Understanding the Ranking Loss for Recommendation with Sparse User Feedback

Zhutian Lin*
Shenzhen International Graduate School, Tsinghua University
linzt22@mails.tsinghua.edu.cn

Junwei Pan*
Tencent Inc.
jonaspan@tencent.com

Shangyu Zhang
Tencent Inc.
vitosyzhang@tencent.com

Ximei Wang
Tencent Inc.
messixmwang@tencent.com

Xi Xiao†
Shenzhen International Graduate School, Tsinghua University
xiaox@sz.tsinghua.edu.cn

Shudong Huang
Tencent Inc.
ericdhuang@tencent.com

Lei Xiao
Tencent Inc.
shawnxiao@tencent.com

Jie Jiang
Tencent Inc.
zeus@tencent.com

ABSTRACT

Click-through rate (CTR) prediction is a crucial area of research in online advertising. While binary cross entropy (BCE) has been widely used as the optimization objective for treating CTR prediction as a binary classification problem, recent advancements have shown that combining BCE loss with an auxiliary ranking loss can significantly improve performance. However, the full effectiveness of this combination loss is not yet fully understood. In this paper, we uncover a new challenge associated with the BCE loss in scenarios where positive feedback is sparse: *the issue of gradient vanishing for negative samples*. We introduce a novel perspective on the effectiveness of the auxiliary ranking loss in CTR prediction: it *generates larger gradients on negative samples*, thereby *mitigating the optimization difficulties when using the BCE loss only* and resulting in *improved classification ability*. To validate our perspective, we conduct theoretical analysis and extensive empirical evaluations on public datasets. Additionally, we successfully integrate the ranking loss into Tencent's online advertising system, achieving notable lifts of 0.70% and 1.26% in Gross Merchandise Value (GMV) for two main scenarios. The code is openly accessible at: https://github.com/SkylerLinn/Understanding-the-Ranking-Loss.

CCS CONCEPTS

• **Information systems** → **Display advertising**; • **Computing methodologies** → **Neural networks**; **Factorization methods**.

*Both authors contributed equally to this research
†Corresponding Author

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

ACM ISBN 979-8-4007-0490-1/24/08.
https://doi.org/10.1145/3637528.3671565

KEYWORDS

Recommendation Systems; CTR Prediction; Gradient Vanishing; Ranking Loss

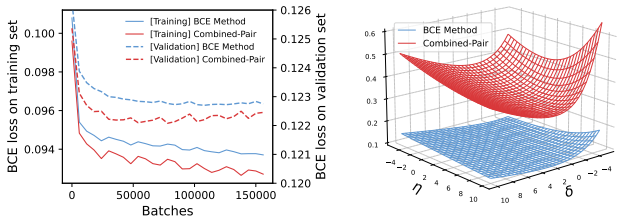
ACM Reference Format:
Zhutian Lin, Junwei Pan, Shangyu Zhang, Ximei Wang, Xi Xiao, Shudong Huang, Lei Xiao, and Jie Jiang. 2024. Understanding the Ranking Loss for Recommendation with Sparse User Feedback. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3637528.3671565

1 INTRODUCTION

In recent decades, users have encountered abundant information while navigating websites or mobile applications. This inundation presents significant challenges for electronic retailers, content providers, and online advertising platforms as they strive to recommend appropriate items to individual users within specific contexts. Thus, the deployment of recommendation systems has become widespread, enabling the prediction of users' preferences from a vast pool of candidate items.

For instance, in effective cost per mille (eCPM) advertising, advertising platforms must bid for each advertisement based on the estimated value of the impression, which relies on the bid value and the estimated Click-through rate (CTR). Consequently, accurately predicting user response has emerged as a critical factor, attracting substantial research attention.

CTR prediction [7, 11, 23, 30] is usually formulated as a binary classification problem and optimized by a binary cross entropy (BCE) loss. Some recent works [1, 18, 34, 42] in the industry propose to combine the BCE loss with an auxiliary *ranking loss*, which was usually used in Learning to Rank (LTR) [2–4, 21, 44]. For example, Combined-Pair [18] stands as one of the pioneering attempts to combine a pairwise loss with a pointwise loss for CTR prediction in the Twitter Timeline. Yan et al. and Bai et al. proposed to combine regression loss with ranking loss to better trade-off between the regression and ranking objectives [1, 42]. Sheng et al. proposed to employ two logits to optimize ranking and calibration



(a) BCE Loss Dynamics along epochs. (b) Loss Landscapes.

Figure 1: (a) BCE Loss Dynamics along epochs on the training and validation set and (b) Loss Landscape of BCE method (blue) and Combined-Pair (red).

objectives [34] jointly. Such combination loss is widely adopted in real-world recommendation systems in Twitter [18], Google [1, 42], and Alibaba [34].

The prevailing literature predominantly attributes the success of combining classification and ranking losses to the augmentation of ranking capability [18, 34]. These studies substantiate their hypothesis by observing an enhancement in the Area Under the Curve (AUC), a metric commonly employed to evaluate ranking quality. However, our curiosity lies in investigating the impact of the combination loss on the model's primary optimization objective: the classification ability, as measured by the BCE loss metric. To explore this, we conducted a comparative analysis between the BCE method, which solely employs the Binary Cross-Entropy (BCE) loss, and the Combined-Pair approach, which incorporates a BCE-ranking combination loss. Our evaluation was performed on the Criteo dataset, with artificial weights on positive samples to simulate the sparse positive scenario.

Surprisingly, our findings on the validation set, as depicted in Fig. 1(a), revealed *a reduction in the BCE loss of the Combined-Pair method (the red dashed line) compared to that of the BCE method (the blue dashed line).* This intriguing observation suggests that *the inclusion of an additional ranking loss not only enhances the model's ranking ability but also improves its classification ability.*

To investigate the cause of this improvement further, we delve into the optimization procedure during model training. In particular, we monitor the BCE loss of these two methods during model training, as the solid lines in Fig. 1(a). To our surprise, we find that *the BCE loss of the Combined-Pair (the red solid line) experiences a significant reduction compared to that of the BCE method (the blue solid line).* Moreover, we visualize the loss landscape of these two methods using Contour Plots & Random Directions [8]. Our analysis reveals that the BCE method exhibits a relatively flat landscape, indicating a slower optimization process. In contrast, the Combined-Pair method demonstrates a significantly steeper landscape, as illustrated in Fig. 1(b).

We proceed to investigate the gradients of the BCE method and the Combined-Pair. In scenarios with sparse positive feedback, such as ad CTR prediction where only a small fraction of samples are positive (clicks), we demonstrate that *negative samples get small gradients in the BCE method, leading to optimization difficulties.* However, *the ranking loss in the Combined-Pair contributes significantly larger gradients, effectively mitigating the gradient vanishing problem.*

To further validate our perspective, we conduct comprehensive experiments. Firstly, we generate artificial datasets with varying degrees of positive sample sparsity and observe that the sparser the positive samples, the greater the performance improvement achieved by incorporating an auxiliary ranking loss. Secondly, in addition to the classification-ranking combination loss, we explore alternative approaches that address the gradient vanishing issue, such as Focal Loss [20] and a new Combined-Contrastive method. Lastly, we successfully deployed the classification-ranking combination loss in the CTR prediction within Tencent's online advertising system, resulting in substantial revenue increases. In summary, our contributions can be summarized as follows:

- We uncover a challenge associated with binary cross entropy loss in recommendation scenarios with sparse positive feedback: the gradient vanishing of negative samples.
- We present a novel perspective on the effectiveness of involving an auxiliary ranking loss in recommendation systems: it introduces larger gradients for negative samples, addressing the gradient vanishing issue.
- We substantiate our claims through theoretical analysis, offline experiments, and online empirical evaluations.

