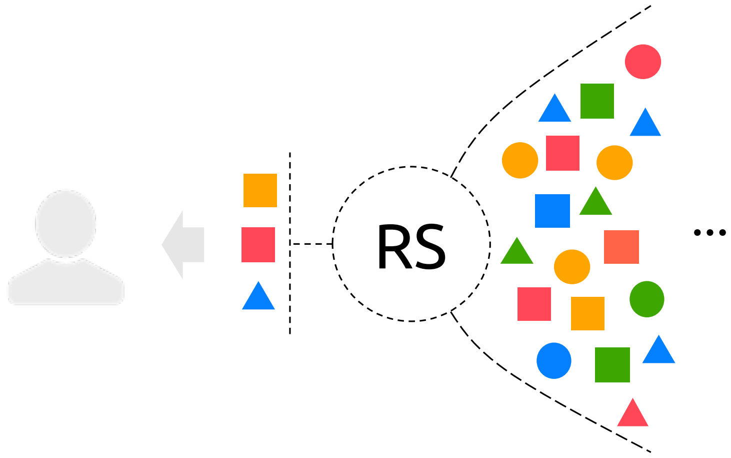
Modularizing Deep Neural Network-Inspired Recommendation Algorithms

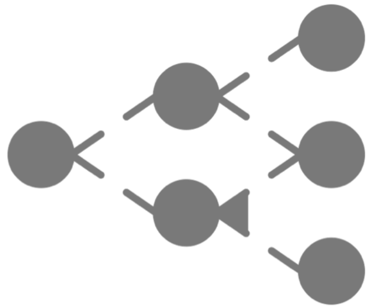
Longqi Yang Eugene Bagdasaryan Hongyi Wen

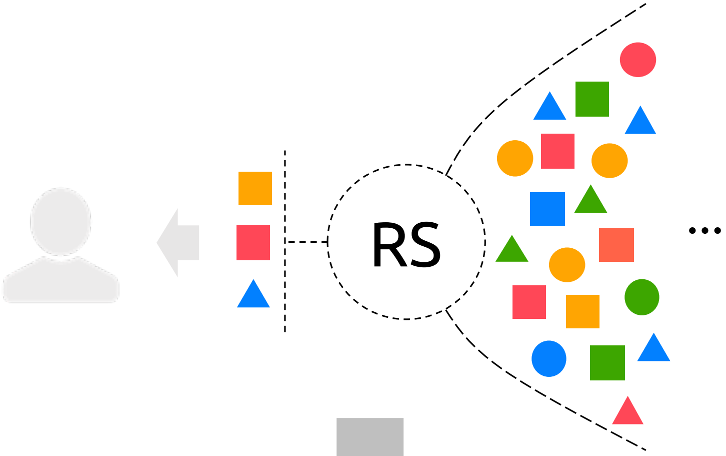
CORNELL TECH

Cornell CIS Computer Science

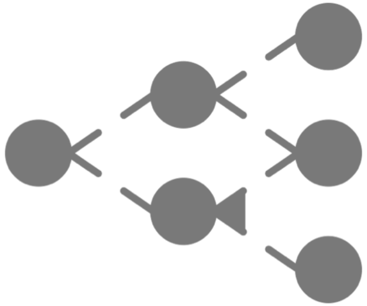
Funders: NSF Oath: A Verizon company


RS
...





OpenRec



Tutorial Agenda

Github link for slides and notebooks: https://github.com/ylongqi/openrec/tree/master/tutorials

Lecture (Longqi, 40 min).

Hands-on sessions (15 min each).

- OpenRec basics + diversity and fairness (Eugene)
- Customizing Deep YouTube Video Recommendation (Hongyi)
- Temporal-aware recommendation (Longqi)

Lecture Agenda

- How we do recommendations, traditionally.
- How deep learning can help to do better recommendations.
- Modularization and OpenRec.

Recommendation Algorithm

items (ranked) | user + context

Recommendation Algorithm
(Policy π)

$$\pi(y|x;\theta)$$

items (ranked) | user + context ; model parameters

Recommendation Algorithm (Policy π)

$$\pi(y|x;\theta)$$

items (ranked) | user + context ; model parameters

items consumed
demographics
tweet
…

location
time
mood
…

How we do recommendations traditionally (with Collaborative Filtering)

$$\pi(y|x;\theta)$$

items (ranked) | user + context ; model parameters

items consumed
demographics
tweets
…

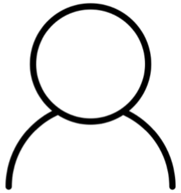
location
time
mood
…

A running example: articles recommendation

y: 

Latent Factor-based models
(Model parameters)

θ:



i ($i \in \{1, \dots, N\}$)

[0.1, -0.2, 0.35, … , 0.15]

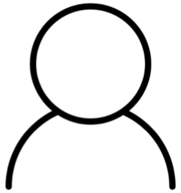
user latent factor $\boldsymbol{v}$

j ($j \in \{1, \dots, M\}$)

[-0.05, 0.5, 0.1, … , -0.3]

article latent factor $\boldsymbol{v}$

Latent Factor-based models (Model parameters)



$i \ (i \in \{1, \ldots, N\})$

$j \ (j \in \{1, \ldots, M\})$

θ:

[0.1, -0.2, 0.35, … , 0.15]

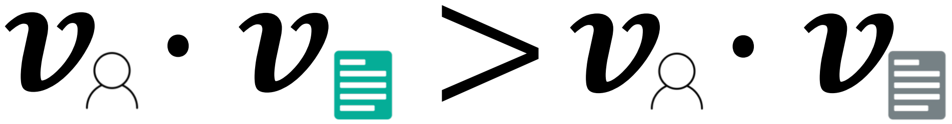
user latent factor $\boldsymbol{v}$

[-0.05, 0.5, 0.1, … , -0.3]

article latent factor $\boldsymbol{v}$

$\boldsymbol{v} \cdot \boldsymbol{v}$ → A user's preference towards an article

Latent Factor-based models (Objective)



Latent Factor-based models (Objective)

$$\boldsymbol{v} \cdot \boldsymbol{v} > \boldsymbol{v} \cdot \boldsymbol{v}$$

Learning from user consumption history

Bayesian Personalized Ranking (**BPR**)
(Pairwise learning)

$$\max \sum \sum \log(\sigma(\boldsymbol{v} \cdot \boldsymbol{v} - \boldsymbol{v} \cdot \boldsymbol{v}))$$

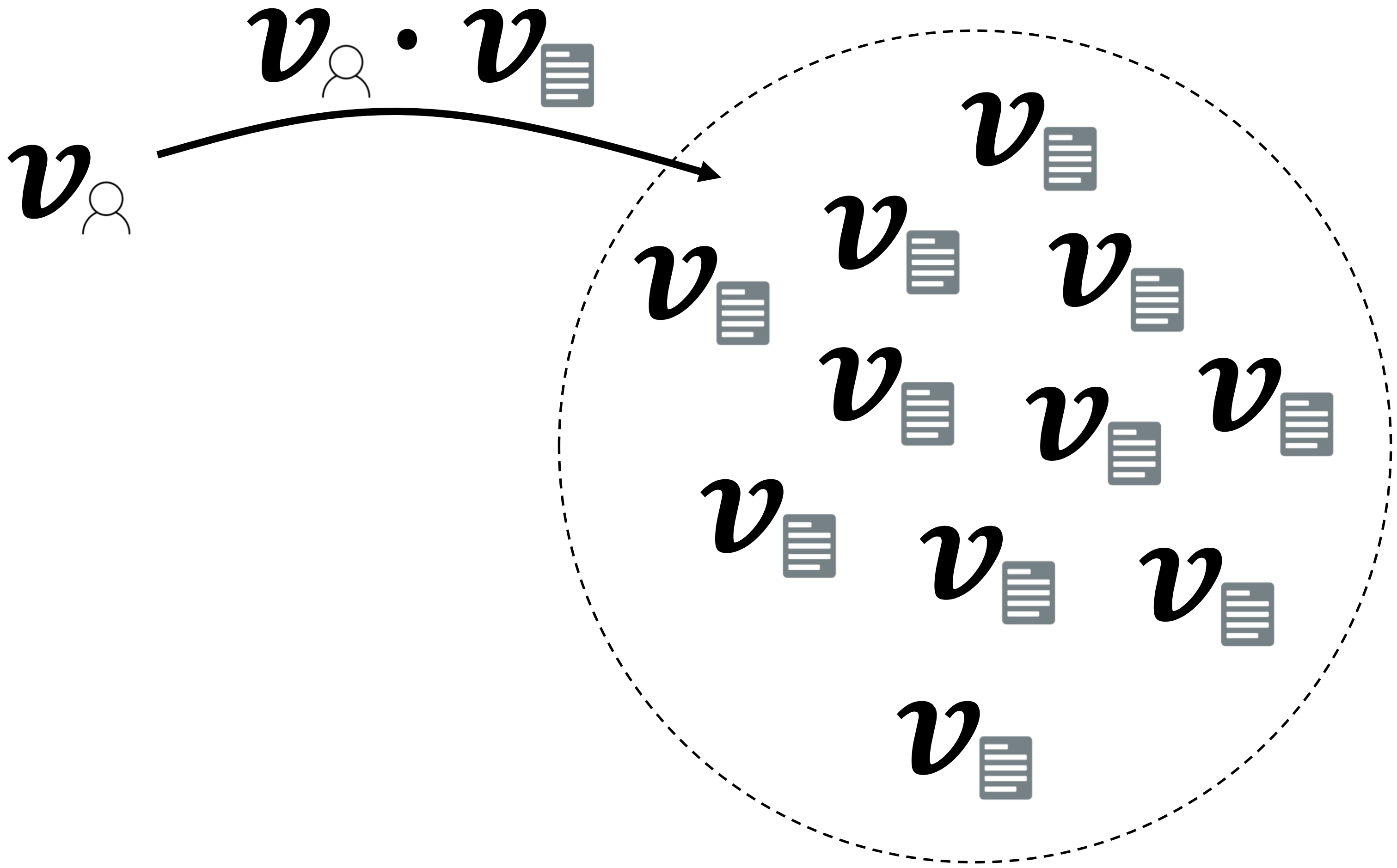
Rendle, Steffen, et al. "BPR: Bayesian personalized ranking from implicit feedback." *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*. AUAI Press, 2009.

Probabilistic Matrix Factorization (**PMF**)
(Pointwise learning)

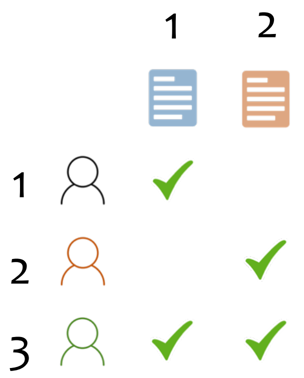
$$\min \sum_{\text{user},\, \text{doc}_{\text{green}},\, \text{doc}_{\text{grey}}} \left(1 - \boldsymbol{v}_{\text{user}} \cdot \boldsymbol{v}_{\text{doc}_{\text{green}}}\right)^2 + c\left(\boldsymbol{v}_{\text{user}} \cdot \boldsymbol{v}_{\text{doc}_{\text{grey}}}\right)^2$$

Mnih, Andriy, and Ruslan R. Salakhutdinov. "Probabilistic matrix factorization." *Advances in neural information processing systems*. 2008.

Recommend based on BPR or PMF

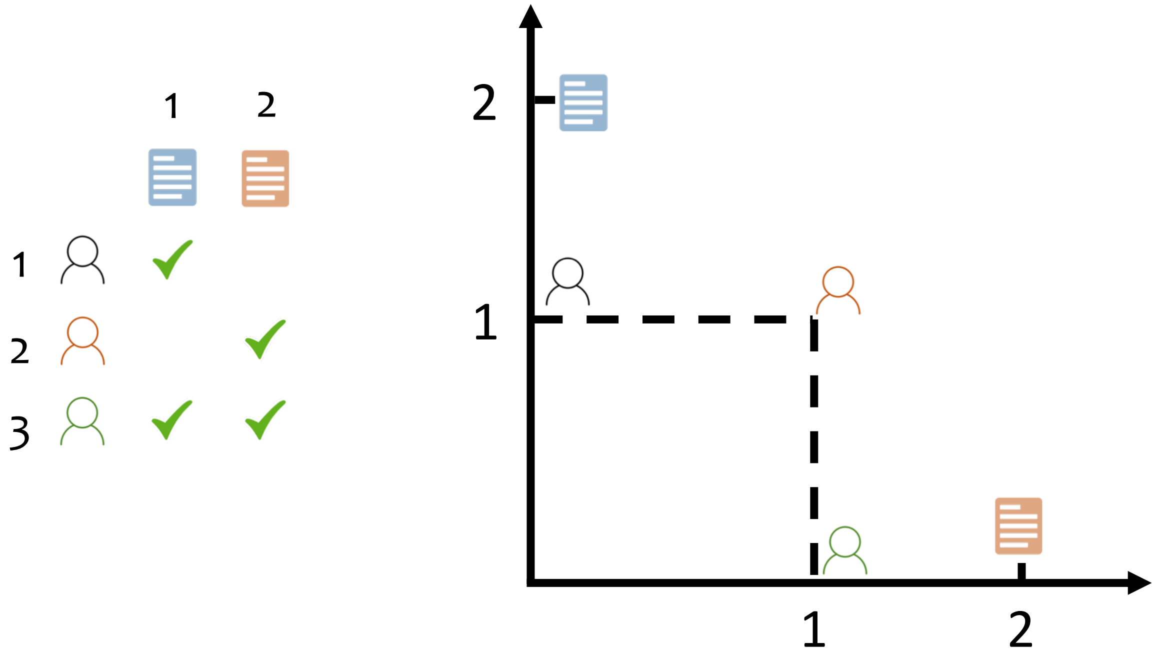


It doesn’t has to be dot-product: dot-product does not generalize



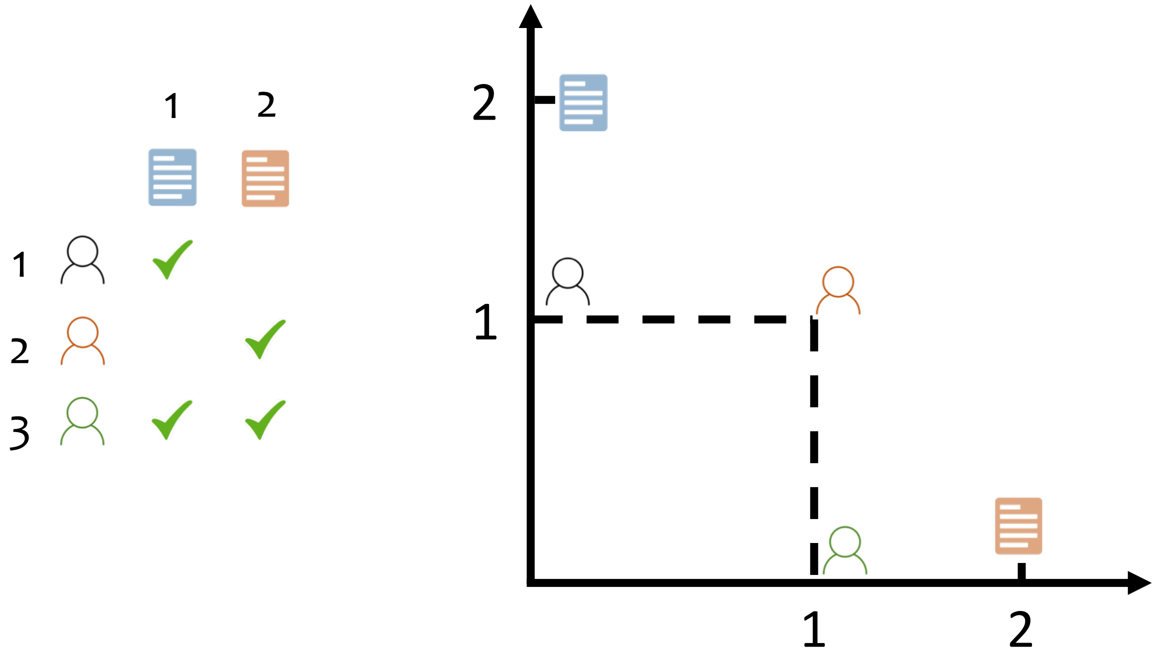
Hsieh, Cheng-Kang, et al. "Collaborative metric learning." *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017.

