GradCraft: Elevating Multi-task Recommendations through Holistic Gradient Crafting

Yimeng Bai*
University of Science and Technology of China
Hefei, China
baiyimeng@mail.ustc.edu.cn

Yang Zhang†
University of Science and Technology of China
Hefei, China
zy2015@mail.ustc.edu.cn

Fuli Feng†
University of Science and Technology of China & USTC Beijing Research Institute
Hefei, China
fulifeng93@gmail.com

Jing Lu
Kuaishou Technology
Beijing, China
lvjing06@kuaishou.com

Xiaoxue Zang
Kuaishou Technology
Beijing, China
zangxiaoxue@kuaishou.com

Chenyi Lei
Kuaishou Technology
Beijing, China
leichy@mail.ustc.edu.cn

Yang Song
Kuaishou Technology
Beijing, China
yangsong@kuaishou.com

Abstract

Recommender systems require the simultaneous optimization of multiple objectives to accurately model user interests, necessitating the application of multi-task learning methods. However, existing multi-task learning methods in recommendations overlook the specific characteristics of recommendation scenarios, falling short in achieving proper gradient balance. To address this challenge, we set the target of multi-task learning as attaining the appropriate magnitude balance and the global direction balance, and propose an innovative methodology named GradCraft in response. GradCraft dynamically adjusts gradient magnitudes to align with the maximum gradient norm, mitigating interference from gradient magnitudes for subsequent manipulation. It then employs projections to eliminate gradient conflicts in directions while considering all conflicting tasks simultaneously, theoretically guaranteeing the global resolution of direction conflicts. GradCraft ensures the concurrent achievement of appropriate magnitude balance and global direction balance, aligning with the inherent characteristics of recommendation scenarios. Both offline and online experiments attest to the efficacy of GradCraft in enhancing multi-task performance in recommendations. The source code for GradCraft can be accessed at https://github.com/baiyimeng/GradCraft.

*Work done at Kuaishou.
†Corresponding author.

KDD '24, August 25–29, 2024, Barcelona, Spain.

ACM ISBN 979-8-4007-0490-1/24/08
https://doi.org/10.1145/3637528.3671585

CCS Concepts

• **Information systems** → **Recommender systems**.

Keywords

Multi-task Learning; Recommender System; Gradient Crafting

ACM Reference Format:
Yimeng Bai, Yang Zhang, Fuli Feng, Jing Lu, Xiaoxue Zang, Chenyi Lei, Yang Song. 2024. GradCraft: Elevating Multi-task Recommendations through Holistic Gradient Crafting. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3637528.3671585

1 Introduction

Recommender systems assume a pivotal role in personalized information filtering, significantly shaping individual online experiences [7, 43, 47]. The effectiveness of the systems often hinges on the ability to thoroughly model user interests, which typically entails simultaneously optimizing multiple user feedback that reflects different facets of user satisfaction [29, 30, 44]. For instance, a short video recommender system needs to optimize both the time of watching a video and the likelihood of liking it [1, 2]. Consequently, there has been an increasing trend towards applying multi-task learning in recommender systems to model the various facets of user satisfaction simultaneously [39], forming the mainstream approach in major industry applications [26, 35].

Multi-task learning aims to optimize multiple objectives simultaneously. Current approaches in recommendation predominantly involve the direct application of general multi-task optimization methods from machine learning. These methods typically focus on achieving a proper balance among tasks to prevent negative transfer effects [26] from two gradient perspectives. The first line of work involves reweighting loss, adjusting the gradient magnitudes based on specific criteria such as uncertainty [5] and update speed [4, 18, 24]

to effectively balance attention across different tasks. However, these methods exhibit limitations in handling task conflicts, showing unstable performance [12], especially when confronted with significant task heterogeneity like recommendation. The second line of work concentrates on manipulating gradient directions to diminish negative cosine similarity between tasks [21, 22, 34, 40, 42]. However, their gradient manipulation is typically executed in pairs, lacking the assurance of global non-conflict. Additionally, their manipulation overlooks interference from gradient magnitudes. These limitations significantly affect their efficacy, particularly in recommendation scenarios involving numerous tasks.

Given the pros and cons of existing methods, we distill the essence of multi-task optimization as achieving both an **appropriate magnitude balance** and a **global direction balance** on gradients, enhancing the suitability for recommendation. Firstly, it is crucial to ensure appropriate consistency in the magnitudes of gradients for heterogeneous recommendation tasks. The absence of such balance may result in certain tasks dominating others [18], thereby leading to subpar recommendation performance. At the same time, it is also imperative to completely resolve any conflicts in gradient directions across numerous recommendation tasks concurrently, thereby ensuring global non-conflict in gradients. Failure to do so could result in residual conflicts between some tasks, hindering the transfer of knowledge and finally compromising the efficacy of multi-task optimization.

In this work, we introduce *GradCraft*, a dynamic gradient balancing method for multi-task optimization. To ensure both magnitude and direction balance simultaneously, we devise a sequential paradigm that involves gradient norm alignment followed by direction projection. Initially, we dynamically align gradient norms across all tasks based on the maximum norm, establishing an appropriate magnitude balance. Subsequently, utilizing this balanced outcome, we apply projections to eliminate gradient conflicts in directions while considering all conflicting tasks concurrently, thereby ensuring global direction balance. In this sequential process, achieving direction balance hinges on attaining magnitude balance, avoiding interference from the magnitude imbalance.

Delving into further detail, in the magnitude balance, we do not pursue an absolute gradient norm alignment across different tasks; rather, we aim to prevent the norm differences from becoming too pronounced (such as spanning multiple orders of magnitude), thus averting dominance by certain tasks while preserving task specificity. In the direction balance, we move beyond mere orthogonality after projections. Our emphasis is on requiring a certain level of positive similarity to facilitate the positive transfer of knowledge across tasks, thereby enhancing conflict resolution. These design principles enable us to achieve a better balance of magnitudes and more thorough conflict resolution. We apply GradCraft to the *Progressive Layered Extraction* (PLE) [35] model and validate it through both offline and online experiments. The resulting empirical findings consistently demonstrate GradCraft's superior performance in multi-task recommendation scenarios.

The main contributions of this work are summarized as follows:

- We underscore the significance of concurrently achieving appropriate magnitude balance and global direction balance, aligning with the characteristics inherent in recommendation scenarios.
- We introduce GradCraft, an innovative methodology that incorporates a flexible magnitude adjustment approach followed by a global direction deconfliction strategy.
- We systematically conduct a series of experiments, both offline and online, showcasing GradCraft's effectiveness in improving multi-task recommendations.

2 Preliminary

2.1 Multi-task Recommendation

Multi-task recommendation aims to optimize multiple recommendation objectives simultaneously. Let $\mathcal{D}$ represent the historical data. Each sample in $\mathcal{D}$ is denoted as $(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x}$ represents the features of a user-item pair, and $\boldsymbol{y} = [y_1, \ldots, y_T]$ denotes T distinct task labels of user behaviors, such as Effective View [3, 44] and Like. The target is to learn a multi-task recommender model f_θ that uses $\boldsymbol{x}$ to predict the labels $\boldsymbol{y}$ by fitting $\mathcal{D}$. Each task involves the prediction of a specific label y_i, and corresponds to a specific loss objective ℓ_i, which can be expressed as

$$\ell_i = L(f_\theta(\boldsymbol{x})_i, y_i; \mathcal{D}) \quad (i = 1, \ldots, T), \tag{1}$$

where L denotes the common recommendation loss function, such as Binary Cross Entropy (BCE) loss [46] and Mean Squared Error (MSE) loss [10]. Here, for briefness, we omit the regularization term which is widely adopted to prevent overfitting.

