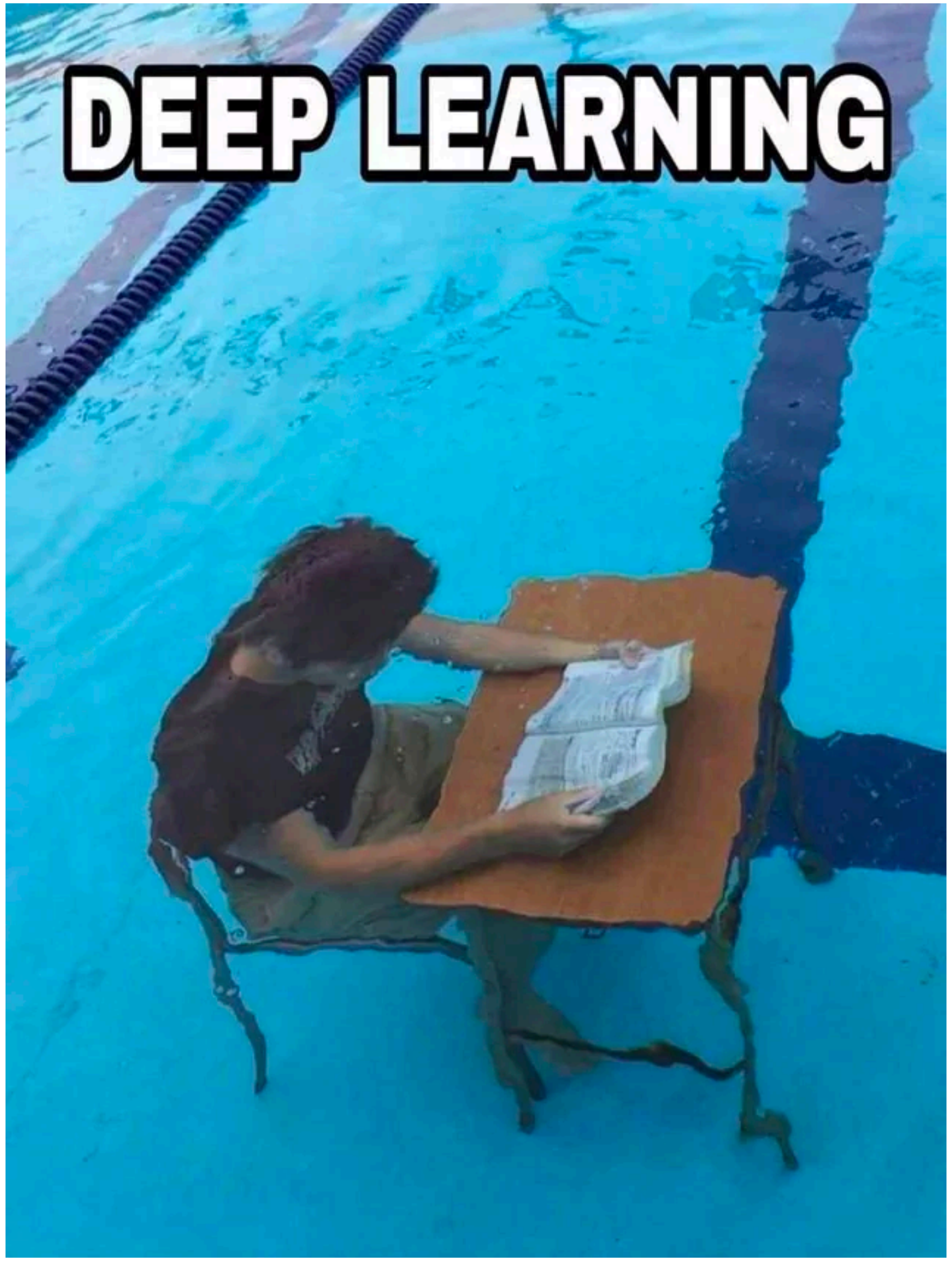
SINGAPORE INSTITUTE OF TECHNOLOGY