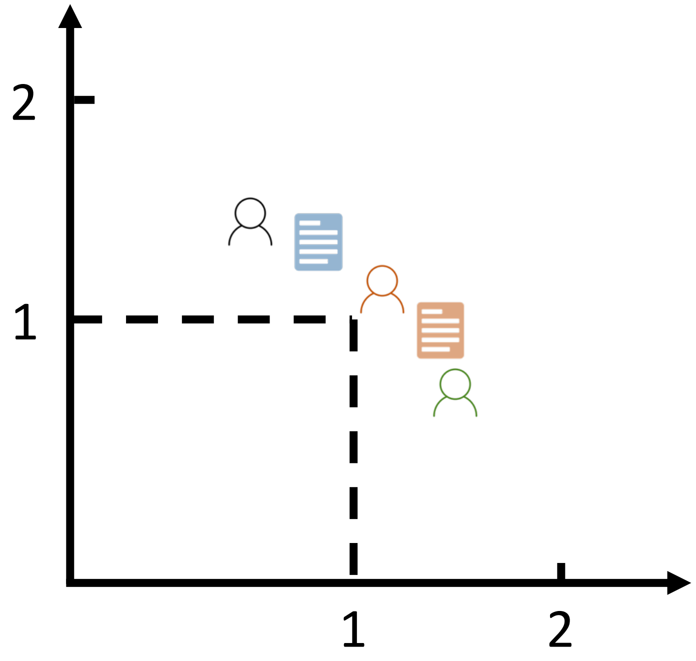
It doesn't has to be dot-product: dot-product does not generalize



Hsieh, Cheng-Kang, et al. "Collaborative metric learning." *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017.

It doesn’t has to be dot-product: dot-product does not generalize





Hsieh, Cheng-Kang, et al. "Collaborative metric learning." *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017.

Collaborative Metric Learning (**CML**)

$$\boldsymbol{v} \cdot \boldsymbol{v} \Rightarrow -||\boldsymbol{v} - \boldsymbol{v}||^2$$

$$-\log(\sigma(*)) \Rightarrow \omega[m - *]_+$$

Hsieh, Cheng-Kang, et al. "Collaborative metric learning." *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017.

Recommend based on CML

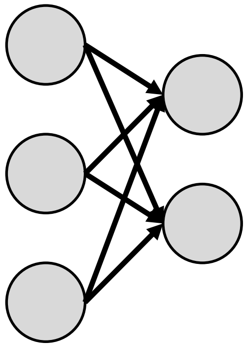
$$-||\boldsymbol{v} - \boldsymbol{v}||^2$$

Lecture Agenda

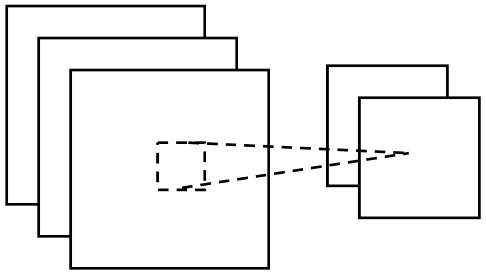
- How we do recommendations, traditionally.
- How deep learning helps to do better recommendations.
- Modularization and OpenRec

Building blocks

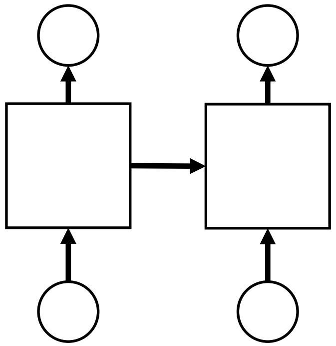
Multi-Layer Perceptron
(MLP)



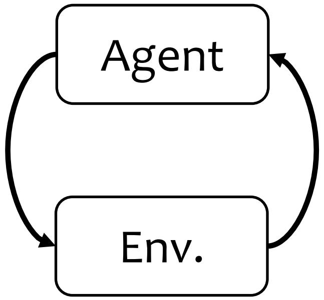
Convolutional Neural Network
(CNN)



Recurrent Neural Network
(RNN)



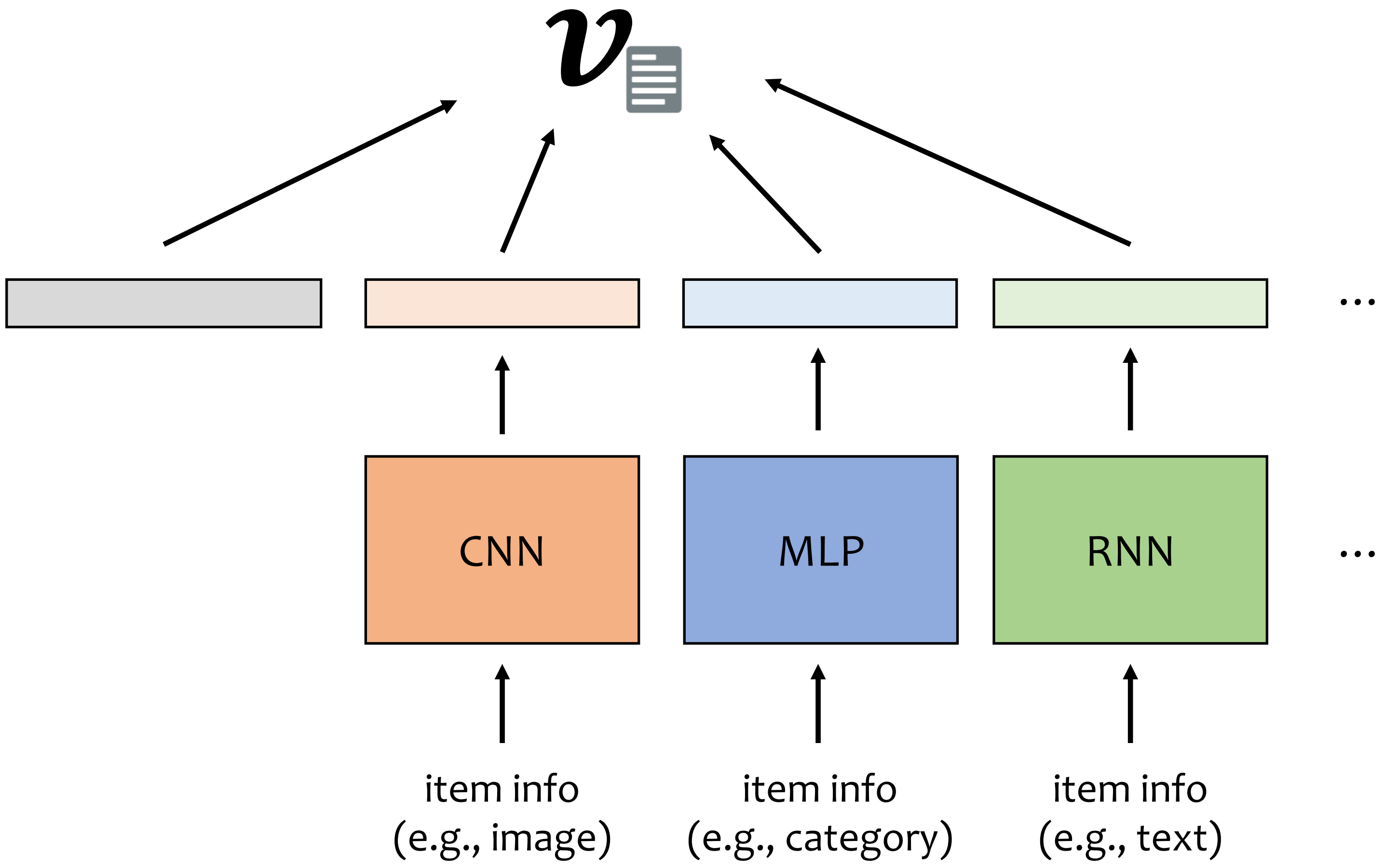
Reinforcement Learning
(RL)



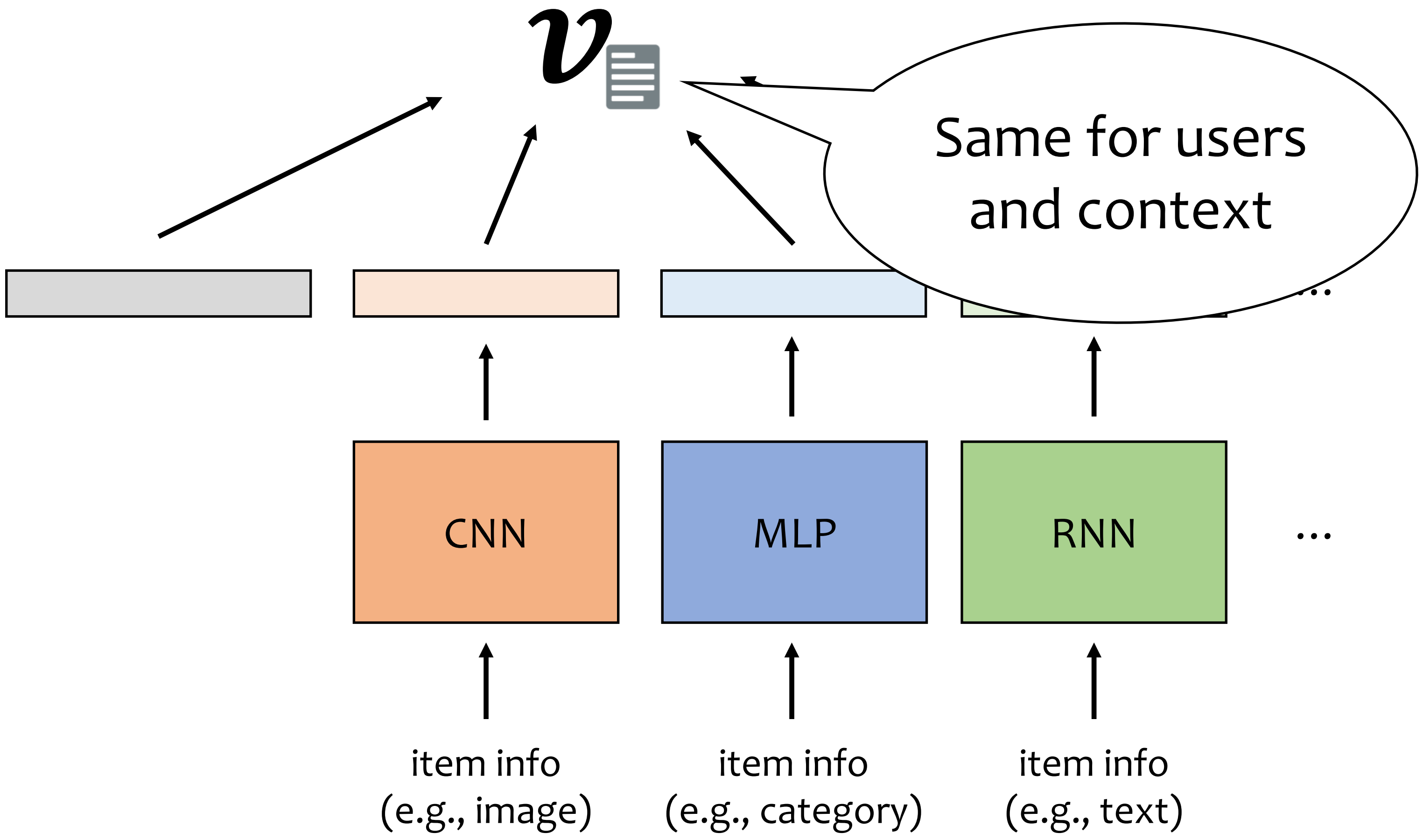
How deep learning helps?

- High-dimensional and multi-channel data streams.
- Nuanced user-item interaction patterns.

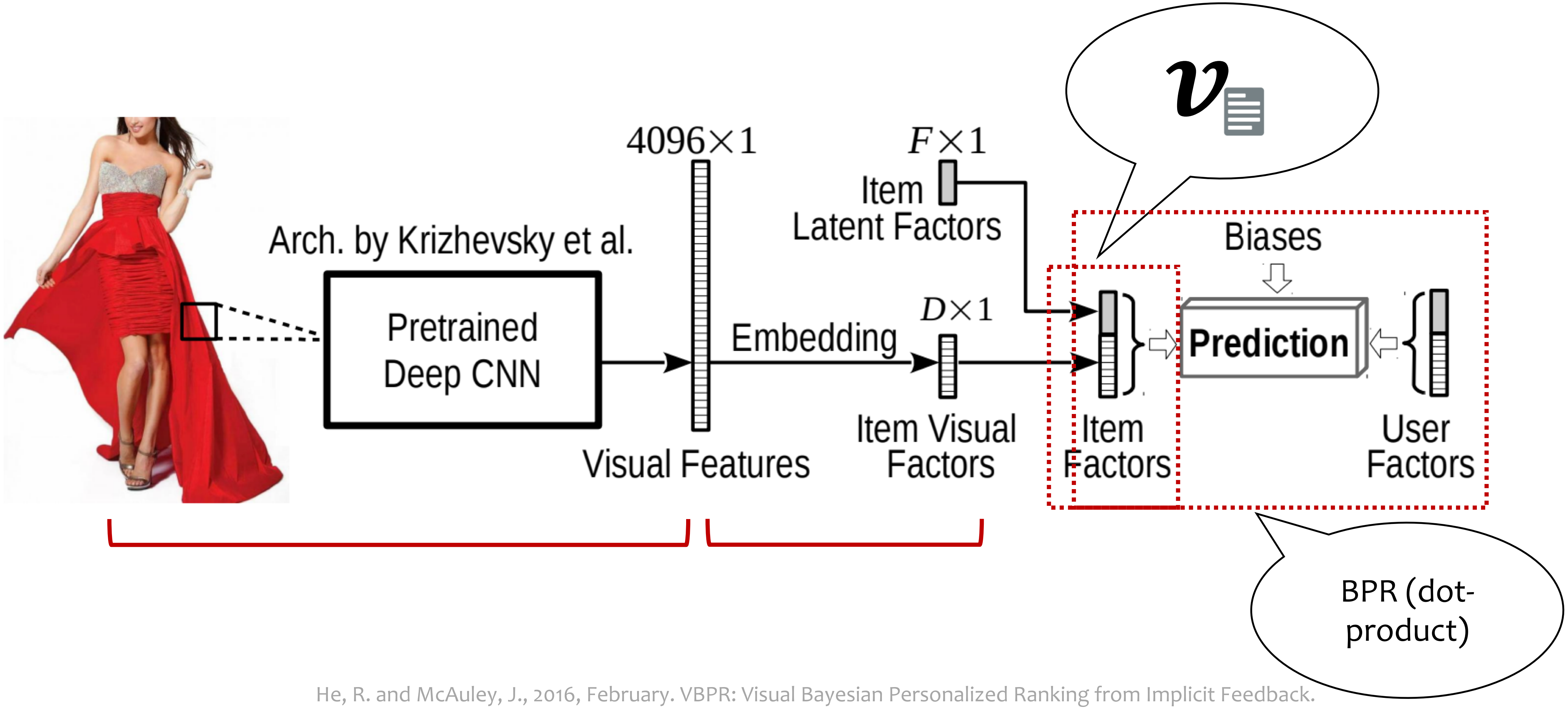
High-dimensional and Multi-channel Data streams



High-dimensional and Multi-channel Data streams

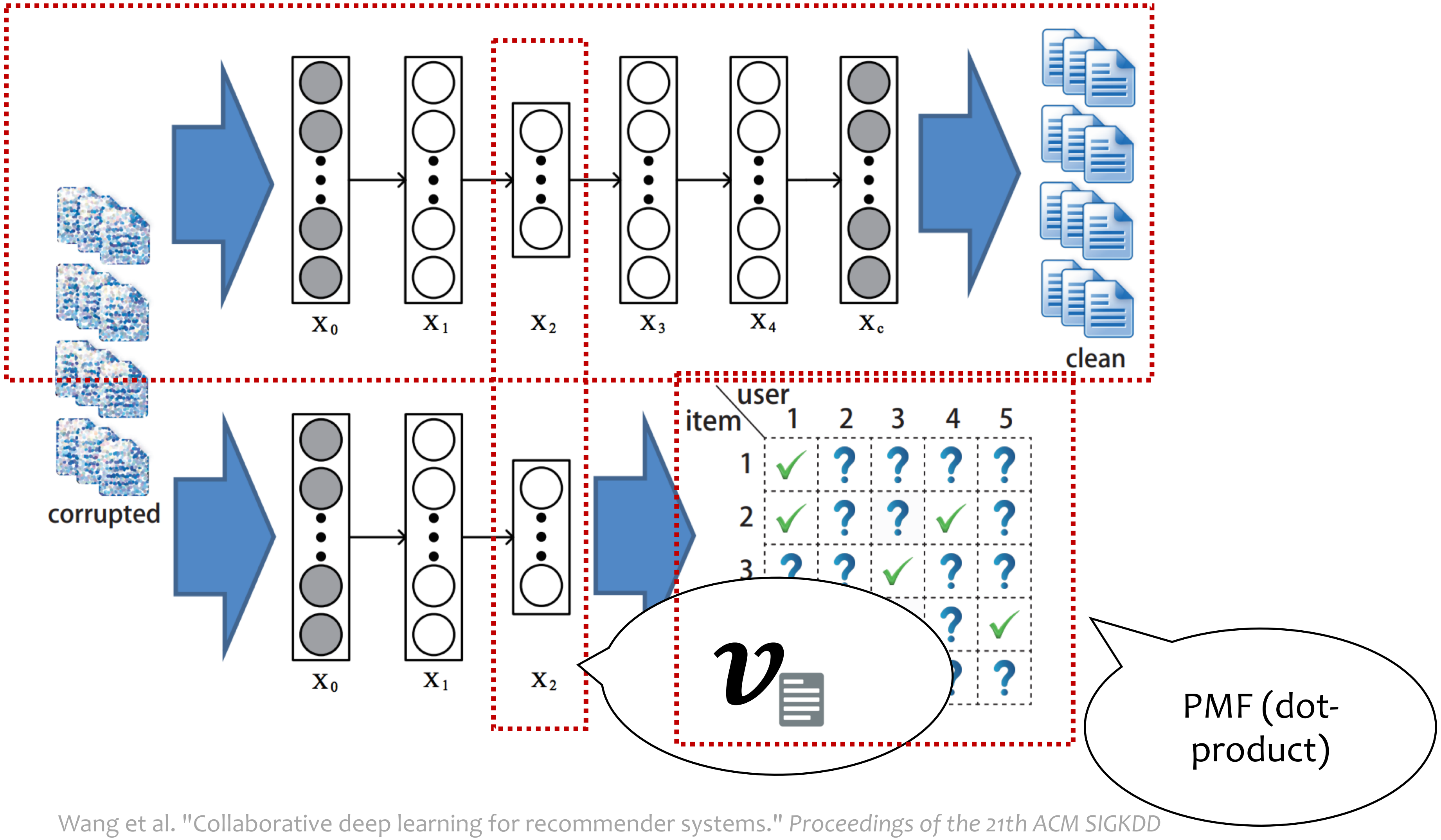


Visual BPR (VBPR)



