Deep-learning Based Models for Recommender Systems

Wide & Deep Learning for Regression/Classification Problems

Xinyi Li

2019 年 8 月 21 日

Overview

Deep Cross

Factorisation Machine supported Neural Network (FNN)

Product-based Neural Network (PNN)

Wide & Deep Learning

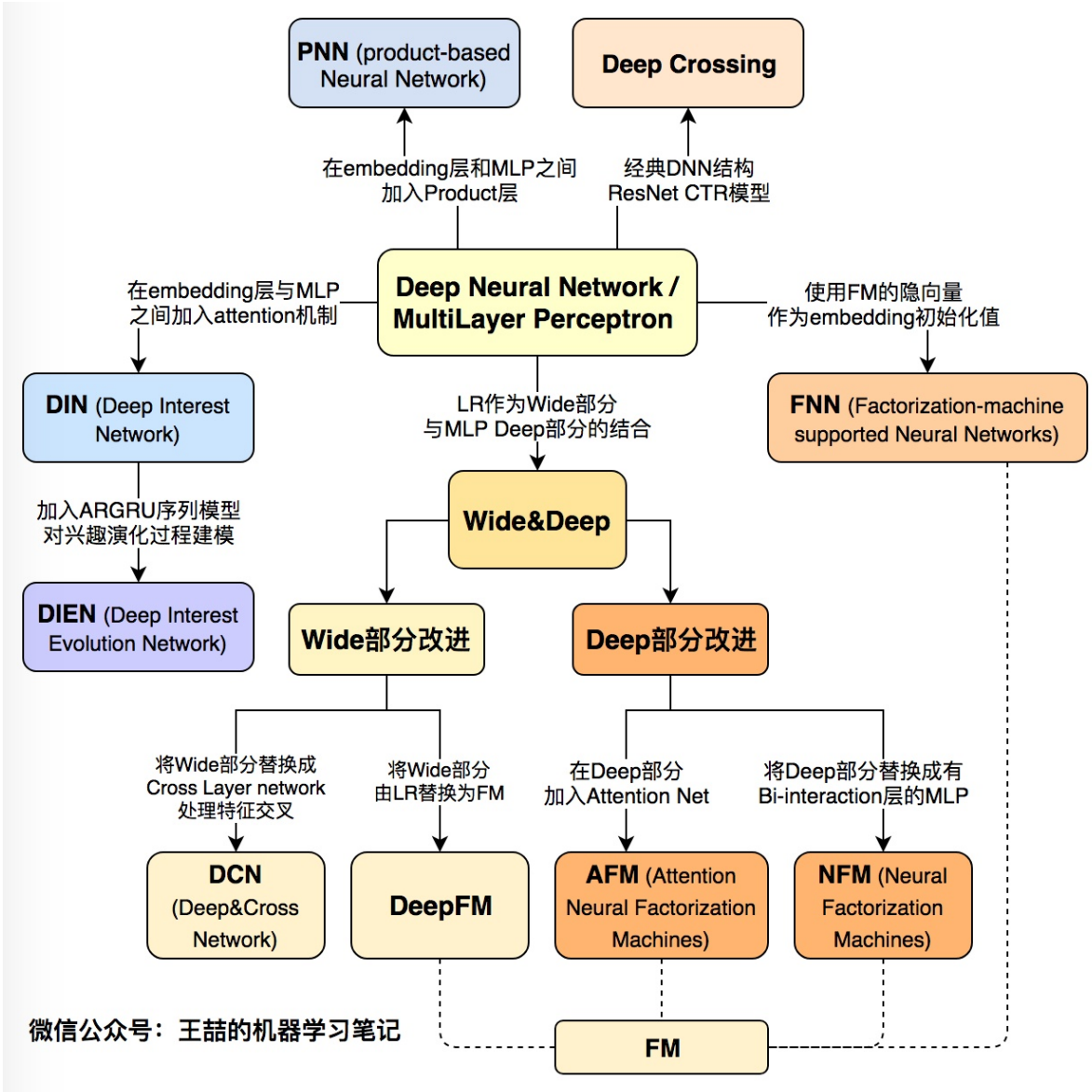
DeepFM

Deep&Cross Network (DCN)

Attentional Factorization Machines (AFM)

