A Pareto-Efficient Algorithm for Multiple Objective Optimization in E-Commerce Recommendation

Xiao Lin[1]*, Hongjie Chen[1]*, Changhua Pei[1]
Fei Sun[1], Xuanji Xiao[1], Hanxiao Sun[1], Yongfeng Zhang[2], Peng Jiang[3], Wenwu Ou[1]
[1] Alibaba Group [2] Rutgers University [3] Kwai Inc.
[1] {hc.lx, hongjie.chen, changhua.pch, ofey.sf, xuanji.xxj, hansel.shx, santong.oww}@alibaba-inc.com
[2] yongfeng.zhang@rutgers.edu [3] jiangpeng@kuaishou.com

ABSTRACT

Recommendation with multiple objectives is an important but difficult problem, where the coherent difficulty lies in the possible conflicts between objectives. In this case, multi-objective optimization is expected to be Pareto efficient, where no single objective can be further improved without hurting the others. However existing approaches to Pareto efficient multi-objective recommendation still lack good theoretical guarantees.

In this paper, we propose a general framework for generating Pareto efficient recommendations. Assuming that there are formal differentiable formulations for the objectives, we coordinate these objectives with a weighted aggregation. Then we propose a condition ensuring Pareto efficiency theoretically and a two-step Pareto efficient optimization algorithm. Meanwhile the algorithm can be easily adapted for Pareto Frontier generation and fair recommendation selection. We specifically apply the proposed framework on E-Commerce recommendation to optimize GMV and CTR simultaneously. Extensive online and offline experiments are conducted on the real-world E-Commerce recommender system and the results validate the Pareto efficiency of the framework.

To the best of our knowledge, this work is among the first to provide a Pareto efficient framework for multi-objective recommendation with theoretical guarantees. Moreover, the framework can be applied to any other objectives with differentiable formulations and any model with gradients, which shows its strong scalability.

ACM Reference Format:
Xiao Lin[1]*, Hongjie Chen[1]*, Changhua Pei[1] and Fei Sun[1], Xuanji Xiao[1], Hanxiao Sun[1], Yongfeng Zhang[2], Peng Jiang[3], Wenwu Ou[1]. 2019. A Pareto-Efficient Algorithm for Multiple Objective Optimization in E-Commerce Recommendation. In *Thirteenth ACM Conference on Recommender Systems (RecSys '19), September 16–20, 2019, Copenhagen, Denmark.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3298689.3346998

1 INTRODUCTION

Recommender systems are emerging as a crucial role in online services and platforms, which prevent users from information overload. The recommendation algorithms (for example Learning To Rank) generate personalized rankings of items and the top-ranked items are recommended to users. Usually, the algorithms need very careful designs to fulfill multiple objectives. However, it is difficult to optimize multiple objectives simultaneously, where the core difficulty lies in the conflicts between different objectives. In E-Commerce recommendation, CTR (Click Through Rate) and GMV (Gross Merchandise Volume) are two important objectives that are not entirely consistent. To validate this inconsistency, we collect one-week online data from a real-world E-Commerce platform and plot the trends of GMV when CTR . According to the trends reflected in Fig. 1, CTR is not entirely consistent with GMV , and a CTR-optimal or GMV-optimal recommendation can be rather sub-optimal or even bad in terms of the other objective.

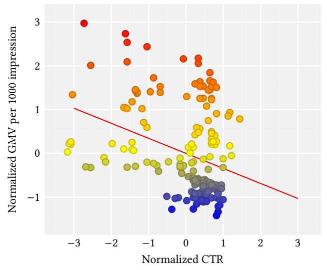

Figure 1: The trade-off between CTR and GMV. The Pearson Correlation Coefficient is -0.343086, with $p < 0.01$.

Therefore, a solution is considered as optimal for two objectives in the sense that no objective can be further improved without hurting the other one. This optimality is widely acknowledged in multiple objective optimization and named as Pareto efficiency or Pareto optimality. In the context of Pareto efficiency, solution A is considered to dominate solution B only when A outperforms B on all the objectives. And the aim of Pareto efficiency is to find solutions that are not dominated by any others.

Existing approaches for Pareto optimization can be categorized into two categories: heuristic search and scalarization. Evolutionary algorithms are popular choices in heuristic search approaches. However, heuristic search can not guarantee Pareto efficiency, it only ensures the resulting solutions are not dominated by each other (but still can be dominated by the Pareto efficient solutions) [45]. Unlike heuristic search, scalarization transforms multiple objectives into a single one with a weighted sum of all the objective functions. With proper scalarization, the Pareto efficient solutions can be achieved by optimizing the reformulated objective function. However, the scalarization weights of objective functions are usually determined

RecSys '19, September 16–20, 2019, Copenhagen, Denmark

ACM ISBN 978-1-4503-6243-6/19/09…$15.00
https://doi.org/10.1145/3298689.3346998

manually and Pareto efficiency is still not guaranteed. To summarize, it is very difficult for existing evolutionary algorithms and scalarization algorithms to find Pareto efficient solutions with a guarantee. Recently, it is pointed out that the Karush-Kuhn-Tucker (KKT) conditions can be used to guide the scalarization [11]. We build our algorithm upon the KKT conditions and propose a novel algorithmic framework that generates the scalarization weights with theoretical guarantees.

Specifically, we propose a Pareto-Efficient algorithmic framework "PE-LTR" that optimizes multiple objectives with an LTR procedure. Given the candidate items generated for each user, PE-LTR ranks the candidates so that the ranking is Pareto efficient with respect to multiple objectives. Assuming that there exist differentiable formulations for each objective correspondingly, we adopt the scalarization technique to coordinate different objectives into a single objective function. As stated before, the scalarization technique can not guarantee Pareto efficiency unless the weights are carefully chosen. Therefore, we first propose a condition for the scalarization weights that ensures the solution is Pareto efficient. The condition is equivalent to a constrained optimization problem, and we propose an algorithm that solves the problem in two steps. First we simplify the problem by relaxing the constraints so that an analytic solution is achieved; then we get the feasible solution by conducting a projection procedure. With PE-LTR as the cornerstone, we provide methods to generate the Pareto Frontier and a specific recommendation, depending on the needs of service providers. To generate the Pareto Frontier, one can run PE-LTR by evenly set the bounds of the objective scalarization weights. To generate a specific recommendation, one can either run PE-LTR once with proper bounds or generate the Pareto Frontier first and choose a "fair" solution with specific fairness metric.

In this paper we apply this framework to optimize two important objectives for E-Commerce recommendation, i.e. GMV and CTR. For E-Commerce platforms, the primary objective is to improve the GMV, but too much sacrifice of CTR may cause a severe decrease of daily active users (DAU) in the long term. Therefore we aim to find Pareto efficient solutions with respect to both objectives. We propose two differentiable formulations for GMV and CTR respectively and apply the PE-LTR framework for generating Pareto-optimal solutions. We conduct extensive experiments on a real-world E-Commerce recommender system and compare the results with state-of-the-art approaches. The online and offline experimental results both indicate that our solution outperforms other baselines significantly and the solutions are nearly Pareto efficient.