2 RELATED WORK

2.1 Click-Through Rate Prediction

In online advertising, the main objective of CTR prediction is to estimate the likelihood of a user $u \in U$ clicking on a given ad $i \in I$. The input $(x, y) \sim (X, Y)$ represents an impression of an ad to a specific user, where x represents the features of that request, and $y \in \{0, 1\}$ serves as the click label. For any given model $f_\theta : \mathcal{R}^d \to \mathcal{R}$, parameterized by θ, the logit is obtained by $z = f_\theta(x)$. Commonly, it is passed through a sigmoid function $\sigma(\cdot)$ to obtain the estimated value as the output, which can be expressed as $\hat{p} = p(y = 1|x) = \sigma(z)$.

Current research in the CTR prediction task primarily focuses on model structure and optimization objectives. Regarding model structure, recent studies have primarily aimed to explore ways to capture higher-order interaction information effectively. This has led to the proposal of various model structures such as Factorization Machines (FM) family [13, 25, 31, 36], Wide&Deep [39], DeepFM [9], IPNN [29], xDeepFM [19], DCN V2 [40], Final MLP [22] and MultiEmbedding [10]. As for optimization objectives, three main loss paradigms have been introduced: pointwise, pairwise, and listwise approaches, which will be discussed in the following sections.

2.2 Binary Cross Entropy

In a pointwise approach, each item is treated *independently*, and the objective is to optimize each item's prediction or relevance score directly. One commonly used objective function in the CTR prediction task is the binary cross entropy (BCE) loss [6, 11, 23], which is defined as the cross entropy between the predicted click-through rate $\sigma(z_i)$ and the true label y. Mathematically,

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\sigma(z_i)) + (1-y_i)\log(1-\sigma(z_i))], \quad (1)$$

where N denotes the number of samples, z_i represents the logit of i-th sample, and y_i denotes the corresponding binary label, 1 for click and 0 for non-click.

2.3 Learning to Rank

In scenarios such as contextual advertising, however, the pointwise approach often falls into sub-optimality. Firstly, the pointwise approach treats each document as an individual input object, disregarding the relative order between documents. Secondly, it fails to consider the query-level and position-based properties of evaluation measures for ranking [21]. In contrast, learning-to-rank (LTR) methods can effectively address these issues and enhance ranking performance.

Specifically, pairwise and listwise approaches are the two main branches of LTR. Pairwise methods aim to ensure that the estimated value of positive samples is greater than that of negative samples for each pair of positive/negative samples. In this field, many effective works like Ranking SVM [12], GBRank [44], RankNet [3] and PRM [28] have been proposed. Among them, RankNet stands out with its clean formulation that effectively captures the essence of pairwise comparisons, which is defined as:

$$\mathcal{L}_{\text{RankNet}} = -\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}[y_{ij}\log(\sigma(z_i - z_j)) + (1-y_{ij})\log(1-\sigma(z_i - z_j))], \quad (2)$$

where $y_{ij} \in \{0, 0.5, 1\}$ corresponds to the conditions that $y_i < y_j, y_i = y_j, y_i > y_j$, respectively. Following enhancements in the optimization process [2] and the incorporation of hinge loss [38] have led to further performance improvements.

Listwise methods encourage positive samples to have higher rankings within the list of all samples. For example, ListNet [4] defines its loss as:

$$\mathcal{L}_{\text{ListNet}} = -\frac{1}{N_+}\sum_{i=1}^{N_+}\log\frac{\exp[z_i]}{\sum_{k=1}^{N}\exp[z_k]}. \quad (3)$$

Some other listwise approaches have also achieved remarkable results, such as ListMLE [41], BayesRank [15] and PiRank [37].

2.4 Tailor Ranking Loss into CTR Prediction

In the context of online advertising, recent studies [1, 34, 42] argued that it is not easy to achieve decent overall outcomes solely by relying on a single form of the objective function. Consequently, several works that combine classification loss and ranking loss have been proposed with the following objective paradigm:

$$\mathcal{L} = \alpha\mathcal{L}_{\text{Clf}} + (1-\alpha)\mathcal{L}_{\text{Rank}}, \quad (4)$$

where $\mathcal{L}_{\text{Clf}}$ and $\mathcal{L}_{\text{Rank}}$ are classification loss and ranking loss, respectively. Researchers have proposed a suite of approaches based on this optimization paradigm. Initial attempts were made with methods that combine Mean Squared Error with ranking loss [32] for the regression task. Subsequently, Combined-Pair [18], being one of the pioneers, successfully integrated BCE loss with pairwise ranking loss in the field of industrial advertising, resulting in a consistent performance improvement.

Inspired by Combined-Pair, several methods propose to combine BCE loss with other forms of ranking loss (*e.g.*, hinge loss [43], triplewise loss [33]), especially the listwise ranking loss. For example, Combined-List [42] combines BCE loss with ListNet loss, RCR [1] combines BCE loss with $\mathcal{L}_{rank}^{RCR}$, defined as:

$$\mathcal{L}_{\text{Rank}}^{\text{RCR}} = -\frac{1}{N_+}\sum_{i=1}^{N_+}\log\frac{\sigma(z_i)}{\sum_{k=1}^{N}\sigma(z_k)}, \quad (5)$$

where $\sigma(\cdot)$ represents sigmoid function. JRC [34] decouples the logit into click/non-click logits and is formalized as :

$$\mathcal{L}_{\text{Clf}}^{\text{JRC}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp[z_{iy_i}]}{\exp[z_{i0}] + \exp[z_{i1}]}, \quad (6)$$

$$\mathcal{L}_{\text{Rank}}^{\text{JRC}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp[z_{iy_i}]}{\sum_{k=1}^{N}\exp[z_{ky_i}]}, \quad (7)$$

where z_{i1} and z_{i0} are the click/non-click logits of the i-th sample, respectively. z_{ky_i} is the click logit of k-th sample if i-th samples is positive. Otherwise, z_{ky_i} signifies the non-click logit. In Sec. 3 and Sec. 4, we will conduct detailed empirical experiments and analyses on these methods.

3 AUXILIARY RANKING LOSS IN CTR PREDICTION: A GRADIENT VANISHING PERSPECTIVE

The classification-ranking combination methods are widely used in real-world recommendation systems, *e.g.*, Twitter [18], Google [1, 42], and Alibaba [34]. However, the effectiveness of such methods remains elusive, which drives us to explore the underlying mechanisms at play. In the following, we study Combined-Pair loss [18] as a representative classification-ranking combination loss and choose DCN V2 [40] as the backbone model. The Combined-Pair loss is defined as a combination of a binary cross entropy (BCE) loss and a RankNet loss [3]:

$$\mathcal{L}^{\text{CP}} = \alpha\mathcal{L}_{\text{BCE}} + (1-\alpha)\mathcal{L}_{\text{RankNet}}, \quad (8)$$

$$\mathcal{L}_{\text{RankNet}} = -\frac{1}{N_+N_-}\sum_{i=1}^{N_+}\sum_{j=1}^{N_-}\log(\sigma(z_i^{(+)} - z_j^{(-)})), \quad (9)$$

where N_+ and N_- represent the numbers of positive and negative samples, respectively. RankNet within the Combined-Pair is a form of Eq. 2 without $y_i = y_j$. In the following discussion, we name the model that optimizes the Combined-Pair loss as the Combined-Pair method, or Combined-Pair in short, and the model that optimizes only the BCE loss as the BCE method.