Implements

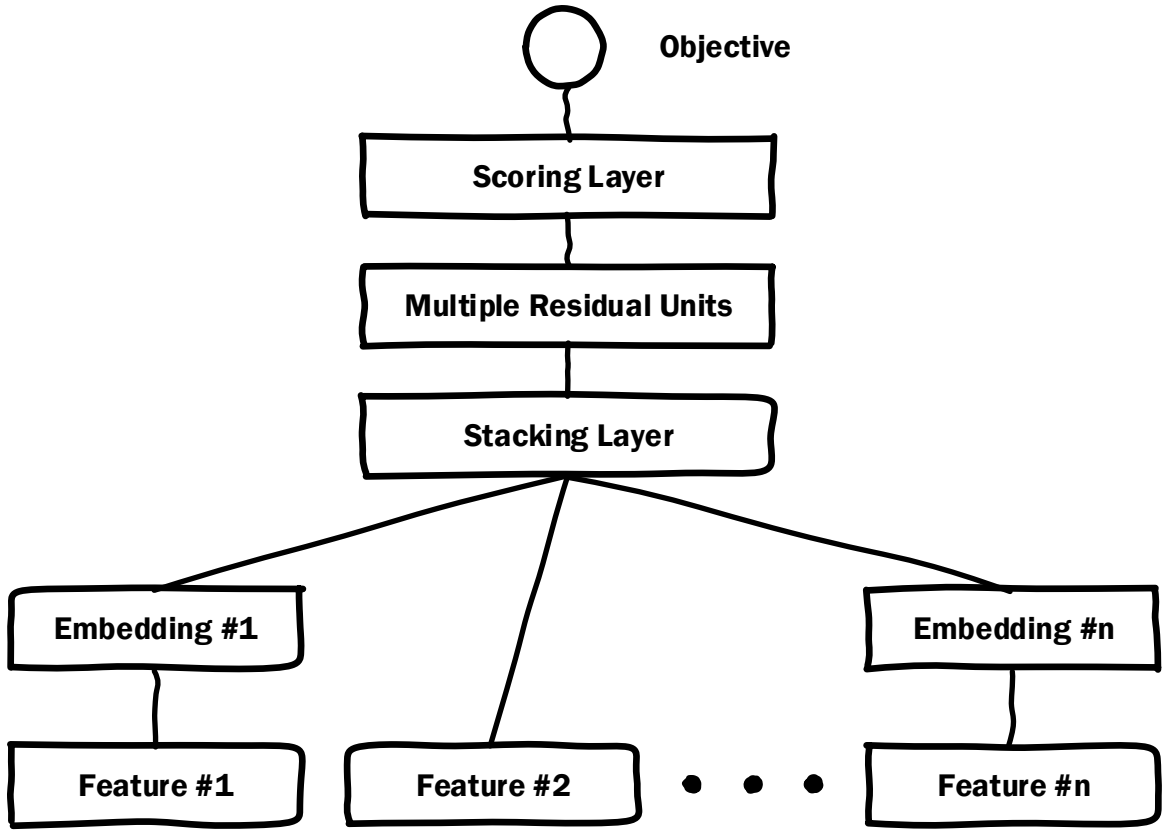
演化图谱[1]



[1]王喆. 王喆的机器学习笔记. URL: https://www.zhihu.com/question/20830906/answer/681688041.

Base Model: Deep Crossing[2]

Microsoft



[2]Ying Shan et al. "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*. ACM. 2016, pp. 255–262.

