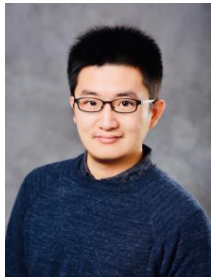
IJCAI/2023 MACAO

Joint Modeling in Recommendations: Fundamentals and Advances


Xiangyu Zhao[1]


Yichao Wang[2]


Bo Chen[2]


Pengyue Jia[1]


Yuhao Wang[1]


Jingtong Gao[1]


Huifeng Guo[2]


Ruiming Tang[2]

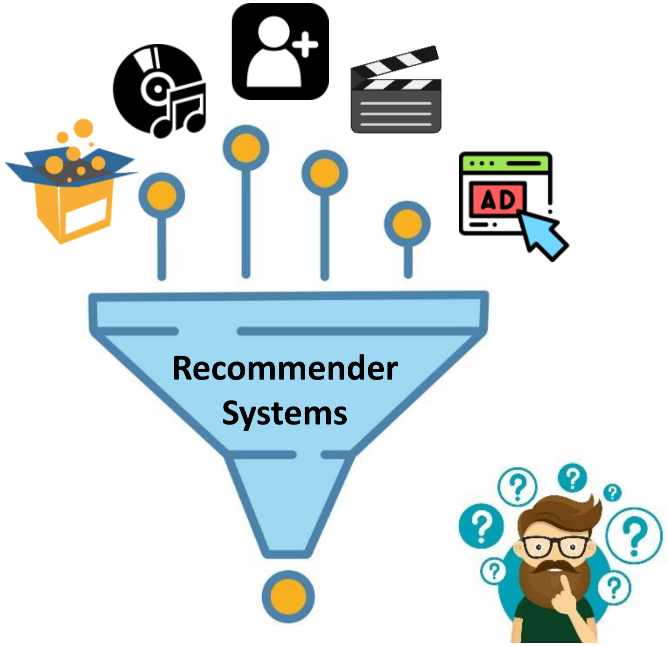
Xiangyu Zhao, City University of Hong Kong **Huawei Noah's Ark Lab**

[1]City University of Hong Kong, [2]Huawei Noah's Ark Lab

Recommender Systems

Age of Information Explosion

Information overload



Recommend item X to user

Items can be Products, News, Movies, Videos, Friends, etc.

Recommender Systems

➢ Recommendation has been widely applied in online services

- **E-commerce**, Content Sharing, Social Networking, etc.

Product Recommendation

Frequently bought together

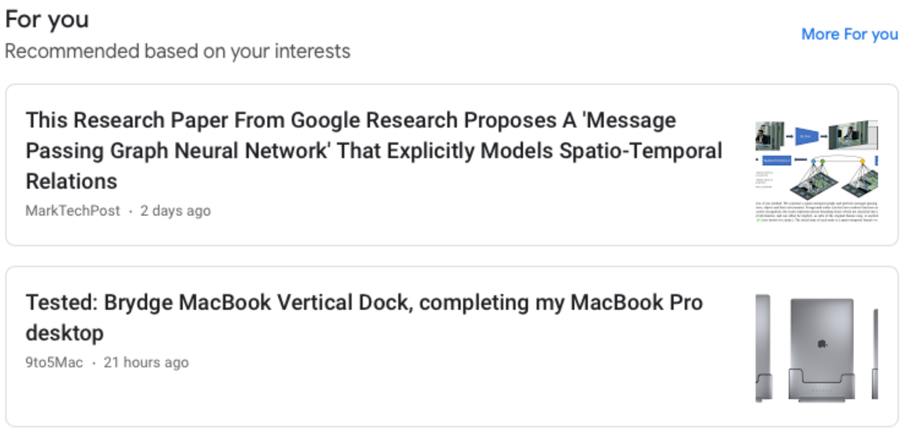
A B C

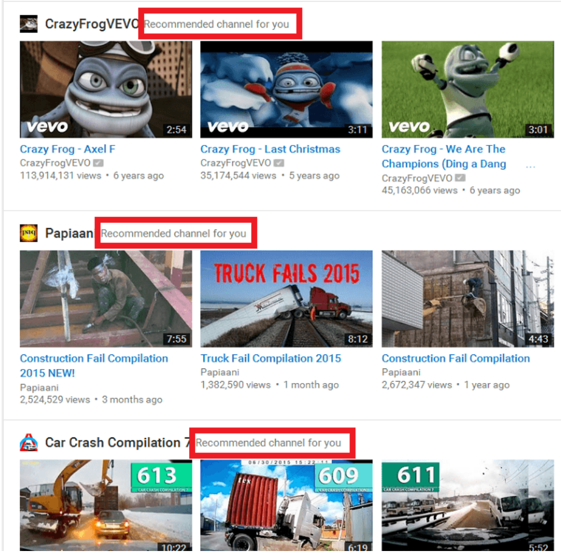
Recommender Systems

➢ Recommendation has been widely applied in online services

- **E-commerce, Content Sharing, Social Networking, etc.**

News/Video/Image Recommendation

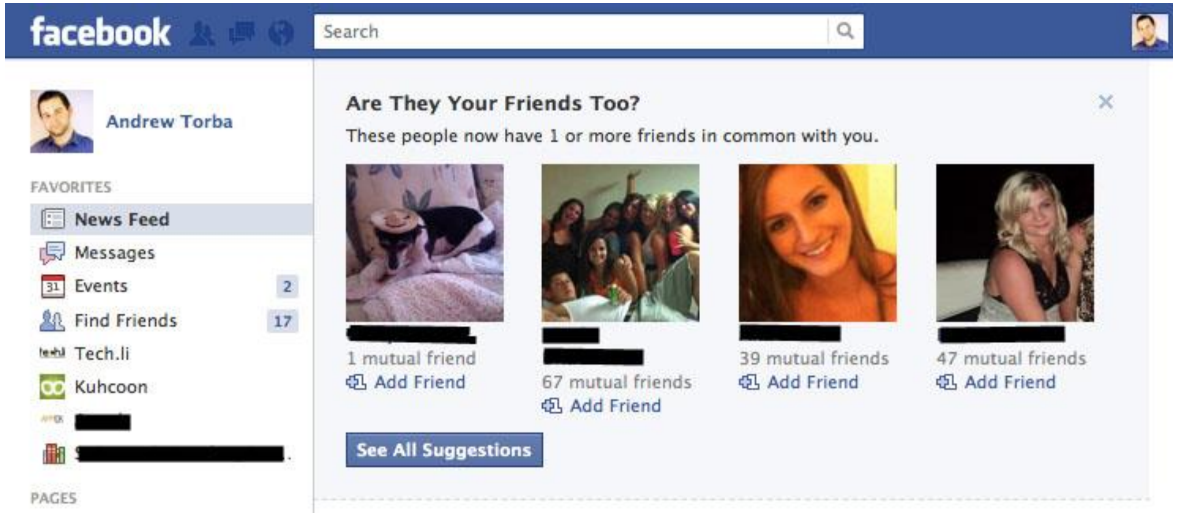




Recommender Systems

- Recommendation has been widely applied in online services
  - **E-commerce, Content Sharing, Social Networking, etc.**

Friend Recommendation



Deep Recommender Architecture

- Advantages
  - **Feature representations of users and items**
  - **Non-linear relationships between users and items**

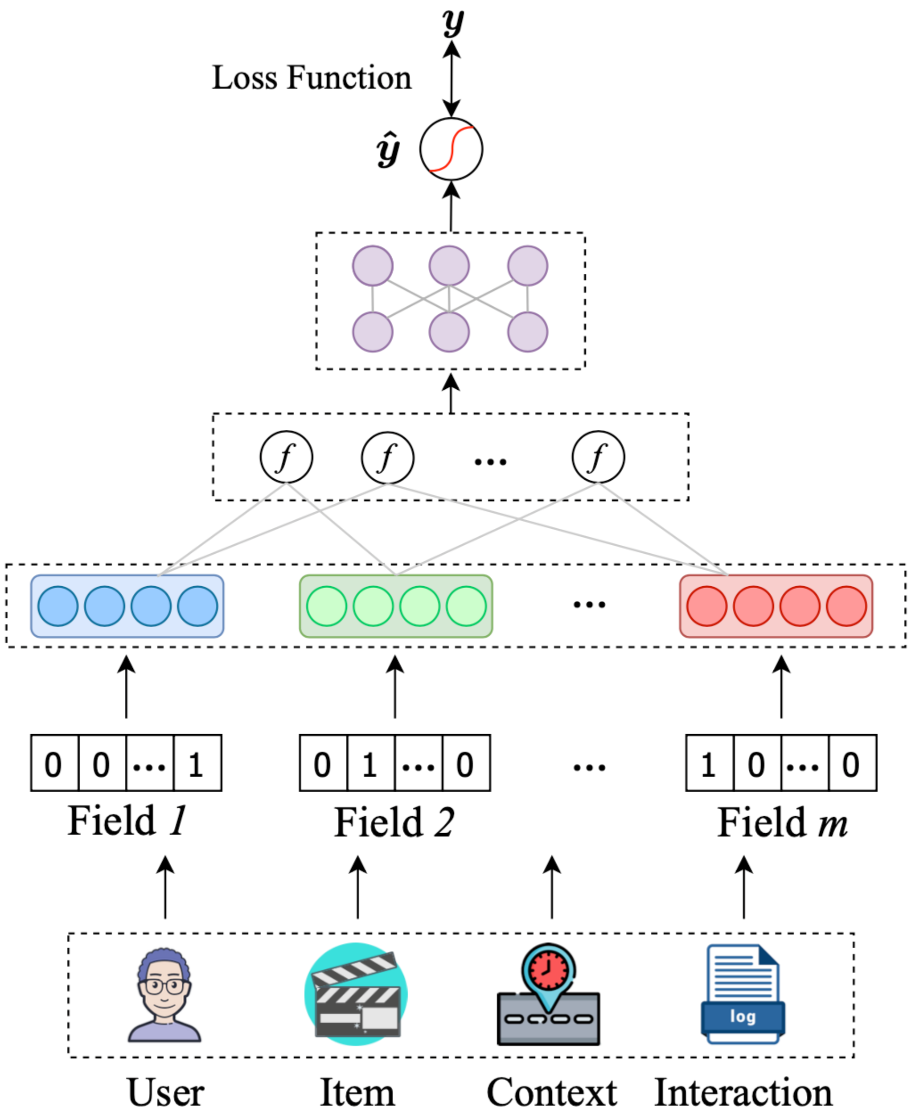
Why Joint Modeling?

Feature Embedding Layer

High/low-frequency features embedding sizes

Input Layer

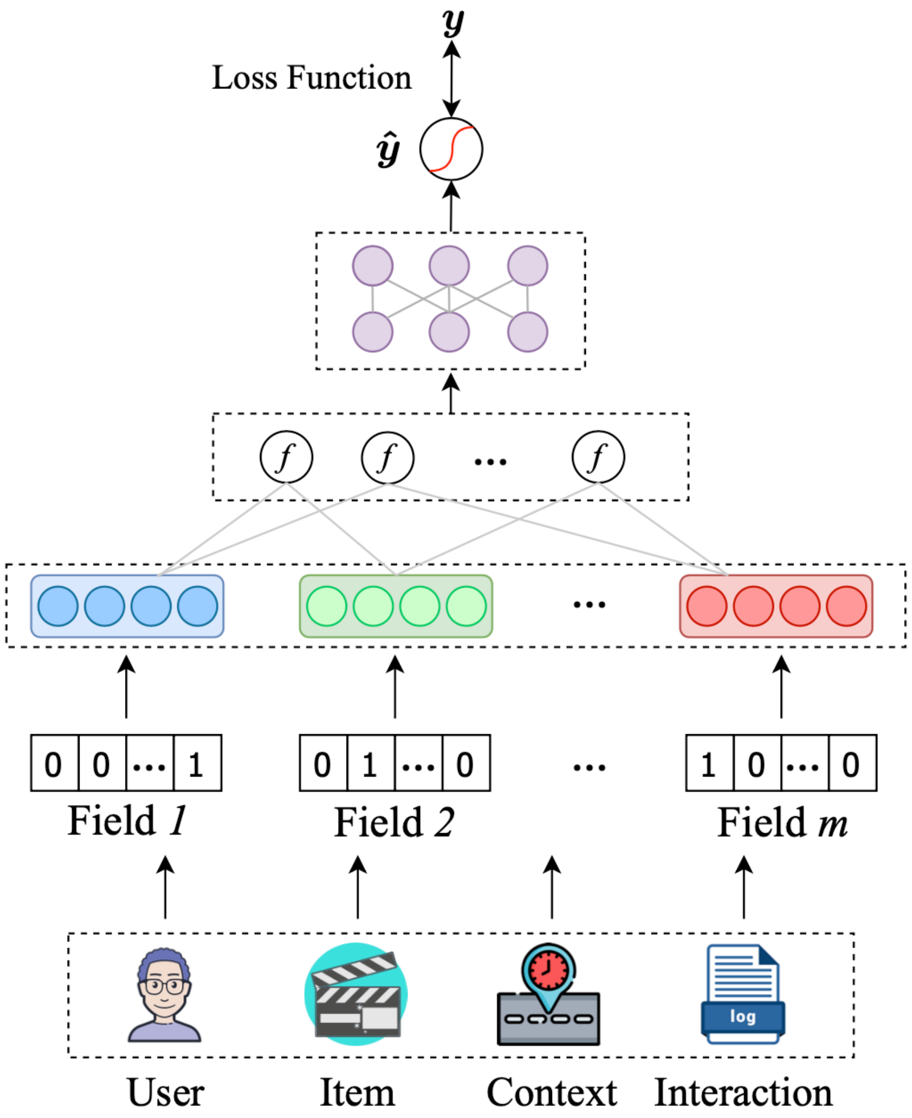
Feature selection



Why Joint Modeling?

Output Layer
BCE, BPR, MSE

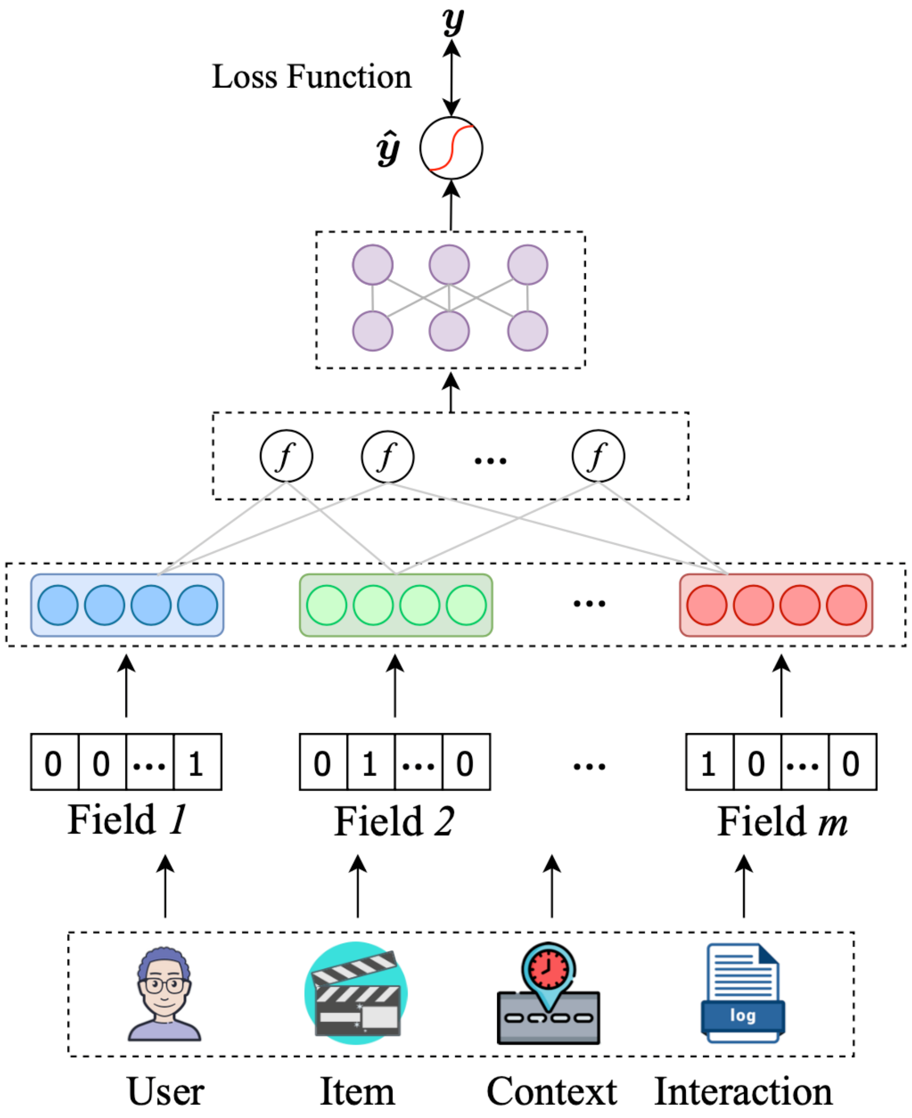
Feature Interaction Layer
Pooling, convolution, and the number of layers, inner product, outer product, convolution, etc.



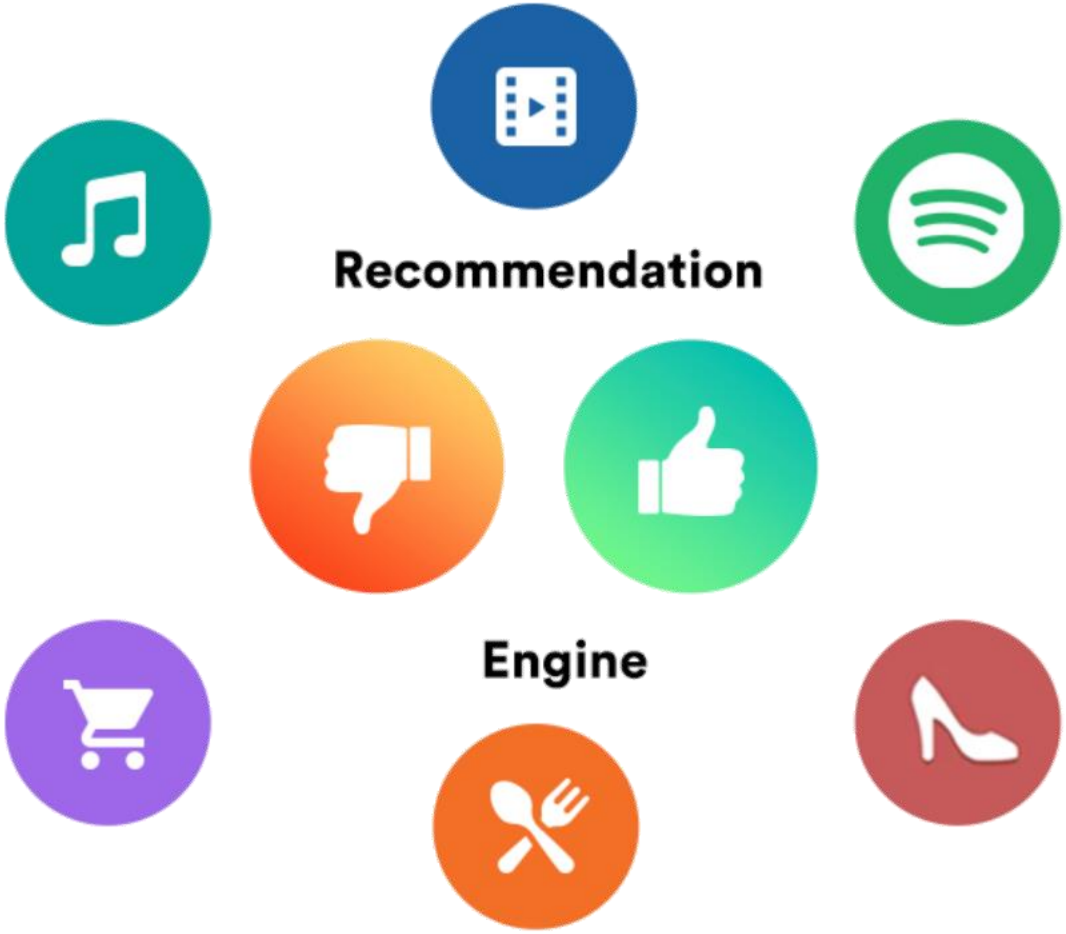
System Design

Hardware infrastructure, data pipeline, information transfer, implementation, deployment, optimization, evaluation, etc.

Why Joint Modeling?

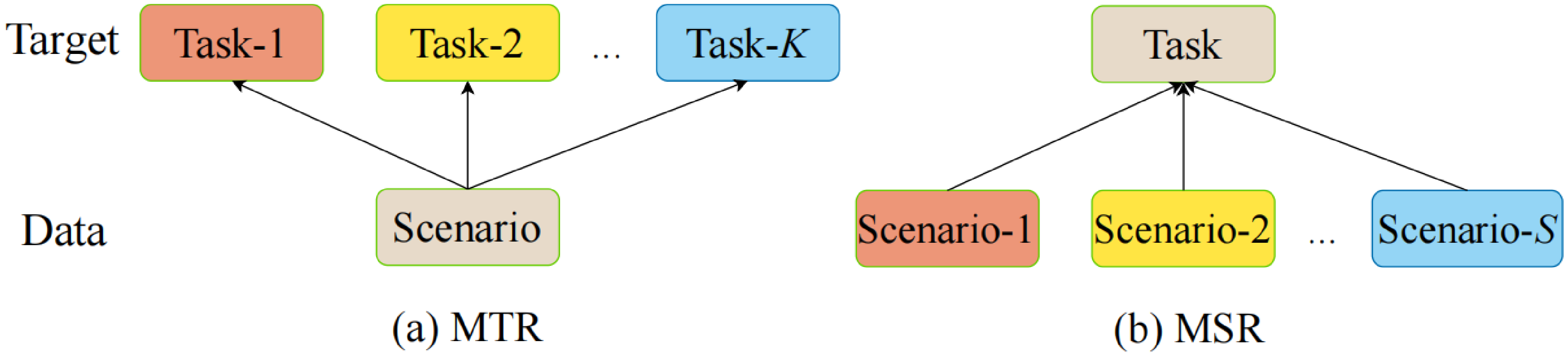


V.S.



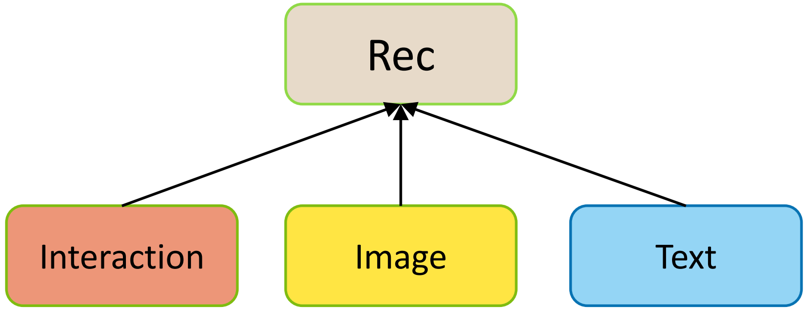
Joint Modeling in Recommendations

- Handling the inter-dependency between users and items under more complex circumstances
- Advantages
  - **One model for several situations**
  - **Performance improvement caused by information sharing in different situations**
- Two typical representatives:
  - **Multi-task recommendation (MTR)**
  - **Multi-scenario recommendation (MSR)**

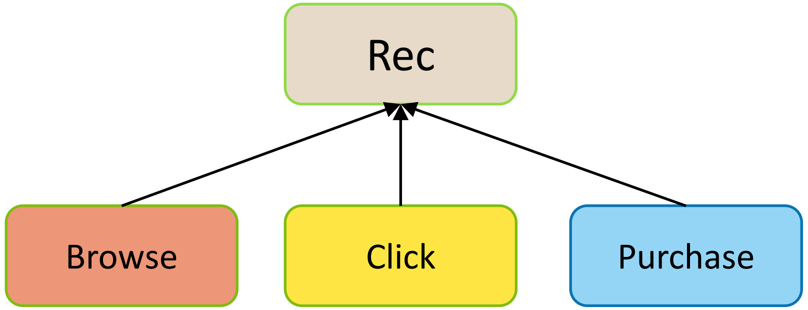


Joint Modeling in Recommendations

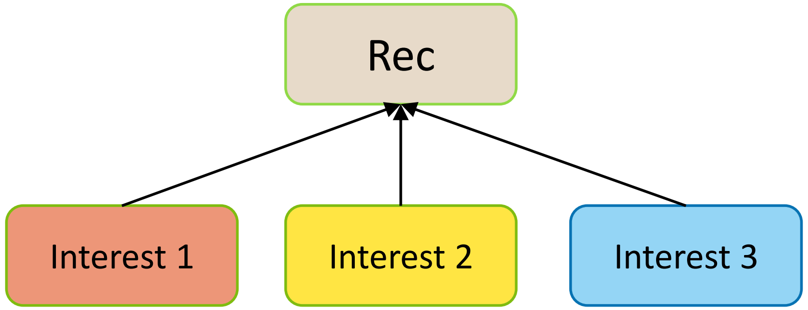
- More joint modeling methods:
  - **- Multi-modal recommendation**
  - **- Multi-interest recommendation**
  - **- Multi-behavior recommendation**
  - **- Large language model-based recommendation**



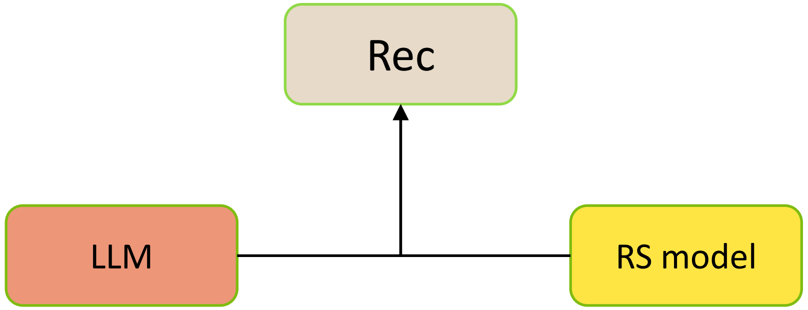
Multi-modal recommendation



Multi-behavior recommendation



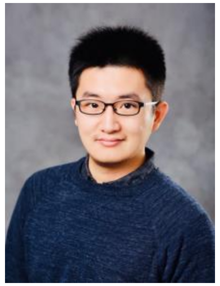
Multi-interest recommendation



Large language model-based recommendation

Agenda

Introduction



Xiangyu Zhao

Preliminary

Yichao Wang

Multi-task Recommendation

Yuhao Wang

Multi-scenario recommendation

MTR+MSR

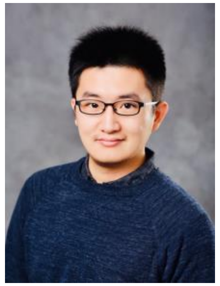
Pengyue Jia

More Joint-learning Methods

Jingtong Gao

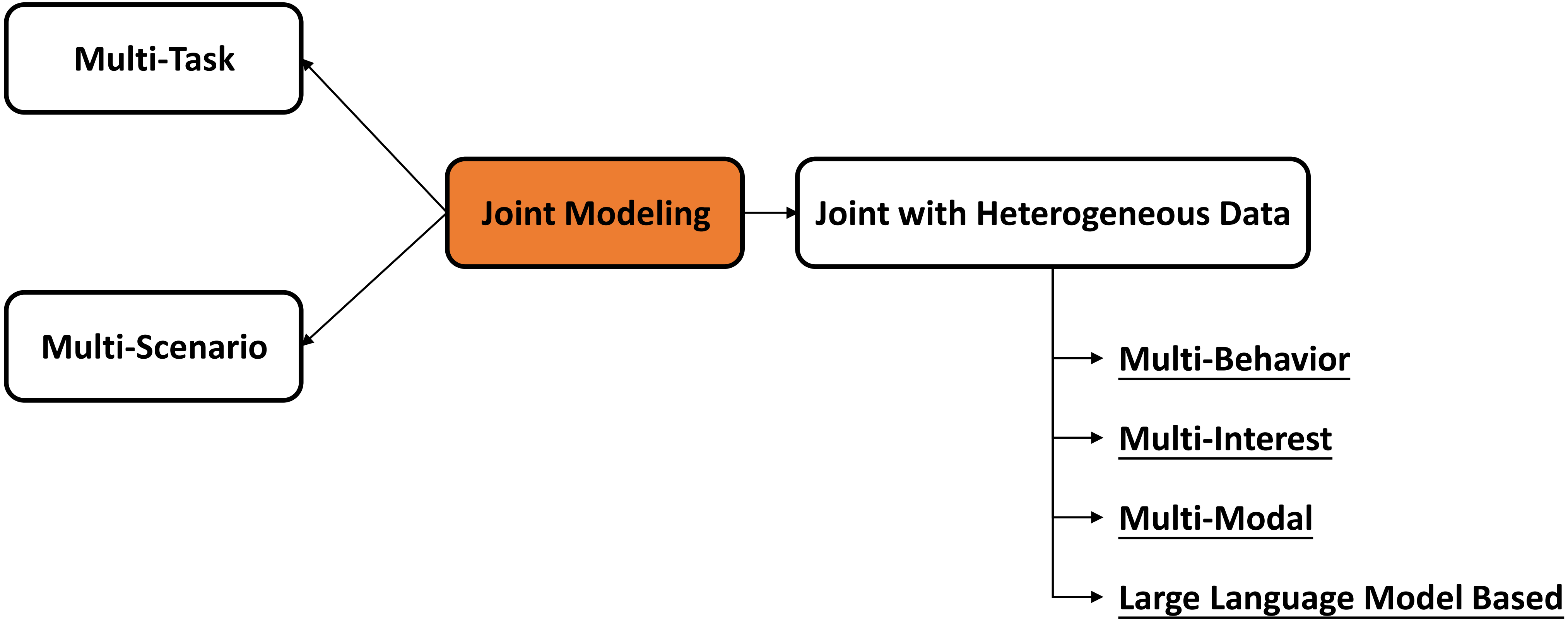
Conclusion

Future Work



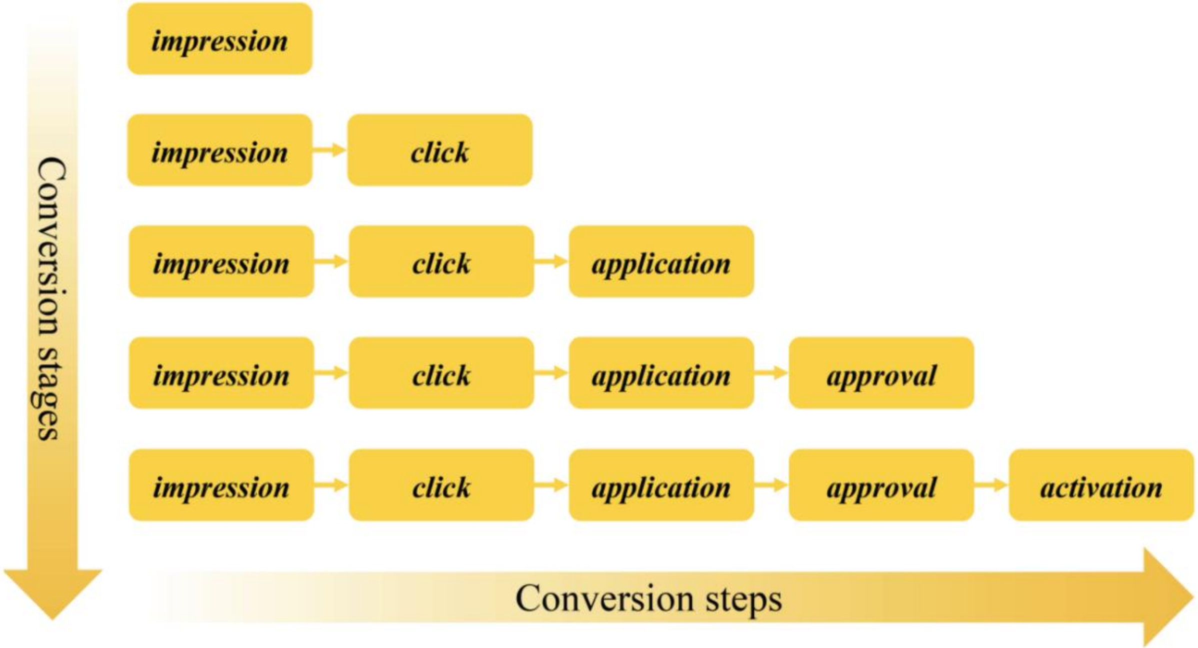
Xiangyu Zhao

Why Joint Modeling ?



Why Joint Modeling ?