Multi-task Optimization. To optimize the multiple objectives, existing methodologies adhere to a unified paradigm: initially, the gradients of different tasks are manipulated and then combined into a single gradient using specialized methods; subsequently, the model parameters are updated according to the combined result. Each task gradient can be obtained through backpropagation. Formally, the gradient of the i-th task can be represented as

$$g_i = \nabla_\theta \ell_i \in \mathbb{R}^d, \tag{2}$$

where d denotes the dimension of the model parameters. Here, without loss of generality, we treat θ and its gradient as a row vector, even though their original form can be a matrix or tensor.

2.2 Gradient Balance

Recommendation tasks often exhibit the significant heterogeneity across various aspects, such as data sparsity [2, 3]. This heterogeneity can lead to differences in gradient magnitudes and inconsistencies in update directions among tasks, leading to potential negative transfer effects [26]. To mitigate such effects, it is essential to achieve magnitude and direction balance.

2.2.1 Magnitude Balance. The assessment of the magnitude of a task gradient g_i typically relies on its norm [11], denoted as $\|g_i\|$. Magnitude balance concerns the consistency in the magnitudes of different task gradients, aiming to prevent situations where tasks i and j exhibit a significant difference in magnitudes, expressed as

$$\|g_i\| \gg \|g_j\| \quad or \quad \|g_i\| \ll \|g_j\|. \tag{3}$$

The lack of magnitude balance may result in specific tasks exerting dominance over the optimization process, ultimately leading to the sub-optimal recommendation performance [18].

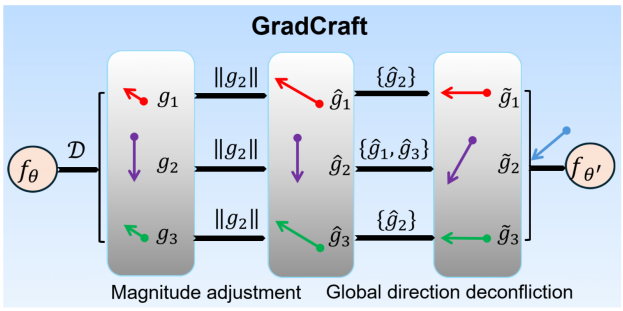

Figure 1: An overview of GradCraft. It initially adjusts the gradient magnitude based on the maximum norm. Subsequently, it performs gradient projections based on the conflicting task gradients and aggregates the gradients to update the recommender model, globally deconflicting in directions.

2.2.2 Direction Balance. Direction balance is aimed at averting conflicts between different tasks, where conflicts are defined by a negative cosine similarity between two task gradients [40, 42]. Specifically, task gradients g_i and g_j are considered in conflict if the inner product between them holds

$$\langle g_i, g_j \rangle < 0. \tag{4}$$

Achieving direction balance involves eliminating such negative similarities. The lack of direction balance can hinder the knowledge transfer among different recommendation tasks, finally compromising the efficacy of multi-task optimization.

3 Methodology

In this section, we commence by furnishing an overview of the proposed methodology. Subsequently, we introduce the magnitude adjustment approach aimed at achieving appropriate magnitude balance, and present the proposed global direction deconfliction strategy, which aims to attain global direction balance. Finally, we delve into a discussion of our gradient projection method.

3.1 Overview

We aim to achieve a simultaneous balance in both the gradient magnitude and direction. To accomplish this, we propose a sequential paradigm that involves aligning gradient norms followed by projection operations, as illustrated in Figure 1. Firstly, we dynamically align gradient norms across all tasks based on the maximum norm, establishing an appropriate balance in magnitudes. Secondly, using this balanced outcome, we apply projections to eliminate gradient conflicts while considering all conflicting tasks concurrently, ensuring a global balance in directions. Finally, we merge the gradients and update the recommender model. Given that our method operates at the gradient level, we name it *GradCraft*.

3.2 Magnitude Adjustment

In order to mitigate interference arising from differences in gradient magnitudes across tasks, our primary focus lies in the adjustment of gradients to ensure an appropriate level of magnitude balance. Rather than pursuing absolute uniformity of gradient norms across different tasks, we aim to prevent excessive differences in norms, such as those spanning multiple orders of magnitude. This helps avert dominance by certain tasks while preserving task specificity. To achieve this, for each task, we adjust its gradient norm by combining its original norm with the maximum norm among tasks. Formally, the adjustment is performed as

$$\hat{g}_i = \tau \frac{\max_j \|g_j\|}{\|g_i\|} g_i + (1-\tau) g_i, \tag{5}$$

where g_i represents the original gradient of task i, $\max_j \|g_j\|$ is the maximum gradient norm among all tasks, and $\hat{g}_i$ denotes the adjusted task gradient. The hyper-parameter $\tau \in [0, 1]$ undergoes tuning based on validation performance. In this manner, we ensure that the difference between the maximum and minimum gradient norms among tasks does not exceed $\frac{1}{\tau}$ times.

3.3 Global Direction Deconfliction

After adjusting the magnitudes, we aim to achieve the global gradient balance. For each task, we utilize projections to ensure its gradient does not conflict with the gradients of all other tasks concurrently. Subsequently, we linearly combine the deconflicted gradients from all tasks for the final model updating.

Gradient projection. For a given task gradient $\hat{g}_i$, we denote the gradients conflicting with it as $G_i = [\hat{g}_{i_1}, \ldots, \hat{g}_{i_n}] \in \mathbb{R}^{n \times d}$, where $\hat{g}_{i_j}$ represents the j-th conflicting gradient. We define a projection target to achieve non-negative similarities between the deconflicted gradient and all conflicting gradients as

$$\begin{aligned} G_i \tilde{g}_i^\top &= z, \\ z &= [\epsilon \|\hat{g}_i\| \|\hat{g}_{i_1}\|, \ldots, \epsilon \|\hat{g}_i\| \|\hat{g}_{i_n}\|], \end{aligned} \tag{6}$$

where $\tilde{g}_i$ represents the deconflicted task gradient, and $\epsilon \geq 0$ serves as a factor for adjusting the desired similarity, with a higher value indicating higher positive similarity. Notably, instead of solely pursuing gradient orthogonality ($\epsilon = 0$) between tasks, we require a certain level of positive similarity to emphasize the positive transfer of knowledge across tasks, thereby enhancing conflict resolution.