The contributions of this work are:

- We propose a general Pareto efficient algorithmic framework (PE-LTR) for multi-objective recommendation. The framework is both model and objective agnostic, which shows its great scalability.
- We propose a two-step algorithm which theoretically guarantees the Pareto efficiency. Despite the algorithm is built upon scalarization technique, it differs from other scalarization approaches with its theoretical guarantee and its automatic learning of scalarization weights rather than manually assignment.
- With PE-LTR as the cornerstone, we present how to generate the Pareto Frontier and a specific recommendation. Specifically, we propose to select a fair recommendation from the Pareto Frontier with proper fairness metrics.
- We use E-Commerce recommendation as a specification of PE-LTR, and conduct extensive online and offline experiments on a real-world recommender system. The results indicate that our algorithm outperforms other state-of-the-art approaches significantly and the solutions generated are Pareto efficient.
- We open-source a large-scale E-Commerce recommendation dataset EC-REC, which contains the real records of impressions, clicks and purchases. To the best of our knowledge, no public dataset includes all three labels and enough features, this dataset can be used for further studies.

2 RELATED WORK

In this section, we provide a detailed introduction to the related studies from the following aspects: recommendation with multiple objectives, E-Commerce recommendation and learning to rank.

2.1 Recommendation with Multiple Objectives

We look at the studies on multi-objective recommendation from two aspects, i.e. the objectives concerned and the approaches for multi-objective recommendations.

Despite the recommendation accuracy is the main concern, some studies argue that other characteristics such as the availability, profitability, or usefulness should be considered simultaneously [15, 22]. Some studies attempt to model the trade-offs between relevance and diversity in recommendation [14, 17, 41]. When multiple objectives are concerned, it is expected to get a Pareto efficient recommendation [27, 28]. Recently, it is pointed out that some multiple objectives are related to users [7, 16, 23, 29]. On one hand, different objectives are related to different user behaviors. For example, both clicks and hides are considered in LinkedIn feeds [33]. On the other hand, the objectives are related to different user statuses, for example different stakeholders [8, 23].

The approaches on recommendation with multiple objectives can be categorized into evolutionary algorithm [45] and scalarization [38]. The evolutionary algorithm has been used for long-tail recommendation [35], diversified recommendation [10], and novelty-aware recommendation [28]. And it has also been used for Pareto efficient hybridization [28] of multiple recommendation algorithms. Scalarization technique is also used for recommendation with multiple objectives [38]. However, existing studies mostly depend on manually assigned weights for scalarization, whose Pareto efficiency can not be guaranteed. Recently, the KKT conditions are used for guiding scalarization techniques [11, 32]. However, existing algorithms based on these conditions are limited to the unconstrained cases and can not fit the requirements in real-world scenarios.

2.2 E-Commerce Recommendation

E-Commerce recommendation is also a popular research topic. Some studies adopt economic theory models and Markov chains for recommendation [12, 19, 42, 43]. While some other studies focus on other aspects in E-Commerce recommendation [1, 30, 34, 40], such as feature learning and diversification. It is pointed out that a good practice in E-Commerce searching is learning to rank [18], which also coincides with the motivation of our framework. Usually there are multiple stages in E-Commerce recommendation, for example clicks and purchases. Therefore the learning-to-rank algorithms need to jointly optimize multiple stages [36]. Some studies focus on the post-click stage in searching and recommendation. For example, the bidding price and revenue are jointly considered with relevance [26, 44]. Recently, two studies focus on the connection between clicks and purchases in E-Commerce searching and advertising [21, 36]. As optimizing clicks and purchases are not entirely consistent, it is necessary to find a Pareto efficient trade-off between them, which is not considered in previous studies on purchase optimization [21, 36].

2.3 Learning to Rank

Learning To Rank (LTR) has been a popular research topic for quite a long time. The studies on LTR can be categorized into point-wise, pair-wise and list-wise approaches. The point-wise scheme [20] predicts the individual instance separately; the pair-wise scheme [4, 13] is approximated as a binary classification problem, which focuses on the relative order of a pair of instances; while the list-wise scheme [5, 6, 37, 39] directly optimizes the metric of a ranking list. Usually, list-wise LTR achieves superior performances than other schemes.

ranking methods have been proposed, such as RankNet [4], RankBoost [13], AdaRank [39], LambdaRank [5], ListNet [6] and LambdaMART [37]. Due to the similarity between searching and recommendation in ranking, LTR approaches are widely used in both scenarios. Recently, it is pointed out that LTR is a key component in E-Commerce searching [18], which is able to exploit multiple user feedback signals for relevance modeling, including clicks, add-to-cart ratios, and revenue.

According to the previous studies, LambdaMART is one of the best performing algorithms [36]. As focus of this paper is not about ranking model, we choose a simple point-wise ranking model for the proposed framework.

3 PROPOSED FRAMEWORK

In this section, we first provide a brief introduction to the concept of Pareto efficiency. Then we introduce the details of the proposed framework, i.e. Pareto-Efficient Learning-to-Rank (PE-LTR). Assuming that there are differentiable loss functions for multiple objectives correspondingly, we propose a condition that guarantees the Pareto efficiency of the solution. We show that the proposed condition is equivalent to a constrained Quadratic Programming problem. Then we propose a two-step algorithm to solve this problem. Moreover, we provide methods to generate both Pareto Frontier and specific single recommendation with PE-LTR.

3.1 Preliminary

First, we provide a brief introduction to Pareto efficiency and some related concepts. Pareto efficiency is an important concept in multiple objective optimization. Given a system which aims to minimize a series of objective functions $f_1, \dots, f_K$, Pareto efficiency is a state when it is impossible to improve one objective without hurting other objectives in terms of multi-objective optimization.

Definition 3.1. Denote the outcomes of two solutions as $\boldsymbol{s}_i = (f_1^i, \dots, f_K^i)$ and $\boldsymbol{s}_j = (f_1^j, \dots, f_K^j)$, $\boldsymbol{s}_i$ **dominates** $\boldsymbol{s}_j$ if and only if $f_1^i \le f_1^j, f_2^i \le f_2^j, \dots, f_K^i \le f_K^j$ (for minimization objectives).

The concept of Pareto efficiency is built upon the definition of domination:

Definition 3.2. A solution $\boldsymbol{s}_i = (f_1^i, \dots, f_K^i)$ is **Pareto efficient** if there is no other solution $\boldsymbol{s}_j = (f_1^j, \dots, f_K^j)$ that dominates $\boldsymbol{s}_i$.

Therefore, a solution that is not Pareto efficient can still be improved for at least one objective without hurting the others, and it is always expected to achieve Pareto efficient solutions in multi-objective optimization. It is worth mentioning that Pareto efficient solutions are not unique and the set of all such solutions is named as the "**Pareto Frontier**".

3.2 Pareto-Efficient Learning to Rank

To achieve a Pareto efficient solution, we propose a Learning-to-Rank scheme that optimizes multiple objectives with the scalarization technique. Assuming that there are K objectives in a given recommender system, a model $F(\boldsymbol{\theta})$ needs to optimize these objectives simultaneously, where $\boldsymbol{\theta}$ denotes the model parameters. Without loss of generality, we assume that there exist K differentiable loss functions $\mathcal{L}_i(\boldsymbol{\theta}),\ \forall i \in \{1, \dots, K\}$ for the K objectives correspondingly.

