Personalized Recommendation Multi-Objective Optimization Model Based on Deep Learning

Zepeng Yang
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, China
E-mail: 1241169825@qq.com

Ping Lu
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, China
E-mail: 1565364293@qq.com

Pingping Liu
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, China
E-mail: 1341369601@qq.com

***Abstract*—Recommended in this paper, because the existing single objective experience is poor, and the recommended model in a large difference of targets under the complex relationship of joint optimization and the conflict caused by faults, this paper proposes a personalized recommendation based on the deep learning multi-objective optimization algorithm, the estimated probability of users on the individual behavior as a model to study target, Multiple objectives are integrated into a model for learning. Firstly, the embedding layer is used to change the feature vectors, so that the bottom layer of the model shares the same feature embedding. Secondly, the factorization machine and deep learning are used to construct high-low order feature interaction. Then, the gating network and multilevel expert network constructed by a fully connected neural network are used to learn the characteristic relationship of user behavior. Finally, make connections between goals. Through experiments on public and real datasets, The results show that the multi-objective model proposed in this paper has better co-optimization performance and increases the AUC value by 0.1% compared with advanced personalized recommendation models such as MMoE and ESMM, to achieve the ultimate goal of increasing the prediction accuracy and improving user satisfaction.**

Keywords-Recommendation Algorithm;Multi-Objective Optimization; Post-Click Conversion Rate; Deep Learning; The Neural Network

I. INTRODUCTION

With the arrival of the big data era and the swift advancement of smart devices, personalized recommendations play a significant role in a variety of applications. Recommendation systems often use estimation models that target users' clicks and do not take sufficient account of the behavior generated by users after clicking, thus trapping users in smaller and smaller interest networks, reducing user engagement and satisfaction, resulting in the uneven development of the recommendation ecosystem and declining corporate interests [1]. Therefore, it has become a trend to apply multi-task learning to simulate both user satisfaction and engagement for multi-objective optimization [2].

In recent years, numerous studies have been conducted on recommendation problems that require the simultaneous optimization of multiple objectives but existing recommendation algorithms have the following problems: 1) Sample data is sparse. In general, users rarely rate items, which also lead to overt data being too sparse, so implicit information needs to be used for recommendations. In traditional CVR estimation models, positive and negative samples are usually extremely unbalanced, which increases the difficulty of model training and poses

DOI: 10.2478/ijanmc-2024-0005

generalisation problems [3]. 2) Sample selection bias. Traditional pCVR estimation uses a technique similar to CTR estimation that is, training by clicking on a subset of samples, and inferring the entire display sample space when reasoning. However, this method has the problem of sample selection bias. 3) Multi-objective "seesaw" phenomenon. Some multi-objective models, while improving some objectives, tend to sacrifice the performance of others. One of the main optimization issues in multi-objective learning comes from different target gradients that tend to clash with each other in ways that are not conducive to progress. In some cases, this collision gradient can cause a significant decrease in performance.

Currently, many large-scale recommendation systems both domestically and internationally have implemented multi-task learning with deep neural network models [4]. Researchers pointed out that multi-objective model can use regularisation and transfer learning to improve the model's predictions for all objectives. However, experimental results show that in fact multi-objective Multi-objective models do not consistently exceed their single-objective counterparts across all objectives. Deep learning-based multi-objective models often exhibit high sensitivity to factors such as distribution of data and variations in relationships between targets. The inherent conflicts brought about by target differences can impair the prediction of at least some targets, especially when the parameters in the model are widely shared among all targets.

Therefore, this paper proposes a multi-objective optimization recommendation algorithm that uses deep learning technology to fuse user behavior information, which can better use the prior knowledge in shared network design to capture complex task correlations [5].

This paper presents deep learning-based multi-objective network architecture for personalized recommendation for the sequencing phase of the recommendation system. On the basis of a shared underlying model, the model proposes to use factor decomposer and deep learning to construct higher-order and lower-order feature interactions, and then introduce a separate gating network for each target, and then introduce a multi-level expert network for the model [6]. In addition, it introduces ESMM to optimize how the loss function is constructed, allowing for more accurate fitting of various conversion rates.

This article applies this model architecture to video recommendations as a case study: using the user's past viewing habits as a basis, recommend the videos you want to watch later. The experiments set up two classification tasks and conducted a large number of offline experiments to evaluate the effectiveness of the model, and the results show that it is helpful to evaluate the significant improvement of the index in this prediction task.

II. RELATED WORK

A. Research Overview

The recommendation model in this paper learns based on feedback from two types of users: (1) participatory behaviors, such as clicking and watching; (2) Satisfying behaviors, such as sharing, commenting, and collecting. Given the historical behavior information of each user, the ranking system takes user characteristics, video content features and historical behavioural features are used as inputs and learn to predict multiple user behaviours. We formulate the sorting problem as a classification task and compute the cross-entropy loss. [7]. Given user characteristics and video characteristics, ranking models predict the probability that users will take actions such as clicks, watch time, shares, and comments.

After identifying multiple ranking objectives and their problem types, multitask ranking models can be trained for these prediction tasks. For each candidate, take these multiple predictions as inputs and output the combined score using a combinatorial function in the form of weighted multiplication to achieve best performance in terms of user engagement and user satisfaction [8].

B. Related Knowledge

Cheng et al. proposed a Wide & Deep Learning model by using multi-source heterogeneous data such as user characteristics, situational characteristics, and project characteristics [9]. As shown in Figure 1, this model combines the training of a broad linear model (on the left side of the figure) and a deep neural network (left side of the figure) to ensure a balance between the ability of model to memorise and generalize. Guo et al. based on Wide & Deep, combined with factorization machine and deep learning, proposed a factorization-Machine based Neural Network ($Deep_{FM}$) for click-through rate prediction, using factorization machine and deep neural network to model low-level and high-level feature interactions, respectively, compared with Wide & Deep [10]. $Deep_{FM}$ does not require manual feature engineering.