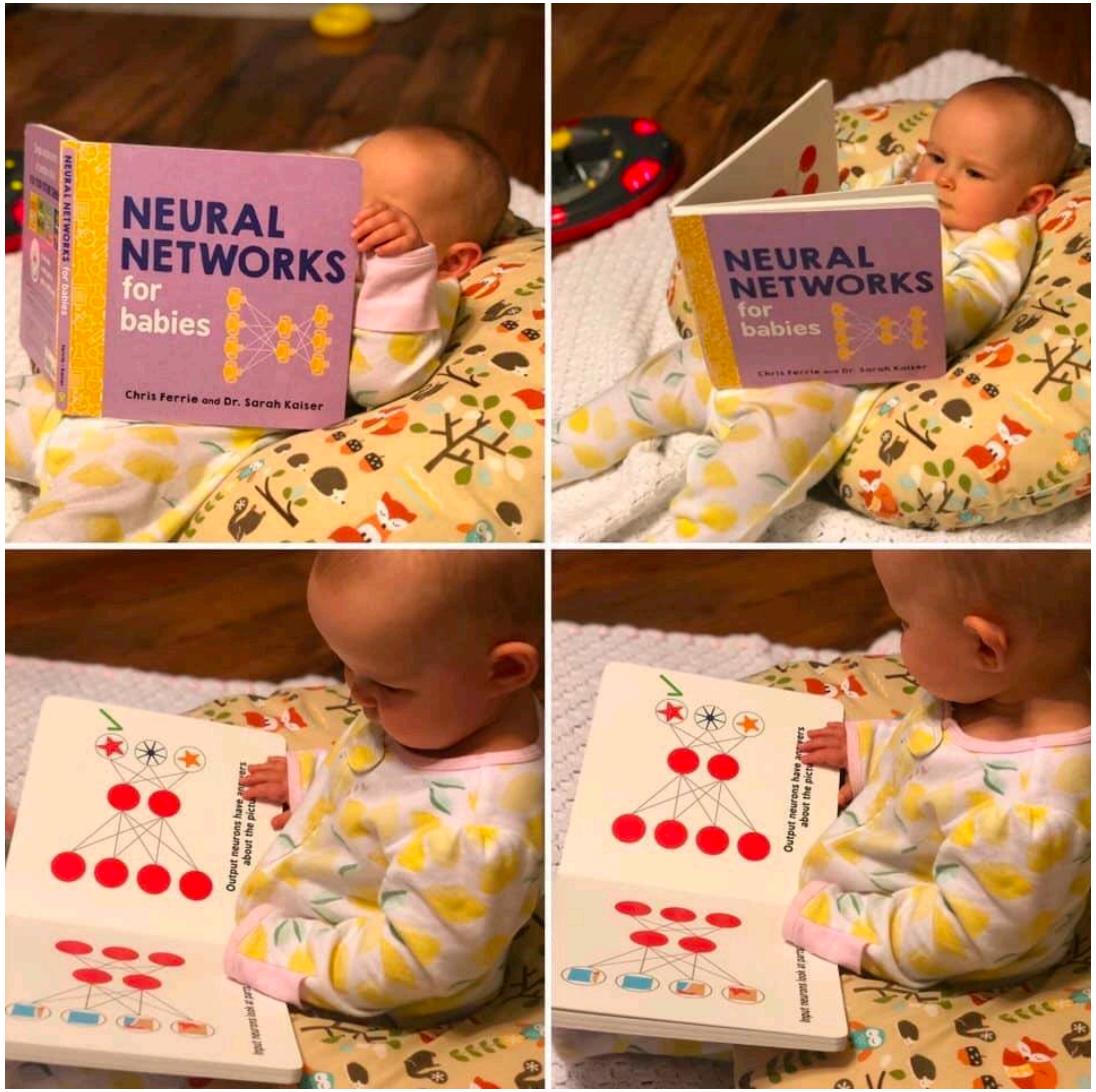
Modern Recommender Systems in Theory and Practice