Given the formulations, optimizing i-th objective is equal to minimizing $\mathcal{L}_i$. However, optimizing these K objectives simultaneously is non-trivial, since the optimal solution to one objective is usually sub-optimal for another one. Therefore, we use the scalarization technique to merge multiple objectives into a single one. Specifically, we aggregate the loss functions $\mathcal{L}_i$ with ω_i, $\forall i \in \{1, \dots, K\}$:

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{K} \omega_i \mathcal{L}_i(\boldsymbol{\theta})$$

where $\sum_{i=1}^K \omega_i = 1$ and $\omega_i \ge 0$, $\forall i \in \{1, \dots, K\}$. In real-world scenarios, the objectives may have different priorities. In our case, we assume that the constraints added to the objectives are predefined boundary constraints, i.e. $\omega_i \ge c_i,\ \forall i \in \{1, \dots, K\}$, where c_i is a constant between 0 and 1, and $\sum_{i=1}^K c_i \le 1$.

Despite the single-objective formulation, it is not guaranteed that the solution to the problem is Pareto efficient, unless proper weights are assigned. Then we derive the condition on the scalarization weights that ensures the solution is Pareto efficient.

3.2.1 *The Pareto Efficient condition.*

To get the Pareto efficient solutions for multiple objectives, we attempt to minimize the aggregated loss function. Consider the KKT conditions (Karush-Kuhn-Tucher Conditions) [2] for the model parameters:

$$\sum_{i=1}^{K} \omega_i = 1,\ \ \exists \omega_i \ge c_i,\ \ i \in \{1, \dots, K\}\text{and } \sum_{i=1}^{K} \omega_i \nabla_{\boldsymbol{\theta}} \mathcal{L}_i(\boldsymbol{\theta}) = 0,$$

where $\nabla_{\boldsymbol{\theta}} \mathcal{L}_i(\boldsymbol{\theta})$ is the gradient of $\mathcal{L}_i$. Solutions that satisfy this condition are referred to as Pareto stationary. The condition can be

Table 1: Notations and Description

Notations	Description
$\boldsymbol{x}$	The feature vector
$y \in \{0, 1\}$	The click label
$z \in \{0, 1\}$	The purchase label
$\mathcal{L}_i(\boldsymbol{\theta})$	The loss function for i-th objective
$F(\boldsymbol{\theta})$	The LTR model
$\boldsymbol{\theta}$	The model parameters
η	The learning rate of $F(\boldsymbol{\theta})$
ω_i	The weight of i-th objective for scalarization
c_i	The boundary constraint for i-th objective
$price_j$	The price of the item in instance $\boldsymbol{x}_j$
price'_i	The price of the item at i-th rank
$\nabla_{\boldsymbol{\theta}}\mathcal{L}_i(\boldsymbol{\theta})$	The gradient of loss $\mathcal{L}_i(\boldsymbol{\theta})$ with respect of $\boldsymbol{\theta}$
$\boldsymbol{G}$	The stacking matrix of the gradients $\nabla\mathcal{L}_i(\boldsymbol{\theta})$
$\boldsymbol{e}$	The vector whose elements are all 1

Algorithm 1 Pareto Efficient LTR:

```
Input: The loss functions of multiple objectives correspondingly: $\mathcal{L}_i(\boldsymbol{\theta})$, $\forall i \in \{1, \ldots, K\}$; The scalarization weights initialized uniformly: $\omega_i = \frac{1}{K}$, $\forall i \in \{1, \ldots, K\}$; The bounds for the objectives: $c_i$, $\forall i \in \{1, \ldots, K\}$;
Output: The model parameters $\boldsymbol{\theta}$;
Get the single aggregated objective function: $\mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{K} \omega_i \mathcal{L}_i(\boldsymbol{\theta})$;
for each batch do
    Update the model $F(\boldsymbol{\theta})$ by optimizing $\mathcal{L}(\boldsymbol{\theta})$ with stochastic gradient descent: $\boldsymbol{\theta} = \boldsymbol{\theta} - \eta \frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$;
    Run Alg. 2 to update $\omega_1, \ldots, \omega_K$ = PECsolver($\boldsymbol{\theta}$);
    Aggregate the objectives: $\mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{K} \omega_i \mathcal{L}_i(\boldsymbol{\theta})$;
end for
```

transformed into the following optimization problem:

$$
\begin{aligned}
&\min . \left\| \sum_{i=1}^{K} \omega_i \nabla_{\boldsymbol{\theta}} \mathcal{L}_i(\boldsymbol{\theta}) \right\|_2^2 \\
&s.t. \sum_{i=1}^{K} \omega_i = 1, \ \ \omega_i \geq c_i, \ \ \forall i \in \{1, \ldots, K\}
\end{aligned}
\tag{1}
$$

It has been proven [32] that either the solution to this optimization problem is 0 so that the KKT conditions are satisfied or the solutions lead to gradient directions that minimizes all the loss functions. If the KKT conditions are satisfied, the solution is Pareto stationary and also Pareto efficient under realistic and mild conditions [32]. Based on this condition, we propose an algorithmic framework named PE-LTR, whose details are illustrated in Alg. 1.

The framework starts with uniform scalarization weights and then updates the model parameters and the scalarization weights alternatively. The core part of PE-LTR is the PECsolver, which generates scalarization weights by solving the condition in Problem (1). Note that the condition is a complex Quadratic Programming problem, we present the detailed process of PECsolver in Alg 2.

Algorithm 2 PECsolver:

```
Formulate the Pareto Efficient condition as Problem (1);
Solve the relaxed Quadratic Programming in Problem (3) with Theorem 3.3;
Get the feasible solution by solving Problem (4);
```

It is worth mentioning that the algorithmic framework does not rely on specific formulations of the loss functions or the model structures. Any model and formulation with gradients can be easily applied to the framework. Despite the algorithms runs with stochastic gradient descent in batches, the algorithm provides a theoretical guarantee of convergence as gradient descent [11].

3.2.2 The Algorithm for Quadratic Programming. Denote $\hat{\omega}_i$ as $\omega_i - c_i$, the Pareto efficient condition becomes:

$$
\begin{aligned}
&\min . \| \sum_{i=1}^{K} (\hat{\omega}_i + c_i) \nabla_{\boldsymbol{\theta}} \mathcal{L}_i(\boldsymbol{\theta}) \|_2^2 \\
&\text{s.t.} \sum_{i=1}^{K} \hat{\omega}_i = 1 - \sum_{i=1}^{K} c_i, \ \ \hat{\omega}_i \geq 0, \ \ \forall i \in \{1, \ldots, K\}
\end{aligned}
\tag{2}
$$

The Pareto-Efficient condition is equivalent to Problem 1, however, it is not a trivial task to solve this problem due to its quadratic programming form. Therefore, we propose a two-step algorithm as the Pareto efficient condition solver. The algorithm is illustrated in Alg. 2. We first relax the problem by only considering the equality constraints and solve the relaxed problem with an analytical solution. Then we introduce a projection procedure that generates a valid solution from the feasible set with all the constraints.

When all the other constraints are omitted except the equality constraints:

$$
\min . \| \sum_{i=1}^{K} (\hat{\omega}_i + c_i) \nabla_{\boldsymbol{\theta}} \mathcal{L}_i(\boldsymbol{\theta}) \|_2^2 \ \ \text{s.t.} \sum_{i=1}^{K} \hat{\omega}_i = 1 - \sum_{i=1}^{K} c_i \tag{3}
$$

The solution to the relaxed problem is given by Theorem 3.3.