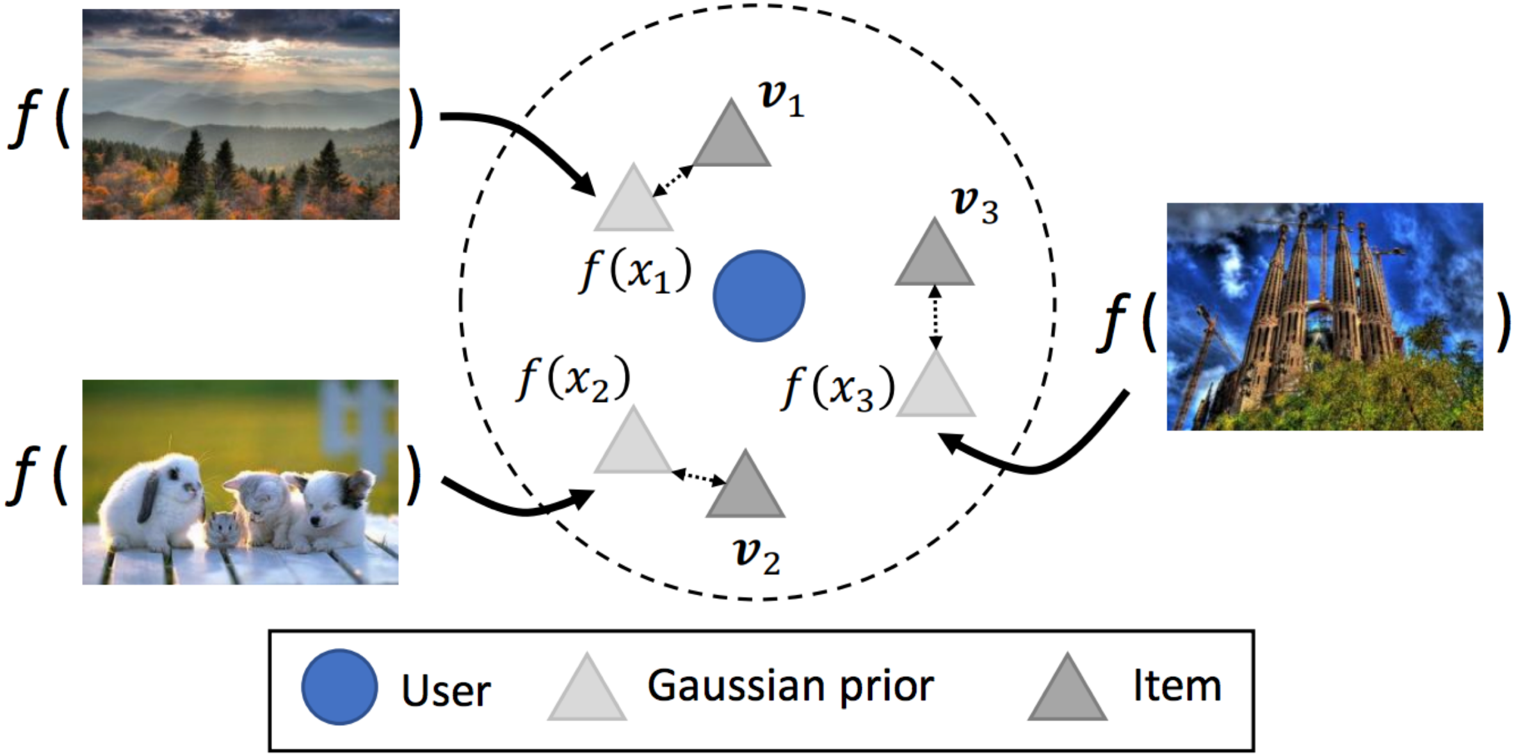
He, R. and McAuley, J., 2016, February. VBPR: Visual Bayesian Personalized Ranking from Implicit Feedback. In *AAAI* (pp. 144-150).

Collaborative Deep Learning (CDL)



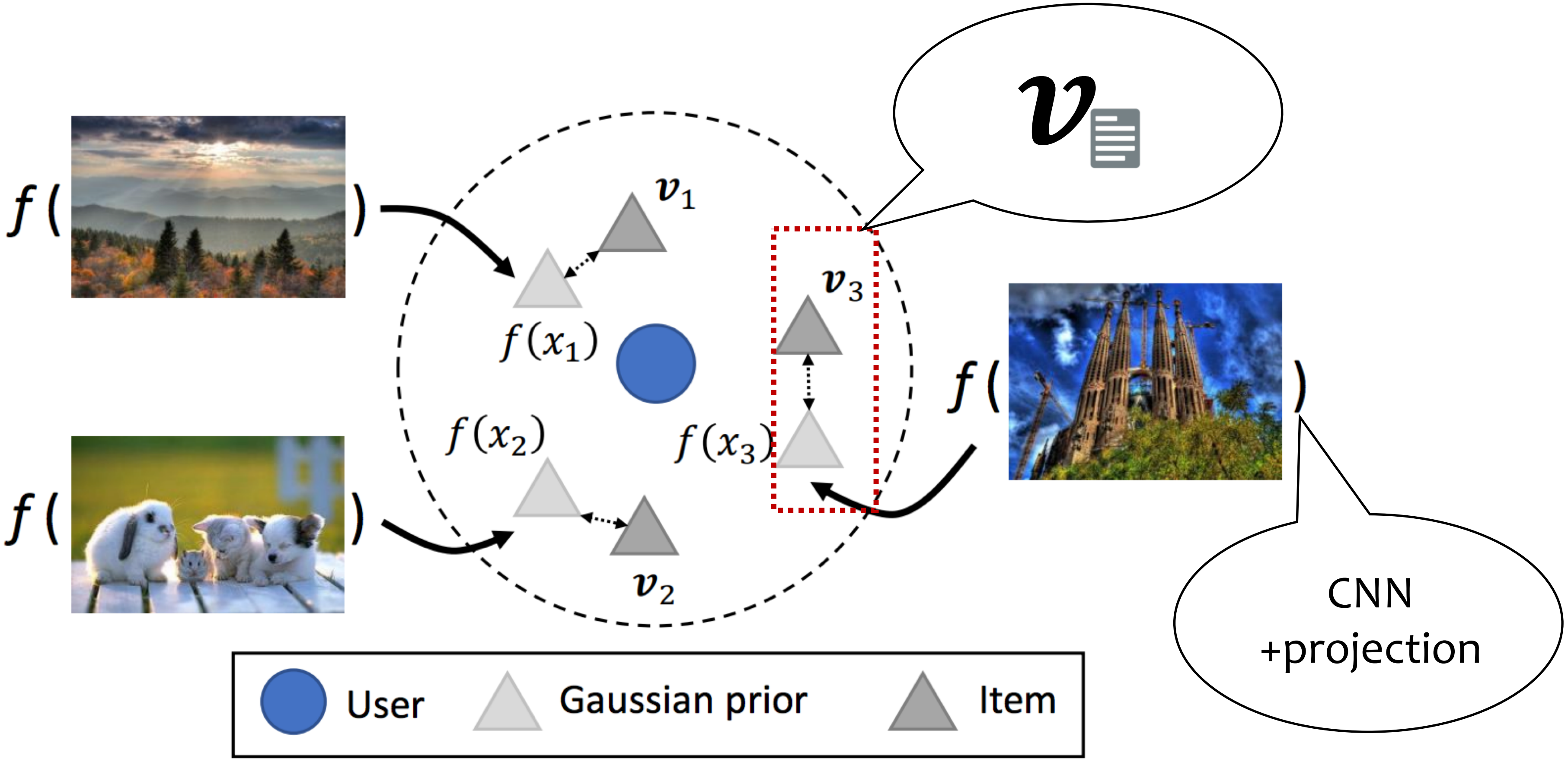
Wang et al. "Collaborative deep learning for recommender systems." *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015.

Collaborative Metric Learning (CML)



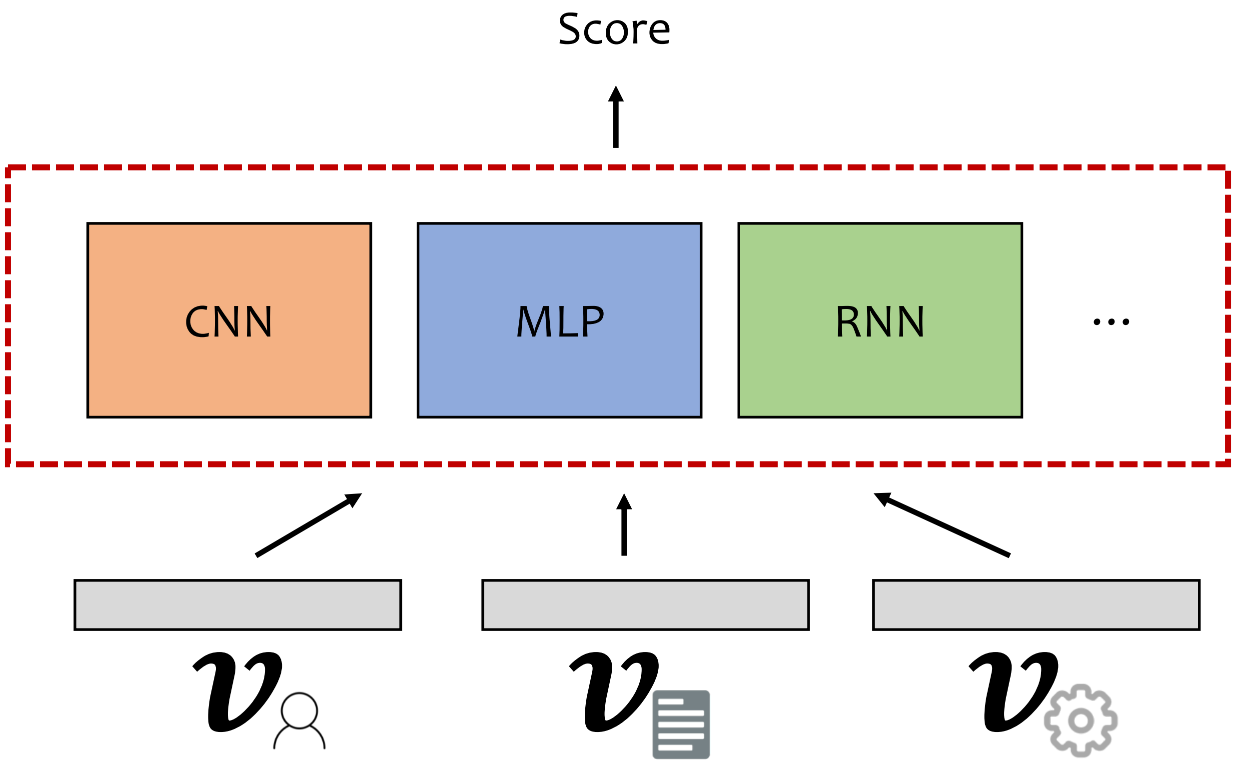
Hsieh, Cheng-Kang, et al. "Collaborative metric learning." *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017.

Collaborative Metric Learning (CML)

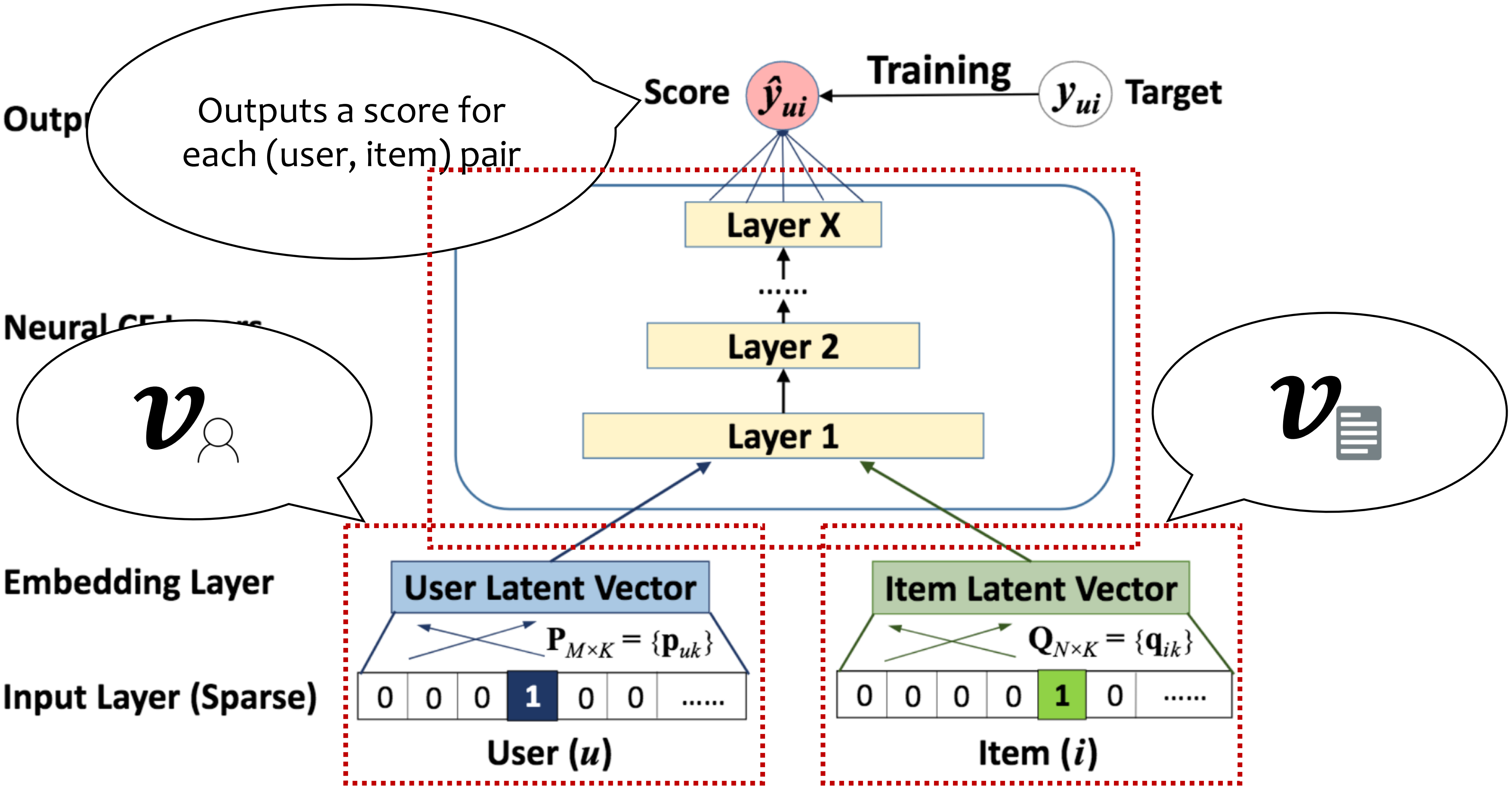


Hsieh, Cheng-Kang, et al. "Collaborative metric learning." *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017.