➢ Multi-Task Recommendation:
- **Independent tasks: Comments, repost, likes, bookmarks**
- **Multi-stage conversion tasks: click, application, approval, activation …**



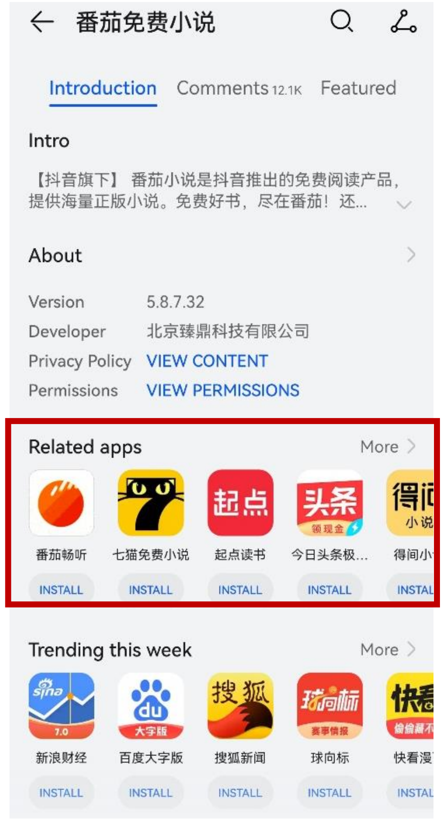
How to extract useful information from other tasks ?

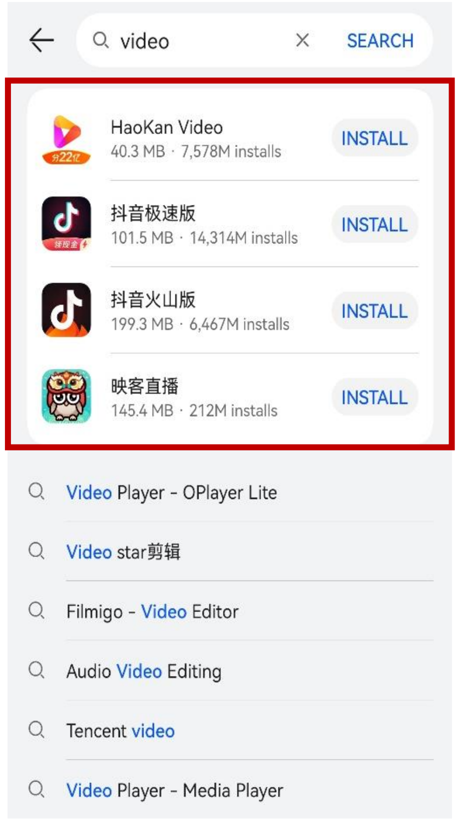
How to capture task dependences and resolve the sparsity issue ?

Modeling the sequential dependence among audience multi-step conversions with multi-task learning in targeted display advertising.KDD 2021.

Why Joint Modeling ?

➢Multi-Scenario Recommendation: construct multiple scenarios for user diverse requirements.

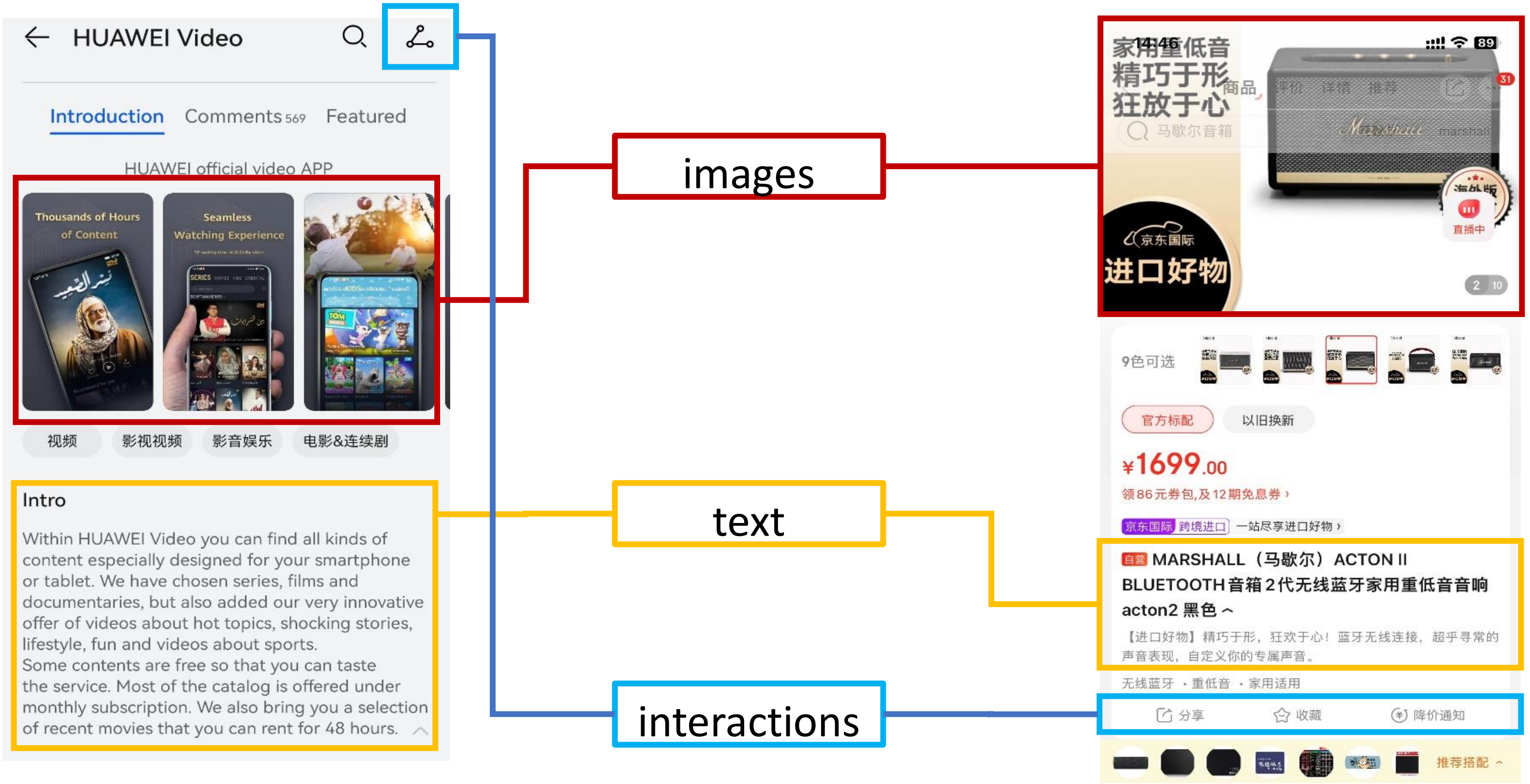




How to extract more comprehensive user portrait from interactions in different scenarios, and make recommendations based on the characteristics of the current scenario ?

Why Joint Modeling ?

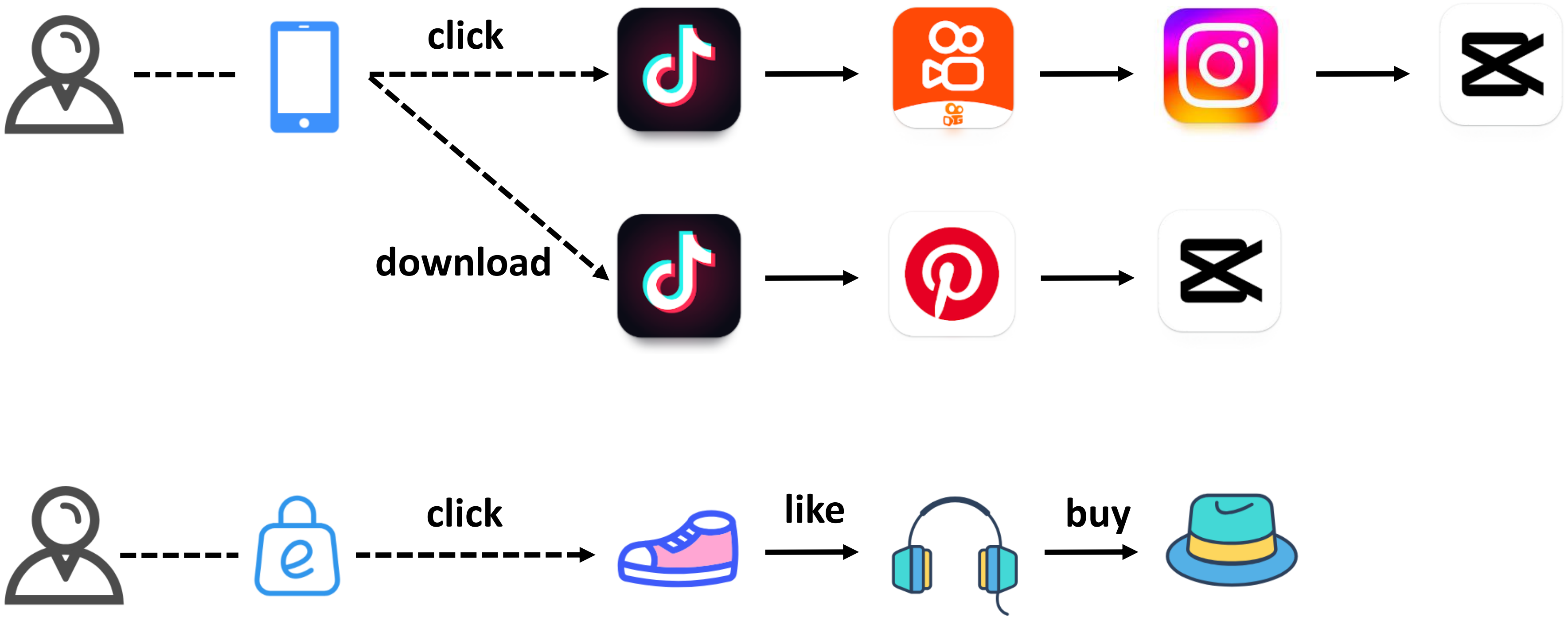
- Multi-Modal Modeling: user interactions, images, text …



How to extract and align data from different modalities ?

Why Joint Modeling ?

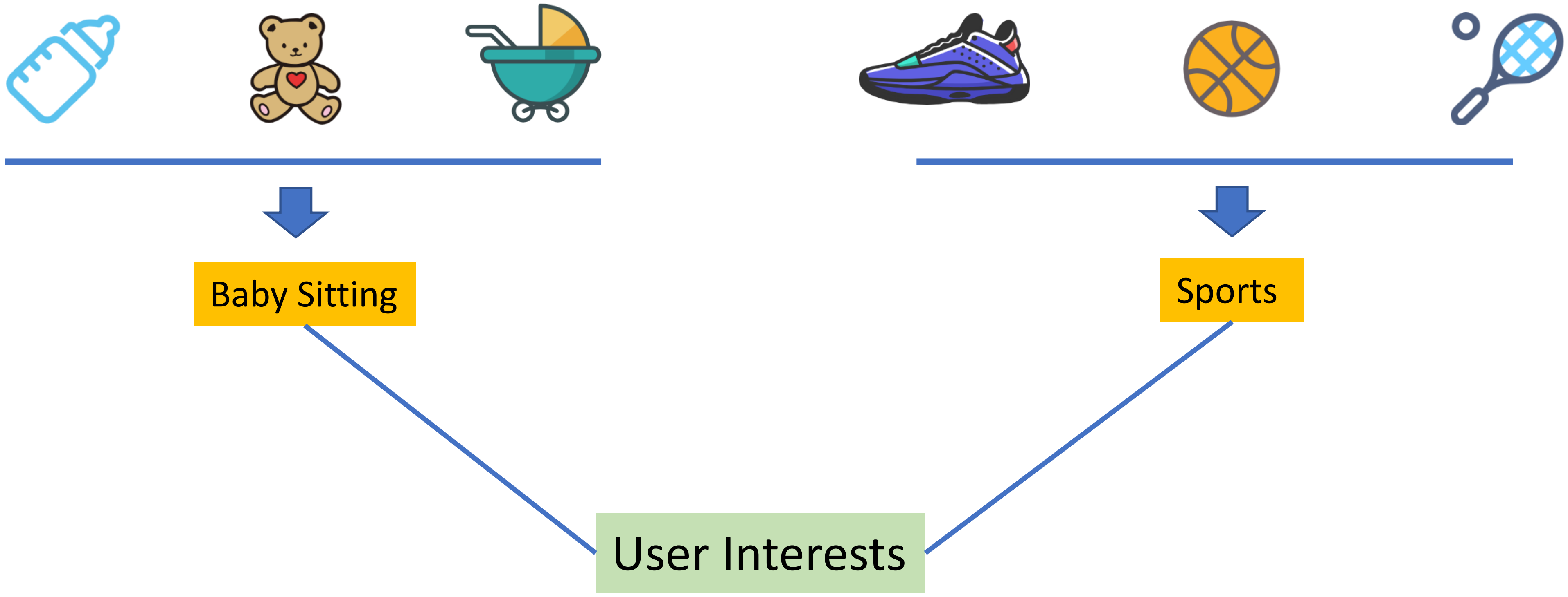
➢Multi-Behavior Modeling: click, download, like, buy



How to learn the relationship between different type of behaviors ?

Why Joint Modeling ?

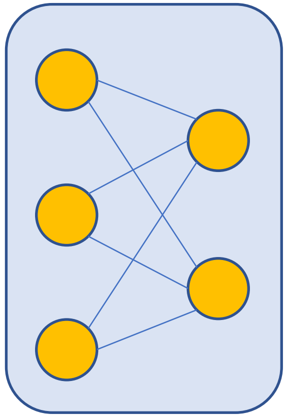
- ➢Multi-Interest Modeling: behaviors → interests



How to accurately and efficiently extract users' diverse interests from user behaviors ?

Why Joint Modeling ?

➢Large Language Model-based Recommendation



DRS

Trained on labeled data with supervised learning

Collaborative signals

ID-based in-domain collaborative knowledge

LLM

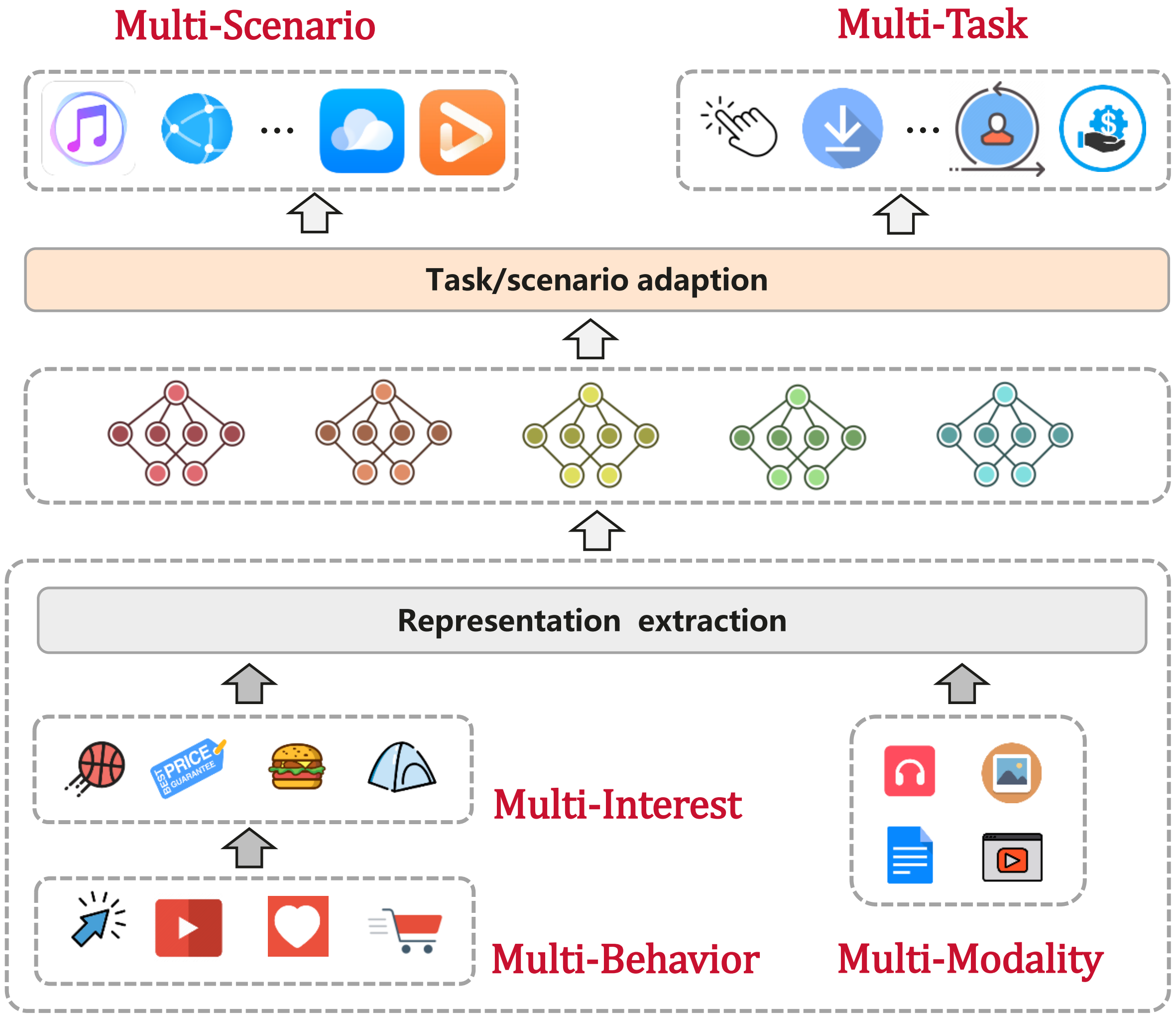
Pre-trained on large-scale corpora with self-supervised learning

Semantic signals

Generalization, reasoning and open-world knowledge

?

Relations and Formulations of Joint Modeling



$$wL(\boxed{E^{Merge}}, \Theta, \Theta^t, \Theta^s)$$

Joint Modeling

$$E^{Merge} = U(E, \boxed{E^{Ext}}, E^m)$$

$$E^{Ext} = F(H^{UB})$$

Multi-Interest

$$H^{UB} = G(H_1, H_2, \ldots, H_N)$$

Multi-Behavior

$$E^m = M(E^{txt}, E^v, \ldots, E^p)$$

Multi-Modality

$$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$$

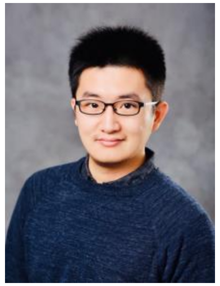
Multi-Scenario

$$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$$

Multi-Task

Agenda

Introduction



Xiangyu Zhao

Preliminary

Yichao Wang

Multi-task Recommendation

Yuhao Wang

Multi-scenario recommendation

MTR+MSR

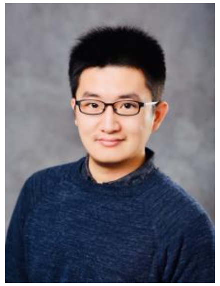
Pengyue Jia

More Joint-learning Methods

Jingtong Gao

Conclusion

Future Work



Xiangyu Zhao

MTR & MTDRS

Multi-Task Recommendation (MTR)

Multi-Task Deep Recommender Systems (MTDRS)

➢ How

- Multi-Task Learning (MTL) + Deep Neural Networks

➢ Why

- Learning high-order feature interactions and
- Modeling complex user-item interaction behaviors

Benefits & Challenges

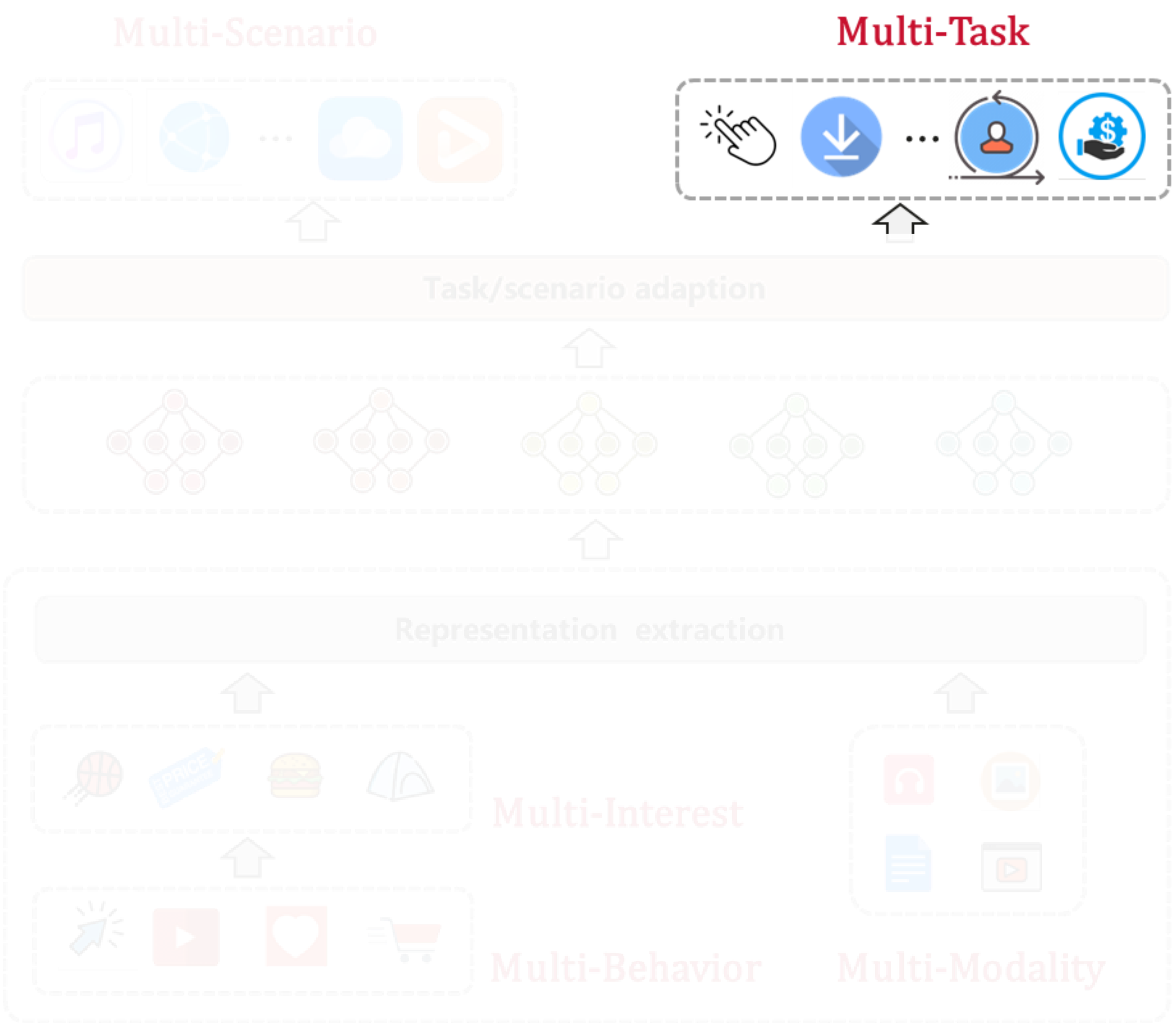
➢ Benefits

- Mutual enhancement among tasks
- Higher efficiency of computation and storage

➢ Challenges

- Effectively and efficiently capture useful information & relevance among tasks
- Data sparsity
- Unique sequential dependency

Multi-task Recommendation



$wL(E^{Merge}, \Theta, \Theta^t, \Theta^S)$ Joint Modeling

$E^{Merge} = U(E, E^{Ext}, E^m)$

$E^{Ext} = F(H^{UB})$ Multi-Interest

$H^{UB} = G(H_1, H_2, \dots, H_N)$ Multi-Behavior

$E^m = M(E^{txt}, E^v, \dots, E^p)$ Multi-Modality

$wL(E^{Merge}, \Theta, \Theta^t, \Theta^S)$ Multi-Scenario

$wL(E^{Merge}, \Theta, \Theta^t, \Theta^S)$ Multi-Task

Formulation

➢ **Problem:**

- Learning MTL model with task-specific parameters $(\theta^1, \ldots, \theta^K)$ and shared parameter θ^s, which outputs the ***K*** task-wise predictions

➢ **Optimization problem:**

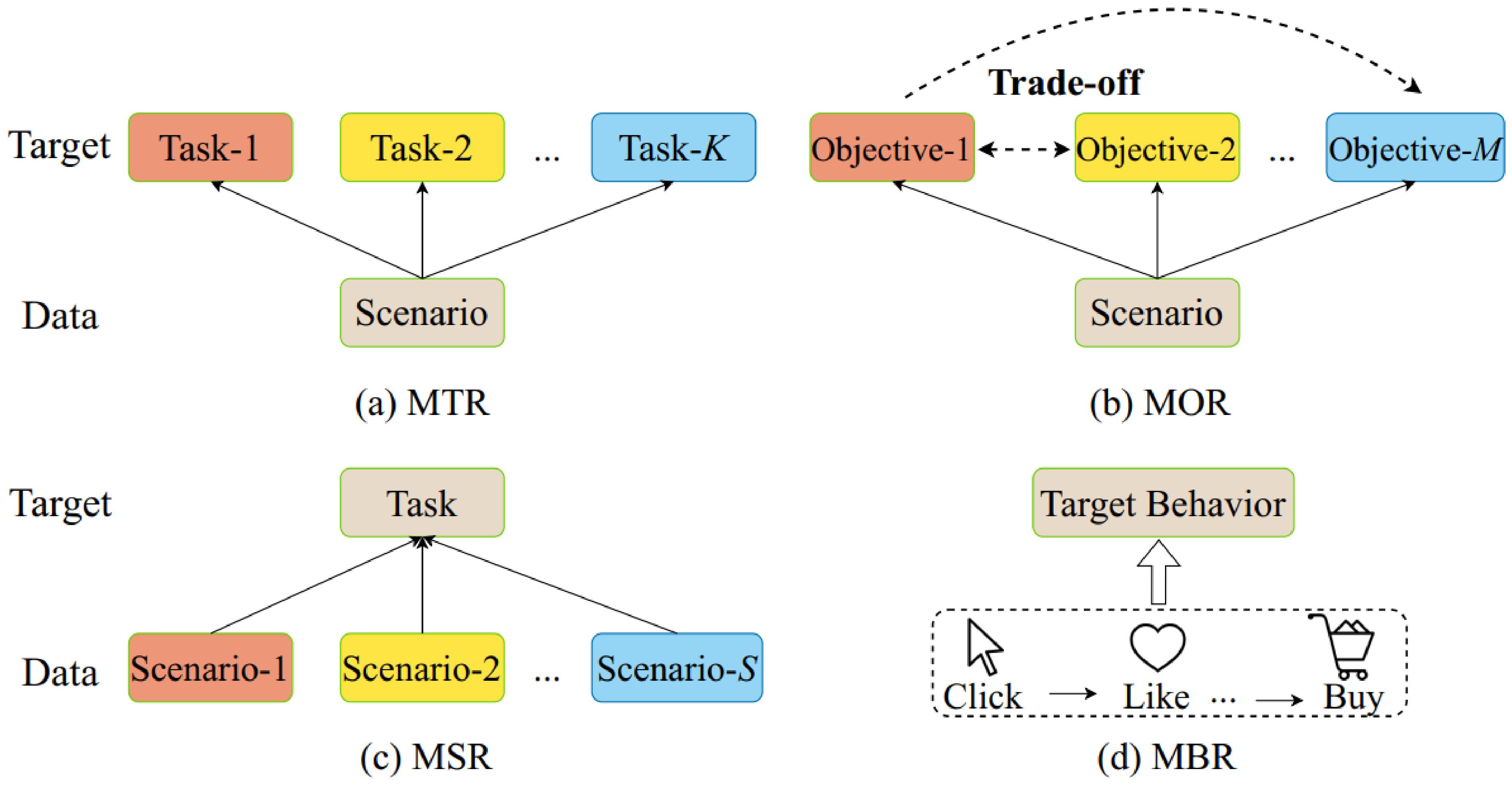
$$\underset{\{\theta^1,\ldots,\theta^K\}}{\arg\min}\ \mathcal{L}\left(\theta^s, \theta^1, \cdots, \theta^K\right) = \underset{\{\theta^1,\ldots,\theta^K\}}{\arg\min} \sum_{k=1}^{K} \omega^k L^k\left(\theta^s, \theta^k\right)$$

- $\mathcal{L}(\theta^s, \theta^k)$: loss function for k-th task with parameter θ^s, θ^k
- ω^k: loss weight for k-th task

BCE loss

$$L^k\left(\theta^s, \theta^k\right) = -\sum_{n=1}^{N}\left[y_n^k \log\left(\hat{y}_n^k\right) + \left(1 - y_n^k\right)\log\left(1 - \hat{y}_n^k\right)\right]$$

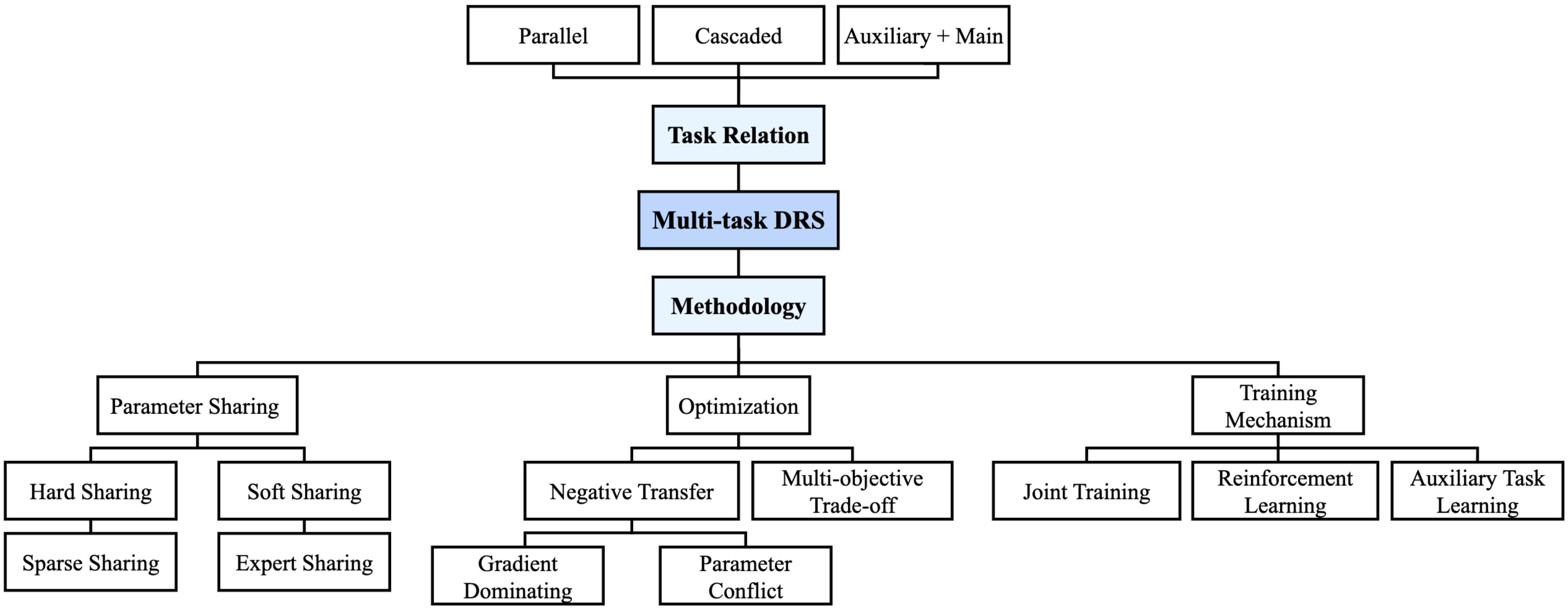
MTR, MOR, MSR, MBR



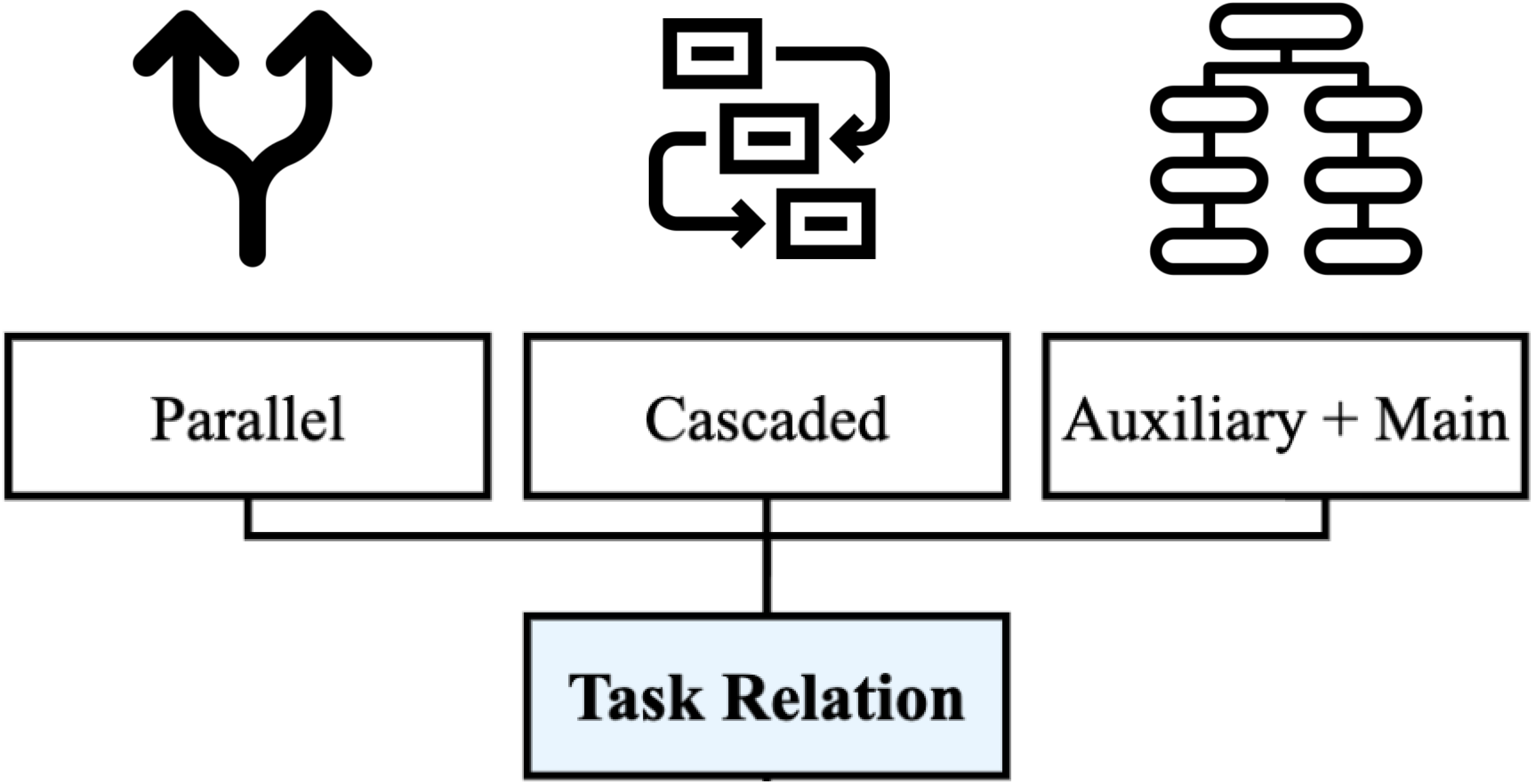
Comparison with CV & NLP

Task	Description	Explanation
CV	Multi-target segmentation and further classification for each object	Utilizing **feature transformation** to represent common features based on a multi-layer feed-forward network
NLP	Mostly focus on the design of MTL architectures	Based on RNN because of the sequence pattern Can be divided into word-, sentence-, and document-level by granularity