Deep Crossing

Scoring layer

- objective: logloss
- sigmoid/softmax

Embedding layer

高维稀疏特征 (id 类,one-hot encode) $\rightarrow$ 低维稠密特征

$$\mathbf{W}_j : (m_j \times n_j), \quad m_j < n_j$$

ReLu

$$X_j^o = \max(\mathbf{0}, \mathbf{W}_j X_j^l + \mathbf{b}_j)$$

Deep Crossing

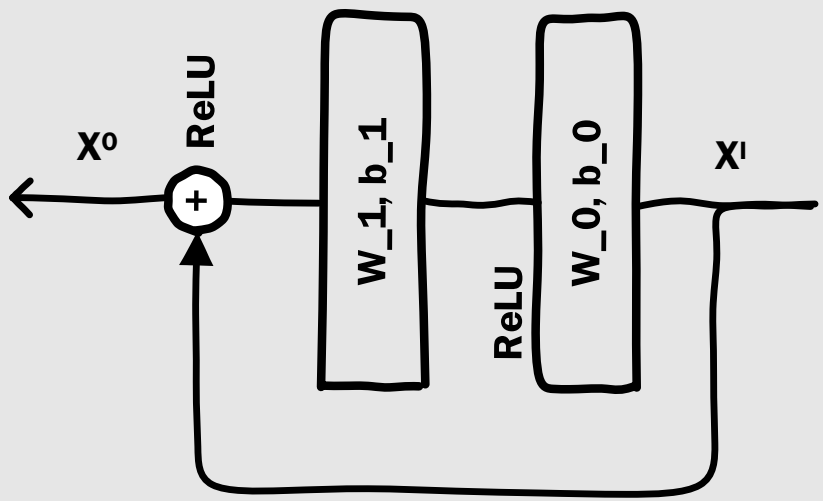
Stacking layer

concatenate: $X^O = [X_0^O, X_1^O, \cdots, X_K^O]$

Residual layer

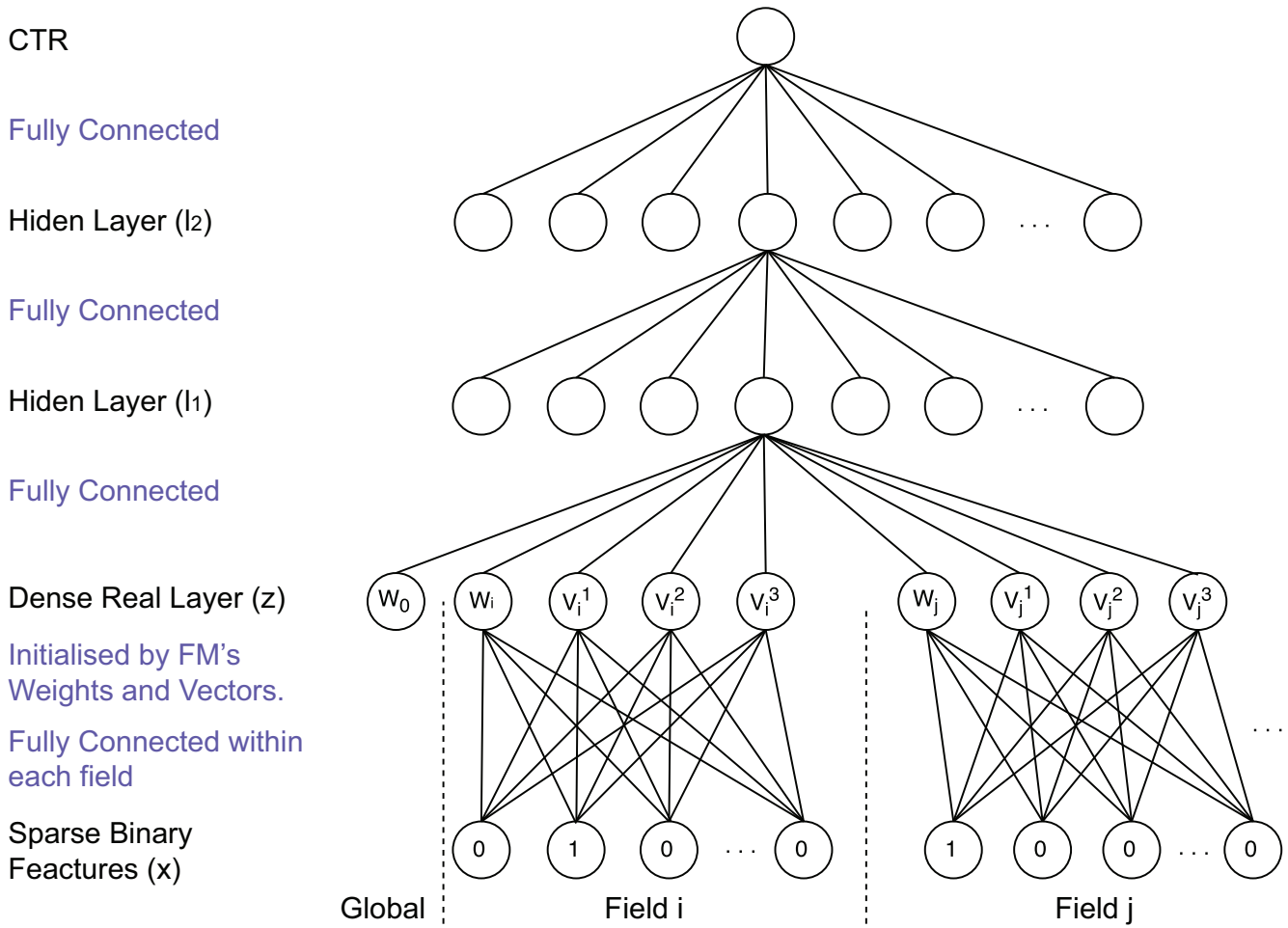
2 layers ReLu transform

$$X^O = \mathcal{F}(X^I, \{\mathbf{W}_0, \mathbf{W}_1\}, \{\mathbf{b}_0, \mathbf{b}_1\}) + X^I$$



FNN

Factorisation Machine supported Neural Network[3]



[3]Weinan Zhang, Tianming Du, and Jun Wang. "Deep learning over multi-field categorical data". In: *European conference on information retrieval*. Springer. 2016, pp. 45–57.

FNN

Hideen layers: l_1,l_2

$l_i = \tanh(\mathbf{W}_i l_{i-1} + \mathbf{b}_i)$

pre-train with FM

map 2 features into vectors in a low-rank latent space $\rightarrow$ interactions

$$y\text{FM}(\mathbf{x}) := \textit{sigmoid}(w_0 + \sum_{i=1}^{N} W_i x_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} < \mathbf{v}_i, \mathbf{v}_j > x_i x_j)$$