Lucas Vinh Tran

30 March 2023

The New Era


DEEP LEARNING

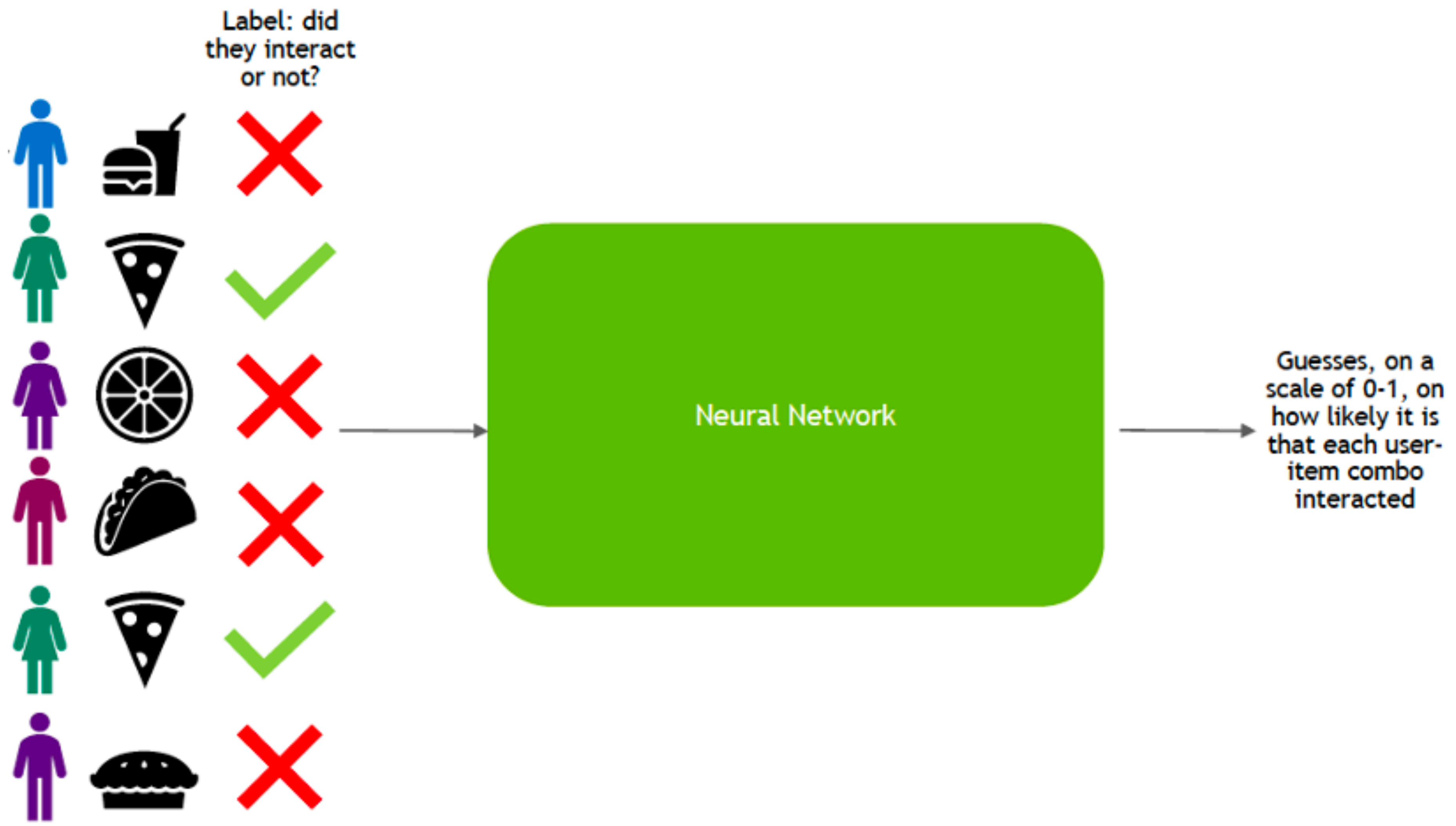

NEURAL NETWORKS
for
babies
Chris Ferrie and Dr. Sarah Kaiser

More Motivations

- More heterogeneous data, more complicated user behaviours, more diverse scenarios, etc.
- For example, short video recommendation
  - What is short video application?
    - User-generated short video clips, usually < 1min
  - Rapid growth of short video applications
    - Number of short video users is 648 millions in China by the end of 2018
    - Number of daily active users is growing at the speed of > 800% each year since 2016

Why Deep Learning?

- Nonlinear Transformation
- Representation Learning
  - Reduce the efforts of hand-craft feature design
  - Heterogeneous content information such as text, images, etc.
- …



Source: Deep learning based recommender system: A survey and new perspectives

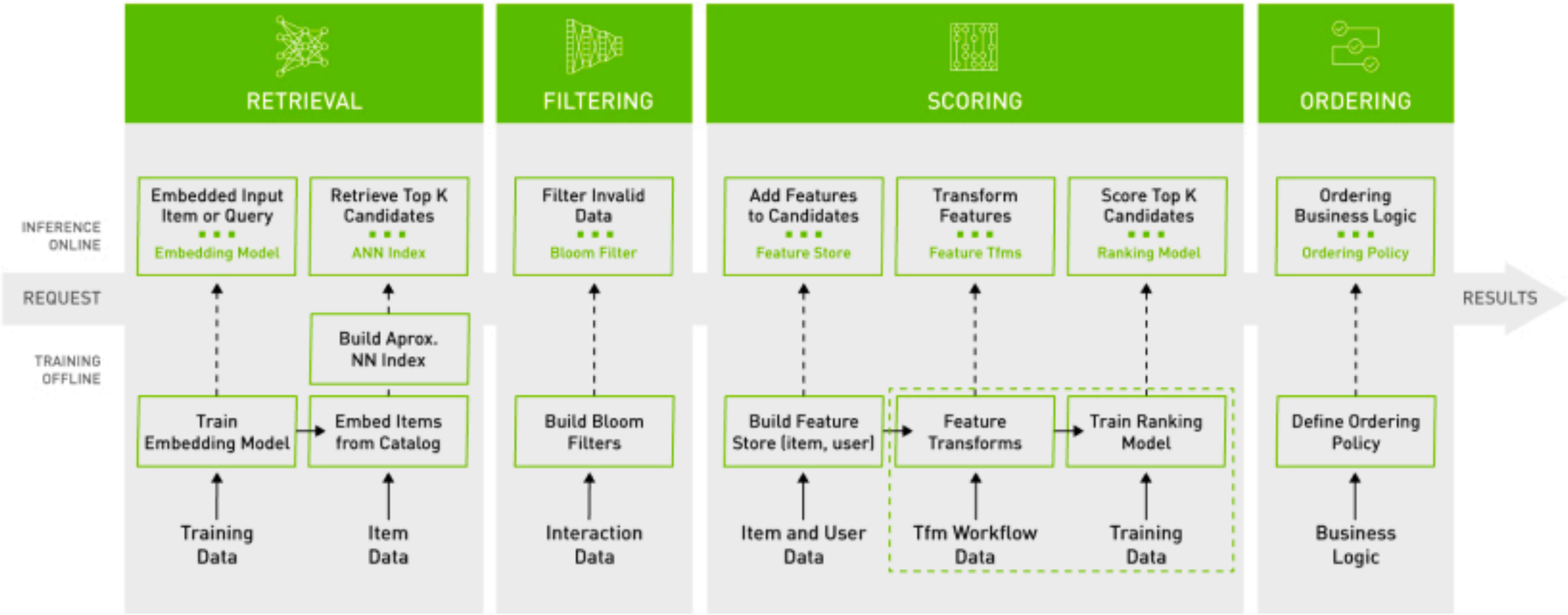
DL empowered RS

tensorflow/
recommenders

TensorFlow Recommenders is a library for building recommender system models using TensorFlow.