3.1 Investigation of Classification Ability

Existing research [18, 34] posits that the BCE Loss mainly provides a decent estimate of click probability and involving a ranking loss in addition to the BCE loss improves its ranking performance. However, we are curious about its impact on the main objective of CTR prediction: the *classification capability*. We train the BCE model and Combined-Pair on the public Criteo dataset [16]. Please note that the Criteo dataset's CTR is 25.6%, which is relatively high due to downsampling on negative samples. A weight β_{pos} for the positive samples was introduced during our model training and evaluation to simulate sparse positive rates in real-world scenarios.

We conducted monitoring of the BCE loss for both the BCE method and the Combined-Pair on the validation set, as depicted by the dashed lines in Fig. 1(a). Interestingly, we made an unexpected observation: the BCE loss of the Combined-Pair method decreases more rapidly than that of the BCE method initially and maintains a consistently lower value thereafter. This finding can be summarized as follows:

> *Finding 1. Combined-Pair gets a lower BCE loss than the BCE method on the validation set, indicating that it improves the classification ability rather than only the ranking ability.*

To investigate whether the improvement in the classification loss is due to better generalization or easier model training when incorporating the ranking loss, we monitored the BCE loss of both methods on the training set, as illustrated by the solid lines in Fig. 1(a). In addition to the previously mentioned surprising results, we made an even more astonishing observation: the BCE loss of the Combined-Pair method on the training set also decreases more rapidly and remains consistently lower than that of the BCE method. We conclude with the second finding:

> *Finding 2. Combined-Pair gets a lower BCE loss than the BCE method on the training set, indicating that involving an auxiliary ranking loss helps the optimization of the BCE loss.*

Besides, we analyze the disparities between the loss landscapes of Combined-Pair and the BCE method. As shown in Fig. 1(b), we observe that the loss landscape of the BCE method exhibits flatter than Combined-Pair.

3.2 Gradients Analysis

To gain further insights into the reasons behind the aforementioned observations, we conduct a detailed analysis of the gradients in both the BCE and Combined-Pair methods. We begin by examining the gradients of the BCE method. According to the chain rule, the gradients of the parameters in each layer are proportional to the gradients of the logits. Hence, our initial focus is on studying the logit gradients.

3.2.1 Gradients of BCE Loss for Negative Samples. The gradient of BCE loss for negative sample x_j's logit $z_j^{(-)}$ can be derived as:

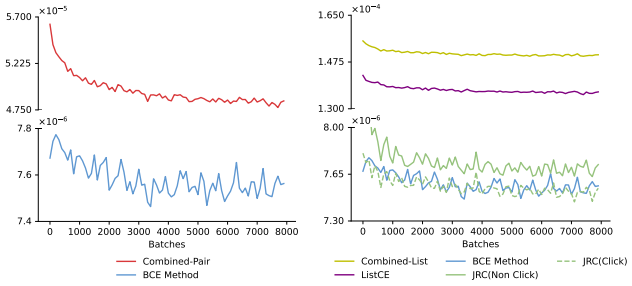


Figure 2: Gradient norm dynamics of negative samples logits in BCE method and BCE-pairwise ranking combination methods (left) and BCE-listwise ranking combination methods (right) on the Criteo dataset in the first training epoch. All methods set $\alpha = 0.5$ in both plots.

$$\begin{aligned}\nabla_{z_j^{(-)}}\mathcal{L}_{\text{BCE}} =& \frac{1}{1-\sigma(z_j^{(-)})}\cdot\sigma(z_j^{(-)})(1-\sigma(z_j^{(-)})) \\ =& \sigma(z_j^{(-)}) = \hat{p}_j. \end{aligned} \tag{10}$$

This equation demonstrates that the gradients of negative samples are proportional to its pCTR value, $\hat{p}_j$. The expected value of $\hat{p}_j$ produced by an unbiased CTR estimation model with BCE loss is close to the underlying global CTR, which equals approximately to the proportion of click samples (*i.e.*, positive feedback) to the total samples. This is because the BCE loss function is *scale calibrated* [42] and its global minima are achieved at $\sigma(z) \to \mathrm{E}[y|x]$.

When the positive feedback is sparse (*e.g.*, the CTR in our real-world advertising platform is usually less than 2%), $\hat{p}_j$ becomes a small value. According to Eq. 10, the gradients of negative samples are proportional to such *small value* of $\hat{p}_j$ and tend to be relatively small. We refer to this as *gradient vanishing of negative samples* under sparse positive feedback. We conclude this finding as:

> *Finding 3. When positive feedback is sparse, the gradients of negative samples vanish since they are proportional to the estimated positive rates, which are small in an unbiased estimator.*

3.2.2 Gradients of BCE Loss for Positive Samples. We are curious whether positive samples also exhibit similar issues. As for a given positive sample x_i, the gradient of BCE for its logit $z_i^{(+)}$ can be derived as follows:

$$\begin{aligned}\nabla_{z_i^{(+)}}\mathcal{L}_{\text{BCE}} =& -\frac{1}{\sigma(z_i^{(+)})}\cdot\sigma(z_i^{(+)})(1-\sigma(z_i^{(+)})) \\ =& -(1-\sigma(z_i^{(+)})) = -(1-\hat{p}_i). \end{aligned} \tag{11}$$

According to Eq. 11, positive samples satisfy the $\nabla_{z_i^{(+)}}\mathcal{L}_{\text{BCE}} \propto 1-\hat{p}_i$, which is a relatively large value (close to 1 when $\hat{p}_i$ is small) and therefore don't have gradient vanishing problems as negative samples do.

3.2.3 Gradients of Combined-Pair for Negative Samples. Combined-Pair contains two losses: BCE loss and RankNet loss. Here, We first discuss the gradients of the negative sample's logit in RankNet loss, which can be derived as:

$$\nabla_{z_j^{(-)}} \mathcal{L}_{\text{Rank}}^{\text{CP}} = \frac{1}{N_+} \sum_{i=1}^{N_+} \sigma(z_j^{(-)} - z_i^{(+)}). \tag{12}$$

In both our online and offline advertising, when positive feedback is extremely sparse, it is observed that even the estimated values of positive samples tend to be much lower than 0.5. Consequently, the logit of positive samples $z_i^{(+)}$ is less than 0. This can result in greater gradients of negative samples in the RankNet Loss, compared to the BCE Loss, as follows:

$$\begin{aligned} \nabla_{z_j^{(-)}} \mathcal{L}_{\text{Rank}}^{\text{CP}} =& \frac{1}{N_+} \sum_{i=1}^{N_+} \sigma(z_j^{(-)} - z_i^{(+)}) \\ >& \frac{1}{N_+} \cdot N_+ \cdot \sigma(z_j^{(-)}) \\ =& \sigma(z_j^{(-)}) = \nabla_{z_j^{(-)}} \mathcal{L}_{\text{BCE}}. \end{aligned} \tag{13}$$