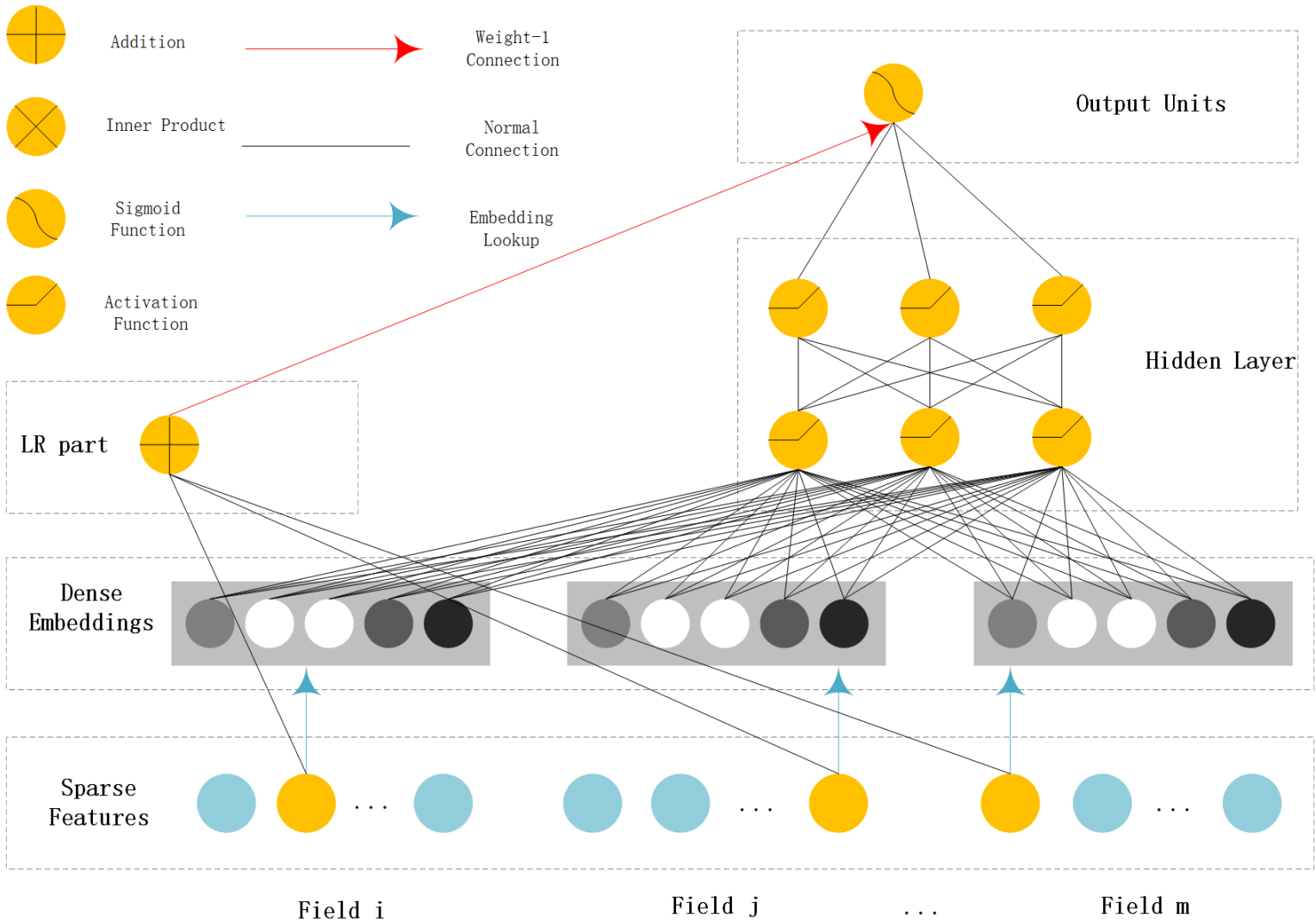
i-th field

$$z_i = \mathbf{W}_0^i \cdot \mathbf{x}[\textit{start}_i : \textit{end}_i] = (w_i, v_i^1, v_i^2, \ldots, v_i^K)$$

$O(\textit{end}_i - \textit{start}_i + 1) \rightarrow O(K+1)$

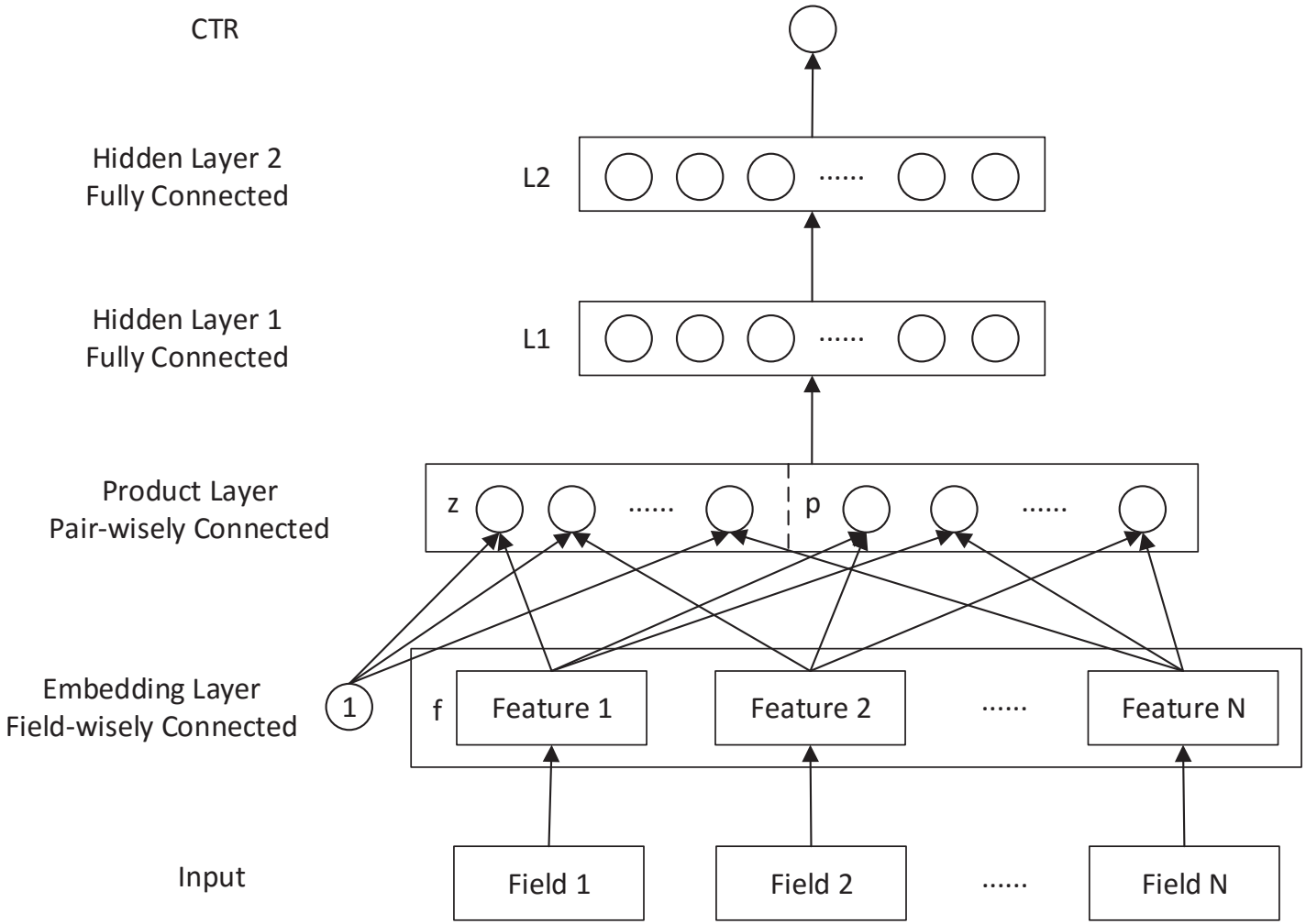
FNN

DeepCTR implement



PNN

Product-based Neural Network[4]



[4] Yanru Qu et al. "Product-based neural networks for user response prediction". In: *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE. 2016, pp. 1149–1154.