Nuanced user-item interaction patterns

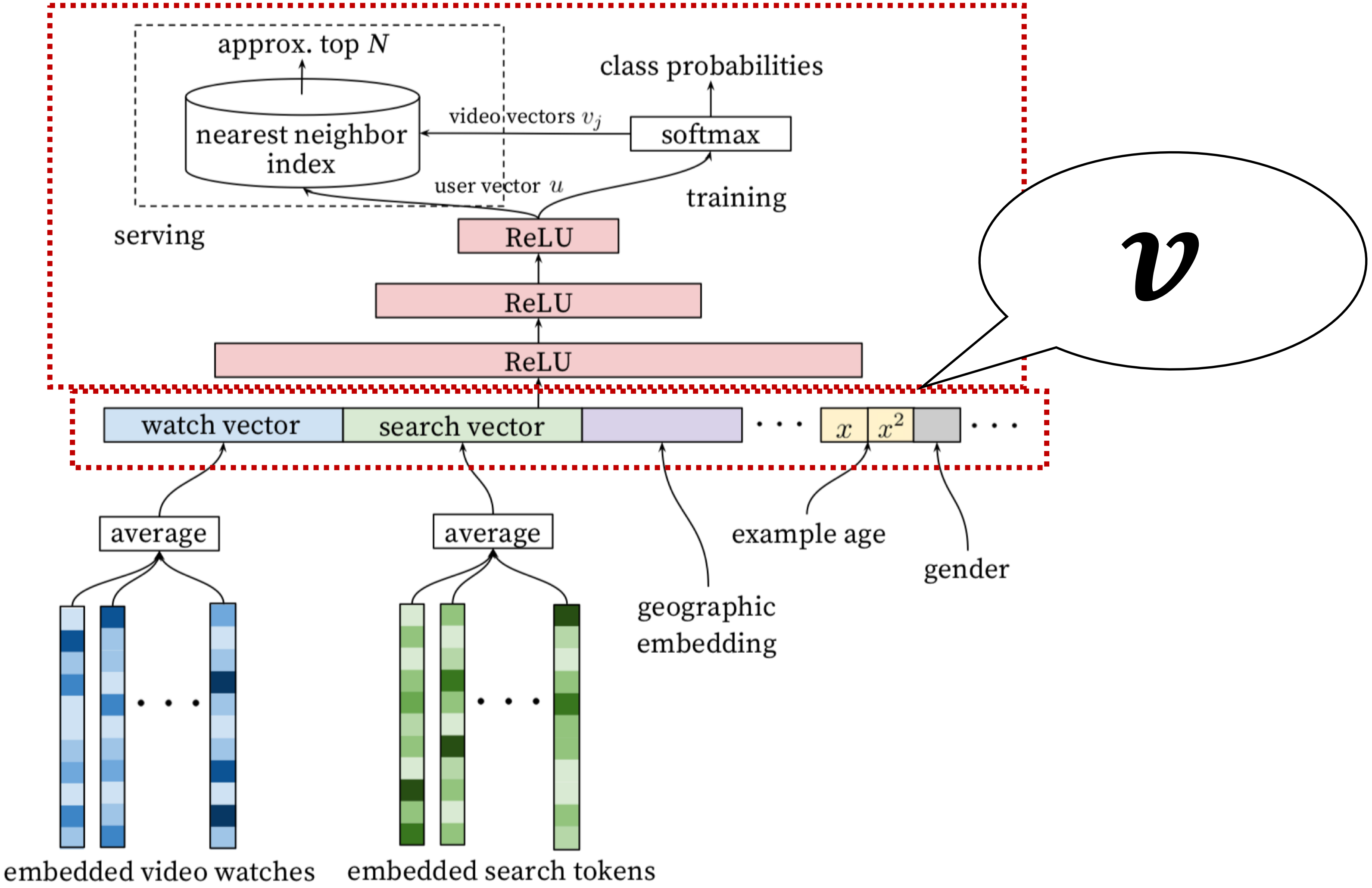


Neural Collaborative Filtering (NCF)



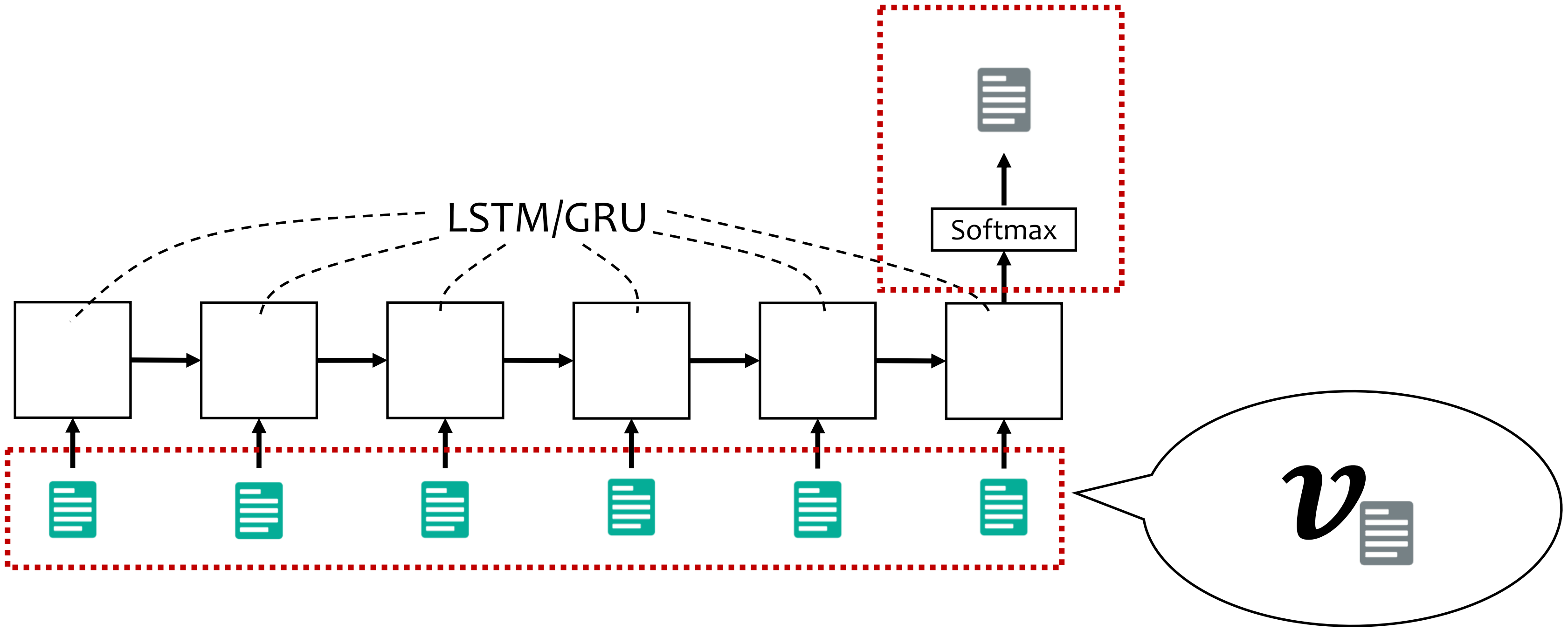
He, Xiangnan, et al. "Neural collaborative filtering." *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017.

Deep YouTube Video Recommendation



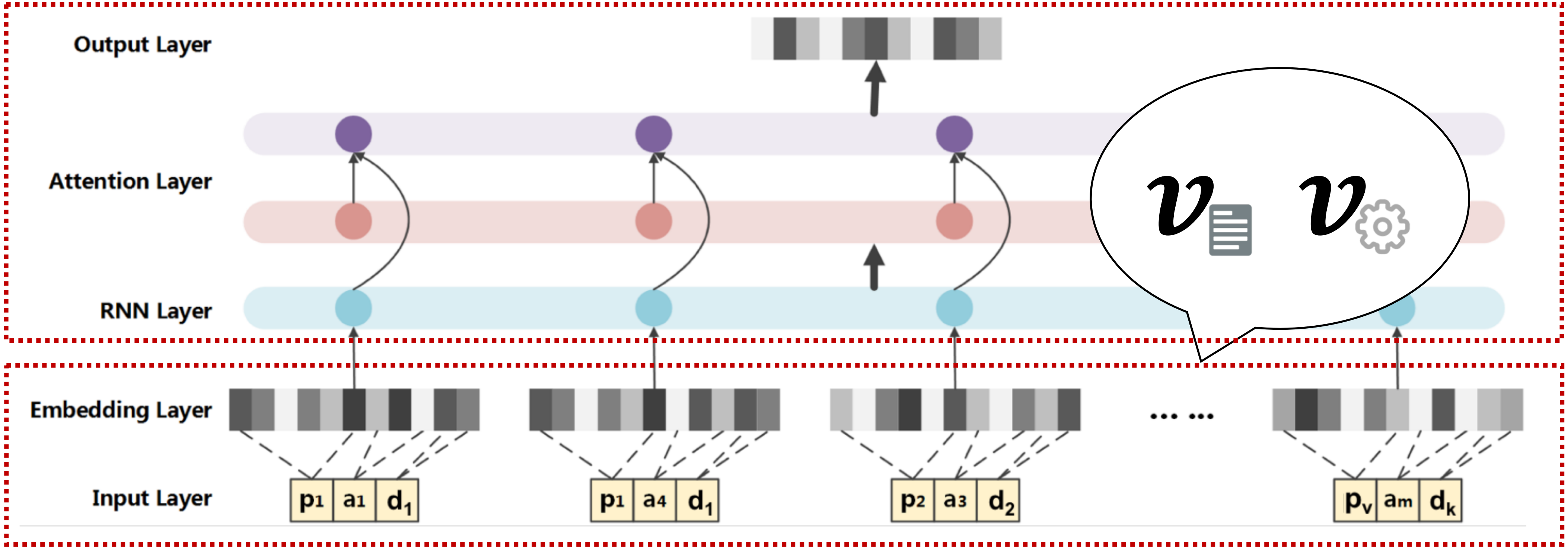
Covington, Paul, Jay Adams, and Emre Sargin. "Deep neural networks for youtube recommendations." *Proceedings of the 10th ACM Conference on Recommender Systems*. ACM, 2016.

Recurrent Recommender Networks



Wu, Chao-Yuan, et al. "Recurrent recommender networks." *Proceedings of the tenth ACM international conference on web search and data mining*. ACM, 2017.

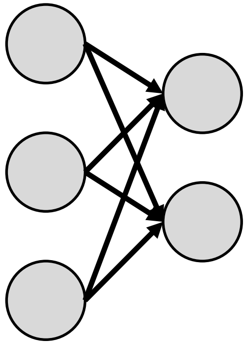
Micro behavior



Zhou et al. Micro behaviors: A new perspective in e-commerce recommender systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining* (pp. 727-735). ACM.

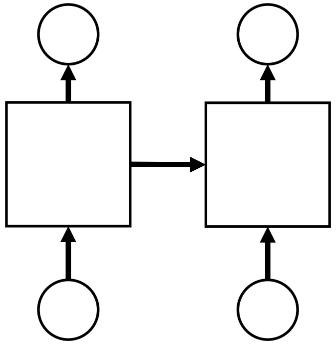
Building blocks

Multi-Layer Perceptron
(MLP)



Convolutional Neural Network
(CNN)

Recurrent Neural Network
(RNN)



Reinforcement Learning
(RL)

Lecture Agenda

- How we do recommendations, traditionally.
- How deep learning helps to do better recommendations.
- Modularization and OpenRec

Structurally Rethink Today's Recommendation Algorithms

Structurally Rethink Today's Recommendation Algorithms

Diverse user feedback signals

rating like skip follow

click save watch listen

…

Heterogeneous data streams and context

user demographics user social media posts

Item descriptions videos images

activities location mood …

Complex goals

accuracy novelty quality

diversity fairness interpretability …

Structurally Rethink Today's Recommendation Algorithms

Diverse user feedback signals

rating like skip follow

click save watch listen

…

Heterogeneous data streams and context

user demographics user social media posts

Item descriptions videos images

activities location mood …

Complex goals

accuracy novelty quality

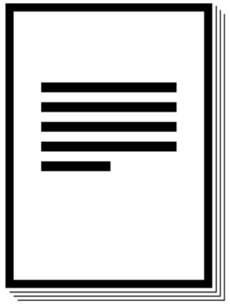
diversity fairness interpretability …

…barriers on harnessing and experimenting with state-of-the-art models

Current practice

Prior research | **Your research/application**

current practice

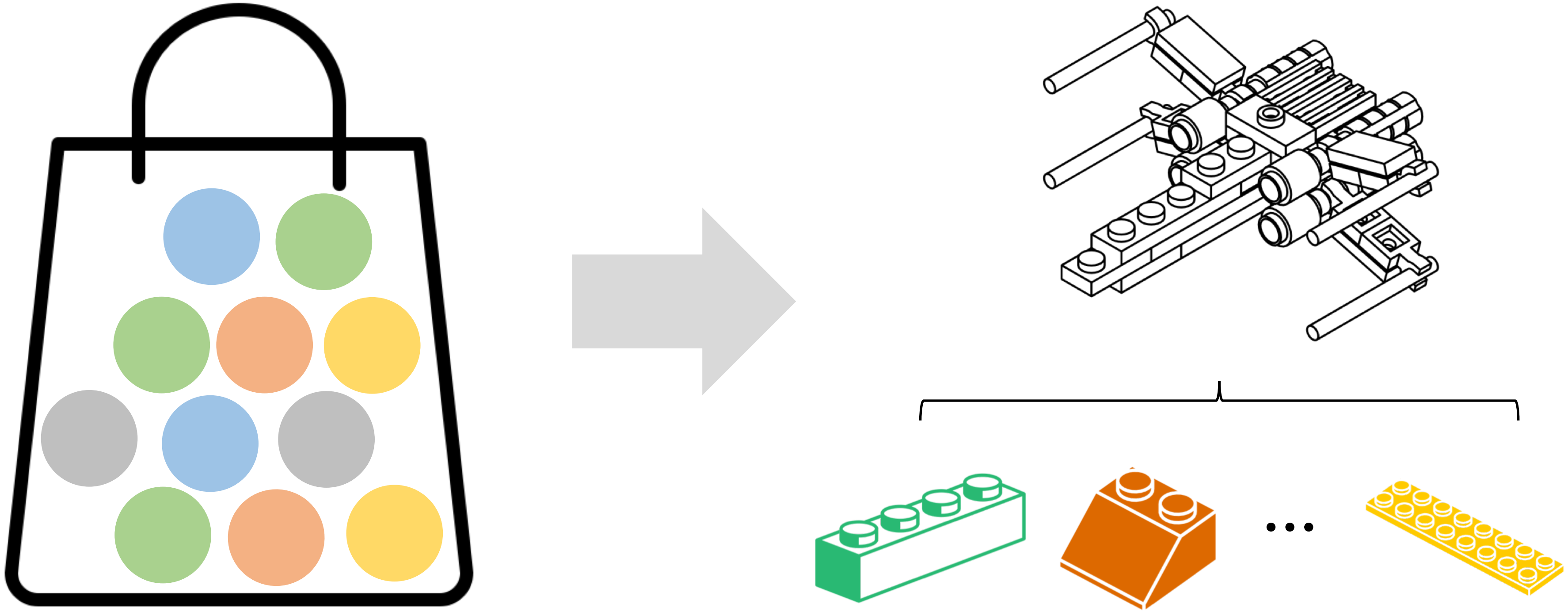


News recommender → Music recommender

different user feedback signals
different data sources
tangled implementations

Bag of algorithms

Ideally, building a recommendation algorithm can maximally reuse <u>existing modules</u>



Bag of algorithms

OpenRec

Apache License 2.0

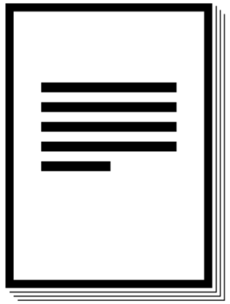
(Built on Tensorflow)

Modularity

- Easy to **extend** and **adapt** to various scenarios.
- Quick experimentation (e.g., model selection) and idea exploration.
- Comparable (sometimes even better) performance.

