MERIT: A Merchant Incentive Ranking Model for Hotel Search & Ranking

Shigang Quan
Shanghai Jiao Tong University
Shanghai, China
quan123@sjtu.edu.cn

Hailong Tan
Alibaba Group
Hangzhou, Zhejiang, China
hailong.thl@alibaba-inc.com

Shui Liu
Alibaba Group
Hangzhou, Zhejiang, China
shui.lius@alibaba-inc.com

Zhenzhe Zheng
Shanghai Jiao Tong University
Shanghai, China
zhengzhenzhe@sjtu.edu.cn

Ruihao Zhu
Cornell University
Ithaca, USA
ruihao.zhu@cornell.edu

Liangyue Li
Alibaba Group
Hangzhou, Zhejiang, China
liliangyue.lly@alibaba-inc.com

Quan Lu
Alibaba Group
Hangzhou, Zhejiang, China
luquan.lq@alibaba-inc.com

Fan Wu
Shanghai Jiao Tong University
Shanghai, China
fwu@cs.sjtu.edu.cn

ABSTRACT

Online Travel Platforms (OTPs) have been working on improving their hotel Search & Ranking (S&R) systems that facilitate efficient matching between consumers and hotels. Existing OTPs focus almost exclusively on improving platform revenue. In this work, we take a first step in incorporating hotel merchants' objectives into the design of hotel S&R systems to achieve an incentive loop: the OTP tilts impressions and better-ranked positions to merchants with high service quality, and in return, the merchants provide better service to consumers. Three critical design challenges need to be resolved to achieve this incentive loop: Matthew Effect in the consumer feedback-loop, unclear relation between hotel service quality and performance, and conflicts between platform revenue and consumer experience.

To address these challenges, we propose **MERIT**, a **MER**chant **I**nce**T**ive ranking model, which can simultaneously take the interests of merchants and consumers into account. We introduce information about the hotel service quality at the input-output level. At the input level, we incorporate factors of hotel service quality as features (as the underlying reasons for service quality), while at the output level, we introduce the metric HRS as a label (as the evaluated outcome of service quality). Also, we design a monotonic structure for Merchant Tower to provide a clear relation between hotel quality and performance. Finally, we propose a Multi-objective Stratified Pairwise Loss, which can mitigate the conflicts between OTP's revenue and consumer experience. To demonstrate the effectiveness of MERIT, we compare our method with several state-of-the-art benchmarks. The offline experiment results indicate that MERIT outperforms these methods in optimizing the demands of consumers and merchants. Furthermore, we conduct an online A/B test and obtain an improvement of 3.02% for the HRS score. Based on these results, we have deployed MERIT online on Fliggy, one of the most popular OTPs in China, to serve tens of millions of consumers and hundreds of thousands of hotel merchants.

CCS CONCEPTS

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

KEYWORDS

Hotel Search & Ranking System; Monotonic Neural Networks; Hotel Service Quality

1 INTRODUCTION

Nowadays, Online Travel Platforms (OTPs) are applying deep learning [15, 16, 18, 36, 39] in their hotel Search & Ranking (S&R) systems, to accurately provide consumers with their interested hotels. Similar to product recommendation on online e-commerce platform [42, 43], hotel S&R systems also need to capture consumers' diverse interests in different hotels to optimize consumer experiences and platform revenue. However, different from conventional e-commerce platforms, which focus on optimizing online platform revenue, OTPs also have to take hotel service quality into account.

To ensure the healthy growth of the OTP and encourage hotel merchants to improve their service quality, it is important to achieve the incentive loop of service quality for hotel merchants as illustrated in Figure 1: the platform tilts impressions and better-ranked positions to the merchants with high service quality, and in return, the merchants provide better online and offline service, attracting more consumers to purchase and repurchase hotels.

However, achieving this incentive loop is not trivial. Though several works [16, 36, 41] started to consider merchant features, such as the price and location, into hotel S&R systems, the objective of these works is platform revenue rather than hotel service quality in practical. Recent studies [14, 38] incorporated consumer

CIKM '23, October 21–25, 2023, Birmingham, United Kingdom.

ACM ISBN 979-8-4007-0124-5/23/10…$15.00
https://doi.org/10.1145/XXXXXX.XXXXXX

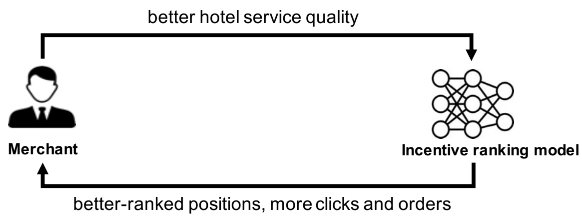

Figure 1: The positive incentive loop of hotel service quality for hotel merchants on OTPs.

implicit and explicit feedback into recommendation systems, but ignore the factors behind consumer ratings. In this work, we take a first step in achieving a positive incentive loop of hotel service quality within the design of hotel S&R systems. Several critical, yet largely overlooked, design challenges during the interaction between consumers and merchants through OTPs are summarized as follows:

• **Matthew Effect in consumer feedback-loop.** On OTPs, impressions and ranked positions of hotels are based on their historical CTRs (Click-Through Rates) and CVRs (Conversion Rates). The consumer feedback about CTRs and CVRs impacted by the current hotel positions will be used for the next ranking stage as a loop [30]. The above feedback-loop in current hotel S&R systems would cause Matthew Effect [5, 37]: more impressions and better-ranked positions are tilted toward those hotels with higher historical CTRs and CVRs, and merchants with better service quality would not obtain their desired ranked positions due to the lack of historical data. The OTPs need to break the Matthew Effect and provide flexible knobs for potential high-service-quality merchants to improve their hotels' exposure and ranking positions.

• **Unclear relation between hotel service quality and performance.** From hotel merchants' perspectives, they hope to receive explicit and confirmed performance feedback after improving the service quality. But existing hotel S&R systems directly take the features of consumers, hotel rooms and context, as the inputs of black-box learning models. The service quality of hotel rooms would be mixed or even under-weighted in various features during the ranking stage. Thus, hotel merchants cannot know a clear relation between hotel service quality and potential performance improvement. This unclear relation of hotel service quality and performance decreases the incentives for hotel merchants to improve their service quality on OTPs.

• **Conflicts between OTP's revenue and consumer experience.** A consumer's whole behavior cycle can be summarized as follows: a consumer books a hotel online on OTPs, checks in offline, and finally submits ratings on OTPs. OTPs focus almost exclusively on improving CTRs and CVRs, which can be thought of as OTP's revenue. However, the service quality of merchants can not be totally reflected by online booking. Ignoring the service quality of merchants will deteriorate the consumer experience. Therefore, from the perspective of OTPs, we should improve the consumer experience while striking to ensure that OTP's revenue does not significantly decrease.

