, emphasis should be placed on sharpness-aware minimization, with the conventional loss serving as an auxiliary task.

The optimal hyper-parameter settings for the Office dataset differ from other datasets, attributed to its denser user-item interactions and more relevant textual and visual features.

VI. Related Work

A. Multimodal Recommender System

Numerous recent studies integrate multimodal information to address the data sparsity challenge in recommendation systems. VBPR [5] pioneered this approach, using visual content to mitigate data sparsity through matrix factorization [22]. Furthermore, several works [59], [60], [61] enhance item representations with both visual and textual modalities to further alleviate the data sparsity issue. Inspired by traditional recommendation systems, MMGCN [6] employs GCN to construct a bipartite graph that extracts latent information from user-item interactions. GRCN [7] prunes false-positive edges based on MMGCN to reduce noise in the user-item bipartite graph. To explicitly mine common preferences between users, DualGNN [42] constructs an additional user co-occurrence graph. LATTICE [8] introduces an item semantic graph to capture latent correlative signals between items. FREEDOM [9] builds on LATTICE by freezing the item semantic graph. DRAGON [43] and COHESION [62] are dedicated to unleashing the representational power of composite graphs. Recently, LGMRec [41] and PMMIR [63] explore the effectiveness of hyper-graph structures and hierarchical state representation in the multimodal recommendation, respectively. Additionally, modality alignment has garnered considerable attention. MENTOR [25] introduces an ID-guided multi-level alignment paradigm, while DiffMM [64] incorporates diffusion-based denoising components to enhance alignment robustness.

B. Sharpness-Aware Minimization

Flat local minima have been consistently linked to improved generalization in deep neural networks [29], [65], [66], [67]. SAM introduced a novel mini-max optimization framework to minimize sharpness and enhance generalization. Building on SAM, ASAM [17] introduced a scale-invariant mechanism with an adaptive radius to improve training stability, while SSAM [19] focused on sparse perturbations, emphasizing critical yet sparsely distributed dimensions in the problem space. Subsequent research has extended SAM by exploring neighborhood geometry [49], surrogate loss functions [68], friendly adversaries [69], and training efficiency [70], [71], [72]. Together, these advancements deepen our understanding of optimization in deep learning models.

VII. Conclusion

In this paper, we analyze the shortcomings of existing robustness and generalization capability enhancement strategies in the multimodal recommendation field. To this end, we propose a sharpness-aware minimization strategy BSAM, which focuses on batch data and its variant BSAM+. Our strategies significantly enhance the robustness and generalization capability of multimodal recommendation systems while overcoming the high cost of hyper-parameter tuning in existing strategies. Extensive experiments conducted on various recommendation models and benchmark datasets, along with strong theoretical evidence, demonstrate the effectiveness, efficiency, compatibility, and universality of our strategies.

References

[1] S. Zhang, L. Yao, A. Sun, and Y. Tay, “Deep learning based recommender system: A survey and new perspectives,” *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–38, 2019.

[2] J. Xu, Z. Chen, Z. Ma, J. Liu, and E. C. Ngai, “Improving consumer experience with pre-purify temporal-decay memory-based collaborative filtering recommendation for graduate school application,” *IEEE Trans. Consum. Electron.*, vol. 71, no. 2, pp. 5783–5791, May 2025.

[3] J. Xu, Z. Chen, J. Li, S. Yang, H. Wang, and E. C.-H. Ngai, “AlignGroup: Learning and aligning group consensus with member preferences for group recommendation,” 2024, *arXiv:2409.02580*.

[4] J. Xu et al., “MDVT: Enhancing multimodal recommendation with model-agnostic multimodal-driven virtual triplets,” 2025, *arXiv:2505.16665*.

[5] R. He and J. McAuley, “VBPR: Visual Bayesian personalized ranking from implicit feedback,” in *Proc. AAAI Conf. Artif. Intell.*, 2016, pp. 144–150.

[6] Y. Wei, X. Wang, L. Nie, X. He, R. Hong, and T.-S. Chua, “MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video,” in *Proc. 27th ACM Int. Conf. Multimedia*, 2019, pp. 1437–1445.

