PHN: Parallel Heterogeneous Network with Soft Gating for CTR Prediction

Ri Su, Alphonse Houssou Hounye, Muzhou Hou, and Cong Cao(✉)

Mathematics and Statistics School, Central South University,
Changsha 410083, China
{suricsu,hounyea,congcao}@csu.edu.cn, houmuzhou@sina.com

Abstract. The Click-though Rate (CTR) prediction task is a basic task in recommendation system. Most of the previous researches of CTR models built based on Wide & deep structure and gradually evolved into parallel structures with different modules. However, the simple accumulation of parallel structures can lead to higher structural complexity and longer training time. Based on the Sigmoid activation function of output layer, the linear addition activation value of parallel structures in the training process is easy to make the samples fall into the weak gradient interval, resulting in the phenomenon of weak gradient, and reducing the effectiveness of training. To this end, this paper proposes a Parallel Heterogeneous Network (PHN) model, which constructs a network with parallel structure through three different interaction analysis methods, and uses Soft Selection Gating (SSG) to feature heterogeneous data with different structure. Finally, residual link with trainable parameters are used in the network to mitigate the influence of weak gradient phenomenon. Furthermore, we demonstrate the effectiveness of PHN in a large number of comparative experiments, and visualize the performance of the model in training process and structure.

Keywords: Recommendation system · Click-though rate · Feature interaction

1 Introduction

The Click-through rate (CTR) prediction is one of the important basic tasks in recommendation system. By predict the click rate of user, the web or application can sort the candidate item list and push them to target user, so as to provide personalized recommendation service for users. Early CTR prediction models output CTR through Logistic Regression, and use automatic feature engineering methods such as Factorization Machine (FM) [10] and Gradient Boosting Decision Tree (GBDT) [3] for business implementation. With the development of deep learning, CTR prediction model based on neural network like PNN [9] has gradually become the mainstream application model in the real application.

Wide&deep [2] CTR predict model structure have used parallel structures of different depths to consider both memorization and generalization. In subsequent

L. Fang et al. (Eds.): CICAI 2022, LNAI 13605, pp. 413–424, 2022.
https://doi.org/10.1007/978-3-031-20500-2_34

studies, FNN [18], DeepCrossing [11], DeepFM [4], DCN [16], xDeepFM [7], DCN-V2 [17], EDCN [1], NFM [5] and other models have similar parallel structure like Wide&deep, and were utilized to analyze public embedding through different modules. Moreover, the generalization of this structure depended on the effectiveness of parallel structures.

There is no comprehensive analysis of feature interaction in previous parallel structure models. Therefore, the generalization of the model is limited. At the end of CTR model, the click rate prediction output have been achieved by linear layer with activation function. During training phase, the activation values between parallel layers tend to fall into the weak gradient interval. This phenomenon will weaken the training effect of each parallel module, and can not improve the generalization while improving the complexity of the model.

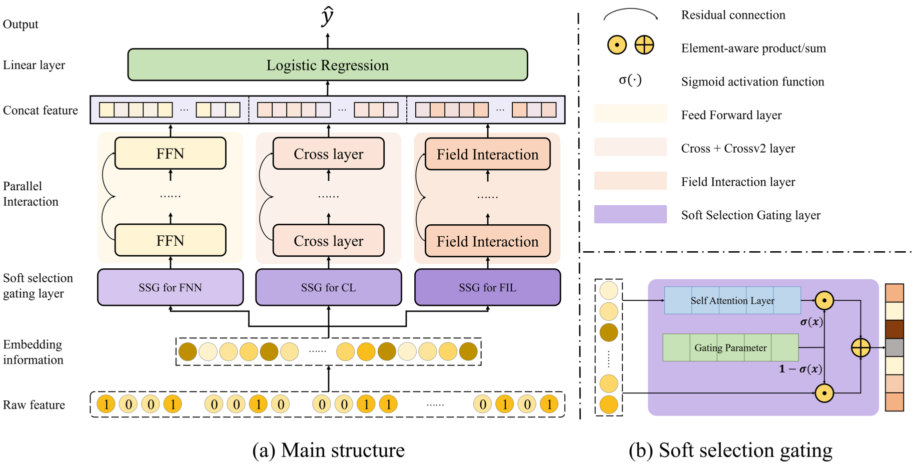
In this paper, we propose a new deep CTR model, named Parallel Heterogeneous Network (PHN). For PHN model, three parallel feature interaction structures were included to analyze CTR features from different perspectives. In order to enhance the independent analysis ability of each parallel module, Soft Select Gating module was constructed after public embedding to enhance the original embedding expression. We also added residual connections with trainable parameters to the model to reduce the weak gradient phenomenon by accumulating gradients during the back propagation process.