THEOREM 3.3. *The solution to the equality constrained problem (3) is given by $\tilde{\boldsymbol{\omega}} = ((\boldsymbol{M}^T \boldsymbol{M})^{-1} \boldsymbol{M} \tilde{\boldsymbol{z}})[1 : K]$, where $\boldsymbol{G} \in \mathcal{R}^{K \times m}$ is the stacking matrix of $\nabla \mathcal{L}_i(\boldsymbol{\theta})$, $\boldsymbol{e} \in \mathcal{R}^K$ is the vector whose elements are all 1, $\boldsymbol{c} \in \mathcal{R}^K$ is the concatenated vector of c_i, $\tilde{\boldsymbol{z}} \in \mathcal{R}^{K+1}$ is the concatenated vector of $-\boldsymbol{G}\boldsymbol{G}^T \boldsymbol{c}$ and $1 - \sum_{i=1}^{K} c_i$, and $\boldsymbol{M}$ is $\begin{bmatrix} \boldsymbol{G}\boldsymbol{G}^T & \boldsymbol{e} \\ \boldsymbol{e}^T & 0 \end{bmatrix}$.*

The proof to this theorem is in the appendix.

However, the solution $\hat{\boldsymbol{\omega}}^*$ to problem 3 may not be valid since the non-negativity constraints are omitted. Therefore, we conduct the following projection step to get a valid solution:

$$
\min . \ \| \tilde{\boldsymbol{\omega}} - \hat{\boldsymbol{\omega}}^* \|_2^2 \ \text{s.t.} \ \sum_{i=1}^{K} \tilde{\omega}_i = 1, \ \ \tilde{\omega}_i \geq 0, \forall i \in \{1, \ldots, K\} \tag{4}
$$

This problem is exactly a non-negative least squares problem, and can be solved easily with the active set method [3]. Due to page limit, we omit the details of the algorithm to Problem 3 [1]. The complexity of Alg. 2 is mostly determined by the pseudo-inverse operation, which relates to the number of objectives. Usually the number of objectives is limited, therefore the running time of Alg. 2 is negligible and the online experiments have verified this

[1] We will include the pseudo codes in a longer version of the paper.

4 PARETO FRONTIER GENERATION AND SOLUTION SELECTION

Multiple objective optimization can either be used to find a certain Pareto solution, or be used to generate a set of solutions to construct the Pareto Frontier. In this section, we introduce the details of generating solutions with Alg. 1 for the two cases.

4.1 Pareto Frontier Generation

With Alg. 1, we can obtain a Pareto optimal solution given the bounds of different objectives. However, there are cases when a series of Pareto optimal solutions are expected, i.e. the Pareto Frontier. This is straight-forward for the algorithmic framework, we can set different values to the bounds of the objectives and perform Alg. 1 with different bounds respectively.

To get a Pareto Frontier, we conduct Alg. 1 for several times, and the solution generated with proper bound in each run yields a Pareto optimal solution. We choose the bounds properly so that the evenly distributed Pareto points make a good evenly distributed approximation of the Pareto Frontier.

4.2 Solution Selection

In cases when a single recommendation is expected, we need to select one certain Pareto optimal solution. When the priorities of different objectives are available, we can obtain a proper Pareto-efficient recommendation by setting a proper bound for the objectives and conduct a single run of Alg. 1.

When the priorities are not available, we can first generate the Pareto Frontier and select a solution that is “fair” for the objectives. There are several definitions of fairness in both economic theories and recommendation system context [38]. One of the most intuitive metrics is Least Misery, which focuses on the most “miserable” objective, in our case, a “Least Misery” recommendation is to minimize the highest loss function of the objectives:

$$\min \max\{\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_K\} \tag{5}$$

Another frequently used measure is fairness marginal utility, i.e., to select a solution where the cost of optimizing one objective is almost equal to the benefit of the other objectives:

$$min. \|\partial(\mathcal{L}_1 \cdot \mathcal{L}_2 \cdot \ldots \mathcal{L}_K)/\partial\boldsymbol{\theta}\|_2 \tag{6}$$

Given the generated Pareto Frontier, the solution with minimum values of Eqn. 5 or Eqn. 6 is selected as the final recommendation, depending on the choice of fairness.

5 SPECIFICATION ON E-COMMERCE RECOMMENDATION

Given the algorithmic framework of PE-LTR, we introduce the details of its specification on E-Commerce recommendation. Two of the most important objectives in E-Commerce recommendation are GMV and CTR. For E-Commerce platforms, GMV is usually the primary objective. However, CTR is a crucial metric for evaluating user experiences thus affects the scale of the platform in the long term. Therefore, we aim to find a recommendation that is Pareto-Optimal with respect to these two objectives.

Considering that in real-life environments, the LTR models take streaming data as input and updates its parameters in an online fashion. Therefore, the online LTR model usually follows the point-wise ranking scheme. We formulate the problem as a binary classification problem and two differentiable loss functions are designed for the two objectives correspondingly.

In E-Commerce recommender systems, user feedbacks can be roughly categorized into three types: the impressions, the clicks and the purchases. Denote the instances as $(\boldsymbol{x}_j,\ y_j,\ z_j),\ \forall j \in [1, \ldots, N]$, given a point-wise ranking model $F(\boldsymbol{\theta})$, we propose to optimize these two objectives, i.e. CTR and GMV. For CTR optimization, we aim to minimize:

$$\mathcal{L}_{CTR}(\boldsymbol{\theta}, \boldsymbol{x}, y, z) = -\frac{1}{N}\sum_{j=1}^{N} log(P(y_j|\boldsymbol{\theta}, \boldsymbol{x}_j))$$

For GMV optimization, we aim to minimize:

$$\mathcal{L}_{GMV}(\boldsymbol{\theta}, \boldsymbol{x}, y, z) = -\frac{1}{N}\sum_{j=1}^{N} h(price_j) \cdot log(P(z_j = 1|\boldsymbol{\theta}, \boldsymbol{x}_j))$$

$$= -\frac{1}{N}\sum_{j=1}^{N} h(price_j) \cdot (log(P(y_j = 1|\boldsymbol{\theta}, \boldsymbol{x}_j)) + log(P(z_j = 1|y_j = 1)))$$

$$= -\frac{1}{N}\sum_{j=1}^{N} h(price_j) \cdot (log(P(y_j = 1|\boldsymbol{\theta}, \boldsymbol{x}_j))) + g(price_j)h(price_j)$$

where $h(price_j)$ is a concave monotone non-decreasing function with respect to $price_j$, $price_j$ denotes the price of the item in $\boldsymbol{x}_j$. In our formulation, we choose $h(price_j) = \log(price_j)$. And we assume $P(z_j = 1|y_j = 1)$ is irrelevant of the model parameters $\boldsymbol{\theta}$. Therefore, given a model $F(\boldsymbol{\theta})$ and the formulation of $\mathcal{L}_{CTR}(\boldsymbol{\theta}, \boldsymbol{x}, y, z)$ and $\mathcal{L}_{GMV}(\boldsymbol{\theta}, \boldsymbol{x}, y, z)$, the E-Commerce recommendation problem becomes:

$$\min.\left\{\mathcal{L}_{CTR}(\boldsymbol{\theta}, \boldsymbol{x}, y, z),\ \mathcal{L}_{GMV}(\boldsymbol{\theta}, \boldsymbol{x}, y, z)\right\}\ \ s.t. \boldsymbol{\theta} \in \mathcal{R}^m$$

Note that the proposed framework does not rely on specific model structure or the formulations of the losses, it works as long as the model has gradients. Thus the formulations of CTR and GMV losses are not the focus of this paper, and more carefully designed formulations can be acommodated into this framework. Meanwhile we do not focus on a specific LTR model but use three different typical models for comparison, i.e. Logistic Regression (LR), Deep Neural Network (DNN) and Wide&Deep (WDL). The DNN model is a three-layer MLP and has a same structure with the deep component in the Wide&Deep model. For all the neural network components, we choose tanh as the activation function for each hidden layer while the final layer employs the linear function as the output. The comparison between three different models is illustrated in Fig 5 in the experiments.

