Bagging-Expert Network for Multi-Task Learning: A Depolarization Solution in Multi-Gate Mixture-of-Experts

Gong-Duo Zhang*, **Ruiqing Chen, Qian Zhao, Zhengwei Wu, Fengyu Han, Huan-Yi Su, Ziqi Liu, Lihong Gu, Lin Zhou**

Ant Group, Hangzhou, China
{gongduo.zgd,chenruiqing.crq,zq317110}@antgroup.com, zhengwei_wu@yeah.net,
{hanfengyu.hfy,suhuanyi.shy,ziqiliu,lihong.glh,lin.zhou}@antgroup.com

Abstract

Multi-task learning (MTL) is widely utilized across a variety of real-world applications, including recommendation systems. For instance, in the field of e-commerce, MTL is commonly employed to simultaneously model click, conversion, and user dwelling time. Among a various of MTL models, the Multi-gate Mixture-of-Experts (MMoE) has gained significant popularity. However, MMoE suffers from the polarization issue during training, where the weights of certain experts tend to converge towards 0. To address this issue, we propose a novel method called **B**agging-**E**xpert **net**work (BEnet) for multi-task learning. BEnet effectively mitigates the problem of polarization and achieves excellent performance in multi-task learning. It incorporates a bagging layer and an attention mechanism to encourage experts focusing on diverse knowledge domains. Simultaneously, polarization is avoided as different experts execute respective duties and specialize in distinct domains. Experimental results on real-world datasets demonstrate that BEnet has strong robustness and outperforms other state-of-the-art (SOTA) MTL methods.

Introduction

In recent years, multi-task learning (MTL) has gained significant attention due to its potential to solve complex and diverse real-world industrial problems. For instance, e-commerce and mobile payment application developers often promote to motivate customer retention and consumption by allocating online coupons (Li et al. 2020a; Wetprasit, Cao, and Seow 2022; Huan et al. 2022; Zhang and Yang 2022; Fang et al. 2023). Typical model tasks for these coupon recommendation systems involve estimating click-through rate (CTR) and conversion rate (CVR) (Fang et al. 2024b,a; Zhao et al. 2024). CTR and CVR tasks are directly linked to the effectiveness of coupon recommendations, while there are indirect application-related tasks, such as daily active use (DAU) and user dwelling time on apps (TOA). These tasks are different but interconnected. For example, users who tend to click on certain coupons tend to redeem them which in return makes them more active in apps. Employing a multi-task model can learn commonalities among the tasks to improve the efficiency and accuracy of the recommendation systems (Li et al. 2020b; Huangfu et al. 2022).

*Corresponding author.

Multi-gate Mixture-of-Experts (MMoE) (Ma et al. 2018a) is a popular MTL model that has been widely used in various applications. To derive the final output, the model employs a gate mechanism that selectively combines the outputs of these experts. This integration process enables MMoE to effectively capture intricate patterns and dependencies within the data. Despite its advantages, MMoE suffers from the polarization problem(Zhao et al. 2019), where the weights of some experts tend to converge to 0 during the training process.

Polarization in gate network has been discussed in prior works (Ma et al. 2018a; Zhao et al. 2019; Ravaut, Joty, and Chen 2022). In paper (Zhao et al. 2019), authors highlight the polarization phenomenon in MMoE during distributed training, occurring with a probability of 20%. This phenomenon has the potential to negatively impact the performance of the gate networks, and subsequently, the overall performance of the model (Zhao et al. 2019). To address this issue, the authors propose a solution by incorporating drop-out into the gate networks. SummaReranker (Ravaut, Joty, and Chen 2022) also employs a drop-out strategy for the expert networks in MMoE model, with the dropout rate for experts set at 50%.

Polarization also occurs in MoE (Mixture-of-Experts), often caused by load imbalance. In cases of load imbalance, MoE-layers converge towards the state where only a few experts are frequently used (Zhou et al. 2022). This could cause both the redundancy of expert parameters and the undermining of training efficiency. Regularization approaches are proposed to encourage the experts to have balanced weights and workload during the training process (Shazeer et al. 2017; Nie et al. 2021; Dai et al. 2024). However, such approaches can bring about drop in performance of MoE models (Chen et al. 2024).

In this paper, we propose the Bagging-Expert network (BEnet), a novel method to effectively tackle polarization problem and enhance the performance of MMoE in MTL. BEnet incorporates a bagging layer to enable experts and gates to randomly generate differences and use attention layer in experts to filter out irrelevant information. By leveraging diverse feature sets, experts can develop unique areas of expertise, thereby ensuring their individual value and reducing the risk of polarization. Bootstrap aggregating(Breiman 1996) abbreviated as bagging, is an ensemble

learning method improves accuracy of classification. Random forest (Leo 2001; Ho 1995; Kam 1998) is the most well-known application of bagging. Building upon the concept of features bagging in random forests, we introduce a bagging layer by randomly selecting features for expert and gate networks

Our motivation is that addressing the issue of polarization necessitates augmenting the diversity among multiple experts. To achieve this objective, we introduce the technique of bagging, which effectively enhances such diversity. In comparison to alternative methods like dropout that introduce differentiation, our proposed method exhibits superior performance and more effectively mitigates polarization phenomenon. This is further supported by the subsequent experimental analysis in Section . The contribution of this paper is threefold:

- We define the problem of polarization through mathematical representation and explain the reason of polarization occurrence.
- We propose a novel BEnet model which effectively mitigates polarization problem and improves the model performance. Through bagging layer, our model generates different subgroups for experts and attention mechanism capture features importance to removing unrelated data.
- Finally, we conduct control experiments on real-world datasets. The experiments demonstrate how polarization affects training dynamics in multitask learning, how the bagging affects model performance and how BEnet tackles polarization problem.

In this paper, we precisely define and comprehensively describe the polarization phenomenon. Furthermore, we introduce a novel application of the bagging method traditionally used in decision trees, adapting it for neural networks and applying it effectively in an industrial context. In the experimental section , we analyze the impact of bagging on overall model performance and conduct a detailed sensitivity analysis of bagging hyperparameters.