This paper mainly contributions are as follow:

- In order to strengthen the expression ability of CTR prediction model, this paper constructed three different linear feature interaction methods from nonlinear interaction, bite-wise interaction and vector-wise interaction based on parallel structure.
- Soft Selection Gating is constructed before the parallel structure, and the features of original embedding are enhanced by self-attention and soft gate structure while retaining the high order crossover characteristics, which improves the ability of the model to express data.
- To solve the weak gradient phenomenon in the parallel model, the residual link with trainable parameters are used in the parallel structure to reinforce the model training process.
- The effectiveness of Soft Selection Gating and the weak gradient phenomenon are visualized, and the effectiveness of PHN is verified by comparison experiments.

2 Proposed Method

The Parallel Heterogeneous Network (PHN) consists of two main structures. One of them is Soft Selection Gating (SSG) module based on self-attention to enhanced embedding features for different structures, and another one is Heterogeneous Interaction Layer (HIL), using different interaction method to analyze the enhanced features, and finally using Logistic Regression to output the confidence of sample. Figure 1(a) illustrates the main structure and the detail of model.



(a) Main structure

(b) Soft selection gating

Fig. 1. The overview structure of our proposed PHN model, which consists of Soft Selection Gating (SSG) module and Heterogeneous Interaction Layer (HIL)

2.1 Heterogeneous Interaction Layer

In PHN, the parallel layers use three kinds of interaction layers to improve model interaction capability: 1) cross layer is the basic part of DCN [16] and DCNV2 [17], which focuses on the element-aware feature interaction; 2) field interaction layer is the basic part of FINT [19], which focuses on the vector-aware field interaction; 3) feed forward layer is used to fitting the non-polynomial information.

Cross Interaction Layer. Feature interaction is a main key point in study of mainly CTR prediction model. As a previous study, the DCN [16] and the DCNV2 [17] proposed two kinds of explicit interaction methods, which achieved the data mode of high-order interaction by realizing the intersection of multi-layer hidden features and original features.

$$y_{dcnv2} = x_0 \odot (W \times x_i + b) + x_i \tag{1}$$

$$y_{dcn} = x_0 \odot x_i^T * w + x_i + b \tag{2}$$

where, x_0 is the input feature of the first cross layer; x_i is the output feature of the i-th cross layer, W and w represent trainable parameter vectors and matrices; $\odot$ is Hadamard product and $\times$ is matrix multiplication. In Eqs. 1 & 2 the DCN and DCNV2 used different parameter forms to interact features, but in general, it achieves element-aware feature interaction. The PHN combines the formulas of DCN and DCNV2 to construct the bit-aware interaction module. As mentioned in Fig. 2, we use the parameter part of the two crossover model and bias to construct cross layer in PHN.

Field Interaction Layer. Besides element-aware interaction, vector-aware interaction is also a key part of the model construction. Field Interaction is

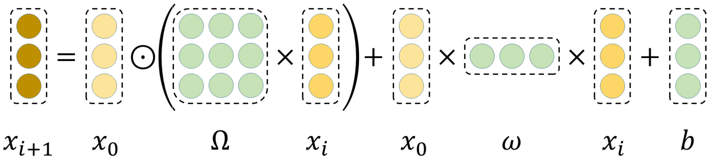


Fig. 2. The calculation diagram of cross layers in PHN

mentioned in FiBiNet [6] and FINT [19], which using the cross method to implement the vector-aware interaction. PHN use the Field Interaction layer in FRNet as a parallel part to enhance the generalization effect of the whole network on the feature crossing pattern.