6 EXPERIMENTS

In this section, we introduce the details of experiments which are designed to answer the following research questions:

- How does the framework perform in comparison with state-of-the-art CTR/GMV oriented approaches and multiple objective recommendation algorithms?
- How is the Pareto efficiency of the proposed framework in terms of the single recommendation and Pareto Frontier?

- How is the scalability of the proposed framework in terms of model selection?

To answer these research questions, we conduct extensive experiments on real-world datasets on a popular E-Commerce website, including online and offline experiments.

6.1 Datasets

To the best of our knowledge, there is no publicly available E-Commerce dataset that contains important features such as price and the labels of impression, click and purchase at the same time. Therefore, we collect a real-world dataset EC-REC [2] from a popular E-Commerce platform. Due to the huge amount of online data, we collect one-week data and sample over seven million impressions for offline experiments, and the dataset will be released to the public to support future studies. Meanwhile, we use PE-LTR to serve the users and conduct A/B test for online experiments. The features are from the user profiles and item profiles, for example the purchasing power of users and the average number of purchases of items.

6.2 Experimental Settings

We conduct both offline and online experiments to validate the effectiveness of the proposed framework. state-of-the-art approaches are selected for comparison.

6.2.1 Baselines. We select the state-of-the-art recommendation approaches for comparison and the baselines can be categorized into the three kinds: the typical approaches (CF, LambdaMART), the GMV-oriented approaches (LETORIF, MTL-REC), and the approaches that optimize both objectives (CXR-RL, PO-EA).

- **ItemCF**: Item-based Collaborative Filtering [31].
- **LambdaMART** [37] is a state-of-the-art learning-to-rank approach. A MART model is used to optimize a differentiable loss for NDCG. However, LambdaMART only concerns with clicks relevance, while purchase is not considered.
- **LETORIF** [36] is a recent learning-to-rank approach for GMV maximization and adopts price*CTR*CVR for ranking, where CTR and CVR are predictions from the two separate models.
- **MTL-REC**: MTL-REC [21] adopts multi-task learning techniques for training both CTR and CVR models. Two models share same user and item embeddings and similar neural network structures. The ranking model is also price*CTR*CVR.
- **CXR-RL** : CXR-RL [24] is a recent value-aware recommendation algorithm that optimizes CTR and CVR simultaneously. CXR is designed as a combination of CTR and CVR. CXR-RL uses reinforcement learning techniques to optimize CXR, thus achieving a trade-off between CTR and CVR.
- **PO-EA**: PO-EA [28] is a state-of-the-art multi-objective recommendation approach which aims to find Pareto efficient solutions. PO-EA assumes that different elementary algorithms have different advantages on the objectives. It aggregates the scores given by multiple elementary algorithms and the weights are generated with an evolutionary algorithm. The elementary algorithms include LETORIF-CTR, LETORIF, CXR-RL, PE-LTR-CTR, and PE-LTR-GMV. LETORIF-CTR refers to the CTR model in LETORIF. Both PE-LTR-CTR and PE-LTR-GMV are PE-LTR models whose boundary constraints are added to optimize CTR and GMV correspondingly. The two LTR models are used as elementary algorithms for a fair comparison with PE-LTR.
- **PO-EA-CTR, PO-EA-GMV**: two solutions generated by PO-EA, which focus on CTR and GMV respectively.
- **PE-LTR-CTR, PE-LTR-GMV**: two solutions generated by PE-LTR, which focus on CTR and GMV respectively.

6.2.2 Experimental Settings. We adopt two typical IR metrics for CTR evaluation, i.e. NDCG and MAP. Meanwhile, we propose two GMV variants for both metrics:

$$\text{G-AP@K} = \frac{1}{K}\sum_{n=1}^{K}\frac{\sum_{i=1}^{n}\text{pay}_i}{n}; \quad \text{G-MAP@K} = \frac{1}{|Q_R|}\sum_{q\in Q_R}\text{G-AP@K}$$

$$\text{G-DCG@K} = \sum_{i=1}^{K}\text{price}'_i \cdot \frac{2^{\text{pay}_i}-1}{\log_2(i+1)}; \quad \text{G-NDCG@K} = \frac{\text{G-DCG@K}}{\text{G-IDCG@K}}$$

where Q_R denotes the set of purchased items, $\text{pay}_i = 1/0$ denotes the whether the item at i-th rank is purchased or not, price'_i denotes the price of the item at i-th rank, G-IDCG@K denotes the maximum possible value of G-DCG@K. G-NDCG considers the position biased GMV in the list, and prefers higher-ranking items that are purchased, while G-MAP considers the number of purchases in the recommendation list. For users without purchase records, the values of two metrics are both 0.

6.3 Offline Experimental Results

6.3.1 Comparison with baselines. To answer the first research question, we present the comparison on NDCG, MAP and the GMV-related metrics in Table 2. PE-LTR is the model selected from Pareto Front with fairness marginal utility and PO-EA is a PO-EA model with comparable CTR metrics with PE-LTR. As shown in the table, PE-LTR outperforms other approaches on all GMV related metrics and a comparable performance with LambdaMART on CTR related metrics. Compared with Item-CF and LambdaMART, PE-LTR achieves much higher G-NDCG and G-MAP. This is reasonable since PE-LTR jointly optimize the GMV and CTR while GMV is not optimized in Item-CF and LambdaMART. Meanwhile, PE-LTR achieves comparable NDCG and MAP with LambdaMART. In previous observations on benchmark studies of web search, LambdaMART is usually the best performing method [36, 37]. This indicates the effectiveness of our framework, which not only optimizes GMV but also guarantees a high CTR.

Compared with LETORIF, MTL-REC, CXR-RL and PO-EA, PE-LTR achieves higher G-NDCG and G-MAP, and at a much lower cost of CTR. There are several reasons behind this:

First, compared with LETORIF and MTL-REC, PE-LTR jointly learns both objectives with a single model, which allows the model to learn clicks and purchases simultaneously; While in LETORIF and MTL-REC, two separate models or components are designed for clicks and purchases, which may cause some inconsistency.

Second, compared with CXR-RL and PO-EA, PE-LTR coordinates two objectives in a Pareto efficient way. CXR-RL optimizes both objectives, yet in a non-Pareto efficient way. Meanwhile, although PO-EA attempts to find Pareto efficient solutions, it only guarantees

[2] https://drive.google.com/open?id=1rbidQksa_mLQz-V1d2X43WuUQQVa7P8H, the codes will be released if the paper is accepted.

Table 2: Comparison between PE-LTR and other baselines in offline experiments, the values are relative improvements over ItemCF. The highest and second highest values are highlighted. All results are statistically significant with $p < 0.01$.