32 Contributors | 195 Used by | 2k Stars | 234 Forks

TorchRec



microsoft/
recommenders

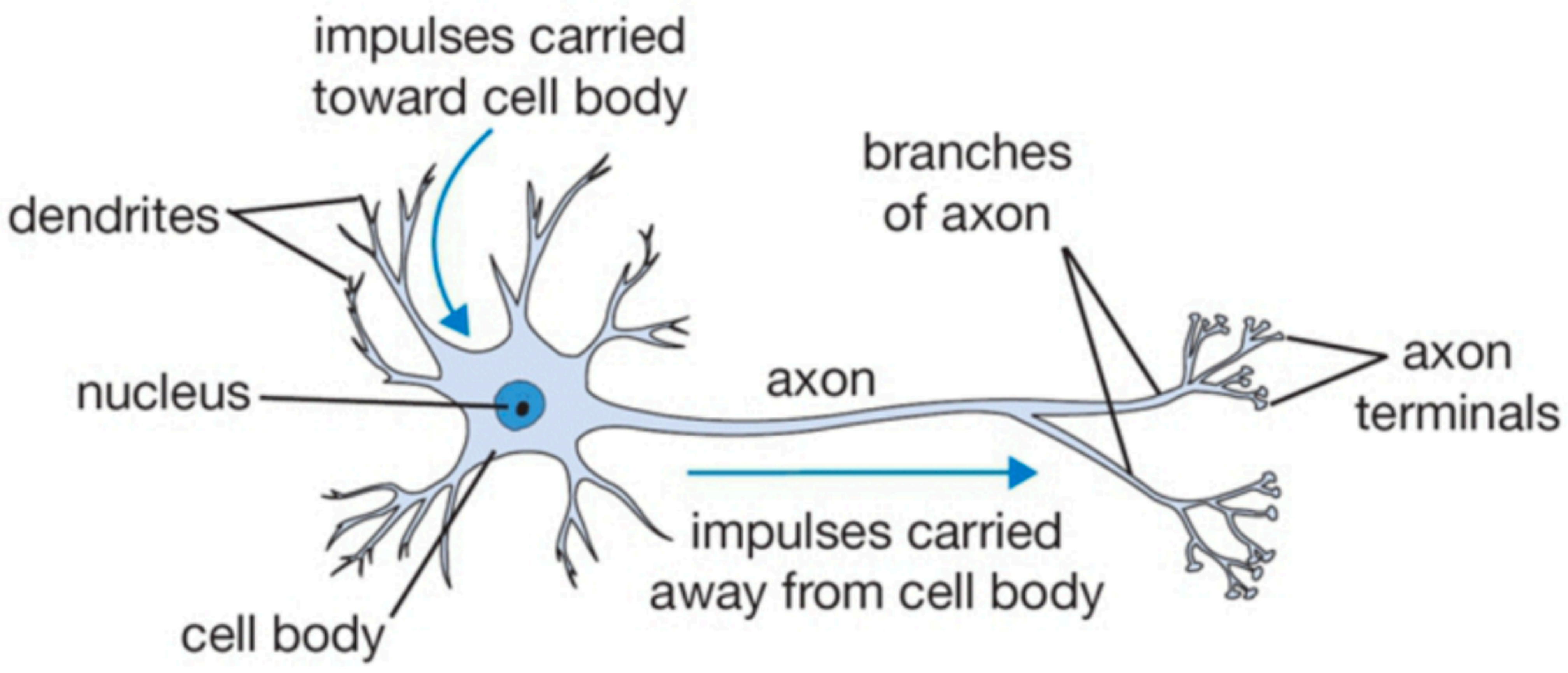
Best Practices on Recommendation Systems

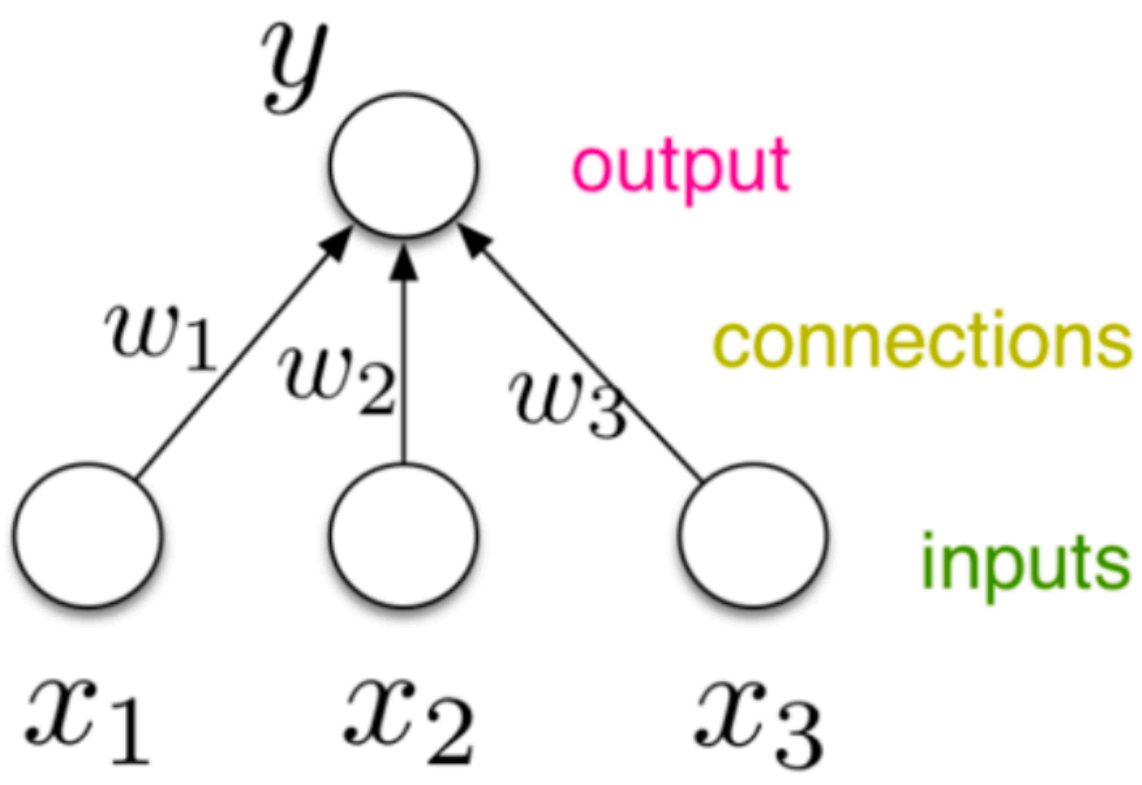
92 Contributors | 59 Used by | 5 Discussions | 15k Stars | 3k Forks