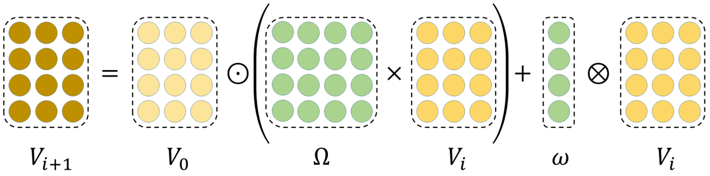


Fig. 3. The calculation diagram of field interaction layers in PHN

As shown in Fig. 3, field interaction layer uses the residual link with trainable parameter vector, which product on different fields feature to screen the output features of the upper layer. In the subsequent experiments, we will discuss and experiment residual link forms in all parallel layers.

Feed Forward Layer. The third part of parallel network is composed of Feed Forward Network. By alternating linear and nonlinear analysis of the original features, FFN complements the analysis of the previous two crossover modes to improve the overall network generalization function.

$$x_{i+1} = \sigma(\omega x_i + b) \tag{3}$$

where, σ is the activation function, which is LeakyReLU in PHN.

2.2 Soft Selection Gating

The ideal of multiple structure parallelism raise up a new question: whether different structures require different input dense feature. The traditional Multi-head self-attention(MSA) mechanism [14] used weight based query vector and key vector to aggregate information in a sequence, which was an ideal method to process feature information.

$$Q_E,\ K_E,\ V_E = W_Q E,\ W_K E,\ W_V E \tag{4}$$

$$MSA(Q_E, K_E, V_E) = Softmax(\frac{Q_E K_E}{\sqrt{d_k}})V_E \tag{5}$$

From the Eqs. 4 & 5, MSA has considered different field weights through the second-order intersection of query vector Q_E and key vector K_E. However, based on the feature interaction in the CTR prediction model, the direct using by the traditional MSA may over-focus on the feature activation value of the second-order crossover, thus losing the performance of the feature at the higher-order crossover. Inspired by the FRNet [15], an information selecting method named Soft Selection Gating (SSG) is used after the sharing embedding $E \in R^n$ in PHN. This soft-gating information selection is designed for choosing activation between MSA result and sharing raw embedding.

$$E_{sg} = G_{sg} \odot E_{sa} + [I - G_{sg}] \odot E_{se} \tag{6}$$

where $G_{sg} \in R^n$ is the trainable gating vector, $E_{sa} \in R^n$ is the sharing self-attention embedding, $E_{se} \in R^n$ is the sharing embedding, and $I \in R^n$ is a unit vector. As the Fig. 1(b) shows, the SSG considers both sharing embedding and self-attention embedding. By using weighting parameters, the model can select the raw feature or the feature enhanced by MSA for different parallel structures. Subsequent experiments will discuss whether to share the weight of self-attention and the gating mechanism, and confirm the effectiveness of SSG.

2.3 Weak Gradient Problem

Basic on the CTR prediction task definition, the key point of improving AUC value is increasing the confidence of positive label samples and decreasing the confidence of negative label samples, which makes model more robust. In the last stage of CTR prediction model, the traditional model usually constructs confidence coefficient of click by using Sigmoid activation function.

The Fig. 4 shows that, we can think of the entire Sigmoid function as two interval, the effective gradient interval (blue) with a normal gradient, and the Weak gradient interval (red) with a gradient approaching zero. When the output value of the parallel model is accumulated at the last linear layer, it is easy to make the samples originally in the effective gradient interval to fall into the Weak gradient interval, thus weakening the learning of each part for valid samples. In this paper, this phenomenon is referred to weak gradient in parallel structures.

To mitigate this phenomenon and achieve effective training, the PHN using residual link in each substructure, enhancing the gradient accumulation of each parallel structure in the process of back propagation. To further accommodate this phenomenon, we also tried to add gating parameters to residual links and used batch normalization of different modes in the final linear layer. We will discuss this further in Sect. 3.