Wide & Deep Learning[5]

Wide

- Memorization（记忆性）
- single-layer sparse features (e.g. id)
- 记住历史数据的现有关联

Deep

- Generalization（泛化性）
- Embedding sparse $\rightarrow$ dense $\rightarrow$ multi-layers DNN
- 挖掘数据的潜在关联

Binary features $\rightarrow$ cross-product transform

$\phi(\mathbf{x}) = \prod_{i=1}^{d} x_i^{c_{ki}} \quad c_{ki} \in \{0, 1\}$

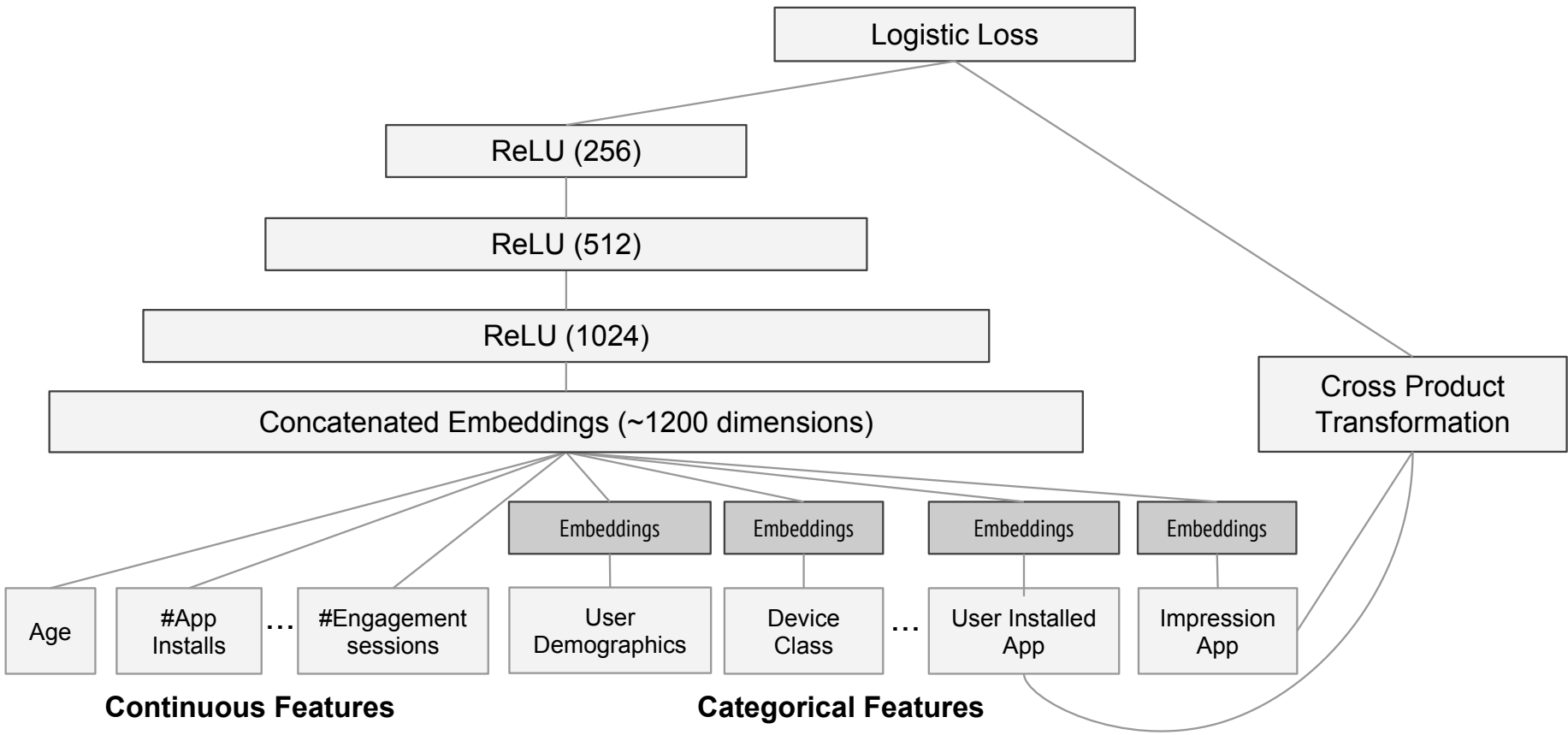
LR

Jointly Training: optimizes all parameters (wide & deep) **simultaneously** (rather than ensemble)

[5] Heng-Tze Cheng et al. "Wide & deep learning for recommender systems". In: *Proceedings of the 1st workshop on deep learning for recommender systems*. ACM. 2016, pp. 7–10.