Theoretically, the desired gradient $\tilde{g}_i$ could be obtained as the sum of the original gradient and the projection onto the linear space of all conflicting gradients, which can be formulated as

$$\tilde{g}_i = \hat{g}_i + \sum_{k=1}^{n} w_k \hat{g}_{i_k} = \hat{g}_i + \boldsymbol{w}^\top G_i, \tag{7}$$

where $\boldsymbol{w} \in \mathbb{R}^{n \times 1}$ is a weight vector that needs to be determined. Combining Equation (6) and Equation (7), we deduce that

$$G_i G_i^\top \boldsymbol{w} = -G_i \hat{g}_i^\top + z. \tag{8}$$

Given that the dimension of model parameters significantly exceeds the number of tasks, *i.e.*, $d \gg n$, it is reasonable to assume that the matrix G_i possesses full rank [16]. Consequently, the positive definiteness of $G_i G_i^\top$ can be attained, enabling the weight vector $\boldsymbol{w}$ to be solved in closed form as

$$\boldsymbol{w} = (G_i G_i^\top)^{-1}(-G_i \hat{g}_i^\top + z). \tag{9}$$

Algorithm 1: GradCraft

Input: Recommender model f_θ, training dataset $\mathcal{D}$, task number T, hyper-parameter τ and ϵ, learning rate η

1 Initialize θ randomly;
2 **while** *stop condition is not reached* **do**
3 // Step 1 (computation of task gradients);
4 **for** $i = 1, \ldots, T$ **do**
5 Compute l_i with Equation (1);
6 Compute g_i with Equation (2);
7 **end**
8 // Step 2 (magnitude adjustment);
9 **for** $i = 1, \ldots, T$ **do**
10 Compute $\hat{g}_i$ with Equation (5);
11 **end**
12 // Step 3 (gradient projection);
13 **for** $i = 1, \ldots, T$ **do**
14 $\tilde{g}_i = \hat{g}_i$;
15 **if** *conflicting gradient set is not empty* **then**
16 Solve $\boldsymbol{w}$ with Equation (9);
17 Compute $\tilde{g}_i$ with Equation (7);
18 **end**
19 **end**
20 // Step 4 (update of model parameters);
21 Update θ with Equation (10);
22 **end**
23 return f_θ

Gradient combination. After deconflicted gradients for all tasks are obtained, we linearly combine them and utilize the aggregated gradient to update the model, which is formulated as

$$\theta' = \theta - \eta \frac{1}{T} \sum_{i=1}^{T} \tilde{g}_i, \tag{10}$$

where η denotes the learning rate.

Algorithm summarization. The intricacies of GradCraft are elucidated in Algorithm 1. During the implementation phase, updates are performed on a batch of data. In each iteration, the algorithm commences by computing all task gradients (lines 4-7). Subsequently, it applies magnitude adjustments to ensure an appropriate magnitude balance, avoiding interference from the gradient magnitude (lines 9-11). Following this, if conflicts arise among task gradients, a gradient projection method is employed to ensure a global direction balance for each task (lines 13-19). Ultimately, the gradients for different tasks are combined to update the model parameters (line 21). It is important to highlight that the update process is adaptable enough to accommodate various optimizers such as Adam [15] and Adagrad [25]. Besides, the update process exclusively involves updating the shared model parameters, which aligns with the approach established in previous research [42].

3.4 Discussion

Our gradient projection method can be viewed as an extension of the normal projection method [42]. Under certain circumstances, our method could degrade to approximate equality with the normal projection method. Specifically, when each given task gradient g_i confronts only a single conflicting gradient denoted as g_{i_1}, and the hyper-parameter ϵ is set to 0, the deconflicted gradient in Equation (7) is computed as

$$\tilde{g}_i = \hat{g}_i - \frac{\langle \hat{g}_i, \hat{g}_{i_1} \rangle}{\|\hat{g}_{i_1}\|} \hat{g}_{i_1}. \tag{11}$$

Disregarding the magnitude adjustment to gradients here, this computation aligns with the normal conflict projection method.

In comparison, our method simultaneously addresses all conflicting tasks for each task while requiring a certain level of positive similarity, resulting in global and thorough conflict resolution. Notably, our method does not significantly introduce extra computation complexity. Considering $G_i G_i^\top \in \mathbb{R}^{n \times n}$, where n is the number of conflicting task gradients for g_i, we can efficiently compute its inverse in Equation (9) and obtain deconflicted gradients.

Table 1: Statistical details of the evaluation datasets.

Dataset	#User	#Item	#Intersection	Density
Wechat	19,997	59,322	7,154,154	0.0060
Kuaishou	8,516	62,699	2,867,290	0.0054

4 Experiment

In this section, we conduct a series of experiments to answer the following research questions:

RQ1: How does GradCraft perform on recommendation data compared to existing multi-task learning methods?

RQ2: What is the impact of the individual components of GradCraft on its effectiveness?

RQ3: How do the specific hyper-parameters of GradCraft influence its recommendation performance?

RQ4: How is the scalability of GradCraft across different levels of the gradient imbalance?

RQ5: How effective is GradCraft when applied to real industry recommender systems?

4.1 Experimental Setting

4.1.1 Datasets. We conduct extensive experiments on an open-world dataset and our product dataset: Wechat and Kuaishou.

- **Wechat**. This public dataset is released as part of the WeChat Big Data Challenge[1], capturing user behaviors on short videos over a two-week period. To ensure dataset quality, we applied a 10-core filtering process, ensuring that each user/video has a minimum of 10 samples.
- **Kuaishou**. This dataset is sourced from our Kuaishou[2] platform, reflecting a real-world scenario for short video recommendations. It comprises short video recommendation records for 10,000 users over a five-day period. Due to the sparser nature of the dataset, we applied a 20-core filtering process during preprocessing.

The summary statistics of the preprocessed datasets are presented in Table 1. Each dataset contains rich features of user and

[1] https://algo.weixin.qq.com/
[2] https://kuaishou.com/

video, along with diverse user feedback. We randomly split them into training, validation, and test sets, following an 8:1:1 ratio.

In short-video recommendation, there are two types of tasks: those related to viewing behaviors and those related to interactive behaviors. Therefore, we set user usage time and engagement as our optimization objectives, which are assessed using viewing labels and engagement labels. Specifically, we select EffectiveView (**EV**) [44], LongView (**LV**) [44], and CompleteView (**CV**) [1] as viewing labels. EV indicates whether the watch time of an example has exceeded 50% of the overall watch time in the dataset, while LV indicates whether the watch time has exceeded 75%. CV reflects whether the watch time of an example has surpassed the video duration. For engagement labels, we directly use **Like**, **Follow**, and **Forward**. All labels above are binary and fitted with BCE loss.

4.1.2 Baselines. We compare the proposed GradCraft with the following multi-task learning methods.

- **Single**. This approach successively assigns a weight of 1 to a specific task and assigns weights of 0 to other tasks.
- **EW**. This method assigns a equal weight of $1/T$ to each task, where T represents the total number of tasks.
- **UC** [5]. This method reweighs loss based on the uncertainty.
- **DWA** [24]. This approach adapts the loss weights by considering the update speed of the loss value.
- **MGDA** [34]. This method manipulates gradients to achieve a local Pareto optimal solution.
- **PCGrad** [42]. This method addresses the gradient conflict in directions by the pair-wise projection.
- **GradVac** [40]. This approach sets adaptive gradient similarity objectives in a learnable manner to improve PCGrad.
- **CAGrad** [21]. This method identifies the optimal update vector within a ball around the average gradient, maximizing the worst local improvement between tasks.
- **IMTL** [22]. This approach learns weights to ensure that the aggregated gradient has equal projections onto each task gradient.
- **DBMTL** [18]. This approach guarantees that all task gradients share the same magnitude as the maximum gradient norm.