[7] Y. Wei, X. Wang, L. Nie, X. He, and T.-S. Chua, “Graph-refined convolutional network for multimedia recommendation with implicit feedback,” in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 3541–3549.

[8] J. Zhang, Y. Zhu, Q. Liu, S. Wu, S. Wang, and L. Wang, “Mining latent structures for multimedia recommendation,” in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 3872–3880.

[9] X. Zhou and Z. Shen, “A tale of two graphs: Freezing and denoising graph structures for multimodal recommendation,” in *Proc. 31st ACM Int. Conf. Multimedia*, 2023, pp. 935–943.

[10] A. Salah, Q.-T. Truong, and H. W. Lauw, “Cornac: A comparative framework for multimodal recommender systems,” *J. Mach. Learn. Res.*, vol. 21, no. 95, pp. 1–5, 2020.

[11] H. Zhou, X. Zhou, Z. Zeng, L. Zhang, and Z. Shen, “A comprehensive survey on multimodal recommender systems: Taxonomy, evaluation, and future directions,” 2023, *arXiv:2302.04473*.

[12] J. Tang, X. Du, X. He, F. Yuan, Q. Tian, and T.-S. Chua, “Adversarial training towards robust multimedia recommender system,” *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 5, pp. 855–867, May 2020.

[13] C. Wu, D. Lian, Y. Ge, Z. Zhu, E. Chen, and S. Yuan, “Fight fire with fire: Towards robust recommender systems via adversarial poisoning training,” in *Proc. 44th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2021, pp. 1074–1083.

[14] Y. Du, M. Fang, J. Yi, C. Xu, J. Cheng, and D. Tao, “Enhancing the robustness of neural collaborative filtering systems under malicious attacks,” *IEEE Trans. Multimedia*, vol. 21, no. 3, pp. 555–565, Mar. 2019.

[15] R. Li, X. Wu, and W. Wang, “Adversarial learning to compare: Self-attentive prospective customer recommendation in location based social networks,” in *Proc. 13th Int. Conf. Web Search Data Mining*, 2020, pp. 349–357.

[16] S. Zhong, Z. Huang, D. Li, W. Wen, J. Qin, and L. Lin, “Mirror gradient: Towards robust multimodal recommender systems via exploring flat local minima,” in *Proc. ACM Web Conf.*, 2024, pp. 3700–3711.

[17] J. Kwon, J. Kim, H. Park, and I. K. Choi, “ASAM: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks,” in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 5905–5914.

[18] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, “Sharpness-aware minimization for efficiently improving generalization,” 2020, *arXiv: 2010.01412*.

[19] P. Mi et al., “Make sharpness-aware minimization stronger: A sparsified perturbation approach,” in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2022, pp. 30950–30962.

[20] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "LightGCN: Simplifying and powering graph convolution network for recommendation," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2020, pp. 639–648.

[21] J. Xu et al., "FourierKAN-GCF: Fourier Kolmogorov-Arnold network–an effective and efficient feature transformation for graph collaborative filtering," 2024, *arXiv:2406.01034*.

[22] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," 2012, *arXiv:1205.2618*.

[23] Z. Tao et al., "Self-supervised learning for multimedia recommendation," *IEEE Trans. Multimedia*, vol. 25, pp. 5107–5116, 2023.

[24] X. Zhou et al., "Bootstrap latent representations for multi-modal recommendation," in *Proc. ACM Web Conf.*, 2023, pp. 845–854.

[25] J. Xu, Z. Chen, S. Yang, J. Li, H. Wang, and E. C. Ngai, "MENTOR: Multi-level self-supervised learning for multimodal recommendation," in *Proc. AAAI Conf. Artif. Intell.*, 2025, pp. 12908–12917.

[26] A. V. D. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv: 1807.03748*.

[27] N. S. Keskar, J. Nocedal, P. T. P. Tang, D. Mudigere, and M. Smelyanskiy, "On large-batch training for deep learning: Generalization gap and sharp minima," in *Proc. 5th Int. Conf. Learn. Representations*, 2017, pp. 2874–2889.

