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

**Author: Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, Xiuqiang He**

**Conference: IJCAI**

**Year: 2017**

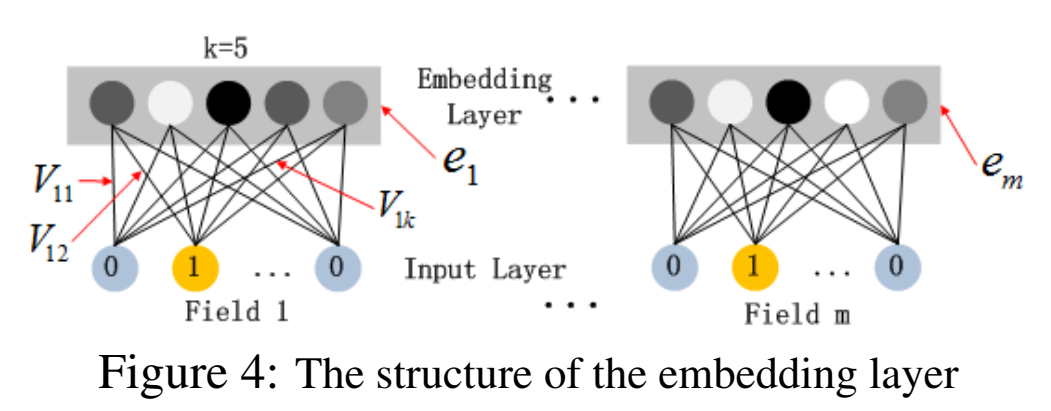
*Learning sophisticated feature interactions behind user behaviors is critical in maximizing click-through rate (CRT) for recommender systems. Existing methods seem to have a strong bias towards low- or high-order interactions, or require expertise feature engineering. The proposed model, DeepFM, combines the power of factorization machines for recommendation and deep learning for feature learning in a new neural network architecture. It emphasizes both low- and high-order feature interactions and requires no pre-training and no feature engineering. And the performance of DeepFM on both benchmark data and commercial data shows consistent improvement over existing models for CTR prediction.*

The prediction of click-through rate (CTR) is critical in recommender system, where the task is to estimate the probability a user will click on a recommended item. The key challenge is in effectively modeling feature interactions. Some feature interactions can be easily understood, thus can be designed by experts. However, most other feature interactions are hidden in data and difficult to identify a priori (for instance, the classic association rule “diaper and beer” is mined from data, instead of discovering by experts), which can only be captured automatically by machine learning. Even for easy-to-understand interactions, it seems unlikely for experts to model them exhaustively, especially when the number of features is large.

DeepFM integrates the architectures of FM and deep neural networks (DNN). It models low-order feature interactions like FM and models high-order feature interactions like DNN. Unlike the wide & deep model, DeepFM can be trained end-to-end without any feature engineering. It consists of two components, FM component and deep component, that share the same input. All parameters are trained jointly for the combined prediction model:

,

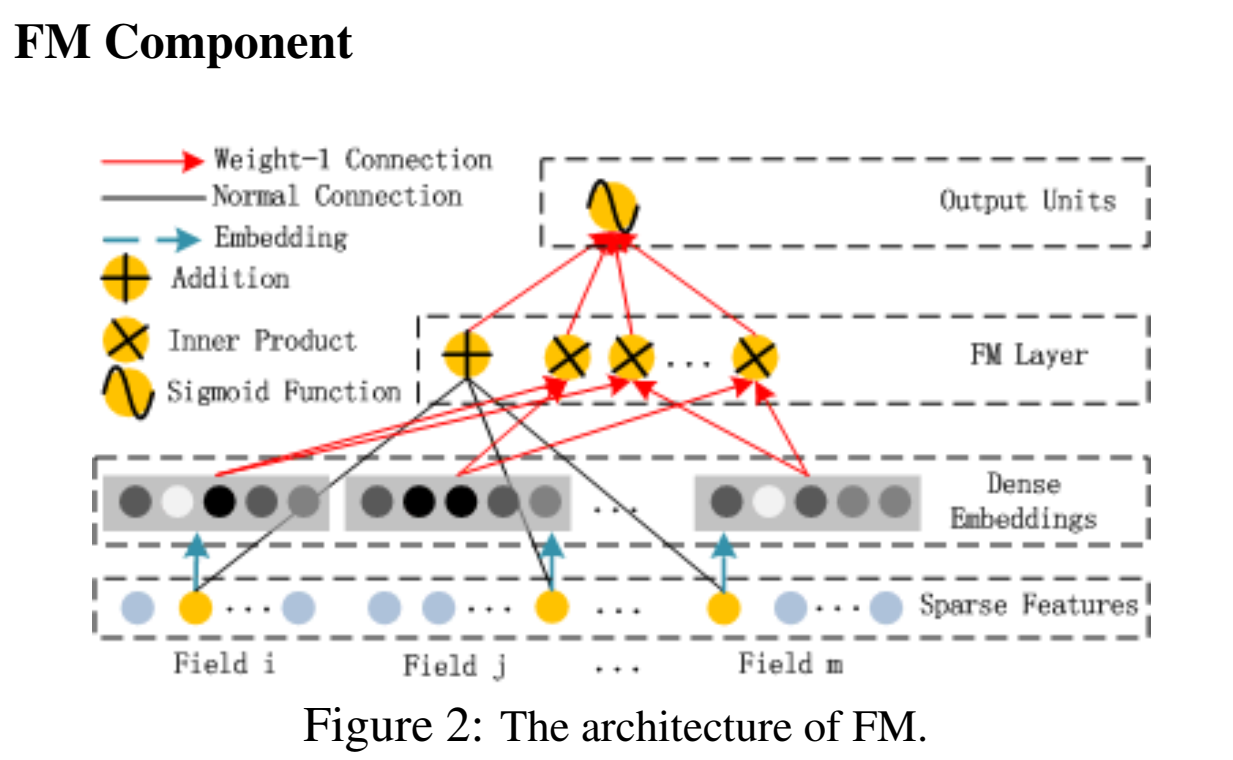
where is the predicted CTR, is the output of FM component, and is the output of deep component.



The raw feature input vector for CTR prediction is usually highly sparse, super high-dimensional, categorical-continuous-mixed, and grouped in fields (e.g., gender, location, age). This suggests an embedding layer to compress the input vector to a lowdimensional, dense real-value vector before further feeding into the first hidden layer. While the lengths of different input field vectors can be different, their embeddings are of the same size ( is a hyper-parameter). Denote the output of the embedding layer as:

,

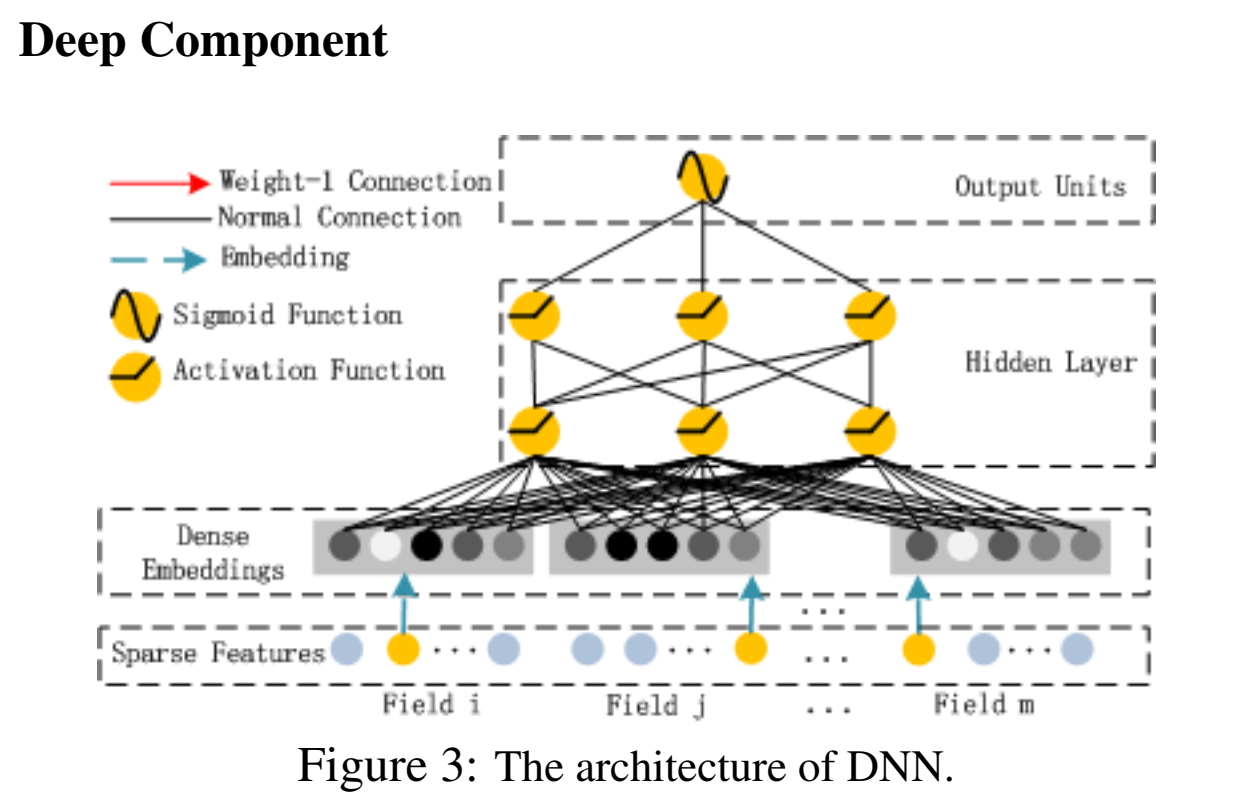
where is the embedding of -th field and is the number of fields.



The FM component is a factorization machine. Besides a linear (order-1) interactions among features, FM models pairwise (order-2) feature interactions as inner product of respective feature latent vectors. In previous approaches, the parameter of an interaction of features and can be trained only when feature and feature both appear in the same data record. While in FM, it is measured via the inner product of their latent vectors and . The output of FM is the summation of an Addition unit and a number of Inner Product units:

,

where and ( is a hyper-parameter). The Addition unit () reflects the importance of order-1 features, and the Inner Product units represent the impact of order-2 feature interactions.



The deep component is a feed-forward neural network, which is used to learn high-order feature interactions. As shown in Figure 3, , the output of the embedding layer, is fed into the deep neural network, and the forward process is:

where is the layer depth and σ is an activation function. , , are the output, model weight, and bias of the-th layer. After that, a dense real-value feature vector is generated, which is finally fed into the sigmoid function for CTR prediction:

where |H| is the number of hidden layers.

The performance of DeepFM on both benchmark data and commercial data shows consistent improvement over existing models for CTR prediction. In this paper, the efficiency of different models is compared on Criteo dataset. The tests are carried on CPU and GPU. The DeepFM achieves almost the most efficient in both tests. The effectiveness for CTR prediction of different models are carried on Criteo dataset and Company∗ dataset. As the best model, DeepFM outperforms LR by 0.86% and 4.18% in terms of AUC (1.15% and 5.60% in terms of Logloss). Compared to the second-best model, DeepFM achieves more than 0.37% and 0.25% in terms of AUC (0.42% and 0.29% in terms of Logloss).

Since DeepFM trains a deep component and a FM component jointly. It gains performance improvement from the following advantages. First, it does not need any pre-training. Second, it learns both high- and low-order feature interactions. Besides, it introduces a sharing strategy of feature embedding to avoid feature engineering. The disadvantages of the model mainly lie on the FM component. FM modify the interaction between two features using the same latent vector. When feature interacts with or , is used to simutaneously modify the interactions, which is not concise enough.

There are some interesting directions for future study. One is exploring some strategies (such as introducing pooling layers) to strengthen the ability of learning most useful high-order feature interactions. Another is to train DeepFM on a GPU cluster for large-scale problems. And another is to find the best hyper-parameters.

By

许俊杰 16319058

黄俊凯 16340082

陈亚楠 16340041