As our gradient projection method can be considered an extension of the normal projection method in PCGrad, we also introduce a variant of PCGrad to ensure fair comparisons, denoted as

- **PCGrad+**. This variant takes into account magnitude balance and adjusts gradient magnitudes based on Equation (5), building upon the foundation of PCGrad.

4.1.3 Evaluation Metrics. In order to conduct a comprehensive evaluation of performance with respect to optimizing multiple recommendation objectives, we employ two widely recognized accuracy metrics: AUC and GAUC [3]. Following previous work [18, 19, 36], we mainly focus on the average performance across all tasks. Specifically, we utilize both the average metric across all tasks and the relative metric improvement compared with the Single baseline across all tasks, which can be expressed as

$$\text{AV-A}(\mathcal{M}) = \frac{1}{T}\sum_{i=1}^{T}\text{AUC}_i(\mathcal{M}), \tag{12}$$

$$\text{AV-G}(\mathcal{M}) = \frac{1}{T}\sum_{i=1}^{T}\text{GAUC}_i(\mathcal{M}), \tag{13}$$

$$\text{RI-A}(\mathcal{M}) = \frac{1}{T}\sum_{i=1}^{T}\frac{\text{AUC}_i(\mathcal{M}) - \text{AUC}_i(Single)}{\text{AUC}_i(Single)}, \tag{14}$$

$$\text{RI-G}(\mathcal{M}) = \frac{1}{T}\sum_{i=1}^{T}\frac{\text{GAUC}_i(\mathcal{M}) - \text{GAUC}_i(Single)}{\text{GAUC}_i(Single)}. \tag{15}$$

Here, $\mathcal{M}$ represents the specific multi-task learning method, with AV-A and AV-G denoting the average value of AUC and GAUC, respectively. Similarly, RI-A and RI-G signify the relative improvement in AUC and GAUC, respectively. Across all metrics, higher values indicate better recommendation results.

4.1.4 Implementation Details. To ensure fair comparisons, we employ the PLE [35] model as the backbone recommender model for all the methods under consideration. Each task is composed of a shared expert, a task-specific expert, a gate network, and a tower network. The experts are instantiated as DeepFM [8], combining a Factorization Machine (FM) [31] component with a Multi-Layer Perceptron (MLP) [6] module. The hidden layer configuration for the MLP is set to $256 \times 128 \times 64$. The tower network is implemented as an MLP with a hidden layer configuration of 32×16. The gate network structure is based on a linear layer with Softmax [41] serving as the activation function. The embedding size is consistently set to 16 for all user and video features.

In terms of model optimization, we employ the Adam optimizer [15], setting the maximum number of optimization epochs to 1000. Optimal models are identified based on validation results, utilizing an early stopping strategy with a patience setting of 10. Parameters for the backbone recommender model are initialized using a Gaussian distribution, where the mean is fixed at 0, and the standard deviation is set to 0.01. The dropout ratio is set to 0.2. We leverage the grid search to find the best hyper-parameters. For our method and all baselines, we search the learning rate in the range of {1e-4, 5e-4, 1e-3}, the size of mini-batch in the range of {2048, 4096}, and the L_2 regularization coefficient in {0, 1e-6, 1e-5, 1e-4, 1e-3}. For the special hyper-parameters of baselines, we search most of them in the ranges provided by their papers. Regarding our methodology, the hyper-parameter τ in Equation (5) to regulate the closeness to the maximum norm is searched within the interval $[0, 1]$ using a step size of 0.1, and the hyper-parameter ϵ in Equation (6) to achieve the desired similarity is searched in the range of {0, 1e-12, 1e-11, 1e-10, 1e-9, 1e-8, 1e-7}.

4.2 Performance Comparison (RQ1)

We begin by assessing the overall performance of the compared methods in optimizing multiple objectives. The summarized results are presented in Table 2, yielding the following observations:

- GradCraft demonstrates superior performance compared to the baselines on both datasets, excelling in metrics of AV-A, AV-G, RI-A, and RI-G. This highlights its ability to achieve the appropriate magnitude balance and global direction balance, showcasing its efficacy in multi-task optimization.
- Although PCGrad+ shows improvement over PCGrad by integrating the magnitude balance, it still falls short of GradCraft.

Table 2: Performance comparison between the baselines and our GradCraft on Wechat and Kuaishou, where the best results are highlighted in bold and sub-optimal results are underlined. The labels Follow and Forward are respectively abbreviated as Fol and For for simplicity. AV-A and AV-G denote the average value of AUC and GAUC across different tasks, respectively. Similarly, RI-A and RI-G signify the relative improvements of AUC and GAUC.

							Wechat						
Method		Single	EW	UC	DWA	MGDA	PCGrad	PCGrad+	GradVac	CAGrad	IMTL	DBMTL	GradCraft
AUC	EV	0.7641	0.7641	0.7633	0.7646	0.7569	0.7651	0.7644	0.7648	0.7647	0.7629	0.7636	**0.7653**
	LV	0.8484	0.8484	0.8479	0.8490	0.8429	**0.8491**	0.8486	0.8489	0.8489	0.8478	0.8479	0.8490
	CV	0.7610	0.7604	0.7596	**0.7620**	0.7515	0.7614	0.7611	0.7613	0.7614	0.7589	0.7597	0.7616
	Like	0.8661	0.8664	0.8671	0.8656	0.8604	**0.8675**	0.8668	0.8665	0.8662	0.8669	0.8650	0.8661
	Fol	0.8829	0.8810	0.8763	0.8809	0.8803	0.8825	0.8827	0.8791	0.8801	0.8827	0.8750	**0.8888**
	For	0.8940	**0.9012**	0.9006	0.8983	0.8937	0.8968	0.9000	0.8991	0.9003	0.9008	0.8987	0.9001
	AV-A	0.8361	0.8369	0.8358	0.8367	0.8309	0.8371	0.8373	0.8366	0.8369	0.8367	0.8350	**0.8385**
	RI-A	0.000%	0.091%	-0.038%	0.078%	-0.639%	0.118%	0.135%	0.065%	0.099%	0.056%	-0.129%	**0.278%**
GAUC	EV	0.6207	0.6209	0.6194	0.6189	0.6055	**0.6226**	0.6195	0.6218	0.6200	0.6201	0.6178	0.6221
	LV	0.7731	0.7745	0.7740	0.7739	0.7684	0.7754	0.7736	**0.7755**	0.7743	0.7742	0.7732	0.7751
	CV	0.6499	0.6503	0.6489	0.6499	0.6345	0.6515	0.6493	0.6509	0.6491	0.6488	0.6464	**0.6518**
	Like	0.6324	0.6382	0.6405	0.6368	0.6328	**0.6422**	0.6380	0.6384	0.6390	0.6393	0.6385	0.6383
	Fol	0.6847	0.6820	0.6962	0.6915	0.6874	0.6899	0.6870	0.6721	0.6930	0.6894	0.6896	**0.7003**
	For	0.7012	0.7129	0.7154	0.7141	0.7021	0.7164	0.7140	0.7152	0.7135	0.7144	0.7124	**0.7176**
	AV-G	0.6770	0.6798	0.6824	0.6809	0.6718	0.6830	0.6802	0.6790	0.6815	0.6810	0.6796	**0.6842**
	RI-G	0.000%	0.413%	0.791%	0.559%	-0.809%	0.887%	0.472%	0.288%	0.653%	0.589%	0.380%	**1.056%**