3 Experiments

In this section, we evaluate PHN on two benchmark data sets. We aim to answer the following research questions:

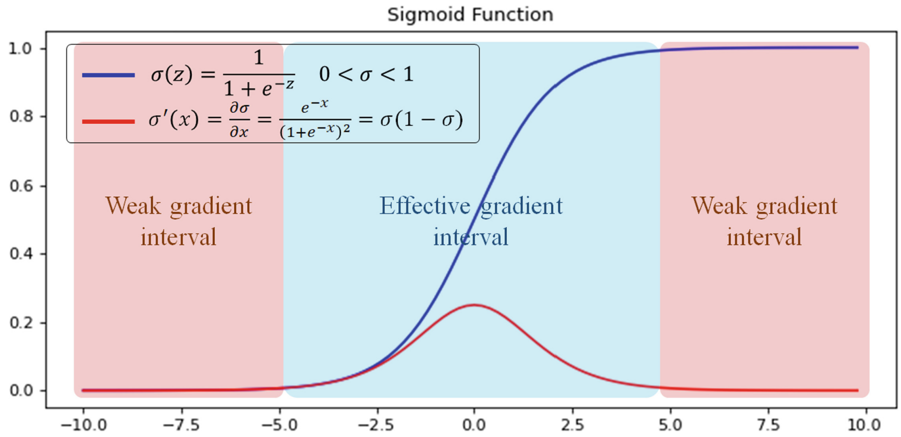


Fig. 4. Gradients of different interval in Sigmoid (Color figure online)

- **RQ1:** Will parallel structure-based PHN perform better than previous CTR prediction models over different classical data sets?
- **RQ2:** Under what circumstances can the Soft Selection Gating reasonably enhance the function of feature representation.
- **RQ3:** Is the parallel structure actually caused the weak gradient phenomenon, and the problem is effectively alleviated by residual connection or batch normalization?

3.1 Datasets Description

To evaluate the effectiveness of the model in this paper, two benchmark off-line datasets are selected for experiment: Criteo[1] and Avazu[2]. The two data sets were divided into training, validation and testing set according to the ratio of 8:1:1. Detailed information on the two benchmark datasets is shown in Table 1.

Table 1. Statistics of the benchmark datasets

Dataset	Sample size	Fields	Features	Positive ratio
Criteo	45,840,618	39	1,086,810	25.6%
Avazu	40,428,966	23	1,544,257	16.9%

For the CTR prediction task, two classical evaluation metrics such as: Logloss and AUC, were used to verify the effective generalization and robustness of the suggested model.

[1] https://www.kaggle.com/c/criteo-display-ad-challenge.

[2] https://www.kaggle.com/c/avazu-ctr-prediction.

3.2 Compared Models

To verify the effectiveness of the proposed PHN model, we compare it with linear model (LR, FM [10], FwFM [8], FmFM [13]), deep model (DNN, W&D [2], DeepFM [4], xDeepFM [7], AutoInt [12]), and interaction model (DCN [16], DCNV2 [17], FiBiNet [6], FINT [19]) on CTR task. All models and experiments are implemented on Huawei FuxiCTR deep learning framework [20].

3.3 Performance Comparison (RQ1)

Effectiveness of PHN. To verify the validity of the model, we followed the original structure of each comparison model, controlled the interaction layer of all models in three layers, and recorded the testing results of each model on two benchmark datasets. The specific experimental results were reported in Table 2.

Table 2. Experiment result of different CTR prediction models on Criteo and Avazu

Model	Criteo			Avazu		
	Logloss	AUC	AUC Impv.	Logloss	AUC	AUC Impv.
LR	0.457334	0.792831	–	0.382039	0.777148	–
FM	0.450260	0.801086	1.041%	0.378750	0.782448	0.681%
FwFM	0.442566	0.809314	2.079%	0.373862	0.790315	1.694%
FmFM	0.444253	0.807395	1.833%	0.376521	0.785998	1.139%
DNN	0.442271	0.809547	2.108%	0.372686	0.792553	1.982%
W&D	0.442627	0.809133	2.056%	0.372663	0.792079	1.921%
DCN	0.442382	0.809390	2.089%	0.372884	0.791767	1.881%
DCNV2	0.440825	0.811139	2.309%	0.372511	0.792352	1.956%
DeepFM	0.444391	0.807686	1.911%	0.372202	0.792856	2.021%
xDeepFM	0.444541	0.807728	1.878%	0.373387	0.791503	1.847%
AutoInt	0.442502	0.809237	2.069%	0.372830	0.791918	1.900%
FiBiNET	0.442335	0.809809	2.141%	0.371139	0.794850	2.278%
FINT	0.442471	0.808409	1.965%	0.372808	0.792043	1.917%
PHN (ours)	**0.439927**	**0.812039**	**2.383%**	**0.370481**	**0.795964**	**2.421%**

The experiment shows that, PHN achieved a good performance in both validation and testing experiments with two benchmark large datasets.