Taxonomy


Parallel
Cascaded
Auxiliary + Main
Task Relation
Multi-task DRS
Methodology
Parameter Sharing
Optimization
Training Mechanism
Hard Sharing
Soft Sharing
Negative Transfer
Multi-objective Trade-off
Joint Training
Reinforcement Learning
Auxiliary Task Learning
Sparse Sharing
Expert Sharing
Gradient Dominating
Parameter Conflict

Task Relation



Parallel

- Tasks independently calculated **without sequential dependency**
- Objective function: Weighted sum with constant loss weights

Cascaded

- Cascaded task relationship: **sequential dependency**
- Computation of current task depends on **previous** ones
  - E.g. CTCVR = CTR × CVR
- General formulation:

$$\hat{y}_n^k\left(\theta^s, \theta^k\right) - \hat{y}_n^{k-1}\left(\theta^s, \theta^k\right) = P\left(\epsilon_k = 0, \epsilon_{k-1} = 1\right)$$

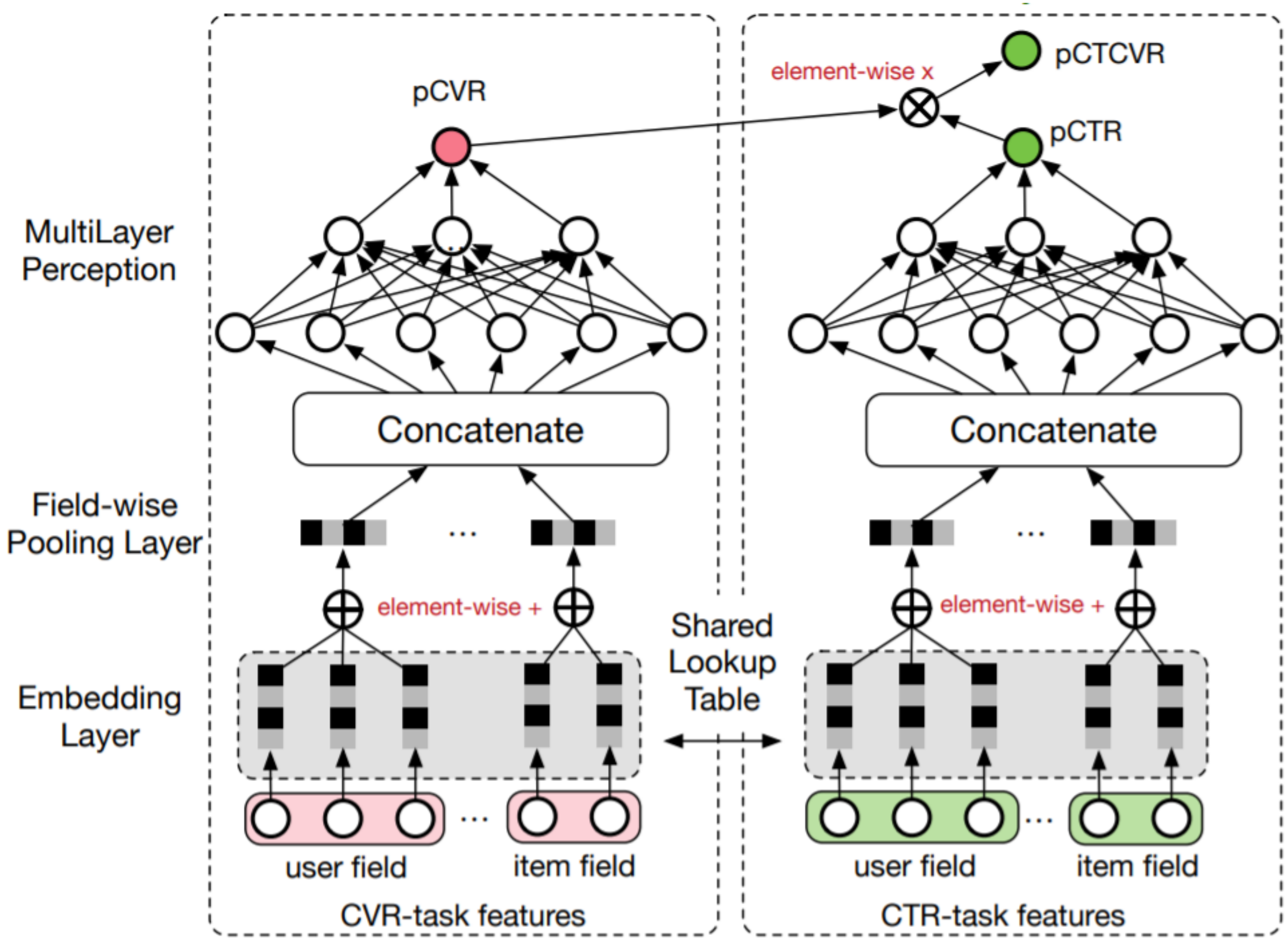
  - ϵ_k: Indicator variable for task k
  - Difference is the probability of the task k not happening while the task $k-1$ is observed

Cascaded

Model	Problem	Behavior Sequence
ESMM [Ma *et al.*, 2018b]	SSB & DS	impression → click → conversion
ESM^2 [Wen *et al.*, 2020]	SSB & DS	impression → click → D(O)Action → purchase
Multi-IPW & DR [Zhang *et al.*, 2020]	SSB & DS	exposure → click → conversion
ESDF [Wang *et al.*, 2020b]	SSB & DS & time delay	impression → click → pay
HM^3 [Wen *et al.*, 2021]	SSB & DS & micro and macro behavior modeling	impression → click → micro → macro → purchase
AITM [Xi *et al.*, 2021]	sequential dependence in multi-step conversions	impression → click → application → approval → activation
MLPR [Wu *et al.*, 2022]	sequential engagement & vocabulary mismatch in product ranking	impression → click → add-to-cart → purchase
ESCM^2 [Wang *et al.*, 2022a]	inherent estimation bias & potential independence priority	impression → click → conversion
HEROES [Jin *et al.*, 2022]	multi-scale behavior & unbiased learning-to-rank	observation → click → conversion
APEM [Tao *et al.*, 2023]	sample-wise representation learning in SDMTL	impression → click → authorize → conversion
DCMT [Zhu *et al.*, 2023]	SSB & DS & potential independence priority (PIP)	exposure → click → conversion

SSB: Sample Selection Bias   DS: Data Sparsity

ESMM



Entire Space Multi-task Model: An Effective Approach for Estimating Post-click Conversion Rate. SIGIR 2018.

Auxiliary with Main Task

- A task specified as the main task
  while associated auxiliary tasks help to improve performance

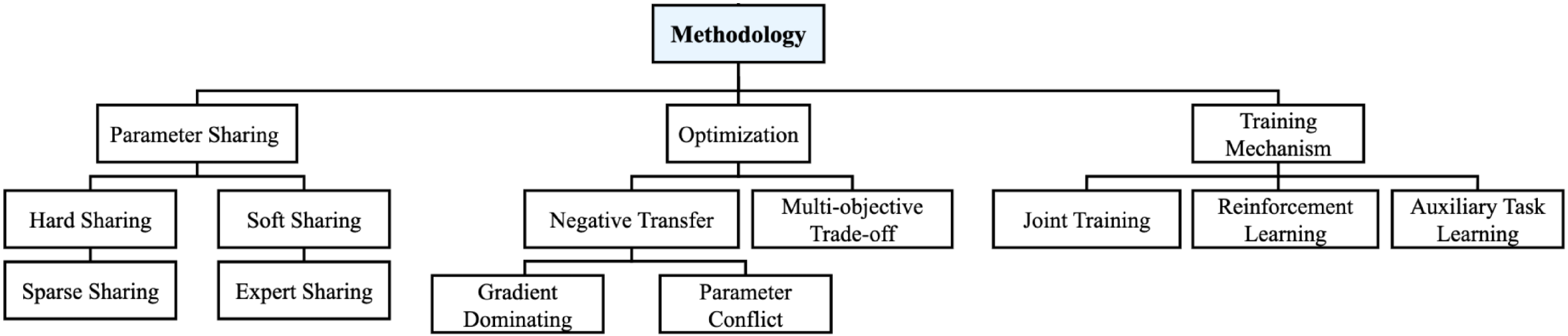
- Probability estimation for main task ⟸ the probability of auxiliary tasks

- Provide richer information across entire space

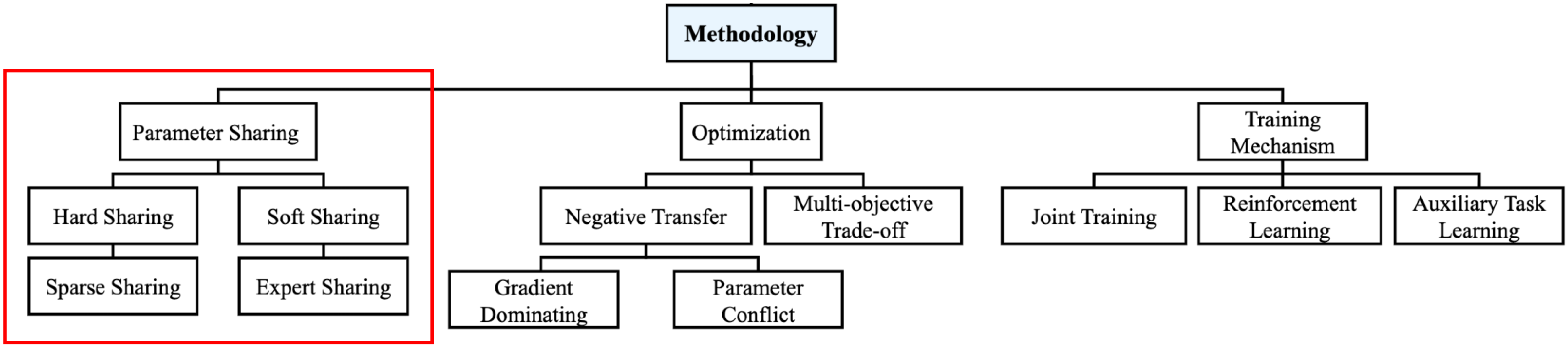
Auxiliary with Main Task

Model	References	Method
ESDF Multi-IPW and Multi-DR DMTL Metabalance	[Wang et al., 2020b] [Zhang et al., 2020] [Zhao et al., 2021] [He et al., 2022]	Adopt the original recommendation tasks as auxiliaries
MTRec PICO MTAE Cross-Distill	[Li et al., 2020a] [Lin et al., 2022] [Yang et al., 2021] [Yang et al., 2022a]	Manually design various auxiliary tasks
CSRec	[Bai et al., 2022]	Contrastive learning as the auxiliary
Self-auxiliary*	[Wang et al., 2022b]	Under-parameterized self-auxiliaries

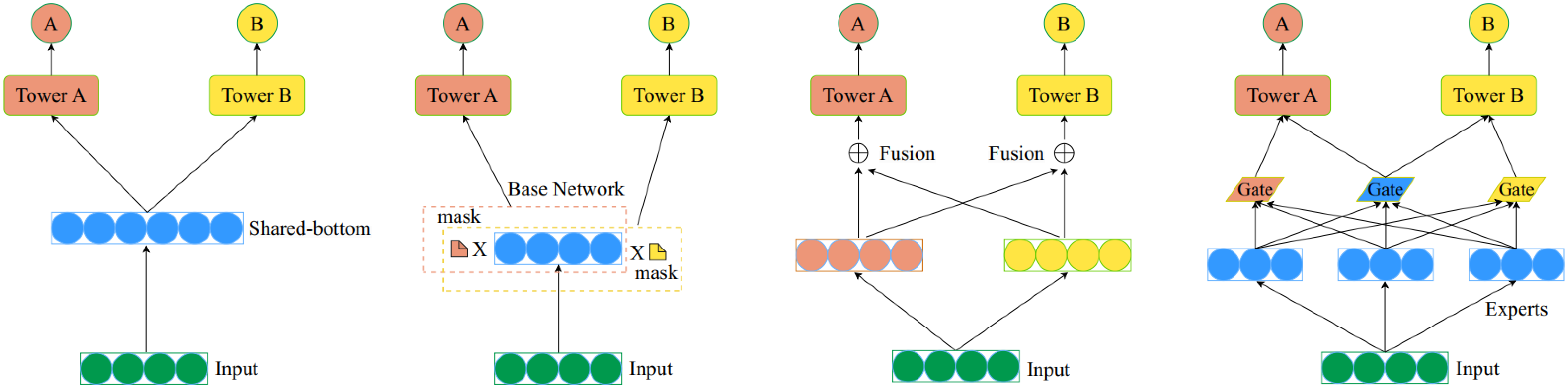
Methodology



Parameter Sharing

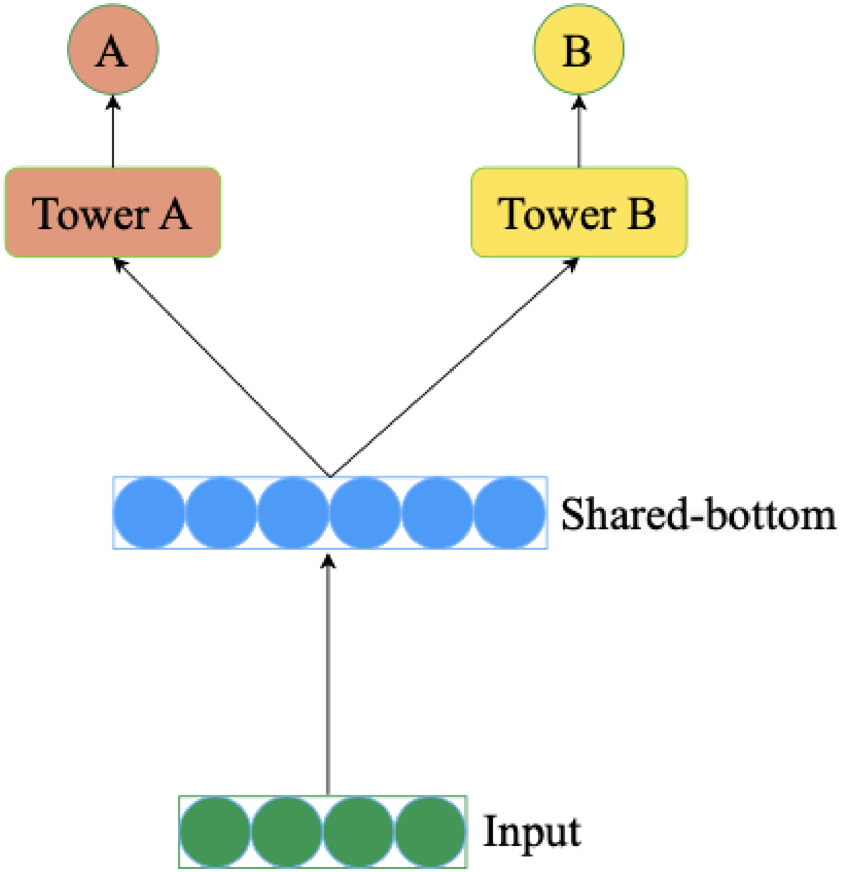


Parameter Sharing



(a) Hard Parameter Sharing (b) Sparse Sharing (c) Soft Parameter Sharing (d) Expert Sharing

Hard Sharing

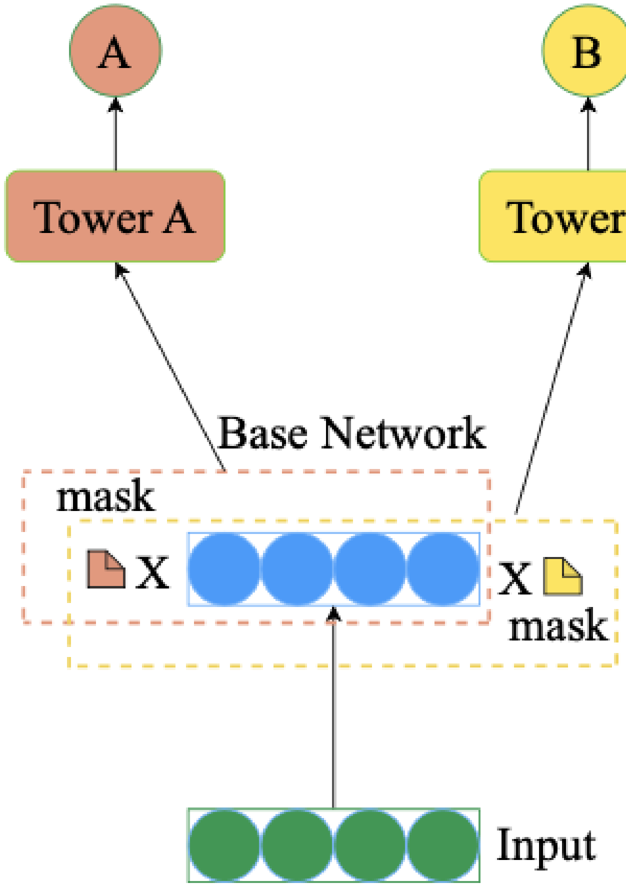


- Shared bottom layers extract the **same** information for different tasks,
- Task-specific top layers are trained individually

✓ Improving computation efficiency and alleviating over-fitting

✗ Limited capacity of the shared parameter space → **Weakly** related tasks and noise

Sparse Sharing

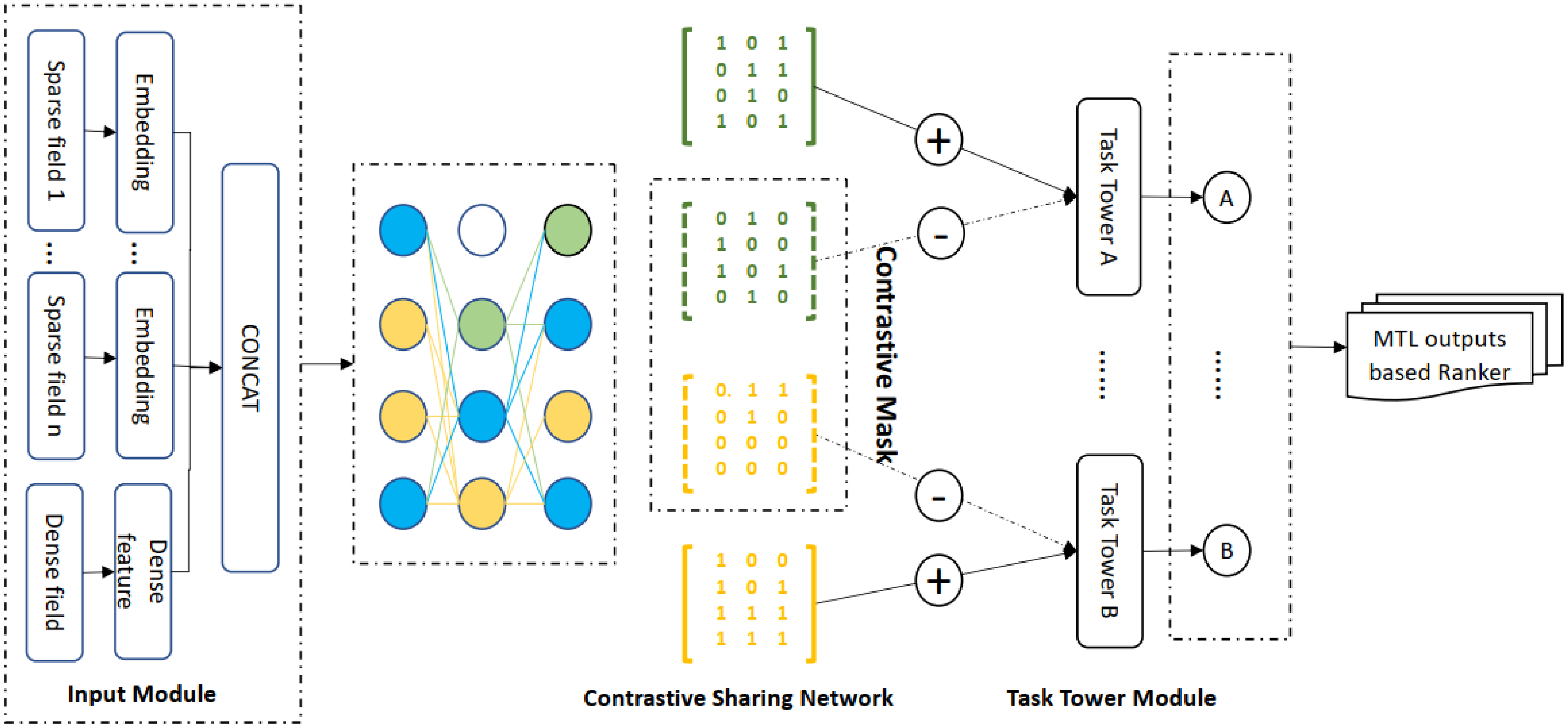


- ➢ Extracting **sub-networks** for each task by parameter masks from a base network
  - ○ **Special case of Hard Sharing**

✓ Coping with the weakly related tasks flexibly

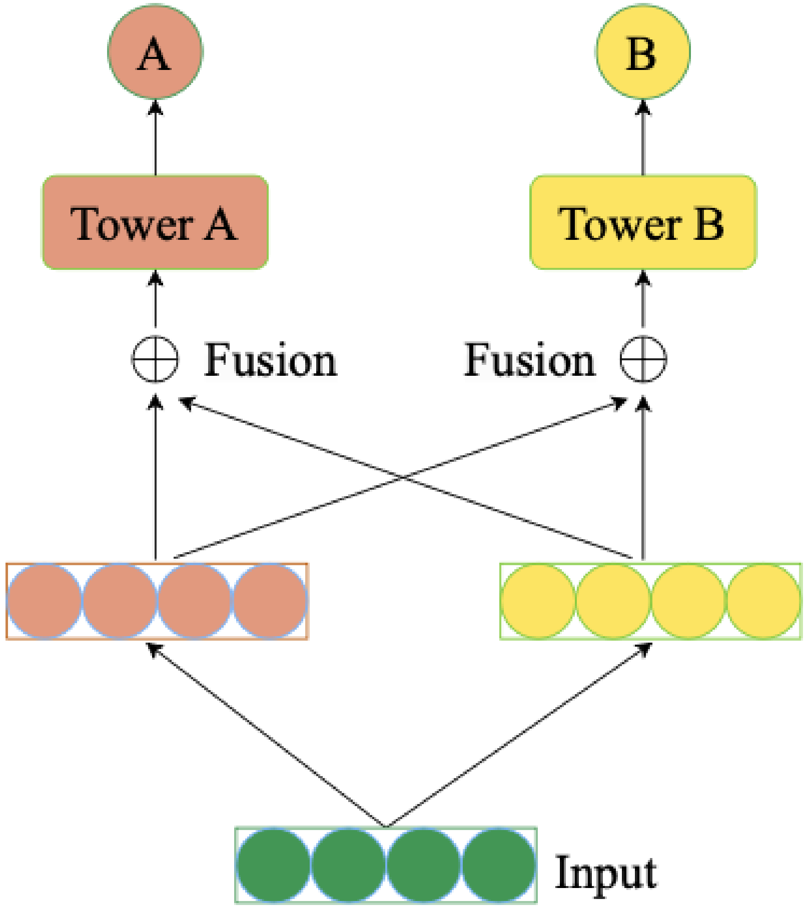
✗ Negative transfer
when updating shared parameters

CSRec



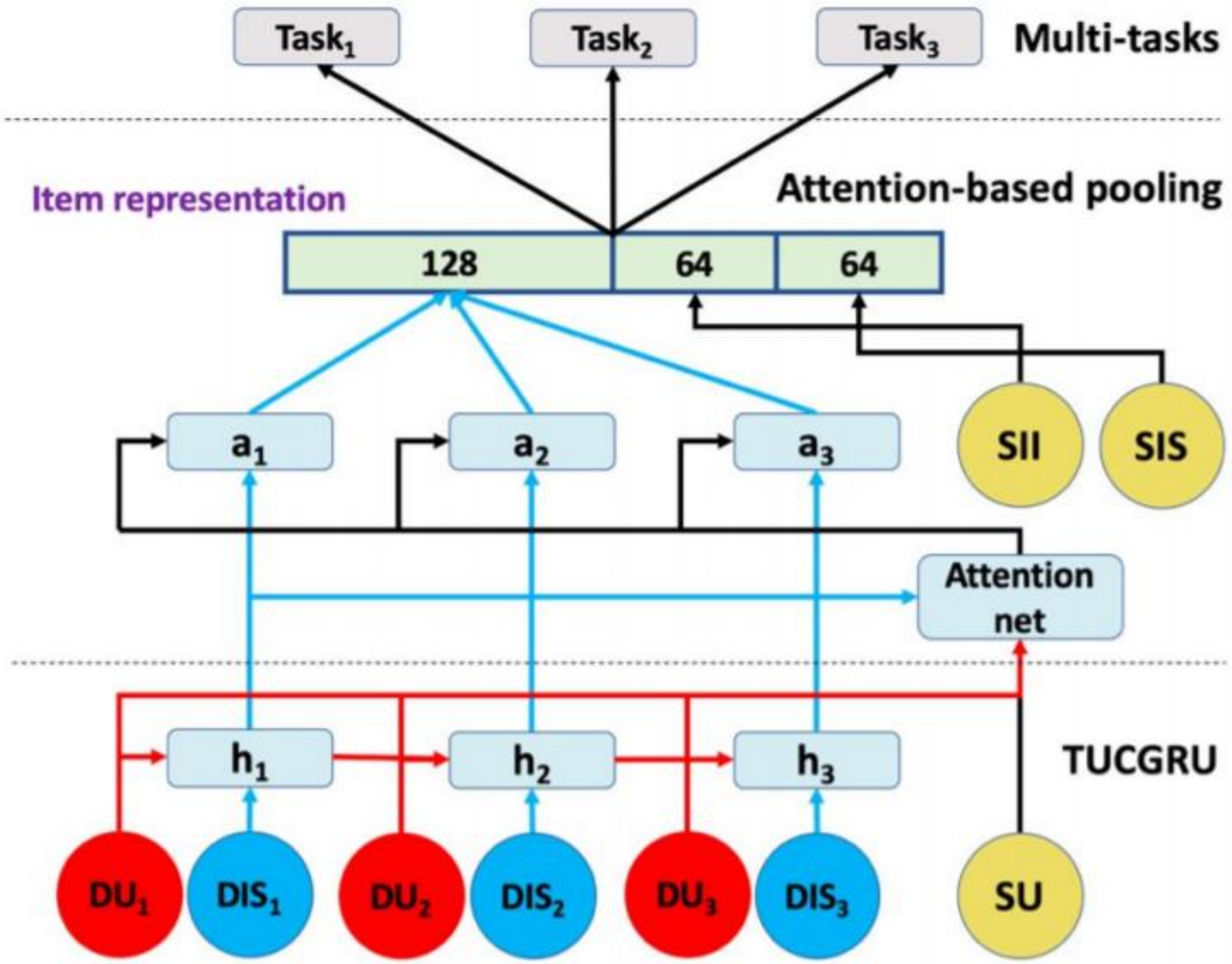
A Contrastive Sharing Model for Multi-task Recommendation. WWW 2022.

Soft Sharing



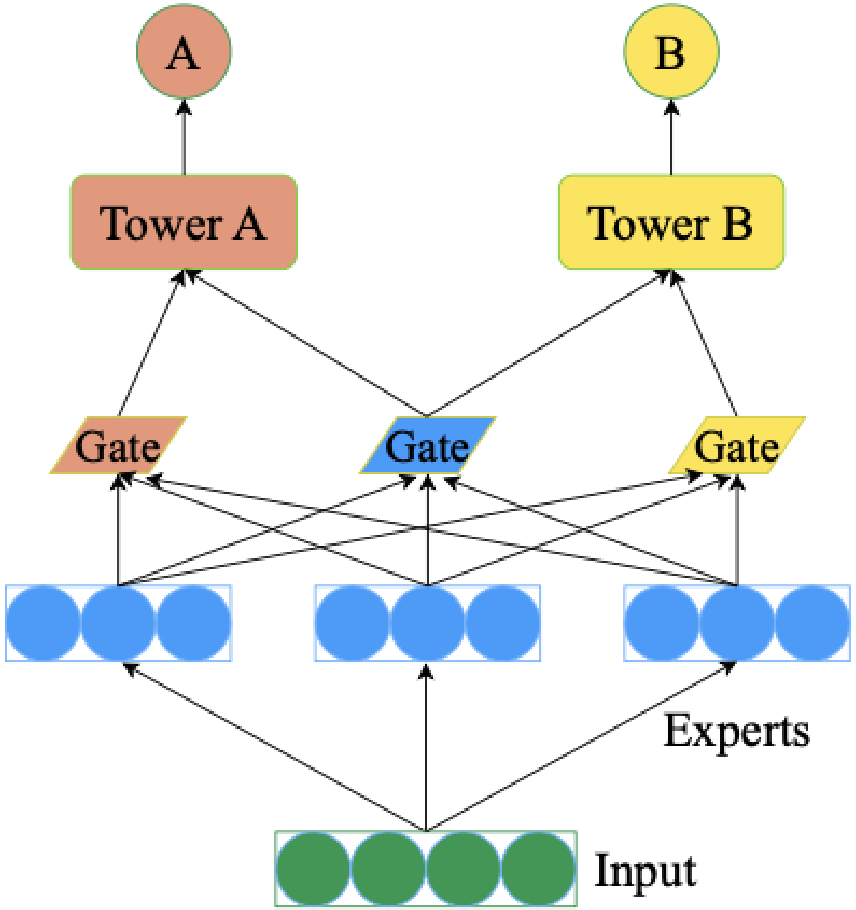
- ➢ Building separate models for tasks but the information among tasks is **fused by weights** of task relevance
- ✓ Relatively high **flexibility** in parameter sharing v.s. hard sharing
- ✗ Can not reconcile the flexibility
- ✗ Computation cost of the model

DINOP



Multi-task Based Sales Predictions for Online Promotions. CIKM 2019.