Contribution of OpenRec

current practice



	Prior research		**Your research/application**
	News recommender	→	Music recommender

different user feedback signals
different data sources
tangled implementations

Contribution of OpenRec

Prior research | **Your research/application**

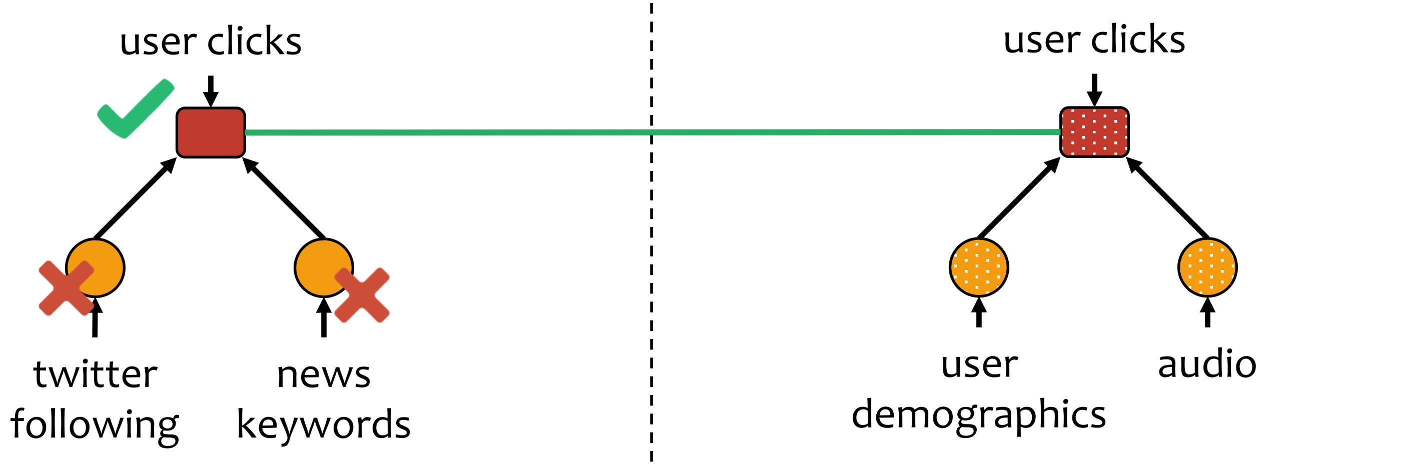
current practice

News recommender → Music recommender

?

different user feedback signals
different data sources
tangled implementations

OpenRec



Dataflow Graph (Computational Graph):
Specify dependencies between operations

$$d = (a + b) * c$$

$\boldsymbol{a}$	$\boldsymbol{b}$	$\boldsymbol{c}$
0.4	0.6	2.5
1.2	4.5	2.0
4.1	0.2	8.0

$\vdots$

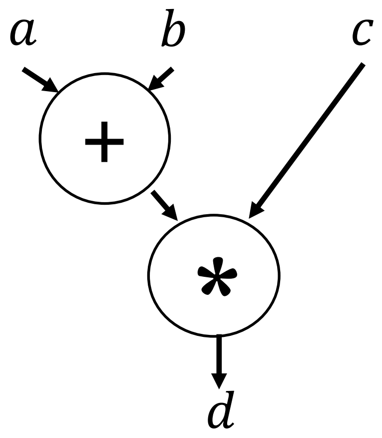
Dataflow Graph (Computational Graph): Specify dependencies between operations

Step 1. define inputs: a, b, c

Step 2. define computational graph:

$$\text{sum\_a\_b} = \text{add}(a + b)$$

$$d = \text{multiply}(\text{sum\_a\_b}, c)$$



Step 3. run computational graph with different inputs:

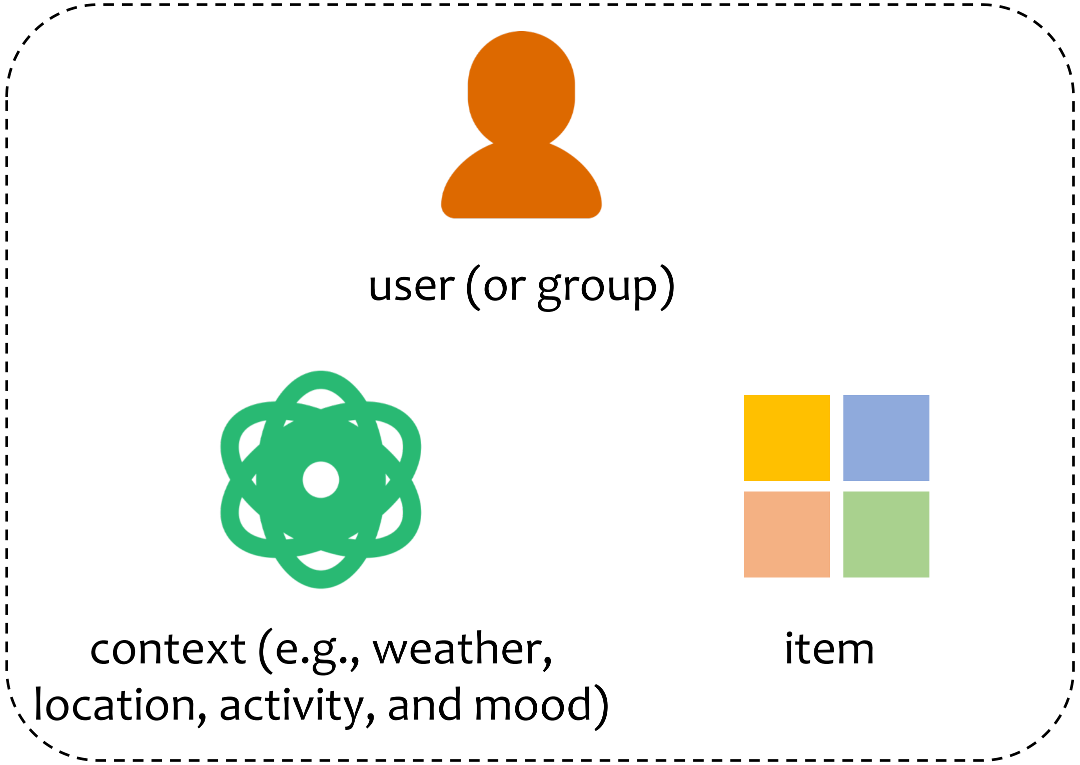
$$\text{result} = \text{run}(d, \{a = 0.4, b = 0.6, c = 2.5\})$$

$$\text{result} = \text{run}(d, \{a = 1.2, b = 4.5, c = 2.0\})$$

$$\text{result} = \text{run}(d, \{a = 4.1, b = 0.2, c = 8.0\})$$

Computational Graphs in OpenRec

Build a (train/serve) graph: Entities

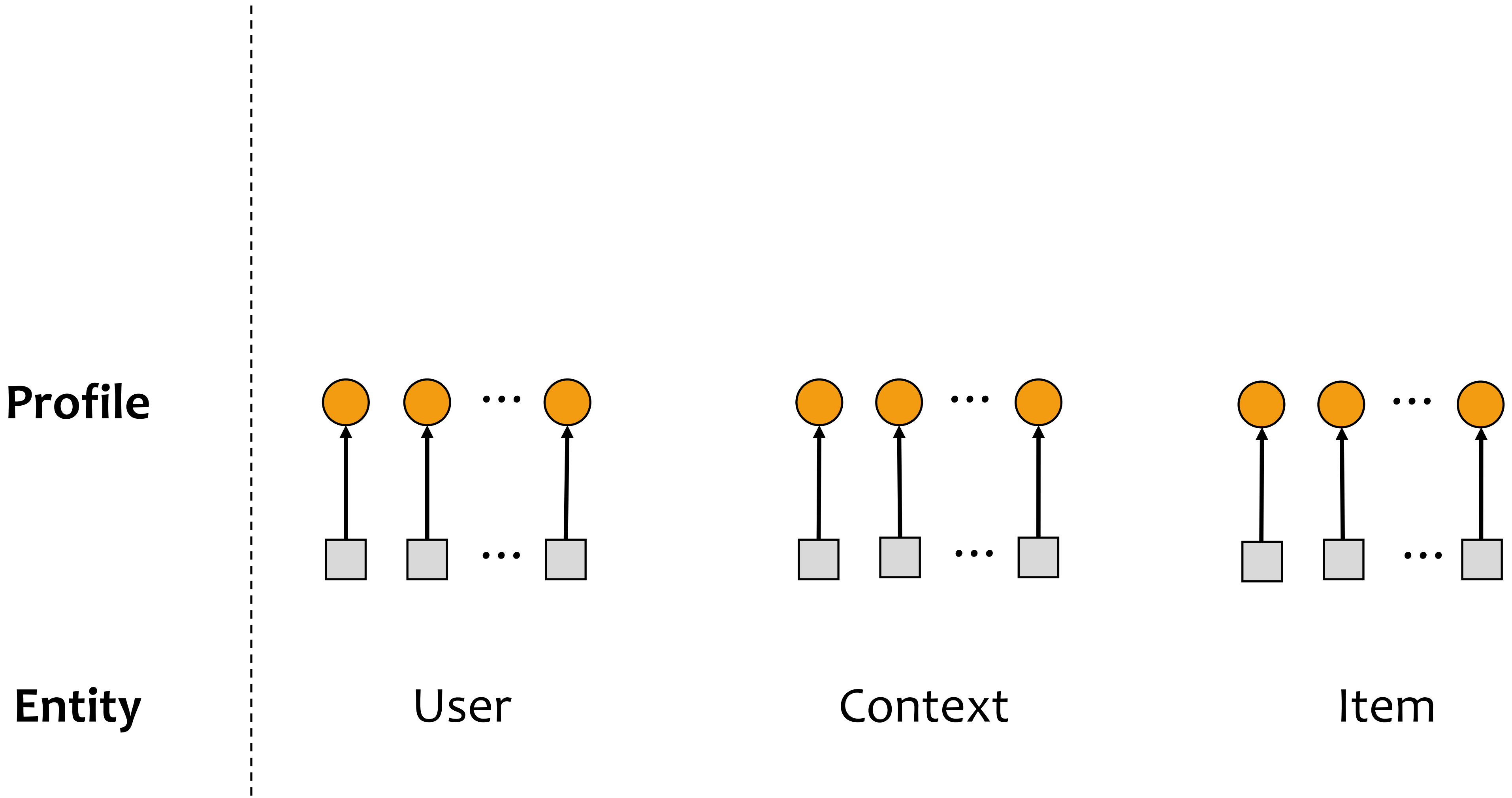


Build a (train/serve) graph: Reusable modules

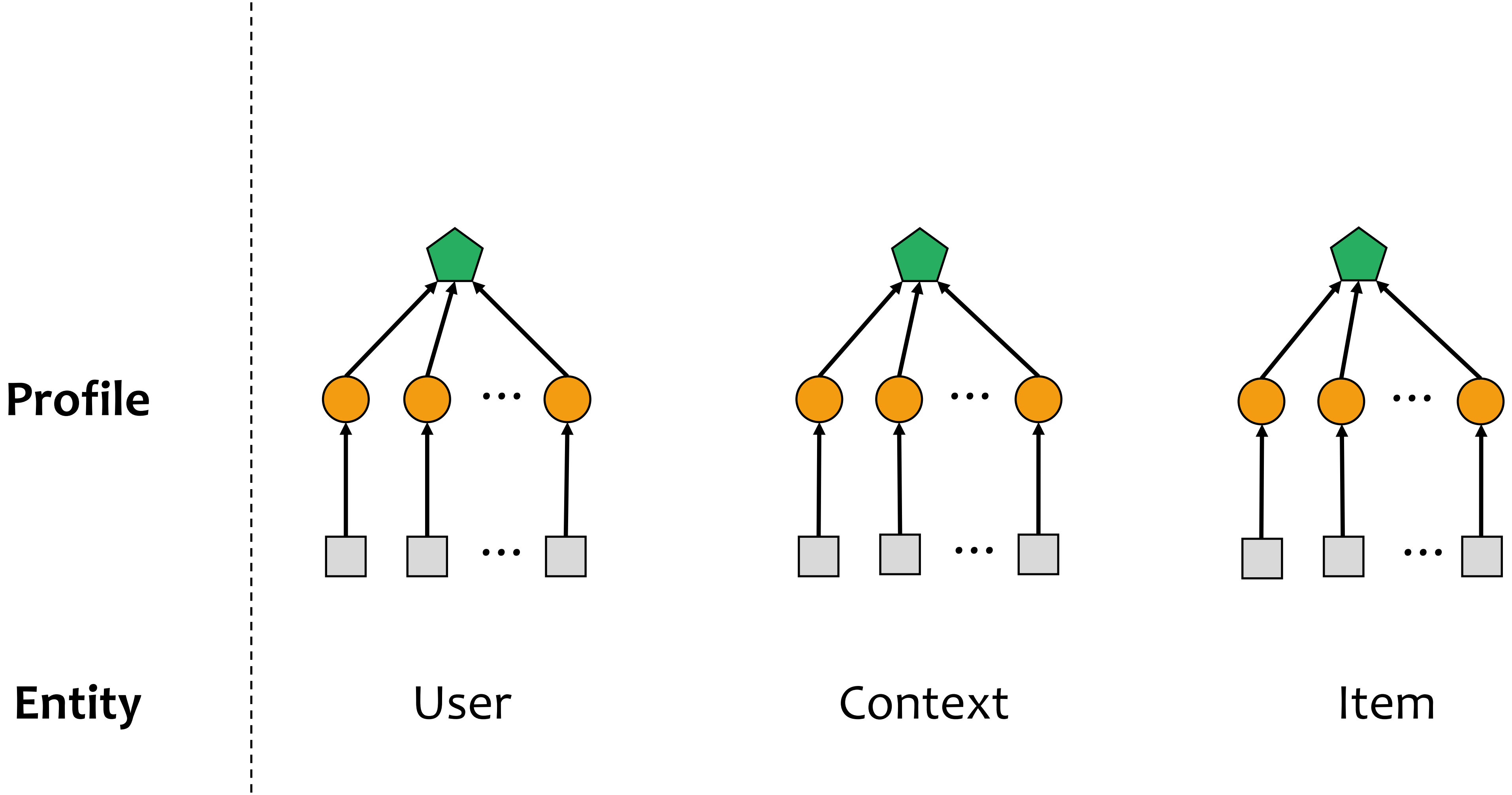
Profile			
Entity	User	Context	Item

Build a (train/serve) graph: Reusable modules

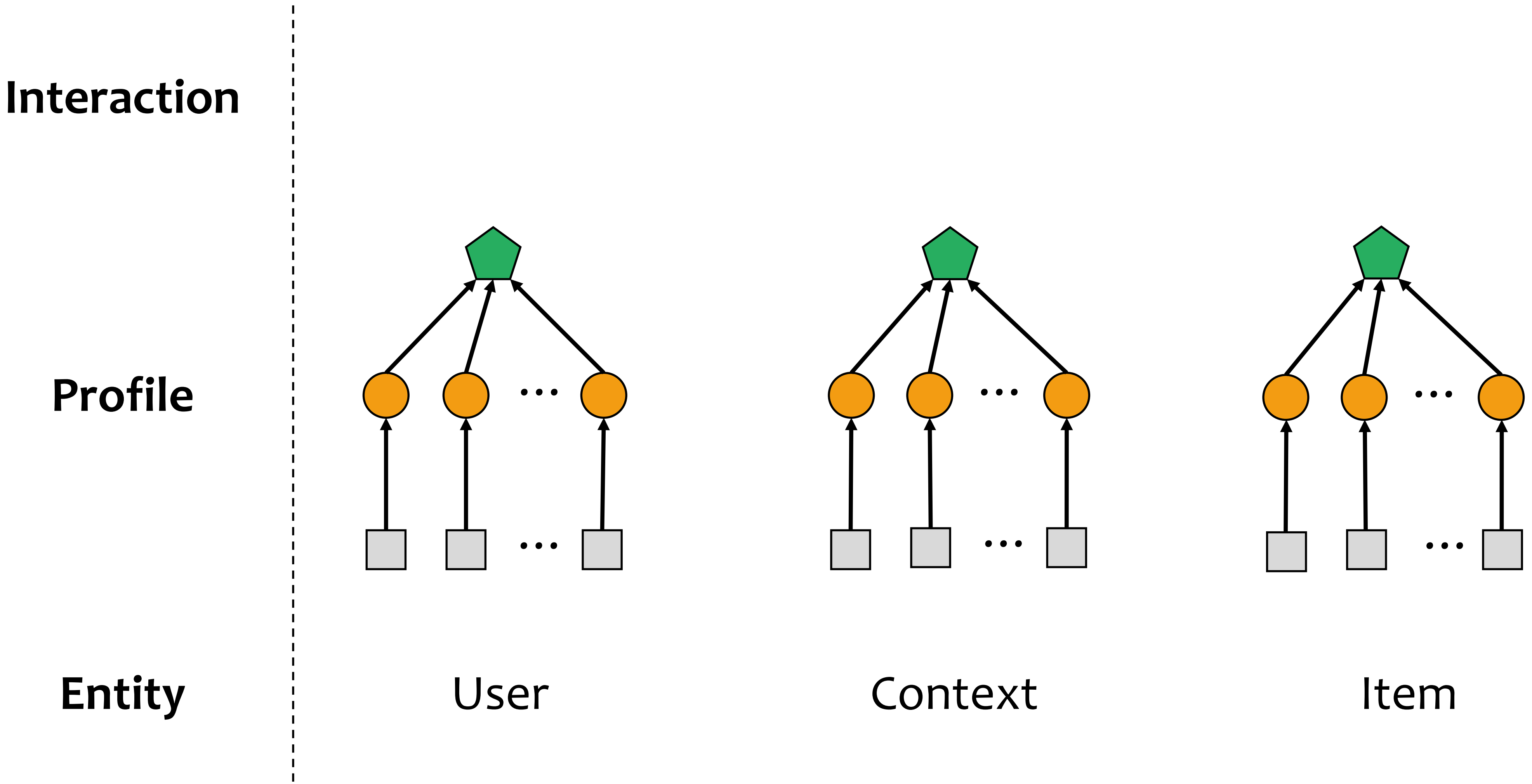
Build a (train/serve) graph: Reusable modules