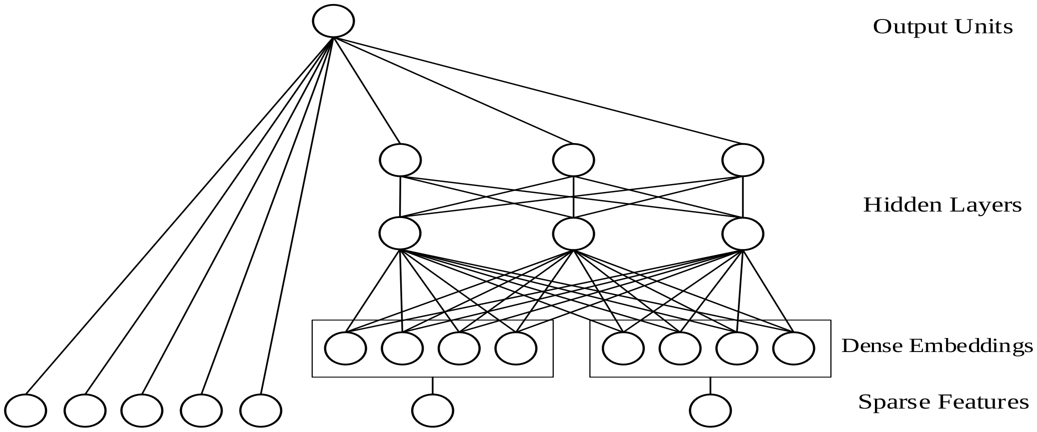

Figure 1. Model structure of the Wide & Deep Learning model.

The shared-substrate multitasking model was proposed in 1998 and is shown in Figure 2. In which the model structure is characterized by the fact that all targets share the same input, and because the underlying parameters are shared by all targets, the risk of overfitting is greatly reduced [11]. At the same time, different goals can also transfer knowledge through these shared parameters when learning, and use the knowledge learned by other goals to help their own goals learn. This model is often regarded as a iconic benchmark approach in multi-objective modelling.

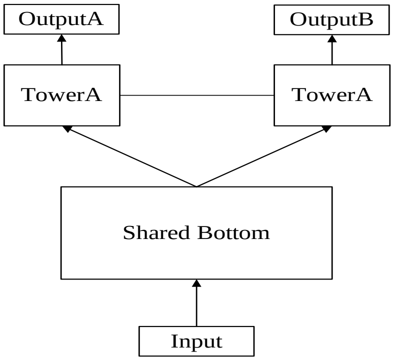

Figure 2. Shared-Bottom model.

Input features in the Web domain are often discrete and sparse, and the interaction between features is critical to effectively model this type of data [12]. In the various multi-objective models of deep learning, it is common to use a model structure that shares underlying parameters. This model structure reduces the risk of overfitting and facilitates the learning of the target because the underlying parameters can be shared by all targets. However, when the correlation between targets is relatively low, this hard parameter sharing will limit the freedom of each target fit, impairing multi-objective learning.

C. FM_Shared Expert Multi-Objective Dependency Model（FSMD）

Figure 3 illustrates the overall modelling framework results, and earning feature relationships for user behaviour using fully connected neural networks, and using noise reduction auto-coding to initialize user behavior information. An efficient multi-objective neural network architecture was designed, which extended the Wide & Deep model and adopted a multi-objective learning model architecture with a mixture of multiple experts. In addition, a shallow tower was introduced to model and eliminate selection bias, and an ESMM way of constructing loss functions was introduced to establish a connection between targets. In addition, a multi-objective sorting model based on deep learning is formed to integrate user behavior characteristics for interest recommendation.

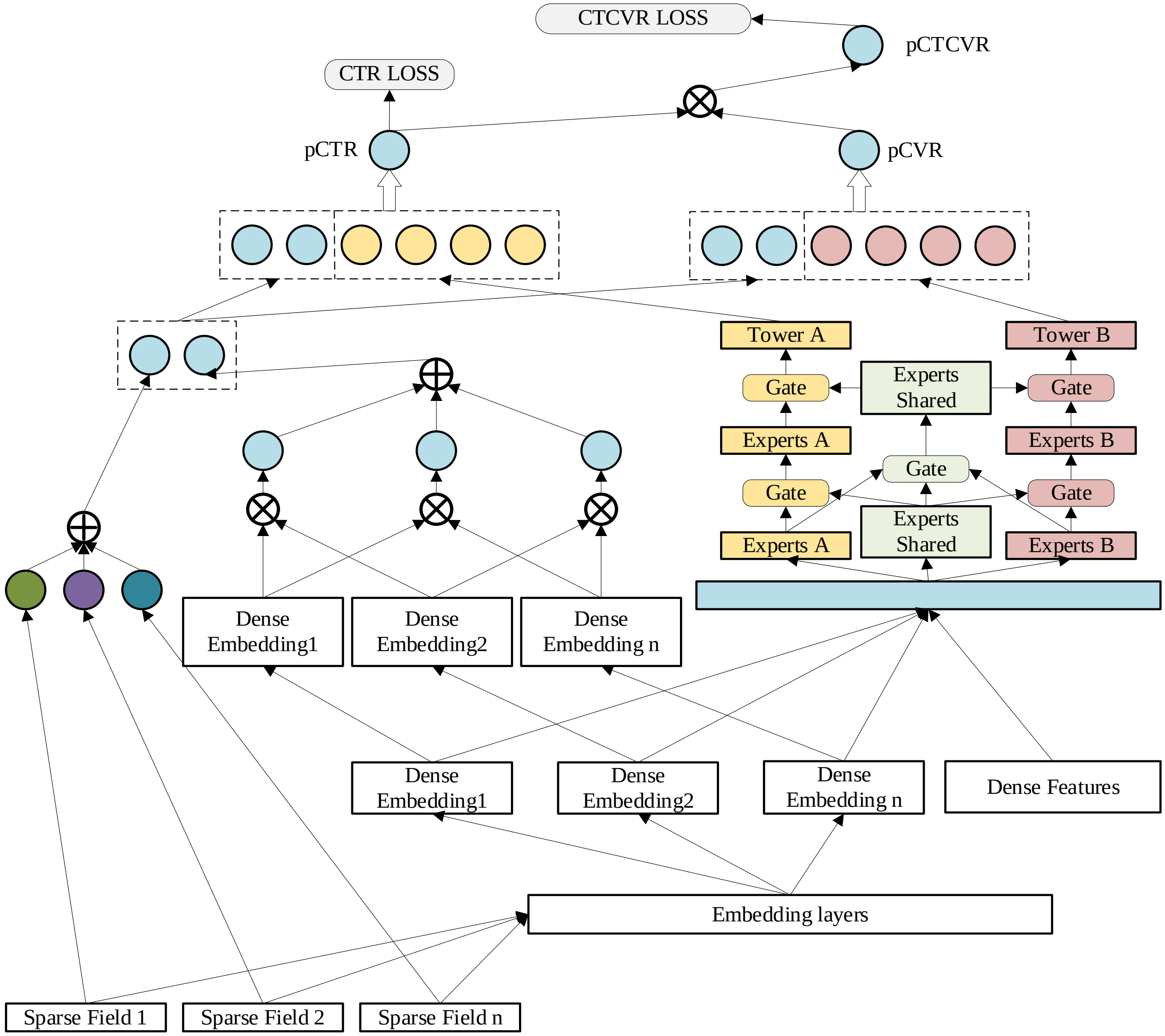

Figure 3. FM_Shared Expert Multi-Objective Dependency (FSMD) Mode

III. MODELING METHODS

A. Bottom Sharing of High and Low Level Feature Interaction

The idea taken in this paper is to combine the factorization machine MLP [13], first use the factorization machine to model the pairwise interaction between features, and then further model the higher-order feature interactions by adding a fully connected layer. To take full advantage of this technology of $Deep_{FM}$, the research at this stage chose to build a multi-objective model based on $Deep_{FM}$, Low-level and high-level feature interactions are modelled using factorial decomposers and deep neural networks, respectively.