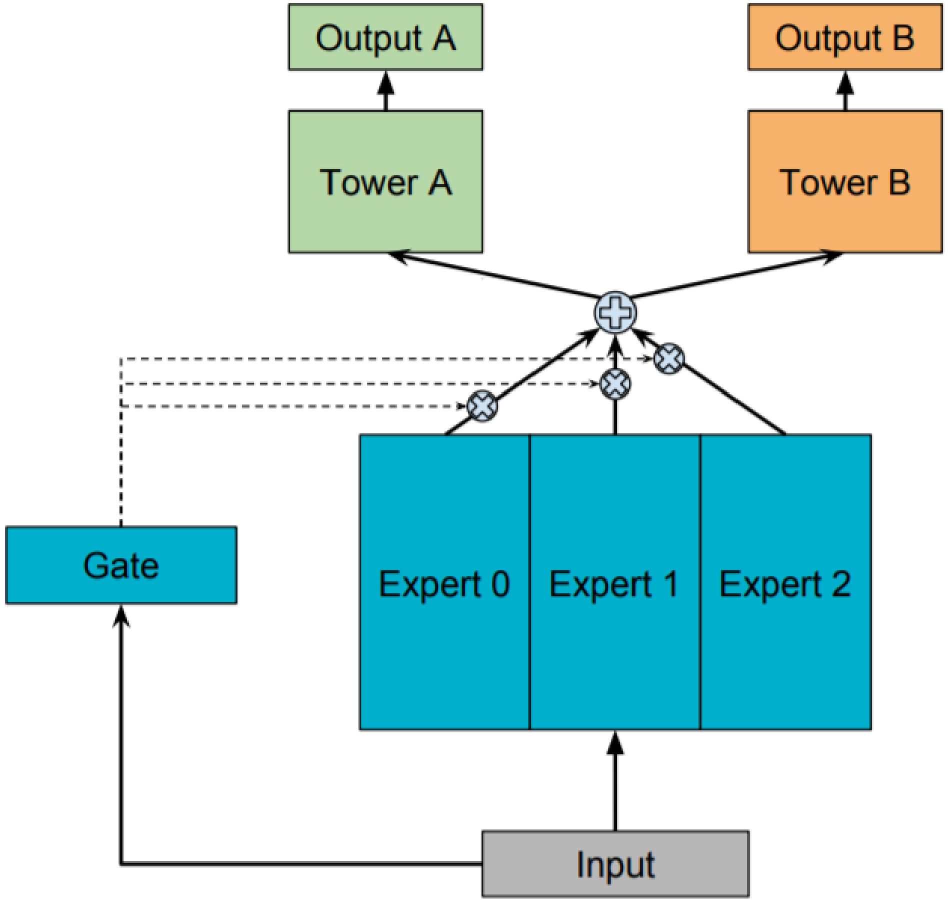
Expert Sharing



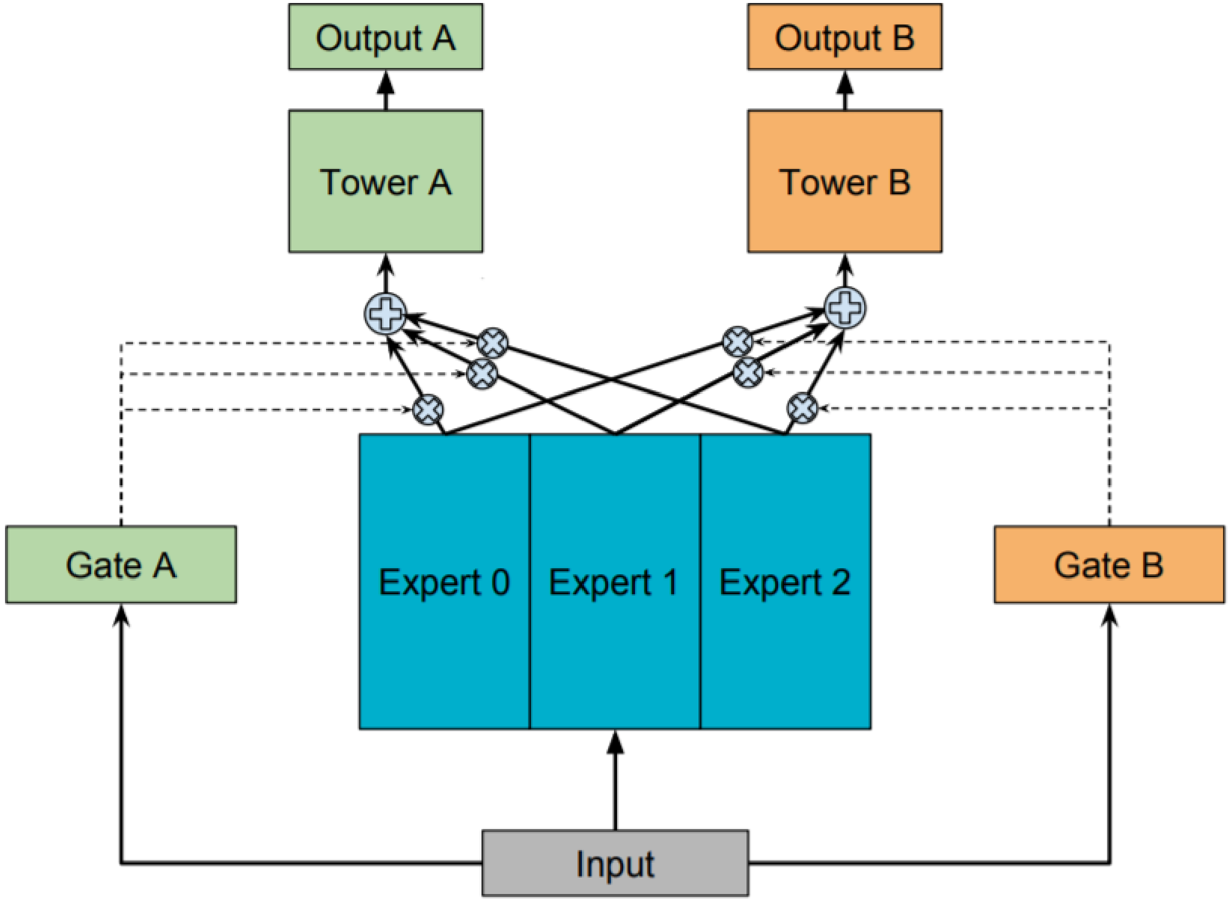
- Employing multiple **expert networks** to extract knowledge from shared bottom
  → Fed into **task-specific** modules like gates
  → Passed into the task-specific tower

- Mainly non-sequential input features
- **Special case** of Soft Sharing

MMoE



$$y = \sum_{i=1}^{n} g(x)_i f_i(x)$$



$$y_k = h^k(f^k(x)),$$

$$\text{where } f^k(x) = \sum_{i=1}^{n} g^k(x)_i f_i(x)$$

Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-experts. KDD 2018.

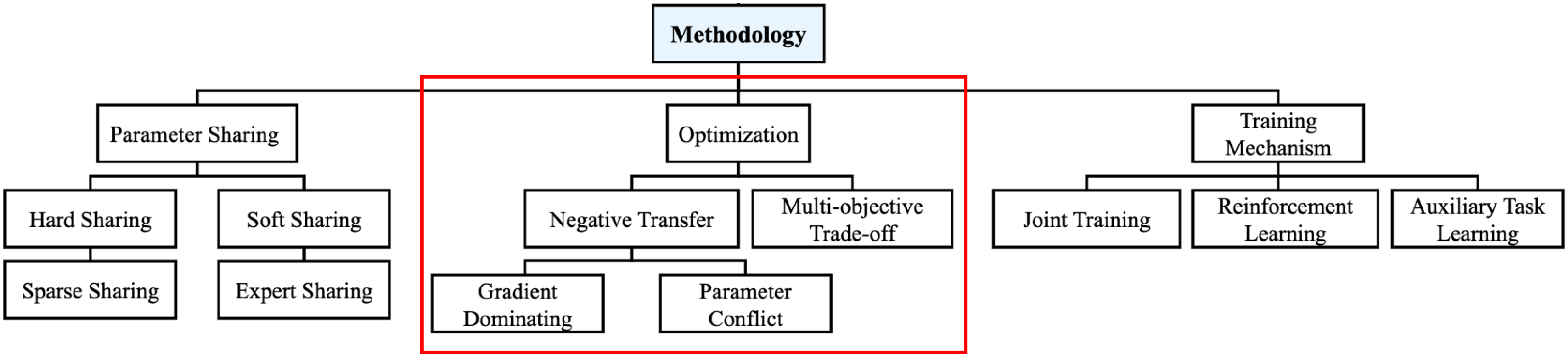
Expert Sharing

Model	Reference
MMoE	[Ma et al., 2018a]
SNR	[Ma et al., 2019]
PLE	[Tang et al., 2020]
DMTL	[Zhao et al., 2021]
DSelect-k	[Hazimeh et al., 2021]
MetaHeac	[Zhu et al., 2021]
PFE	[Xin et al., 2022]
MVKE	[Xu et al., 2022]
FDN	[Zhou et al., 2023]
MoSE	[Qin et al., 2020]

Processing **non-sequential** input features, while the remaining models is ameliorated based on MMoE

Processing **sequential** input features utilizing LSTM & sequential experts

Optimization



Negative Transfer

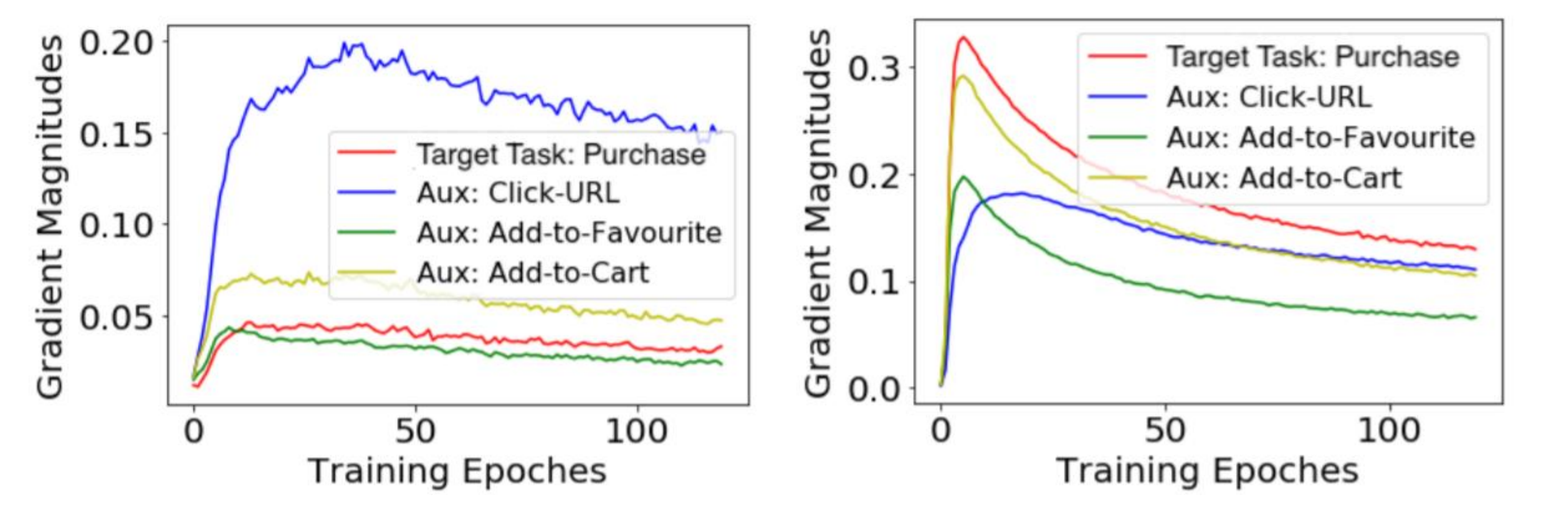
Gradient dominating $\|\nabla_\theta L^k(\theta)\|$

Works	Approach
AdaTask [Yang et al., 2022b]	Quantifying task dominance of shared parameters, calculate task-specific accumulative gradients
MetaBalance [He et al., 2022]	Flexibly balancing the gradient magnitude proximity between auxiliary and target tasks by a relax factor

Opposite directions of gradient + - $\nabla_\theta L^k(\theta)$

Works	Approach
PLE [Tang et al., 2020]	Proposing customized gate control (CGC) separating shared and task-specific experts
CSRec [Bai et al., 2022]	Alternating training procedure and contrastive learning on parameter masks to reduce the conflict probability

MetaBalance

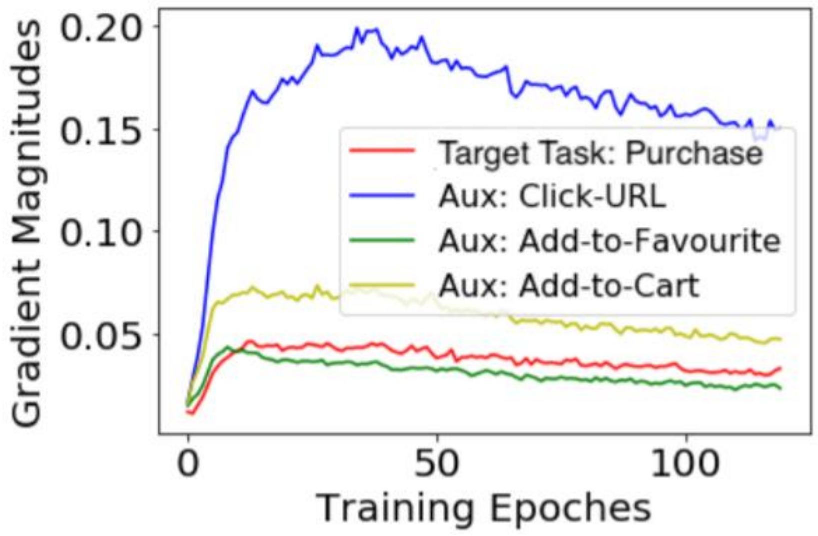


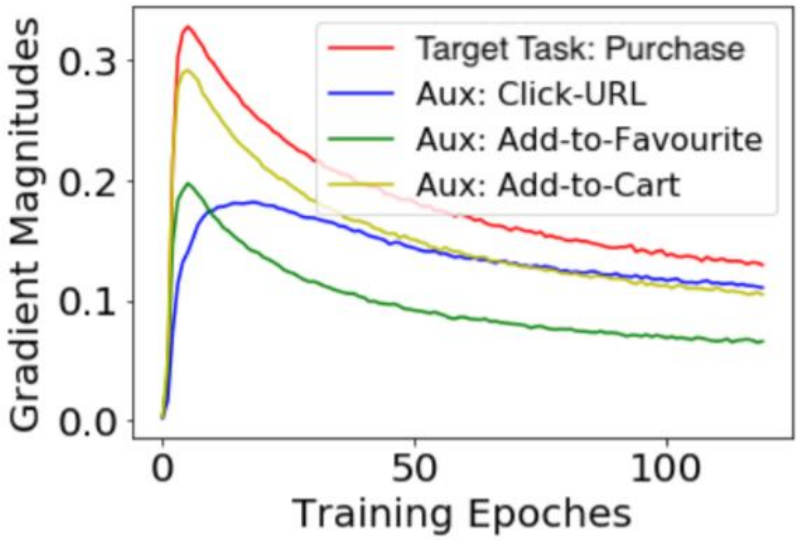
$$\theta^{t+1} = \theta^{t} - \alpha * \mathbf{G}^{t}_{total}$$

$$\mathbf{G}^{t}_{total} = \nabla_{\theta}\mathcal{L}^{t}_{total} = \nabla_{\theta}\mathcal{L}^{t}_{tar} + \sum_{i=1}^{K} \nabla_{\theta}\mathcal{L}^{t}_{aux,i}$$

Metabalance: Improving Multi-task Recommendations via Adapting Gradient Magnitudes of Auxiliary Tasks. WWW 2022.

MetaBalance



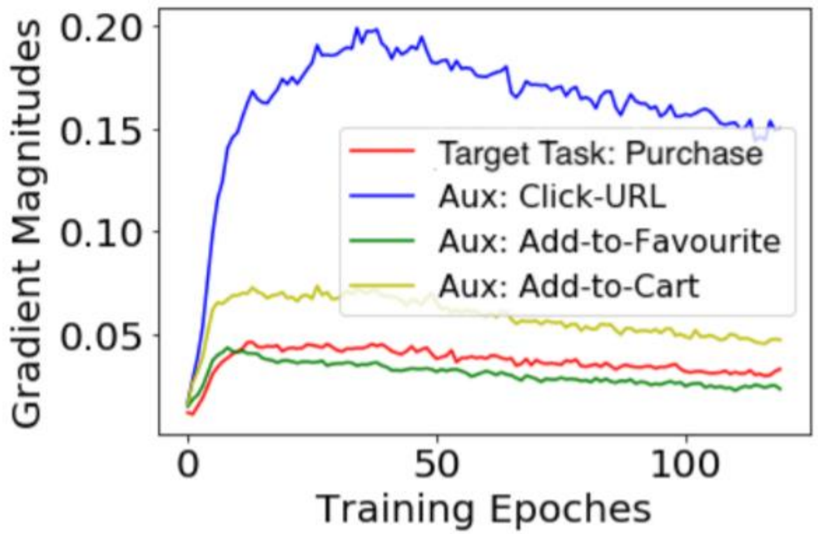


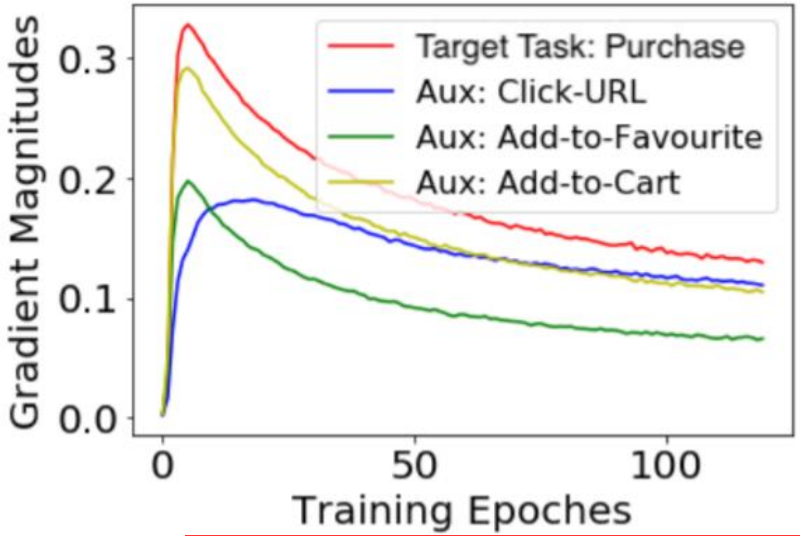
$$\theta^{t+1} = \theta^{t} - \alpha * \mathbf{G}^{t}_{total}$$

$$\mathbf{G}^{t}_{total} = \nabla_{\theta}\mathcal{L}^{t}_{total} = \nabla_{\theta}\mathcal{L}^{t}_{tar} + \sum_{i=1}^{K} \nabla_{\theta}\mathcal{L}^{t}_{aux,i}$$

Metabalance: Improving Multi-task Recommendations via Adapting Gradient Magnitudes of Auxiliary Tasks. WWW 2022.

MetaBalance





$$\theta^{t+1} = \theta^t - \alpha * \mathbf{G}^t_{total}$$

$$\mathbf{G}^t_{total} = \nabla_\theta \mathcal{L}^t_{total} = \nabla_\theta \mathcal{L}^t_{tar} + \sum_{i=1}^{K} \nabla_\theta \mathcal{L}^t_{aux,i}$$

$$\boxed{\mathbf{G}^t_{aux,i} \leftarrow \left(\mathbf{G}^t_{aux,i} * \frac{\|\mathbf{G}^t_{tar}\|}{\|\mathbf{G}^t_{aux,i}\|}\right) * r + \mathbf{G}^t_{aux,i} * (1 - r)}$$

Metabalance: Improving Multi-task Recommendations via Adapting Gradient Magnitudes of Auxiliary Tasks. WWW 2022.

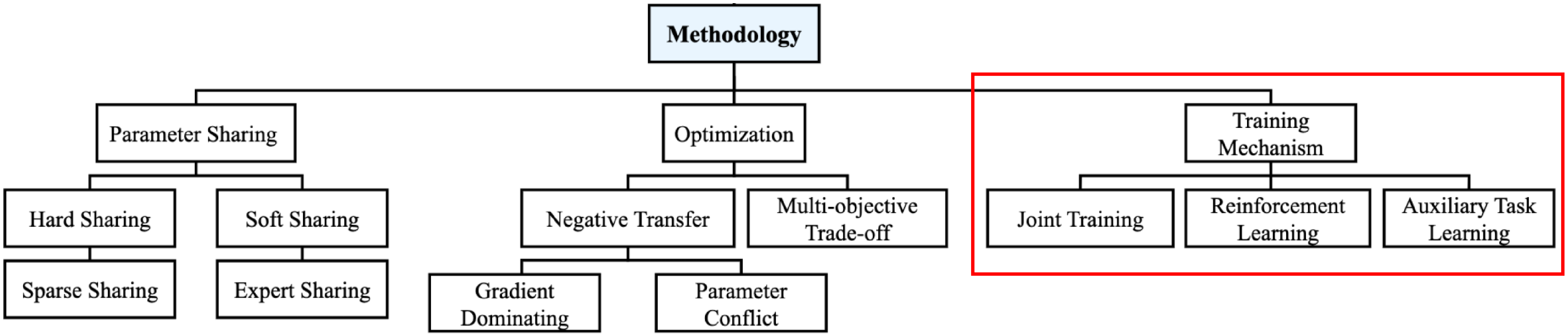
Multi-objective Trade-off

Objectives optimized regardless of the **potential conflict**

Works	Trade-off
[Wang *et al.*, 2021]	Group fairness and accuracy
[Wang *et al.*, 2022b]	Minimizing task conflicts and improving multi-task generalization

Training Mechanism

Training process & Learning strategy



Joint Training

Parallel manner

Category	Reference
Session-based RS	[Shalaby et al., 2022] [Qiu et al., 2021] [Meng et al., 2020]
Route RS	[Das, 2022]
Knowledge graph enhanced RS	[Wang et al., 2019]
Explainability	[Lu et al., 2018] [Wang et al., 2018]
Graph-based RS	[Wang et al., 2020a]

Reinforcement Learning

Sequential user behaviors as MDP

Summary	Reference
Formulating MTF as MDP and use batch RL to optimize long-term user satisfaction	[Zhang et al., 2022b]
Using an actor-critic model to learn the optimal fusion weight of tasks rather than greedy ranking strategies	[Han et al., 2019]
Using dynamic critic networks to adaptively adjust the fusion weight considering the session-wise property	[Liu et al., 2023]

Auxiliary Task Learning

Joint training & Others

Summary	Reference
Employing Expectation-Maximization (EM) algorithm for optimization	ESDF [Wang et al., 2020b]
Trained with task-specific sub- networks	Self-auxiliaries [Wang et al., 2022b]

Application Fields

- **E-commerce** : Main focus
- **Advertising**
  - Utility & Cost
    i. MM-DFM [Hou et al., 2021]: Performing multiple conversion prediction tasks in different observation duration
    ii. MetaHeac [Zhu et al., 2021]: Handling audience expansion tasks on content-based mobile marketing
    iii. MVKE [Xu et al., 2022]: Performing user tagging for online advertising
- **Social media**
    i. MMoE [Zhao et al., 2019b]: YouTube - engagement and satisfaction
    ii. LT4REC [Xiao et al., 2020]: Tencent Video
    iii. BatchRL-MTF [Zhang et al., 2022b]: Tencent short video platform

Datasets

Datasets	Stage	Tasks	Website
Ali-CCP [42]	Ranking	CTR, CVR	https://tianchi.aliyun.com/dataset/408/
Criteo [13]	Ranking	CTR, CVR	https://ailab.criteo.com/criteo-attribution-modeling-bidding-dataset/
AliExpress [32]	Ranking	CTR, CTCVR	https://tianchi.aliyun.com/dataset/74690/
MovieLens [23]	Recall & Ranking	Watch, Rating	https://grouplens.org/datasets/movielens/
Yelp	Recall & Ranking	Rating, Explanation	https://www.yelp.com/dataset/
Amazon [25]	Recall & Ranking	Rating, Explanation	http://jmcauley.ucsd.edu/data/amazon/
Kuairand [18]	Recall & Ranking	Click, Like, Follow, Comment, . . .	https://kuairand.com/
Tenrec [77]	Recall & Ranking	Click, Like, Share, Follow, . . .	https://github.com/yuangh-x/2022-NIPS-Tenrec/

Challenges & Future Directions

Topic	Challenge & future direction
Negative Transfer	• Extra complex inter-task correlation • What, where, and when to transfer to alleviate negative transfer
AutoML	• Existing models only focus on the **parameter sharing routing**, while other components and hyper-parameters still under-explored
Explainability	• Complex task relevance
Task-specific Biases	• Most existing models only focus on **one** specific bias • **Multiple** bias should be tackled in future

Challenges & Future Directions

Topic	Challenge & future direction
Negative Transfer	• Extra complex inter-task correlation • What, where, and when to transfer to alleviate negative transfer
AutoML	• Existing models only focus on the **parameter sharing routing**, while other components and hyper-parameters still under-explored
Explainability	• Complex task relevance
Task-specific Biases	• Most existing models only focus on **one** specific bias • **Multiple** bias should be tackled in future

Challenges & Future Directions

Topic	Challenge & future direction
Negative Transfer	• Extra complex inter-task correlation • What, where, and when to transfer to alleviate negative transfer
AutoML	• Existing models only focus on the **parameter sharing routing**, while other components and hyper-parameters still under-explored
Explainability	• Complex task relevance
Task-specific Biases	• Most existing models only focus on **one** specific bias • **Multiple** bias should be tackled in future

Challenges & Future Directions

Topic	Challenge & future direction
Negative Transfer	• Extra complex inter-task correlation • What, where, and when to transfer to alleviate negative transfer
AutoML	• Existing models only focus on the **parameter sharing routing**, while other components and hyper-parameters still under-explored
Explainability	• Complex task relevance
Task-specific Biases	• Most existing models only focus on **one** specific bias • **Multiple** bias should be tackled in future

Conclusion

- Task relation:

  Parallel, Cascaded, Auxiliary with Main

- Methodology:

  Parameter Sharing, Optimization, Training Mechanism

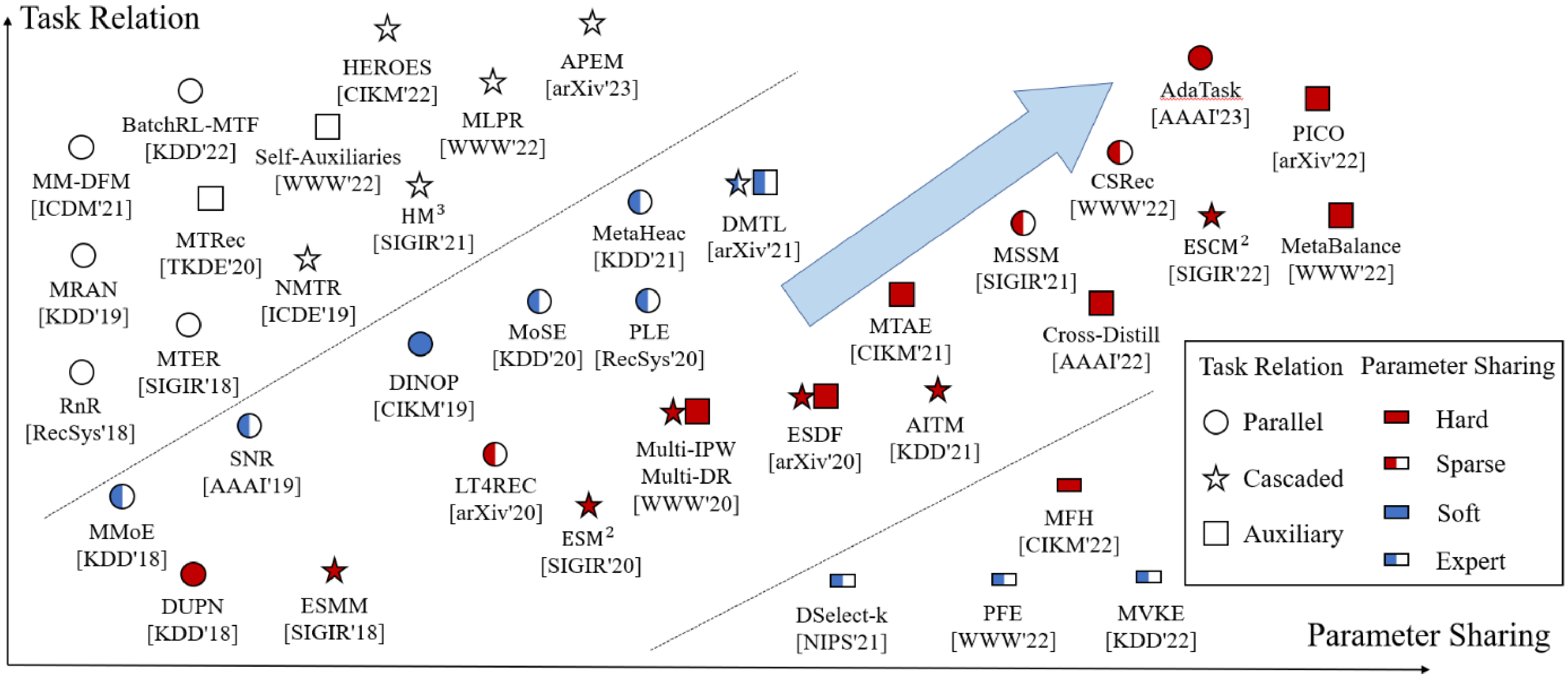
Survey

https://arxiv.org/abs/2302.03525

Multi-Task Deep Recommender Systems: A Survey

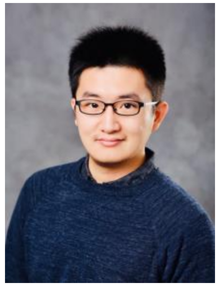
YUHAO WANG*, HA TSZ LAM*, and YI WONG*, City University of Hong Kong
ZIRU LIU, City University of Hong Kong
XIANGYU ZHAO†, City University of Hong Kong
YICHAO WANG, BO CHEN, HUIFENG GUO, and RUIMING TANG†, Huawei Noah's Ark Lab