By jointly considering these challenges, we propose **MERIT**, a **MER**chant **I**nce**T**ive ranking model, to explicitly represent the relation between hotel merchants and consumers, and optimize the OTP's revenue via the ranking stage in the hotel S&R system. For the first challenge, we aim to alleviate the Matthew Effect by introducing information about the hotel service quality at the input-output level. At the input level, we incorporate factors of hotel service quality as features (as the underlying reasons for service quality), while at the output level, we introduce the metric HRS as a label (as the outcome of service quality) to expect a final ranking score that reflects the overall hotel service quality. To address the second challenge, we introduce a monotonic neural network into the Merchant Tower. The monotonic neural network has been implemented in deep learning to guarantee the model interpretability [3, 8, 10, 26, 31]. The monotonicity in the Merchant Tower will provide an explicit ranking rule that can be understood by the hotel merchants, and guide them to further optimize factors of hotel service quality for higher ranking scores. To resolve the final challenge, we define a Multi-objective Stratified Pairwise Loss to mitigate conflicts between the OTP's revenue and consumer experience, which guarantees that we can improve the consumer experience without sacrificing the OTP's revenue too much. Based on the solution described above, MERIT jointly learns the interests and intentions of merchants and consumers, improving the consumer-hotel matching performance of the hotel S&R system in the long run. We conduct an extensive offline experiment on a large-scale offline hotel S&R dataset, and results show that MERIT is effective in improving merchant satisfaction and platform performance. In the online A/B test, the average HRS scores have increased by 3.02% compared with the baseline model. Furthermore, MERIT has been deployed on Fliggy[1], one of the most popular OTPs in China, where it serves tens of millions of consumers and hundreds of thousands of hotel merchants. We summarize our main contributions in this work as follows:

(1) We explicitly incorporate the hotel merchants' service quality into hotel (S&R) systems. To evaluate hotel service quality, we define the HRS metric and propose a novel MERIT model to effectively model the multiple factors that contribute to hotel service quality.
(2) We introduce the monotonicity property for the merchant Tower to establish a clear and positive correlation between hotel service quality and ranking performance.
(3) In order to mitigate the conflicts between OTP's revenue and consumer experience, we propose the Multi-objective Stratified Pairwise Loss. The extensive offline experiment indicates that the ranking process can be improved via this loss function as opposed to multi-objective pairwise loss.
(4) We conduct both offline evaluations on a large-scale hotel S&R dataset and an online A/B test. Evaluation results show that MERIT is effective in improving hotel service quality and platform performance. Furthermore, MERIT has been deployed on Fliggy and brought remarkable profit growth.

[1] http://www.fliggy.com/

2 RELATED WORK

2.1 Click-through Rate Prediction

In the search engine, online advertising, and recommendation system, predicting consumers' CTR has become a key problem and a lot of works have focused on this area. Recently, the deep learning methods [6, 12, 17, 18, 27, 34, 42, 43] have been introduced into CTR prediction models due to the strong representation of deep learning methods. These CTR prediction models often follow the paradigm of Embedding Layer and MLP. To strengthen the capability of capturing the nonlinear feature interaction, some works propose specific networks to learn high-order cross features. Cheng et al. [6] use the wide structure to memorize nonlinear features and utilize the deep model to improve the generalization of the feature interaction. Guo et al. [12] propose the DeepFM to combine the traditional factorization machine method [28] and DNN. Another line of the CTR prediction model is capturing consumer interests from consumer historical behaviors. Zhou et al. [43] propose DIN to utilize the attention mechanism to capture consumers' diverse interests. DIEN [42] extends DIN and uses the Gated Recurrent Unit (GRU) [7] layer to model the process of sequential interest evolving.

Our research is different from CTR prediction methods in two aspects: First, the CTR prediction model considers the consumer click behavior as its label while our method considers both consumer sequential behaviors and the merchant quality as our labels. Another distinction is that we incorporate and model merchant interests, which is not taken into account in previous work.

2.2 Multi-task Learning

Multi-task learning is to simultaneously utilize a composite model to complete different tasks. The original multi-task learning model often takes the shared-bottom layers [4] to model the complex relation between various tasks. But this design of shared-bottom layersis strongly constrained. MoE [19] proposes the gate layer to balance the conflicts of different tasks. MMoE [23] extends MoE to add multiple gates. Tang et al. [32] propose PLE and utilize task-specific experts and task-shared experts to mitigate the negative transfer problem between tasks. The multi-task learning models in the recommendation system and online advertising are often used to predict consumer CTRs and CVRs. Ma et al. [24] propose ESMM to tackle the problems of data sparsity and sample selection bias. Wen et al. [35] propose the consumer post-click behavior decomposition method to address the aforementioned problems.

Our method also attempts to optimize multi-tasks, but we do not adhere to the basic paradigm of Shared&Gated Networks because the objectives of consumers and merchants are heterogeneous and the shared structure is inappropriate for this scenario.

2.3 Monotonicity in Neural Networks

Adding prior knowledge to a model can reduce the search space of model parameters. As one of prior information, monotonicity can ensure the neural networks can generalize better and improve the interpretability of a black-box neural network. Current adding monotonicity in neural networks can be classified into two types. The first type is constraining the structure of neural networks. The

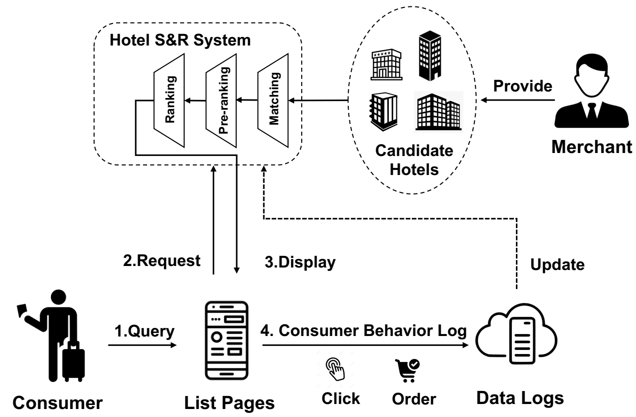

Figure 2: The overview framework of hotel S&R system.

MIN-MAX Network [9] utilizes a three-layer network to make the model partial monotonic. You et al. [40] propose Deep Lattice Networks (DLN) and take the ensembles of lattice and calibrators as the constraints of monotonicity. Though DLN can improve the precision, the complexity of the model also increases. Another type is changing the loss function. Gupta et al. [13] propose pointwise monotonic loss to penalize the negative gradients. Liu et al. [21] add the monotonicity regularization with the uniform data sampling.

In contrast to former works [16, 36], which utilize the monotonicity structure to improve the accuracy, in this paper, we follow the monotonicity structure to provide an explicit ranking rule that can be understood by the merchants, and guide them to further optimize their hotels for higher ranking scores.

3 PRELIMINARIES

In this section, we first give a brief overview of the hotel S&R system on Fliggy, and then define the Hotel Rating Score (HRS) in this scenario. We finally formulate the hotel S&R problem on OTPs.