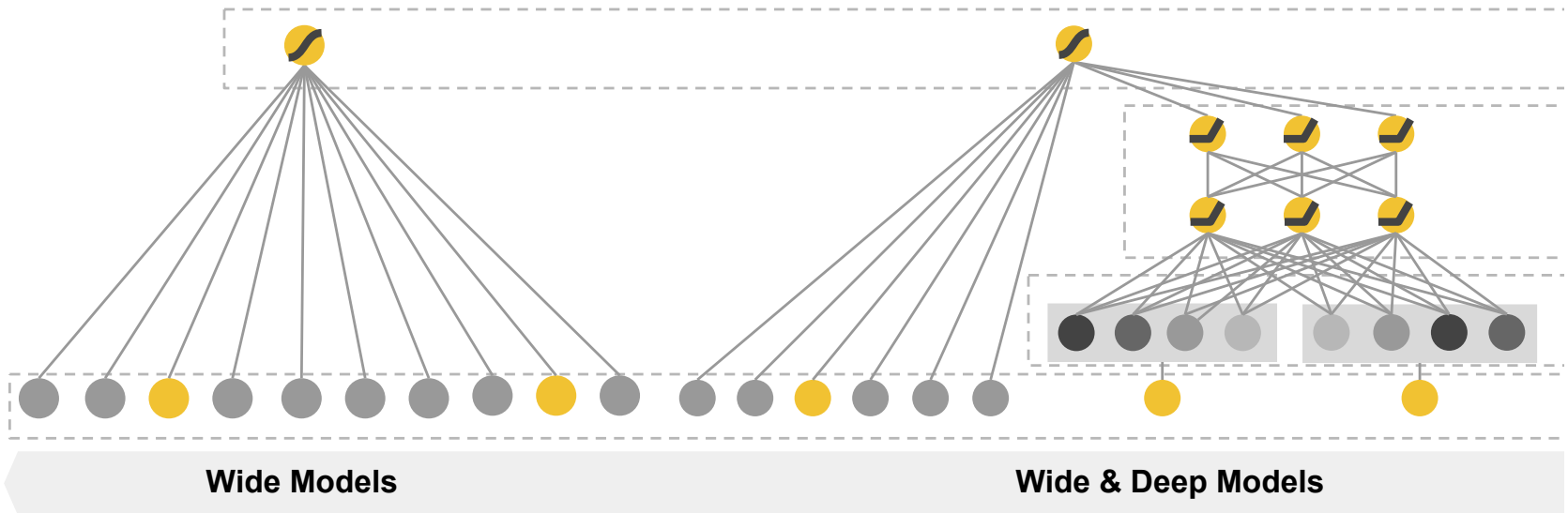
Wide & Deep Learning

Framework for Google Store



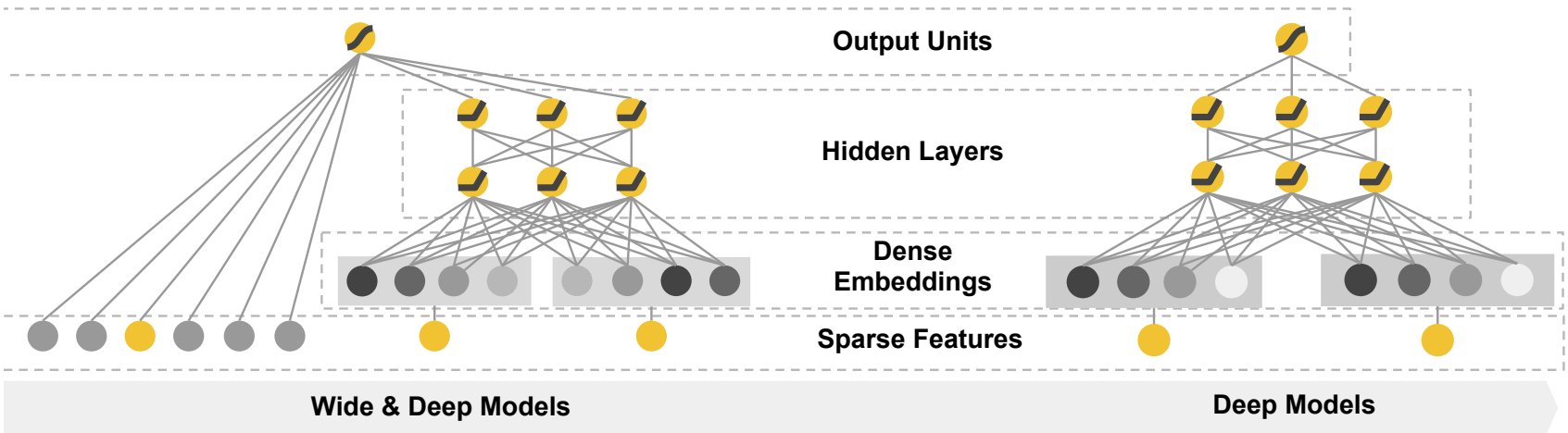
Wide Part

Generalized Framework