Trend of MTDRS



Agenda

Introduction



Xiangyu Zhao

Preliminary

Yichao Wang

Multi-task Recommendation

Yuhao Wang

Multi-scenario recommendation

MTR+MSR

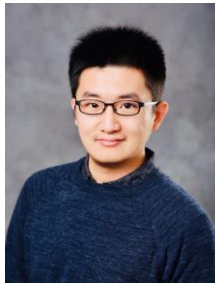
Pengyue Jia

More Joint-learning Methods

Jingtong Gao

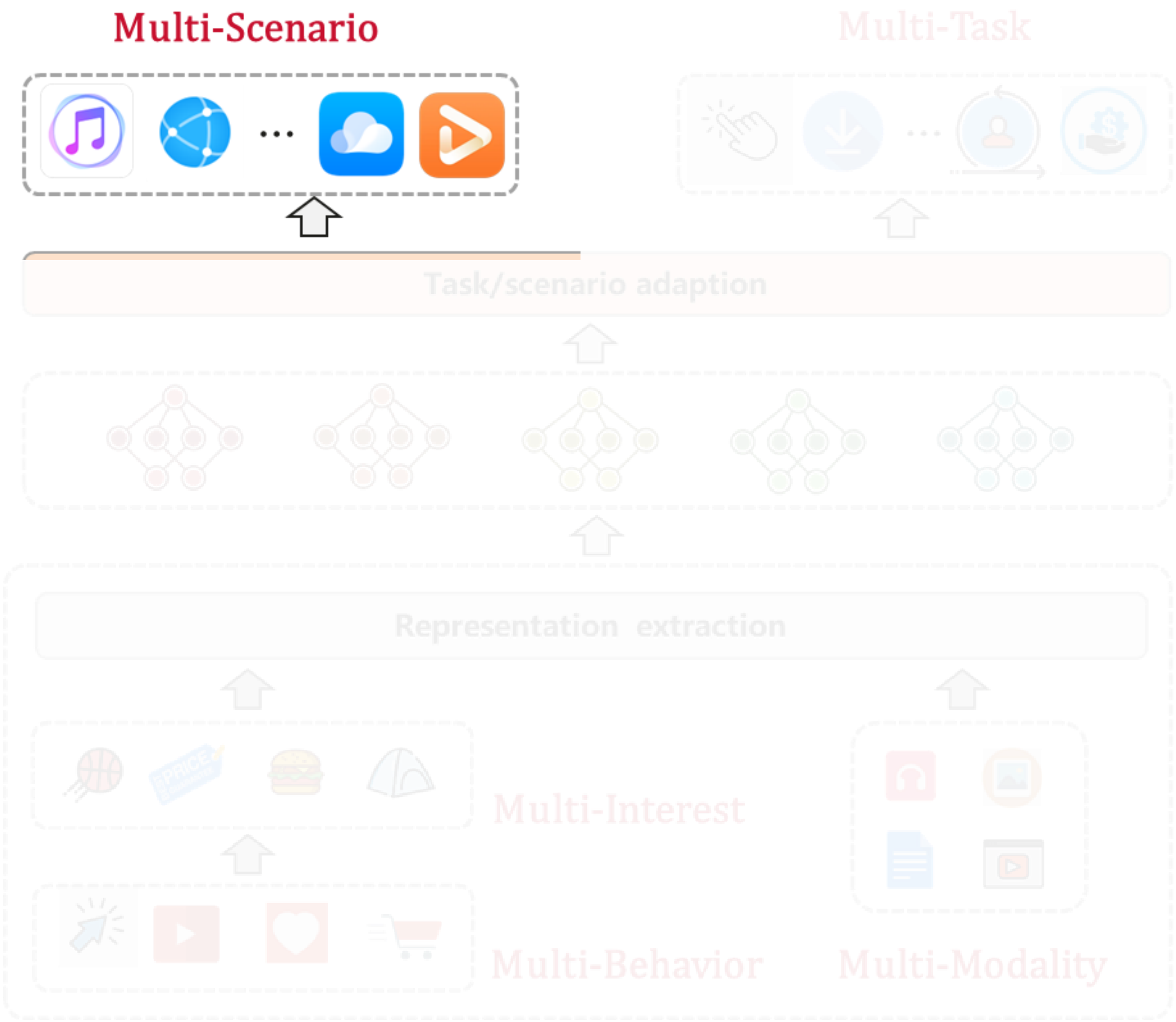
Conclusion

Future Work



Xiangyu Zhao

Multi-Scenario Recommender Systems



$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$ Joint Modeling

$E^{Merge} = U(E, E^{Ext}, E^m)$

$E^{Ext} = F(H^{UB})$ Multi-Interest

$H^{UB} = G(H_1, H_2, \ldots, H_N)$ Multi-Behavior

$E^m = M(E^{txt}, E^v, \ldots, E^p)$ Multi-Modality

$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$ **Multi-Scenario**

$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$ Multi-Task

Background

- Multi-Scenario Recommender Systems:
  - By using a unified model to simultaneously model multiple scenarios, the goal of improving the effects of different scenarios at the same time is achieved through information transfer between scenarios.
- Importance:
  - Time/Memory efficiency; Maintenance cost
  - Accuracy
- Classification on Methods:
  - Shared-Specific network paradigm
  - Dynamic weight
  - Multi-scenario & Multi-task recommendation
- Formulation:

$$wL(E^{Merge}, \Theta, \Theta^{t}, \Theta^{S})$$

  - θ: parameters of the backbone network
  - θ^{S}: parameters of modeling scenarios

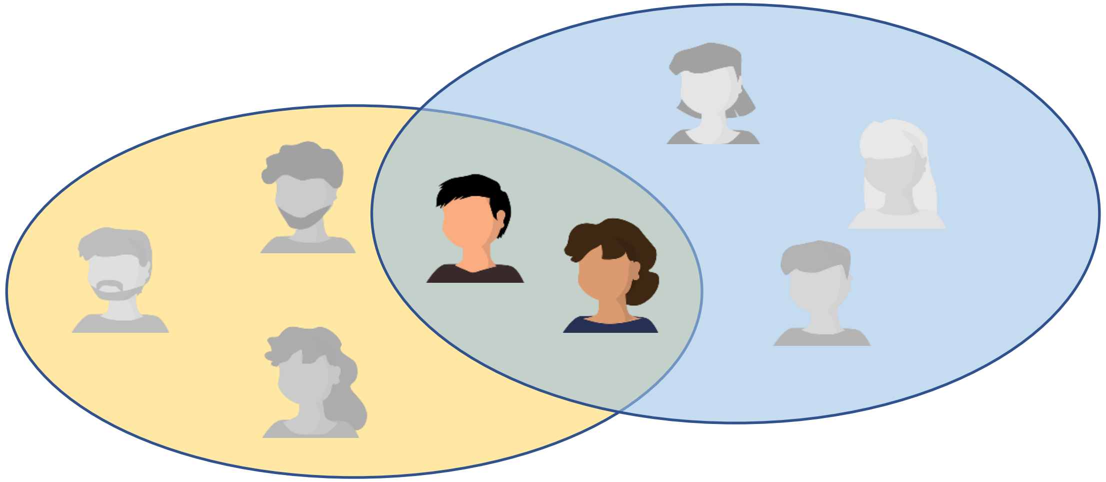
Recommendation Scenarios

- What is Scenario?
  - Homepage, Searching page, Detailed page …
  - Food, Leisure and entertainment, …
  - Usually refers to different business scenarios

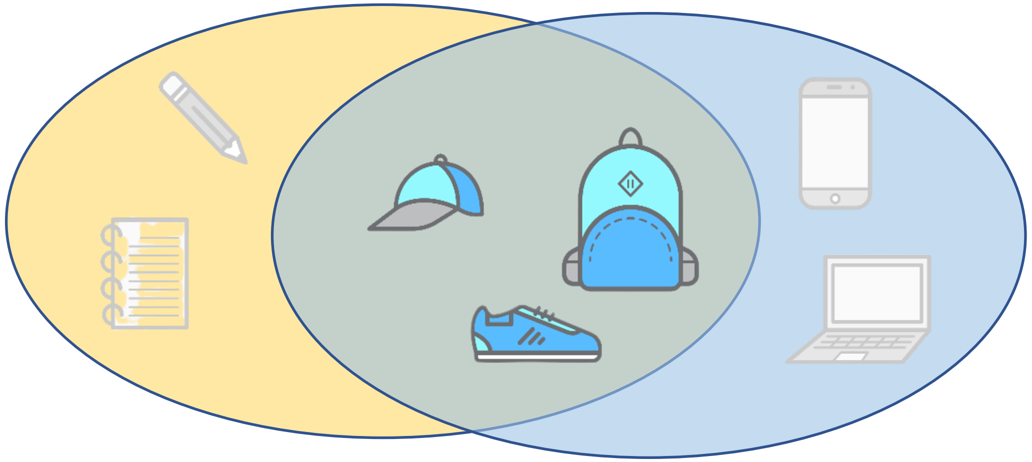
- Scenario and Domain?
  - Generally do not make a distinction
  - The same in this tutorial

Commonalities and Diversities

- Commonalities
  - User Overlap



- Commonalities
  - Item Overlap



Commonalities and Diversities

- Diversities
  - The specific user group may be different
  - User's interest changes with the scenarios

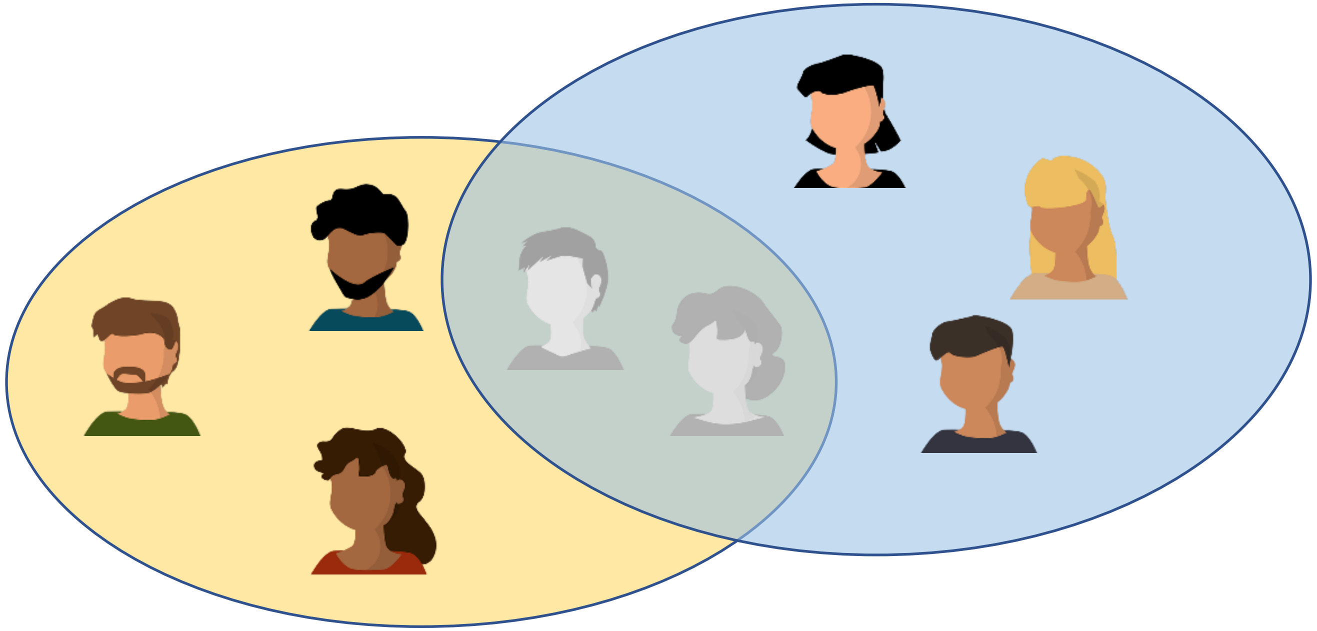
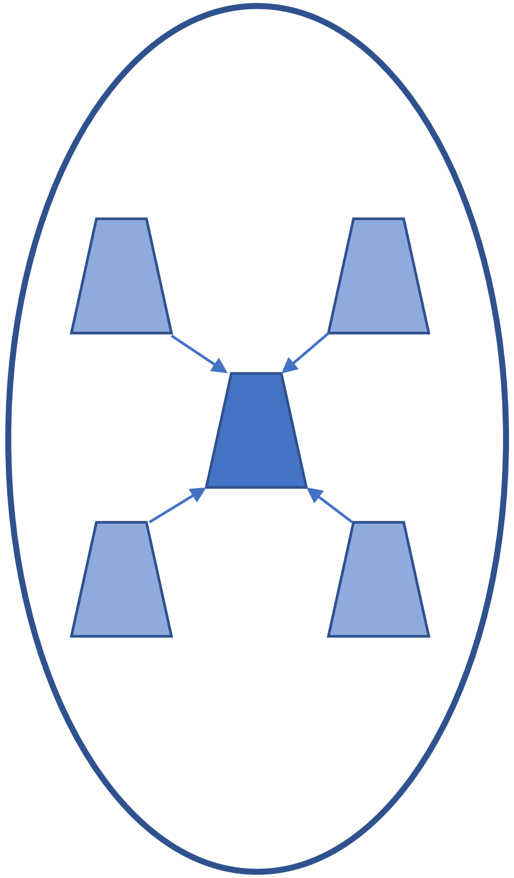
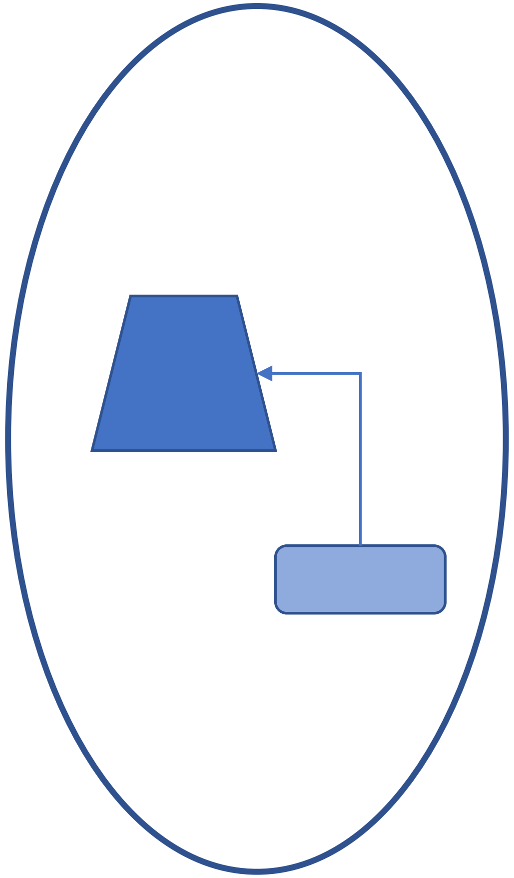


Table of Contents



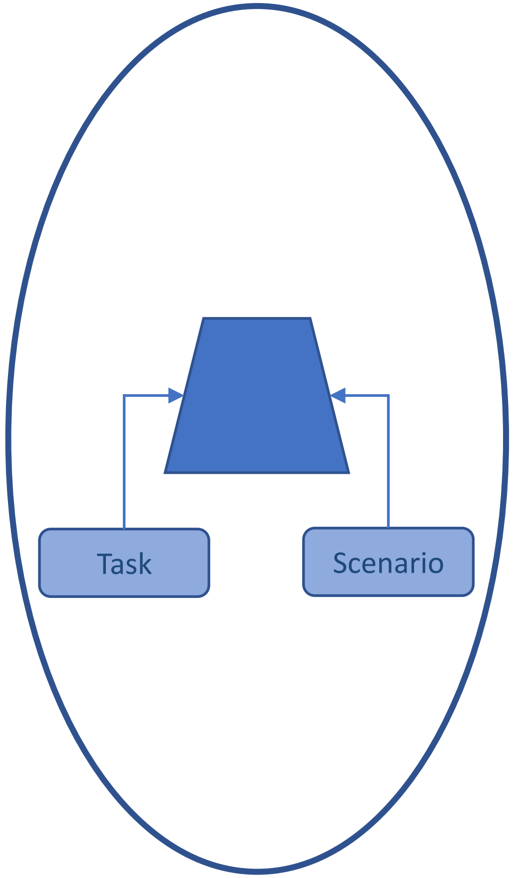
Shared-specific network paradigm

$wL(E^{Merge}, \Theta, \Theta^t, (\Theta^{shared}, \Theta^{specific}))$



Dynamic weight

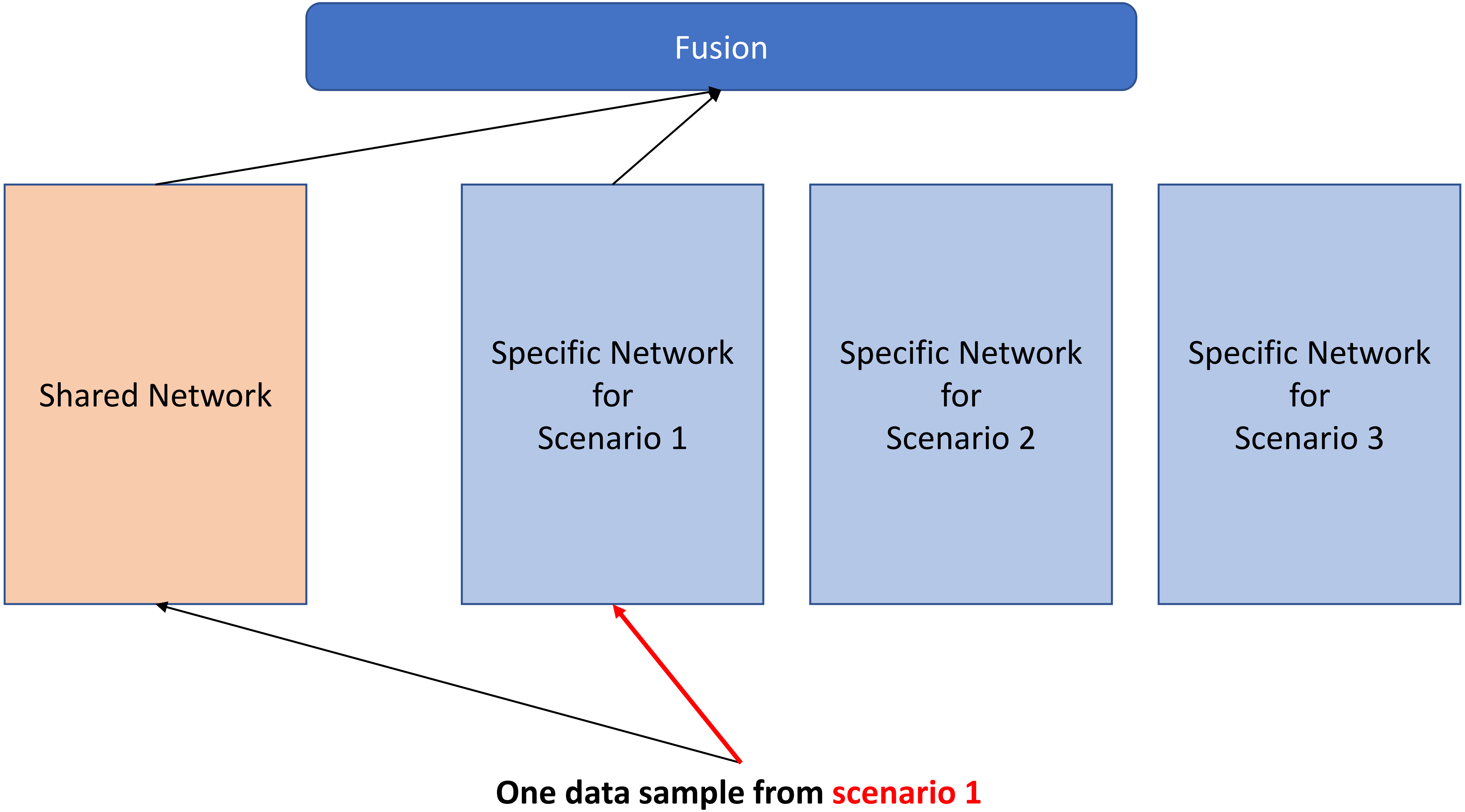
$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$



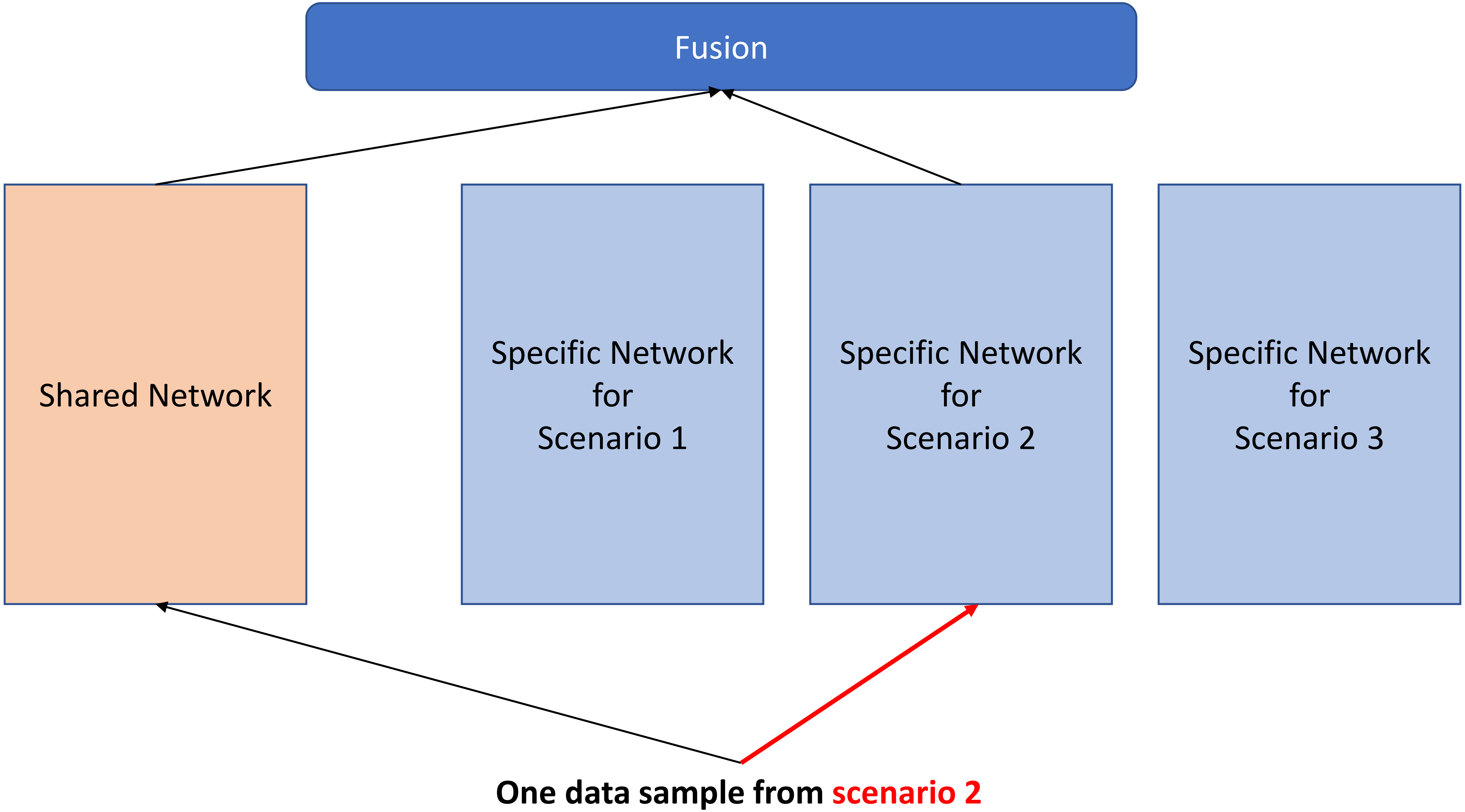
Multi-Scenario & Multi-Task

$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s, \Theta^T)$

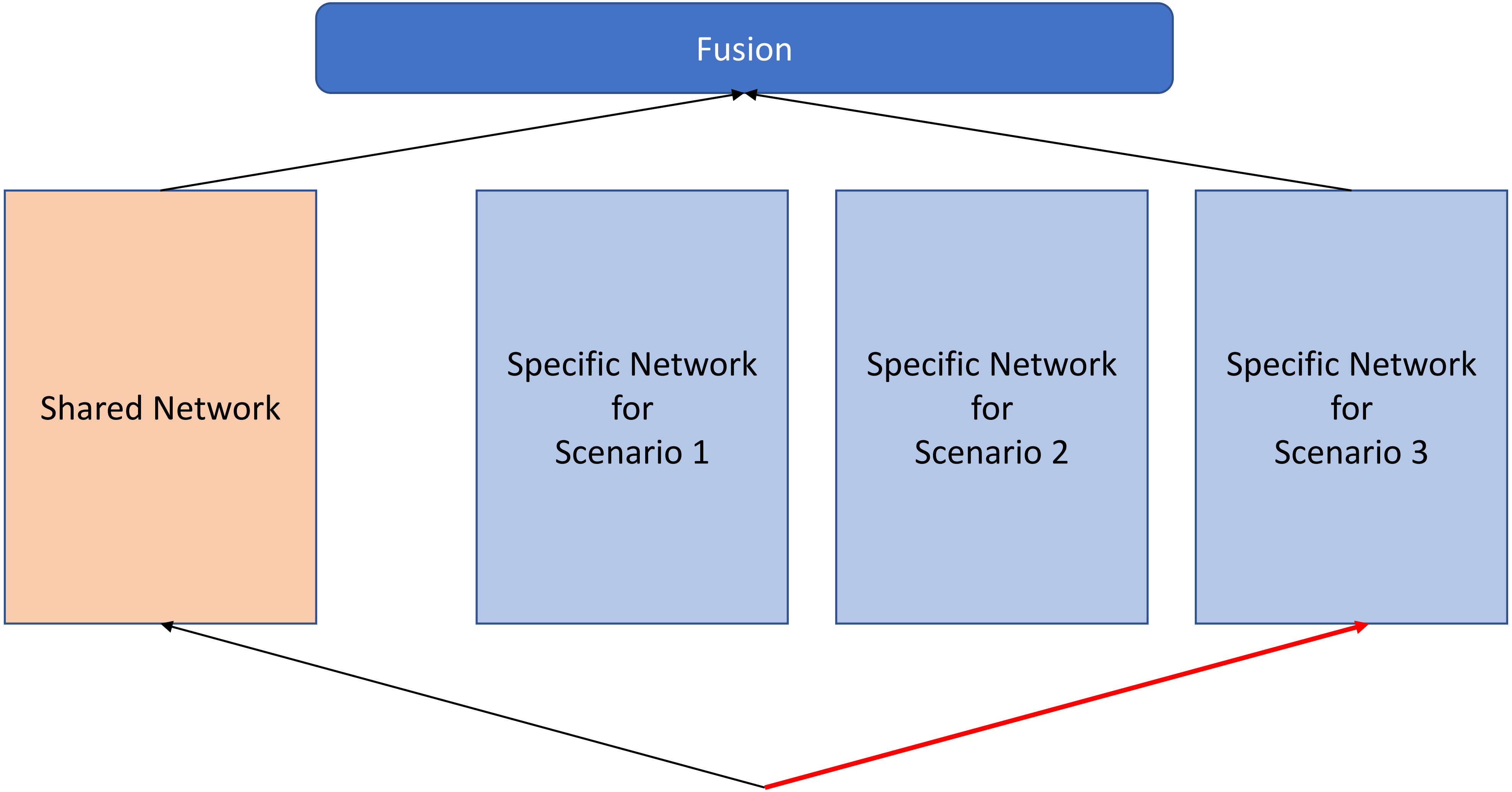
Shared-specific Network Paradigm



Shared-specific Network Paradigm



Shared-specific Network Paradigm



One data sample from scenario 3

STAR

- Motivation:
  - Training individual models for each domain → does not fully use the data from all domains
  - Data across domains owns commonalities and characteristics

- Target:
  - Use a single model to serve multiple domains simultaneously
  - Shared network → commonalities
  - Specific network → characteristics

- Methods:
  - Partitioned Normalization
  - STAR Topology
  - Auxiliary Network

Banner

Guess What You Like

STAR Details

- Partitioned Normalization (PN)
- Training

$$z' = (\gamma * \gamma_p)\frac{z-\mu}{\sqrt{\sigma^2+\epsilon}} + (\beta+\beta_p)$$

- Testing

$$z' = (\gamma * \gamma_p)\frac{z-E_p}{\sqrt{Var_p+\epsilon}} + (\beta+\beta_p)$$

Compared to BN

- Batch Normalization (BN)
- Training

$$\mathbf{z}' = \gamma\frac{\mathbf{z}-\mu}{\sqrt{\sigma^2+\epsilon}} + \beta$$

- Testing

$$\mathbf{z}' = \gamma\frac{\mathbf{z}-E}{\sqrt{Var+\epsilon}} + \beta$$

STAR Details

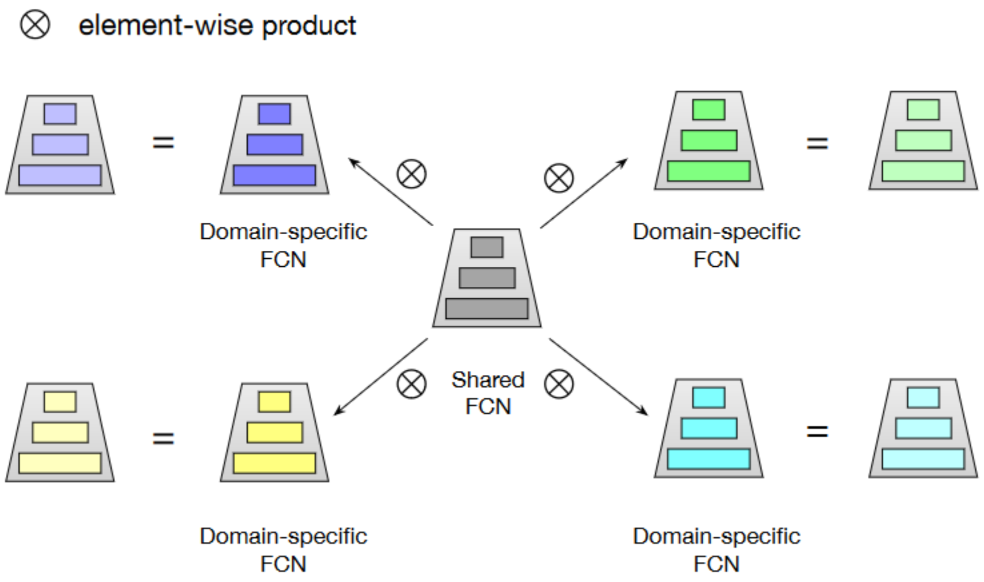
STAR Topology

The final weight and bias for p-th domain is obtained by:

$$W_p^{\star} = W_p \otimes W, b_p^{\star} = b_p + b$$

The output for p-th domain is derived by:

$$out_p = \phi((W_p^{\star})^{\top} in_p + b_p^{\star})$$



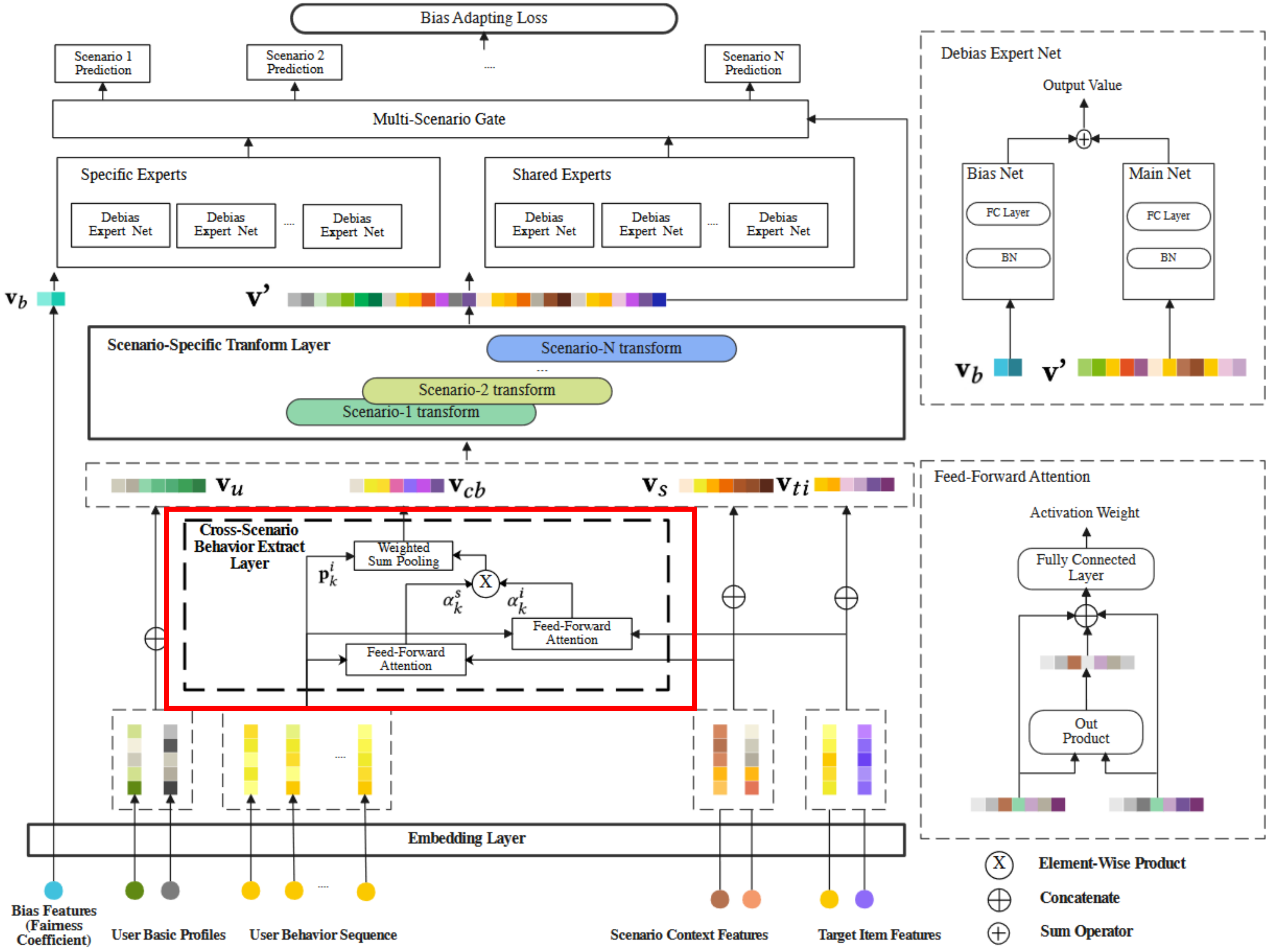
SAR-Net

- Motivation
  - Traffic characteristics of different scenarios are significantly different (individual data scale or representative topic)

- Target
  - Train a unified model to serve all scenarios

SAR-Net: A Scenario-Aware Ranking Network for Personalized Fair Recommendation in Hundreds of Travel Scenarios. CIKM 21.