This indicates that in the sparse positive scenario and for the same negative sample logit, *RankNet Loss may have larger gradients than BCE Loss.* Thus, the following inequation between the BCE method and the Combined-Pair method holds:

$$\begin{aligned} \nabla_{z_j^{(-)}} \mathcal{L}^{\text{CP}} =& \alpha \nabla_{z_j^{(-)}} \mathcal{L}_{\text{BCE}} + (1-\alpha) \nabla_{z_j^{(-)}} \mathcal{L}_{\text{Rank}}^{\text{CP}} \\ >& \alpha \nabla_{z_j^{(-)}} \mathcal{L}_{\text{BCE}} + (1-\alpha) \nabla_{z_j^{(-)}} \mathcal{L}_{\text{BCE}} \\ =& \nabla_{z_j^{(-)}} \mathcal{L}_{\text{BCE}}. \end{aligned} \tag{14}$$

We conclude with the following finding:

Finding 4. When positive feedback is sparse, Combined-Pair has larger gradients for negative samples than the BCE method.

3.3 Empirical Analysis of Gradient Vanishing

To empirically validate the analysis, we examine the dynamics of gradient norms for negative sample logits in the first epoch of training. To simulate the low proportion of positive samples often encountered in real-world scenarios, we adjust the positive sample rates, denoted as β_{pos}, to achieve an equivalent Click-Through Rate (CTR) of 3.3% for the dataset. Further details regarding this adjustment will be discussed in Sec. 4.1.

As depicted in Fig. 2, we observe that the BCE method (the blue line) exhibits negligible gradient norms for negative samples. In contrast, the Combined-Pair method (the red line) demonstrates significantly larger gradient norms for negative samples. This empirical observation aligns with our previous analysis of the gradients.

We further investigate the optimization procedure of the trainable parameters in the entire model architecture. Specifically, we compare the Combined-Pair method with the BCE method by examining the gradient norms of the trainable parameters in the bottom layers of the Deep Neural Network (DNN) and CrossNet in DCN V2. To provide a comprehensive understanding of the optimization dynamics, we report the dynamics of the 90th percentile and average values of the gradient norms during the training process. These results are depicted in Fig. 3. Remarkably, we observe that the Combined-Pair method consistently achieves higher values on both metrics compared to the BCE method. This difference persists throughout the entire training process. This finding further validates that the Combined-Pair method effectively alleviates the issue of gradient vanishing in the learnable parameters.

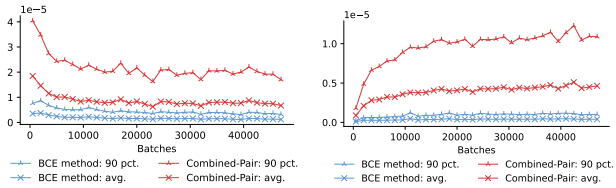


Figure 3: Gradient norm dynamics of DNN (left) and CrossNet (right) in DCN V2 through the training process. pct and avg. are shorts for percentile and average.

4 EXPERIMENTS

In this section, we conduct empirical experiments to answer the following research questions (RQs): If the sparse positive rate is the main cause of the gradient vanishing of negative samples, how would models perform under various sparsity of positives? What's the trade-off between classification loss and ranking loss (RQ2)? How do the other methods with classification-ranking combination losses perform (RQ3)? Last, can our perspective be extended to methods beyond the classification-ranking combination loss (RQ4)?

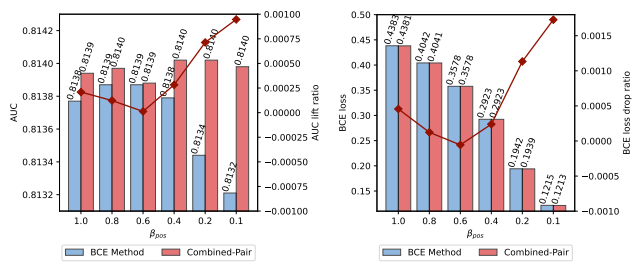
4.1 Dataset and Experimental Setting

We conducted experiments on the public Criteo dataset[1] [16], a widely used advertising recommendation dataset. It consists of 13 numerical features and 26 categorical features. Specifically, we utilized the criteo_x1 version, where the training, validation, and testing data are divided in a 7:2:1 ratio. We employed DCN V2 [40] as the backbone model and utilized the FuxiCTR [47] implementation[2] with the same settings as BARS [46][3]. We evaluated the performance regarding two metrics: BCE loss (*i.e.*, binary-cross entropy loss) to measure the classification ability and AUC to measure the ranking ability.

Specifically, we create artificial datasets based on the Criteo dataset and control the sparsity degree of positives by assigning a weight $0 < \beta_{\text{pos}} \leq 1$ for all its positive samples. This weight is used to down-weight positive samples in the training loss:

$$\mathcal{L}_{BCE} = -\frac{1}{N} \sum_{i=1}^{N} [\beta_{\text{pos}} \cdot y_i \log(\sigma(z_i)) + (1-y_i) \log(1-\sigma(z_i))]. \tag{15}$$

[1] https://github.com/reczoo/Datasets/tree/main/Criteo/Criteo_x1
[2] https://github.com/reczoo/FuxiCTR/tree/main/model_zoo/DCNv2
[3] https://github.com/reczoo/BARS/tree/main/ranking/ctr/DCNv2/DCNv2_criteo_x1



(a) AUC (b) BCE loss

Figure 4: Performance evaluation of Combined-Pair and the BCE method under varying positive sparsity rates. Here β_{pos} denotes the weights of positive samples.

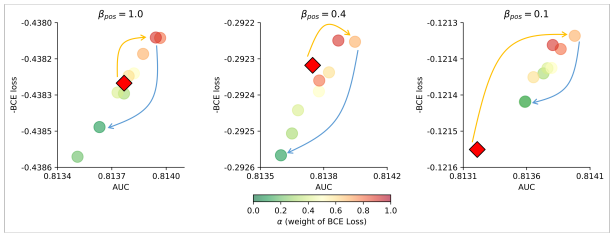


Figure 5: AUC and negative BCE loss of Combined-Pair and BCE method in Criteo test set. α and $1-\alpha$ are the weights of BCE loss and RankNet loss within Combined-Pair, respectively. The diamond in red represents the BCE method, *i.e.*, $\alpha = 1$. We shows results with α ranging between $[0.1, 1]$

For example, by setting $\beta_{\text{pos}} = 0.1$ for all positive samples, we generate a new dataset with sparsity degree of positive as $\frac{25.6\% \times 0.1}{25.6\% \times 0.1 + 1 - 25.6\%} = 3.3\%$.

4.2 RQ1: Performance Evaluation with various Positive Sparsity Rates

To validate our hypothesis, we generated multiple artificial datasets with varying degrees of sparsity in positive feedback by adjusting the values of β_{pos}. We then evaluated the performance of both the BCE method and the Combined-Pair method on these datasets. If our hypothesis holds true, datasets with sparser positive feedback should be more susceptible to the issue of gradient vanishing. Consequently, we would expect the BCE method to exhibit poorer performance on these datasets, while the Combined-Pair method should demonstrate significantly better performance, resulting in a larger performance gap compared to the BCE method.