Grid Search. Figure 5 shows the grid search experiment result of PHN. With the increase of the number of cross layers, the robustness of the model also increases, which may benefit from the improvement of the expression ability of cross layers for higher-order crosses. However, as the number of layers increases, the model complexity also increase, which slow down the training process of the model. The values of AUC and Logloss shown in Fig. 5 tend to be stable when the number of cross layers is five.

3.4 Selection Information (RQ2)

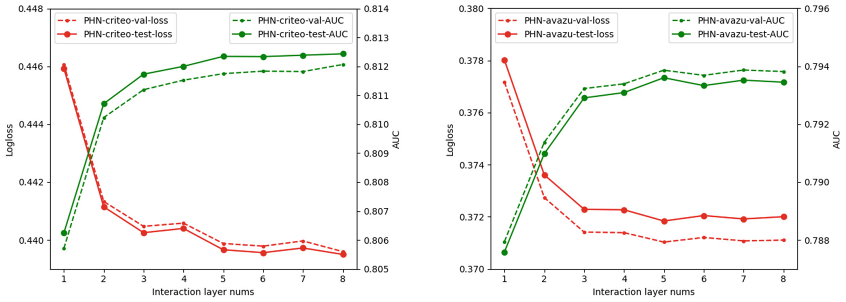
Data Skew Visualization. Based on the trained PHN structure, we visualized the tensor amplification ratio after SSG output. As shown in Fig. 6, the characteristics of the three cross-layers have some similarity, such as the high proportion of field 13 and the low proportion of field 39. At the same time, the feature scaling of the three parts is somewhat different, as in field 8 and field 24.

Selection Pattern. The SSG module is designed for enhance the representation of embedding feature, which select the feature from raw embedding and self-attention embedding. Depending on the design, the selection pattern of self-attention layer and gating layer in this module can be classified as public or private. To further validate the effectiveness of SSG, we also conducted a comparison experiment. Different subscripts represent different selection patterns: *embed* (public embedding feature), *sa* (public self-attention), *sg* (public soft gating). the subscripts with a prefix "*P*" means that the PHN contains private layers for each parallel layer. Table 3 shows the results of comparison experiment.

This experiments show that, single self-attention layer (public of private) cannot replace the embedding feature represent, but it can help to enhance the feature by using the soft selection gating, and the AUC value of the algorithm increases by 0.123% on average. From a theoretical point of view, a feature without a high activation value in the first-order feature cannot be completely transferred, because it may show a high activation value in the high-order interaction with other features.

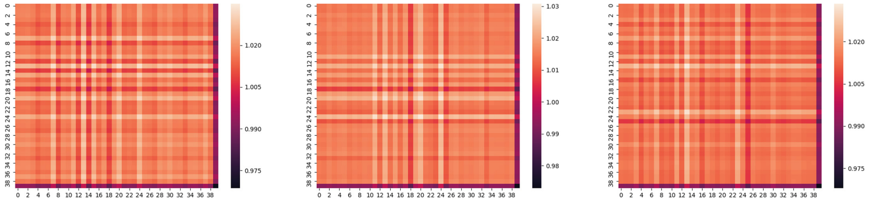
3.5 Weak Gradient Phenomenon (RQ3)

Efficiency Analysis. To reduce the impact of weak gradient phenomenon, there has been an attempt in PHN to enhance the data gradient flow in training through residual links (RL) or batch normalization (BN) to reduce the training pressure of each parallel part. In experiments, we tried to introduce gating parameters for RL and discussed the independence of BN in different parallel modules in the last linear layer.



(a) Grid search experiment on Criteo (b) Grid search experiment on Avazu

Fig. 5. The grid search performance of PHN with different interaction layers on two benchmark datasets.



(a) FFN layer (b) Cross layer (c) Field interaction layer

Fig. 6. The heatmap of different parallel layers input feature scaling ratio

Different model subscripts represent different structures in PHN: *base* (basic model), *rl* (normal RL), *Prl* (parameter RL), *bn* (BN), and *pbn* (private BN). Table 4 shows the result of comparison experiments.