[28] G. K. Dziugaite and D. M. Roy, "Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data," 2017, *arXiv: 1703.11008*.

[29] S. Hochreiter and J. Schmidhuber, "Simplifying neural nets by discovering flat minima," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 1994, pp. 529–536.

[30] G. E. Hinton and D. Van Camp, "Keeping the neural networks simple by minimizing the description length of the weights," in *Proc. 6th Annu. Conf. Comput. Learn. Theory*, 1993, pp. 5–13.

[31] Y. Jiang, B. Neyshabur, H. Mobahi, D. Krishnan, and S. Bengio, "Fantastic generalization measures and where to find them," 2019, *arXiv: 1912.02178*.

[32] J. Xu et al., "A survey on multimodal recommender systems: Recent advances and future directions," 2025, *arXiv:2502.15711*.

[33] M. Mueller, T. Vlaar, D. Rolnick, and M. Hein, "Normalization layers are all that sharpness-aware minimization needs," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2024, Art. no. 3031.

[34] M. Andriushchenko and N. Flammarion, "Towards understanding sharpness-aware minimization," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 639–668.

[35] T. Li, P. Zhou, Z. He, X. Cheng, and X. Huang, "Friendly sharpness-aware minimization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 5631–5640.

[36] S. Ghadimi and G. Lan, "Stochastic first-and zeroth-order methods for nonconvex stochastic programming," *SIAM J. Optim.*, vol. 23, no. 4, pp. 2341–2368, 2013.

[37] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, Springer, 2016, pp. 795–811.

[38] Y. Zhao, H. Zhang, and X. Hu, "Penalizing gradient norm for efficiently improving generalization in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 26982–26992.

[39] J. McAuley, C. Targett, Q. Shi, and A. Van Den Hengel, "Image-based recommendations on styles and substitutes," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2015, pp. 43–52.

[40] X. Zhou, "MMRec: Simplifying multimodal recommendation," 2023, *arXiv:2302.03497*.

[41] Z. Guo, J. Li, G. Li, C. Wang, S. Shi, and B. Ruan, "LGMRec: Local and global graph learning for multimodal recommendation," in *Proc. AAAI Conf. Artif. Intell.*, 2024, pp. 8454–8462.

[42] Q. Wang, Y. Wei, J. Yin, J. Wu, X. Song, and L. Nie, "DualGNN: Dual graph neural network for multimedia recommendation," *IEEE Trans. Multimedia*, vol. 25, pp. 1074–1084, 2023.

[43] H. Zhou, X. Zhou, L. Zhang, and Z. Shen, "Enhancing dyadic relations with homogeneous graphs for multimodal recommendation," in *Proc. 26th Eur. Conf. Artif. Intell.*, 2023, pp. 3123–3130.

[44] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2019, pp. 165–174.

[45] X. Zhou, D. Lin, Y. Liu, and C. Miao, "Layer-refined graph convolutional networks for recommendation," in *Proc. IEEE 39th Int. Conf. Data Eng.*, 2023, pp. 1247–1259.

[46] Z. Yang et al., "The dawn of LMMs: Preliminary explorations with GPT-4V (ision)," 2023, *arXiv:2309.17421*.

[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[48] S. Liang, Z. Huang, M. Liang, and H. Yang, "Instance enhancement batch normalization: An adaptive regulator of batch noise," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 4819–4827.

[49] M. Kim, D. Li, S. X. Hu, and T. Hospedales, "Fisher SAM: Information geometry and sharpness aware minimisation," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 11148–11161.

[50] A. Graves, "Generating sequences with recurrent neural networks," 2013, *arXiv:1308.0850*.

[51] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, 2011.

[52] L. Bottou, "Stochastic gradient descent tricks," in *Neural Networks: Tricks of the Trade*, 2nd ed. Berlin, Germany: Springer, 2012, pp. 421–436.

[53] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1139–1147.