SAR-Net Details



Cross-Scenario Behavior Extract Layer

How to aggregate the sequence?

p^{B^i} is item behavior sequence

$$\mathbf{p}_k^i = [\mathbf{e}_{itemId} || \mathbf{e}_{destination} || \mathbf{e}_{category} || \cdots]$$

p^{B^s} is scenario context sequence

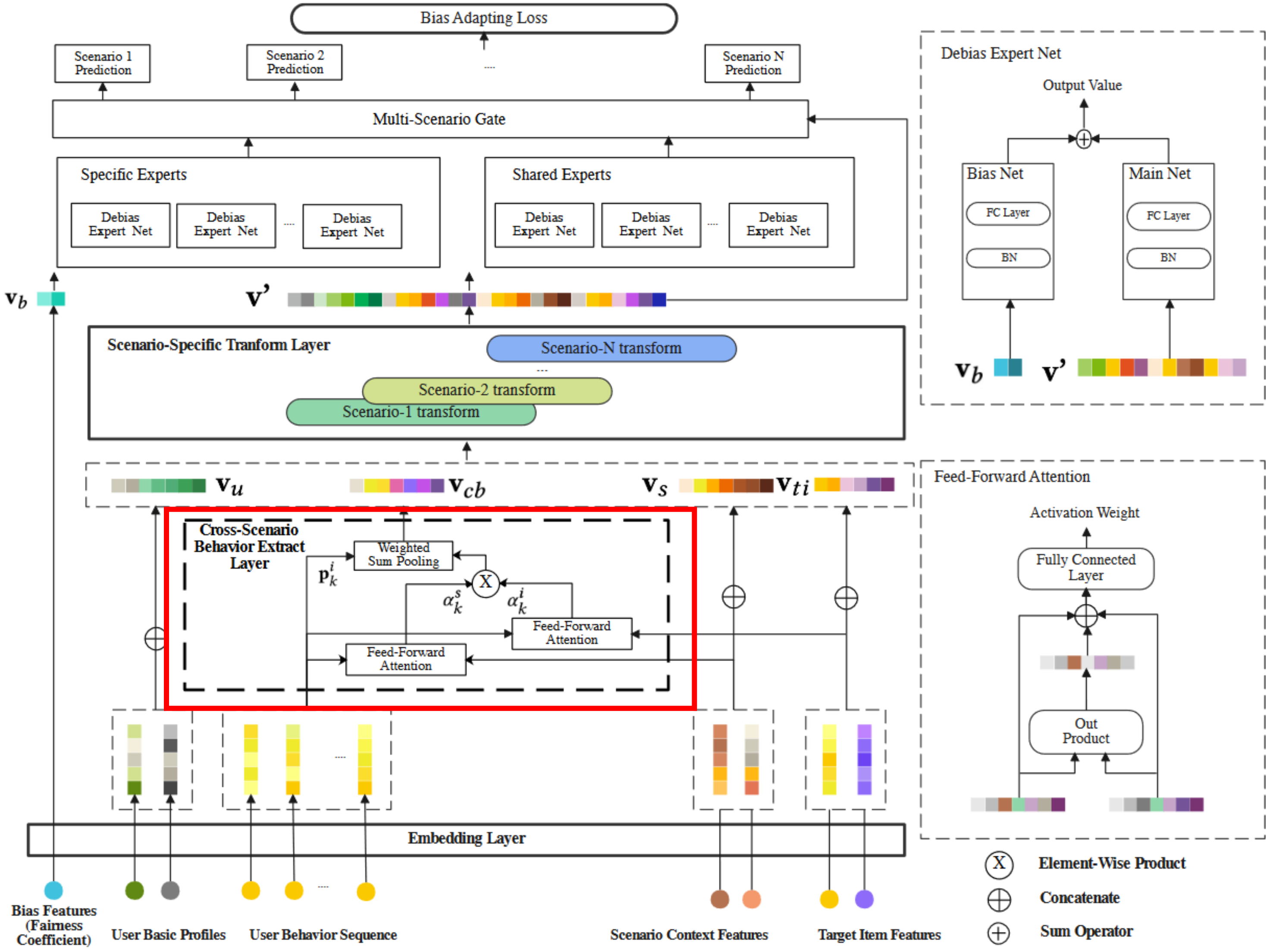
$$\mathbf{p}_k^s = [\mathbf{e}_{scenarioId} || \mathbf{e}_{scenarioType} || \mathbf{e}_{behaviorTime} || \cdots]$$

$$\alpha_k^i = \frac{\exp(\psi(\mathbf{p}_k^i, \mathbf{p}_t^i))}{\sum_{l=1}^{|\mathbf{p}(B^i)|} \exp(\psi(\mathbf{p}_l^i, \mathbf{p}_t^i))},$$

$$\alpha_k^s = \frac{\exp(\psi(\mathbf{p}_k^s, \mathbf{p}_t^s))}{\sum_{l=1}^{|\mathbf{p}(B^s)|} \exp(\psi(\mathbf{p}_l^s, \mathbf{p}_t^s))},$$

α_k^i and α_k^s indicate the relevance between user's kth behavior item and the target item or target scenario

SAR-Net Details



Cross-Scenario Behavior Extract Layer

How to aggregate the sequence?

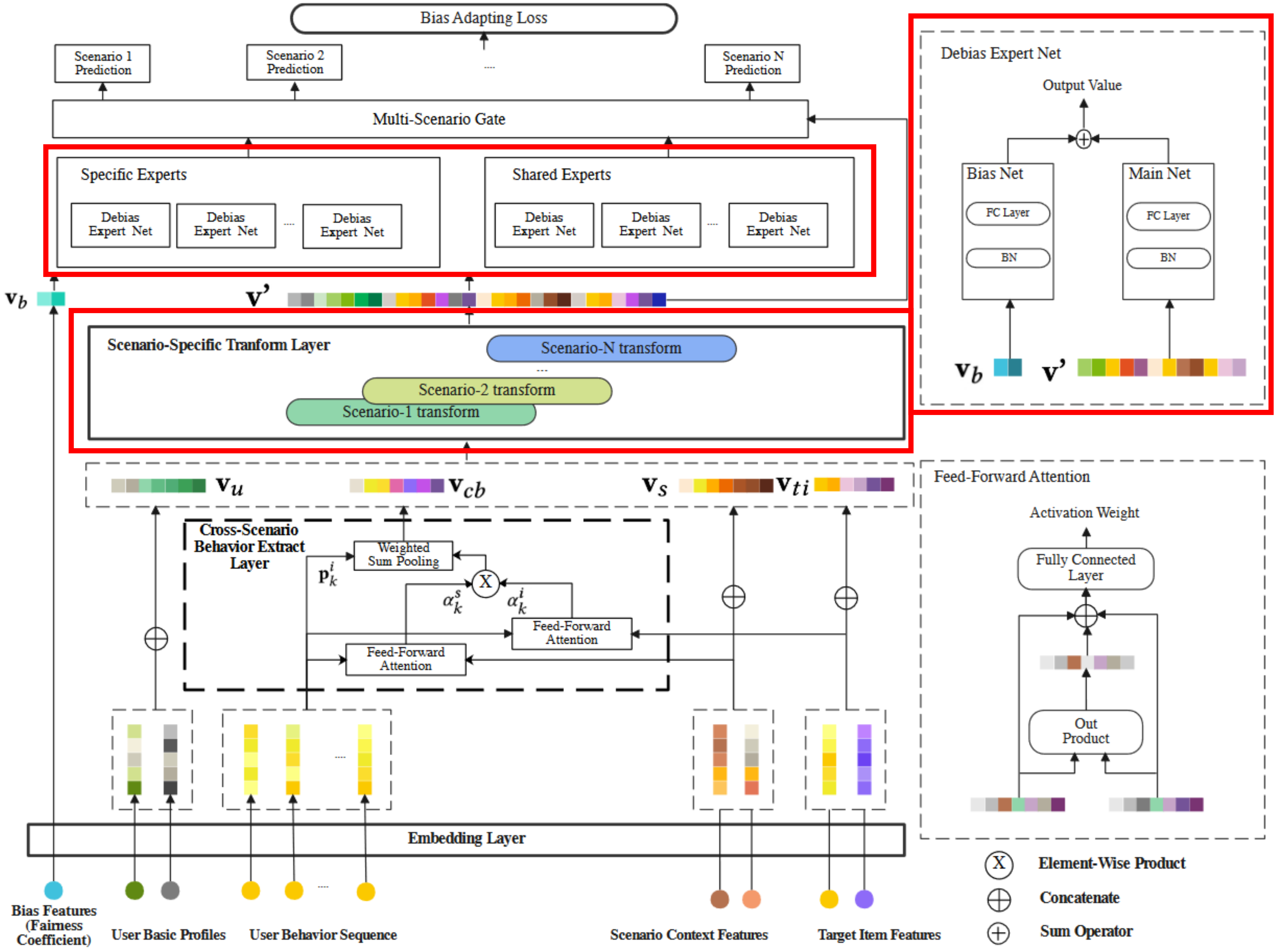
$$\alpha_k^i = \frac{\exp(\psi(\mathbf{p}_k^i, \mathbf{p}_t^i))}{\sum_{l=1}^{|\mathbf{p}(B^i)|} \exp(\psi(\mathbf{p}_l^i, \mathbf{p}_t^i))},$$

$$\alpha_k^s = \frac{\exp(\psi(\mathbf{p}_k^s, \mathbf{p}_t^s))}{\sum_{l=1}^{|\mathbf{p}(B^s)|} \exp(\psi(\mathbf{p}_l^s, \mathbf{p}_t^s))},$$

$$\mathbf{p}_k^i = [\mathbf{e}_{itemId} || \mathbf{e}_{destination} || \mathbf{e}_{category} || \cdots]$$

$$\mathbf{v}_{cb} = \sum_{k=1}^{t} \alpha_k^i * \alpha_k^s * \mathbf{p}_k^i$$

SAR-Net



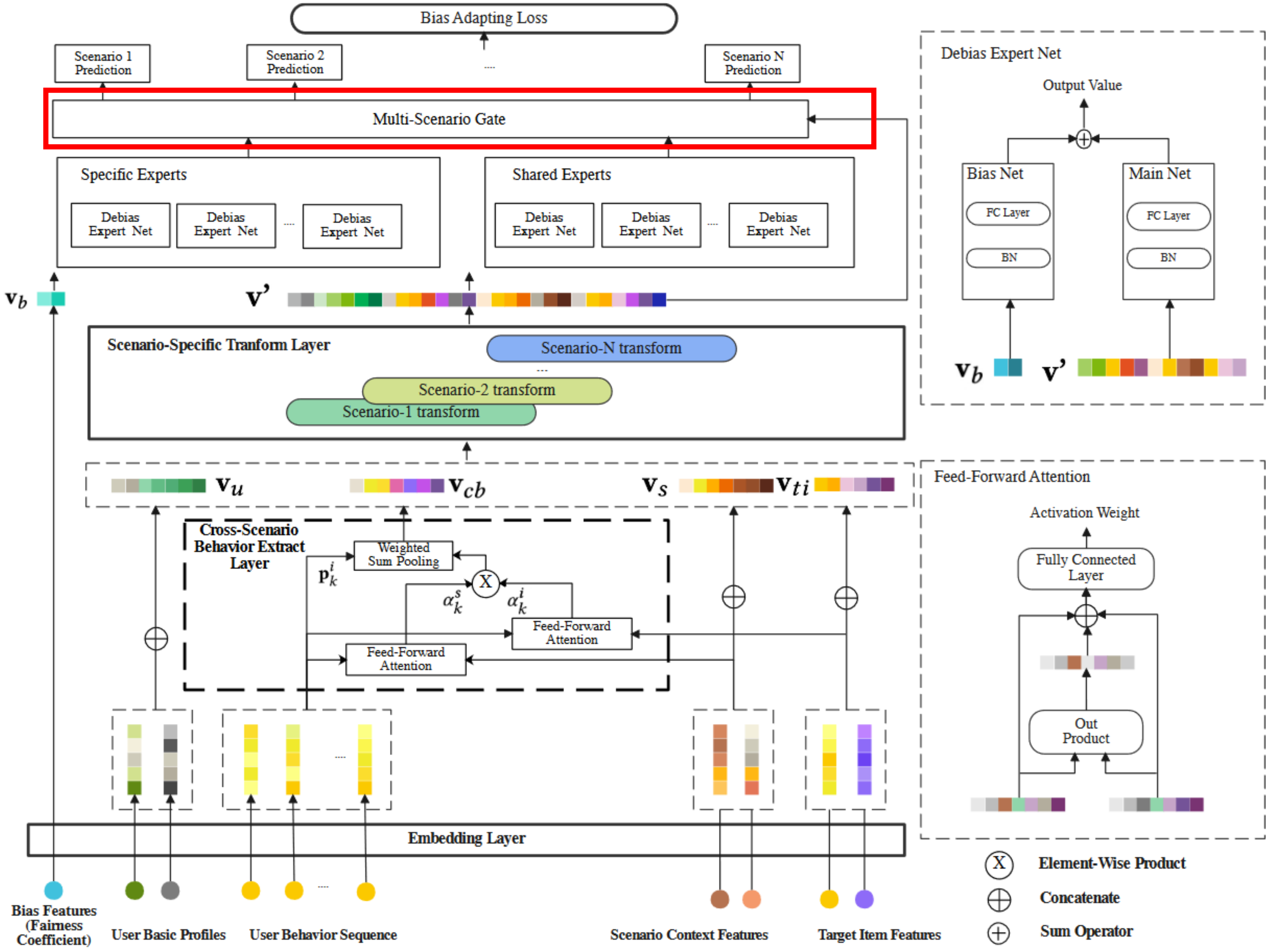
Scenario-Specific Transform Layer

$$\mathbf{v}' = \mathbf{v} \otimes \beta_i + \gamma_i$$

Mixture of Debias Experts

Multi-expert network. Each scenario has some scenario-specific experts and all the scenarios share several common experts.

SAR-Net



Multi-Gate Network & Prediction

The output of experts:

$$S^k(x) = [o_{k,1}, o_{k,2}, \cdots, o_{k,m_k}, o_{s,1}, o_{s,2}, \cdots, o_{s,m_s}]^T$$

Final predicted score of scenario k

$$y^k(x) = w^k(x) S^k(x)$$

$w^k(x)$ is derived by a single-layer feed-forward network with a SoftMax activation function

ADI

➢Motivation

- Separate model for each scenario, ignoring the cross-domain overlapping of user groups and items
- One shared model trained on mix data, model performance may decrease when different domains conflict

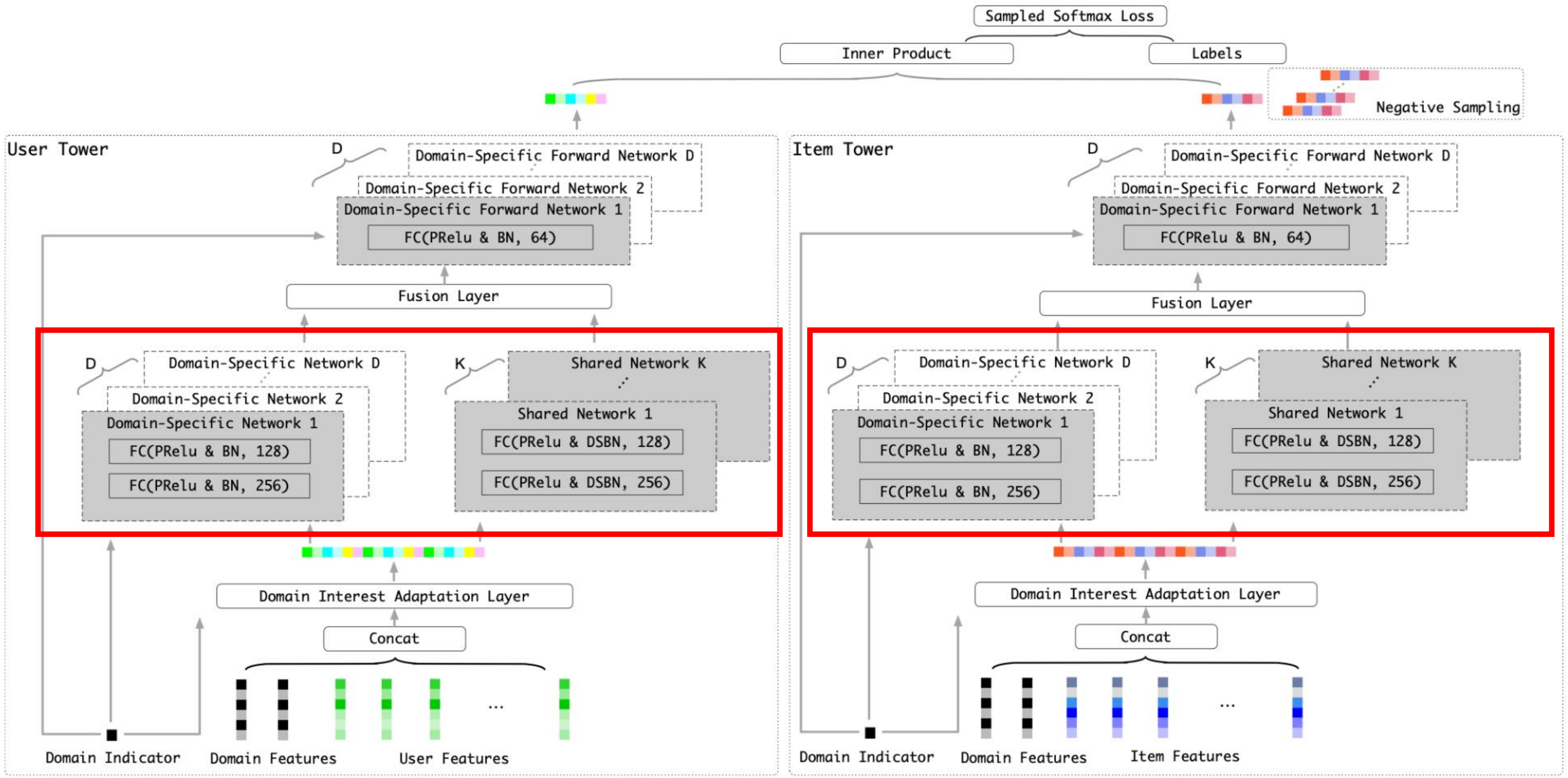
➢Target

- Modeling commonalities and diversities → common networks and domain-specific networks
- Tackle the feature-level domain adaptation → domain-specific batch normalization, domain interest adaptation layer

Adaptive Domain Interest Network for Multi-domain Recommendation. CIKM 22.

ADI Details

Backbone Network



Shared Network & Domain-Specific Network

$$az_k = \frac{W^k_{shared}(f_{domain}) + b^k_{shared}}{\sum_{n=1}^{K}(W^n_{shared}(f_{domain}) + b^n_{shared})}$$

$$E_{shared} = \sum_{k=1}^{K} \alpha_k MLP^k_{shared}(\mathbf{F})$$

$$E^{(d)}_{spec} = MLP^{(d)}_{spec}(\mathbf{F}^{(d)})$$

f_{domain} Domain indicator embedding

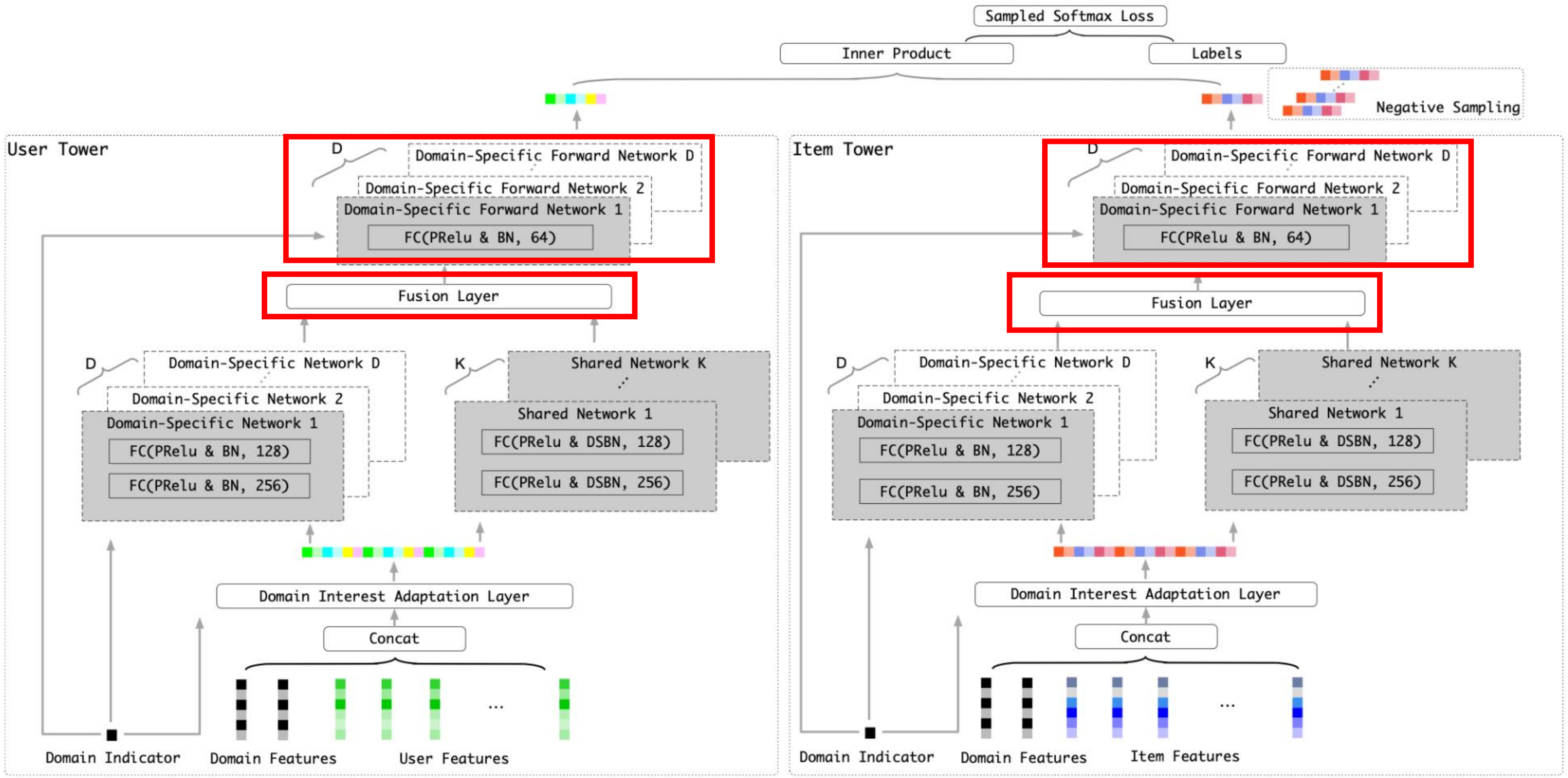
$\mathbf{F}^{(d)}$ Data from domain d

K hyperparameter, number of Shared Network

D domains, D Domain-Specific Network

ADI Details

Backbone Network



Fusion Layer

$$\beta_1^{(d)} = \sigma(W_{fusion\_spec}^{(d)}(f_{domain}))$$
$$\beta_2^{(d)} = \sigma(W_{fusion\_shared}^{(d)}(f_{domain}))$$

$$E_{fusion}^{(d)} = concat(\beta_1^{(d)} E_{spec}^{(d)} \mid \beta_1^{(d)} E_{spec}^{(d)} \odot \beta_2^{(d)} E_{shared} \mid \beta_2^{(d)} E_{shared})$$

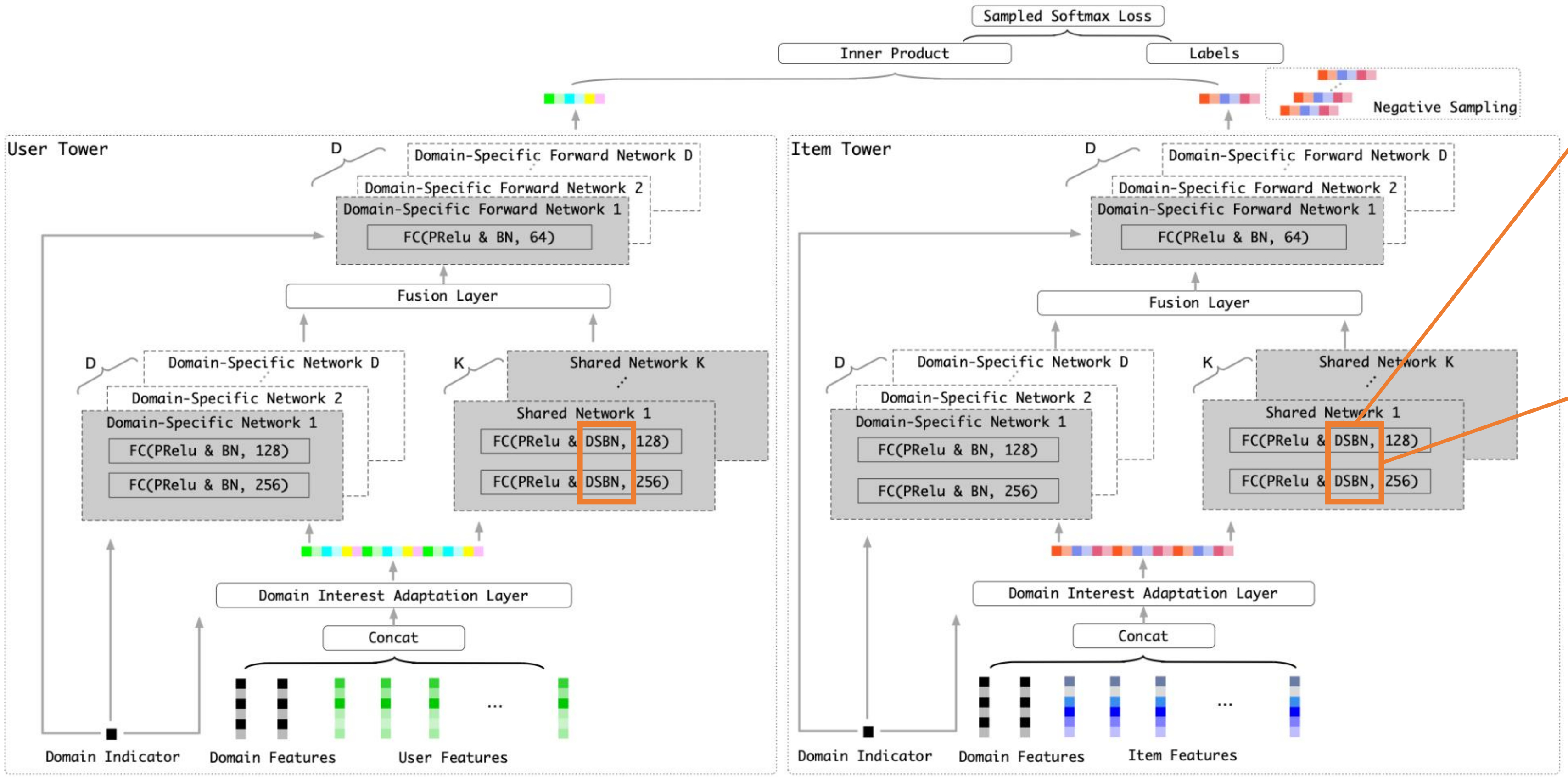
Domain-Specific Forward Network

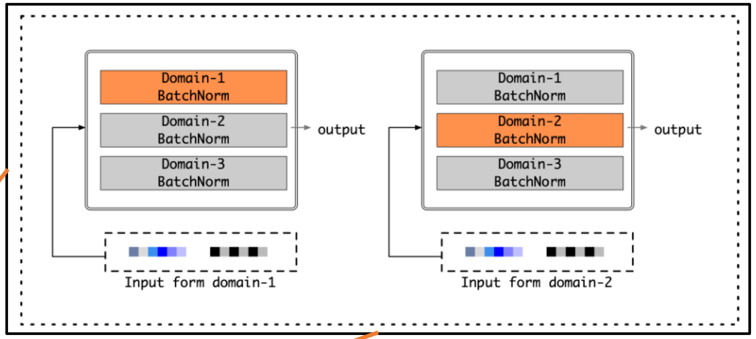
$$E = FC_{forward}^{(d)}(E_{fusion}^{(d)})$$

ADI Details

Domain Adaptation

Domain-Specific Batch Normalization (DSBN)



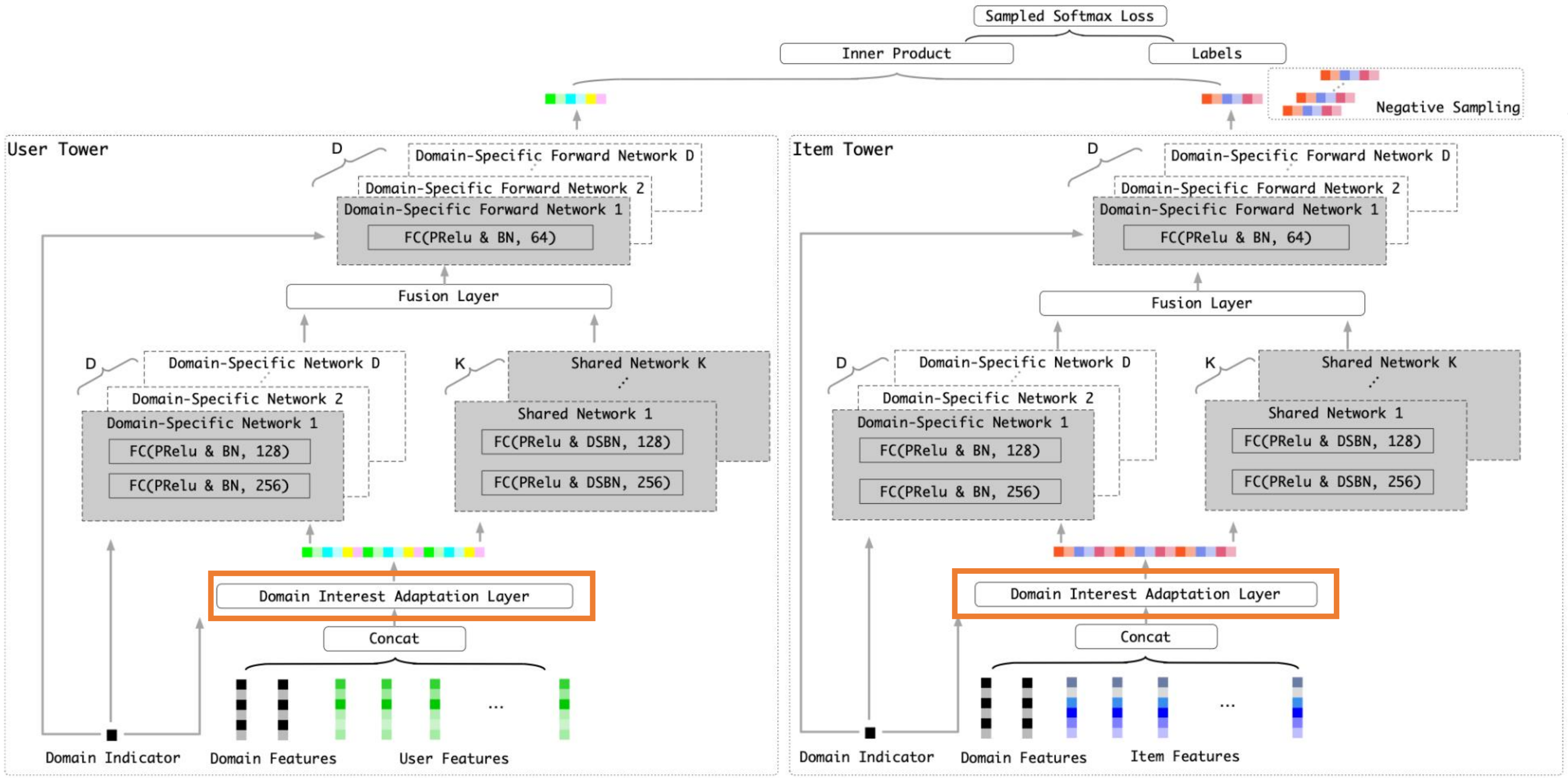


$$\hat{\mathbf{X}}^{(d)} = \alpha^{(d)} \frac{\mathbf{X}^{(d)} - \mu^{(d)}}{\sqrt{(\sigma^{(d)})^2 + \epsilon}} + \beta^{(d)}$$

ADI Details

Domain Adaptation

Domain Interest Adaptation Layer



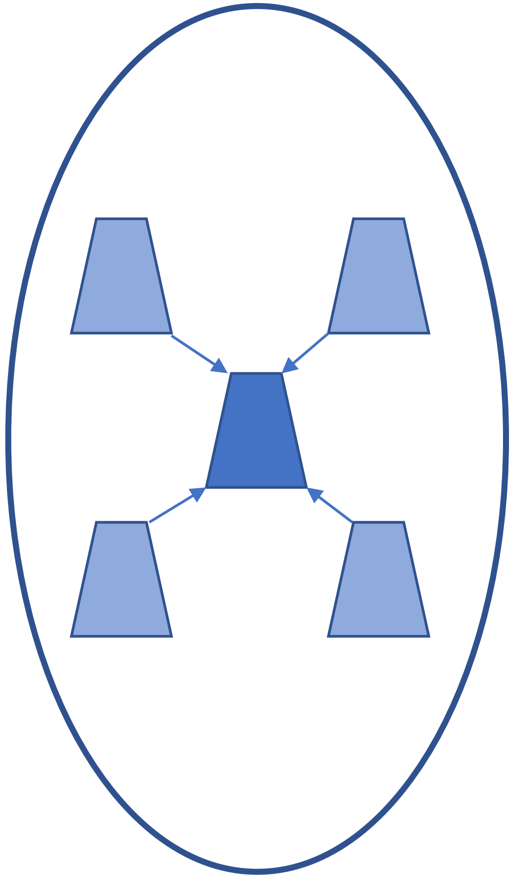
$$\alpha^{(d)} = F_{se}(concat(F_{avg}(F_1^{(d)}) \mid \cdots \mid F_{avg}(F_N^{(d)})))$$

$$\hat{F}^{(d)} = \alpha^{(d)} \otimes concat(F_1^{(d)} \mid \cdots \mid F_N^{(d)})$$

$F_i^{(d)}$ denotes ith feature of embedded input collected from domain d

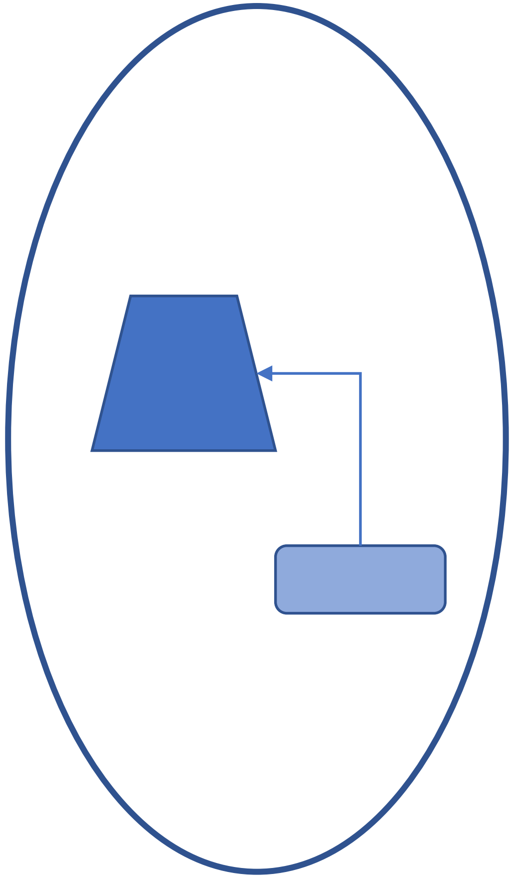
F_{se} denotes a $(FC, Relu, FC)$ block and F_{avg} denotes average pooling operator.