The experiments show that, In the back propagation process, the weak gradient problem can be improved by using gradient accumulation in the RL. In the

Table 3. Experiment result of different schemes of information selection module on Criteo and Avazu

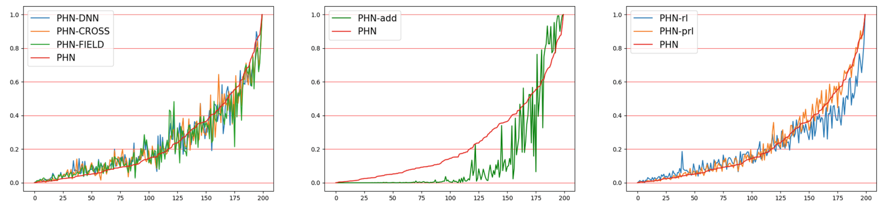
Model	Criteo		Avazu	
	Logloss	AUC	Logloss	AUC
PHN_{embed}	0.440543	0.811647	0.371538	0.794388
PHN_{sa}	0.441301	0.810525	0.373018	0.792504
PHN_{Psa}	0.441445	0.810554	0.372369	0.792640
PHN_{sa+sg}	0.440210	0.811782	0.371608	**0.794622**
PHN_{Psa+sg}	**0.440031**	**0.811902**	**0.371351**	0.794570
PHN_{sa+Psg}	0.440256	0.811771	0.371398	0.794385
$PHN_{Psa+Psg}$	0.440692	0.811595	0.371405	0.794458

Table 4. Experiment result of different solutions to the weak gradient problem on Criteo and Avazu

Model	Criteo		Avazu	
	Logloss	AUC	Logloss	AUC
PHN_{base}	0.440034	0.811914	0.372209	0.793206
PHN_{rl}	0.439763	0.812111	0.372294	0.793044
PHN_{prl}	**0.439540**	**0.812428**	0.372084	0.793392
$PHN_{base+bn}$	0.440111	0.811879	**0.371117**	**0.795087**
PHN_{rl+bn}	0.441548	0.810359	0.372410	0.793084
PHN_{prl+bn}	0.440307	0.811711	0.371189	0.794911
$PHN_{base+pbn}$	0.443966	0.811865	0.373950	0.794839
PHN_{rl+pbn}	0.445333	0.809268	0.379022	0.791024
$PHN_{prl+pbn}$	0.444590	0.811813	0.377631	0.794621

case of parameters, the overall network can better fit the data flow in the feed forward and reverse process, and strengthen the fitting effect of different parallel structures. However, the subsequent addition of BN has not been very effective. This may be due to the uneven distribution of data flows in different parallel structures, but forced unification with normalization weakens the representation of data. This also explains to some extent that BN layer is not separable from linear layer. The last two groups of experiments also showed that the specificity of the data stream fitted was enhanced when the RL strengthened different parallel structures, while BN had certain side effects. Therefore, in PHN, the RL and BN had better be realized in an independent parallel structure.

Visualization of Activation Value. A more robust model should output more closely to the confidence of the label worthiness. Figure 7 shows the confidence curve of PHN in 200 samples after training one epoch in different configurations, to show the changes of different structures during training phase.



(a) Independent structure (b) Add results (c) with residual link

Fig. 7. Experiments on the last activation value of PHN. (Color figure online)

The red line in Fig. 7 represents the PHN model without RL and BN as the benchmark. Figure 7(a) shows that, the single interaction module showed higher negative confidence and lower positive confidence than PHN. This means that PHN is superior to partial cross structure in sample resolution. Figure 7(b) shows that, the sigmoid calculation after summing up the activation values of the parallel models can show more robustness than PHN, which means the fitting effect of a single PHN model on the data set is weakened by weak gradient phenomenon. Figure 7(c) shows that, RL can enhance the high confidence of negative samples, but also reduce the confidence of positive samples, and the RL with parameters can effectively improve the performance of the model on the PHN infrastructure.

