DeepFM: A Factorization-Machine based Neural Network for CTR Prediction

Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, Xiuqiang He,Shenzhen Graduate School, Harbin Institute of Technology, China, Noah’s Ark Research Lab, Huawei, China

Group Number: 47

Group Members: 蘇桓平　P76071462

Motivation

- Learning sophisticated feature interactions behind user behaviors is critical in maximizing CTR for recommender systems

- Existing methods seem to have a strong bias towards low- or high-order interactions, or require expertise feature engineering

- Online advertising is important to improve revenue

Objective

- end-to-end learning model that emphasizes both low and high order feature interactions
- combines factorization machines (FM) and deep neural network (DNN)
- Reduce training time
- Improve advertise click through rate
- Learn implicit feature interactions behind user click behaviors

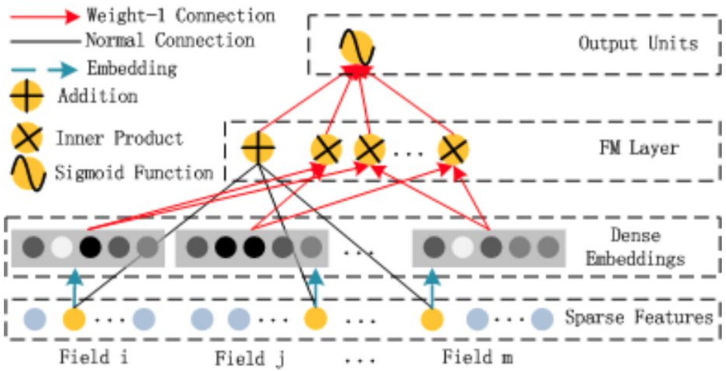
Contribution

- **DeepFM** introduce FM and DNN, use both strength to learn both low-order and high-order data
- **FM** run faster than LR, and able to estimate parameters under huge sparsity
- **DNN** can learn 3 or more feature interaction in data

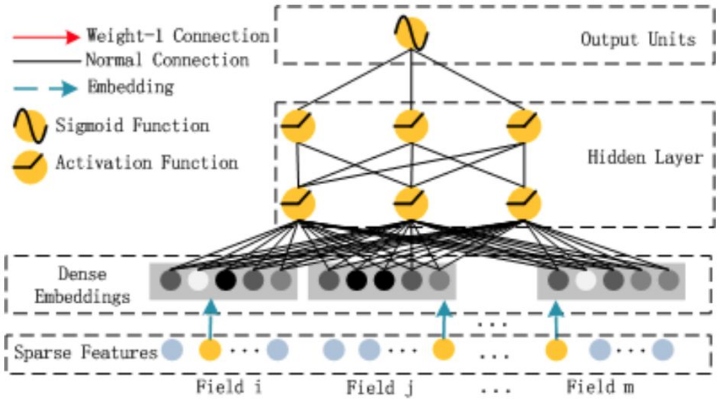
System Framework: Training Process

1. Download dataset from [Criteo dataset]
2. Use first 1M data for training
3. 60% for training, 20% for validating, 20% for testing
4. For numerical features,normalzied to continous values
5. For categorical features, removed long-tailed data appearing less than 200 times.
6. Feature embedding
7. Feed data to FM & DNN to input
8. Train FM & DNN
9. Sigmod(FM + DNN output)