Approaches	G-NDCG@10	G-NDCG@ALL	G-MAP@10	G-MAP@ALL	NDCG@10	NDCG@ALL	MAP@10	MAP@ALL
ItemCF	-	-	-	-	-	-	-	-
LambdaMART	0.7357	-0.0324	-0.0982	-0.0869	**0.1602***	**0.0849***	**0.1503***	**0.1531***
LETORIF	0.1360	0.0660	0.1310	0.1108	-0.0327	-0.0189	-0.0357	-0.0283
MTL-REC	0.1092	0.0491	0.0952	0.0803	-0.0318	-0.0191	-0.0370	-0.0286
CXR-RL	0.0851	0.0443	0.0971	0.0796	0.0969	0.0538	0.0965	0.0945
PO-EA	0.0539	0.0246	0.0541	0.0435	0.0941	0.0510	0.0918	0.0914
PO-EA-GMV	0.3328	0.1890	0.3912	0.3368	0.0620	0.0319	0.0505	0.0596
PO-EA-CTR	0.0203	0.0052	0.0142	0.0102	0.1349	0.0744	0.1315	0.1318
PE-LTR	0.2707	0.1588	0.3292	0.2867	0.1150	0.0617	0.1080	0.1109
PE-LTR-GMV	**0.3629***	**0.2088***	**0.4311***	**0.3747***	0.0620	0.0306	0.0509	0.0589
PE-LTR-CTR	0.0268	0.0100	0.0231	0.0189	0.1412	0.0772	0.1351	0.1367

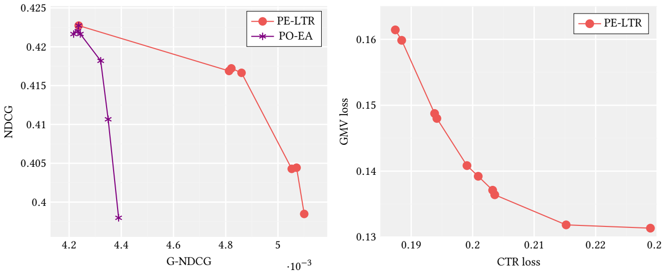

Figure 2: The left figure is the comparison between models generated by PO-EA and PE-LTR; The right figure is the Pareto front of CTR and GMV traning losses in PE-LTR.

that the final solution is selected from a series of solutions that are not dominated by each other. We further plot the NDCG versus G-NDCG curve of PO-EA and PE-LTR in Fig 2 (Due to page limit, we just plot G-NDCG and NDCG in the figures of the paper, and the results are similar for MAP and G-MAP). As the figure shows, any solution generated by PO-EA is not dominated by the other one from PO-EA; and the case is same with PE-LTR. However, we observe that the curves of PE-LTR are above the curves of PO-EA, which means the solutions from PO-EA are dominated by those generated by PE-LTR. Note that two PE-LTR algorithms are already used as the elementary components in PO-EA, the comparison indicates that the proposed framework is more capable to generate Pareto efficient solutions.

Moreover, the real-world data in E-Commerce platforms may not follow the typical i.i.d. assumption. And scalarization weights are adjusted every batch in PE-LTR, which allows it to adjust to the training data dynamically during the training process. Meanwhile, PO-EA requires several well-trained algorithms for aggregation, which makes it more difficult to meet the requirements of online learning environments.

We further compare the quality of recommendations at the top of the ranking list. Since users usually focus more on the top-ranked items, the metrics at the top are more important in recommendation. The results are presented in Fig 3 . As shown in the figures, PE-LTR outperforms the other baselines on GMV related metrics, and at a low cost of CTR. This illustrates the importance of Pareto efficiency in real-world recommender systems. Optimizing a single objective alone may hurt the other objectives severely. Therefore it is necessary to jointly consider multiple objectives simultaneously and a Pareto efficiency recommendation makes it possible to achieve high GMV at a low cost of CTR.

6.3.2 The Pareto Efficiency of PE-LTR. To answer the second research question, we first generate the Pareto Frontier of CTR and GMV losses by running Alg. 1 with different bounds and plot the Pareto Frontier in Fig 2. It can be observed that the losses under different constraints basically follow Pareto efficiency, i.e. no point achieves both lower CTR and GMV losses than other points. When the model focuses more on CTR, CTR loss is lower and GMV loss is higher, and vice versa. This coincides with the Pareto efficient scalarization scheme of the proposed framework.

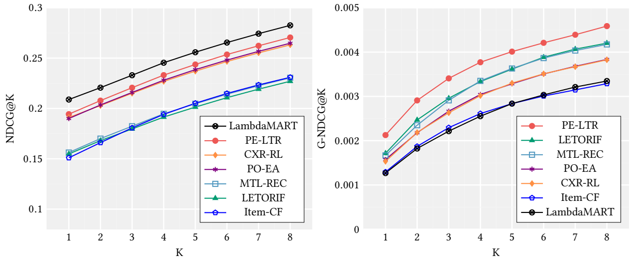

Figure 3: Comparison between the performances of PE-LTR and other baselines at the Top.

Then we compare the solution of PE-LTR under different solution selection strategies. We predefine two series of bounds for ctr and gmv: (ω_{ctr} ≥ 0, $omega_{gmv}$ ≥0.8) and (ω_{ctr} ≥ 0.8,$omega_{gmv}$ ≥ 0.0), and get two PE-LTRs (PE-LTR-GMV and PE-LTR-CTR) which focus on GMV and CTR respectively. Then we choose two PE-LTRs (PE-LTR-LM and PE-LTR-MU) from the Pareto Frontier with LM fairness and MU fairness. We plot the comparison between these PE-LTRs in Fig 4.

The performances of PE-LTR-CTR and PE-LTR-GMV are consistent with the constraints added to the objectives. Therefore when the priority of GMV and CTR are available (i.e. GMV or CTR is preferred), the recommendation can be achieved by setting the bounds correspondingly. When the priorities are not available, a fair solution can be achieved by selecting from Pareto Frontier with highest fairness. Despite the performance of selected PE-LTR (PE-LTR-LM and PE-LTR-MU) is not the best on all metrics, it achieves

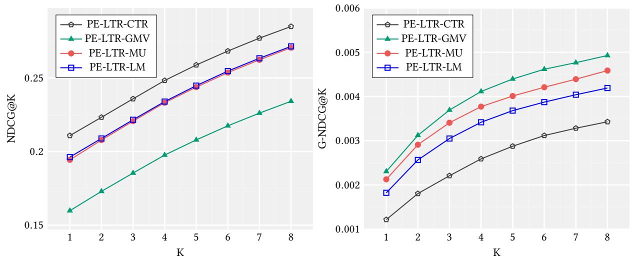

Figure 4: Comparison between different recommendation selection strategies in PE-LTR.

a relatively good trade-off between the two objectives. Comparing PE-LTR-LM with PE-LTR-MU, we find the two recommendations selected with LM and MU fairness are relatively balanced. PE-LTR-MU outperforms PE-LTR-LM in GMV while PE-LTR-LM is slightly better in CTR.

6.3.3 The Scalability of PE-LTR. To answer the third research question, we conduct experiments to show the scalability of PE-LTR in terms of model selection. We use LR, DNN and WDL as the model in PE-LTR framework, and the details of the models can be found in Section 5. We set same bounds for the models and the results are plotted in Fig 5.