							Kuaishou						
Method		Single	EW	UC	DWA	MGDA	PCGrad	PCGrad+	GradVac	CAGrad	IMTL	DBMTL	GradCraft
AUC	EV	0.7569	0.7581	**0.7582**	0.7575	0.7400	0.7558	0.7564	0.7556	0.7560	0.7579	0.7568	0.7565
	LV	0.8263	0.8269	**0.8275**	0.8266	0.8143	0.8263	0.8265	0.8264	0.8266	0.8273	0.8265	0.8265
	CV	0.8550	0.8559	**0.8561**	0.8555	0.8421	0.8551	0.8548	0.8551	0.8548	0.8560	0.8547	0.8548
	Like	0.9347	0.9287	0.9310	0.9303	0.9297	0.9325	0.9345	0.9329	0.9340	0.9307	0.9343	**0.9347**
	Fol	0.8322	0.8463	0.8503	0.8469	0.8430	0.8444	0.8586	0.8437	0.8581	0.8503	0.8555	**0.8592**
	For	0.8156	0.8180	0.8163	0.8133	0.8118	0.8241	0.8302	0.8239	0.8288	0.8171	0.8267	**0.8309**
	AV-A	0.8368	0.8390	0.8399	0.8384	0.8302	0.8397	0.8435	0.8396	0.8431	0.8399	0.8424	**0.8438**
	RI-A	0.000%	0.280%	0.383%	0.197%	-0.817%	0.355%	0.811%	0.342%	0.758%	0.379%	0.682%	**0.844%**
GAUC	EV	0.6724	0.6746	**0.6749**	0.6738	0.6546	0.6721	0.6715	0.6718	0.6719	0.6742	0.6730	0.6718
	LV	0.7798	0.7800	**0.7810**	0.7797	0.7689	0.7797	0.7802	0.7794	0.7800	0.7801	0.7799	0.7801
	CV	0.8317	0.8317	**0.8326**	0.8313	0.8223	0.8316	0.8315	0.8317	0.8316	0.8321	0.8314	0.8315
	Like	0.6556	0.6617	0.6621	0.6616	0.6417	**0.6661**	0.6605	0.6647	0.6574	0.6624	0.6621	0.6600
	Fol	0.5987	0.6443	0.6529	0.6603	0.6176	0.6349	0.6525	0.6297	0.6490	**0.6629**	0.6375	0.6567
	For	0.5714	0.6318	0.6287	0.6299	0.6108	0.6393	0.6422	0.6405	0.6370	0.6253	**0.6450**	0.6425
	AV-G	0.6849	0.7040	0.7054	0.7061	0.6860	0.7039	0.7064	0.7030	0.7045	0.7062	0.7048	**0.7071**
	RI-G	0.000%	3.248%	3.451%	3.601%	0.457%	3.243%	3.671%	3.087%	3.351%	3.594%	3.402%	**3.791%**

This suggests that GradCraft's global gradient projection method surpasses PCGrad's pair-wise projection method, leading to the global and thorough direction deconfliction.

- In contrast, loss reweighting methods such as EW, UC, and DWA exhibit poor performance. Their reliance on overall loss values, without granular gradient analysis, limits their effectiveness in enhancing multi-task optimization. This highlights the importance of taking into account more fine-grained gradient magnitude and direction for enhanced performance.
- Methods that exclusively prioritize either magnitude balance or direction balance struggle to achieve optimal recommendation performance and may even lead to degradation (MGDA). This emphasizes the need of holistically addressing both magnitude and direction balance in multi-task recommendations.

4.3 Ablation Study (RQ2)

To enhance the multi-task recommendation performance in GradCraft, we propose the incorporation of a magnitude adjustment approach and a gradient projection method, with two hyper-parameters τ and ϵ. To substantiate the rationale behind these design decisions, we conduct an exhaustive evaluation by systematically disabling one critical design element at a time to obtain various variants. Specifically, the following variants are introduced:

- **GradCraft-fix ϵ**, which sets ϵ to 0 and uses zero vector as the projection target in Equation (6);
- **GradCraft-fix τ**, which sets τ to 1 and disables the control of the proximity of task gradients in Equation (5);
- **GradCraft-ori**, which removes the magnitude adjustment and preserves the original magnitudes without any alteration;

Table 3: Results of the ablation study for our GradCraft method on Wechat.

Method	AV-A	RI-A	AV-G	RI-G
GradCraft	0.8385	0.278%	0.6842	1.056%
GradCraft-fix ϵ	0.8382	0.250%	0.6837	0.981%
GradCraft-fix τ	0.8365	0.039%	0.6798	0.392%
GradCraft-ori	0.8370	0.113%	0.6835	0.959%
GradCraft-local	0.8371	0.118%	0.6830	0.887%

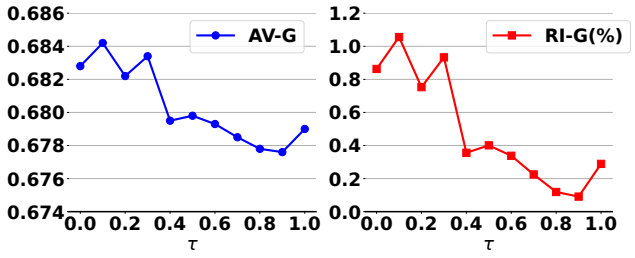

Figure 2: Results of the performance of GradCraft across different values of τ on Wechat.

- **GradCraft-local**, which removes the global gradient projection and replaces it with the normal projection in PCGrad.

Table 3 illustrates the comparison results on Wechat, from which we draw the following observations:

- When GradCraft disables the factors ϵ and the τ, there are decreases in performance across all metrics. These results confirm the pivotal role of ϵ in maintaining a certain level of positive similarity to facilitate the transfer of knowledge across tasks, and τ in controlling magnitude proximity levels.
- Comparatively, GradCraft-ori outperforms GradCraft-fix τ. These variants correspond to aligning the magnitude with the maximum norm and retaining the original magnitude, with τ set to 1 and 0, respectively. This observation suggests that indiscriminate adjustment of magnitude to match the maximum norm may detrimentally impact recommendation performance, underscoring the significance of appropriate proximity.
- The performance of GradCraft-ori and GradCraft-local is similar, indicating no advantage of global gradient projection over the normal projection strategy when magnitude adjustment is absent. However, the performance gap between GradCraft and PCGrad+ in Table 2 underscores the superiority of the gradient projection method. This outcome can be attributed to disruption caused by magnitudes, underscoring the critical role of initially adjusting magnitudes to achieve magnitude balance.

4.4 In-depth Analysis (RQ3 & RQ4)

4.4.1 The Effect of Hyper-parameter τ & ϵ. In our investigation, the two factors τ and ϵ assume pivotal roles in influencing the effectiveness of GradCraft. We undertake a systematic examination to scrutinize the impact of varying them on the performance. We report the AV-G and RI-G for simplicity, as shown in Figure 2 and Figure 3. It becomes evident that GradCraft achieves optimal AV-G

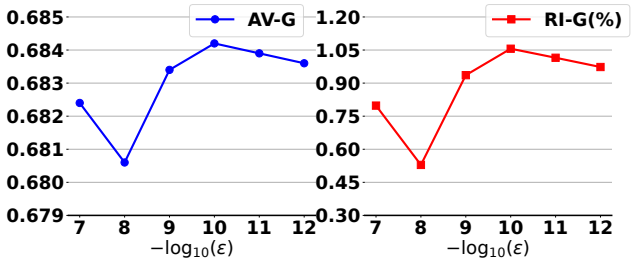

Figure 3: Results of the performance of GradCraft across different values of ϵ on Wechat.