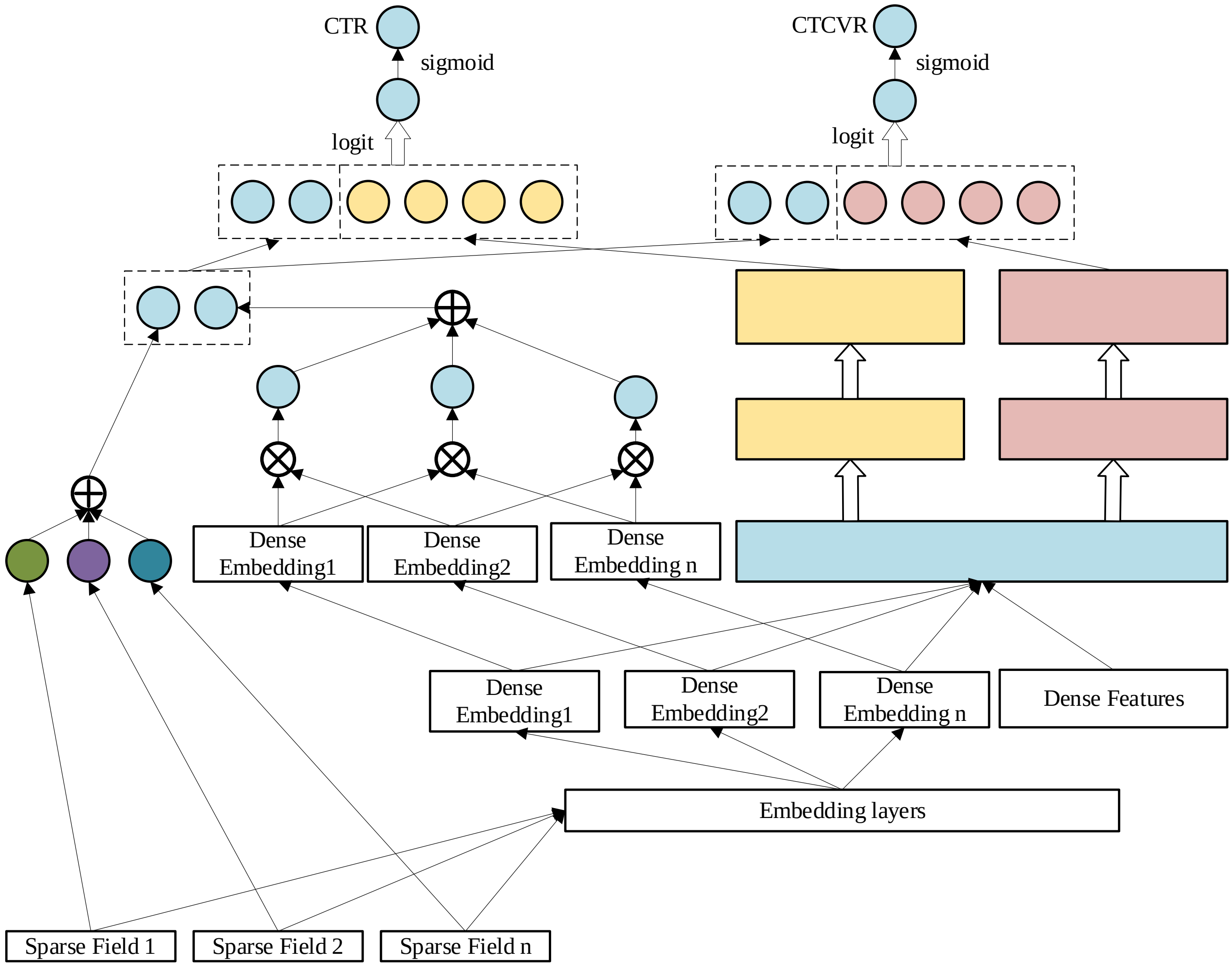

Figure 4. Multi-objective Base Model.

In this phase, the two goals of click and watch time are modeled. Leave the FM part of $Deep_{FM}$ unchanged, replace the DNN part of $Deep_{FM}$ with the Share Bottom structure of hard parameter sharing, and obtain a multi-objective model combining $Deep_{FM}$ and Share Bottom Model as the baseline for the study of the multi-objective model.

As shown in Figure 4, the FM subnetwork on the left calculates the second-order crossover fraction of sparse features and dense features, and the deep subnetwork on the right stitches dense features and continuous features into the network. Finally, the FM first-order, second-order fractions, and the last layer of deep inputs are stitched together, and the estimated value is obtained by sigmoid.

The model predicts the following: (1)

$$\hat{y} = sigmoid(y_{Deep} + y_{FM}) \qquad (1)$$

Among them, y ̂∈(0,1) is the predicted CTR, y_{FM} and y_{Deep} are the outputs of the FM component and the output of the deep component, respectively.

$$y_{FM} = <w,x> + \sum_{j_1=1}^{d} \sum_{j_2=j_1+1}^{d} <V_i,V_j> x_{j_1}, x_{j_2} \qquad (2)$$

Where $x = [x_{field1}, x_{field2}, ..., x_{fieldj}, x_{fieldn}]$ is a vector of d-dimension, x_{field} is the vector representation of the j_{th} domain of X. For feature i, the importance of the 1st order is measured by a scalar w_{i}, and the impact of its interaction with other features is measured by a latent vector v_i. The addition unit <w,x> in the network architecture functions to capture the significance of the 1st-order feature, while the inner product element signifies the impact of the 2nd-order feature interaction. This approach allows for the consideration of both individual feature importance and their interactions, enhancing the model's ability to make personalized and accurate recommendations.

$$y_{deepk} = h^k(f(x)) \qquad (3)$$

The output result of the shared hidden layer, f(x), is input to the respective tower network (subnetwork) h^k, and finally, each target k gets an output y_{deepk} . $\hat{y} = sigmoid(y_{FM} + y_{Deep})$

B. Gated Network Adaptive Weighting

Because of the shortcomings of the Share Bottom model structure, Google proposed the Multi-gated Mixture of Experts (MMoE) model in 2018 [14], which introduces multiple expert subnetworks and gating structures, and uses different expert combinations to learn different goals through gating so that each goal can be better learned. As shown in figure 5 is a schematic diagram of the MMoE model structure.