The remainder of this paper is structured as follows. The Method section introduces general MMoE concepts, defines polarization, and details the BEnet model. The Experiment section presents offline experiment results and online A/B test results to confirm the effectiveness of BEnet.

Related Work

One-gate Mixture-of-Experts (OMoE) model is proposed with all tasks sharing one gate, while MMoE (Ma et al. 2018a) is multi-task model with multi-gate. Sub-Network Routing (SNR) (Ma et al. 2019) controls connections between sub-networks by binary random variables and applies Neural Architecture Search (NAS) (Zoph and Le 2017) to learn an optimal model structure for MTL. As an improvement over SNR, Progressive Layered Extraction (PLE) (Tang et al. 2020) introduces a progressive routing approach, separating task-sharing and task-specific parameters explicitly to avoid negative transfer and seesaw phenomenon in MTL.

MMoE (Ma et al. 2018a) mentions that the gate for satisfaction subtask is focused on a single expert. Expert dropout is used to avoid polarization (Ravaut, Joty, and Chen 2022). YouTube mentions the gate network stability and polarization issue of MMoE model (Zhao et al. 2019) . Authors mitigate polarization by applying a certain probability of setting utilization of experts to 0 and re-normalizing the softmax outputs.

Recently, the MoE architecture has been successfully adapted for Large Language Models (LLMs), resulting in remarkable performance enhancements. By leveraging MoE, LLMs can be trained and deployed more efficiently, thereby achieving significant improvements in computational resource utilization (Du et al. 2022; Reid et al. 2024; Raposo et al. 2024). MoE models also suffer from polarization problems mainly due to load imbalance (Zhou et al. 2022). The work in (Dai et al. 2024) highlights the load imbalance problem inherent in MoE, which entails two major drawbacks. Firstly, routing collapse occurs when the model always selects only a few experts, preventing other experts from sufficient training. Secondly, when experts are trained in a distributed manner, this load imbalance exacerbates computational bottlenecks.

Regularization approaches have been proposed to mitigate the issue of load imbalance in LLMs. For instance, tunable Gaussian noise is added in gate networks and a new term is introduce to the loss function. This regularization term effectively weakens polarization and encourages the experts to have more balanced weights across tasks, which improves the overall performance of the model. In (Nie et al. 2021), authors propose the dense-to-sparse gate that can automatically adjust the number of active experts to balance the load. Moreover, the balance loss is employed as a crucial auxiliary training objective, thereby promoting a more stable and effective learning process (Dai et al. 2024). Octavius (Chen et al. 2024) implements balance loss to mitigate load imbalance for instance-based gate routing and observes a drop in performance of the LoRA-MoE model they propose.

In summation, prior research has chiefly employed strategies associated with dropout to mitigate polarization, with limited success in improving generalization. Our empirical evaluation, as detailed in the experimental section of this work, reveals a notable disparity in performance, with bagging demonstrating superior effectiveness when compared to dropout.

Problem Formulation

In this section, mathematical formalization of the polarization phenomenon is presented and depolarization method BEnet is introduced.

The input to MMoE is denoted as $\boldsymbol{x} \in \mathbb{R}^{m \times d}$, where m is the number of samples, and d is the dimension of input features. MMoE model has K tasks, each consisting of a shared-bottom network represented by function f^k, and a task-specific tower network h^k, where $k \in \{1, 2, ..., K\}$. The shared-bottom network is connected to the input layer and the tower networks are built upon the output of the shared-bottom. Let y_k as the label of task k, the corresponding task-specific tower produces an individual output $\hat{y}_k$ for

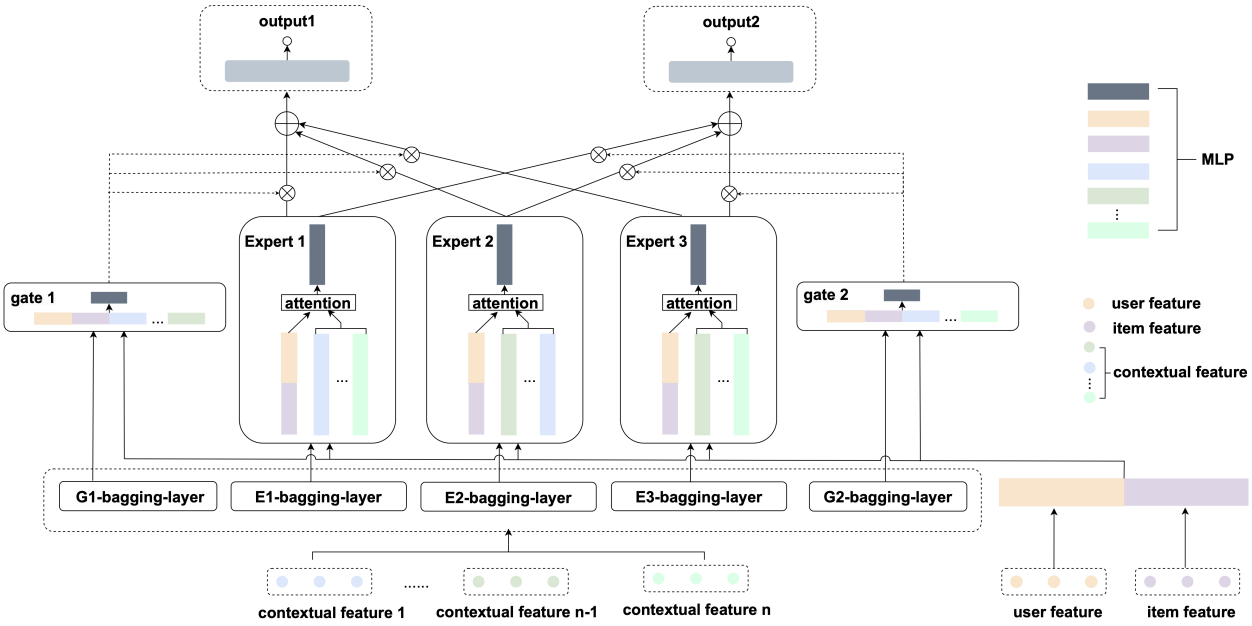


Figure 1: Illustration for the BEnet model structure

each task. Thus, the model can be formulated as follows for task k:

$$\hat{y}_k = h^k(f^k(\boldsymbol{x})). \tag{1}$$

For the $f^k(\boldsymbol{x})$, we have

$$f^k(\boldsymbol{x}) = \sum_{i=1}^{n} g_i^k(\boldsymbol{x}) f_i(\boldsymbol{x}), \tag{2}$$

$$g^k(\boldsymbol{x}) = \text{softmax}\left(W_k \boldsymbol{x}\right), \tag{3}$$

where $g^k(\boldsymbol{x}) = \left[g_1^k(\boldsymbol{x}), g_2^k(\boldsymbol{x}), \ldots, g_n^k(\boldsymbol{x})\right]$ is weight of n experts for the kth task and $\sum_{i=1}^{n} g_i^k(\boldsymbol{x}) = 1$. $g^k(\boldsymbol{x})$ represents the gate network output for the kth task, and $f_i(\boldsymbol{x})$ is the ith expert output. The loss function for MMoE is :

$$L_{MMoE} = \sum_{k=1}^{K} \alpha_k L_k(y_k, \hat{y}_k), \tag{4}$$

where $\hat{y}_k = h^k(f^k(\boldsymbol{x}))$ and α_k denotes the weight of task k, $L_k(y_k, \hat{y}_k)$ represents the loss function on task k. $\hat{y}_k$ is the model output on task k, and y_k is the label of task k.

Subsequently, the polarization problem could be defined as :

$$\forall \boldsymbol{x}, \exists i, \lim_{t \to \infty} g_i^k(\boldsymbol{x}) = 0. \tag{5}$$

where t denotes iteration step, when the number of experts is equal to 2, polarization problem is transformed into one expert with weight 1 and the other expert with weight 0.

Here, we elaborate on the occurrence and detrimental effects of polarization in MMoE. During the initial stages of training, all expert networks are updated in a random manner, and the gate weights are initially distributed between 0 and 1. However, as training progresses, a few expert networks may gain a relative advantage, leading to a disproportionate contribution towards the final estimated scores. In order to expedite the minimization of loss, these advantaged networks may continue to gain further advantages, resulting in faster learning rates and gate weights that become increasingly pronounced. As a consequence, the parameter updating become concentrated on these few networks, with the gate weights of other experts rapidly declining until they eventually collapse to 0.

As a result of polarization problem, most of neurons in the expert networks are no longer updated and the original intentions of the multi-expert network mechanisms are invalid, leading to a decline in model performance. Furthermore, these invalid neuron parameters continue to occupy machine memory, rendering the network redundant and leading to machine resources wasting.

Bagging Expert Network

In this section, we introduce the Bagging Expert Network (BEnet) designed to enhance recommendation system performance. The following subsections will elaborate on how our model processes contextual features and applies attention mechanisms within expert networks to improve its effectiveness.

Contextual Feature Processing

In real-world recommendation applications, the input features of industry recommendation systems are typically categorized into user features, item features, and contextual features. User and item features serve as the foundational representations, while contextual features capture user or item behaviors within specific sub-contexts. For example,

in a company's coupon recommendation system, contextual features might include user interactions and item statistical data across various scenarios, such as online shopping, transportation, takeaway services, and insurance. These contextual features provide valuable insights from different perspectives. To better integrate these contextual features and prevent polarization, we propose an enhanced model that employs bagging and attention mechanisms.[1]

An intuition is that if different experts rely on distinct feature sets, this heterogeneity will yield a unique value proposition for each expert, rendering them indispensable in their respective domains, and mitigating the likelihood of polarization. Inspired by bootstrap aggregating (Breiman 1996), a bagging layer is proposed to randomly select a portion of contextual features, making differences among experts and studying diverse information of contextual features. Suppose the input data x is d-dimensional vector drawn from the training data samples:

$$\boldsymbol{x} = \{\boldsymbol{x}^u, \boldsymbol{x}^m, \boldsymbol{x}^c\}, \tag{6}$$

where $\boldsymbol{x}^u$ is a vector of user feature, $\boldsymbol{x}^m$ represent item feature and $\boldsymbol{x}^c$ represents contextual feature vector, $d = |\boldsymbol{x}^u| + |\boldsymbol{x}^m| + |\boldsymbol{x}^c|$. Bagging layer randomly select a portion of contextual features. Denote $\boldsymbol{x}$ after bagging as $\boldsymbol{x}^b$, the feature sampling probability of bagging is p.

$$\boldsymbol{x}^b = \{\boldsymbol{x}^u, \boldsymbol{x}^m, \tilde{\boldsymbol{x}}^c\}, \tag{7}$$

$\tilde{\boldsymbol{x}}^c$ is a subgroup of $\boldsymbol{x}^c$. The feed-forward operation could be defined as :

$$\begin{aligned} \boldsymbol{r}^c &\sim \mathcal{B}\left(c, p\right), \\ \tilde{\boldsymbol{x}}^c &= \text{squeeze}(\boldsymbol{r}^c \circ \boldsymbol{x}^c). \end{aligned} \tag{8}$$

Here $\circ$ denotes an element-wise product. $\boldsymbol{r}^c \in \{0, 1\}^{1\times c}$ is a vector of independent Bernoulli random variables (1 with probability p and 0 with $1 - p$). In fact, we implement subgroup sampling proportion to ensure that each expert has an equal number of subgroups as input. $\boldsymbol{r}^c$ is sampled and multiplied element-wise with $\boldsymbol{x}^c$, then we removing zero in $\boldsymbol{r}^c \circ \boldsymbol{x}^c$ by squeeze operation. To maintain a stable network structure during training, bagging is performed only once, and the network structure is fixed afterward, similar to the approach used in random forests.