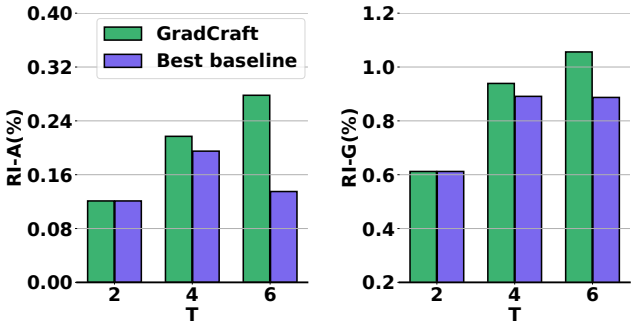

Figure 4: Results of the performance of GradCraft in comparison with the best baseline across different task number T on Wechat.

and RI-G when τ is set to 0.1 and ϵ is set to $1e$-10. However, the performance tends to deteriorate when they become excessively large. This underscores the significance of selecting an appropriate value for τ and ϵ. Further analysis reveals that when $\tau \in [0, 0.3]$ and $\epsilon \in [1e-12, 1e-9]$, the performance remains consistently stable, indicating the robustness within the range. This stability is crucial for ensuring reliable performance of the magnitude adjustment approach and the gradient projection method.

4.4.2 The Effect of Task Number T. In multi-task recommendations, the degree of gradient imbalance is intricately linked to the task number, with higher task numbers leading to an increase in the number of potential conflicting task pairs. Consequently, we conduct a comprehensive study to evaluate the impact of varying task numbers on GradCraft's performance. We also present the performance of the best baseline for comparative analysis. Specifically, we adjust the task number in the range of $\{2, 4, 6\}$ while ensuring an equal number of viewing labels and engagement labels. For $T = 2$, we designate EV and Like as the tasks, and for $T = 4$, we incorporate EV, LV, Like, and Follow. For $T = 6$, we use all the labels mentioned. We depict the relative improvement metrics RI-A and RI-G in Figure 4, and our observations are as follows:

- Both the metrics of GradCraft exhibit a consistent increase with the task number. In contrast, the best baseline method does not display a similar trend. This stark contrast suggests that GradCraft possesses a unique capability to achieve gradient balance,

Table 4: Results of the online experiment conducted over one week. It is noteworthy that performance improvements exceeding 0.1% for WT and VV, and 1.0% for Share, are considered significant [3].

	WT	VV	Share
Base	-	-	-
GradCraft	**+0.505%**	**+0.950%**	**+1.746%**

which scales up effectively with the increasing complexity introduced by a growing number of tasks. Consequently, GradCraft showcases its potential for practical application in complex recommendation scenarios. This enhanced performance can be attributed to the implementation of flexible magnitude adjustment and thorough direction conflict elimination in GradCraft.

- Moreover, as the number of tasks increases, the performance gap between GradCraft and the best baseline method widens. This expanding gap provides further evidence supporting the advantages of GradCraft in achieving both appropriate magnitude balance and global direction balance at the gradient level. It is worth mentioning that for $T = 2$, both methods yield similar results in terms of the RI-A and RI-G metrics. This similarity can be attributed to the fact that when there is only one pair of tasks, the global projection method employed by GradCraft closely resembles the normal conflict projection method. This consistency aligns with the earlier discussion presented in Equation (11).

4.5 Online Experiment (RQ5)

We conduct an online A/B experiment on our production platform, leveraging traffic from over 15 million users. We assesse three key business and engagement metrics: the average time users spend watching videos (WT), the number of effective video viewing records (VV), and the instances of video sharing (Share). Our findings, presented in Table 4, demonstrate notable performance enhancements achieved by our method compared to the state-of-the-art multi-task learning baseline implemented in Kuaishou.

5 Related work

In this section, we navigate through existing research on multi-task learning, encompassing the optimization methodologies and model architectures. Our particular emphasis is on their application within the realm of recommender system.

5.1 Multi-task Optimization

Multi-task learning necessitates the simultaneous optimization of multiple tasks. Prior research has proposed various optimization methods to mitigate the imbalance among different tasks, broadly categorized into two lines. The first category involves reweighting loss, adjusting the gradient magnitudes based on different aspects of the specific criteria [4, 5, 12, 18, 24, 39]. For example, UC [5] adjusts the loss weights according to the uncertainty associated with each task, while DWA [24] adapts the loss weights by taking into account the rate of change of the loss value. The second category focuses on manipulating gradient directions to diminish the direction conflict [21, 22, 34, 40, 42]. For instance, MGDA [34] manipulates gradients to achieve a local Pareto optimal solution. PCGrad [42] addresses gradient interference by pair-wise projections. CAGrad [21] identifies the optimal update vector within a sphere around the average gradient and maximizes the worst local improvement between tasks. IMTL [22] learns weights to ensure that the aggregated gradient has equal projections onto each task gradient. Among the mentioned works, CAGrad implicitly considers the gradient magnitude. However, it only imposes restrictions on the magnitude of the update vector, rather than finely modifying the magnitudes of each individual task like our proposed GradCraft.

In recent times, there has been a growing focus on developing tailored strategies specifically for the recommender system [1, 11, 14, 20], with a particular emphasis on diverse optimization objectives. PE-LTR [20] introduces a Pareto-efficient algorithmic framework for e-commerce recommendations. LabelCraft [1] proposes a labeling model that aligns with the objectives of short video platforms. MetaBalance [11] aims to achieve equilibrium among auxiliary losses by manipulating their gradients to enhance knowledge transfer for the target task. SoFA [14] optimizes item-side group fairness while maintaining recommendation accuracy constraints. Among these works, MetaBalance bears resemblance to our GradCraft as it incorporates adjustments to gradient magnitudes. However, MetaBalance primarily focuses on multi-behavior learning and solely optimizes performance on the target task. Additionally, it rigidly employs the gradients of the target task as adjustment criteria. In contrast, GradCraft focuses on the optimization of multiple objectives and dynamically utilizes the maximum norm of gradients across all tasks, resulting in greater adaptability and versatility.

5.2 Multi-task Model

Multi-task models aim to excel in multiple interrelated tasks simultaneously, extracting shared information to enhance proficiency in each task. While hard parameter sharing models are commonly used, they may suffer from detrimental transfer effects due to task disparities. To address this, soft parameter sharing models have been introduced, such as the cross-stitch network [28] and sluice network [32], which combine task-specific hidden layers using linear combinations. Gating and attention mechanisms have also been utilized for effective information fusion. Examples include MoE [13], which uses a gate structure to combine various experts, and MTAN [24], which incorporates task-specific attention modules within a shared network.