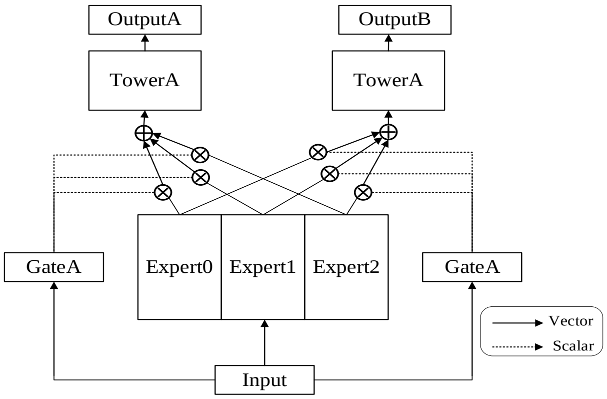

Figure 5. Multi-gate MoE Model.

The design of this stage is influenced by the Multi-Task Mixture of Experts (MMoE) architecture, which involves incorporating a distinct gated network g^k for each target k. These networks are added to the deep part of the model, building upon the framework established in the previous stage. This approach allows for the creation of specialized networks for individual targets, enabling the model to effectively capture the intricacies and nuances specific to each task. g is the gating network that combines the results of the experts, the internal implementation of the gating is composed of the same multilayer perceptron with ReLU activation, and the gating network is a simple linear transformation of the input with the softmax layer, as shown in figure 6:

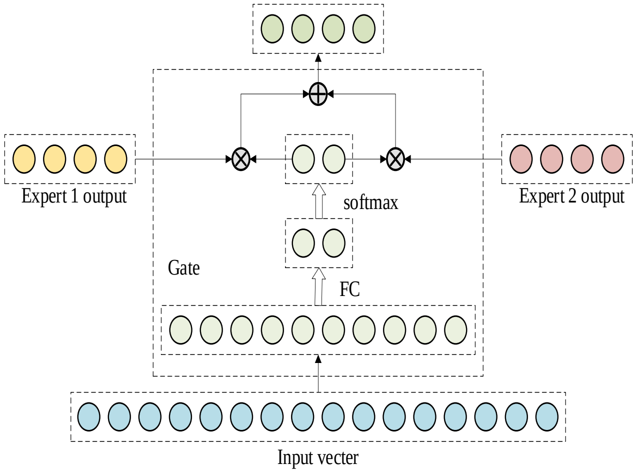

Figure 6. Internal Structure of Gated Network.

The input vector and the output vector of each expert will be passed into the gating network, the input vector will first pass through MLP, and the last layer of softmax will get the weight of each expert, and the output of the gating is the weight on all experts:

$$g^k(x) = soft\max(W_{hk}x) \qquad (4)$$

Where $W_{gk} \in R^{nxd}$ is the trainable matrix, n and d denote the number of experts and the edge dimension, respectively.

Multiple expert networks are added at the same time as the introduction of a gating network for each goal, and the gating network learns different combinations of the expert network for their respective tasks, and the output of the expert network is adaptively weighted. For a task, the output of its corresponding network of experts is:

$$f^k(x) = \sum_{i=1}^{n} g^k(x)_i f_i(x) \qquad (5)$$

Where f_i (i=1,…,n) is a network of n experts.

The new multi-objective model based on the gated network is shown in Figure 7, and the improved advantage is that each target can train a gated network individually, and the weight of each expert network owned by each target task is adjusted according to the objective adaptively. Get the output of the deep part target k as:

$$y_k = h^k(f^k(x)) \qquad (6)$$

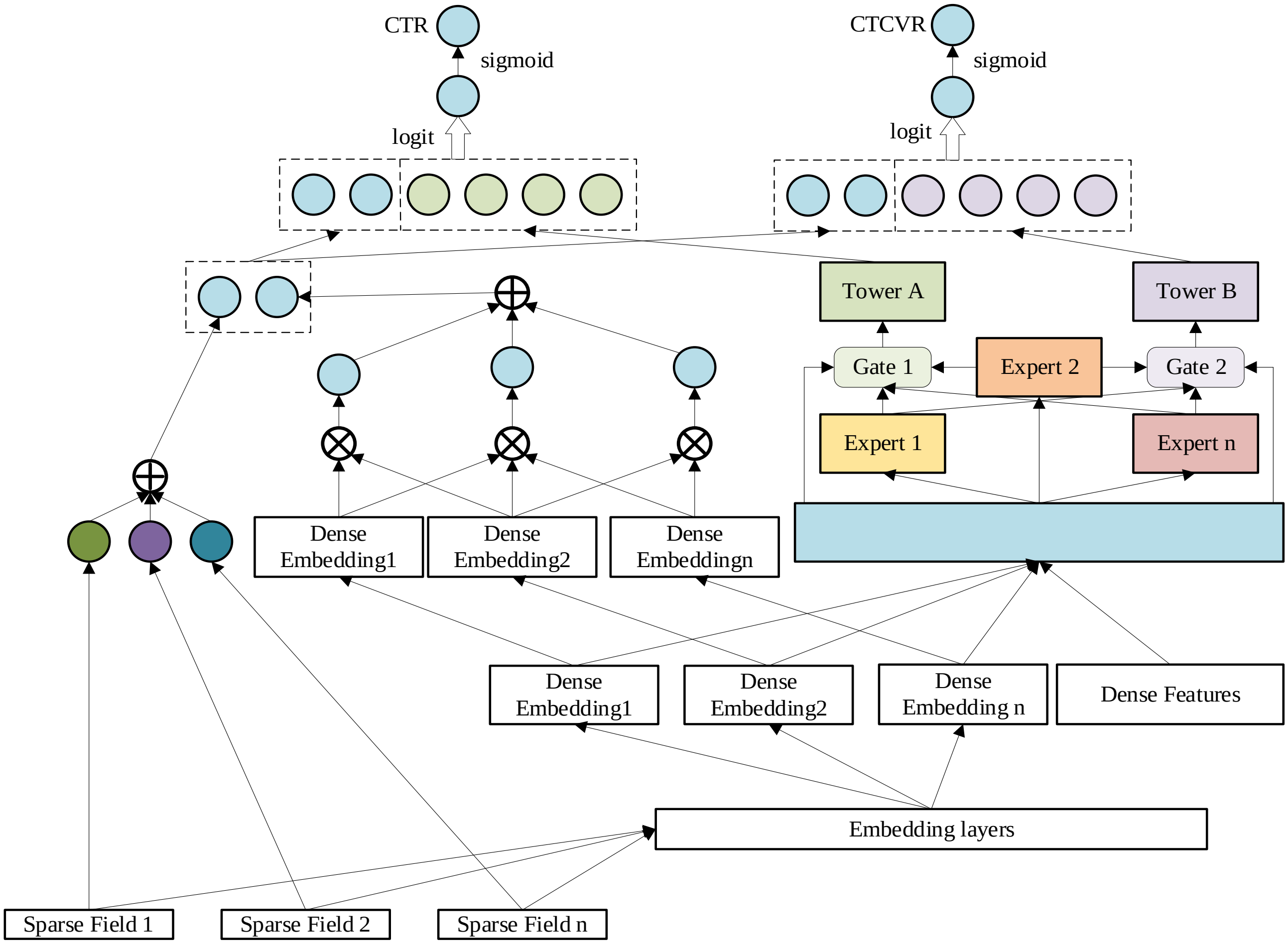

Figure 7. FM_Gate (FG) Model.

In this stage of the model, the target uses the binary classification cross-entropy to make losses, and then the loss weights of the two targets are summed to obtain a total loss function, and the model parameters are solved by optimizing this total loss function. The total loss function is as follows:

$$L = \mathrm{m}\underset{\theta}{i}n w_1 \sum_{i=1}^{N} L_1(y_i, P_{ctr}(x_i, \theta)) + w_2 \sum_{i=1}^{N} L_2(z_i, P_{cvr}(x_i, \theta)) \qquad (7)$$

Among them, L_1 and L_2 are the loss functions of fitting CTR and CTCVR, respectively, and both are binary classification cross-entropy; x_i indicates the input feature; y_i is the click target, click is 1, exposure unclicked is 0; z_i is the conversion goal, converted to 1, click not converted to 0; $P_{ctr}(x_i,\theta)$ is an estimate of CTR, $P_{cvr}(x_i,\theta)$ is an estimate of CVR, θ is the model parameter, w_1 and w_2 are the weights of the two losses, and N represents the total quantity of samples.

C. Multi-level Expert Network

Multi-objective modeling often has a seesaw phenomenon, usually, multi-objective learning relative to multiple single-objective learning models can improve the effectiveness of a part of the goal, but some multi-objective models often at the expense of the performance of other targets to improve some goals, this problem is called seesaw phenomenon. One of the primary challenges in multi-objective learning arises from the gradients of diverse objectives, which tend to clash with each other in a way that is not conducive to progress, and in some cases can lead to a significant decrease in performance.

To solve the "seesaw" problem that multiple objectives are prone to, a multi-level expert network is introduced based on the previous FG model, and an independent expert network is established for each target while retaining a shared expert network [15]. Introducing multi-level experts, which consist of shared experts and specific target experts, helps mitigate harmful parameter interference and enables the integration of multi-objective features through gated networks. Implementing a novel progressive separation approach facilitates the emulation of interactions among experts, leading to more effective knowledge transfer between intricate and interconnected objectives. Get FM_Sharing Expert (FS) model to solve the seesaw problem, ensure stable optimization, and the deep part of the model is shown in the figure 8.

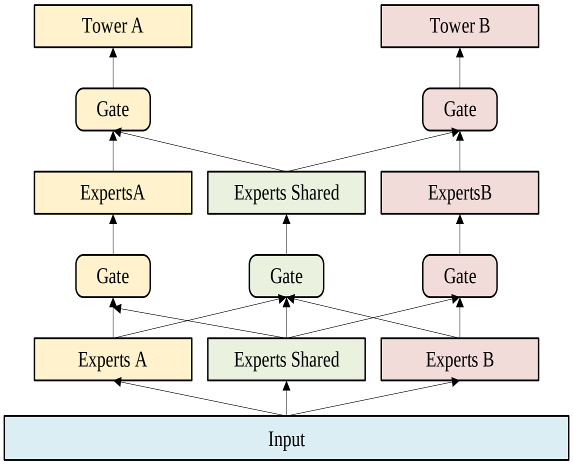

Figure 8. FM_Sharing the deep part of the Expert (FS) Model.

Definition of the gated network in the j_{th} extraction network of the kth sub-target in the FS model deep section:

$$g^{k,j}(x) = w^{k,j}\left(g^{k,j-1}(x)\right)S^{k,j}(x)\right) \qquad (8)$$

Where $w^{k,j}$ the weight is function of the target k as the input to $g^{k,j-1}$, and $S^{k,j}$ is the selection matrix of the j_{th} extraction network for the target k. After calculating all the gating networks and expert networks, the final output of the k_{th} sub target of the deep part of the FS model is:

$$y^k(x) = t^k(g^{k,N}(x)) \qquad (9)$$

D. Build Target Dependencies

In a recommendation scenario, the user's behavior is generally more than one, and the different behaviors occur in order and dependencies. Each of a user's behavior can be a target in a multi-objective model, and there are dependencies between these targets. For this correlated multi-objective model, if each target fits independently, the information about the dependencies between the targets will be lost, and the accuracy of the model will be lost, affecting the sorting effect. Therefore, when doing correlation multi-objective modeling, it is necessary to model every step of the transformation in user behavior. In the case of video recommendations, for example, in this scenario, the goal we need to model is clicked and playtime and the conversion relationship involved in these two goals can be described as: showing the video to the user——the user clicks on the video——the viewing time exceeds a certain threshold.

Use x to represent the characteristics of the user and the video; y indicates the label of the click, y=1 indicates the click, and y=0 indicates that the exposure is not clicked; z indicates the label of the playback duration, z=1 indicates that the playback time exceeds the threshold, and z=0 indicates that the threshold has not been exceeded, including no clicks. The conversion relationship and probability quantification of exposure, clicks, and duration in user behavior can be expressed as follows [16]:

$$\underbrace{p(y=1,z=1\,|\,x)}_{pCTCVR} = \underbrace{p(y=1\,|\,x)}_{pCTR} * \underbrace{p(z=1\,|\,y=1,x)}_{pCVR} \qquad (10)$$

To do this, we combined the loss function of the previous version of the multi-objective model and the ESMM to obtain a multi-objective model as shown in the figure 9.