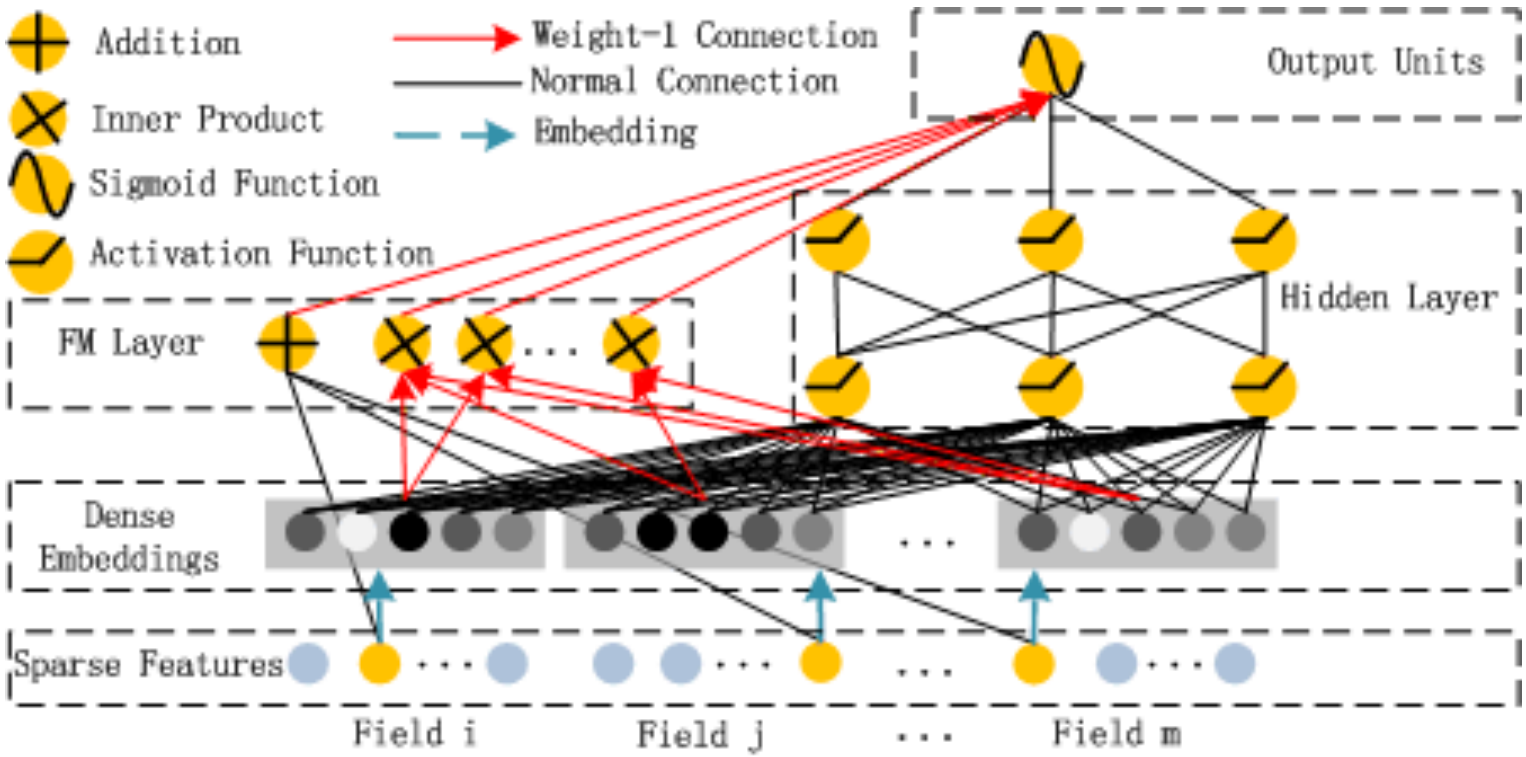
Deep Part

Generalized Framework



DeepFM

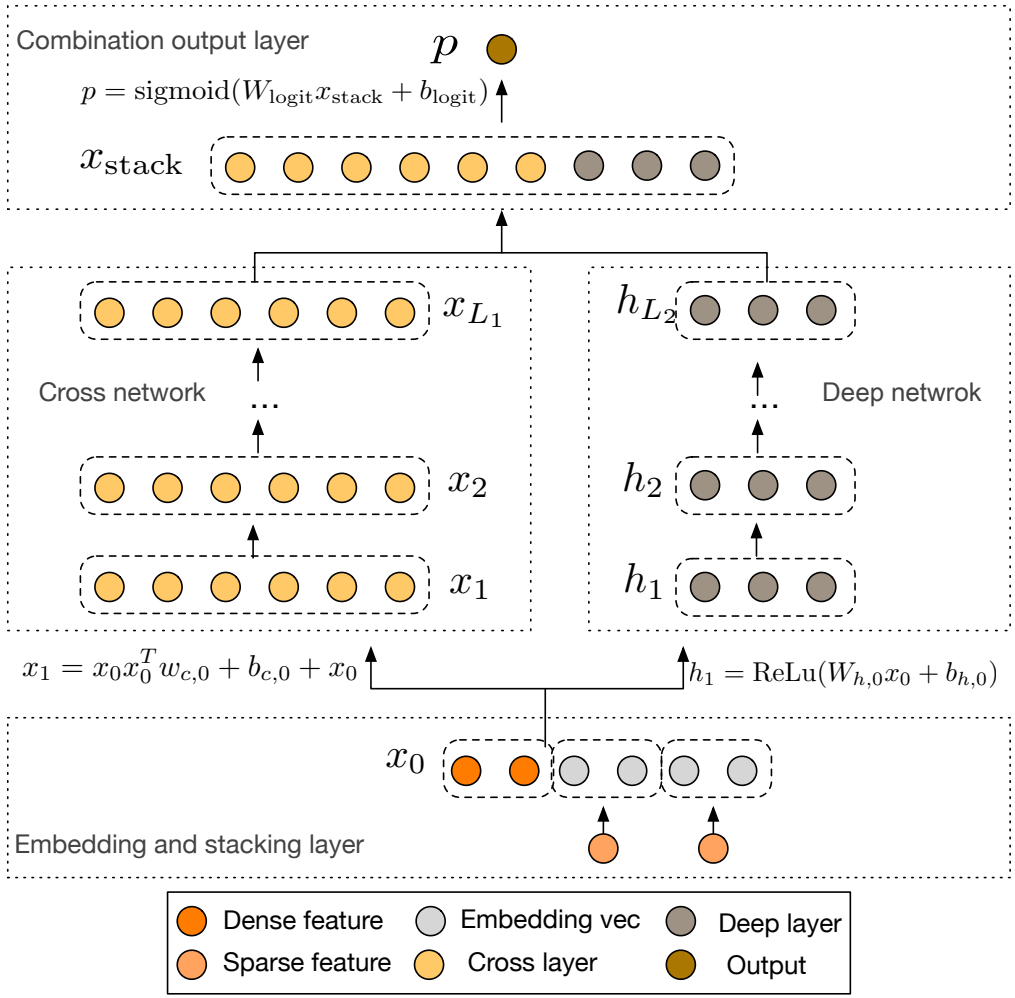
Factorization-Machine based Neural Network[6]