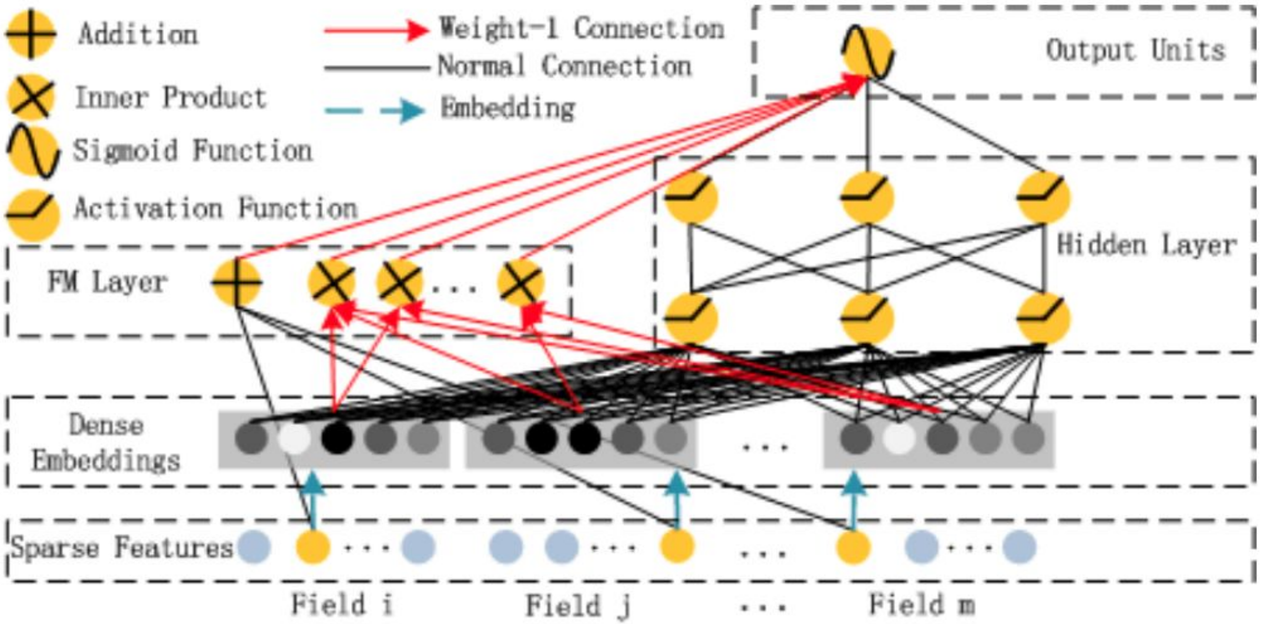
Deep Learning Architecture



FM



DNN



$$\hat{y} = sigmoid(y_{FM} + y_{DNN}),$$

1. DNN

$$a^{(0)} = [e_1, e_2, \ldots, e_m]$$

$$a^{(l+1)} = \sigma(W^{(l)} a^{(l)} + b^{(l)})$$

$$y_{DNN} = \sigma(W^{|H|+1} \cdot a^{|H|} + b^{|H|+1})$$

2. FM

$$\begin{aligned}
&\sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j && (1) \\
&= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j - \frac{1}{2} \sum_{i=1}^{n} \langle \mathbf{v}_i, \mathbf{v}_i \rangle x_i x_i && (2) \\
&= \frac{1}{2} \left(\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{f=1}^{k} v_{i,f} v_{j,f} x_i x_j - \sum_{i=1}^{n} \sum_{f=1}^{k} v_{i,f} v_{i,f} x_i x_i \right) && (3) \\
&= \frac{1}{2} \sum_{f=1}^{k} \left(\left(\sum_{i=1}^{n} v_{i,f} x_i \right) \cdot \left(\sum_{j=1}^{n} v_{j,f} x_j \right) - \sum_{i=1}^{n} v_{i,f}^2 x_i^2 \right) && (4) \\
&= \frac{1}{2} \sum_{f=1}^{k} \left(\left(\sum_{i=1}^{n} v_{i,f} x_i \right)^2 - \sum_{i=1}^{n} v_{i,f}^2 x_i^2 \right)
\end{aligned}$$