3.1 An Overview of Hotel S&R System

We briefly introduce the hotel S&R system illustrated in Figure 2. In the scenario of the hotel S&R system, a consumer can issue a query depending on her travel plan. This request will be sent to the hotel S&R system, which will select the candidate hotels provided by the merchant through different stages based on the amount of data, computational resources, and response latency: matching, pre-ranking, ranking, and so on. After the final ranking stage, the system recommends the top hotels sorted by their ranking scores, and the consumer will click on the corresponding hotel and enter the detail page. The detailed information includes available hotel rooms, prices, and consumer reviews. Orders are then placed by consumers. Each time a consumer interacts sequentially with the hotel S&R system, her sequential behaviors (click, order, etc.) will be logged in data logs and used to update the parameters of the hotel S&R system.

(a) HRS on the listing page **(b) Consumer rating scores on the comment page**

Figure 3: Areas highlighted with red rectangles are HRS and consumer rating scores in Fliggy.

3.2 Hotel Service Quality

From the perspective of OTPs, we not only want consumers to purchase hotels but also to like purchasing more hotels after they have purchased hotels. Attracting consumers to purchase more hotels primarily depends on hotel service quality, so in the following subsections, we show aspects of input features and ranking labels to incorporate hotel service quality into our proposed MERIT.

3.2.1 Hotel Rating Score. The consumer implicit feedbacks (e.g. click, order) indicate the consumer's online preference on OTPs while consumer rating scores (from 1 to 5) reflect the hotel service quality in Figure 3(b). We do not follow previous works [20, 22], which adopt individual consumer rating scores as the ground truth label. Our objective is to assess the overall quality of the hotel's service, not merely the consumer's personal preferences. Therefore, in this paper, we utilize the metric *Hotel Rating Score* (HRS) as an evaluation of hotel service quality. As illustrated in Figure 3(a), HRS, which is determined by the average rating given by consumers for the current hotel, reflects the overall evaluation of the hotel service quality by consumers. HRS will be used as a ranking label z in Section 4.1.

3.2.2 Factors of Hotel Service Quality. HRS is the result feedback on hotel service quality, while hotel service quality is mainly composed of several service factors. As previously mentioned, consumers undergo a long behavior cycle on OTPs: first, consumers place orders on OTPs with a probability of acceptance by merchants. Then during the offline check-in stage, hotel merchants with no remaining rooms will reject consumer requests to check in. After check in and check out, consumers may complain about hotels with their poor service. Throughout the entire behavior cycle, consumers require the response of manual customer service for any problems they encounter. Therefore, we list the impact factors on the hotel service quality of merchants for each stage in Table 1. All these factors will be used as input features, named as X^s in Section 4.1.

Table 1: Factors of Hotel Service Quality. (The higher the scores of these factors, the better the hotel service quality.)

Factors	Descriptions
Order Confirmation Rate	The percentage of online orders for this hotel that are confirmed and accepted.
Check-in Success Rate	The success rate of offline check-in for consumers who booked the hotel online.
Non-Complaint Rate	The non-complaint rate of consumers who booked the hotel online.
Manual Service Response Rate	The proportion of consumers who successfully connect to manual customer service of this hotel by phone.

3.3 Problem Definition

In the scenario of the hotel S&R system, we assume the dataset to be $\mathcal{D} = \{(\boldsymbol{x}_i \rightarrow y_i, z_i)\}|_{i=1}^{N}$, and we draw the sample $(\boldsymbol{x} \rightarrow y, z)$ from domain $\mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$, where $\mathcal{X}$ is feature space, $\mathcal{Y}$ and $\mathcal{Z}$ are label spaces, and N is the size of $\mathcal{D}$. $\boldsymbol{x}$ can be defined in the form of $\boldsymbol{x} = (x^u, x^q, x^h)$, in which u is the consumer on the OTP, q is the query issued by u, and h is the target hotel. y is a three-class label, with $y = 0$ indicating not being clicked and ordered, $y = 1$ indicating being clicked but not ordered, and $y = 2$ indicating being clicked and ordered. z is the HRS label. So on the OTP, the objective of the hotel S&R system is to learn a model $\mathcal{F}$ from the dataset $\mathcal{D}$. $\mathcal{F}$ aims to predict the ranking score $\hat{s}$ for consumer u and target hotel h based on the input features $\boldsymbol{x}$:

$$\hat{s} = p\left(y, z \mid \boldsymbol{x}\right), \tag{1}$$

where $\hat{s}$ is the $pCTCVR$ ($pCTCVR$ will be explained in Section 4.2) and is obtained by $\hat{s}^{CTR} \times \hat{s}^{CVR}$.

4 METHODOLOGY

In this section, we give a detailed introduction to our approach as illustrated in Figure 4. Features and their representation are provided in Section 4.1. Section 4.2 introduces the framework of the Merchant Incentive Layer and to tackle the conflict of multi-objective pairwise loss, we propose the MSPL in Section 4.3.

4.1 Feature Representation

We use four types of features: consumer features, query features, hotel features, and hotel service quality features. Consumer features are divided into three categories: consumer basic profile x^p (age, purchase level, etc.), consumer historical preference features x^b (consumer preference on the price and location of a hotel based on her historical interaction behaviors), and consumer context features x^c (time, pid, etc.). The query features x^q include search keywords. The hotel features are hotel item basic profile x^h. The features of hotel service quality in Table 1 will be represented as x^s.

The aforementioned categorical features are encoded as one-hot vectors, while continuous features are discretized and encoded as vectors. Input vectors are represented by the following symbols: X^p, X^b, X^c, X^q, X^h, X^s. To transform these high-dimensional, sparse one-hot vectors into low-dimensional, dense vectors, we utilize