The output $\boldsymbol{x}^b$ is then used as input to the expert and gate networks in MMoE, where different inputs $\boldsymbol{x}_i^b$ of the ith expert produced by different Bernoulli samplings. Gate inputs are also generated from different bagging Bernoulli column sampling. $f_i(\boldsymbol{x}_i^b)$ is the output of the ith expert network corresponding to $\boldsymbol{x}_i^b$. The $\boldsymbol{x}_i^b$ tends to be different from each other with the feature columns sample ratio.

We group contextual features into sub-groups, such as online shopping, transportation, takeaway, and insurance. We then apply bagging based on these sub-groups. This approach enhances the diversity of the contextual features provided to the experts in the MMoE model. Both item features and user features are directly included in the expert features.

[1]Note that certain datasets do not include contextual features. Nevertheless, BEnet consistently performs excellently on such general cases which will be illustrated in the subsequent content.

As shown in Figure 1, the model comprises three experts and two task gates. Contextual feature groups are sampled into three expert-specific bagging features and two gate-specific bagging features. We then calculate an attention product between the fundamental (user and item) features and the bagging contextual features. The gate networks output softmax gates to assemble the experts with varying weights, enabling different tasks to leverage the experts differently. The results from the assembled experts are subsequently passed into the task-specific tower networks.

Before entering experts, bagging subgroups need to multiply the attention weight which describes importance of contextual features(Vaswani et al. 2017). This proposed approach aims to make experts focus more on specific domain knowledge and filter out noise by attention to preserve important information.

It is important to note that while many industrial datasets consist of user features, item features, and contextual features, there are many recommendation datasets that do not distinctly divide features into these categories. In such cases, we apply bagging and attention mechanism across all features without differentiating between categories. We show that BEnet still demonstrates its effectiveness on public datasets with general features in the subsequent experimental section.

Attention Mechanism in Expert

In order to learn different domain among various contextual features better, we utilize fundamental features (user and item features) to guide the attention mechanism towards the contextual features in the expert networks. Before entering experts, bagging subgroups need to multiply the attention weight which describes importance of contextual features(Vaswani et al. 2017). This proposed approach aims to make experts focus more on specific domain knowledge and filter out noise by attention to preserve important information.

Followed by previous definition, let $\boldsymbol{x}^u, \boldsymbol{x}^m, \boldsymbol{x}^c$ represent user features, item features, context features respectively. $\boldsymbol{x}^c$ is divided into sub-groups $\tilde{\boldsymbol{x}}^c$, where $\boldsymbol{x}^c = \tilde{\boldsymbol{x}}_1^c \cup \cdots \cup \tilde{\boldsymbol{x}}_g^c$ where g is the number of sub-groups. We define $\mathcal{E}$ as the set of experts, for experts set $\mathcal{E}$, the procedure of output is next:

- Bagging layers generate subgroups of the context features $\boldsymbol{x}^c = \{\cup_{i\in\mathcal{E}} \tilde{\boldsymbol{x}}_i^c\}$ that are fed into the experts via bagging operation.
- Target attention is applied to $\boldsymbol{x}^c$ using ($\boldsymbol{x}^u$, $\boldsymbol{x}^m$) as the input, then weights $\boldsymbol{w}_i$ for $i \in \mathcal{E}$ multiply $\tilde{\boldsymbol{x}}_i^c$ to get weighted bagging context feature $\boldsymbol{x}_i^w$.
- Concat user features $\boldsymbol{x}^u$, item features $\boldsymbol{x}^m$, and weighted bagging context feature $\cup_{i\in\mathcal{E}} \boldsymbol{x}_i^w$ as $\boldsymbol{x}^b$.
- Input $\boldsymbol{x}^b$ to fully connected layers and output the $f_i(\boldsymbol{x}^b)$.

Figure 2 illustrates an example that is used in practical applications where the y-axis represents different type items, while the x-axis represents the subgroups contextual features entering the experts.

To provide a clear description, we select two experts of them. This visualization displays the features of the sub-

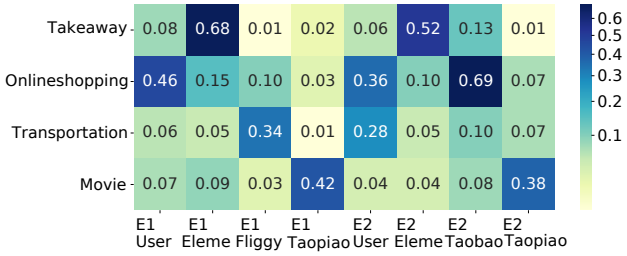


Figure 2: Items Attention to Bagging Sub-Groups

groups in different experts after bagging. The y-axis concludes four items, which are quite different type coupons in prior knowledge. Therefore, the model network needs to learn in a targeted manner. Figure 2 shows that each item has different dependency on subgroup features.

Our subgroup features consist of a total of five groups, namely user features, Eleme features[2], Fliggy features[3], Taobao features[4], and Taopiao features[5]. Among these, user features are used to compare the attention strength of our subgroup features.

In the presented information, we can observe that expert E1 and E2 have two shared subgroups of information (basic information), namely Eleme features and Taopiao features. It can be seen that the takeaway item exhibits a dominant focus on the Eleme subgroup, while online shopping concentrates on the Taobao subgroup. Similarly, transportation emphasizes the Fliggy subgroup, while movies primarily prioritize the Taopiao subgroup.

Furthermore, for the online shopping item, attention is primarily focused on Taobao features. The first expert does not possess any Taobao features, while the second expert does, indicating that the second expert's emphasis on learning Taobao features contributes significantly to the online shopping item. The first expert has an exclusive subgroup of features related to Fliggy, which implies that transportation-related items will be learned more effectively, allowing for a more focused and specialized approach in this domain.