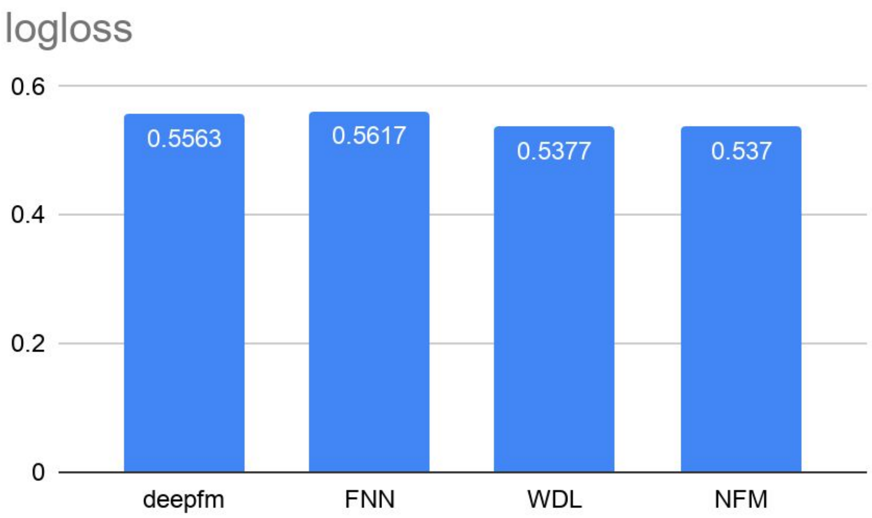
3. sigmod

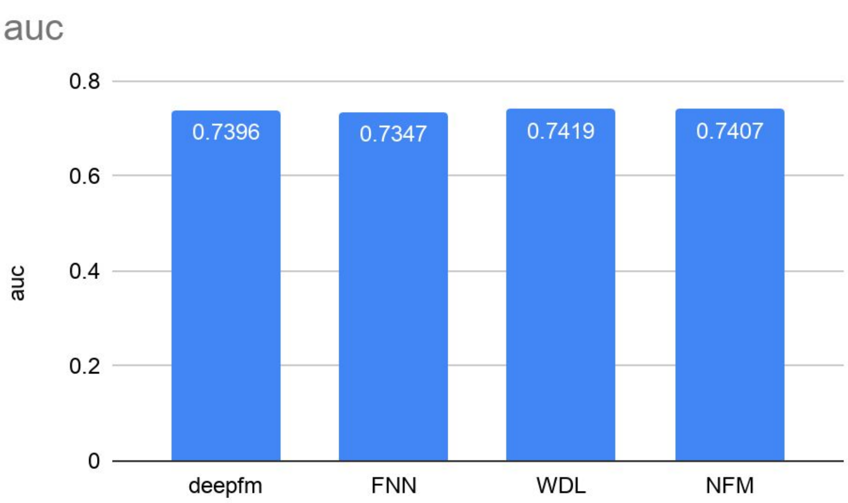
$$\hat{y} = sigmoid(y_{FM} + y_{DNN}),$$

Experimental Result

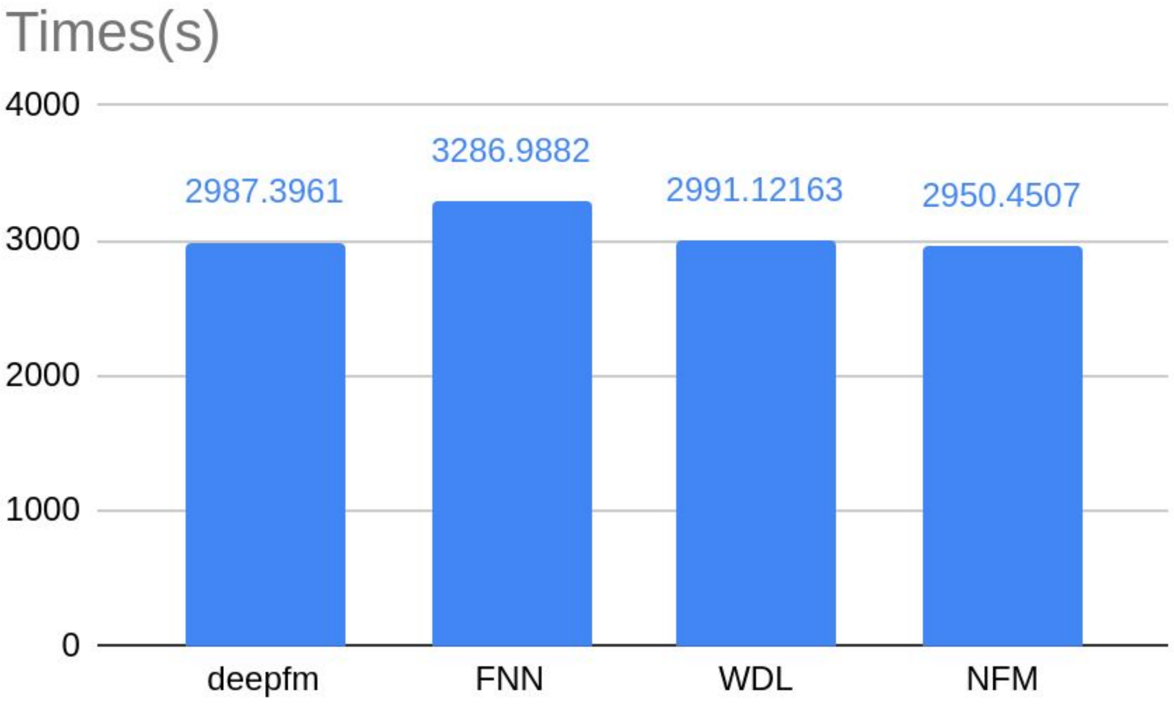
	AUC	LogLoss	Times(s)
DeepFM	0.7396	0.5563	**2987.3961**
FNN	0.7347	0.5617	3286.9882
WDL	**0.7419**	0.5377	2991.12163
NFM	0.7407	**0.537**	3100.4507

Logloss vs AUC





Times


Times(s)
4000
3000
2000
1000
0
2987.3961
3286.9882
2991.12163
2950.4507
deepfm
FNN
WDL
NFM

- Correct
    1. Deepfm run the most efficiency than any others

- Incorrect
    1. DeepFM’s auc is lower than any others, we only use 1M rows data, far less than 45M rows data that paper uses
    2. DeepFM’s auc only has 0.73, less than 0.8715 in paper, we should use more data to get more accuracy

Conclusion

- Compare DeepFM and other ctr model, the evaluation result indicates that DeepFM is more effective than any state-of-the-art model.
- Combine FM and DNN model to achive better performance
- DeepFM learn both low-order and high-order feature
- Without model pre-training
- Without feature engineering

Conclusion

	No-pretraining	High-order Feature	Low-order Feature	No Feature Engineering
FM	√	×	√	√
FNN	×	√	×	√
PNN	√	√	×	√
Wide & Deep	√	√	√	×
DeepFM	√	√	√	√