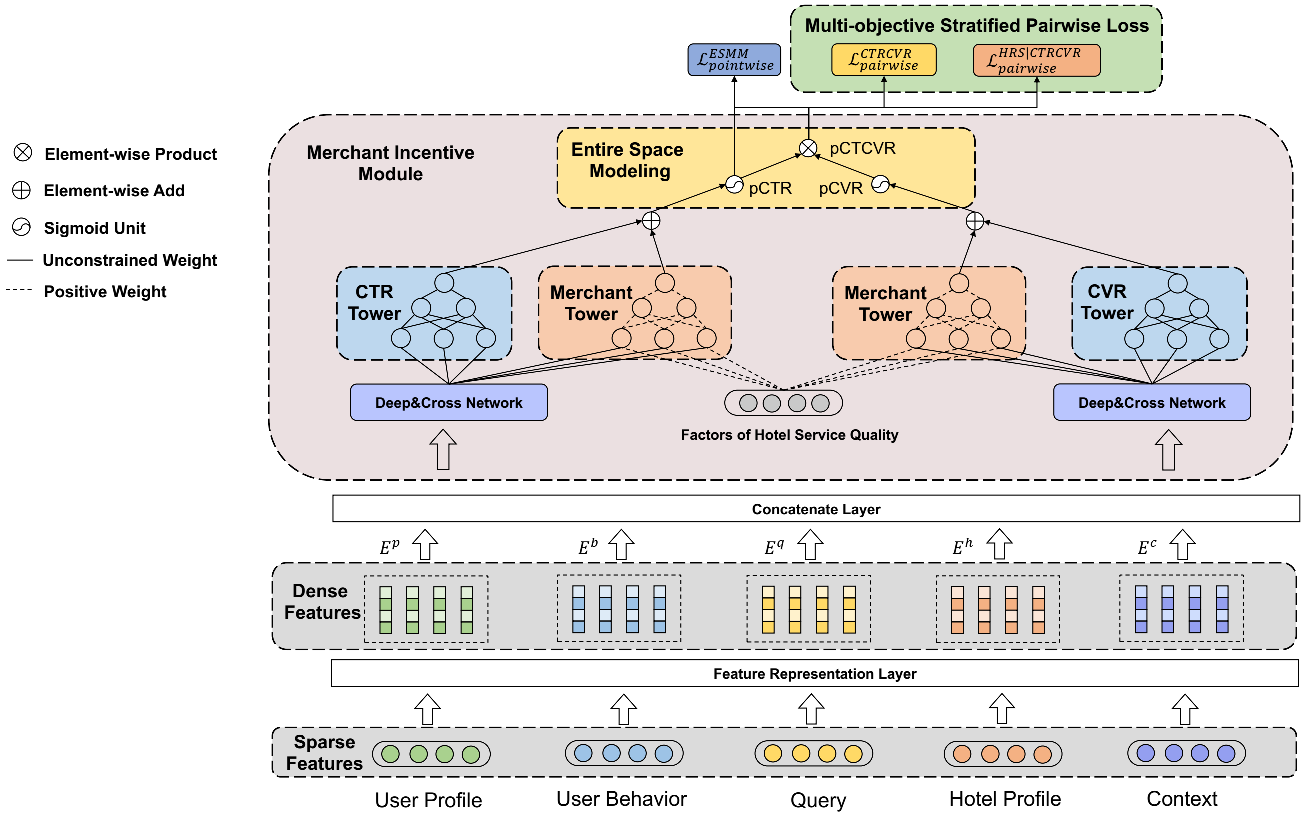

Figure 4: The overview architecture of MERIT, which consists of the Feature Representation Layer, Concatenate Layer, Merchant Incentive Module, and MSPL (Multi-objective Stratified Pairwise Loss). The Merchant Incentive Module consists of Deep&Cross Network, CTR/CVR Tower, Merchant Tower, and Entire Space Modeling.

embedding layers [2, 25]. These low-dimensional, dense vectors can be denoted as $(E^p, E^b, E^c, E^q, E^h)$.

4.2 Merchant Incentive Module

To learn the relation between the consumer and the target hotel, we utilize a concatenate layer to concatenate all the embedded feature vectors:

$$E = [E^p, E^b, E^q, E^h, E^c], \tag{2}$$

where r denotes the concatenated embedded vector. Then in order to learn the high-order and low-order interactions of features, we adopt the Deep&Cross Network [34] into our model as follows:

$$\begin{aligned} E^{CTR} &= DCN(E), \\ E^{CVR} &= DCN(E), \end{aligned} \tag{3}$$

where E^{CTR} and E^{CVR} denote the representation vectors for CTR and CVR predictions, respectively. For the CTR part, the CTR Tower[2] will learn a parameterized mapping function from the representation E^{CTR} to $pCTR$ while the Merchant Tower will learn a parameterized mapping function for E^{CTR} and a monotonic parameterized mapping function for X^s as follows:

$$pCTR = \phi^B_\theta(E^{CTR}, X^s) + \psi^{CTR}_\theta(E^{CTR}), \tag{4}$$

where $\phi^B_\theta(\cdot)$ is a positive-weight feed-forward network [1] and $\psi^{CTR}_\theta(\cdot)$ is a feed-forward network. The Merchant Tower is designed to calibrate the predicting process and strengthen the monotonicity of our model. And CVR prediction structure is similar to the CTR prediction structure as follows:

$$pCVR = \phi^B_\theta(E^{CVR}, X^s) + \psi^{CVR}_\theta(E^{CVR}), \tag{5}$$

where $\psi^{CVR}_\theta(\cdot)$ is a feed-forward network. The relation between the features of hotel service quality and the final ranking score will adjust based on the context, so E^{CTR} and E^{CVR} will be also fed into the Merchant Tower ($\phi^B_\theta(\cdot)$) to cross X^s and other features.

In order to avoid sample Selection Bias (SSB) and Data Sparsity (DS) issues [24], we follow the entire space modeling [24]. We calculate $pCTCVR$ (predicted post-view Click-Through rate&ConVersion Rate) as our final ranking score, and its definition is as follows:

$$pCTCVR = pCTR \times pCVR. \tag{6}$$

[2]"Tower" is a term used in the field of multi-task learning and typically refers to a neural network designed for individual tasks.

4.3 Multi-objective Stratified Pairwise Loss

Unlike pointwise loss, which calculates the numerical gap of each sample, pairwise loss [29] measures the range between positive and negative samples. Pairwise loss outperforms pointwise loss in representing the overall ranking performance. Consequently, we incorporated pairwise loss to evaluate the ranking outcomes. However, a conflict may arise when undertaking multi-objective optimization using pairwise loss:

Conflict of Multi-objective Pairwise Loss For the sake of simplicity, we will use two objectives, Y and Z, as an example, where Y is the primary objective and Z is the secondary objective. And for each objective, $\mathcal{Y}^+, \mathcal{Y}^-$ are the positive and negative sample sets of Y. $\mathcal{Z}^+$ and $\mathcal{Z}^-$ are the positive and negative sample sets of Z. Pairwise loss functions can be defined as follows:

$$\begin{aligned} \mathcal{L}_{pairwise}^{Y} &= \sum_{i=1}^{N}\sum_{j=1}^{N} \ell(\hat{s}_i - \hat{s}_j, \mathbb{I}(Y_i > Y_j)), \\ \mathcal{L}_{pairwise}^{Z} &= \sum_{i=1}^{N}\sum_{j=1}^{N} \ell(\hat{s}_i - \hat{s}_j, \mathbb{I}(Z_i > Z_j)), \end{aligned} \tag{7}$$

where $\mathbb{I}$ is the indicator function and $\ell(\cdot)$ is the negative log-likelihood loss function. Suppose that if $i \in \mathcal{Y}^+ \cap \mathcal{Z}^-$ and $j \in \mathcal{Y}^- \cap \mathcal{Z}^+$, then the conflict problem occurs: For objective Y, the pairwise loss function $\mathcal{L}_{pairwise}^{Y}$ is encouraged to enlarge the ranking score gap between i and j, while for Z, the pairwise loss function $\mathcal{L}_{pairwise}^{Z}$ is encouraged to enlarge the ranking score gap between j and i. During the training process, inconsistency in gradient directions will deteriorate the effectiveness of multi-objective optimization. In the scenario of the hotel S&R system, the above conflict corresponds to Challenge 3: the conflict of platform revenue and consumer experience.