In recommendations, hard parameter sharing at the bottom (SharedBottom) [23] remains pervasive owing to its simplicity and efficiency, effectively addressing the oversight of task correlations in traditional models rooted in collaborative filtering and matrix factorization [9, 10, 38, 45]. MMoE [26] goes a step further by sharing all experts across diverse tasks, utilizing distinct gates for each task to augment the capabilities of the MoE framework. Conversely, ESMM [27] adopts a soft parameter sharing structure, simultaneously optimizing two correlated tasks through sequential modes to mitigate the sparsity inherent in the prediction target. Expanding upon the shared experts paradigm in MMoE, PLE [35] establishes independent experts for each task, and adopts multi-level extraction networks with progressive separation routing. Furthermore, AdaTT [17] enhances its capability by utilizing an adaptive fusion mechanism, enabling the model to more effectively select

fine-grained feature representations for individual tasks. Our work diverges from the aforementioned research as it concentrates on optimization perspective and remains model-agnostic.

6 Conclusion

This study investigated the application of multi-task learning methods in the recommender system. Recognizing the distinct characteristics of recommendations, we proposed GradCraft to simultaneously achieve an appropriate magnitude balance and a global direction balance to enhance the multi-task optimization. GradCraft dynamically adjusted the gradient magnitudes to align with the maximum gradient norm to establish the appropriate magnitude balance, mitigating interference from gradient magnitudes for subsequent manipulation. Subsequently, it employed projections to eliminate gradient conflicts in directions while considering all conflicting tasks concurrently, thereby ensuring global direction balance. Extensive experiments conducted on both real-world datasets and our production platform provided empirical evidence of its effectiveness in enhancing multi-task recommendations.

In our future work, we will enhance the comprehensiveness of our method by integrating the resolution of conflicting gradients with the improvement of consistency among other gradients. Additionally, we plan to apply our method to other domains, including Computer Vision (CV) [37] and Natural Language Processing (NLP) [33], in order to evaluate its general applicability. Moreover, we recognize the complexity of industrial recommendation scenarios and will focus on developing more effective multi-task learning methods tailored for large-scale industrial settings.

Acknowledgments

This work is supported by the National Key Research and Development Program of China (2022YFB3104701), the National Natural Science Foundation of China (62272437), and the CCCD Key Lab of Ministry of Culture and Tourism.

References

[1] Yimeng Bai, Yang Zhang, Jing Lu, Jianxin Chang, Xiaoxue Zang, Yanan Niu, Yang Song, and Fuli Feng. 2024. LabelCraft: Empowering Short Video Recommendations with Automated Label Crafting. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining* (Merida, Mexico) *(WSDM '24)*. Association for Computing Machinery, New York, NY, USA, 10 pages. https://doi.org/10.1145/3616855.3635816

[2] Qingpeng Cai, Zhenghai Xue, Chi Zhang, Wanqi Xue, Shuchang Liu, Ruohan Zhan, Xueliang Wang, Tianyou Zuo, Wentao Xie, Dong Zheng, Peng Jiang, and Kun Gai. 2023. Two-Stage Constrained Actor-Critic for Short Video Recommendation. In *Proceedings of the ACM Web Conference 2023* (Austin, TX, USA) *(WWW '23)*. Association for Computing Machinery, New York, NY, USA, 865–875. https://doi.org/10.1145/3543507.3583259

[3] Jianxin Chang, Chenbin Zhang, Yiqun Hui, Dewei Leng, Yanan Niu, Yang Song, and Kun Gai. 2023. PEPNet: Parameter and Embedding Personalized Network for Infusing with Personalized Prior Information. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 3795–3804. https://doi.org/10.1145/3580305.3599884

[4] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. 2018. GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*. PMLR, 794–803. https://proceedings.mlr.press/v80/chen18a.html

[5] Roberto Cipolla, Yarin Gal, and Alex Kendall. 2018. Multi-task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7482–7491. https://doi.org/10.1109/CVPR.2018.00781

[6] Jingtong Gao, Xiangyu Zhao, Muyang Li, Minghao Zhao, Runze Wu, Ruocheng Guo, Yiding Liu, and Dawei Yin. 2024. SMLP4Rec: An Efficient All-MLP Architecture for Sequential Recommendations. *ACM Trans. Inf. Syst.* 42, 3, Article 86 (jan 2024), 23 pages. https://doi.org/10.1145/3637871

[7] Xudong Gong, Qinlin Feng, Yuan Zhang, Jiangling Qin, Weijie Ding, Biao Li, Peng Jiang, and Kun Gai. 2022. Real-Time Short Video Recommendation on Mobile Devices. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (Atlanta, GA, USA) *(CIKM '22)*. Association for Computing Machinery, New York, NY, USA, 3103–3112. https://doi.org/10.1145/3511808.3557065

[8] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence* (Melbourne, Australia) *(IJCAI'17)*. AAAI Press, 1725–1731.

[9] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, YongDong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 639–648. https://doi.org/10.1145/3397271.3401063

[10] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia) *(WWW '17)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 173–182. https://doi.org/10.1145/3038912.3052569

[11] Yun He, Xue Feng, Cheng Cheng, Geng Ji, Yunsong Guo, and James Caverlee. 2022. MetaBalance: Improving Multi-Task Recommendations via Adapting Gradient Magnitudes of Auxiliary Tasks. In *Proceedings of the ACM Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 2205–2215. https://doi.org/10.1145/3485447.3512093

[12] Yuzheng Hu, Ruicheng Xian, Qilong Wu, Qiuling Fan, Lang Yin, and Han Zhao. 2023. Revisiting Scalarization in Multi-Task Learning: A Theoretical Perspective. *arXiv preprint arXiv:2308.13985* (2023).

[13] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation* 3, 1 (1991), 79–87.

[14] Jinqiu Jin, Haoxuan Li, Fuli Feng, Sihao Ding, Peng Wu, and Xiangnan He. 2023. Fairly Recommending with Social Attributes: A Flexible and Controllable Optimization Approach. In *Thirty-seventh Conference on Neural Information Processing Systems*.

[15] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[16] Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70* (Sydney, NSW, Australia) *(ICML'17)*. JMLR.org, 1885–1894.

[17] Danwei Li, Zhengyu Zhang, Siyang Yuan, Mingze Gao, Weilin Zhang, Chaofei Yang, Xi Liu, and Jiyan Yang. 2023. AdaTT: Adaptive Task-to-Task Fusion Network for Multitask Learning in Recommendations. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 4370–4379. https://doi.org/10.1145/3580305.3599769

[18] Baijiong Lin, Weisen Jiang, Feiyang Ye, Yu Zhang, Pengguang Chen, Ying-Cong Chen, and Shu Liu. 2023. Dual-Balancing for Multi-Task Learning. *arXiv preprint arXiv:2308.12029* (2023).

[19] Baijiong Lin, Feiyang Ye, Yu Zhang, and Ivor W Tsang. 2021. Reasonable effectiveness of random weighting: A litmus test for multi-task learning. *arXiv preprint arXiv:2111.10603* (2021).