MLP based

- Neural extension of Traditional RS (e.g., MF)
  - Neural Collaborative Filtering (NCF)
- Feature Representation Learning with MLP
  - Wide and Deep (WideDeep)
- And many more…

MLP based



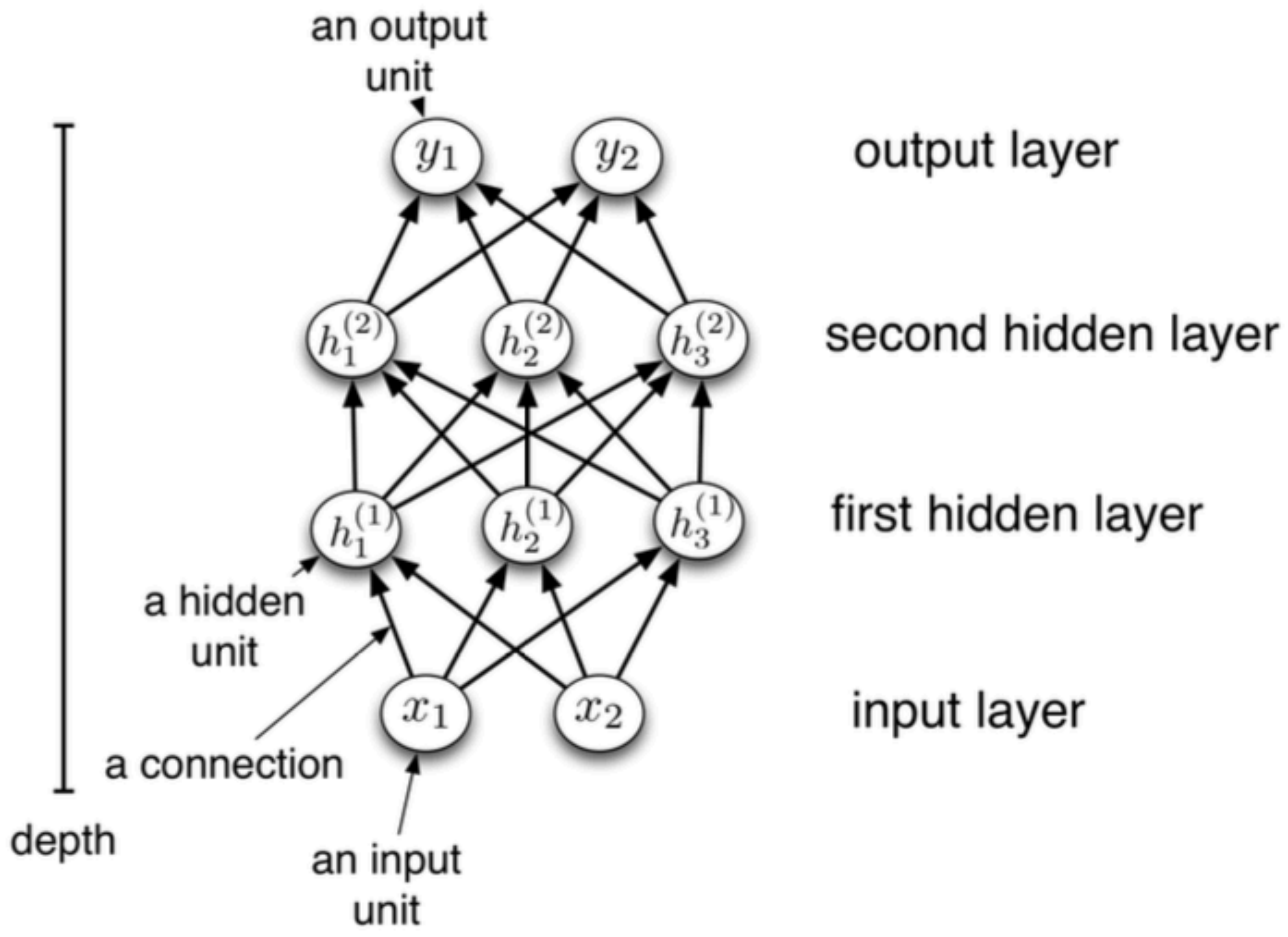


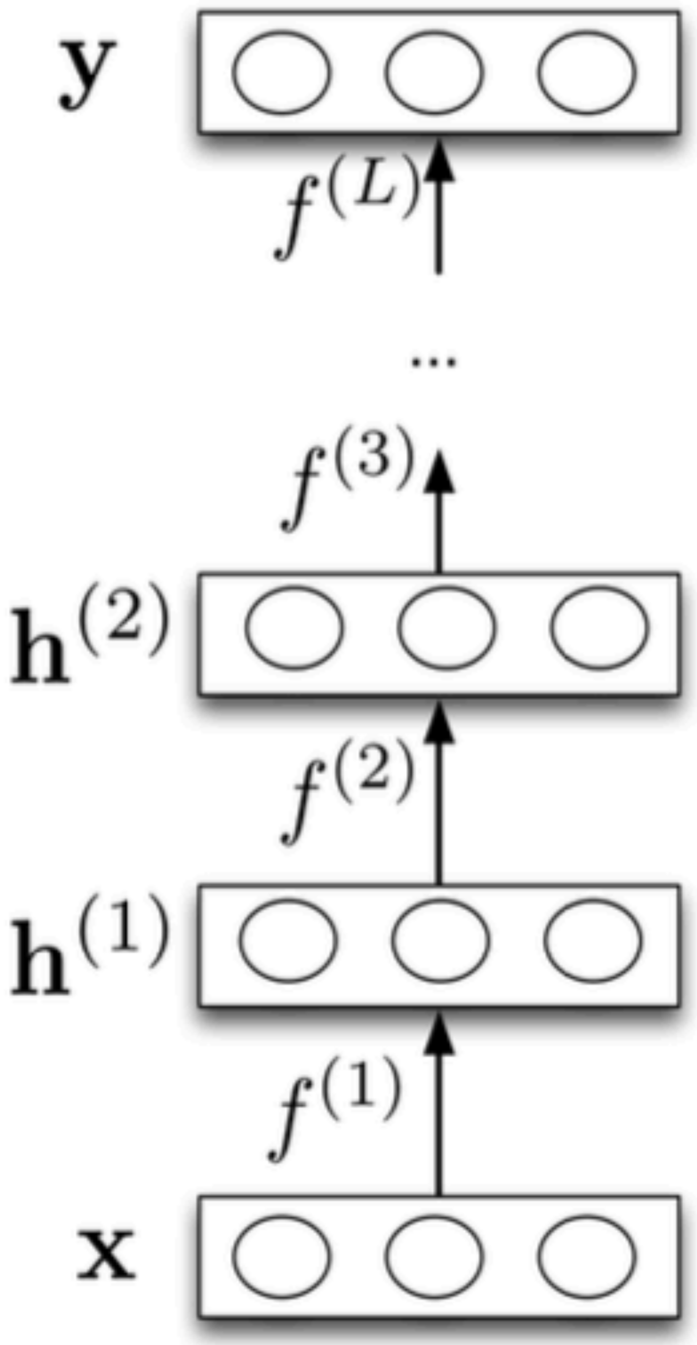
output weights bias

$$y = \phi\left(\mathbf{w}^\top \mathbf{x} + b\right)$$