Multi-objective Stratified Pairwise Loss In order to tackle the above conflict problem, we define the Multi-objective Stratified Pairwise Loss (MSPL) for the multi-objective ranking. It can be defined as follows:

$$\mathcal{L}_{pairwise}^{Z|Y} = \sum_{i=1}^{N}\sum_{j=1}^{N} \mathbb{I}(Y_i \geq Y_j)\ell(\hat{s}_i - \hat{s}_j, \mathbb{I}(Z_i > Z_j)). \tag{8}$$

The above pairwise loss function $\mathcal{L}_{pairwise}^{Z|Y}$ can be effective when the second objective Z is consistent with the primary objective Y, and, otherwise, the loss function will be masked. And it is noted that samples i and j are symmetrical, so we only consider the greater relation.

In the hotel scenario, y (corresponding Y) is the primary objective and z (corresponding Z) is the secondary objective, so the loss function can be defined as follows:

$$\begin{aligned} \mathcal{L}_{pairwise}^{CTRCVR} &= \sum_{i=1}^{N}\sum_{j=1}^{N} \ell(\hat{s}_i - \hat{s}_j, \mathbb{I}(y_i > y_j)), \\ \mathcal{L}_{pairwise}^{HRS|CTRCVR} &= \sum_{i=1}^{N}\sum_{j=1}^{N} \mathbb{I}(y_i \geq y_j)\ell(\hat{s}_i - \hat{s}_j, \mathbb{I}(z_i > z_j)), \end{aligned} \tag{9}$$

where $\mathcal{L}_{pairwise}^{CTRCVR}$ (corresponding $\mathcal{L}_{pairwise}^{Y}$) a pairwise ranking loss, which aims to enlarge the ranking score gap between i and j when label level y_i is larger than y_j. $\mathcal{L}_{pairwise}^{HRS|CTRCVR}$ (corresponding $\mathcal{L}_{pairwise}^{Z|Y}$) is also a pairwise ranking loss for label z, but we add an extensive stratified constraint $\mathbb{I}(y_i \geq y_j)$ to ensure the loss is only effective when label level y_i is larger or equal than y_j.

We follow the ESMM pointwise loss function [24] as the basic loss function:

$$\mathcal{L}_{pointwise}^{ESMM} = \sum_{k=1}^{N} \ell(\hat{s}_k^{CTR}, \mathbb{I}(y_k > 0)) + \sum_{k=1}^{N} \ell(\hat{s}_k, \mathbb{I}(y_k = 2)), \tag{10}$$

where $\hat{s}_k^{CTR}$ denotes the predicted CTR score of sample k. And training loss function $\mathcal{L}$ consists of three parts: $\mathcal{L}_{pointwise}^{ESMM}$, $\mathcal{L}_{pairwise}^{CTRCVR}$, and $\mathcal{L}_{pairwise}^{HRS|CTRCVR}$:

$$\mathcal{L} = \mathcal{L}_{pointwise}^{ESMM} + \lambda_1 \mathcal{L}_{pairwise}^{CTRCVR} + \lambda_2 \mathcal{L}_{pairwise}^{HRS|CTRCVR}, \tag{11}$$

where λ_1 and λ_2 are hyper-parameters that balance the above three loss functions. We empirically explore their influence in Section 5.3.

5 EXPERIMENTS

To evaluate the effectiveness of our proposed method, we compare it with state-of-the-art methods and report our experimental results and corresponding analysis in this section. In order to choose the appropriate hyper-parameters, we explain the detailed process of choosing λ_1 and λ_2. Finally, an online A/B test is conducted in the hotel S&R system on Fliggy.

5.1 Dataset and Experimental Settings

Dataset Descriptions The offline hotel S&R dataset is generated based on the consumer logs collected from the hotel S&R system on Fliggy. As illustrated in Figure 2, each sample in this dataset is based on the impression of each consumer's search result. We take the clicking and ordering samples as positive samples, while others as negative samples. The features of these samples primarily contain three aspects of information, namely consumers, hotels, and search keywords. The consumer features include basic consumer attributes, long-term and short-term hotel preferences, and the distance between the consumer and the hotel. The hotel features include hotel basic attributes, real-time prices and inventories, historical behavior statistics, and factors of hotel service quality. The keyword features include searching scene types, the distance between searching location and candidate hotels, the semantic similarity between search words and candidate hotels, etc. The offline hotel S&R dataset statistics are presented in Table 4. The dataset contains 300 million training samples and 14 million test samples, which are partitioned by time.

Experimental Settings MERIT is deployed in TensorFlow[3]. We employ a grid search strategy to determine optimal hyper-parameters for our models. The ADAM optimizer is used to train all models, with a fixed learning rate of 0.001 and a batch size of 2048. To prevent overfitting, we apply L_2 regularization with a weight of 0.00001, as well as batch normalization with a dropout rate of 0.3. The balancing hyper-parameters λ_1 and λ_2 we use are 1.0 and 0.1. The embedding size for consumer and query features

[3]http://tensorflow.org/

Table 2: Comparison of multi-task learning models and monotonic networks on the offline hotel S&R dataset. Results of CTR (Click-Through Rate), CVR (Conversion Rate), and CTCVR (post-view Click-Through&Conversion Rate) are presented. The best results of all methods are indicated in bold, while the second best results are indicated in underlined. The Gain means the AUC improvement of MERIT+MSPL compared with DNN.

Models	AUC			GAUC		
	CTR AUC	CVR AUC	CTCVR AUC	CTR GAUC	CVR GAUC	CTCVR GAUC
DNN (Base)	0.7454	0.8870	0.8966	0.7737	0.8106	0.8201
Shared Bottom	0.7453	0.8767	**0.8979**	0.7727	0.8014	0.8210
MMoE	0.7463	0.8787	0.8975	0.7724	0.8024	**0.8216**
CGC	**0.7464**	**0.8885**	0.8972	0.7720	0.7978	0.8206
MERIT	0.7453	0.8882	0.8969	0.7734	0.8093	0.8195
MERIT (Point-wise Monotonic Loss)	0.7451	**0.8885**	0.8970	0.7735	**0.8114**	0.8206
MERIT (MIN-MAX Network)	0.7455	0.8880	0.8970	**0.7740**	0.8098	0.8202
MERIT+MPL	0.7452	0.8820	0.8940	0.7737	0.8039	0.8178
MERIT+MSPL	0.7452	0.8824	0.8943	0.7739	0.8049	0.8188
Gain	-0.0002	-0.0046	-0.0023	+0.0002	-0.0057	-0.0013

Table 3: Comparison of multi-task learning models and monotonic networks on the offline hotel S&R dataset. The HRS ranking results are presented. The best results of all methods are indicated in bold, while the second best results are indicated in underlined. The Gain means the NDCG improvement of MERIT+MSPL compared with DNN and $*$ means p-value < 0.001 in significance tests compared to the best baseline.