Table of Contents



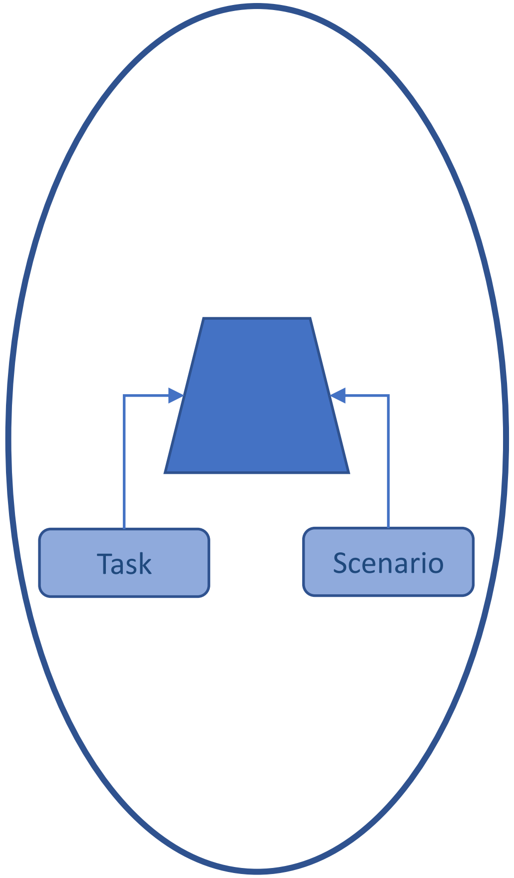
Shared-specific network paradigm

$wL(E^{Merge}, \Theta, \Theta^{t}, (\Theta^{shared}, \Theta^{specific}))$



Dynamic weight

$wL(E^{Merge}, \Theta, \Theta^{t}, \Theta^{s})$

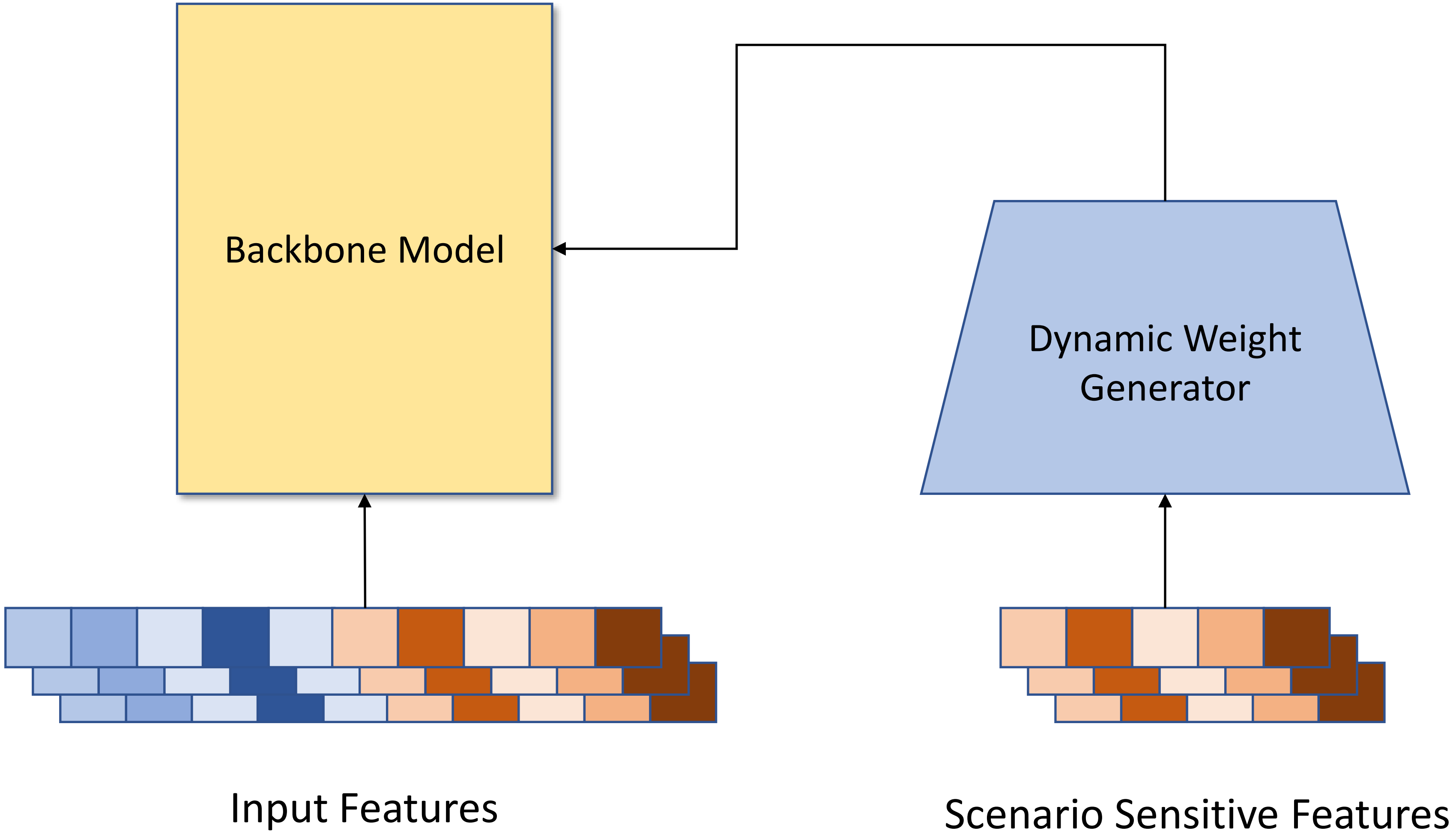


Multi-Scenario & Multi-Task

$wL(E^{Merge}, \Theta, \Theta^{t}, \Theta^{s}, \Theta^{T})$

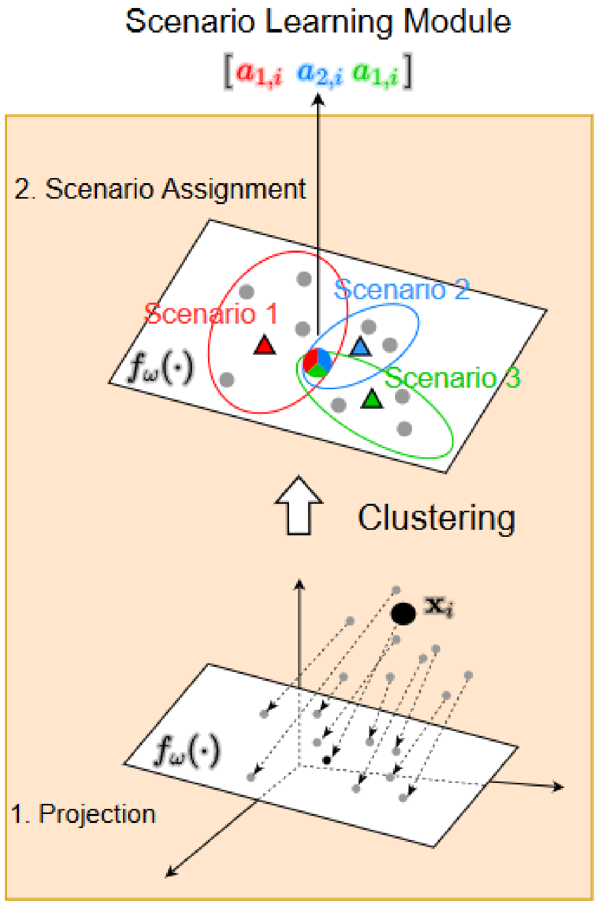
Dynamic Weight

➢ Why Dynamic?



MUSENET

- ➢ Target
  - To mine and model implicit scenarios
- ➢ Methods
  - Scenario Learning Module to project data samples, and assign scenarios to these data samples



MUSENET: Multi-Scenario Learning for Repeat-Aware Personalized Recommendation. WSDM 23.

MUSENET Details

Soft Assignment

$$\Lambda = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K\}$$

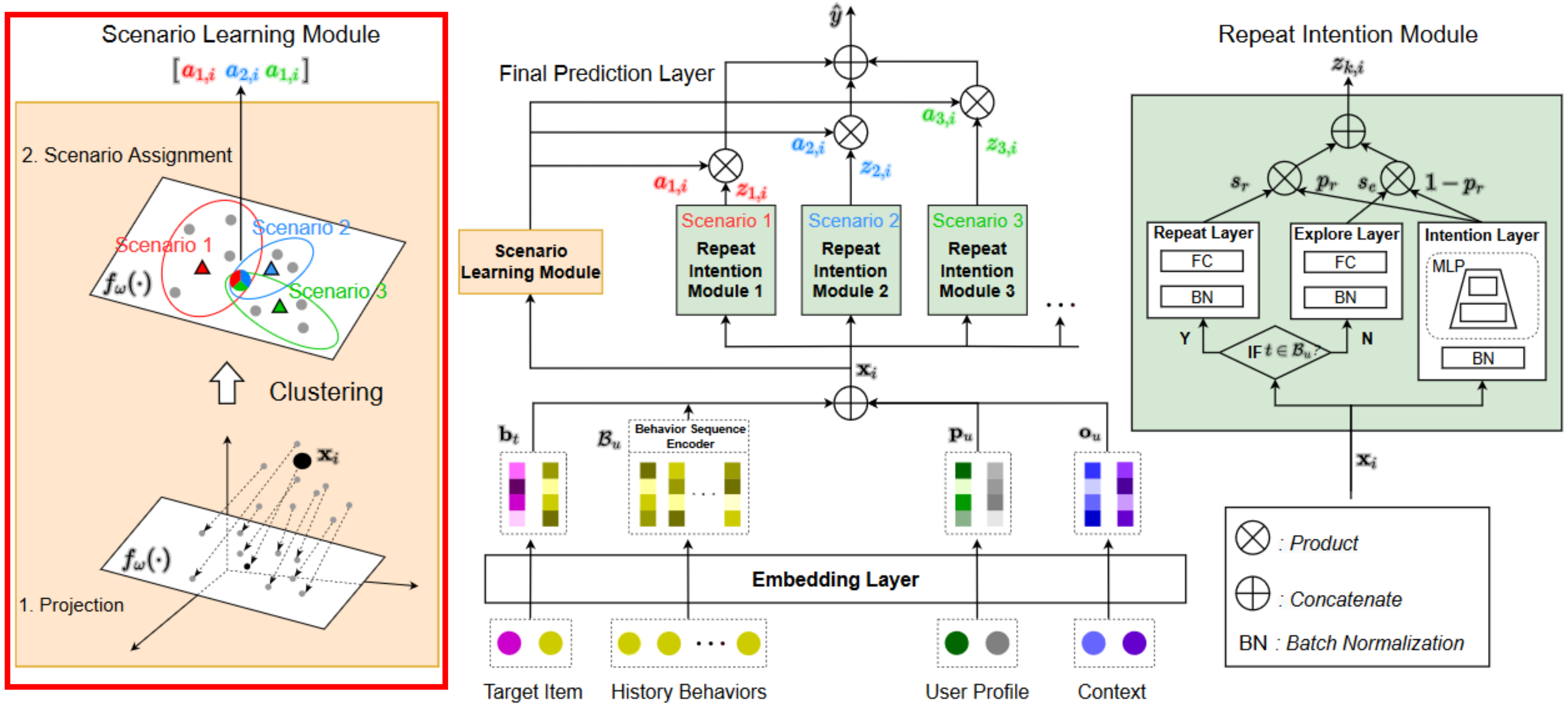
$$Q_s(c|x) = P_\omega(c|x) = \frac{\exp(-d(f_\omega(\mathbf{x}), \mathbf{c}))}{\sum_{\mathbf{c}'} \exp(-d(f_\omega(\mathbf{x}), \mathbf{c}'))} \rightarrow \{a_{1,i}, a_{2,i}, \ldots, a_{k,i}\}$$

Hard Assignment

Gumbel-Softmax trick

$$a_{k,i} = \frac{\exp((\log \pi_{k,i} + g_{k,i})/\tau)}{\sum_{k'=1}^{K} \exp((\log \pi_{k',i} + g_{k',i})/\tau)}$$

$$\pi_{k,i} = \frac{\exp(-d(f_\omega(\mathbf{x_i}), \mathbf{c_k}))}{\sum_{k'=1}^{K} \exp(-d(f_\omega(\mathbf{x_i}), \mathbf{c_{k'}}))}$$



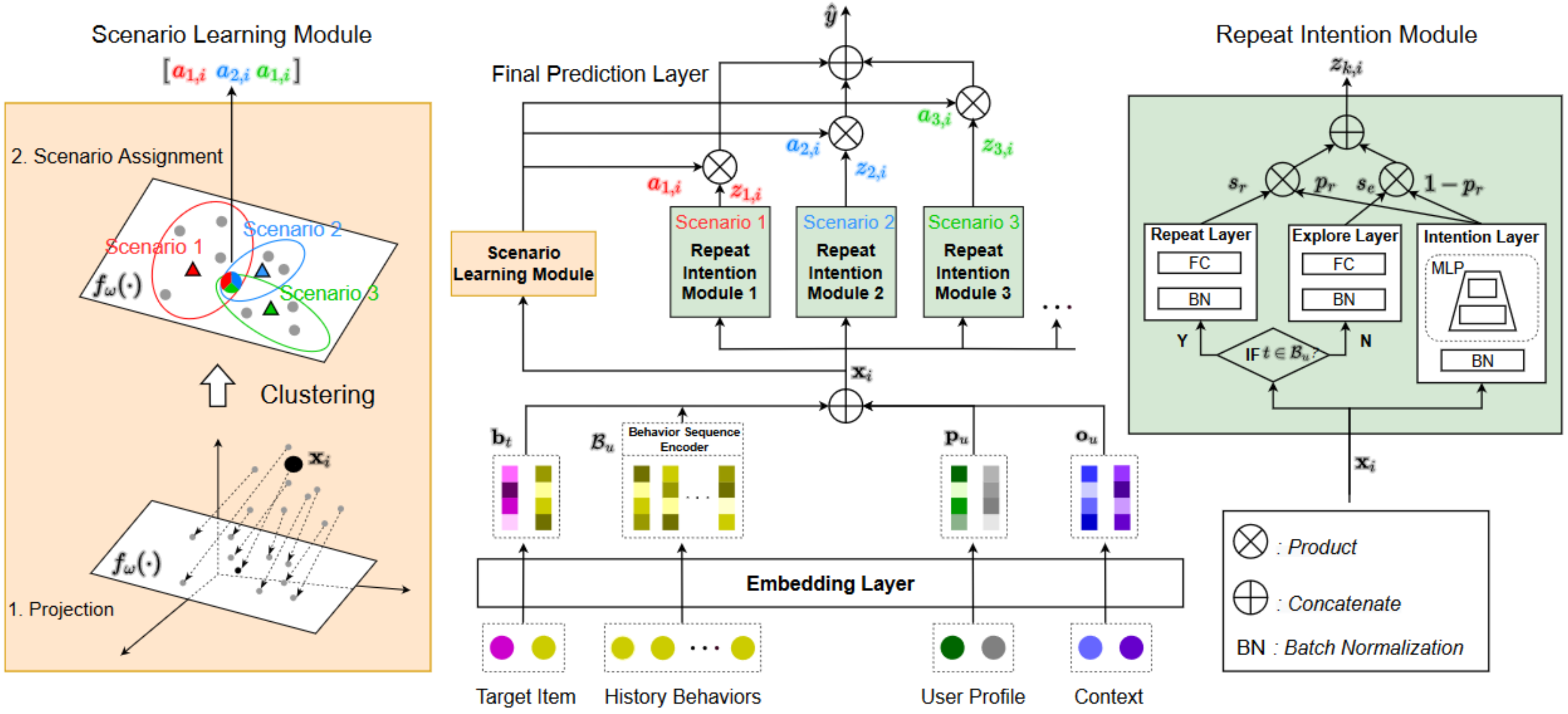
MUSENET Details

Given the ω, the objective is to minimize the distance expectation from each data sample to the corresponding scenario prototypes

$$\mathcal{L}_C(\Lambda, \Theta) = \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} a_{k,i} d(f_\omega(\mathbf{x}_i), \mathbf{c}_k)$$

Final Prediction

$$\hat{y}_i = \sigma(\sum_{k=1}^{K} a_{k,i} z_{k,i})$$



SASS

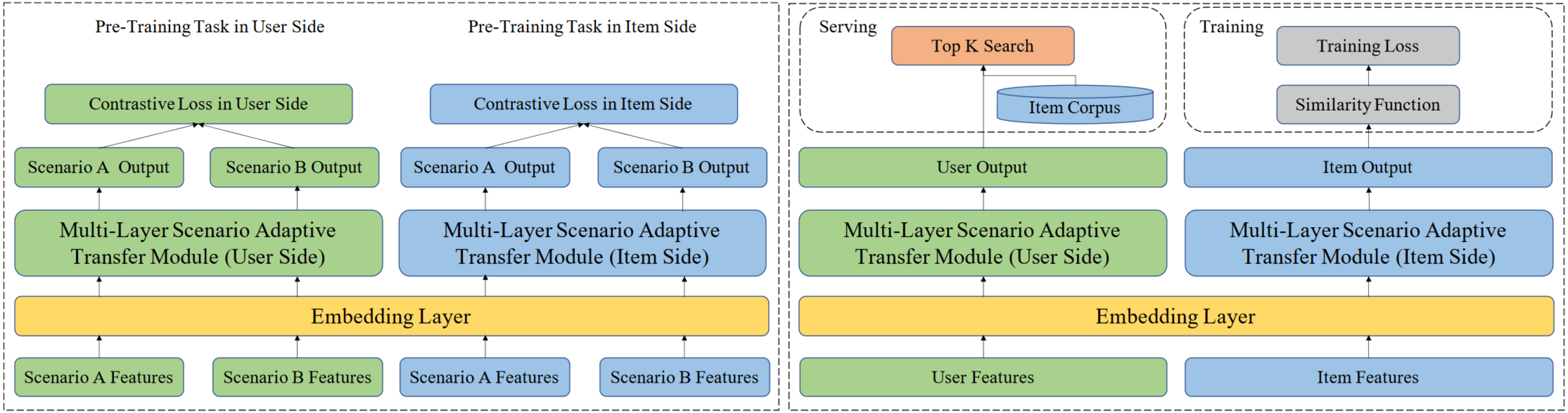
- Motivation
  - Lacking of fine-grained and decoupled information transfer controls among multiple scenarios
  - Insufficient exploitation of entire space samples
  - Item's multi-scenario representation disentanglement problem

A:Main Feed B:Immersive Feed C:Homepage Feed

- Methods
  - Multi-Layer Scenario Adaptive Transfer (ML-SAT) module
  - Two-stage training process including pre-training and fine-tune

SASS Details

Pre-training Stage and Fine-Tune Stage



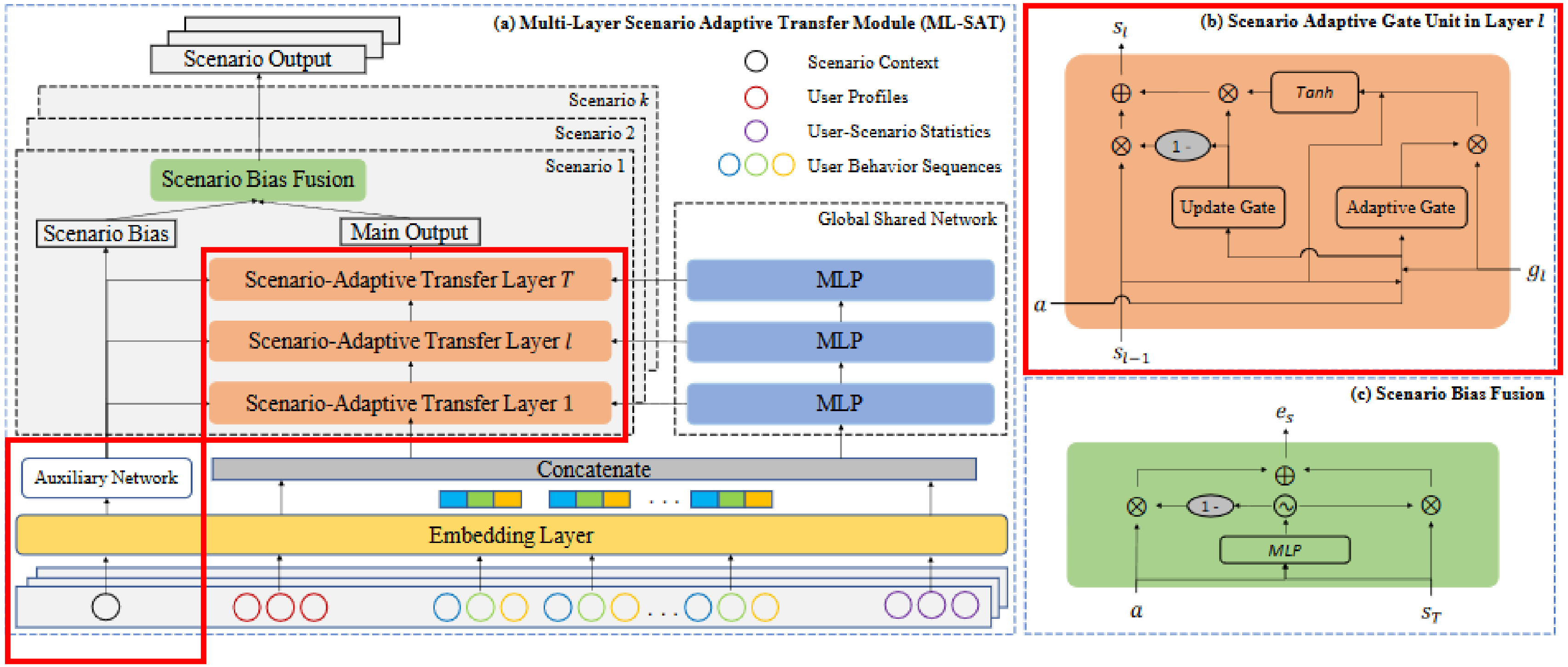
(a) Pre-Training Stage of SASS

(b) Fine-Tune Stage of SASS

$$\mathcal{L}_{ij} = -log \frac{\exp(sim(e_s^i, e_s^j)/\tau)}{\sum_{k=1,k\neq i}^{2N} \exp(sim(e_s^i, e_s^k)/\tau)}$$

SASS Details

Multi-Layer Scenario Adaptive Transfer Module



Scenario Modeling

$$a = f(W_a x_a + b_a)$$

Scenario-adaptive gate unit

$$r_l = \sigma(W_r^l[g_l, s_{l-1}] + W_{br}a)$$

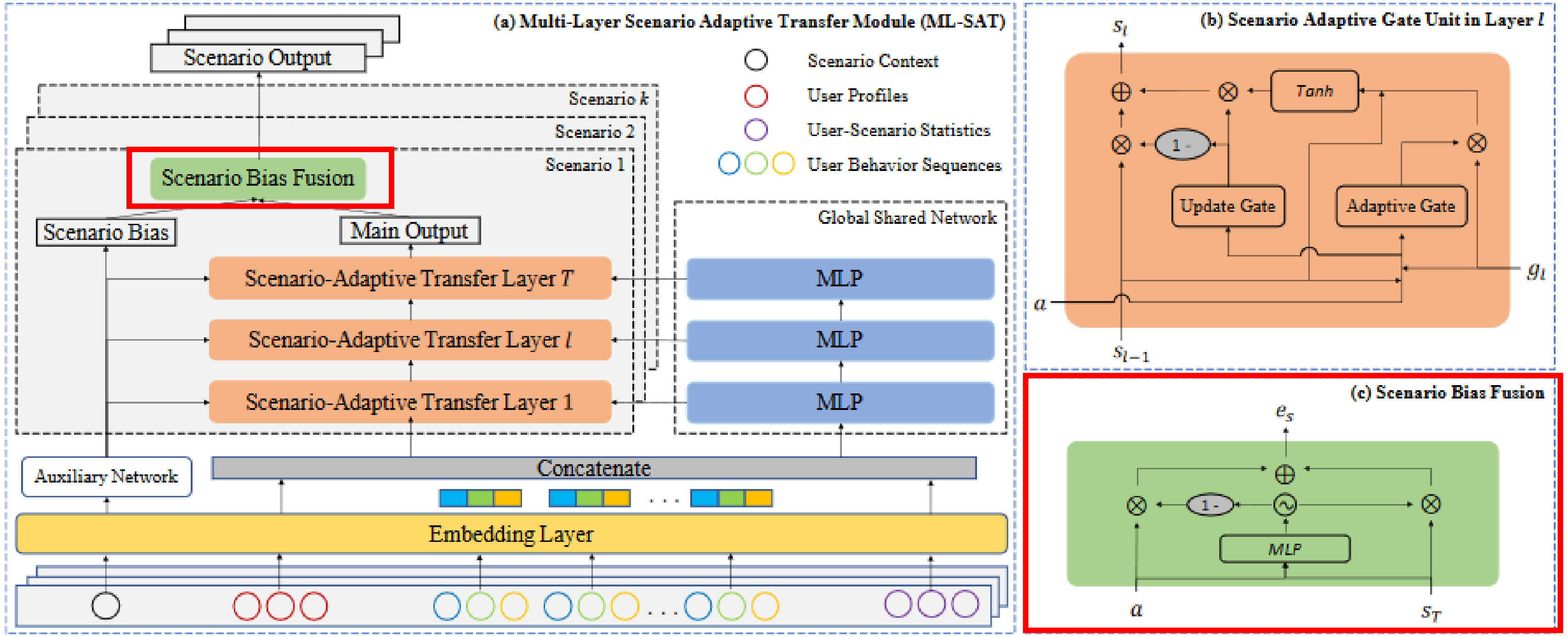
$$h_l = tanh(W_h^l[r_l \cdot g_l, s_{l-1}])$$

$$z_l = \sigma(W_z^l[g_l, s_{l-1}] + W_{bz}a)$$

$$s_l = (1 - z_l) \cdot s_{l-1} + z_l \cdot h_l$$

SASS Details

Multi-Layer Scenario Adaptive Transfer Module

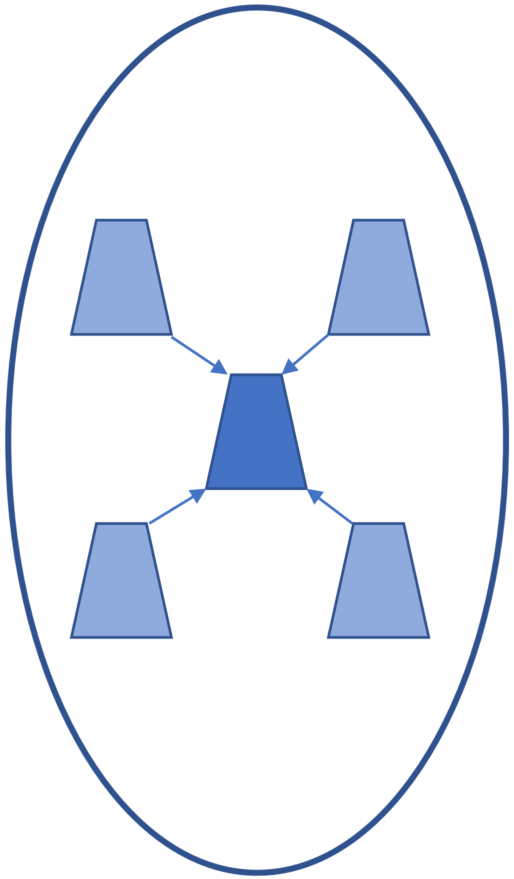


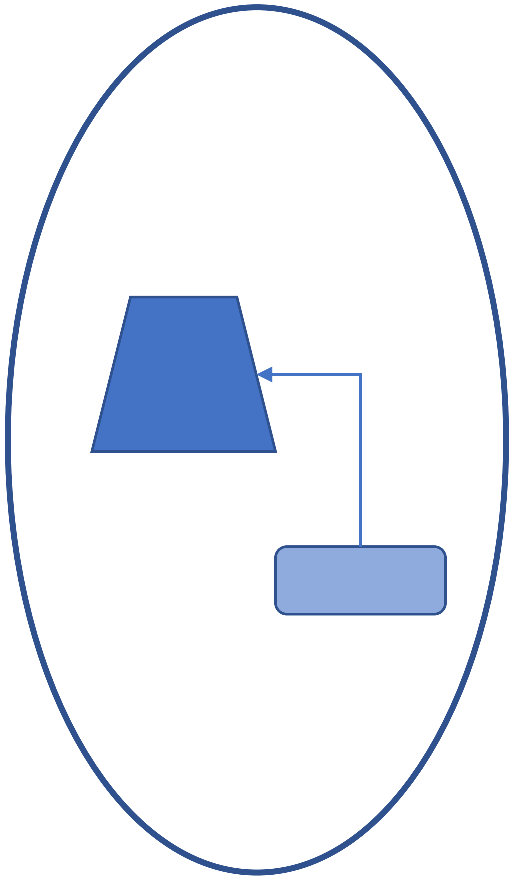
Scenario Bias Fusion

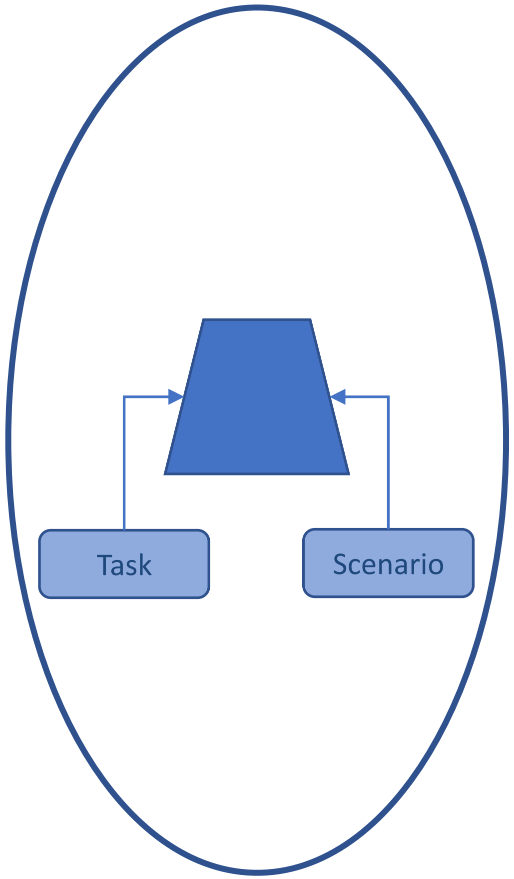
$$e_s = \alpha \cdot s_T + (1 - \alpha) \cdot a$$