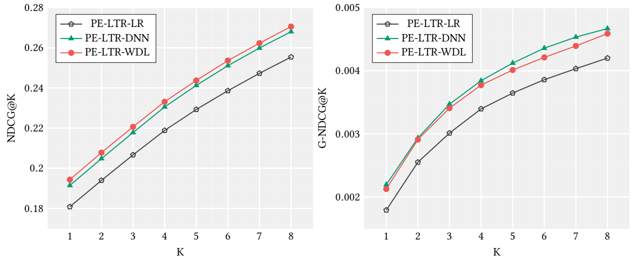

Figure 5: Comparison between different models with PE-LTR framework.

Judging from the results, we observe that the model selection has an important impact on the performance of PE-LTR. Among the three PE-LTR variants, PE-LTR-WDL outperforms the rest and PE-LTR-DNN outperforms PE-LTR-LR. This is reasonable since neural networks capture more complex relationships between features than linear models. And Wide&Deep model combines both neural networks and linear models into a single model, which enables better generalization and memorization for recommendation [9]. Therefore, PE-LTR is able to accommodate with varies kinds of models and stronger models can lead to better performances. This also illustrates the potential of PE-LTR, whose performance can be further enhanced by more carefully designed models.

6.4 Online Experimental Results

The online experiments are conducted on the real-world E-Commerce platform for three days. For online experiments, CTR-only approaches hurt GMV severely. Therefore, the approaches that only concern with CTR are not included in the online experiments.

We concern with four metrics in the online experiments, i.e. CTR (Click Through Rate), IPV (Individual Page View), PAY (number of payments) and GMV (Gross Merchandise Volume). We compute the average performances of three days and present the results in Table 3. Due to the large number of users, the results are statistically significant. We use LETORIF as the baseline, and present the relative improvements of compared approaches on LETORIF in the table.

Table 3: Comparison between PE-LTR and other baselines in online experiments, the values are the relative improvements over LETORIF in percentage. All results are statistically significant with $p < 0.01$.

Approaches	CTR	IPV	PAY	GMV
CXR-RL	13.68	20.60	-1.027	-3.197
PO-EA	4.442	8.957	3.399	-3.038
PE-LTR	**13.80***	**23.76***	**20.09***	**3.623***

From the results we observe that our approaches outperform other baselines on all the four metrics. This basically coincides with the offline experimental results. Note that PE-LTR achieves significant improvements on GMV with a high CTR, this illustrates the advantage of Pareto efficient recommendation. Meanwhile, PO-EA requires offline models for aggregation and can not learn the weights online, making it less effective in the experiments.

7 CONCLUSIONS

In this paper, we concern with the problem of recommendation with multiple objectives. We propose a general algorithmic framework that generates Pareto efficient solutions with theoretical guarantees. We propose a theoretical condition ensuring the Pareto efficiency, and a two-step algorithm which can be further accommodated with constraints on the objectives. We specifically apply this framework on E-Commerce recommendation to optimize both GMV and CTR simultaneously. Extensive experiments have been conducted on a real-world E-Commerce recommender system. The experimental results validate the effectiveness of the proposed framework. Meanwhile, the framework is model and objective agnostic, which shows its strong scalability.

A APPENDIX

A.1 The Proofs of Two Theorems

PROOF. The problem in Theorem 3.3 can be written as:

$$\begin{aligned} \min.\ & \frac{1}{2}\hat{\omega}^T GG^T \hat{\omega} + c^T GG^T \hat{\omega} + \frac{1}{2} c^T GG^T c \\ s.t.\ & e^T \hat{\omega} = 1 - e^T c \end{aligned}$$

We apply the Lagrange multipliers and get the Lagrangian:

$$\mathcal{L}(\hat{\omega}, \lambda) = \frac{1}{2}\hat{\omega}^T GG^T \hat{\omega} + c^T GG^T \hat{\omega} + \lambda(e^T \hat{\omega} - 1 + e^T c)$$

The solution to the problem is given by:

$$\nabla_{\hat{\omega}} \mathcal{L}(\hat{\omega}, \lambda) = 0, \text{ and } \nabla_{\lambda} \mathcal{L}(\hat{\omega}, \lambda) = 0,$$

therefore the solution can be achieved by solving the linear system:

$$\begin{bmatrix} GG^T & e \\ e^T & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = M \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -GG^T c \\ 1 - e^T c \end{bmatrix}$$

. And according to the study on Moore-Penrose inverse [25], the solution to this system is

$$\begin{bmatrix} x \\ \lambda \end{bmatrix} = (MM^T)^{-1} M \begin{bmatrix} -GG^T c \\ 1 - e^T c \end{bmatrix}$$

□

REFERENCES

[1] Kamelia Aryafar, Devin Guillory, and Liangjie Hong. 2017. An Ensemble-based Approach to Click-Through Rate Prediction for Promoted Listings at Etsy. In *Proceedings of the ADKDD'17 (ADKDD'17)*. ACM, New York, NY, USA, Article 10, 6 pages.

[2] Stephen Boyd and Lieven Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press, New York, NY, USA.

[3] Rasmus Bro and Sijmen De Jong. 2015. A Fast Non-negativity-constrained Least Squares Algorithm. *Journal of Chemometrics* 11, 5 (2015), 393–401.

[4] 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 (ICML '05)*. ACM, New York, NY, USA, 89–96.

[5] Chris J.C. Burges. 2010. *From RankNet to LambdaRank to LambdaMART: An Overview*. Technical Report.

[6] 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 (ICML '07)*. ACM, New York, NY, USA, 129–136.

[7] Abhijnan Chakraborty, Saptarshi Ghosh, Niloy Ganguly, and Krishna P. Gummadi. 2017. Optimizing the Recency-Relevancy Trade-off in Online News Recommendations. In *Proceedings of the 26th International Conference on World Wide Web (WWW '17)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 837–846.

[8] Xiang Chen, Bowei Chen, and Mohan Kankanhalli. 2017. Optimizing Trade-offs Among Stakeholders in Real-Time Bidding by Incorporating Multimedia Metrics. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. 205–214.

[9] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems (DLRS 2016)*. 7–10.

[10] Laizhong Cui, Peng Ou, Xianghua Fu, Zhenkun Wen, and Nan Lu. 2017. A Novel Multi-objective Evolutionary Algorithm for Recommendation Systems. *J. Parallel Distrib. Comput.* 103, C (May 2017), 53–63.

[11] Jean-Antoine Désidéri. 2009. Multiple-Gradient Descent Algorithm (MGDA).

[12] Huizhong Duan, ChengXiang Zhai, Jinxing Cheng, and Abhishek Gattani. 2013. Supporting Keyword Search in Product Database: A Probabilistic Approach. *Proc. VLDB Endow.* 6, 14 (Sept. 2013), 1786–1797.

[13] Yoav Freund, Raj Iyer, Robert E. Schapire, and Yoram Singer. 2003. An Efficient Boosting Algorithm for Combining Preferences. *J. Mach. Learn. Res.* 4 (Dec. 2003), 933–969.

[14] Neil J. Hurley. 2013. Personalised Ranking with Diversity. In *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys '13)*. 379–382.

[15] Tamas Jambor and Jun Wang. 2010. Optimizing Multiple Objectives in Collaborative Filtering. In *Proceedings of the Fourth ACM Conference on Recommender Systems (RecSys '10)*. ACM, New York, NY, USA, 55–62.

[16] Dietmar Jannach, Zeynep Karakaya, and Fatih Gedikli. 2012. Accuracy Improvements for Multi-criteria Recommender Systems. In *Proceedings of the 13th ACM Conference on Electronic Commerce (EC '12)*. ACM, New York, NY, USA, 674–689.