In particular, we created artificial datasets with β_{pos} equals to 0.8, 0.6, 0.4, 0.2, 0.1, respectively. A smaller β_{pos} indicates sparser positive feedback. As shown in Fig. 4, we observe that the Combined-Pair always gets better AUC and BCE loss at all β_{pos}. Especially, from $\beta_{\text{pos}} = 0.6$ to $\beta_{\text{pos}} = 0.1$, *the sparser positive rates (i.e., smaller β_{pos}), the larger AUC lift (from 0.020% to 0.095%) and BCE loss drop (from 0.045% to 0.168%) between the Combined-Pair and the BCE method.* This validates our hypothesis that Combined-Pair achieves a larger performance lift than the BCE method when sparser positives reach a sparsity threshold (here $\beta_{\text{pos}} = 0.6$).

4.3 RQ2: Trade-off between Classification and Ranking Loss

Our aim to examine the trade-off between the classification and ranking loss components within the Combined-Pair loss. To achieve this, we vary the loss weight parameter, denoted as α, in the Combined-Pair loss from 1.0 to 0.1. We evaluate the negative BCE loss and the Area Under the Curve (AUC) as performance metrics, as depicted in Fig. 5. Based on our observations, we note the following:

First, starting with $\alpha = 1.0$, which corresponds to the Binary Cross-Entropy (BCE) method (represented by the red diamond), we observe that decreasing α, *i.e.*, reducing the weight of the BCE loss while increasing the weight for the ranking loss, leads to simultaneous improvements in both the negative BCE loss and AUC. This trend is represented by the orange arrow in the figure. For instance, with $\beta_{\text{pos}} = 0.1$, reducing α from 1.0 to 0.7 results in a decrease in negative BCE loss from 0.1215 to 0.1213, and an increase in AUC from 0.8132 to 0.8139. This suggests that *the classification and ranking abilities can be improved monotonically by decreasing α up to a certain threshold.*

Second, we observe that both metrics deteriorate when α is further increased. In other words, as the ranking loss becomes more dominant in the combination loss beyond a certain threshold, both the classification and ranking abilities deteriorate monotonically (as shown by the blue arrow).

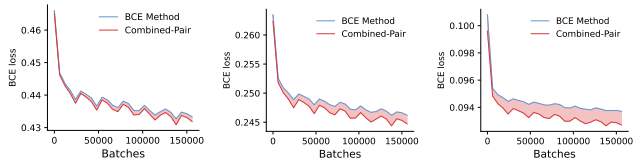
However, when the positive feedback is very sparse (*e.g.*, $\beta_{\text{pos}} = 0.1$), even with a very large weight for the ranking loss (*e.g.*, $1 - \alpha = 0.9$), the model's performance remains superior to the BCE method.

4.4 RQ3: Evaluation of Different Ranking Losses

In this section, we expand our analysis to include other classification-ranking combination methods beyond Combined-Pair. We aim to examine whether these methods possess properties similar to the Combined-Pair, thereby enhancing the applicability of our theory to a broader range of methods.

While Combined-Pair integrates BCE loss with pairwise ranking loss, other methods combine BCE loss with listwise ranking loss. For example, Combined-List [42] employs BCE loss in conjunction with the original ListNet loss [4]. RCR [1] combines BCE loss with ListCE loss, a variant of ListNet loss designed to align its minima with that of BCE loss. JRC [34] decoupled the logit into click and non-click logits and proposed a corresponding combination method with listwise-like loss.

These combination methods also incorporate ranking loss with BCE loss. So, can they also alleviate gradient vanishing regarding negative samples? We monitor the gradient norm dynamics of negative samples for these methods on the Criteo dataset in the first training epoch. Through the analysis shown in Fig. 2, we found that their gradient norms are relatively improved compared to the BCE method to varying degrees across different forms, indicating that gradient vanishing is also alleviated in these methods. Among



(a) $\beta_{\text{pos}} = 1.0$ (b) $\beta_{\text{pos}} = 0.4$ (c) $\beta_{\text{pos}} = 0.1$

Figure 6: Training error (*i.e.*, BCE loss) of the BCE method and Combined-Pair with various β_{pos} on Criteo.

them, JRC decouples the logit into click and non-click logit, and the gradient vanishing is mainly mitigated on the non-click logit.

Table 1: The performance of combining different ranking losses under sparse positive feedback situations. The ↑ and ↓ represent increasing in AUC and decreasing in BCE loss compared to the BCE method, respectively.

Metric	BCE	BCE+Pairwise	BCE+Listwise		
		Combined-Pair	JRC	Combined-List	RCR
AUC↑	0.81321	**0.81398**$^{\uparrow}$	0.81355$^{\uparrow}$	0.81351$^{\uparrow}$	0.81349$^{\uparrow}$
BCE loss↓	0.12152	**0.12131**$^{\downarrow}$	0.12146$^{\downarrow}$	0.12152	0.12141$^{\downarrow}$

We then compare the performance of JRC, Combined-List, and RCR methods on the Criteo dataset with $\beta_{\text{pos}} = 0.1$. We found they all show improved ranking and classification performance, as shown in Tab. 1. Overall, it can be concluded that these combination methods can also achieve performance improvement by introducing ranking loss to alleviate gradient vanishing of negative samples.

4.5 RQ4: Beyond Ranking Loss

We're curious whether approaches beyond ranking loss can also alleviate the gradient vanishing of negative samples and hence improve classification performance. We first study the following two methods: Focal Loss [20] and Negative Sampling [24]. Then, we designed a novel approach called Combined-Contrastive to validate our perspective further.

4.5.1 Focal Loss. It assigns higher weights to poorly classified samples [20], *i.e.*, negative samples suffering from gradient vanishing in our scenario. Specifically, Focal Loss introduces a weight with hyper-parameter γ to control the weight of samples:

$$\mathcal{L}_{\text{Focal}} = -\frac{1}{N}\sum_{i=1}^{N}[y_i(1-\hat{p}_i)^{\gamma}\log(\hat{p}_i) + (1-y_i)\hat{p}_i^{\gamma}\log(1-\hat{p}_i))], \tag{16}$$

where γ controls the relative weight. For those negative samples that may suffer from gradient vanishing when only using the BCE loss, their prediction score $\hat{p}_i$ should be wrongly high, making them have a higher weights $\hat{p}_i^{\gamma}$ than those negatives that are well-classified, *i.e.*, with a low score. The larger γ, the higher the relative weights to those poorly-classified samples.

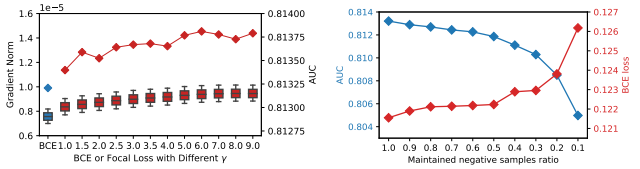


Figure 7: Evaluation of Focal loss and negative sampling. Left: Gradient norm (boxplot) and AUC (solid line) along with different γ for Focal Loss. Right: AUC (blue) and BCE loss (red) after isotonic regression with increasingly aggressive Negative Sampling.