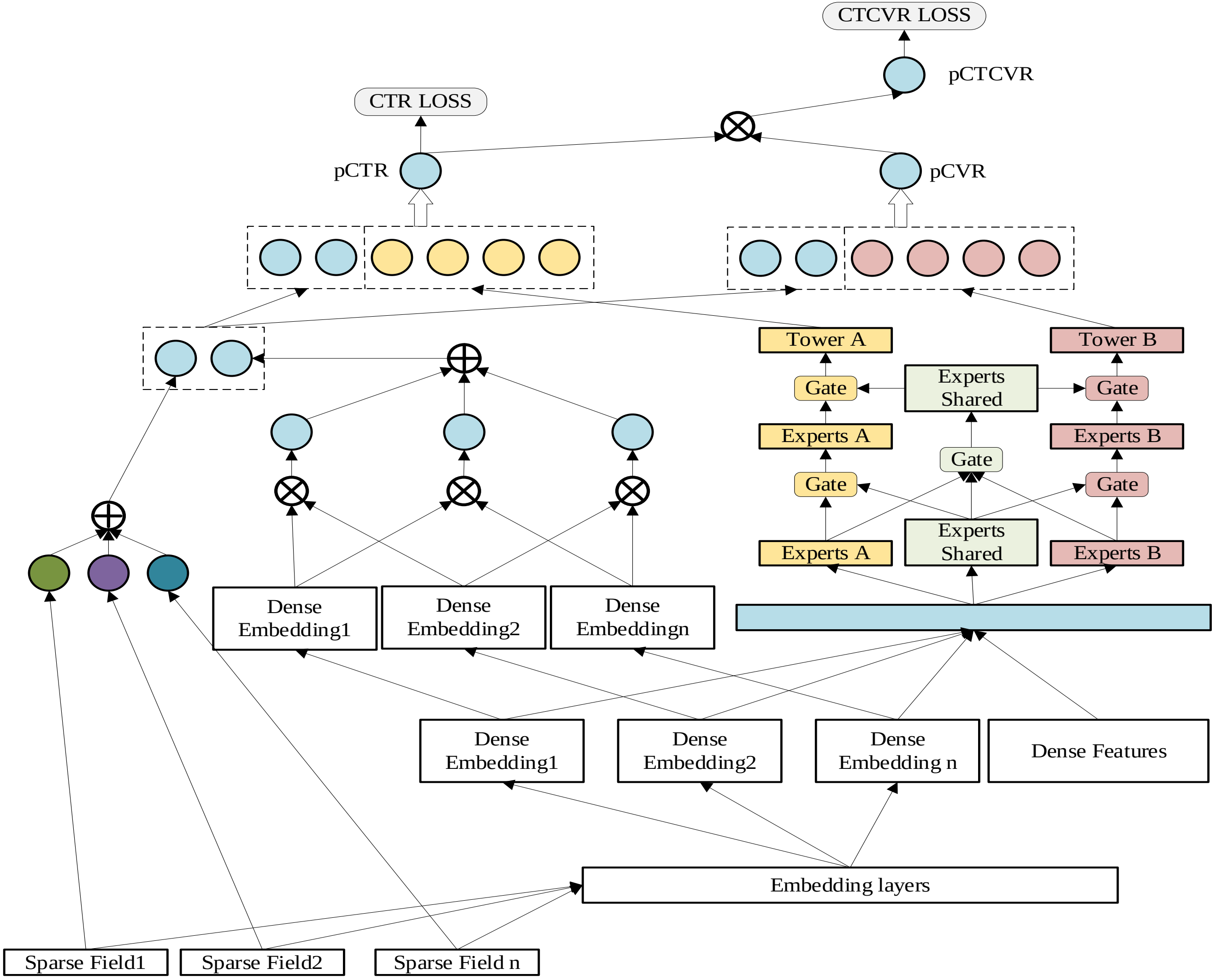

Figure 9. FM_Shared Expert Multi-Objective Dependency(FSD)Model.

Finally, the CTR loss and CTCVR loss weighted sum give a total loss, and the model parameters are solved by minimizing the total loss.

$$L = \min_{\theta} w_1 \sum_{i=1}^{N} L_1(y_i, P_{ctr}(x_i, \theta)) + w_2 \sum_{i=1}^{N} L_2(zi, P_{ctr}(x_i, \theta) * P_{cvr}(x_i, \theta)) \quad (11)$$

Among them, L_1 and L_2 are the loss functions that fit CTR and CTCVR, respectively, and both are two-classification cross-entropy; y_i is the class label of the click; z_i as a class indicator for the length of playback (1 for longer playback than the threshold, 0 for otherwise), $P_{ctr}(x_i, \theta)$ is an estimate of CTR; $P_{cvr}(x_i, \theta)$ is an estimate of CVR; θ is the model parameter; w_1 and w_2 are the weights of the two losses respectively, with N being the total number of samples.

IV. EXPERIMENT

A. UCI Census Dataset

1) The dataset description

This experiment constructed a multi-task learning problem with multiple features as prediction targets, which used the UCI Census income dataset:

Goal one: whether the forecast revenue exceeds $50,000;

Goal two: to forecast if the individual is married.

In this data set, there are 42 characteristics, including important information such as age, job type, education, occupation, ethnicity, etc., 199523 training examples, and 49881 test examples.

2) Experimental settings

Since both goals were binary classification problems, the experiment used the AUC score as an evaluation indicator. The income task is the primary task and the marital status task is a secondary task. Each model uses the same hyper parameters, and the parameter settings are shown in the following table. Every model is trained on the training dataset using identical parameter initialization, and the findings are then presented for the test dataset.

TABLE I. PARAMETER SETTINGS

Parameter Name	Value
batch_size	256
optimizer	adam
learning_rate	0.001
embedding_size	4
dnn_layers	(512, 256)
dnn_use_bn	True

Batch_size: the number of samples used to calculate the gradient, which in this chapter is set to 256;

Optimizer: The optimizer for parameter optimization of the constructed network model, in this experiment, the selected optimizer is Adam;

learning_rate: learning rate set to 0.001;

embedding_size: Used for the Dense Embedding layer, combined with the value characteristics of each feature, the corresponding embedding_size is set;

dnn_layers: Represents the number of neurons in the hidden layers of the feed-forward neural network;

dnn_use_bn: is a Boolean value that controls whether to use the BN layer, and its value is set to True;

In the experiment, the task was a binary classification task trained using cross-entropy loss and evaluated with AUC.