[6] Huifeng Guo et al. "DeepFM: a factorization-machine based neural network for CTR prediction". In: *arXiv preprint arXiv:1703.04247* (2017).

DCN

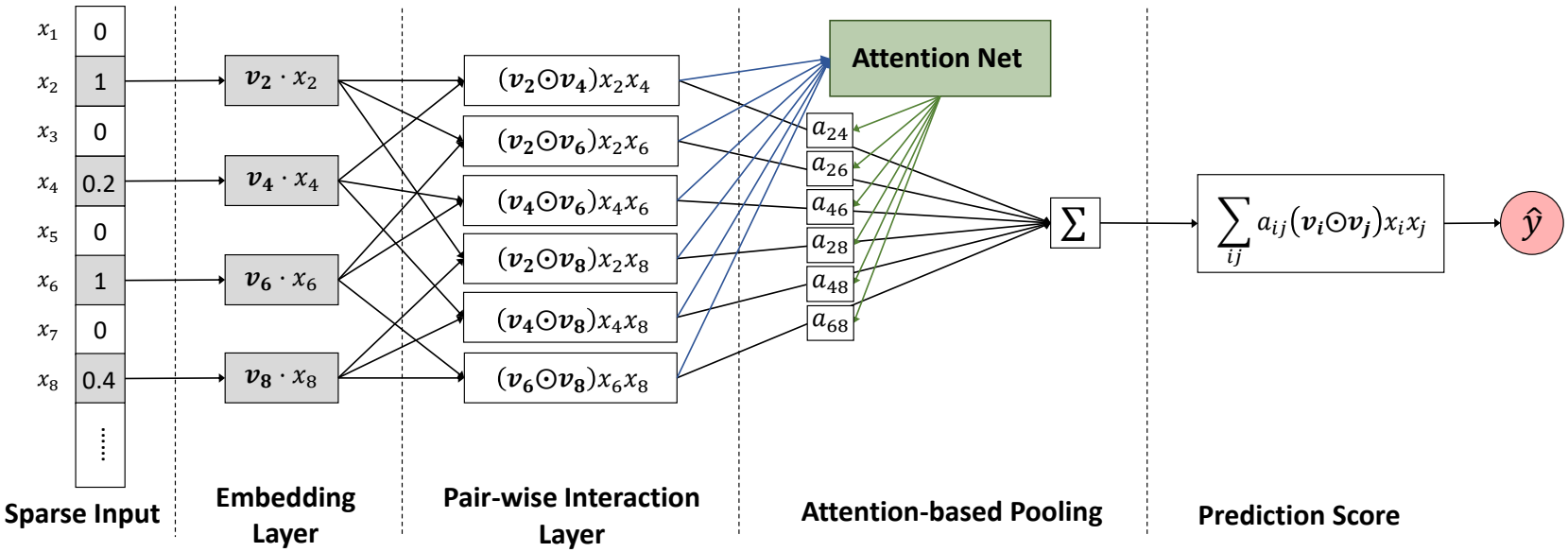
Deep&Cross Network[7]



[7] Ruoxi Wang et al. "Deep & cross network for ad click predictions". In: *Proceedings of the ADKDD'17*. ACM. 2017, p. 12.

AFM

Attentional Factorization Machines[8]



[8] Jun Xiao et al. "Attentional factorization machines: Learning the weight of feature interactions via attention networks". In: *arXiv preprint arXiv:1708.04617* (2017).

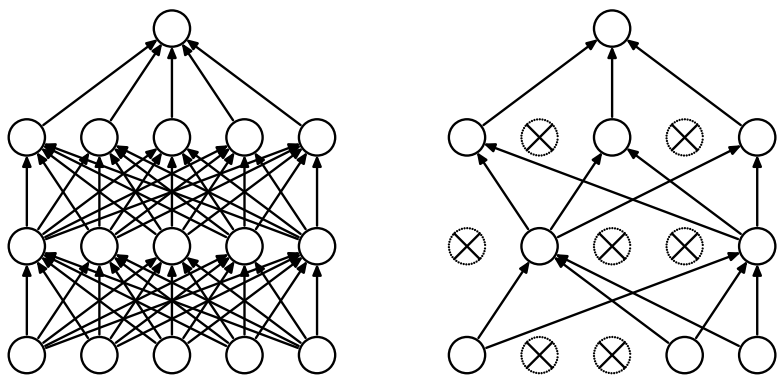
Implements

Demo / Libraries

1. Keras functional API demo:
   `https://github.com/sararob/keras-wine-model`
2. TensorFlow built-in
3. **DeepCTR**: `https://github.com/shenweichen/DeepCTR`

Practice Tips:

1. Text description features (e.g. address)$\rightarrow$ deep part input
2. Dense dropout
3. Increase hidden units/layers above embedding layers
4. ......



Thanks