[54] H. Chen and J. Li, "Adversarial tensor factorization for context-aware recommendation," in *Proc. 13th ACM Conf. Recommender Syst.*, 2019, pp. 363–367.

[55] S. Luo et al., "Integrating large language models into recommendation via mutual augmentation and adaptive aggregation," 2024, *arXiv:2401.13870*.

[56] F. Huang, Z. Yang, J. Jiang, Y. Bei, Y. Zhang, and H. Chen, "Large language model interaction simulator for cold-start item recommendation," 2024, *arXiv:2402.09176*.

[57] W. Wei et al., "LLMRec: Large language models with graph augmentation for recommendation," in *Proc. 17th ACM Int. Conf. Web Search Data Mining*, 2024, pp. 806–815.

[58] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 6391–6401.

[59] X. Chen et al., "Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2019, pp. 765–774.

[60] S. Liu, Z. Chen, H. Liu, and X. Hu, "User-video co-attention network for personalized micro-video recommendation," in *Proc. World Wide Web Conf.*, 2019, pp. 3020–3026.

[61] P. Yu, Z. Tan, G. Lu, and B.-K. Bao, "Multi-view graph convolutional network for multimedia recommendation," in *Proc. 31st ACM Int. Conf. Multimedia*, 2023, pp. 6576–6585.

[62] J. Xu, Z. Chen, W. Wang, X. Hu, S.-W. Kim, and E. C. Ngai, "COHESION: Composite graph convolutional network with dual-stage fusion for multimodal recommendation," 2025, *arXiv:2504.04452*.

[63] Y. Wu, C. Macdonald, and I. Ounis, "Personalised multi-modal interactive recommendation with hierarchical state representations," *ACM Trans. Recommender Syst.*, vol. 2, no. 3, pp. 1–25, 2024.

[64] Y. Jiang, L. Xia, W. Wei, D. Luo, K. Lin, and C. Huang, "DiffMM: Multi-modal diffusion model for recommendation," in *Proc. 32nd ACM Int. Conf. Multimedia*, 2024, pp. 7591–7599.

[65] S. Hochreiter and J. Schmidhuber, "Flat minima," *Neural Computation*, vol. 9, no. 1, pp. 1–42, 1997.

[66] H. He, G. Huang, and Y. Yuan, "Asymmetric valleys: Beyond sharp and flat local minima," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 2549–2560.

[67] G. Shi, J. Chen, W. Zhang, L.-M. Zhan, and X.-M. Wu, "Overcoming catastrophic forgetting in incremental few-shot learning by finding flat minima," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021, pp. 6747–6761.

[68] J. Zhuang et al., "Surrogate gap minimization improves sharpness-aware training," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 2553–2576.

[69] B. Li and G. Giannakis, "Enhancing sharpness-aware optimization through variance suppression," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2024, Art. no. 3104.

[70] J. Du et al., "Efficient sharpness-aware minimization for improved training of neural networks," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 9703–9720.

[71] Y. Liu, S. Mai, X. Chen, C.-J. Hsieh, and Y. You, "Towards efficient and scalable sharpness-aware minimization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 12360–12370.

[72] J. Du, D. Zhou, J. Feng, V. Tan, and J. T. Zhou, "Sharpness-aware training for free," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2022, pp. 23439–23451.

[73] J. M. Kohler and A. Lucchi, "Sub-sampled cubic regularization for non-convex optimization," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1895–1904.

[74] M. Staib, S. Reddi, S. Kale, S. Kumar, and S. Sra, "Escaping saddle points with adaptive gradient methods," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 5956–5965.

Jinfeng Xu received the BS degree in software engineering from the Beijing University of Technology, China, in 2023, and the BS degree in computer science from the University College Dublin, Ireland, in 2023. He currently working toward the PhD degree with the University of Hong Kong, China. His research interests include recommender system, multimodal learning, and graph learning.