[20] Xiao Lin, Hongjie Chen, Changhua Pei, Fei Sun, Xuanji Xiao, Hanxiao Sun, Yongfeng Zhang, Wenwu Ou, and Peng Jiang. 2019. A pareto-efficient algorithm for multiple objective optimization in e-commerce recommendation *(RecSys '19)*. Association for Computing Machinery, New York, NY, USA, 20–28. https://doi.org/10.1145/3298689.3346998

[21] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. 2021. Conflict-Averse Gradient Descent for Multi-task learning. In *Advances in Neural Information Processing Systems*, Vol. 34. Curran Associates, Inc., 18878–18890. https://proceedings.neurips.cc/paper_files/paper/2021/file/9d27fdf2477ffbff837d73ef7ae23db9-Paper.pdf

[22] Liyang Liu, Yi Li, Zhanghui Kuang, Jing-Hao Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. 2021. Towards Impartial Multi-task Learning. In *9th International Conference on Learning Representations* (Virtual Event, Austria). OpenReview.net, 12 pages. https://openreview.net/pdf?id=IMPnRXEWpvr

[23] Qi Liu, Zhilong Zhou, Gangwei Jiang, Tiezheng Ge, and Defu Lian. 2023. Deep Task-specific Bottom Representation Network for Multi-Task Recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (Birmingham, United Kingdom) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 1637–1646. https://doi.org/10.1145/3583780.3614837

[24] Shikun Liu, Edward Johns, and Andrew J Davison. 2019. End-to-End Multi-task Learning with Attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1871–1880. https://doi.org/10.1109/CVPR.2019.00197

[25] Agnes Lydia and Sagayaraj Francis. 2019. Adagrad—an optimizer for stochastic gradient descent. *Int. J. Inf. Comput. Sci* 6, 5 (2019), 566–568.

[26] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling Task Relationships in Multi-Task Learning with Multi-Gate Mixture-of-Experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (London, United Kingdom) *(KDD '18)*. Association for Computing Machinery, New York, NY, USA, 1930–1939. https://doi.org/10.1145/3219819.3220007

[27] Xiao Ma, Liqin Zhao, Guan Huang, Zhi Wang, Zelin Hu, Xiaoqiang Zhu, and Kun Gai. 2018. Entire space multi-task model: An effective approach for estimating post-click conversion rate. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1137–1140.

[28] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. 2016. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3994–4003.

[29] Yunzhu Pan, Chen Gao, Jianxin Chang, Yanan Niu, Yang Song, Kun Gai, Depeng Jin, and Yong Li. 2023. Understanding and Modeling Passive-Negative Feedback for Short-Video Sequential Recommendation *(RecSys '23)*. Association for Computing Machinery, New York, NY, USA, 540–550. https://doi.org/10.1145/3604915.3608814

[30] Yunzhu Pan, Nian Li, Chen Gao, Jianxin Chang, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2023. Learning and Optimization of Implicit Negative Feedback for Industrial Short-Video Recommender System. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (London, United Kingdom) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 4787–4793. https://doi.org/10.1145/3583780.3615482

[31] Steffen Rendle. 2010. Factorization machines. In *2010 IEEE International conference on data mining*. IEEE, 995–1000.

[32] Sebastian Ruder, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. 2017. Sluice networks: Learning what to share between loosely related tasks. *arXiv preprint arXiv:1705.08142* 2 (2017).

[33] Indu S., Srinivas N.K., Harish P.J., GangaPrasad R., Nobby Varghese, N.S. Sreekanth, and Supriya N. Pal. 2013. NLP@Desktop: a service oriented architecture for integrating NLP services in desktop clients. *SIGSOFT Softw. Eng. Notes* 38, 4 (jul 2013), 1–4. https://doi.org/10.1145/2492248.2492265

[34] Ozan Sener and Vladlen Koltun. 2018. Multi-Task Learning as Multi-Objective Optimization. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* (Montreal, Canada) *(NIPS'18)*. Curran Associates Inc., Red Hook, NY, USA, 525–536.

[35] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive Layered Extraction (PLE): A Novel Multi-Task Learning (MTL) Model for Personalized Recommendations. In *Proceedings of the 14th ACM Conference on Recommender Systems* (Virtual Event, Brazil) *(RecSys '20)*. Association for Computing Machinery, New York, NY, USA, 269–278. https://doi.org/10.1145/3383313.3412236

[36] Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. 2021. Multi-task learning for dense prediction tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence* 44, 7 (2021), 3614–3633.

[37] Deborah Walters. 2003. *Computer vision*. John Wiley and Sons Ltd., GBR, 431–435.

[38] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 165–174. https://doi.org/10.1145/3331184.3331267

[39] Yuhao Wang, Ha Tsz Lam, Yi Wong, Ziru Liu, Xiangyu Zhao, Yichao Wang, Bo Chen, Huifeng Guo, and Ruiming Tang. 2023. Multi-Task Deep Recommender Systems: A Survey. *arXiv preprint arXiv:2302.03525* (2023).

[40] Zirui Wang, Yulia Tsvetkov, Orhan Firat, and Yuan Cao. 2021. Gradient Vaccine: Investigating and Improving Multi-task Optimization in Massively Multilingual Models. In *9th International Conference on Learning Representations* (Virtual Event, Austria). OpenReview.net, 12 pages. https://openreview.net/forum?id=F1vEjWK-lH_

[41] Jiancan Wu, Xiang Wang, Xingyu Gao, Jiawei Chen, Hongcheng Fu, Tianyu Qiu, and Xiangnan He. 2023. On the Effectiveness of Sampled Softmax Loss for Item Recommendation. *ACM Trans. Inf. Syst.* (dec 2023). https://doi.org/10.1145/3637061 Just Accepted.

[42] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient Surgery for Multi-Task Learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Vancouver, BC, Canada) *(NIPS '20)*. Curran Associates Inc., Red Hook, NY, USA, Article 489, 13 pages. https://proceedings.neurips.cc/paper_files/paper/2020/file/3fe78a8acf5fda99de95303940a2420c-Paper.pdf

[43] Ruohan Zhan, Changhua Pei, Qiang Su, Jianfeng Wen, Xueliang Wang, Guanyu Mu, Dong Zheng, Peng Jiang, and Kun Gai. 2022. Deconfounding Duration Bias in Watch-Time Prediction for Video Recommendation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Washington DC, USA) *(KDD '22)*. Association for Computing Machinery, New York, NY, USA, 4472–4481. https://doi.org/10.1145/3534678.3539092

[44] Yang Zhang, Yimeng Bai, Jianxin Chang, Xiaoxue Zang, Song Lu, Jing Lu, Fuli Feng, Yanan Niu, and Yang Song. 2023. Leveraging Watch-Time Feedback for Short-Video Recommendations: A Causal Labeling Framework. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (Birmingham, United Kingdom) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 4952–4959. https://doi.org/10.1145/3583780.3615483

[45] Yang Zhang, Zhiyu Hu, Yimeng Bai, Fuli Feng, Jiancan Wu, Qifan Wang, and Xiangnan He. 2023. Recommendation unlearning via influence function. *arXiv preprint arXiv:2307.02147* (2023).

[46] Yang Zhang, Tianhao Shi, Fuli Feng, Wenjie Wang, Dingxian Wang, Xiangnan He, and Yongdong Zhang. 2023. Reformulating CTR Prediction: Learning Invariant Feature Interactions for Recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, China) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 1386–1395. https://doi.org/10.1145/3539618.3591755

[47] Haiyuan Zhao, Lei Zhang, Jun Xu, Guohao Cai, Zhenhua Dong, and Ji-Rong Wen. 2023. Uncovering User Interest from Biased and Noised Watch Time in Video Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems* (Singapore, Singapore) *(RecSys '23)*. Association for Computing Machinery, New York, NY, USA, 528–539. https://doi.org/10.1145/3604915.3608797