In many recommendation tasks, users and items often exhibit significant variations. BEnet effectively capture the diverse characteristics of these multi-type items and learn users' multi-type interests in this situation. This is analogous to the human brain in biology, which comprises language modules and image modules for different cognitive tasks. Our model handles multi-type item via different experts that focus on the specific domain knowledge. This enables each expert to specialize in its respective domain, thereby avoiding the issue of expert homogeneity, which often leads to the polarization phenomenon. Through extensive experiments on real-world datasets later, we demonstrate the effectiveness of our proposed depolarization model in terms of improved recommendation performance and enhanced integration of diverse features.

[2]prominent online food delivery platform in China, connects consumers with restaurants through a mobile application, offering convenient ordering services

[3]popular Chinese online travel platform, provides comprehensive travel services, including flight and hotel bookings

[4]leading Chinese online marketplace, offers a wide range of products and services to consumers and businesses

[5]prominent online ticketing platform in China that allows users to conveniently purchase tickets for various entertainment events, such as movies and concerts

Experiment

To demonstrate BEnet's effectiveness, we conducted offline experiments using public real-world datasets and a proprietary dataset from Company-A's online environment. In addition to offline experiments, we also conduct A/B testing online under the real-world app environment.

Datasets with Multiple Optimization Objective

We conduct offline experiments using three real datasets, namely MultiMNIST, AliExpress and Company-A.

MultiMNIST The MultiMNIST dataset, introduced by (Sabour, Frosst, and Hinton 2017), serves as an extended multi-digit classification benchmark. The resulting dataset encompasses a training set comprising 60 million instances and a testing set containing 10 million instances.

AliExpress The AliExpress Searching System Dataset comprises data collected from real-world traffic logs of the search system in AliExpress, specifically in Spanish and French (Li et al. 2020b). The Spanish subset includes a training set with 22 million instances and 1.4 million unique users, alongside a test set presenting 9.3 million instances and 0.6 million users. Conversely, the French subset is comprised of a training set with 18 million instances and 1.2 million users, and a test set containing 8.8 million instances and 0.57 million users.

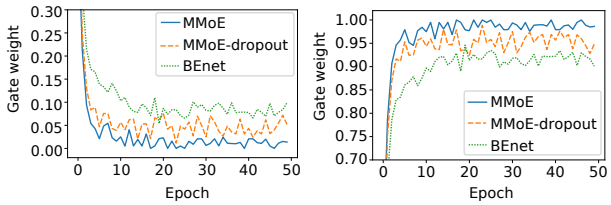
Company-A The Company-A dataset has been gathered from Company-A's promotion campaigns, which entail a series of extensive coupon distributions during major shopping events such as *double 11*, *double 12*, *Chinese New Year's WuFu Cards*, etc. These campaigns attract participation from several hundreds of millions of individuals and typically extend over multiple days. A randomized subset of data was meticulously extracted from various intervals of the online promotional campaigns. For the construction of the training set, data amassed from the initial phases of the promotional campaigns, amounting to 50 million instances, has been employed. Conversely, the testing set comprising 5 million instances, which were collected on the concluding day of the campaigns. The multi-task learning involves the estimation of CTR, CVR and user dwelling TOA.

Experiment Setting

The Company-A dataset comprises user features, item features, and contextual features. Following the methodology described in the previous section, we perform bagging on the contextual features. However, for MultiMNIST dataset and AliExpress dataset, features do not significantly differentiate into user features, item features and contextual features. Therefore, all features on these public dataset are applied bagging, and the attention mechanism shifts to self-attention across all features. Additionally, for MultiMNIST, which is

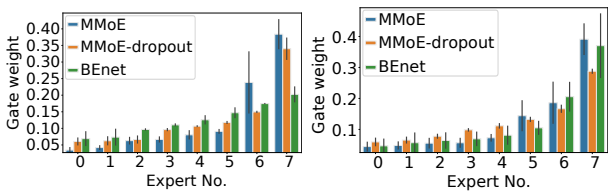
	MultiMNIST			AliExpress-French			AliExpress-Spain			Company-A			
Methods	Mean	ACC-L	ACC-R	Mean	AUC-CTR	AUC-CVR	Mean	AUC-CTR	AUC-CVR	Mean	AUC-CTR	AUC-CVR	AUC-TOA
PLE	0.9426	0.9514	0.9338	0.7981	0.7244	0.8717	0.8088	0.7280	0.8896	0.7819	0.7649	0.8776	0.7031
OMoE	0.9351	0.9451	0.9251	0.8019	0.7281	0.8757	0.8126	0.7319	0.8933	0.7804	0.7633	0.8758	0.7022
MMoE	0.9546	0.9614	0.9468	0.7977	0.7217	0.8736	0.8053	0.7236	0.8870	0.7821	0.7654	0.8779	0.7030
MMoE-dropout	0.9584	0.9658	0.9511	0.7992	0.7219	0.8764	0.8108	0.7286	0.8931	0.7806	0.7636	0.8767	0.7015
ESMM	0.9370	0.9481	0.9260	0.8086	0.7313	0.8859	0.8085	0.7303	0.8866	0.7818	0.7652	0.8778	0.7025
BEnet	**0.9629**†	**0.9693**†	**0.9566**†	**0.8103**†	**0.7343**†	**0.8864**†	**0.8143**†	**0.7339**†	**0.8947**†	**0.7848**†	**0.7682**†	**0.8809**†	**0.7054**†

Table 1: Offline results between BEnet and baselines on MultiMNIST, AliExpress, and Company-A datasets.



(a) MultiMNIST Expert 0 (b) MultiMNIST Expert 1

Figure 3: Depolarization capabilities for diffrenent methods on MultiMNIST dataset.



(a) AliExpress-French (b) AliExpress-Spain

Figure 4: Depolarization capabilities for diffrenent methods on AliExpress dataset.

an image dataset, we randomly mask a portion of the image as bagging action, and then input the rest of the image after bagging to the expert. We uniformly used a 64-core CPU with 128GB of memory, totaling 20 workers for training.