[17] Marius Kaminskas and Derek Bridge. 2016. Diversity, Serendipity, Novelty, and Coverage: A Survey and Empirical Analysis of Beyond-Accuracy Objectives in Recommender Systems. *ACM Trans. Interact. Intell. Syst.* 7, 1, Article 2 (Dec. 2016), 42 pages.

[18] Shubhra Kanti Karmaker Santu, Parikshit Sondhi, and ChengXiang Zhai. 2017. On Application of Learning to Rank for E-Commerce Search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. ACM, New York, NY, USA, 475–484.

[19] Beibei Li, Anindya Ghose, and Panagiotis G. Ipeirotis. 2011. Towards a Theory Model for Product Search. In *Proceedings of the 20th International Conference on World Wide Web (WWW '11)*. ACM, New York, NY, USA, 327–336.

[20] Ping Li, Chris J.C. Burges, and Qiang Wu. 2008. *Learning to Rank Using Classification and Gradient Boosting* (advances in neural information processing systems 20 ed.). Technical Report.

[21] 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 &#38; Development in Information Retrieval (SIGIR '18)*. ACM, New York, NY, USA, 1137–1140.

[22] Sean M. McNee, John Riedl, and Joseph A. Konstan. 2006. Being Accurate is Not Enough: How Accuracy Metrics Have Hurt Recommender Systems. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems (CHI EA '06)*. ACM, New York, NY, USA, 1097–1101.

[23] Phong Nguyen, John Dines, and Jan Krasnodebski. 2017. A Multi-Objective Learning to re-Rank Approach to Optimize Online Marketplaces for Multiple Stakeholders. *CoRR* abs/1708.00651 (2017).

[24] Changhua Pei, Xinru Yang, Qing Cui, Xiao Lin, Fei Sun, Peng Jiang, Wenwu Ou, and Yongfeng Zhang. 2019. Value-aware Recommendation based on Reinforcement Profit Maximization *(WWW'19)*.

[25] R. Penrose. 1956. On Best Approximate Solutions of Linear Matrix Equations. *Proceedings of the Cambridge Philosophical Society* 52, 1 (1956), 17–19.

[26] Filip Radlinski, Andrei Broder, Peter Ciccolo, Evgeniy Gabrilovich, Vanja Josifovski, and Lance Riedel. 2008. Optimizing Relevance and Revenue in Ad Search: A Query Substitution Approach. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08)*. ACM, New York, NY, USA, 403–410.

[27] Marco Tulio Ribeiro, Anisio Lacerda, Adriano Veloso, and Nivio Ziviani. 2012. Pareto-efficient Hybridization for Multi-objective Recommender Systems. In *Proceedings of the Sixth ACM Conference on Recommender Systems (RecSys '12)*. ACM, New York, NY, USA, 19–26.

[28] Marco Tulio Ribeiro, Nivio Ziviani, Edleno Silva De Moura, Itamar Hata, Anisio Lacerda, and Adriano Veloso. 2014. Multiobjective Pareto-Efficient Approaches for Recommender Systems. *ACM Trans. Intell. Syst. Technol.* 5, 4, Article 53 (Dec. 2014), 20 pages.

[29] Mario Rodriguez, Christian Posse, and Ethan Zhang. 2012. Multiple Objective Optimization in Recommender Systems. In *Proceedings of the Sixth ACM Conference on Recommender Systems (RecSys '12)*. ACM, New York, NY, USA, 11–18.

[30] Rómer Rosales, Haibin Cheng, and Eren Manavoglu. 2012. Post-click Conversion Modeling and Analysis for Non-guaranteed Delivery Display Advertising. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining (WSDM '12)*. ACM, New York, NY, USA, 293–302.

[31] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based Collaborative Filtering Recommendation Algorithms. In *Proceedings of the 10th International Conference on World Wide Web (WWW '01)*. 285–295.

[32] Ozan Sener and Vladlen Koltun. 2018. Multi-Task Learning as Multi-Objective Optimization. *CoRR* abs/1810.04650 (2018).

[33] Liang Tang, Bo Long, Bee-Chung Chen, and Deepak Agarwal. 2016. An Empirical Study on Recommendation with Multiple Types of Feedback. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. ACM, New York, NY, USA, 283–292.

[34] Damir Vandic, Flavius Frasincar, and Uzay Kaymak. 2013. Facet Selection Algorithms for Web Product Search. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management (CIKM '13)*. ACM, New York, NY, USA, 2327–2332.

[35] Shanfeng Wang, Maoguo Gong, Haoliang Li, and Junwei Yang. 2016. Multi-objective Optimization for Long Tail Recommendation. *Know.-Based Syst.* 104, C (July 2016), 145–155.

[36] Liang Wu, Diane Hu, Liangjie Hong, and Huan Liu. 2018. Turning Clicks into Purchases: Revenue Optimization for Product Search in E-Commerce. In *The 41st International ACM SIGIR Conference on Research &#38; Development in Information Retrieval (SIGIR '18)*. ACM, New York, NY, USA, 365–374.

[37] Qiang Wu, Christopher J. Burges, Krysta M. Svore, and Jianfeng Gao. 2010. Adapting Boosting for Information Retrieval Measures. *Inf. Retr.* 13, 3 (June 2010), 254–270.

[38] Lin Xiao, Zhang Min, Zhang Yongfeng, Gu Zhaoquan, Liu Yiqun, and Ma Shaoping. 2017. Fairness-Aware Group Recommendation with Pareto-Efficiency. In *Proceedings of the Eleventh ACM Conference on Recommender Systems (RecSys '17)*. ACM, New York, NY, USA, 107–115.

[39] Jun Xu and Hang Li. 2007. AdaRank: A Boosting Algorithm for Information Retrieval. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*. ACM, New York, NY, USA, 391–398.

[40] Jun Yu, Sunil Mohan, Duangmanee (Pew) Putthividhya, and Weng-Keen Wong. 2014. Latent Dirichlet Allocation Based Diversified Retrieval for e-Commerce Search. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining (WSDM '14)*. ACM, New York, NY, USA, 463–472.

[41] Mi Zhang and Neil Hurley. 2008. Avoiding Monotony: Improving the Diversity of Recommendation Lists. In *Proceedings of the 2008 ACM Conference on Recommender Systems (RecSys '08)*. ACM, New York, NY, USA, 123–130.

[42] Yongfeng Zhang, Qi Zhao, Yi Zhang, Daniel Friedman, Min Zhang, Yiqun Liu, and Shaoping Ma. 2016. Economic Recommendation with Surplus Maximization. In *Proceedings of the 25th International Conference on World Wide Web (WWW '16)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 73–83.

[43] Qi Zhao, Yongfeng Zhang, Yi Zhang, and Daniel Friedman. 2017. Multi-Product Utility Maximization for Economic Recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (WSDM '17)*. ACM, New York, NY, USA, 435–443.

[44] Yunzhang Zhu, Gang Wang, Junli Yang, Dakan Wang, Jun Yan, and Zheng Chen. 2009. Revenue Optimization with Relevance Constraint in Sponsored Search. In *Proceedings of the Third International Workshop on Data Mining and Audience Intelligence for Advertising (ADKDD '09)*. ACM, New York, NY, USA, 55–60.

[45] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. 2001. *SPEA2: Improving the Strength Pareto Evolutionary Algorithm*. Technical Report.