We then conducted a set of experiments by comparing gradients and performance for Focal Loss with different γ. As shown in Fig. 7 (left), similar to Combined-Pair, Focal Loss also gets higher gradients on negative samples than the BCE loss. The larger the γ, the more weights on the poorly classified samples, and the larger the performance lift than the BCE method. These results validate that Focal Loss can also mitigate the gradient vanishing of negative samples and, hence, improve classification performance.

Please note that the original $\hat{p_i}^{\gamma}$ is always less than 1.0, and the gradients of negative samples in Focal loss are constantly smaller than BCE loss. For a fair comparison, we introduce a slight modification to the original formulation by replacing $\hat{p_i}^{\gamma}$ with $[\hat{p_i}^{\gamma} + (1 - \sum_{k\in N_-}\hat{p_k}^{\gamma}/N_-)]$. This adjustment normalizes the average weights of the negative samples to 1.0, aligning it with the BCE loss. Formally, the modified version is defined as:

$$\mathcal{L}_{\text{Focal'}} = -\frac{1}{N}\sum_{i=1}^{N}[y_i(1-\hat{p}_i)^{\gamma}\log(\hat{p}_i) + [\hat{p_i}^{\gamma} + (1 - \sum_{k=1}^{N_-}\hat{p_k}^{\gamma}/N_-)](1-y_i)\log(1-\hat{p}_i))], \tag{17}$$

4.5.2 Negative Sampling. Another trial is to reduce the proportion of negative samples through negative sampling [24], thereby increasing the estimated CTR. This may consequently increase the gradient of negative samples because their gradients are proportional to the estimated CTR (Eq. 10). However, downsampling negative samples may lead to information degradation, especially in datasets like Criteo, which has already undergone negative sampling. We analyze the AUC and BCE loss of negative sampling, and for a fair comparison, we report the calibrated results by isotonic regression [5, 17]. We observe that as negative sampling becomes more aggressive, both the model's ranking and classification capabilities deteriorate in Fig. 7 (right), which indicates that negative sampling fails to improve performance.

4.5.3 Combined-Contrastive: A New Method. Besides validating our perspective on existing methods, such as Combined-Pair and Focal Loss, we'd like to derive new methods based on our perspective. We speculate that introducing an auxiliary loss that considers the label information, *i.e.*, an *auxiliary supervised loss*, may provide larger gradients than the Binary Cross-Entropy (BCE) loss itself and hence mitigate the gradient vanishing issue. To this end, inspired by

the supervised contrastive learning [14], we have devised a novel approach termed Combined-Contrastive Loss, which integrates the BCE loss with Contrastive Loss. Specifically, the Contrastive Loss is employed to encourage embeddings belonging to the same class to be closely grouped while ensuring distinct separation between embeddings from different classes. Formally,

$$\mathcal{L}^{\mathrm{CC}} = \alpha \mathcal{L}_{\mathrm{BCE}} + (1-\alpha)\mathcal{L}_{\mathrm{Contr}}, \tag{18}$$

$$\mathcal{L}_{\mathrm{Contr}} = \frac{1}{|N|}\sum_{i=1}^{N}\frac{-1}{|P(i)|}\sum_{p\in P(i)}\log\frac{\exp(\mathbf{z}_i\mathbf{z}_p/\tau)}{\sum_{a\in A(i)}\exp(\mathbf{z}_i\mathbf{z}_a/\tau)}, \tag{19}$$

where $\mathcal{L}_{\mathrm{CC}}$ and $\mathcal{L}_{\mathrm{Contr}}$ represent the Combined-Contrastive Loss and the Contrastive Loss, respectively. N denotes the number of samples in the batch, $A(i)$ denotes the whole samples set except i-th sample itself, $P(i)$ denotes a sample subset that contains all samples in $A(i)$ that are with the same label as i-th sample, $\mathbf{z_i}$ denotes the embedding of the i-th sample. We set $\alpha = 0.9$ and $\tau = 0.4$.

Table 2: Performance and gradient norm of negative samples of BCE method and Combined-Contrastive. The training stage's average values on the first epoch are reported.

Stage	Metrics	BCE Method	Combined-Contrastive
Training	Gradient Norm	4.9×10^{-6}	$\mathbf{7.5\times10^{-6}}$
	BCE loss ↓	0.09667	**0.09428**$^{\downarrow}$
Testing	AUC↑	0.81321	**0.81340**$^{\uparrow}$
	BCE loss↓	0.12152	**0.12147**$^{\downarrow}$

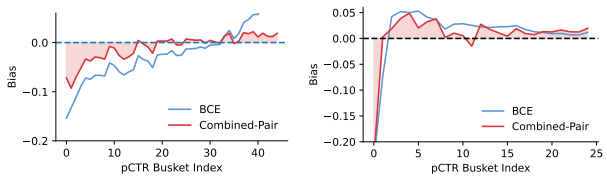
We conduct experiments on the same artificial Criteo dataset with $\beta_{\mathrm{pos}} = 0.1$ as mentioned in Sec. 4.1. As shown in Tab. 2, Combined-Contrastive gets a higher AUC and smaller BCE loss in the testing set, indicating better classification and ranking ability. In addition, similar to the Combined-Pair, it also gets lower training BCE loss and larger gradients than the BCE method. This verifies that by introducing the auxiliary contrastive loss, Combined-Contrastive can also mitigate the gradient vanishing issue and hence improve the classification ability.

5 STABILITY AND COMPATIBILITY

Recent studies[1, 42] have raised concerns regarding tailoring ranking loss into CTR prediction. Some studies [42] are concerned that ranking loss introduces *score drifting*, leading to *optimization instability*, while others [1] express concerns about *compatibility* issues between ranking loss and BCE loss. Both issues can also lead to negative effects on optimization. Hence, we aim to investigate whether classification-ranking loss, especially Combined-Pair, also has these two issues or not.

5.1 Stablility

Stability [42] refers to the phenomenon that the model scores may keep drifting during model training. Intuitively, singly employing ranking loss causes the model to focus solely on the relative orders between samples, neglecting the absolute prediction scores. This leads to *score drifting* [42], resulting in non-scale calibrated outcomes and exacerbating bias distribution. Hence, we wonder whether combining ranking loss with BCE loss in Combined-Pair may worsen the bias compared to using only BCE loss (*i.e.*, the BCE method). It is worth mentioning that we did not employ any post-processing calibration techniques here. Otherwise, the bias would not reflect its training stability.



(a) Offline Bias (b) Online Bias

Figure 8: The bias distribution over different pCTR buskets for both online and offline experiments.

We bucketize the samples with equal frequency of positive samples and plot the bias of corresponding samples within each bucket. As shown in Fig. 8, in both online and offline advertising, the BCE method severely underestimates the click-through rate (CTR) in the lower buckets, while Combined-Pair has a much smaller bias. In the higher buckets, the BCE method overestimates the CTR, and Combined-Pair also has a small bias in those buckets. The reason is probably that Combined-Pair mitigates the gradient vanishing issue, thus facilitating model optimization and leading to lower bias. In summary, Combined-Pair demonstrates *superior calibration ability* compared to the BCE method, avoiding the issue of score drifting and ensuring stability.

5.2 Compatibility

Recent research [1] argued that the pointwise and pairwise loss may not be compatible since they have different global minima. Thus, we are curious about the *compatibility* of the two losses within Combined-Pair: the BCE loss vs. the RankNet loss. We refrain from comparing their global minima [1], as differences in global minima among losses do not inherently indicate incompatibility. For instance, while *l1* or *l2* regularization losses do not share the same global minimum as most optimization objectives, they are still effectively utilized in tandem to mitigate overfitting.