Efficiency Comparison on Baselines

For the efficiency comparison, we select the SOTA MOE network methods as baselines, including MMoE-dropout (Ravaut, Joty, and Chen 2022; Zhao et al. 2019), OMoE(Dai et al. 2024; Nie et al. 2021), and MMoE (Ma et al. 2018a; Su et al. 2024), as well as two classic multi-task models, namely PLE (Tang et al. 2020) and ESMM (Ma et al. 2018b). Note that the ESMM approach estimates the click-through conversion rate (CTCVR) by multiplying the CTR and the CVR. However, for non-CTCVR tasks in our experiments, such as the tasks in the MultiMNIST dataset and the TOA task in the Company-A dataset, there is no multiplication step involved. For other baselines and BEnet, the expert number of MultiMNIST is set 2, the expert numbers of AliExpress and Company-A are set 8. We also conduct experiments with other expert numbers and observed similar experimental results. Here, we only post the results of chosen parameters 2 and 8 in order to better observe the depolarization phenomenon. The bagging ratio of $\mathbf{r_c}$ for feature selection are fixed during training. We randomly choose 90 percent of each dataset as training data and the others as validation data. Besides, we use Area Under Curve (AUC) as the evaluation metrics for both AliExpress dataset and Company-A dataset since they are binary classification ranking datasets. In the case of Company-A, we divide user dwelling time into two labels $\{0, 1\}$ based on a threshold and simplify the task of Time-On-App (TOA) into binary classification task. For the MultiMNIST dataset, we use Accuracy (ACC) to measure the efficiency since it is a multi-classification dataset. A higher AUC or ACC score means that the method is more effective.

The effciency comparison results are shown in Table 1. The best result is bolded and the best result of baselines is underlined. We use "$\dagger$" to indicate that BEnet is significantly different from the best baseline based on paired t-tests at the significance of 0.01. ACC-L and ACC-R represent the two tasks of overlaying digits of the MultiMNIST dataset. Mean is the average of ACC-L and ACC-R. For AliExpress and Company-A, Mean is the average of the multi tasks' AUC. It can be found that BEnet achieves the best performance in terms of ACC and AUC on these three datasets. The experimental results demonstrate that BEnet effectively improves the performance of multi-task learning.

Depolarization Comparison of Different Methods

In this section, we conduct experiments to compare the depolarization capabilities for different methods. We compare BEnet, MMoE-dropout (Zhao et al. 2019) and original MMoE without depolarization. The MultiMNIST results are shown in Figure 3. The horizontal axis denotes the training epoch of each method while the vertical axis denotes the gate weights of the two experts. We calculate the average weights of the two tasks and report the results. It can be found that, across all methods, expert 1 obtains a greater weight than expert 0. However, the difference in convergence weight between the two experts of BEnet is smaller compared to the other methods.

Additionally, we present the depolarization of AliExpress. Given the high number of experts in AliExpress, we did not plot the convergence graphs of all 8 weights. Instead, we report the weight convergence results in Figure 4. The vertical axis denotes the weight convergence results of the 8 experts, arranged in ascending order based on the average

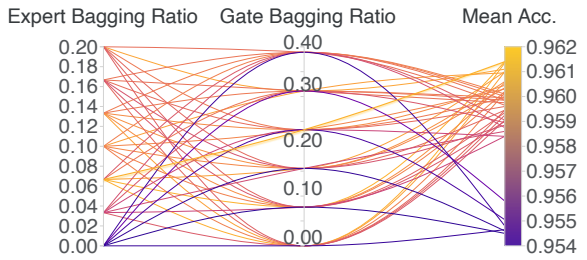


Figure 5: The results demonstrate the effect of altering the bagging ratios on the performance of the MultiMNIST dataset.

value of the two tasks. The horizontal axis denotes the 8 experts. We can find that for AliExpress-French, BEnet exhibits the smallest difference in convergence weight, while for AliExpress-Spain, MMoE-dropout exhibits the smallest difference. These results suggest that both bagging and dropout can mitigate polarization, with their efficacy dependent on the nature of the dataset. Nevertheless, in terms of performance metrics such as ACC and AUC, BEnet achieves the best results.

Ablation of Bagging

In this section, we executed a series of ablation studies to evaluate the contributory efficacy of the bagging network and to elucidate the beneficial role of depolarization. In the previous section, we employ a $Bernoulli\,(p)$ probability distribution to modulate the bagging proportion, indicative of the sampling fraction for the sub-groups. To understand how different bagging ratios affect our model's performance, we execute a battery of experiments on the MultiMNIST dataset by varying the $Bernoulli\,(p)$ parameter. The accuracy (ACC) outcomes of these experiments are depicted in Figure 5. In the presented figure, the parameters in the first column, labeled *Expert Bagging Ratio*, represent the proportion of features discarded in the expert network. Concurrently, the parameters in the second column, *Gate Bagging Ratio*, indicate the proportion of features discarded in the gating network. The third column elucidates the resultant model accuracy (ACC) contingent upon the parameter configurations specified in the preceding two columns. A connecting line is employed to visually associate the parameter pairs from the first two columns with their corresponding ACC outcome in the third column. It should be noted that the scenario wherein both the Expert Bagging Ratio and Gate Bagging Ratio are configured to 0 signifies a complete absence of bagging within the network topology, thereby permitting the manifestation of polarization.

In our empirical analysis, performance improvements are denoted by yellow lines while degraded performance outcomes are depicted by purple lines. Pertinent to our findings, the optimal range (yellow lines) for expert bagging ratio is approximately 0.06, whereas for the gate bagging ratio, it is proximate to 0.23. Scenarios characterized by the absence of expert bagging (i.e., when *Expert Bagging Ratio* equals 0 and *Gate Bagging Ratio* equals 0) yield poor accuracy (ACC) metrics, as illustrated by the purple lines. The empirical evidence suggests that the omission of bagging strategies and the subsequent emergence of polarization effects are detrimental to model efficacy. To achieve optimal system performance, meticulous fine-tuning of the bagging ratio parameters for both experts and gates is essential.