3) Experimental Results

TABLE II. RESULTS OF THE UCI CENSUS INCOME DATASET

Models	AUC/ Income	AUC/ Marital	Mean
Single-Task	0.9198	0.9748	0.9473
Shared-Bottom	0.9148	0.9754	0.9451
MMoE	0.9152	0.9756	0.9454
PLE	0.9161	**0.9764**	0.9463
Base	0.9134	0.9706	0.9420

Models	AUC/ Income	AUC/ Marital	Mean
FG	0.9146	0.9693	0.9420
FS	**0.9216**	0.9756	**0.9486**

From the respective AUC and average data of the two targets in the results, it can be concluded that the FS model can improve the AUC of the first target by 0.0055 without significantly reducing the AUC of the second target compared with the PLE and optimize the average AUC of the two targets. It can be concluded that the FS model that combines the high order and the low order is better than the PLE model with only the deep part, thus verifying the fusion effect of the interaction between the low order and the high order features. From the comparison of the model base and FS model, the first goal is improved by 0.0082, the second goal is increased by 0.005, and the average AUC of the two goals is increased by 0.0066.

B. The Video Site Plays the Dataset

1) The dataset description

The dataset used in study is the user log of a video website for 15 consecutive days, including user characteristics, video content characteristics, and user historical behavior data. The data includes user dimensions, video dimensions, and user historical behavior data, which are described separately in these three dimensions.

User-side attributes information: user ID, age range, gender, province or city, city, city level, and device type.

Video side attributes information: video ID, video age, video month, video rating, and video duration.

User behavior information: user ID, video ID, whether to play, whether to share, whether to favorite, whether to comment, watch time, play tag, watch date.

2) Experimental results

a) Comparative experiments

The multi-objective model constructed in this paper is compared with the single-objective model, the classic multi-objective model, and the multi-objective model designed in the previous period, and the model effect is analyzed according to the rating index AUC. Each model in the model comparison experiment uses the same parameters, and the parameter settings are shown in the following table.

TABLE III. PARAMETER SETTINGS

Parameter Name	Value
batch_size	2048
optimizer	adam
learning_rate	0.001
embedding_size	4
dnn_layers	(256, 128)
dnn_use_bn	True
dropout	0.5

The chart below shows the test results after 30 rounds of training.

TABLE IV. PARAMETER SETTINGS

Models	AUC/ Income	AUC/ Marital	Mean
Single-Task	0.7192	0.6635	0.6914
DeepFM	0.719	0.658	0.689
Shared-Bottom	0.7184	0.6916	0.705
ESMM	0.7189	0.6897	0.7043
MMoE	0.7201	0.7057	0.7129
Base	0.7193	0.7086	0.714
FG	0.7201	0.7084	0.7143
FS	0.7204	0.7114	0.7159
FSMD	**0.7205**	**0.7117**	**0.7161**

Table shows the prediction performance of various models on the video dataset. The results show that the model proposed in this paper significantly outperforms all baseline models for the transformation goal. Due to the complex correlation between click goals and duration goals, the seesaw phenomenon can be observed from the results, with some models improving click goals but hurting duration goals, and others improving duration goals but hurting click targets. Specifically, the baseline model that combines FM and deep improves both goals compared to the single-target model, but the improvement is not significant, while the FS model with gating and expert networks but does not establish a connection between the modular targets only improves the click target, but damages the duration goal. Compared with the typical and widely used multi-objective models MMoE and ESMM, this model has a much greater improvement over the duration target and a small improvement on the click target. Finally, this model converges at a similar rate and achieves significant improvements on the above model with one of the AUCs.

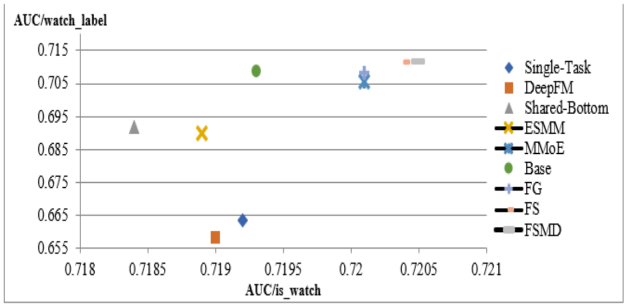

Figure 10. The seesaw phenomenon in each model under complex target association

There is a complex relationship between click targets and duration targets, so modeling two targets at the same time will make the "seesaw phenomenon" more obvious. As can be seen from the figure 10, with the Base model as the baseline zero point, only FS and FSMD are surpassed in the two targets at the same time, and the other models have obvious seesaw phenomenon, only the FSMD model designed in this paper achieves the optimal at the same time.

b) Ablation experiment

This experiment compares the AUC of the designed multi-objective model under different network layers. In the experiment, the FSMD model of the two-layer underlying network, the two-layer tower network, the one-layer gating network, and the 8-expert network is selected as the skeleton network and baseline of the ablation experiment, and different network layer combinations are modified on the skeleton network for training and evaluation, and the final ablation results are shown in the table:

In the first ablation experiment , the number of layers of the gated network is set to 2 layers, the number of expert networks is set to 8, and the underlying network is set to (256, 128), comparing the AUC under different tower network shapes.

The experiment tested four structurally different tower network shapes: constant, incremental, decreasing, and diamond. When changing the shape of its network, the number of layers of hidden layers is fixed. For example, when the number of hidden layers is 3, then the four different shapes are constant (128-128-128), increasing (64-128-256), decreasing (256-128-64), and diamond (64-128-64).

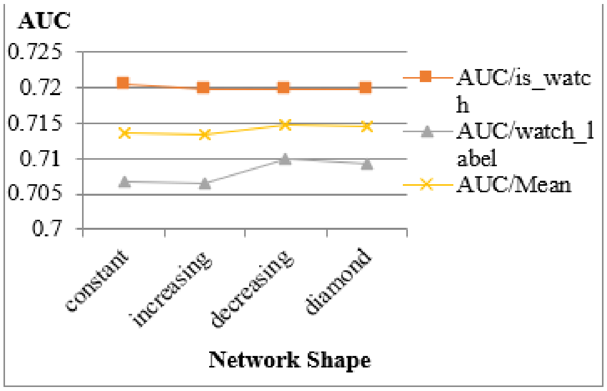

Figure 11. AUC comparison of network shapes.