Table 3: Gradient of positive and negative sample logit.

Samples	BCE Loss	RankNet Loss	Direction
Negative	$\frac{\hat{p}_j}{N} > 0$	$\frac{\sum_{i=1}^{N_+}\sigma(z_j^{(-)}-z_i^{(+)})}{N_+N_-} > 0$	Same
Positive	$-\frac{1-\hat{p}_i}{N} < 0$	$-\frac{\sum_{j=1}^{N_-}[1-\sigma(z_i^{(+)}-z_j^{(-)})]}{N_+N_-} < 0$	Same

Instead, we investigate the compatibility of these losses by analyzing their gradients, examining whether there is a conflict in gradients between them. As shown in Tab. 3, the BCE loss and RankNet loss gradients are always aligned in the same sign, leading to the same optimization directions by two objectives. Therefore,

the optimization directions of the combined-pair loss remain compatible without conflict.

6 ONLINE DEPLOYMENT

We conduct an empirical evaluation of Combined-Pair on the Click-Through Rate (CTR) prediction task across three distinct online advertising scenarios in Tencent: *WeChat Channels*, *WeChat Moments*, and the *Demand-Side Platform (DSP)*.

6.1 Deployment Details

The underlying model architecture utilizes Heterogeneous Experts with Multi-Embedding framework [27]. Specifically, our approach involves training multiple feature interaction experts, such as GwPFM [27] (a variant of FFM [13] and FwFM [25]), IPNN [29], DCN V2 [40], or FlatDNN, to capture diverse feature interactions for sparse ID features. Additionally, we learn multiple embedding tables for all features [10, 26, 35], with each table corresponding to one or several experts. For sequence features, we employ the TIN [45] to capture the semantic-temporal correlation.

Based on the above backbone architecture, we deployed Combined-Pair and conducted online A/B testing from early July 2023 to August 2023. We configured the Combined-Pair with $\alpha = 0.9$ and adopted streaming training. The CTR varies from 0.1% to 2.0% in different scenarios.

The ranking loss adds an extra computation cost of $O(N_+ \times N_-)$ per batch, where N_+ and N_- denote the number of positive and negative samples within a batch. Such additional computation cost is negligible compared with the complexity of the backbone model. Besides, the ranking loss only influences the training time and doesn't affect the inference time. During the online A/B test, the inference time and QPS are consistent with the baseline.

6.2 Overall Performance

We examined the gradient distribution of negative samples and normalized their frequency, as shown in Fig. 9. The analysis reveals that the gradient distribution of negative samples for the Combined-Pair method is significantly right-skewed compared to the BCE method, indicating that the Combined-Pair method obtains larger gradients.

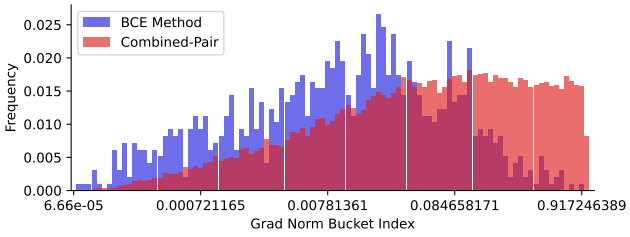


Figure 9: Distribution of gradient norms for negative samples in BCE method and Combined-Pair in online experiments.

Upon further examination of the model's online performance metrics, we observed that the Combined-Pair method significantly improves all business metrics compared to the BCE method, as shown in Tab. 4. For instance, during one month of A/B testing with 20% traffic, the Combined-Pair method achieves a cost lift of 0.59% and a Gross Merchandise Value (GMV) lift of 0.70% over the BCE method in the WeChat Moments scenario.

Table 4: Online A/B Testing Results.

Ad Scenario	CTR	GMV	Cost
WeChat Channels	+0.91%	+1.08%	+0.29%
WeChat Moment	+0.16%	+0.70%	+0.59%
DSP	-0.04%	+0.55%	+0.15%

Table 5: Online A/B Testing Results for New Ads.

Launch Date	GMV	Cost
T	+1.04%	+0.27%
T-1	+1.04%	+0.27%
T-2	+0.83%	+0.47%
T-3	+0.81%	+0.17%
Total	+1.26%	+0.34%

In addition, we also examined the BCE loss for online deployment. Our observations indicate that the reduction in BCE loss for the Combined-Pair method ranges from 0.01% to 0.1% over a span of 7 days compared to the BCE method during the A/B testing period.

6.3 New Ads Performance

Given that new ads have only a few training samples and are more prone to optimization difficulties, we specifically focused on the performance of the Combined-Pair method in the WeChat Channels scenario. The results, as shown in Tab. 5, demonstrate a significant performance improvement achieved by the Combined-Pair method. Specifically, it achieves a GMV lift of 1.26% and a cost lift of 0.34%, which are statistically significant, as confirmed by the t-test.

7 CONCLUSION

In this paper, we have identified a challenge associated with using only binary-cross entropy loss for Click-Through Rate (CTR) prediction when positive feedback is sparse. Specifically, we have observed the issue of gradient vanishing for negative samples in such scenarios. To address this challenge, we propose a novel perspective by introducing an auxiliary ranking loss. We explain that the inclusion of this additional ranking loss leads to the generation of larger gradients for negative samples, effectively mitigating the problem of gradient vanishing. Through comprehensive experiments and analysis, we have provided strong evidence to support our perspective.

ACKNOWLEDGMENTS

We want to express our sincere gratitude to the following individuals for their invaluable contributions to this research regarding the analysis, online implementation, and experiments: Ming Yue, Kaixin Li, Zhixiang Feng, Xian Hu, Yaqian Zhang, Shifeng Wen, Jiancheng Wang, Yiding Deng, Zeen Xu, Xiaochen Wang, Chen Cai, GenBao Chen, Chaonan Guo, Junjie Zhai.

REFERENCES

[1] Aijun Bai, Rolf Jagerman, Zhen Qin, Le Yan, Pratyush Kar, Bing-Rong Lin, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2023. Regression Compatible Listwise Objectives for Calibrated Ranking with Binary Relevance. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 4502–4508.

[2] Christopher Burges, Robert Ragno, and Quoc Le. 2006. Learning to rank with nonsmooth cost functions. *Advances in neural information processing systems* 19 (2006).

[3] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. 2005. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*. 89–96.

[4] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*. 129–136.

[5] Nilotpal Chakravarti. 1989. Isotonic median regression: a linear programming approach. *Mathematics of operations research* 14, 2 (1989), 303–308.

[6] Olivier Chapelle, Eren Manavoglu, and Romer Rosales. 2014. Simple and scalable response prediction for display advertising. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 4 (2014), 1–34.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.

[8] Ian J Goodfellow, Oriol Vinyals, and Andrew M Saxe. 2014. Qualitatively characterizing neural network optimization problems. *arXiv preprint arXiv:1412.6544* (2014).

[9] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *International Joint Conference on Artificial Intelligence (IJCAI)*. 1725–1731.

[10] Xingzhuo Guo, Junwei Pan, Ximei Wang, Baixu Chen, Jie Jiang, and Mingsheng Long. 2024. On the Embedding Collapse when Scaling up Recommendation Models. *International Conference on Machine Learning (ICML)* (2024).