activation function inputs

MLP based





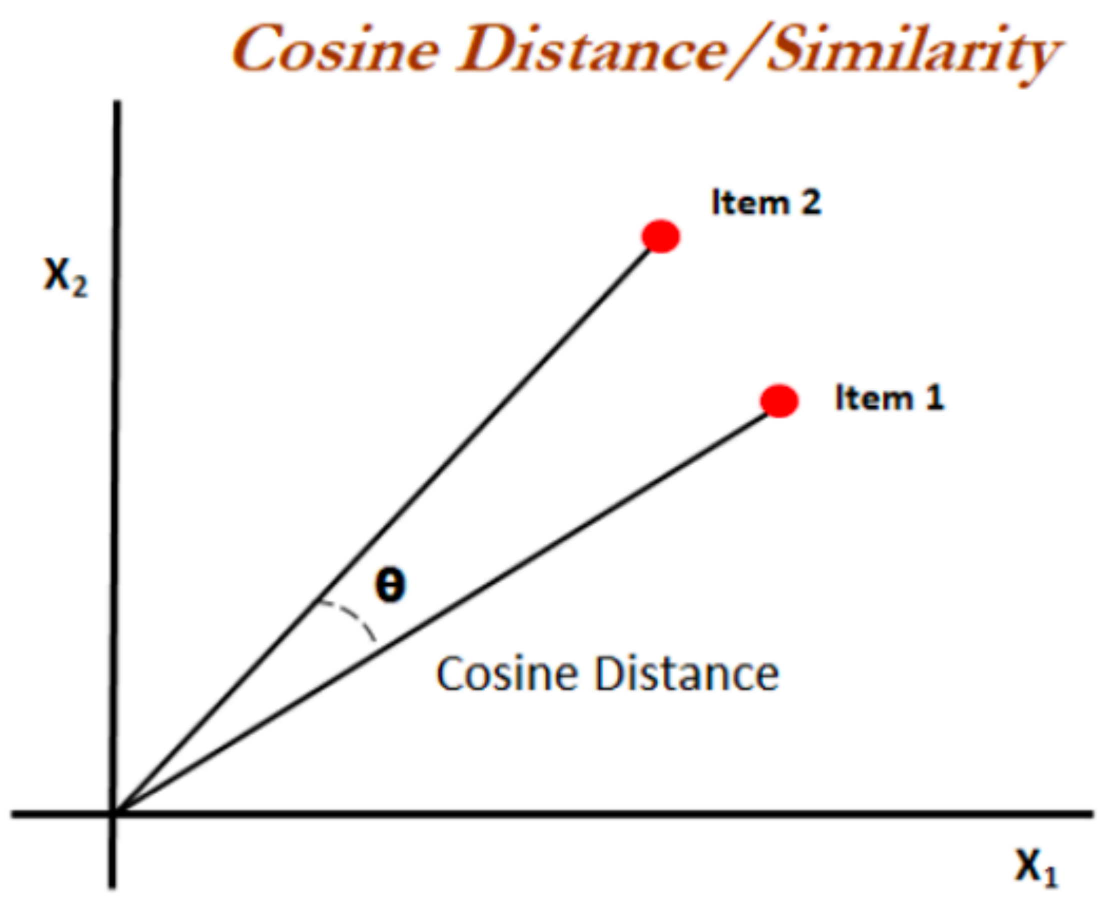
$$
\begin{aligned}
h^{(1)} &= f^{(1)}\Big(\sum_j w_{ij}^{(1)} x_j + b_i^{(1)}\Big) \\
h^{(2)} &= f^{(2)}\Big(\sum_j w_{ij}^{(2)} h_j^{(1)} + b_i^{(2)}\Big) \\
y_i &= f^{(3)}\Big(\sum_j w_{ij}^{(3)} h_j^{(2)} + b_i^{(3)}\Big)
\end{aligned}
$$