Zheyu Chen received the MSc degree in electronic and information engineering from the Hong Kong Polytechnic University, in 2025 Spring. He is currently working toward the PhD degree with the Beijing Institute of Technology. During 2025, he is a research assistant with ASTAPLE Lab, Hong Kong Polytechnic University. His research interests include data mining, especially for the multimodal and graph-based recommendation systems.

Jinze Li received the MS degree from the University of Chinese Academy of Sciences, in 2023. He is currently working toward the PhD degree with the University of Hong Kong. His research interests include LLM efficient generation, federated learning, and multimodal machine learning.

Shuo Yang received the MEng degree from Tsinghua University, in 2023. He is currently working toward the PhD degree with the Department of Electrical and Electronic Engineering, University of Hong Kong. His research interests include cybersecurity, machine learning, and trustworthy artificial intelligence.

Wei Wang (Senior Member, IEEE) received the PhD degree in software engineering from the Dalian University of Technology, in 2018. He is currently a professor with the Artificial Intelligence Research Institute, Shenzhen MSU-BIT University, Shenzhen China, and also with the School of Medical Technology, Beijing Institute of Technology, Beijing, China. He had been the UM Macao research fellow with the University of Macau, Macau SAR. His research interests include computational social science, data mining, Internet of Things, and artificial intelligence.

Xiping Hu (Senior Member, IEEE) received the PhD degree from the University of British Columbia, Canada. He is currently a professor with the Beijing Institute of Technology, and with Shenzhen MSU-BIT University. He has more than 150 papers published and presented in top conferences and journals, such as *IEEE Transactions on Pattern Analysis and Machine Intelligence/IEEE Transactions on Mobile Computing/IEEE Transactions on Parallel and Distributed Systems/IEEE Transactions on Image Processing/IEEE Journal on Selected Areas in Communications/IEEE Communications Surveys and Tutorials*, ACM MobiCom/MM/SIGIR/WWW, AAAI, and IJCAI. He has been serving as associate editor of the *IEEE Transactions on Computational Social Systems*, and the lead guest editors of the *IEEE Internet of Things Journal* and *IEEE Transactions on Automation Science and Engineering* etc. He has been granted several key research projects with more than 50,000,000 RMB as principal investigator. He was the co-founder and CTO of Bravolol Ltd., Hong Kong, a leading language learning mobile application company with more than 100 million users, and listed as the top two language education platform globally. His research areas consist of mobile cyber-physical systems, crowdsensing, and affective computing.

Raymond Chi-Wing Wong received the BSc, MPhil, and PhD degrees in computer science and engineering from the Chinese University of Hong Kong, in 2002, 2004, and 2008, respectively. He is a professor in computer science and engineering with the Hong Kong University of Science and Technology. He is currently the associate head with the Department of Computer Science and Engineering and the director with Undergraduate Research Opportunities Program. He was the associate director with the Data Science & Technology Program, the director with the Risk Management and Business Intelligence Program, the director with the Computer Engineering Program, and the associate director with the Computer Engineering Program. His research interests include database and data mining. In 2004–2005, he worked as a research and development assistant under an R&D project funded by ITF and a local industrial company called Lifewood. His research interests include database and data mining.

Edith C. H. Ngai (Senior Member, IEEE) is currently an associate professor with the Department of Electrical and Electronic Engineering, University of Hong Kong. Before joining HKU in 2020, she was an associate professor with the Department of Information Technology, Uppsala University, Sweden. Her research interests include Internet of Things, edge intelligence, and smart cities. She was a VINNMER fellow awarded by Swedish Governmental Research Funding Agency VINNOVA. She was an area editor of the *IEEE Internet of Things Journal* from 2020 to 2022. She is currently an associate editor of *IEEE Transactions of Mobile Computing*, *IEEE Transactions of Industrial Informatics*, *IEEE Network*, *Ad Hoc Networks*, and *Computer Networks*. She has served as a program chair in IEEE GreenCom 2022, IEEE/ACM IWQoS 2024, and IEEE CloudCom 2025. She was selected as one of the N ^{2} Women Stars in computer networking and communications, in 2022. She was a distinguished lecturer in IEEE Communication Society in 2023–2024. She is a senior member of the ACM.