[11] Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, et al. 2014. Practical lessons from predicting clicks on ads at facebook. In *International Workshop on Data Mining for Online Advertising (ADKDD)*. 1–9.

[12] Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. 133–142.

[13] Yuchin Juan, Damien Lefortier, and Olivier Chapelle. 2017. Field-aware factorization machines in a real-world online advertising system. In *International Conference on World Wide Web (WWW)*. 680–688.

[14] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in neural information processing systems* 33 (2020), 18661–18673.

[15] Jen-Wei Kuo, Pu-Jen Cheng, and Hsin-Min Wang. 2009. Learning to rank from bayesian decision inference. In *Proceedings of the 18th ACM conference on Information and knowledge management*. 827–836.

[16] Criteo Labs. 2014. *Display Advertising Challenge*. https://www.kaggle.com/c/criteo-display-ad-challenge

[17] Jan de Leeuw, Kurt Hornik, and Patrick Mair. 2009. Isotone optimization in R: pool-adjacent-violators algorithm (PAVA) and active set methods. (2009).

[18] Cheng Li, Yue Lu, Qiaozhu Mei, Dong Wang, and Sandeep Pandey. 2015. Click-through prediction for advertising in twitter timeline. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1959–1968.

[19] Jianxun Lian, Xiaohuan Zhou, Fuzheng Zhang, Zhongxia Chen, Xing Xie, and Guangzhong Sun. 2018. xDeepFM: Combining explicit and implicit feature interactions for recommender systems. In *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*. 1754–1763.

[20] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.

[21] Tie-Yan Liu et al. 2009. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval* 3, 3 (2009), 225–331.

[22] Kelong Mao, Jieming Zhu, Liangcai Su, Guohao Cai, Yuru Li, and Zhenhua Dong. 2023. FinalMLP: An Enhanced Two-Stream MLP Model for CTR Prediction. *arXiv preprint arXiv:2304.00902* (2023).

[23] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. 2013. Ad click prediction: a view from the trenches. In *ACM SIGKDD International conference on Knowledge Discovery & Data Mining (KDD)*. 1222–1230.

[24] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems* 26 (2013).

[25] Junwei Pan, Jian Xu, Alfonso Lobos Ruiz, Wenliang Zhao, Shengjun Pan, Yu Sun, and Quan Lu. 2018. Field-weighted factorization machines for click-through rate prediction in display advertising. In *Proceedings of the 2018 World Wide Web Conference*. 1349–1357.

[26] Junwei Pan, Jian Xu, Alfonso Lobos Ruiz, Wenliang Zhao, Shengjun Pan, Yu Sun, and Quan Lu. 2018. Field-weighted factorization machines for click-through rate prediction in display advertising. In *World Wide Web Conference (WWW)*. 1349–1357.

[27] Junwei Pan, Wei Xue, Ximei Wang, Haibin Yu, Xun Liu, Shijie Quan, Xueming Qiu, Dapeng Liu, Lei Xiao, and Jie Jiang. 2024. Ads Recommendation in a Collapsed and Entangled World. *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)* (2024).

[28] Changhua Pei, Yi Zhang, Yongfeng Zhang, Fei Sun, Xiao Lin, Hanxiao Sun, Jian Wu, Peng Jiang, Junfeng Ge, Wenwu Ou, et al. 2019. Personalized re-ranking for recommendation. In *Proceedings of the 13th ACM conference on recommender systems*. 3–11.

[29] Yanru Qu, Han Cai, Kan Ren, Weinan Zhang, Yong Yu, Ying Wen, and Jun Wang. 2016. Product-based neural networks for user response prediction. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 1149–1154.

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

[31] Steffen Rendle. 2010. Factorization machines. In *IEEE International Conference on Data Mining (ICDM)*. 995–1000.

[32] David Sculley. 2010. Combined regression and ranking. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 979–988.

[33] Lili Shan, Lei Lin, and Chengjie Sun. 2018. Combined regression and tripletwise learning for conversion rate prediction in real-time bidding advertising. In *The 41st international ACM SIGIR conference on research & development in information retrieval*. 115–123.

[34] Xiang-Rong Sheng, Jingyue Gao, Yueyao Cheng, Siran Yang, Shuguang Han, Hongbo Deng, Yuning Jiang, Jian Xu, and Bo Zheng. 2023. Joint Optimization of Ranking and Calibration with Contextualized Hybrid Model. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4813–4822.

[35] Liangcai Su, Junwei Pan, Ximei Wang, Xi Xiao, Shijie Quan, Xihua Chen, and Jie Jiang. 2024. STEM: Unleashing the Power of Embeddings for Multi-task Recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 9002–9010.

[36] Yang Sun, Junwei Pan, Alex Zhang, and Aaron Flores. 2021. FM2: field-matrixed factorization machines for recommender systems. In *Proceedings of the Web Conference 2021*. 2828–2837.

[37] Robin Swezey, Aditya Grover, Bruno Charron, and Stefano Ermon. 2021. Pirank: Scalable learning to rank via differentiable sorting. *Advances in Neural Information Processing Systems* 34 (2021), 21644–21654.

[38] Yukihiro Tagami, Shingo Ono, Koji Yamamoto, Koji Tsukamoto, and Akira Tajima. 2013. Ctr prediction for contextual advertising: Learning-to-rank approach. In *Proceedings of the Seventh International Workshop on Data Mining for Online Advertising*. 1–8.

[39] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *International Workshop on Data Mining for Online Advertising (ADKDD)*. 1–7.

[40] Ruoxi Wang, Rakesh Shivanna, Derek Cheng, Sagar Jain, Dong Lin, Lichan Hong, and Ed Chi. 2021. Dcn v2: Improved deep & cross network and practical lessons for web-scale learning to rank systems. In *Proceedings of the web conference 2021*. 1785–1797.

[41] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. 2008. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*. 1192–1199.

[42] Le Yan, Zhen Qin, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2022. Scale calibration of deep ranking models. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4300–4309.

[43] Yuguang Yue, Yuanpu Xie, Huasen Wu, Haofeng Jia, Shaodan Zhai, Wenzhe Shi, and Jonathan J Hunt. 2022. Learning to Rank For Push Notifications Using Pairwise Expected Regret. *arXiv preprint arXiv:2201.07681* (2022).

[44] Zhaohui Zheng, Keke Chen, Gordon Sun, and Hongyuan Zha. 2007. A regression framework for learning ranking functions using relative relevance judgments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 287–294.

[45] Haolin Zhou, Junwei Pan, Xinyi Zhou, Xihua Chen, Jie Jiang, Xiaofeng Gao, and Guihai Chen. 2024. Temporal Interest Network for User Response Prediction. In *Companion Proceedings of the ACM on Web Conference 2024*. 413–422.

[46] Jieming Zhu, Quanyu Dai, Liangcai Su, Rong Ma, Jinyang Liu, Guohao Cai, Xi Xiao, and Rui Zhang. 2022. BARS: towards open benchmarking for recommender systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2912–2923.

[47] Jieming Zhu, Jinyang Liu, Shuai Yang, Qi Zhang, and Xiuqiang He. 2021. Open benchmarking for click-through rate prediction. In *Proceedings of the 30th ACM international conference on information & knowledge management*. 2759–2769.