Models	NDCG			wNDCG		
	NDCG@5	NDCG@10	NDCG@20	wNDCG@5	wNDCG@10	wNDCG@20
DNN (Base)	0.8468	0.8804	0.9067	0.7529	0.7844	0.8228
Shared Bottom	0.8458	0.8793	0.9057	0.7523	0.7833	0.8213
MMoE	0.8462	0.8794	0.9057	0.7526	0.7833	0.8210
CGC	0.8446	0.8781	0.9047	0.7500	0.7811	0.8193
MERIT	0.8525	0.8848	0.9102	0.7615	0.7921	0.8293
MERIT (Point-wise Monotonic Loss)	0.8485	0.8818	0.9078	0.7555	0.7869	0.8248
MERIT (MIN-MAX Network)	0.8499	0.8829	0.9086	0.7579	0.7890	0.8267
MERIT+MPL	0.8708	0.9011	0.9235	0.7838	0.8156	0.8516
MERIT+MSPL	**0.8718***	**0.9020***	**0.9243***	**0.7850***	**0.8171***	**0.8531***
Gain	+0.0250	+0.0216	+0.0176	+0.0321	+0.0327	+0.0303

is set at 4, while the embedding size for context and item features is set at 8. The CTR and CVR Towers are implemented via a three-layer fully connected network with sizes of 256, 128, and 64. For multi-task learning models, a three-tower MLP is constructed to predict CTR, CVR, and HRS, with the gated network implemented via a softmax layer. MMoE utilizes 8 expert networks, while CGC employs 1 shared expert network and 1 specific expert network. The MIN-MAX network uses 10 groups of 10 linear functions.

Table 4: Statistics of the offline hotel S&R dataset.

Category	#User	#Hotel	#Impression
Number	5,614,476	775,190	354,319,050
Category	#Click	#Conversion	
Number	31,326,730	2,317,167	

Baselines To evaluate the effectiveness and superiority of our methodology, we compare our method with state-of-the-art multi-task learning models. And we also choose some monotonic networks to compare the ranking results. These baselines are as follows:

- **DNN** [11]: We only use a Multi-Layer Perceptron to construct the CTR Tower and CVR Tower, and input features are the same as MERIT.
- **Shared Bottom** [4]: The Shared Bottom model uses the unified bottom layer and different towers for all the tasks. It aims to utilize a shared bottom layer to learn correlations between different tasks.
- **MMoE** [23]: The MMoE method with multiple gate networks is designed to control the expert networks for different tasks and relieve the negative transfer problem.
- **CGC** [33]: Customized Gate Control (CGC) with an Expert-Bottom layer intends to mitigate the seesaw phenomenon [33] via task-specific experts and task-shared experts. For

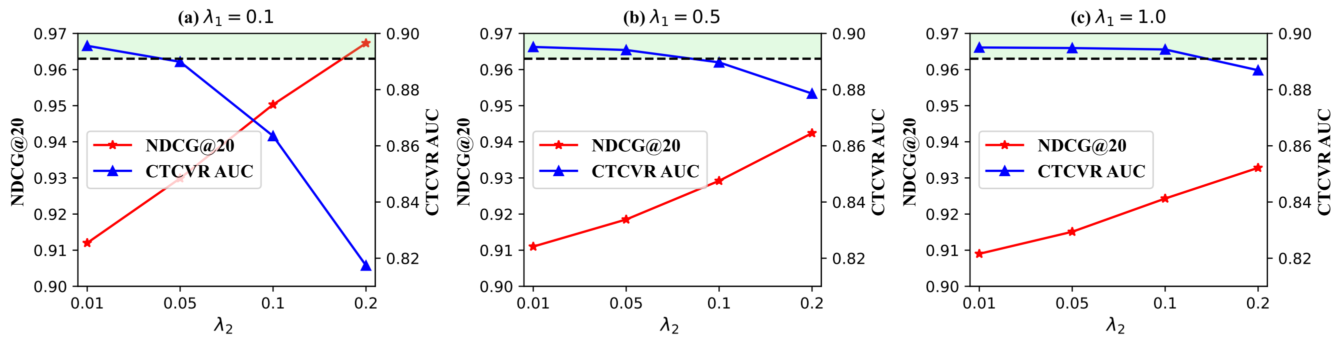

Figure 5: The NDCG@20 of HRS ranking and CTCVR AUC for different hyper-parameters λ_1 and λ_2. The dashed line indicates the lower bound of CTCVR AUC we can tolerate (From the online experiment, we consider that a 0.005 decline of CTCVR AUC is acceptable, so we set the lower bound of CTCVR AUC as 0.892 with the best 0.897.), and we choose λ_1 and λ_2 from points in the light green area.

a fair comparison, we do not take the progressive layered designing.

- **MERIT (MIN-MAX Network)** [9]: It follows the MERIT structure and replaces the Merchant Tower with the MIN-MAX Network. The MIN-MAX Network is a three-layer structure with max-pooling and min-pooling.
- **MERIT (Point-wise Monotonic Loss)** [13]: It follows the MERIT structure and replaces the Merchant Tower with a Multi-Layer Perceptron and Point-wise Monotonic Loss. Point-wise Monotonic Loss is the loss function penalizing the negative gradients.
- **MERIT+MPL**: It follows the MERIT structure and we add Multi-objective Pairwise Loss.
- **MERIT+MSPL**: It follows the MERIT structure and we add Multi-objective Stratified Pairwise Loss as defined in Equation (9).

Evaluation Metrics Our approach is to strike a balance between consumer and merchant engagement. Specifically, our model focuses on binary classification of whether a hotel has been clicked and ordered by consumers, while also ranking hotels according to HRS for hotels. To evaluate our method against baselines, we employ two types of metrics.:

- **AUC**: *Area Under Curve* (AUC) is a widely used metric to measure the ranking result for the whole model. It indicates the probability that positive samples rank higher than negative samples.
- **GAUC**: *Group Area Under Curve* (GAUC) partitions test samples into groups via the consumer id, and AUC is calculated by each group. It is defined as follows:

$$\mathrm{GAUC} = \frac{\sum_u w_u \times \mathrm{AUC}_u}{\sum_u w_u}, \tag{12}$$

where w_u is the sample size of consumer u.

- **NDCG@K**: *Normalized Discounted Cumulative Gain* (NDCG) indicates the ratio between current ranking performance and the ideal ranking performance. It considers the ranking position in terms of relation score. In this experiment, the ranking performance totally via the HRS label will be the ideal ranking performance. We choose Top-K test samples as the evaluated group.
- **wNDCG@K**: *weighted Normalized Discounted Cumulative Gain* (wNDCG) partitions the test samples into each group via the session id, and for each session, we calculate the NDCG. Its mathematical formulation is as follows:

$$\mathrm{wNDCG@K} = \frac{\sum_s w_s \times \mathrm{NDCG@K}_s}{\sum_s w_s}, \tag{13}$$

where w_s is the length of session s.