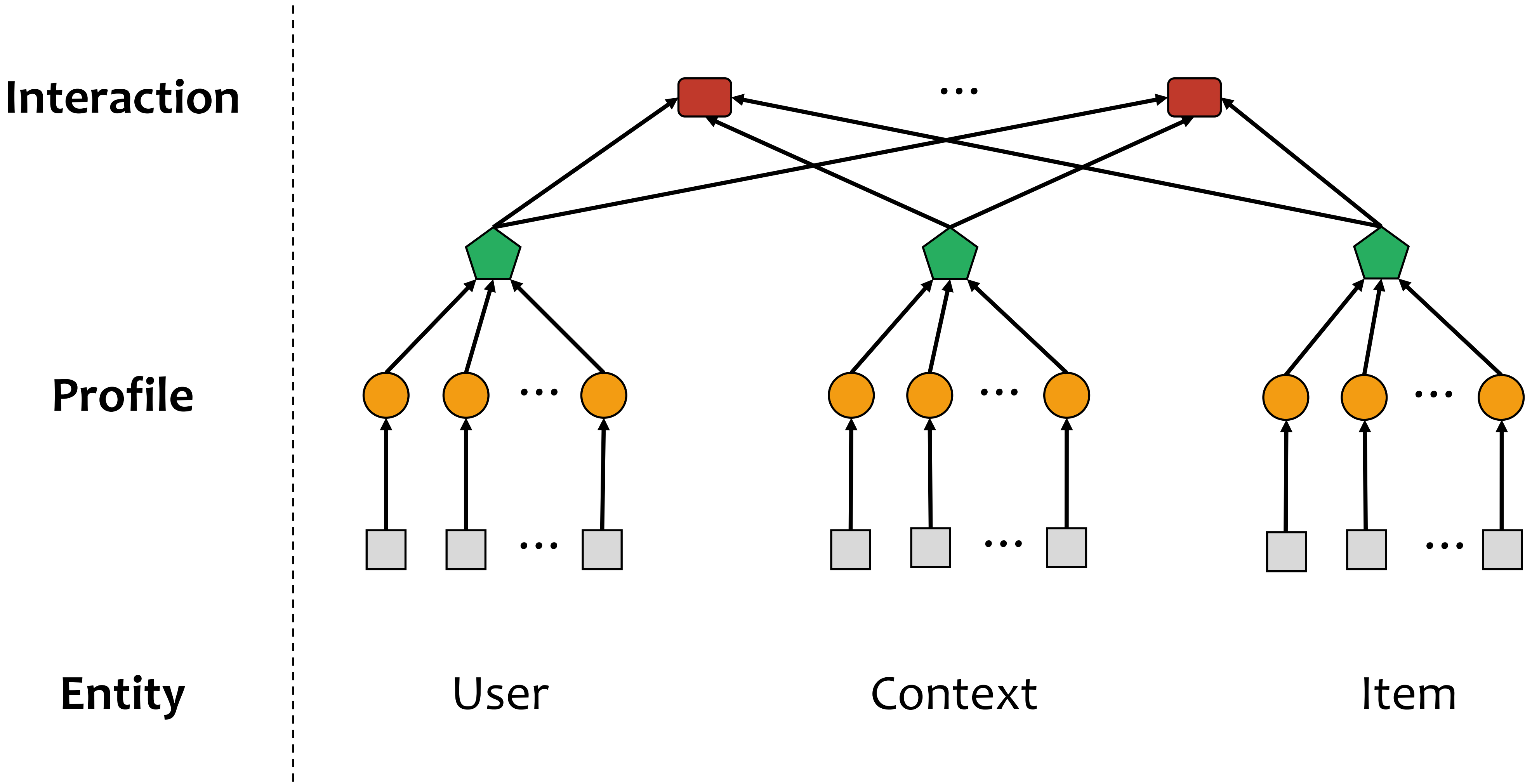
Build a (train/serve) graph: Reusable modules



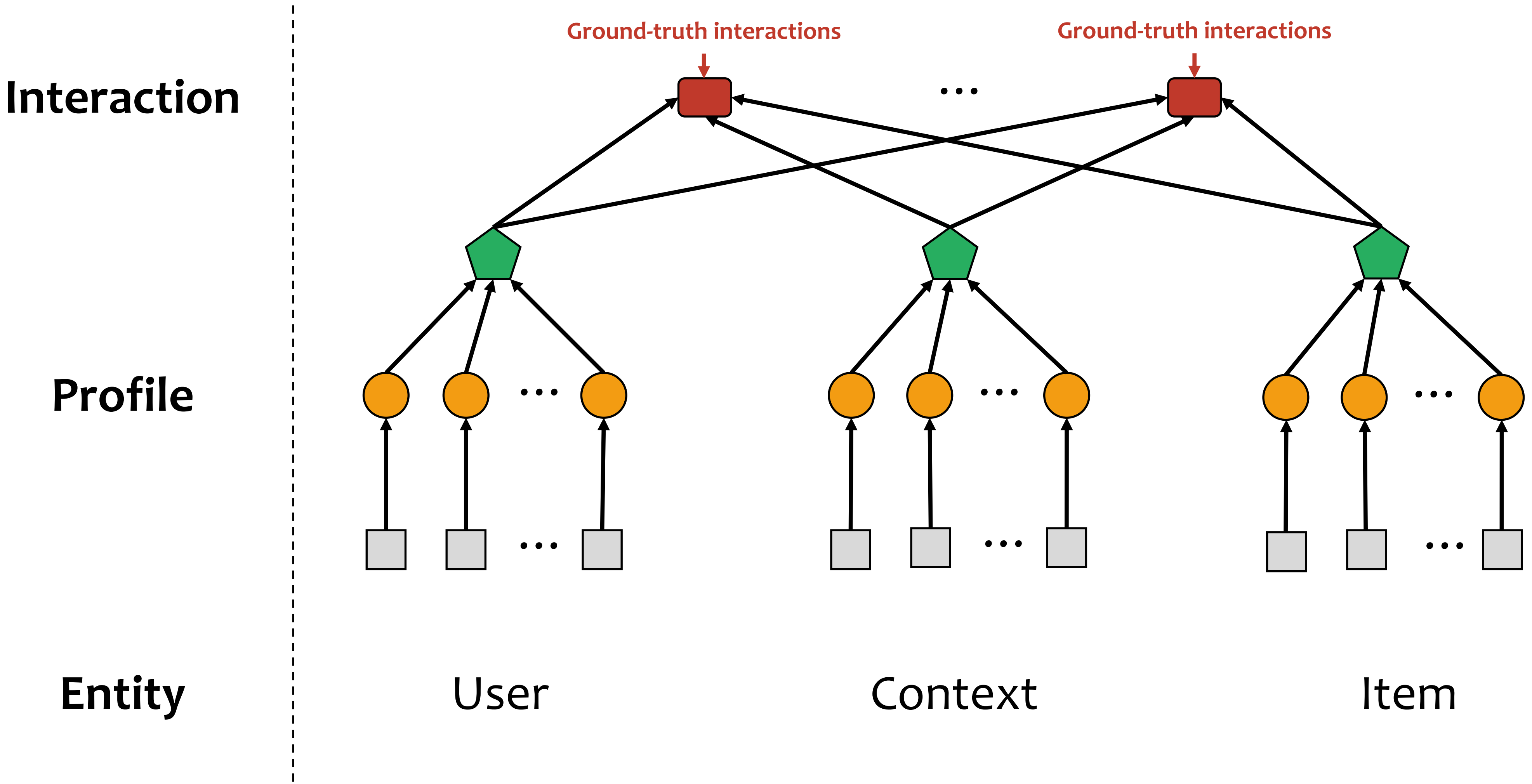
Build a (train/serve) graph: Reusable modules



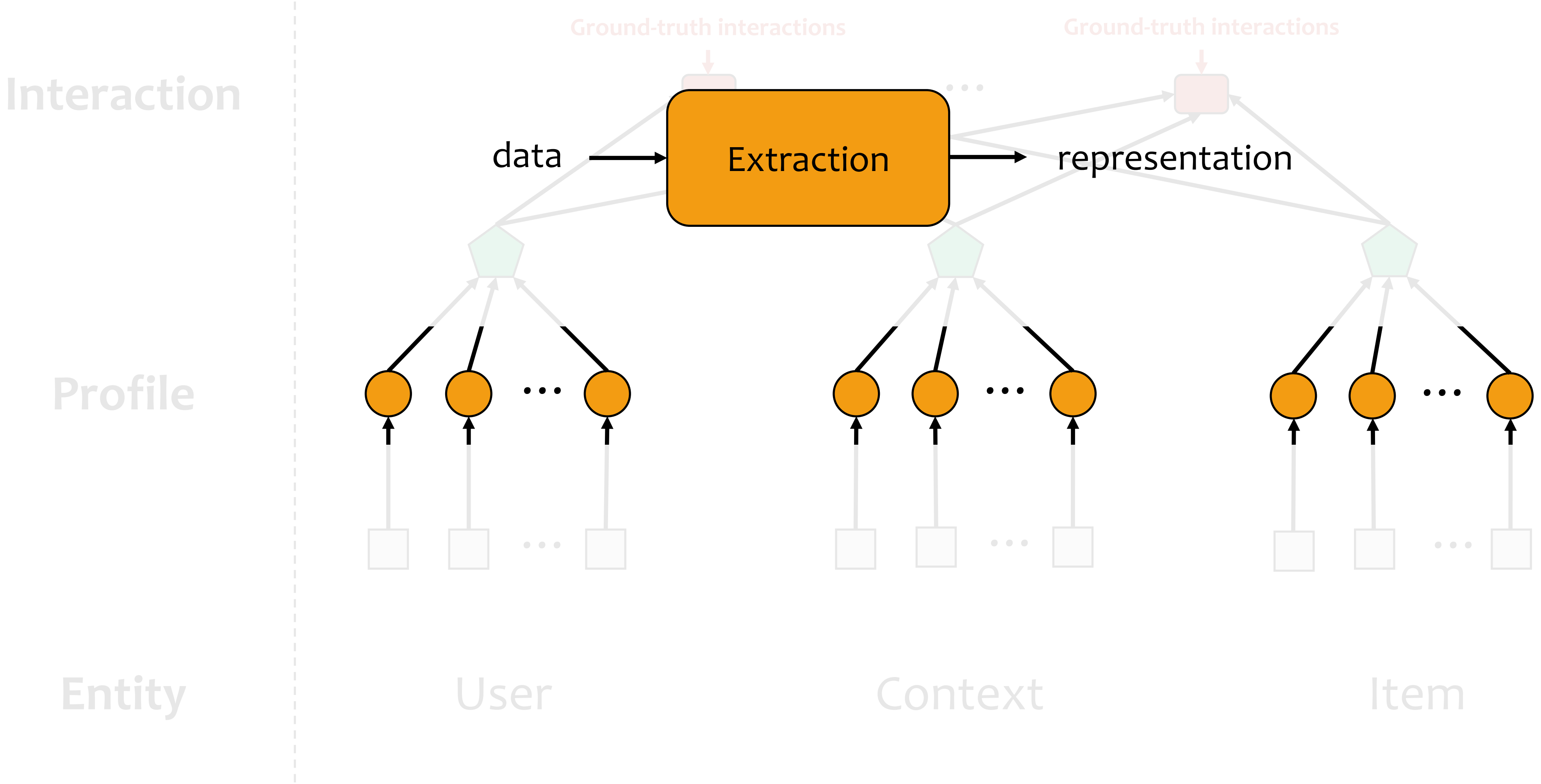
Build a (train/serve) graph: Reusable modules



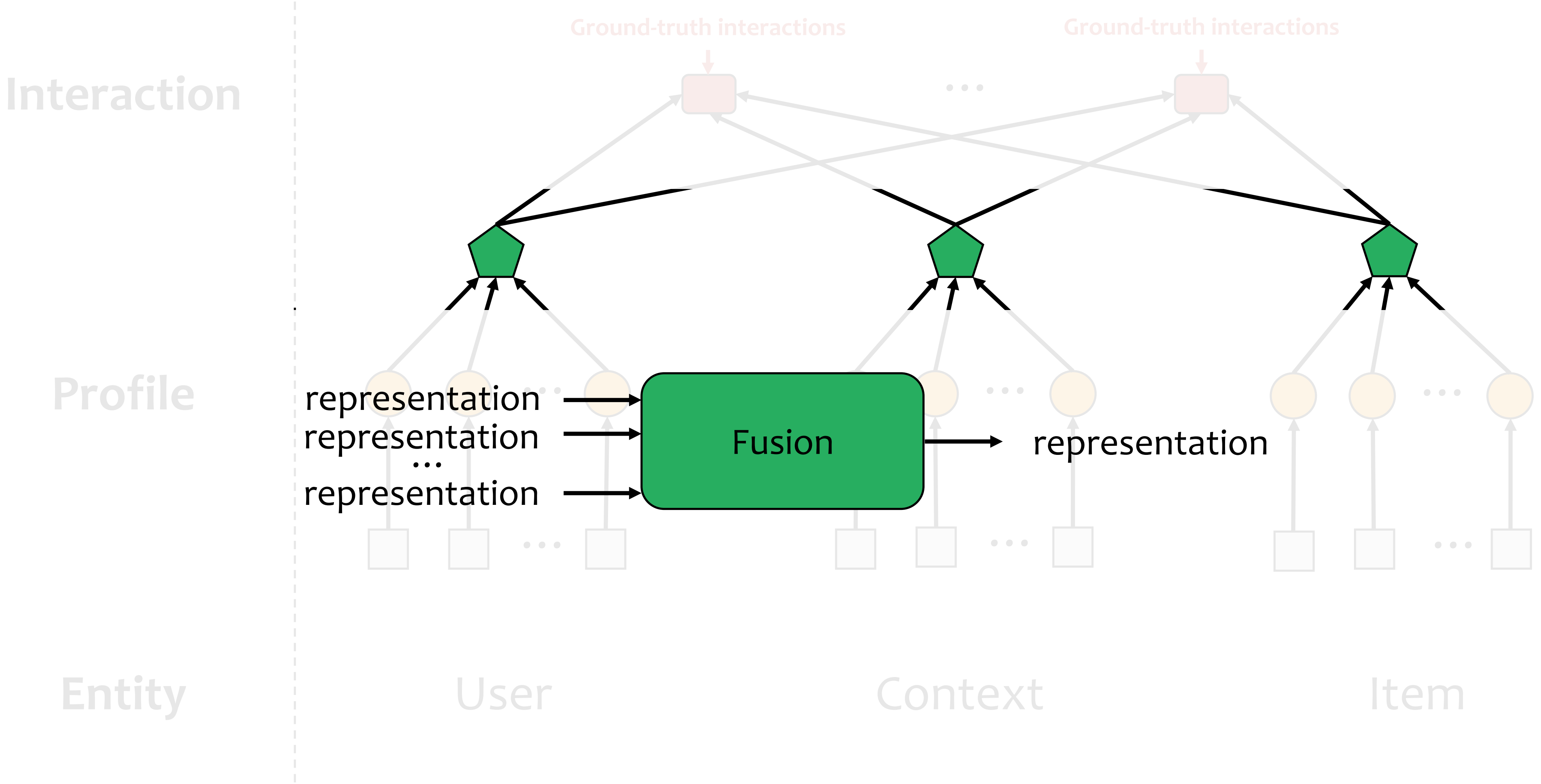
Build a (train/serve) graph: Reusable modules