It can be concluded from the results, as shown in figure 11, that the diamond network is higher than other network shapes in both the conversion target and the average AUC of both goals based on not significantly reducing the AUC of the click target. It can therefore be concluded that the diamond is the optimal choice in the choice of tower network shape.

In the ablation experiment, set Tower_mlp_dims to (256,128) and Bottom_mlp_dims to (256,128), the number of experts is 8, comparing AUC under different gating network layers.

It can be concluded from the result, as shown in figure 12, that when the number of layers of the gating network is 3, the model achieves the highest number of layers compared to the duration target and the average AUC of the two targets. It can be concluded that the number of layers in the gating network should not be too small, nor too much, the effect of 3 or 4 layers is better, considering the complexity of the model, it is considered that 3 layers are the best choice.

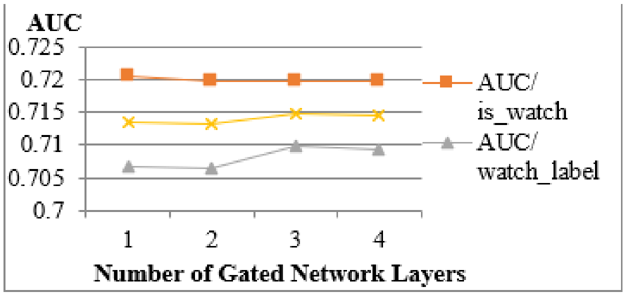

Figure 12. AUC comparison of the number of layers in a gated network.

The third ablation experiment, set Tower_mlp_dims to (256,128), Bottom_mlp_dims (256,128), and the number of gated network layers is 1, comparing AUC under different numbers of expert networks.

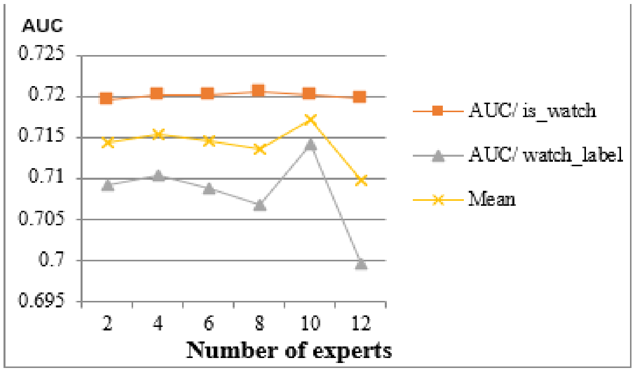

Figure 13. AUC comparison of the number of experts.

As can be seen from the figure 13, a change in the number of expert networks does not have much impact on the click target, but it changes a lot on the duration goal. When the number of expert networks rose from 2 to 10, the conversion goal improved significantly, and the model reached the highest number of experts compared to the number of other experts in terms of duration target and average AUC of both goals. As the number of expert networks continues to increase, it can be seen that there is a significant decrease in the AUC of the duration target, so the more expert networks the better.

V. Conclusions

In this article, some of the real-world challenges of current recommendation systems are first described, including sparse sample data, selection bias implicit in user feedback, and the phenomenon of "seesaws." To address these challenges, a multi-objective optimization ranking model based on deep learning is proposed and applied to the question of recommending what videos to watch next. To effectively optimize multiple ranking targets, the multi-expert hybrid model architecture is extended, and an effective method is built to reduce and model the bias of selecting multi-objective models by using soft parameter sharing and combining high- and low-level feature interactions and multi-level expert networks. In addition, through experiments on different datasets, it can be concluded that the model proposed in this paper is a significant improvement over the existing single-objective model for all objectives in all target groups, concluding that the model in the multi-objective case shows the benefits of facilitating goal co-operation and preventing negative migration and see-saw phenomena. Therefore, it is confirmed that the multi-objective model based on deep learning designed in this paper shows greater advantages in improving the shared learning efficiency of different scale target groups, and the technology we propose has achieved substantial improvement in participation and satisfaction indicators.

References

[1] Zhou G, Mou N, Fan Y, et al. Deep interest evolution network for click-through rate prediction. AAAI 2019, 33: 5941-5948.

[2] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In Computer Vision and Pattern Recognition, 2018.

[3] Chen C,Meng X,Xu Z,et al.Location-aware personalized news recommendation with deep semantic analysis.IEEE Access,2017:173-182.

[4] Wang R, Fu B, Fu G, et al. Deep & cross network for ad click predictions. ADKDD 2017: 1-7.

[5] LeCun Y, Bengio Y, Hinton G. Deep Learning. Nature, 2015, 521(7553): 436-444.

[6] Song W, Shi C, Xiao Z, et al. Autoint: Automatic feature interaction learning via self-attentive neural networks. CIKM 2019: 1161-1170.

[7] Chen Q, Zhao H, Li W, et al. Behavior sequence transformer for e-commerce recommendation in Alibaba. Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data. 2019: 1-4.

[8] Shikun Liu, Edward Johns, and Andrew J Davison. 2019. End-to-end multi-task learning with attention. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 1871–1880.

[9] Cheng H T,Koc L,Harmsen J,et al.Wide & deep learning for recommender systems//Proceedings of the 1st Workshop on Deep Learning for Recommender Systems.Boston,USA,2016:7-10.

[10] Guo H,Tang R,Ye Y,et al.DeepFM:A factorization-machine based neural network for CTR prediction//Proceedings of the 26th International Joint Conference on Artificial Intelligence.Melbourne,Australia,2017:1725-1731.

[11] Rich Caruana. 1998. Multitask learning. In Learning to learn. Springer, 95–133.

[12] Chen C,Meng X,Xu Z,et al.Location-aware personalized news recommendation with deep semantic analysis.IEEE Access,2017:173-182.

[13] Rendle S.Factorization machines//Proceedings of the 2010 IEEE 10th International Conference on Data Mining.Sydney,Australia,2010:995-1000.

[14] Jiaqi Ma,Zhe Zhao,Xinyang Yi,Jilin Chen,Lichan Hong,Ed H. Chi. Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts[P]. Knowledge Discovery & Data Mining,2018: 1930-1939.

[15] 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 (RecSys '20). Association for Computing Machinery, New York, NY, USA, 269–278.

[16] Xiao Ma et al. “Entire Space Multi-Task Model: An Effective Approach for Estimating Post-Click Conversion Rate” International ACM SIGIR Conference on Research and Development in Information Retrieval (2018).