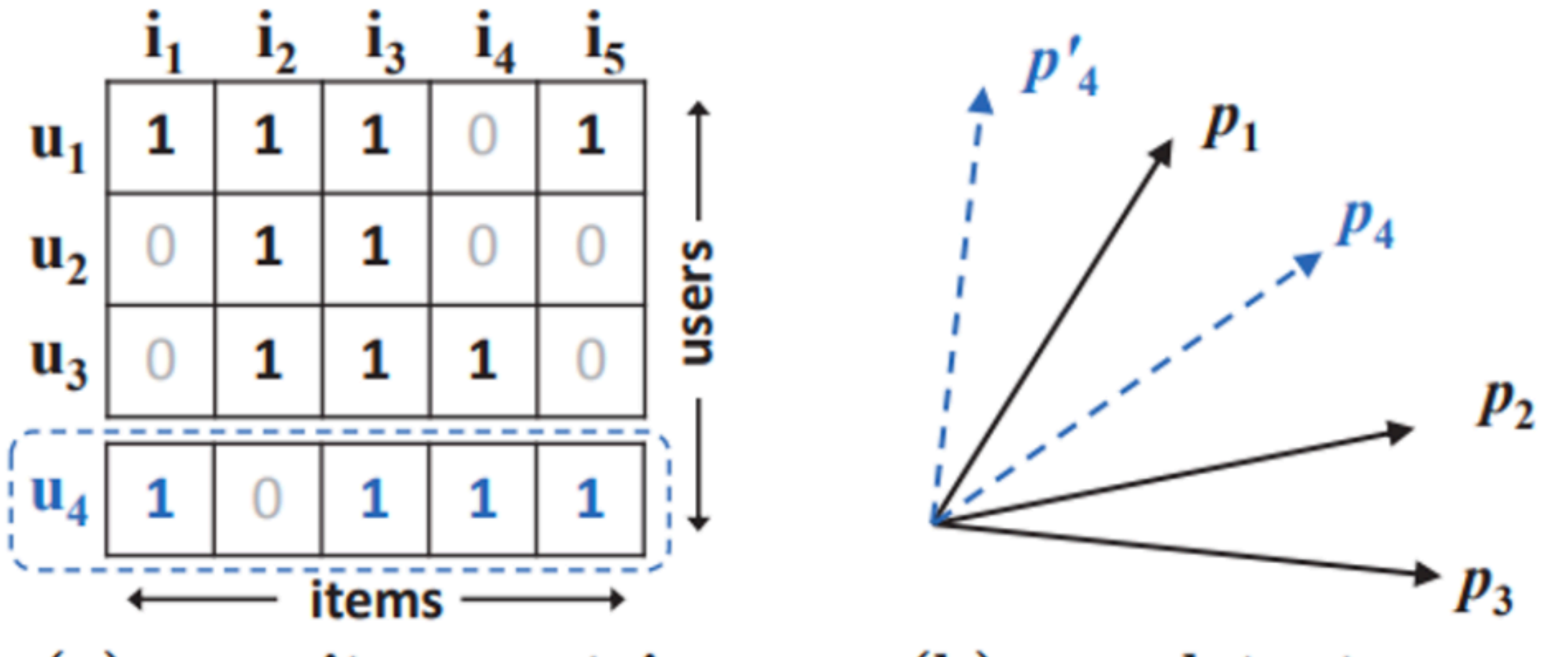
$$\Downarrow$$

$$
\begin{aligned}
\mathrm{h}^{(1)} &= f^{(1)}(\mathrm{W}^{(1)}(x) + \mathrm{b}^{(1)}) \\
\mathrm{h}^{(2)} &= f^{(2)}(\mathrm{W}^{(2)}\mathrm{h}^{(1)} + \mathrm{b}^{(2)}) \\
\mathrm{y} &= f^{(3)}(\mathrm{W}^{(3)}\mathrm{h}^{(2)} + \mathrm{b}^{(3)})
\end{aligned}
$$

MLP based

Neural Collaborative Filtering (NCF)