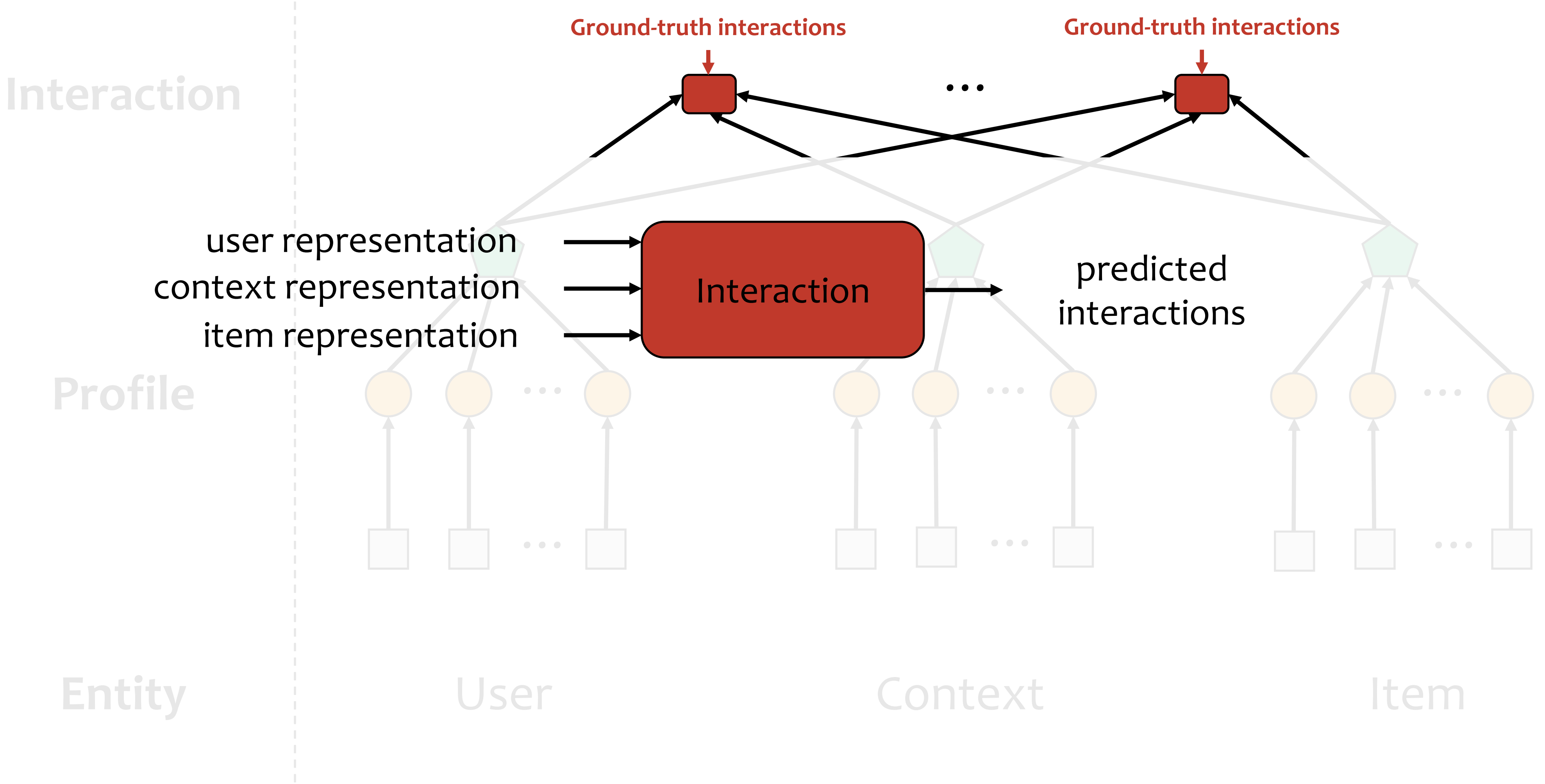
Extraction: extract representations



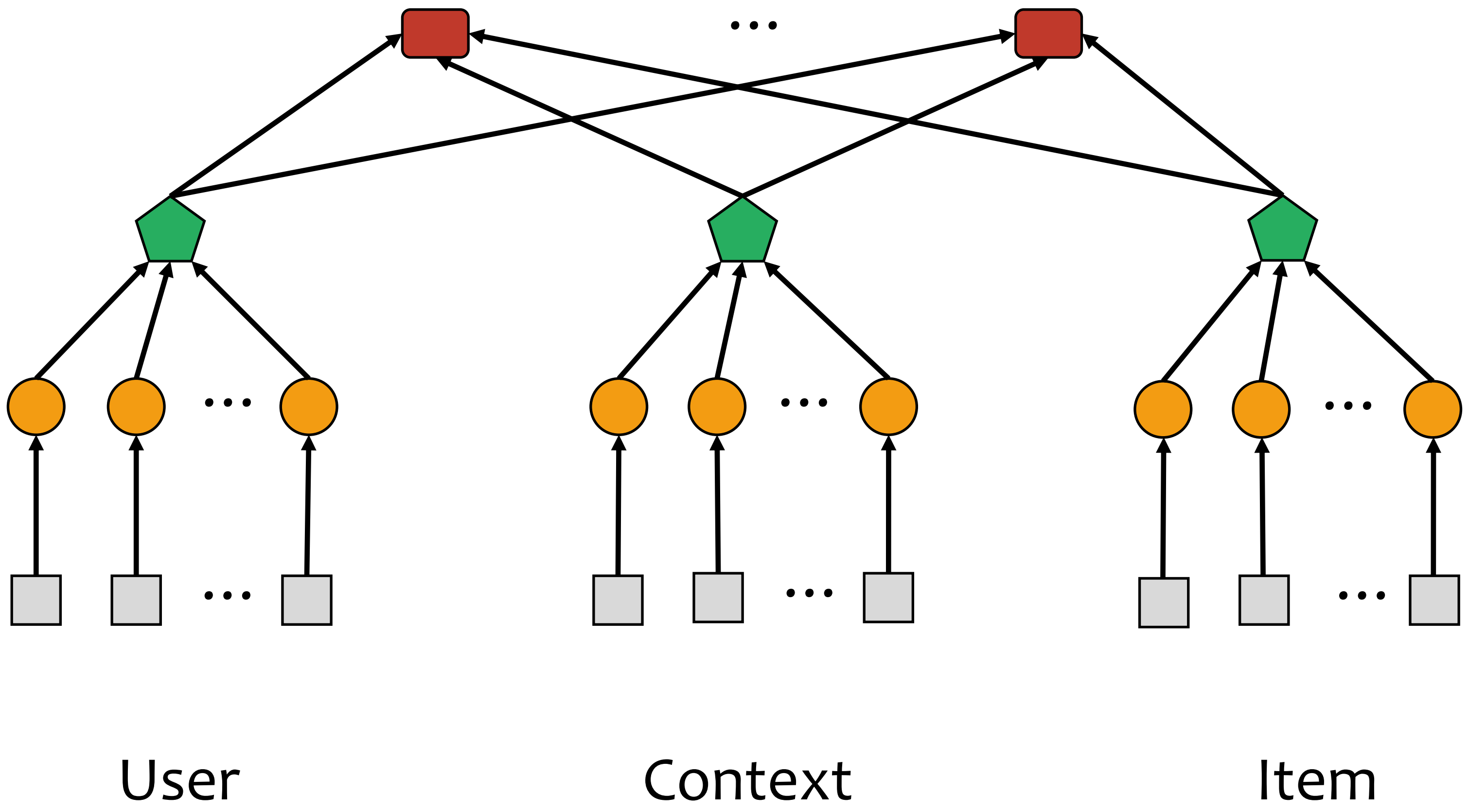
Fusion: fuse representations



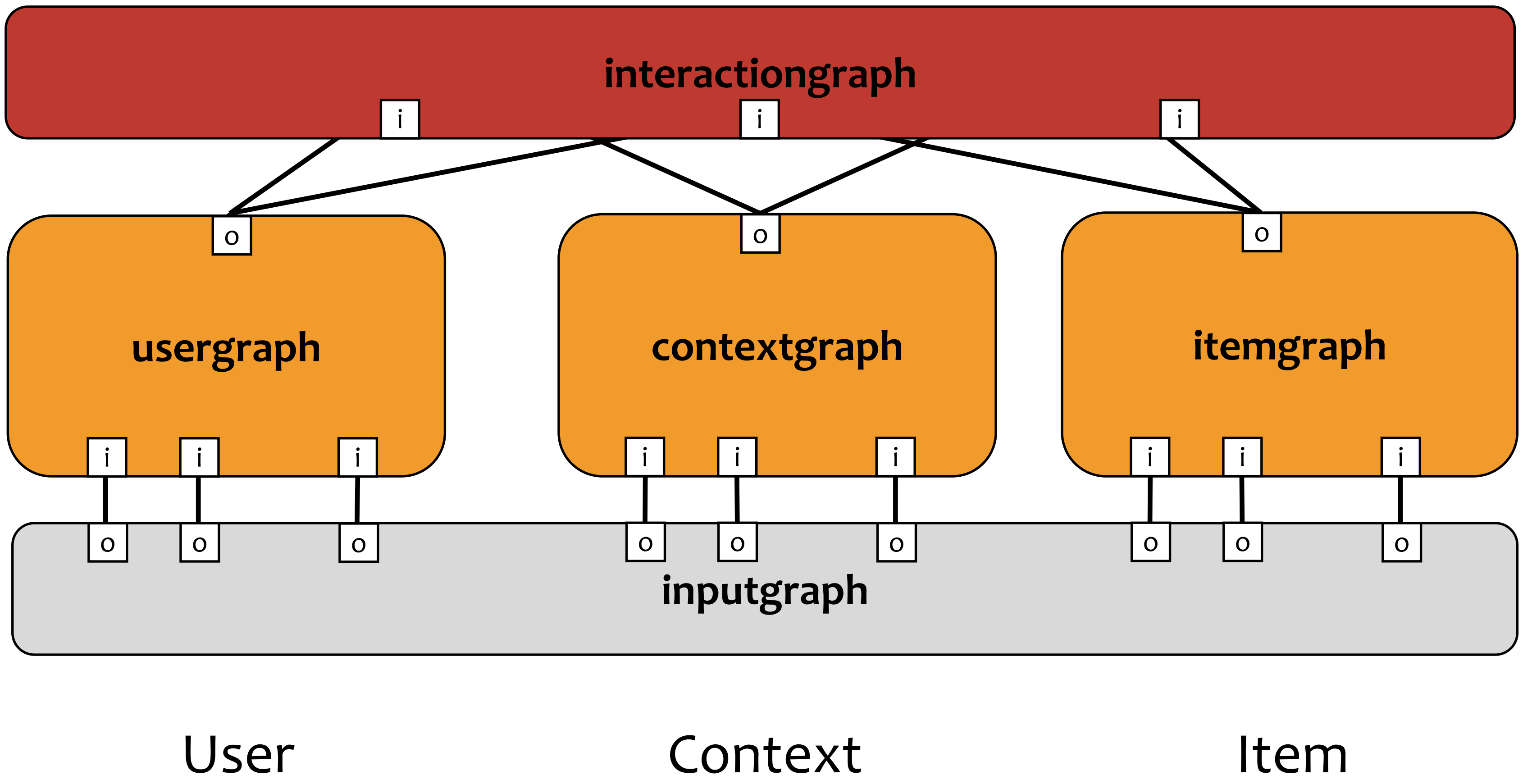
Interactions, e.g., clicks/likes/ratings…



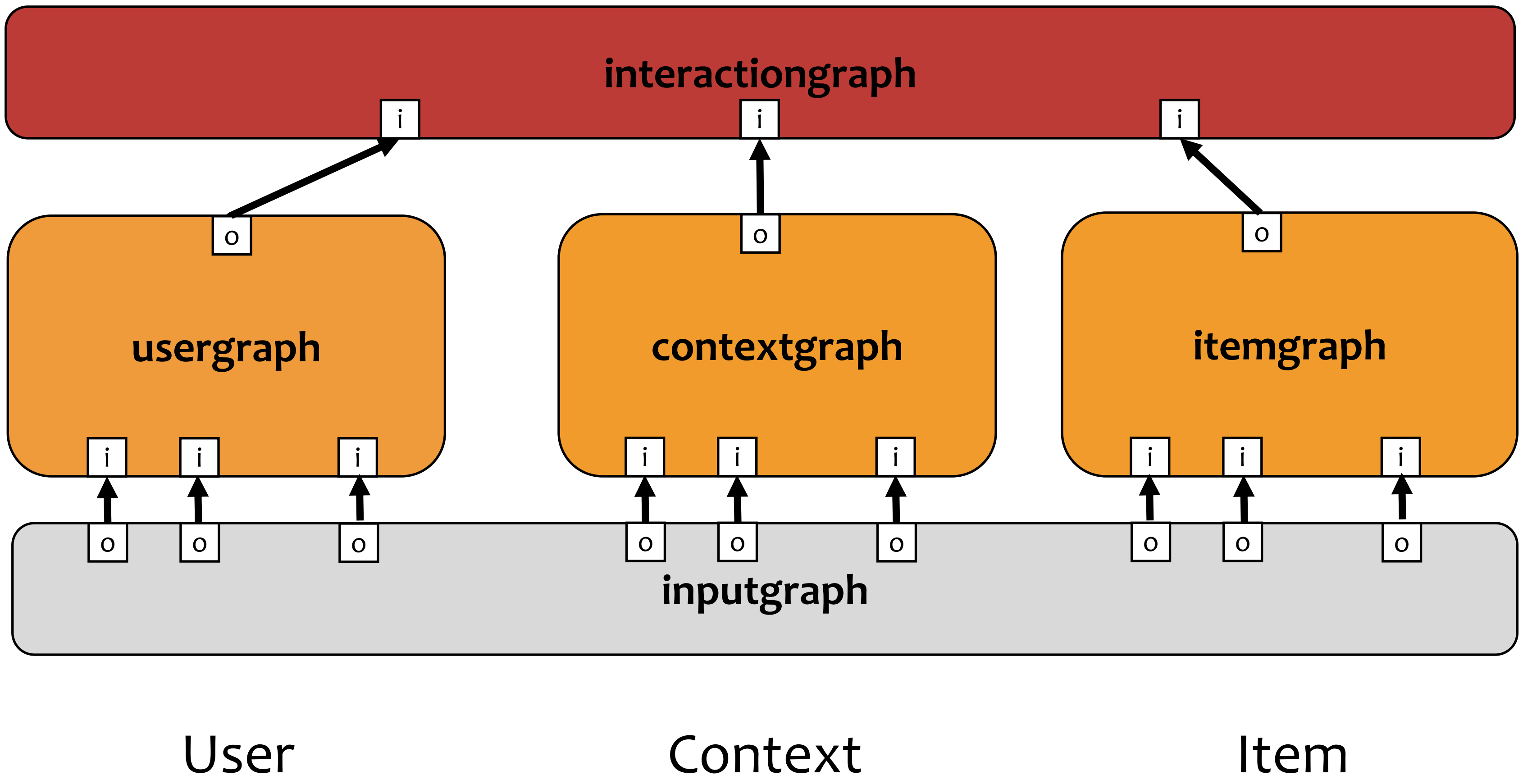
Build a (train/serve) graph: Reusable subgraphs



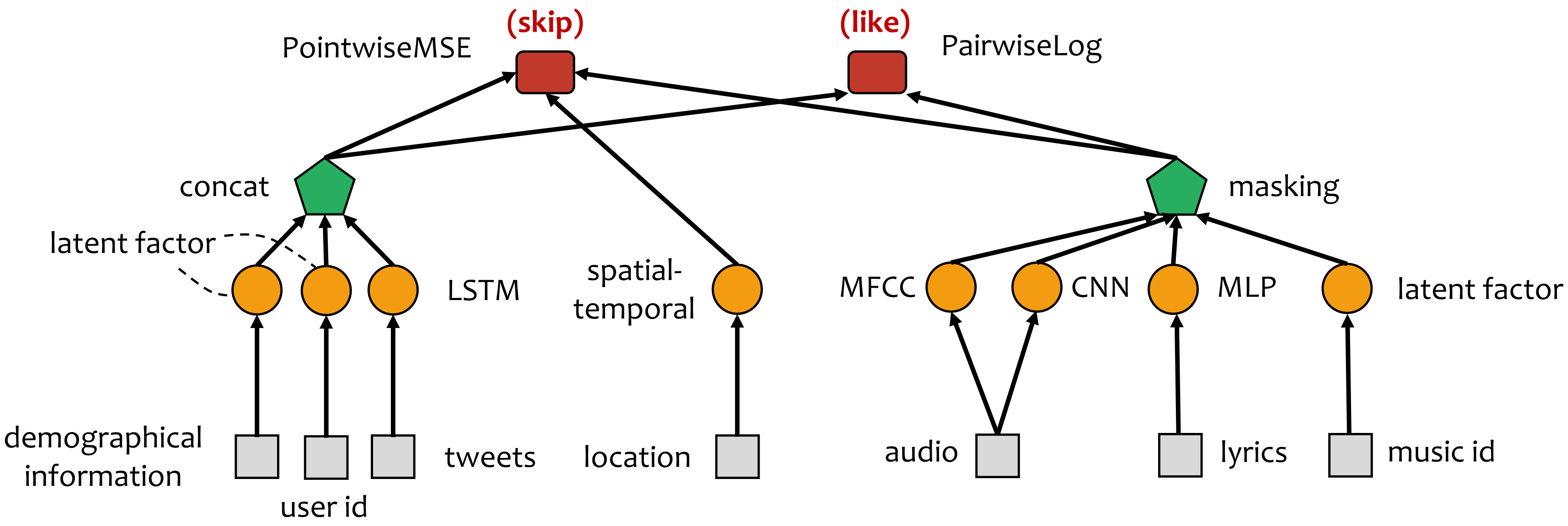
Build a (train/serve) graph: Reusable subgraphs