Methods	exposure count	send count	use count	send rate	use rate
Online Baseline	5,690,847	154,920	10,026	2.72%	6.47%
BEnet	5,689,901	168,022	11,931	**2.95%** (**+8.5%**)	**7.10%** (**+9.3%**)

Table 2: Online results between BEnet and online running baseline.

Online A/B Test

To evaluate the performance of BEnet in real-world environment, we conducted an A/B test on the real *double 11 Promotion Campaign*. This promotional campaign involved sending coupons to users, aiming to encourage them to use the Company-A app. We allocated 10 percent of the real online traffic flow to conduct the A/B test. After running the test, we collected data on coupon exposure count, coupon send count, and coupon use count for different experimental groups. The exposure count represents the number of users who were exposed to the coupons, while the send count represents the number of users who clicked on the coupons and received them. The use count represents the number of coupons that were actually used by users. Our objectives were to increase both the send count and the use count. The online comparison results are presented in Table 2. The use rate is calculated by dividing the use count by the send count, while the send rate is calculated by dividing the send count by the exposure count. The results show that BEnet outperforms the online running baseline with a notable improvement of $+8.5\%$ on the send rate and $+9.3\%$ on the use rate. The online baseline model is ESMM, which was engineered by pioneering researchers and is providing services to app users. Our BEnet has successfully cleared the A/B testing phase, demonstrably outperforming the benchmark and subsequently superseding it in application.

Conclusion

This paper focuses on the significant multi-task learning challenges in industrial applications, with a specific emphasis on addressing the polarization phenomena of MMoE. In this study, we propose a novel depolarization method for MMoE that effectively mitigates polarization and enhances model performance in complex industrial MTL scenarios. Our proposed method, referred to as BEnet, facilitates balanced weight distributions among experts by integrating a bagging layer and an attention mechanism. Both the offline experimental results and online A/B test results demonstrate the effectiveness of BEnet. Moreover, our solution has been successfully deployed in numerous industrial coupon recommendation scenarios at Company-A, further validating its practical applicability and real-world effectiveness.

References

Breiman, L. 1996. Bagging Predictors. *Mach. Learn.*, 24(2): 123–140.

Chen, Z.; Wang, Z.; Wang, Z.; Liu, H.; Yin, Z.; Liu, S.; Sheng, L.; Ouyang, W.; and Shao, J. 2024. Octavius: Mitigating Task Interference in MLLMs via LoRA-MoE. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Dai, D.; Deng, C.; Zhao, C.; Xu, R. X.; Gao, H.; Chen, D.; Li, J.; Zeng, W.; Yu, X.; Wu, Y.; Xie, Z.; Li, Y. K.; Huang, P.; Luo, F.; Ruan, C.; Sui, Z.; and Liang, W. 2024. DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models. *CoRR*, abs/2401.06066.

Du, N.; Huang, Y.; Dai, A. M.; Tong, S.; Lepikhin, D.; Xu, Y.; Krikun, M.; Zhou, Y.; Yu, A. W.; Firat, O.; Zoph, B.; Fedus, L.; Bosma, M. P.; Zhou, Z.; Wang, T.; Wang, Y. E.; Webster, K.; Pellat, M.; Robinson, K.; Meier-Hellstern, K. S.; Duke, T.; Dixon, L.; Zhang, K.; Le, Q. V.; Wu, Y.; Chen, Z.; and Cui, C. 2022. GLaM: Efficient Scaling of Language Models with Mixture-of-Experts. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvári, C.; Niu, G.; and Sabato, S., eds., *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, 5547–5569. PMLR.

Fang, J.; Cui, Q.; Zhang, G.; Tang, C.; Gu, L.; Li, L.; Gu, J.; Zhou, J.; and Wu, F. 2023. Alleviating Matching Bias in Marketing Recommendations. In Chen, H.; Duh, W. E.; Huang, H.; Kato, M. P.; Mothe, J.; and Poblete, B., eds., *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, 3359–3363. ACM.

Fang, J.; Zhang, G.; Cui, Q.; Gu, L.; Li, L.; Gu, J.; and Zhou, J. 2024a. Counterfactual Data Augmentation for Debiased Coupon Recommendations Based on Potential Knowledge. In Chua, T.; Ngo, C.; Lee, R. K.; Kumar, R.; and Lauw, H. W., eds., *Companion Proceedings of the ACM on Web Conference 2024, WWW 2024, Singapore, Singapore, May 13-17, 2024*, 93–102. ACM.

Fang, J.; Zhang, G.; Cui, Q.; Tang, C.; Gu, L.; Li, L.; Gu, J.; and Zhou, J. 2024b. Backdoor Adjustment via Group Adaptation for Debiased Coupon Recommendations. In Wooldridge, M. J.; Dy, J. G.; and Natarajan, S., eds., *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, 11944–11952. AAAI Press.

Ho, T. K. 1995. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, 278–282. IEEE.

Huan, Z.; Zhang, G.; Zhang, X.; Zhou, J.; Wu, Q.; Gu, L.; Gu, J.; He, Y.; Zhu, Y.; and Mo, L. 2022. An Industrial Framework for Cold-Start Recommendation in Zero-Shot Scenarios. In Amigó, E.; Castells, P.; Gonzalo, J.; Carterette, B.; Culpepper, J. S.; and Kazai, G., eds., *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, 3403–3407. ACM.

Huangfu, Z.; Zhang, G.; Wu, Z.; Wu, Q.; Zhang, Z.; Gu, L.; Zhou, J.; and Gu, J. 2022. A Multi-Task Learning Approach for Delayed Feedback Modeling. In Laforest, F.; Troncy, R.; Simperl, E.; Agarwal, D.; Gionis, A.; Herman, I.; and Médini, L., eds., *Companion of The Web Conference 2022, Virtual Event / Lyon, France, April 25 - 29, 2022*, 116–120. ACM.

Kam, H. T. 1998. The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence*, 20(8): 832–844.

Leo, B. 2001. Random forests. *Machine learning*, 45: 5–32.