(a) user–item matrix (b) user latent space

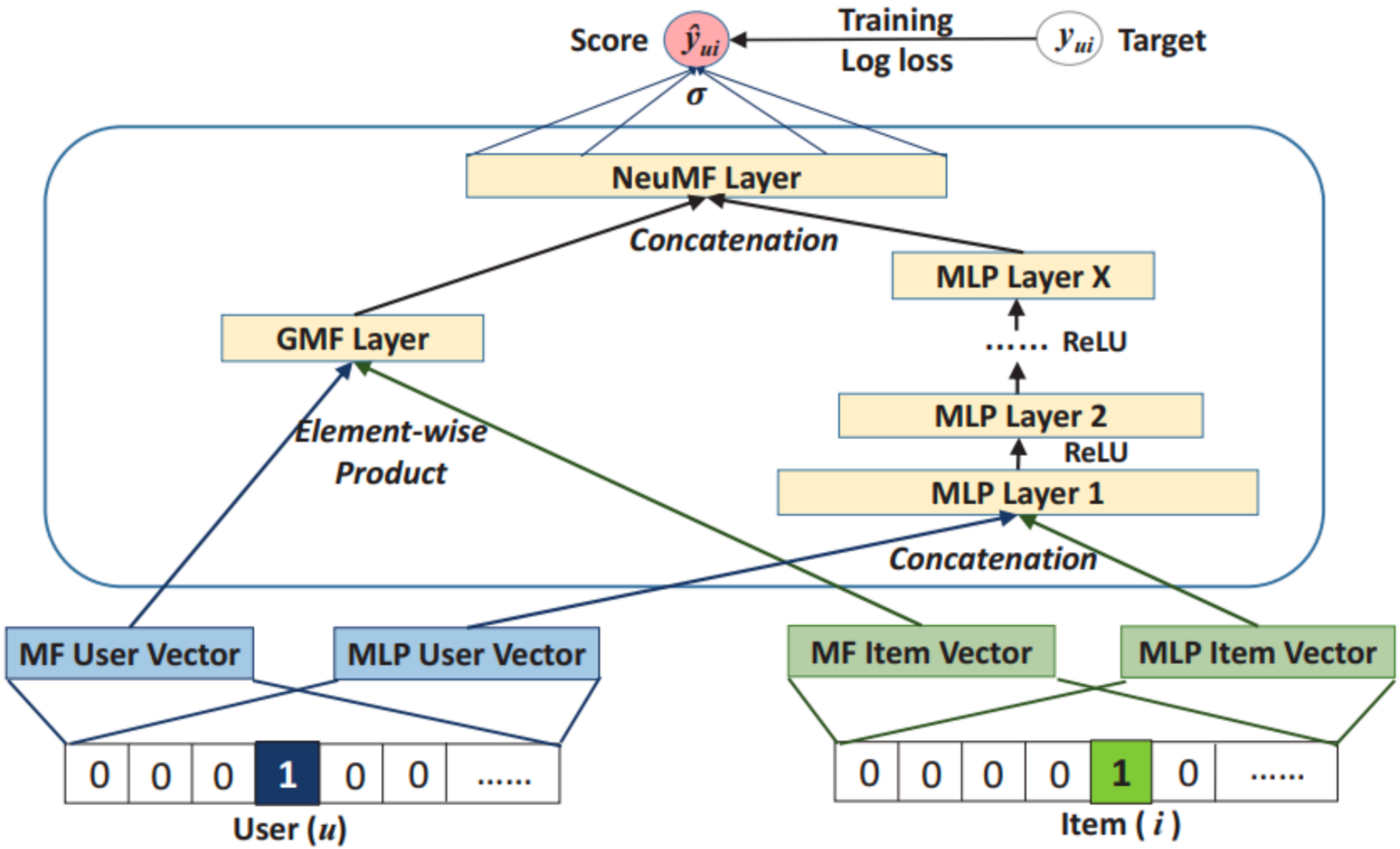
p1 = [1,1,1,0,1]
p2 = [0,1,1,0,0]
p3 = [0,1,1,1,0]
p4 = [1,0,1,1,1]

Sim(23) = 2/3 = 0.67
Sim(12) = 2/4 = 0.5
Sim(13) = 2/5 = 0.4

Sim(41) = 3/5 = 0.6
Sim(42) = 1/5 = 0.2
Sim(43) = 2/5 = 0.4

MLP based

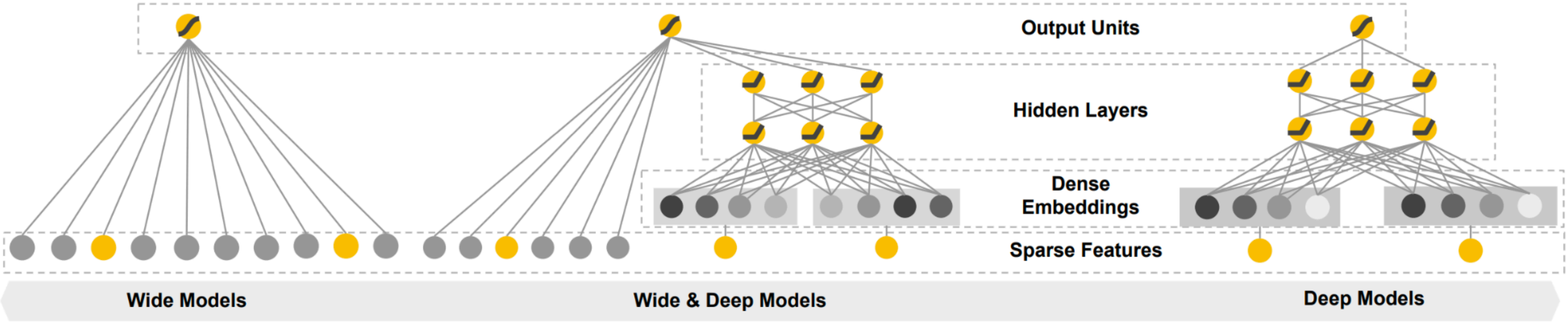
Neural Collaborative Filtering (NCF)



MLP based

Wide and Deep

- Wide component: a cross-product transformation is used to capture the interactions between binary features (e.g., AND(gender=female, language=en) is 1 if “gender is female” and “language=en”, 0 otherwise)
- Deep component: each of these sparse, high-dimensional categorical features are first converted into a low-dimensional and dense real-valued vector; then fed into MLP



Autoencoder based