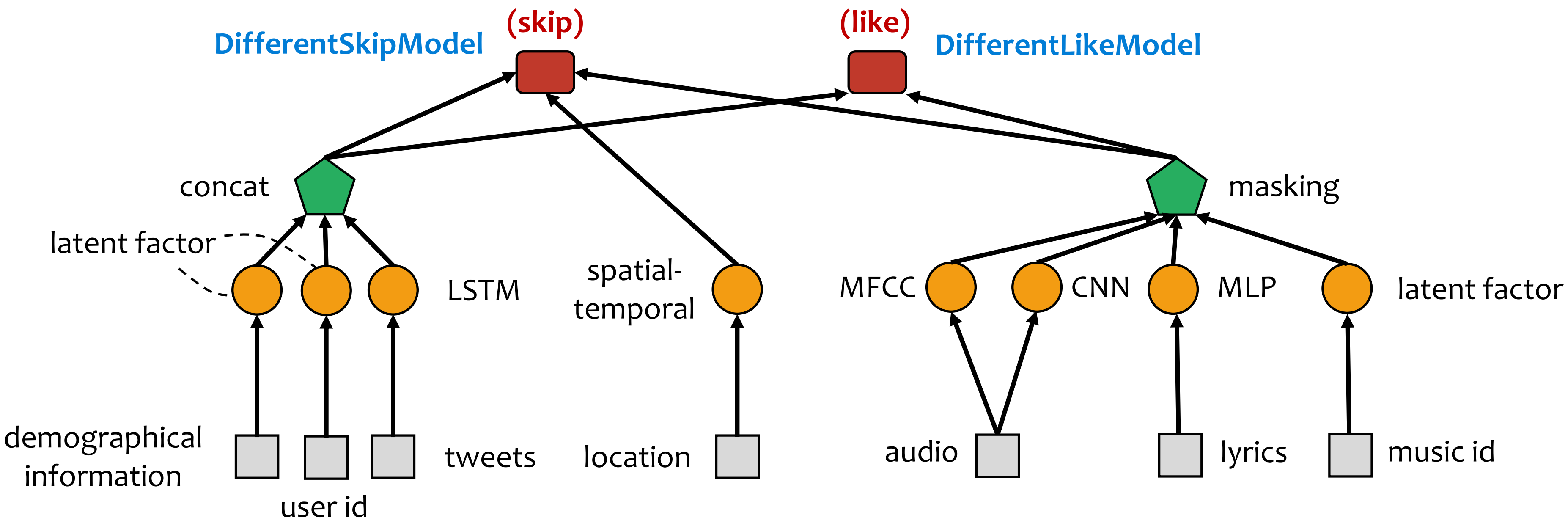
Build a (train/serve) graph: Reusable subgraphs



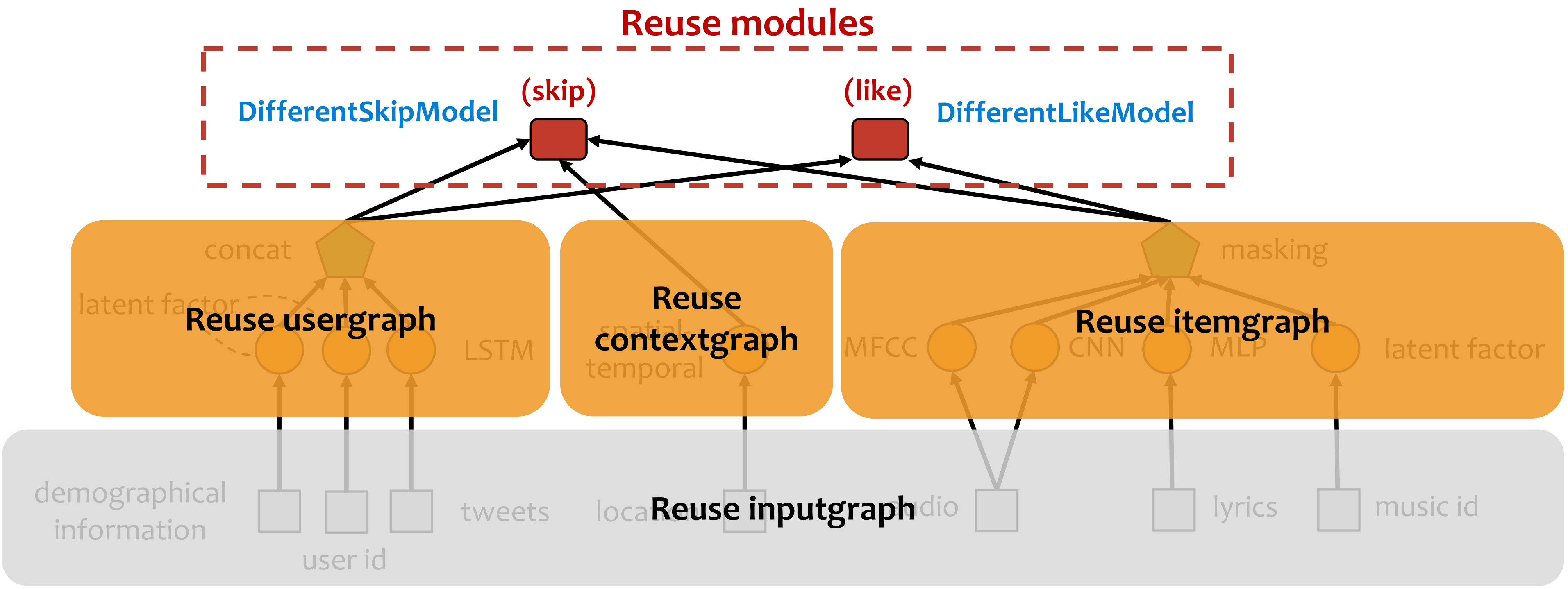
A hypothetical servegraph of a music recommender



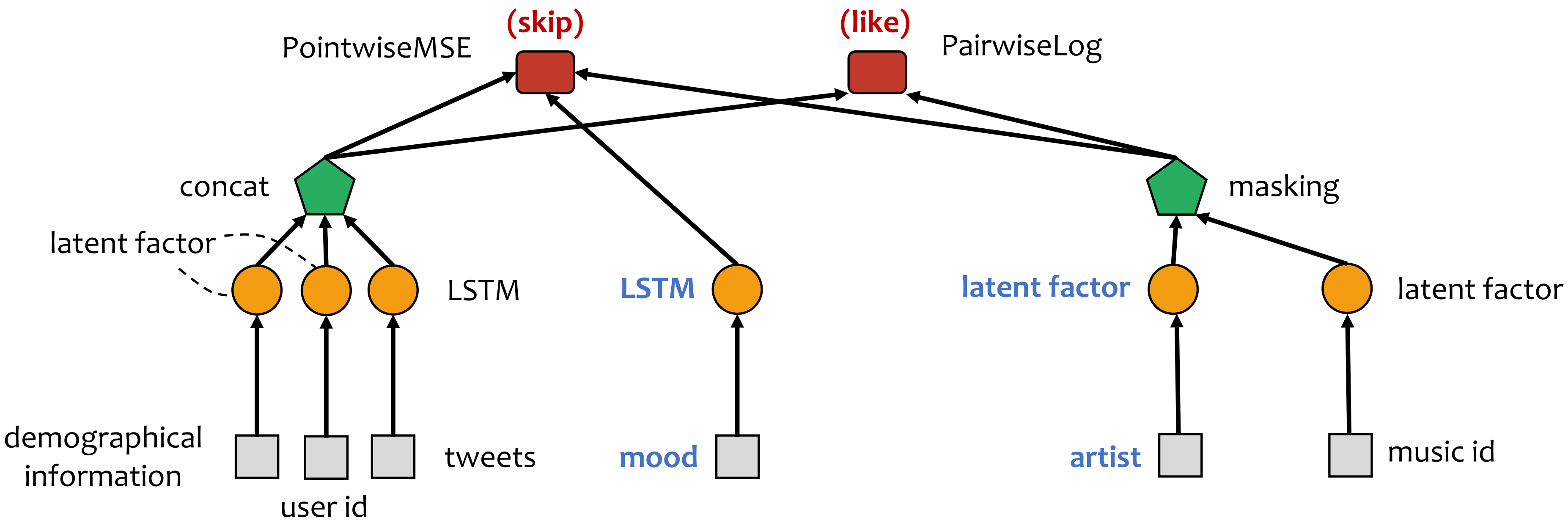
A hypothetical servegraph of a music recommender



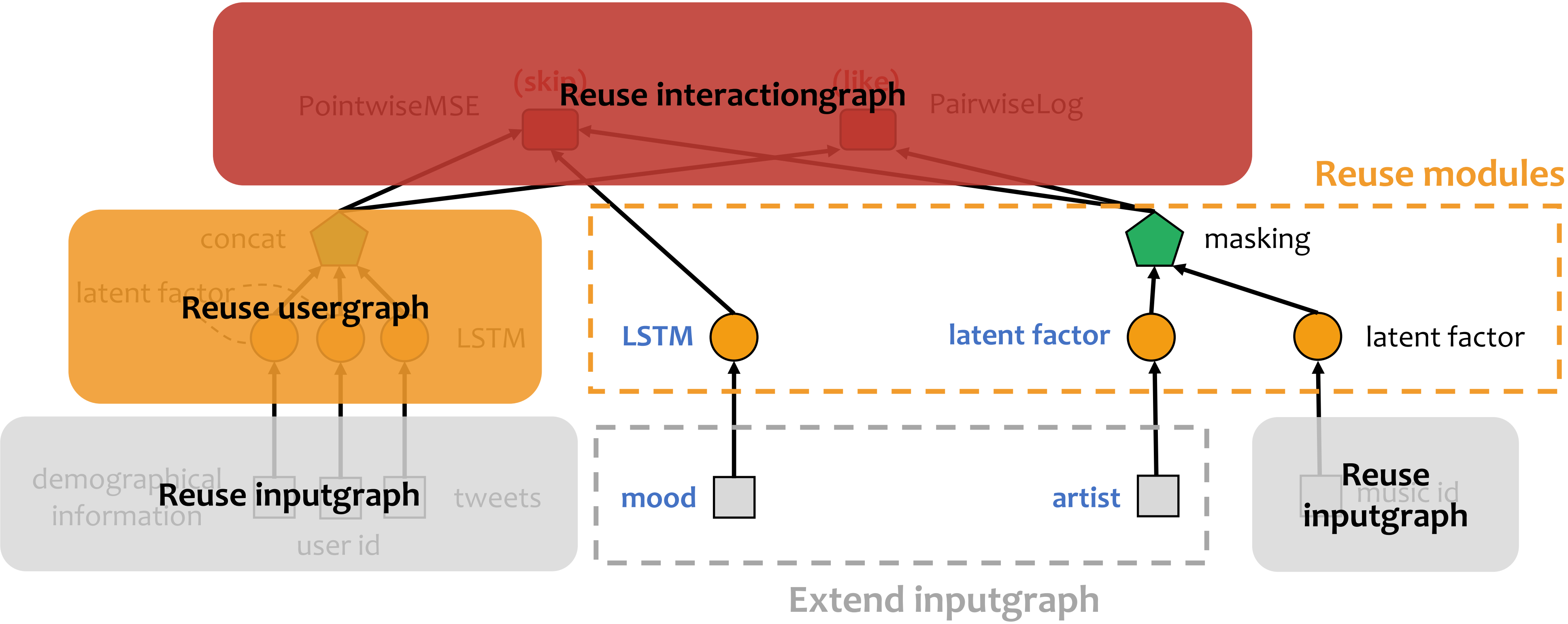
A hypothetical servegraph of a music recommender



A hypothetical servegraph of a music recommender



A hypothetical servegraph of a music recommender



Build a serve graph: Using Python Macros to define construction functions

```
rec = Recommender(... )

@rec.servegraph.usergraph(ins=['input_1', 'input_2'],
                          outs=['output_1', 'output_2'])
f(subgraph):
    ... = ... subgraph['input_1']
    subgraph['output_1'] = ...
```

openrec.recommenders.Recommender

Specify names of input ports

Specify names of output ports

Function that constructs the usergraph of servegraph

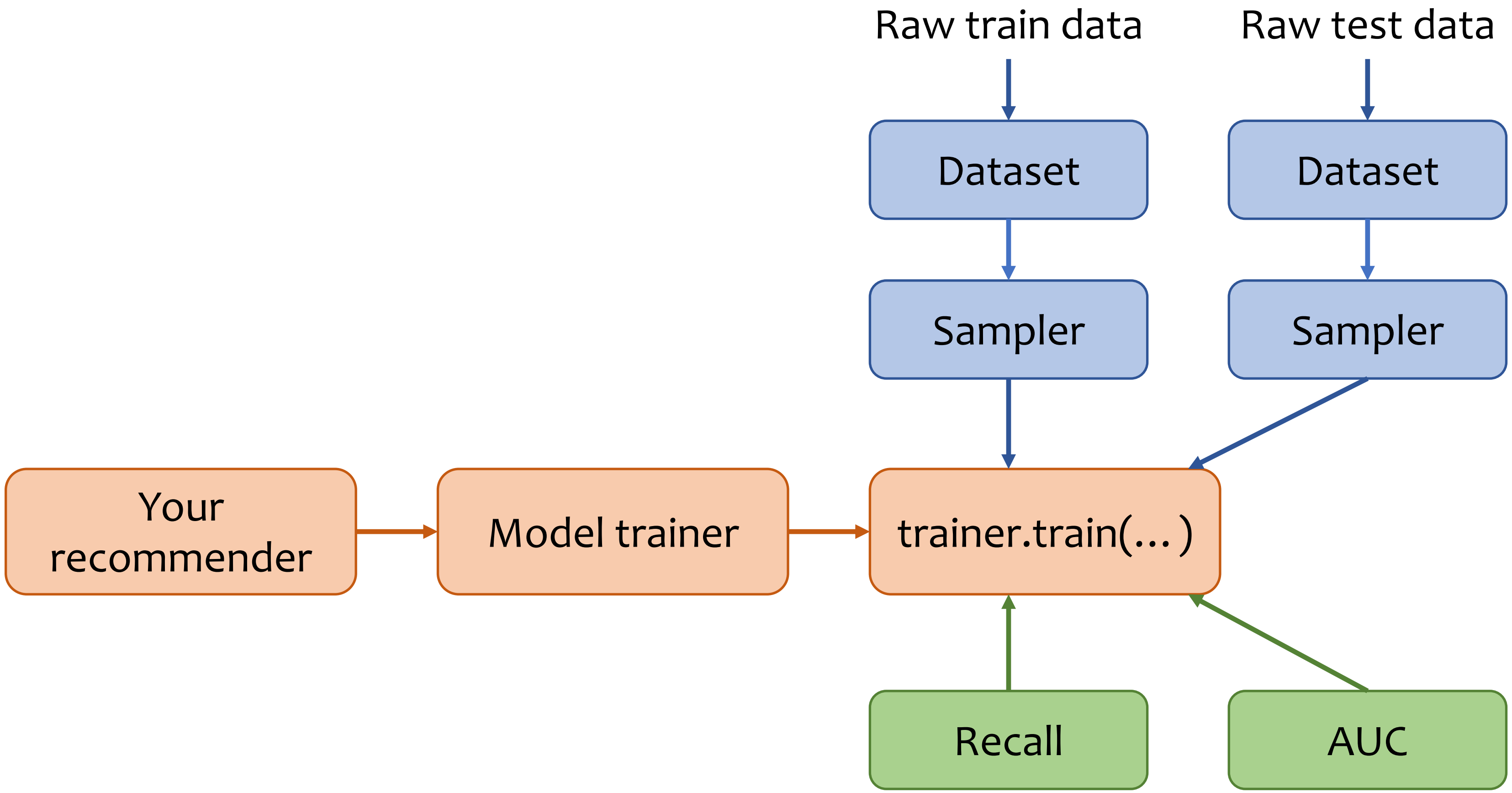
Fetch input

Set output

Reuse a subgraph

rec.servegraph.itemgraph = other_rec.servegraph.itemgraph

After a recommender is built, how should we train and evaluate it?



OpenRec

http://www.openrec.ai

OpenRec for researchers:

- Demonstrate model generalizability.
- Facilitate comparisons.
- Encourage usage.

OpenRec for practitioners:

- Select models/parameters.
- Adapt state-of-the-art solutions.

Share the same programming model and low-level APIs with Tensorflow/Keras.

OpenRec

Join our google group: openrec-lib@googlegroups.com

Questions?

Coming Next

Hands-on sessions (15 min each).

- OpenRec basics + diversity and fairness (Eugene)
- Customizing Deep YouTube Video Recommendation (Hongyi)
- Temporal-aware recommendation (Longqi)

OpenRec

http://www.openrec.ai

Github link, documents, and tutorials

Join our google group
openrec-lib@googlegroups.com



Contact: ylongqi@cs.cornell.edu

Twitter: @ylongqi

Connected Experiences Lab

http://cx.jacobs.cornell.edu/

Small Data Lab

http://smalldata.io/

CORNELL TECH

Cornell CIS Computer Science

Funders: NSF Oath: A Verizon company