4 Conclusion

In this paper, we described the parallel structure of the current mainstream CTR model and the weak gradient phenomenon in the parallel structure, and introduce a parallel structure model named Parallel Heterogeneous Network (PHN) in response to these phenomena. PHN model used Soft Selecting Gating (SSG) structure to isomerize features, and used Feed Forward network, cross interaction layers and field interaction layers to build the subsequent parallel part. The performance experiment results show that PHN shows the State of the Art on two large benchmark data sets, and explores the interaction layer num of the model. The comparative experimental results show that SSG can effectively improve the representation based on public embedding, and the residual link with trainable parameters can improve the representation ability of the model while maintaining the robustness of the results. Based on the overall experimental results, this work brings us to one step closer to being able to determine the optimal structure of PHN.

Acknowledgements. This study was supported by the Hunan Province Natural Science Foundation (grant number 2022JJ30673).

References

1. Chen, B., et al.: Enhancing explicit and implicit feature interactions via information sharing for parallel deep CTR models. In: Proceedings of the 30th ACM International Conference on Information and Knowledge Management, pp. 3757–3766 (2021)
2. Cheng, H.-T., et al.: Wide & deep learning for recommender systems. In: Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, pp. 7–10 (2016)
3. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. **2001**, 1189–1232 (2001)
4. Guo, H., Tang, R., Ye, Y., Li, Z., He, X.: DeepFM: a factorization-machine based neural network for CTR prediction. arXiv preprint arXiv:1703.04247 (2017)
5. He, X., Chua, T.-S.: Neural factorization machines for sparse predictive analytics. In: Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval, pp. 355–364 (2017)

6. Huang, T., Zhang, Z., Zhang, J.: FiBiNET: combining feature importance and bilinear feature interaction for click-through rate prediction. In: Proceedings of the 13th ACM Conference on Recommender Systems, pp. 169–177 (2019)
7. Lian, J., Zhou, X., Zhang, F., Chen, Z., Xie, X., Sun, G.: xdeepfm: Combining explicit and implicit feature interactions for recommender systems. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1754–1763 (2018)
8. Pan, J., et al.: Field-weighted factorization machines for click-through rate prediction in display advertising. In: Proceedings of the 2018 World Wide Web Conference, pp. 1349–1357 (2018)
9. Qu, Y., et al.: Product-based neural networks for user response prediction. In: 2016 IEEE 16th International Conference on Data Mining (ICDM), pp. 1149–1154. IEEE (2016)
10. Rendle, S.: Factorization machines. In: 2010 IEEE International Conference on Data Mining, pp. 995–1000. IEEE (2010)
11. Shan, Y., Hoens, T.R., Jiao, J., Wang, H., Yu, D., Mao, J.C.: Deep crossing: web-scale modeling without manually crafted combinatorial features. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 255–262 (2016)
12. Song, W., et al.: Autoint: automatic feature interaction learning via self attentive neural networks. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1161–1170 (2019)
13. Sun, Y., Pan, J., Zhang, A., Flores, A.: FM2: field-matrixed factorization machines for recommender systems. In: Proceedings of the Web Conference 2021, pp. 2828–2837 (2021)
14. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
15. Wang, F., et al.: Enhancing CTR prediction with context-aware feature representation learning. arXiv preprint arXiv:2204.08758 (2022)
16. Wang, R., Fu, B., Fu, G., Wang, M.: Deep & cross network for ad click predictions. In: Proceedings of the ADKDD 2017, pp. 1–7 (2017)
17. Wang, R., et al.: DCN V2: improved deep & cross network and practical lessons for web-scale learning to rank systems. In: Proceedings of the Web Conference 2021, pp. 1785–1797 (2021)
18. Zhang, W., Du, T., Wang, J.: Deep learning over multi-field categorical data. In: Ferro, N., et al. (eds.) ECIR 2016. LNCS, vol. 9626, pp. 45–57. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-30671-1_4
19. Zhao, Z., Yang, S., Liu, G., Feng, D., Xu, K.: FINT: field-aware INTeraction neural network For CTR prediction. arXiv preprint arXiv:2107.01999 (2021)
20. Zhu, J., Liu, J., Yang, S., Zhang, Q., He, X.: Fuxictr: an open benchmark for click-through rate prediction. arXiv preprint arXiv:2009.05794 (2020)