5.2 Offline Comparison Results

In this subsection, we compare our proposed model with several multi-task learning models and monotonic networks on the test set of the offline hotel S&R dataset. The AUC and GAUC results of predicting consumer feedback are reported in Table 2, while NDCG and wNDCG results of ranking HRS label are illustrated in Table 3. Based on our analysis of predicting consumer implicit feedback, we made the following observations:

- Compared with the DNN model, multi-task learning models can improve part of the predicted scores. Specifically, the CGC model achieves the best performance compared with the DNN model and other multi-task learning models, because it follows the different designs of expert networks.
- The MERIT model can get nearly equal results compared with the base model. Especially, the CVR AUC improves by 0.0012, 0.0015, and 0.0010 for different monotonic models compared with the base model, which indicates the monotonicity for X^s can better predict consumers' purchase behaviors. But the HRS is weakly related to the click behaviors, so the improvement is not noticeable.
- When we add MSPL into the MERIT model, some metrics will decrease. But this decrease is modest and tolerable. The objective of our method is to optimize the merchant demand, so the minor decline in consumer demand is unavoidable and reasonable. Also, we can recognize that the decline is primarily due to CVR ranking results.

In order to obtain the ranking results of hotel service quality, we compare the listed hotel HRS level for different methods. And we summarize our assessments of hotel service quality ranking and draw the following findings:

- When compared to the DNN model, multi-task learning models do not take hotel service quality into account, hence the NDCG shows no discernible improvement. In addition, the relation between consumers and merchants may be negative and insignificant, so the NDCG of multi-task learning models falls dramatically. Specifically, for CGC, which has the best performance in multi-task learning models, the NDCG@5 declines by 0.0022.
- The monotone structure has the potential to improve hotel ranking results. In comparison to the base model, three monotonic structures (MERIT, MIN-MAX Net, and Pointwise Monotonic Loss) enhance NDCG@5 by 0.0057, 0.0031, and 0.0017. Because of the extensive batch norm component, our proposed MERIT approach has the best performance.
- The Multi-objective Pairwise Loss is capable of strengthening the training objective on the HRS. Specifically, MERIT+MPL and MERIT+MSPL improve on NDCG@5 by 0.0240 and 0.0250, respectively. The MSPL can mitigate the conflict problem of multi-objective optimization and hence outperform MPL in the ranking of HRS.

Based on the experimental observations presented above, we can summarize that our proposed methods have the following advantages: 1) Monotonic structures can better learn the correlation between X^s and the hotel ranking score. 2) MSPL can better seek a balance between the ranking results based on user implicit feedback and merchant service quality.

5.3 Influence of Hyper-parameters λ_1 and λ_2

The hyper-parameters λ_1 and λ_2 aim to balance CTRCVR and HRS ranking results. In order to explore the influence of these two hyper-parameters, we choose the CTCVR AUC and NDCG@20 as the measuring metrics. The range of λ_1 is in $\{0.1, 0.5, 1.0\}$ and the range of λ_2 is in $\{0.01, 0.05, 0.1, 0.2\}$. The larger λ_1 will enlarge the influence of $\mathcal{L}_{pairwise}^{CTRCVR}$ while the larger λ_2 will enlarge the influence of $\mathcal{L}_{pairwise}^{HRS|CTRCVR}$. With the growth of λ_2, the result of CTCVR AUC gradually increases while NDCG@20 declines as shown in Figure 5. The influence of λ_1 is the opposite. From the online experiment, we consider that a 0.005 decline of CTCVR AUC is acceptable, so we set the lower bound of CTCVR AUC as 0.892 (The best CTCVR AUC is 0.897 as shown in Table 2.). In order to choose the best hyper-parameters, we choose the points in the light green area (above the dashed line) and then we choose the point with the largest NDCG@20. Accordingly, we set $\lambda_1 = 1.0$ and $\lambda_2 = 0.1$ in Figure 5(c) as the appropriate hyper-parameters."

5.4 Online A/B Test

We conduct an online A/B test in the hotel S&R system on Fliggy within two weeks of July 2022. Table 5 illustrates the online performance that our method is compared with the baseline method (DNN). It is apparent that our method is superior to the baseline method for both consumers and merchants. The CVR has increased by 0.54%, and the Sales Volume has increased by 0.97% (Given the substantial sales volume of the OTP, with hotel price in the hundreds of yuan, a 0.97% increase in sales volume indicates a nearly one million yuan increase in sales volume.). The mean HRS score of displayed hotels has risen by 3.02%, indicating that displayed merchant service quality has been improved. Our proposed method has been deployed online and brought remarkable OTP profit growth.

Table 5: Online performance of the proposed MERIT model.

	CVR	Sales Volume	HRS Score
Lift	+0.54%	+0.97%	+3.02%

We selected several major cities in China to compare the variations in the A/B test across different scenarios. And we evaluated two significant online metrics, *Imp.(%)* and *UCVR*, to assess the performance. *Imp.(%)* measures the lift ratio of the average displayed number for each hotel. *UCVR* measures the proportion of users with orders among all users. Figure 6(a) illustrates the *Imp.(%)* results for different HRS-level hotels in the online A/B test. With our proposed method, the results indicate that the HRS levels from 3.0 to 5.0 have an improvement of impression in comparison to the base model. Hotels with an HRS level greater than 3.0 can, for example, achieve significant impression gains in Hangzhou, whereas hotels with an HRS level greater than 3.5 can achieve comparable impression gains in Chengdu. Additionally, Hangzhou's biggest growth is 19.31% at HRS level 5.0, whereas Chengdu's largest growth is 32.66% at HRS level 5.0. Similarly, in Figure 6(b) we evaluate how *UCVR* has changed for hotels with various HRS levels, and the results show that our proposed method has reallocated resources to hotels with high HRS levels. For example, hotels at the HRS level 5.0 have boosted their *UCVR*s by 2.88% in Hangzhou, 4.93% in Chengdu, and 0.40% in all cities. Therefore, we conclude that our proposed method has remarkable impression improvement on hotels with high HRS scores, which can address the former two challenges on OTPs: Matthew Effect in the consumer feedback-loop and the unclear relation between hotel service quality and performance.

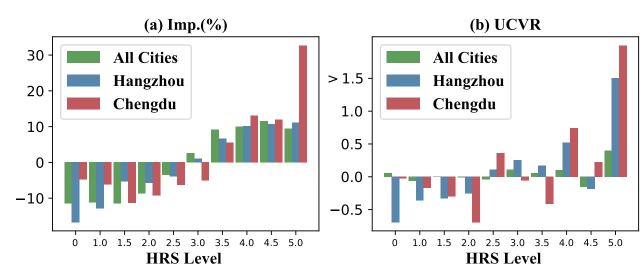

Figure 6: The online performance of MERIT model for hotels with different HRS levels among cities. The HRS level 0 denotes that there are no consumer ratings for hotels

6 CONCLUSION

In this paper, we identify three main challenges in the scenario of the hotel S&R system on Online Travel Platforms (OTPs): Matthew

Effect in the consumer feedback-loop, unclear relation between hotel service quality and performance, and conflicts between platform revenue and consumer experience. A new Merchant Incentive Ranking Model representing the merchant-consumer relation for the hotel S&R system, namely MERIT, is proposed to address these three challenges. We define factors of hotel service quality, which represent the hotel service quality of the entire consumer behavior cycle, and propose an HRS metric as an evaluation of hotel service quality. Also, we introduce a monotonic structure into MERIT to provide clear relation between factors of hotel service quality and ranking performance. Finally, we propose a novel Multi-objective Stratified Pairwise Loss to mitigate conflicts between platform revenue and consumer experience. Extensive experiment findings on the offline hotel S&R dataset demonstrate the superiority of MERIT over existing state-of-the-art benchmarks in terms of NDCG, as well as the effectiveness of imposing monotonicity of merchant service quality. We also conduct an online A/B test and obtain a 3.02% improvement in the HRS score. MERIT has been deployed into Fliggy to serve tens of millions of consumers and hundreds of thousands of hotel merchants.