Li, C.; Chu, M.; Zhou, C.; and Zhao, L. 2020a. Two-period discount pricing strategies for an e-commerce platform with strategic consumers. *Comput. Ind. Eng.*, 147: 106640.

Li, P.; Li, R.; Da, Q.; Zeng, A.; and Zhang, L. 2020b. Improving Multi-Scenario Learning to Rank in E-commerce by Exploiting Task Relationships in the Label Space. In d'Aquin, M.; Dietze, S.; Hauff, C.; Curry, E.; and Cudré-Mauroux, P., eds., *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, 2605–2612. ACM.

Ma, J.; Zhao, Z.; Chen, J.; Li, A.; Hong, L.; and Chi, E. H. 2019. SNR: Sub-Network Routing for Flexible Parameter Sharing in Multi-Task Learning. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 216–223. AAAI Press.

Ma, J.; Zhao, Z.; Yi, X.; Chen, J.; Hong, L.; and Chi, E. H. 2018a. Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. In Guo, Y.; and Farooq, F., eds., *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, 1930–1939. ACM.

Ma, X.; Zhao, L.; Huang, G.; Wang, Z.; Hu, Z.; Zhu, X.; and Gai, K. 2018b. Entire Space Multi-Task Model: An Effective Approach for Estimating Post-Click Conversion Rate. In Collins-Thompson, K.; Mei, Q.; Davison, B. D.; Liu, Y.; and Yilmaz, E., eds., *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, 1137–1140. ACM.

Nie, X.; Cao, S.; Miao, X.; Ma, L.; Xue, J.; Miao, Y.; Yang, Z.; Yang, Z.; and Cui, B. 2021. Dense-to-Sparse Gate for Mixture-of-Experts. *CoRR*, abs/2112.14397.

Raposo, D.; Ritter, S.; Richards, B. A.; Lillicrap, T. P.; Humphreys, P. C.; and Santoro, A. 2024. Mixture-of-

Depths: Dynamically allocating compute in transformer-based language models. *CoRR*, abs/2404.02258.

Ravaut, M.; Joty, S. R.; and Chen, N. F. 2022. SummaReranker: A Multi-Task Mixture-of-Experts Re-ranking Framework for Abstractive Summarization. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, 4504–4524. Association for Computational Linguistics.

Reid, M.; Savinov, N.; Teplyashin, D.; Lepikhin, D.; Lillicrap, T. P.; Alayrac, J.; Soricut, R.; Lazaridou, A.; Firat, O.; Schrittwieser, J.; Antonoglou, I.; Anil, R.; Borgeaud, S.; Dai, A. M.; Millican, K.; Dyer, E.; Glaese, M.; Sottiaux, T.; Lee, B.; Viola, F.; Reynolds, M.; Xu, Y.; Molloy, J.; Chen, J.; Isard, M.; Barham, P.; Hennigan, T.; McIlroy, R.; Johnson, M.; Schalkwyk, J.; Collins, E.; Rutherford, E.; Moreira, E.; Ayoub, K.; Goel, M.; Meyer, C.; Thornton, G.; Yang, Z.; Michalewski, H.; Abbas, Z.; Schucher, N.; Anand, A.; Ives, R.; Keeling, J.; Lenc, K.; Haykal, S.; Shakeri, S.; Shyam, P.; Chowdhery, A.; Ring, R.; Spencer, S.; Sezener, E.; and et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *CoRR*, abs/2403.05530.

Sabour, S.; Frosst, N.; and Hinton, G. E. 2017. Dynamic Routing Between Capsules. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 3856–3866.

Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q. V.; Hinton, G. E.; and Dean, J. 2017. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Su, L.; Pan, J.; Wang, X.; Xiao, X.; Quan, S.; Chen, X.; and Jiang, J. 2024. STEM: Unleashing the Power of Embeddings for Multi-Task Recommendation. In Wooldridge, M. J.; Dy, J. G.; and Natarajan, S., eds., *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, 9002–9010. AAAI Press.

Tang, H.; Liu, J.; Zhao, M.; and Gong, X. 2020. Progressive Layered Extraction (PLE): A Novel Multi-Task Learning (MTL) Model for Personalized Recommendations. In Santos, R. L. T.; Marinho, L. B.; Daly, E. M.; Chen, L.; Falk, K.; Koenigstein, N.; and de Moura, E. S., eds., *RecSys 2020: Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020*, 269–278. ACM.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 5998–6008.

Wetprasit, S.; Cao, Q.; and Seow, C. K. 2022. Recommender System for Coupon Discount of E-commerce Applications. In *5th International Conference on Data Science and Information Technology, DSIT 2022, Shanghai, China, July 22-24, 2022*, 1–6. IEEE.

Zhang, Y.; and Yang, Q. 2022. A Survey on Multi-Task Learning. *IEEE Trans. Knowl. Data Eng.*, 34(12): 5586–5609.

Zhao, Q.; Qian, H.; Liu, Z.; Zhang, G.; and Gu, L. 2024. Breaking the Barrier: Utilizing Large Language Models for Industrial Recommendation Systems through an Inferential Knowledge Graph. In Serra, E.; and Spezzano, F., eds., *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM 2024, Boise, ID, USA, October 21-25, 2024*, 5086–5093. ACM.

Zhao, Z.; Hong, L.; Wei, L.; Chen, J.; Nath, A.; Andrews, S.; Kumthekar, A.; Sathiamoorthy, M.; Yi, X.; and Chi, E. H. 2019. Recommending what video to watch next: a multi-task ranking system. In Bogers, T.; Said, A.; Brusilovsky, P.; and Tikk, D., eds., *Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019*, 43–51. ACM.

Zhou, Y.; Lei, T.; Liu, H.; Du, N.; Huang, Y.; Zhao, V.; Dai, A. M.; Chen, Z.; Le, Q. V.; and Laudon, J. 2022. Mixture-of-Experts with Expert Choice Routing. In *NeurIPS*.

Zoph, B.; and Le, Q. V. 2017. Neural Architecture Search with Reinforcement Learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.