REFERENCES

[1] Norman P Archer and Shouhong Wang. 1993. Application of the back propagation neural network algorithm with monotonicity constraints for two-group classification problems. *Decision Sciences* 24, 1 (1993), 60–75.

[2] Oren Barkan and Noam Koenigstein. 2016. Item2vec: neural item embedding for collaborative filtering. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 1–6.

[3] K. Canini, A. Cotter, M. R. Gupta, M. Milani Fard, and J. Pfeifer. 2016. Fast and Flexible Monotonic Functions with Ensembles of Lattices. In *Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16)*. 2927–2935.

[4] Rich Caruana. 1997. Multitask learning. *Machine learning* 28, 1 (1997), 41–75.

[5] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2020. Bias and debias in recommender system: A survey and future directions. *arXiv preprint arXiv:2010.03240* (2020).

[6] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.

[7] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014).

[8] Guy W Cole and Sinead A Williamson. 2019. Avoiding resentment via monotonic fairness. *arXiv preprint arXiv:1909.01251* (2019).

[9] Hennie Daniels and Marina Velikova. 2010. Monotone and partially monotone neural networks. *IEEE Transactions on Neural Networks* 21, 6 (2010), 906–917.

[10] Ad J Feelders. 2000. Prior knowledge in economic applications of data mining. In *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, 395–400.

[11] Matt W Gardner and SR Dorling. 1998. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric environment* 32, 14-15 (1998), 2627–2636.

[12] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247* (2017).

[13] Akhil Gupta, Naman Shukla, Lavanya Marla, Arinbjörn Kolbeinsson, and Kartik Yellepeddi. 2019. How to incorporate monotonicity in deep networks while preserving flexibility? *arXiv preprint arXiv:1909.10662* (2019).

[14] Guy Hadash, Oren Sar Shalom, and Rita Osadchy. 2018. Rank and rate: multi-task learning for recommender systems. In *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018*, Sole Pera, Michael D. Ekstrand, Xavier Amatriain, and John O'Donovan (Eds.). ACM, 451–454.

[15] Malay Haldar, Mustafa Abdool, Prashant Ramanathan, Tao Xu, Shulin Yang, Huizhong Duan, Qing Zhang, Nick Barrow-Williams, Bradley C Turnbull, Brendan M Collins, et al. 2019. Applying deep learning to airbnb search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1927–1935.

[16] Malay Haldar, Prashant Ramanathan, Tyler Sax, Mustafa Abdool, Lanbo Zhang, Aamir Mansawala, Shulin Yang, Bradley Turnbull, and Junshuo Liao. 2020. Improving deep learning for airbnb search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2822–2830.

[17] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.

[18] Zai Huang, Mingyuan Tao, and Bufeng Zhang. 2021. Deep Inclusion Relation-aware Network for User Response Prediction at Fliggy. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 3059–3067.

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

[20] Shih-Yang Liu, Hsien-Hao Chen, Chih-Ming Chen, Ming-Feng Tsai, and Chuan-Ju Wang. 2022. IPR: Interaction-level Preference Ranking for Explicit feedback. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*. ACM, 1912–1916.

[21] Xingchao Liu, Xing Han, Na Zhang, and Qiang Liu. 2020. Certified monotonic neural networks. *Advances in Neural Information Processing Systems* 33 (2020), 15427–15438.

[22] Babak Loni, Roberto Pagano, Martha A. Larson, and Alan Hanjalic. 2016. Bayesian Personalized Ranking with Multi-Channel User Feedback. In *Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, September 15-19, 2016*. ACM, 361–364.

[23] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1930–1939.

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

[25] 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).

[26] An-phi Nguyen and María Rodríguez Martínez. 2019. Mononet: towards interpretable models by learning monotonic features. *arXiv preprint arXiv:1909.13611* (2019).

[27] 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.

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

[29] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).

[30] D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-François Crespo, and Dan Dennison. 2015. Hidden Technical Debt in Machine Learning Systems. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems*. 2503–2511.

[31] David Simchi-Levi, Rui Sun, Michelle Wu, and Ruihao Zhu. 2022. Calibrating Sales Forecast in a Pandemic Using Competitive Online Non-Parametric Regression. In *SSRN 3670264*.

[32] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations. In *Fourteenth ACM Conference on Recommender Systems*. 269–278.

[33] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations. In *Fourteenth ACM Conference on Recommender Systems*. 269–278.

[34] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *Proceedings of the ADKDD'17*. 1–7.

[35] Hong Wen, Jing Zhang, Yuan Wang, Fuyu Lv, Wentian Bao, Quan Lin, and Keping Yang. 2020. Entire space multi-task modeling via post-click behavior decomposition for conversion rate prediction. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 2377–2386.

[36] Han Wu, Hongzhe Zhang, Liangyue Li, Zulong Chen, Fanwei Zhu, and Xiao Fang. 2022. Cheaper Is Better: Exploring Price Competitiveness for Online Purchase Prediction. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 3399–3412.

[37] Kailun Wu, Weijie Bian, Zhangming Chan, Lejian Ren, Shiming Xiang, Shu-Guang Han, Hongbo Deng, and Bo Zheng. 2022. Adversarial Gradient Driven Exploration for Deep Click-Through Rate Prediction *(KDD '22)*. 2050–2058.

[38] Ruobing Xie, Cheng Ling, Yalong Wang, Rui Wang, Feng Xia, and Leyu Lin. 2020. Deep Feedback Network for Recommendation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, Christian Bessiere (Ed.). ijcai.org, 2519–2525.

[39] Jia Xu, Fei Xiong, Zulong Chen, Mingyuan Tao, Liangyue Li, and Quan Lu. 2022. G2NET: A General Geography-Aware Representation Network for Hotel Search Ranking. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4237–4247.

[40] Seungil You, David Ding, Kevin Canini, Jan Pfeifer, and Maya R. Gupta. 2017. Deep Lattice Networks and Partial Monotonic Functions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*. 2985–2993.

[41] Xiaokun Zhang, Bo Xu, Liang Yang, Chenliang Li, Fenglong Ma, Haifeng Liu, and Hongfei Lin. 2022. Price DOES Matter! Modeling Price and Interest Preferences in Session-based Recommendation. *arXiv preprint arXiv:2205.04181* (2022).

[42] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 5941–5948.

[43] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1059–1068.