$$\alpha = \sigma(W_0[s_T, a])$$

Table of Contents







Shared-specific network paradigm

$wL(E^{Merge}, \Theta, \Theta^{t}, (\Theta^{shared}, \Theta^{specific}))$

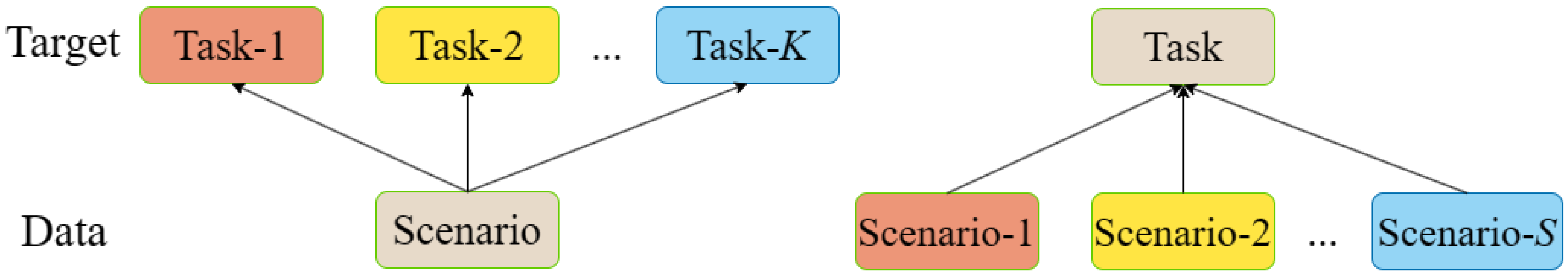
Dynamic weight

$wL(E^{Merge}, \Theta, \Theta^{t}, \Theta^{s})$

Multi-Scenario & Multi-Task

$wL(E^{Merge}, \Theta, \Theta^{t}, \Theta^{s}, \Theta^{T})$

Multi-Scenario & Multi-Task Studies



(a) MTR (b) MSR

MTMS

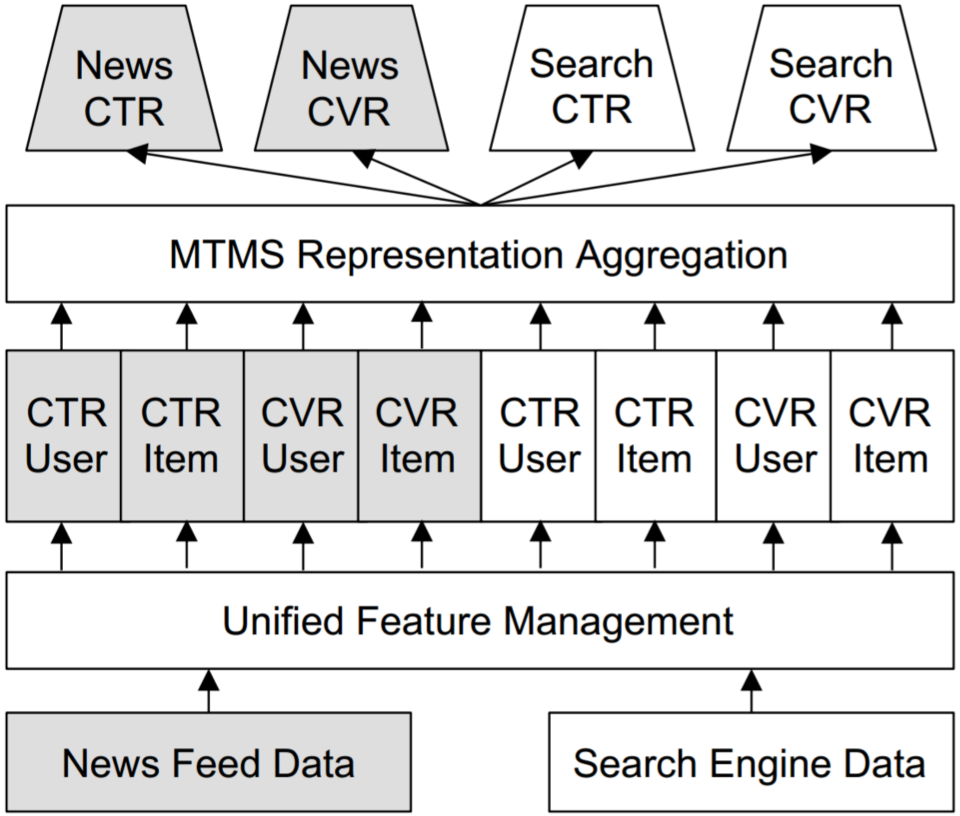
- Target
  - Develop a unified ranking model for multi-task and multi-scenario problem

- Methods
  - Independent/non-shared embeddings for each task and scene, new tasks or scenes could be added easily
  - A simplified network is chosen beyond the embedding layer, which largely improves the ranking efficiency for online service.

Multi-Task and Multi-Scene Unified Ranking Model for Online Advertising. Big Data 2021.

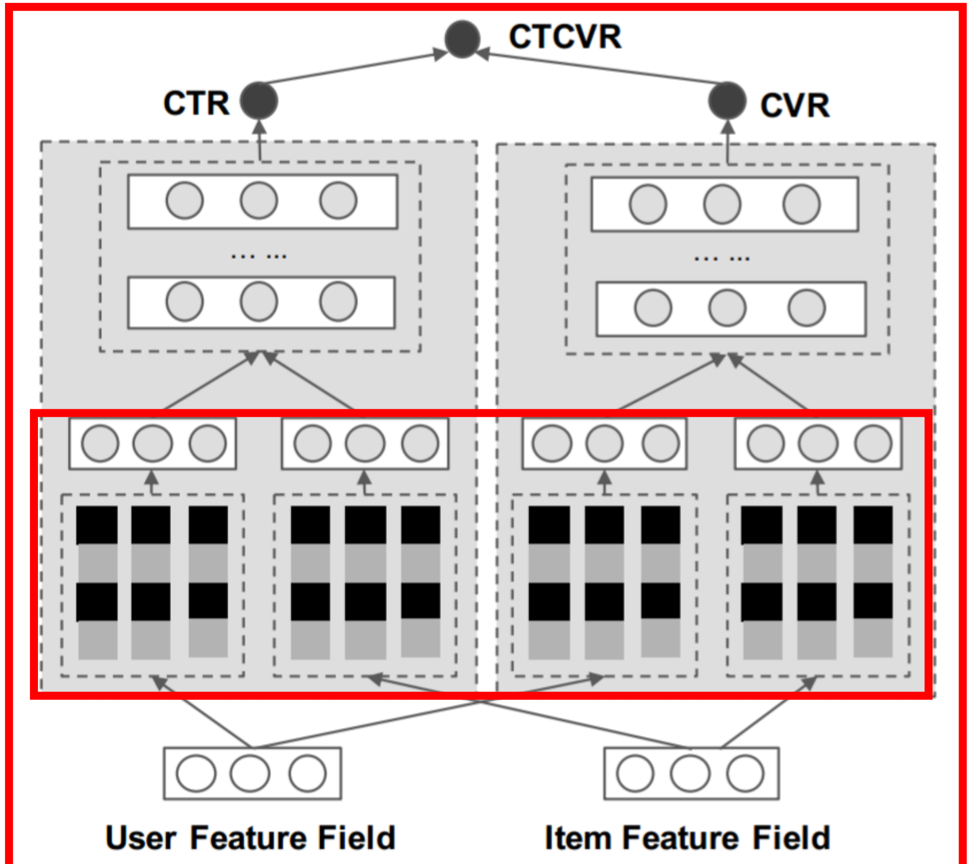
MTMS Details

Independent embeddings for every “task+scenario”
Aggregation of different components -> shared modeling
Loss function: sum of different tasks, -> performance not be hurt by auxiliary tasks (E.g. CTR)

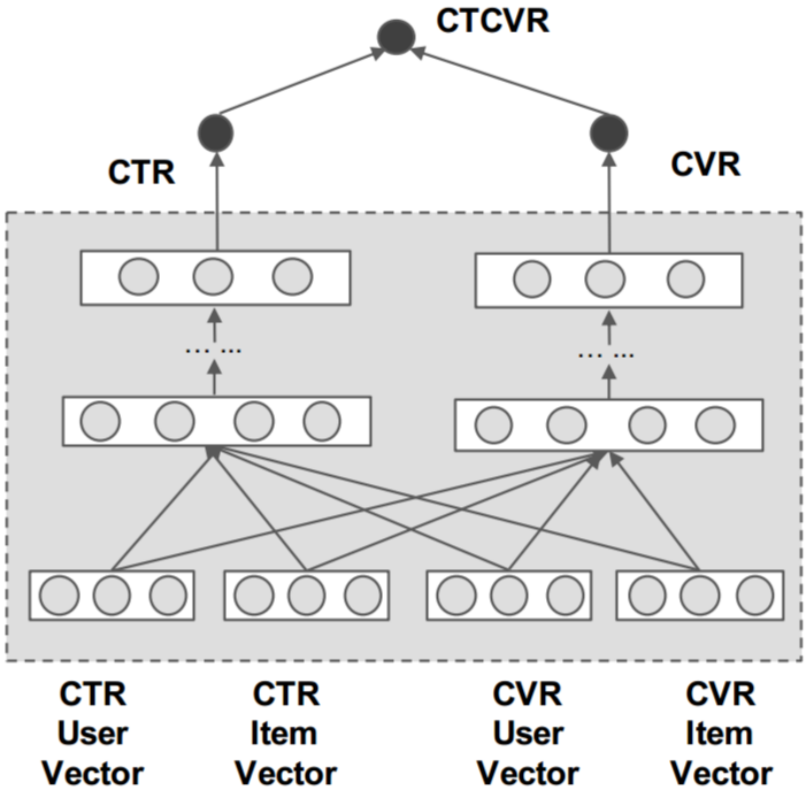


MTMS Details

➢ First step: embedding update, no shared information modeling



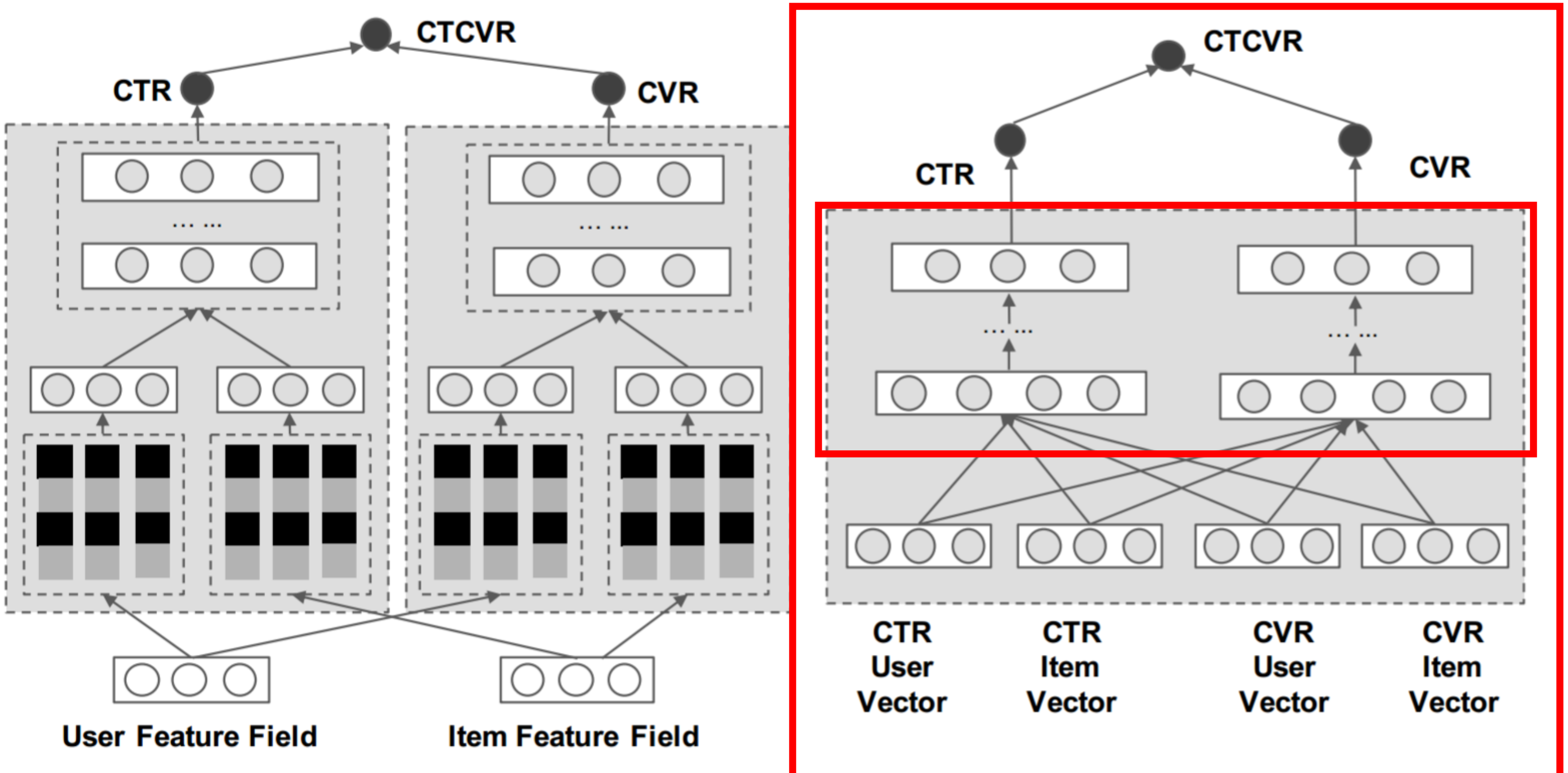
(b) MTMS: embedding update



(c) MTMS: network fine tune

MTMS Details

- Second step: network fine tune. Embedding is fixed. DNN has more fields for inputs



(b) MTMS: embedding update

(c) MTMS: network fine tune

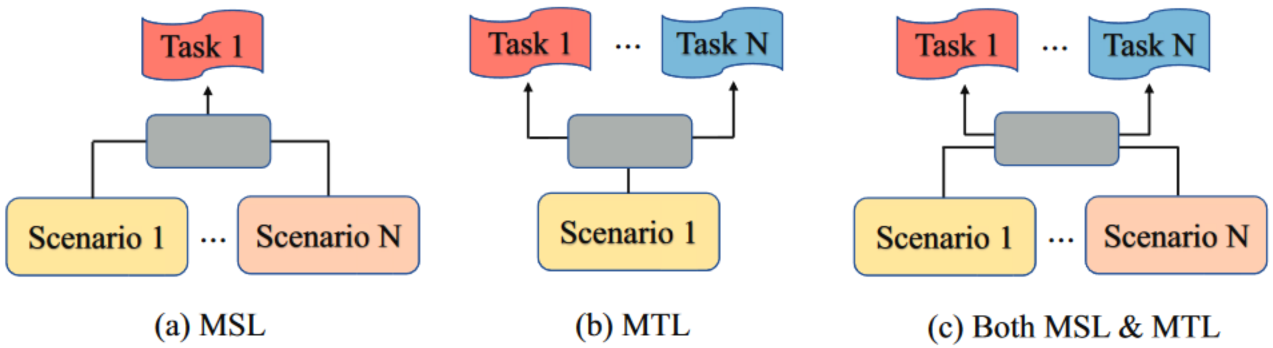
AESM^2

➢ Target

- Develop a unified framework that could realize both MSL and MTL requirements

➢ Methods

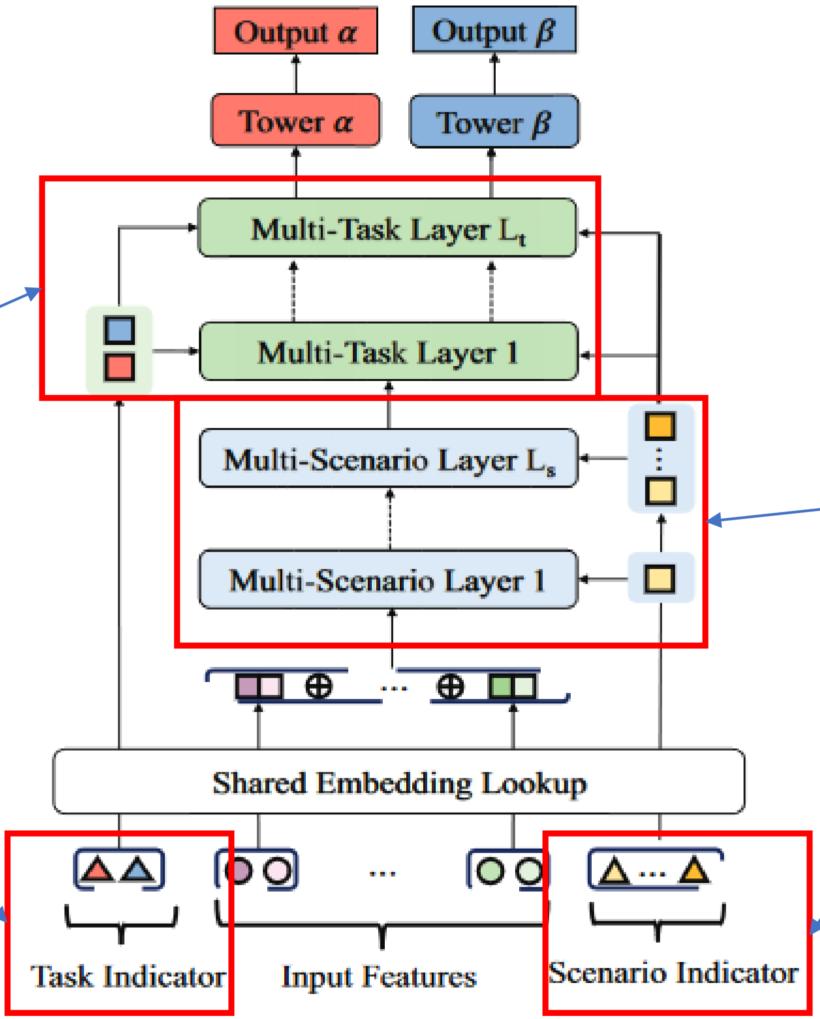
- Propose AESM^2, a flexible hierarchical structure where the multi-task layers are stacked over the multi-scenario layers
- General expert selection algorithm



Automatic Expert Selection for Multi-Scenario and Multi-Task Search. SIGIR 2022.

AESM²

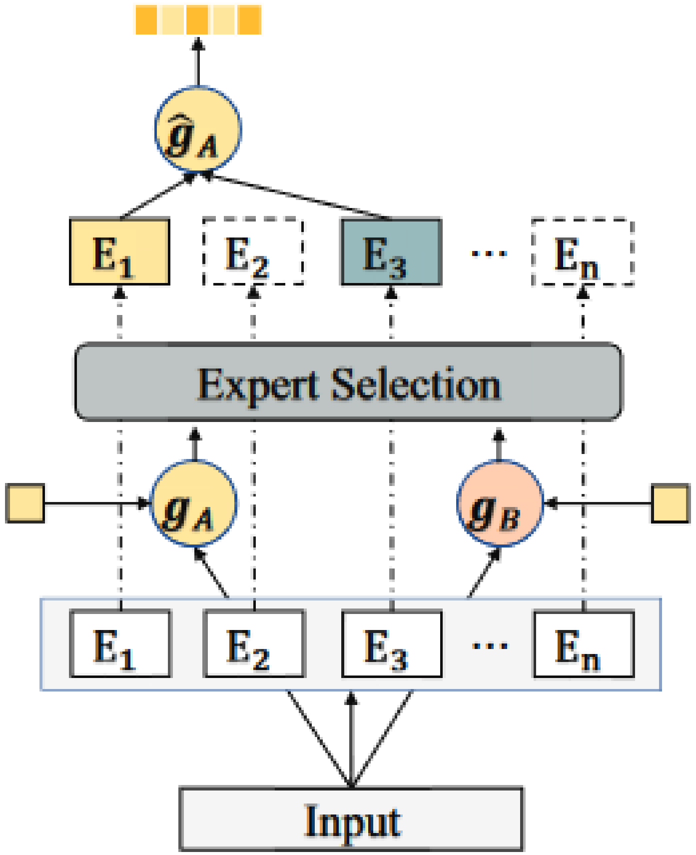
Task-related



Scenario-related

AESM^2

Multi-Scenario Layer



(b) Multi-Scenario Layer

- Input x, scenario embedding s, Gaussian noise n_j, learnable parameter s_j, m scenarios/gates. For every expert:

$$\mathbf{G} = [\mathbf{g}_1, \cdots, \mathbf{g}_m]$$
$$\mathbf{g}_j = \mathbf{S}_j[\mathbf{x}, \mathbf{s}] + \eta_j$$
$$\tilde{\mathbf{G}} = softmax(\mathbf{G})$$

- Expert selection

$$\left.\begin{aligned} \mathcal{E}_{sp} &= TopK(h_1^p, \cdots, h_n^p) \\ h_k^p &= -KL(\mathbf{p}_j, \tilde{\mathbf{G}}[k,:]) \end{aligned}\right\} \text{Specific}$$

$$\left.\begin{aligned} \mathcal{E}_{sh} &= TopK(h_1^q, \cdots, h_n^q) \\ h_k^q &= -KL(\mathbf{q}_j, \tilde{\mathbf{G}}[k,:]) \end{aligned}\right\} \text{Shared}$$

$$\mathbf{p}_j (e.g., [1, \cdots, 0])$$
$$\mathbf{q}_j = [1/m, \cdots, 1/m]$$

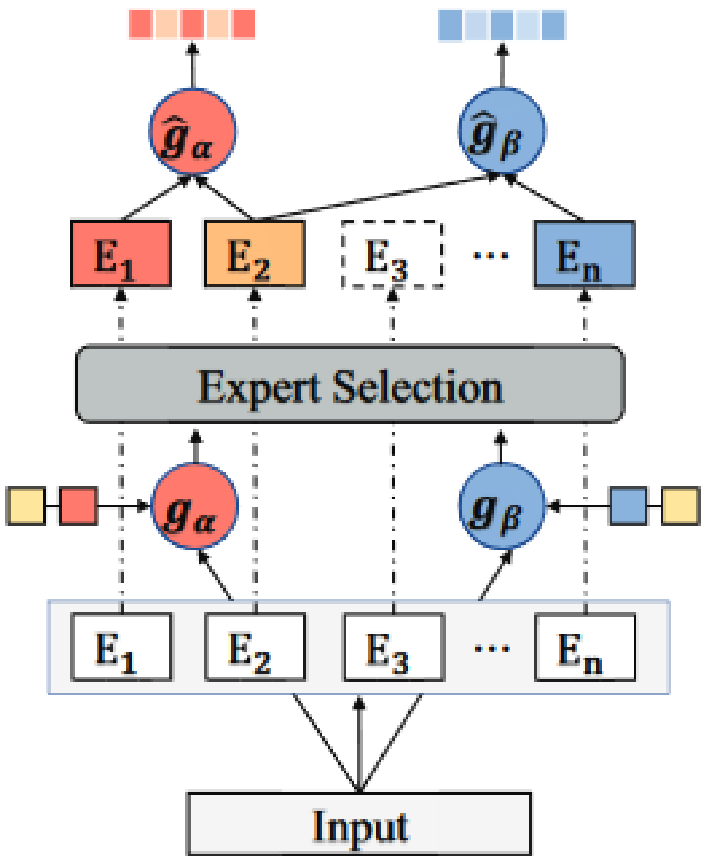
- Expert aggregation:
  - (k-th expert, j-th scenario)

$$\hat{\mathbf{g}}_j[k] = \begin{cases} \mathbf{g}_j[k], & if \quad k \in \mathcal{E}_{sh} \cup \mathcal{E}_{sp} \\ -\infty, & else \end{cases}$$

$$\mathbf{z}_j = ScenarioLayer(\mathbf{x}, \mathbf{s}_j) = MMoE(\mathbf{x}, \hat{\mathbf{g}}_j)$$

AESM^2

Multi-Task Layer



- Input x, scenario embedding s, task embedding t_k, Gaussian noise n_j, learnable parameter T_k, the gating scalar g_k for k-th task:

$$\mathbf{g}_k = \mathbf{T}_k[\mathbf{x}, \mathbf{s}, \mathbf{t}_k] + \boldsymbol{\eta}_k$$

$$\mathbf{z}_k = TaskLayer(\mathbf{z}_j, \mathbf{t}_k) = MMoE(\mathbf{z_j}, \hat{\mathbf{g}}_k)$$

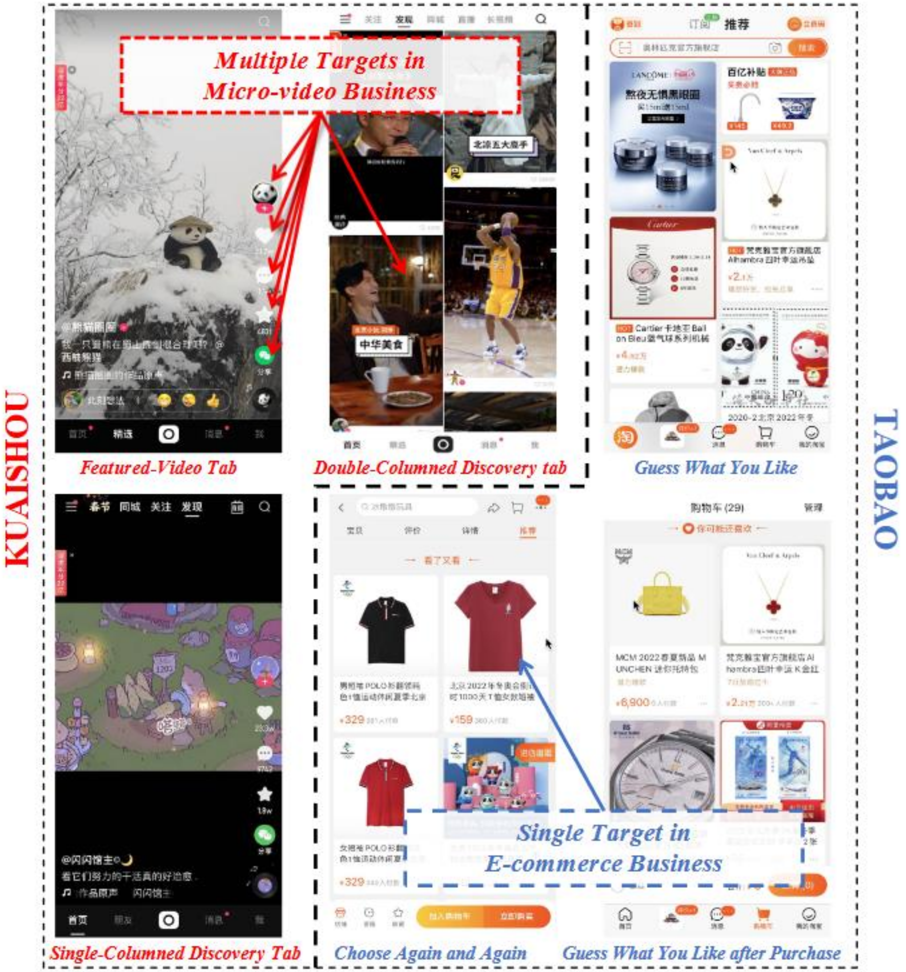
- Output layer

$$\hat{y}_k = \sigma(MLP(\mathbf{z}_k))$$

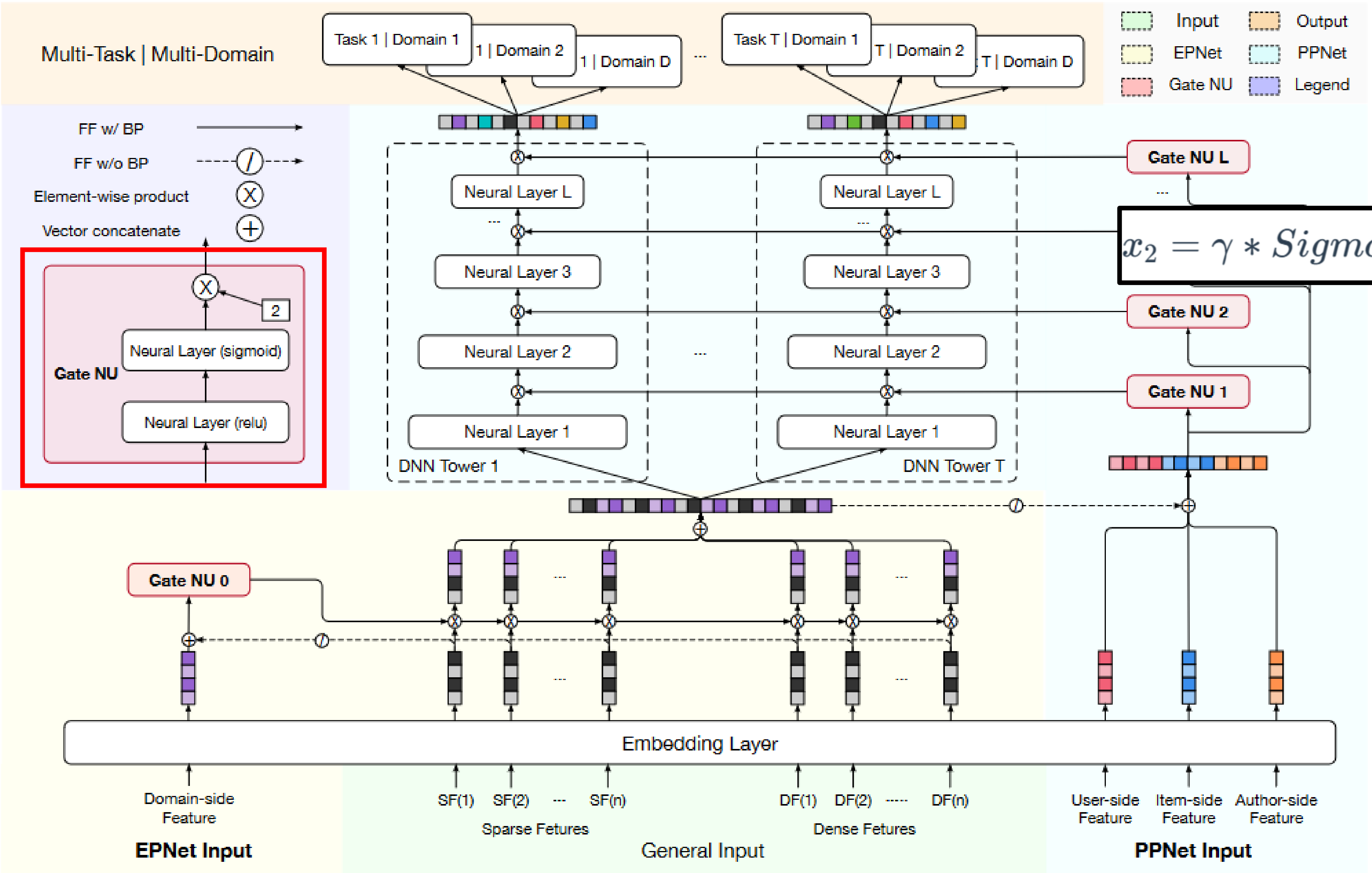
PEPNet

- Motivation
  - The imperfectly double seesaw phenomenon
  - More accurate personalization estimates can alleviate the imperfectly double seesaw problem

- Target
  - Jointly model multi-domain and multi-task
  - an efficient, low-cost deployment and plug-and-play method that can be injected in any network.



PEPNet Details

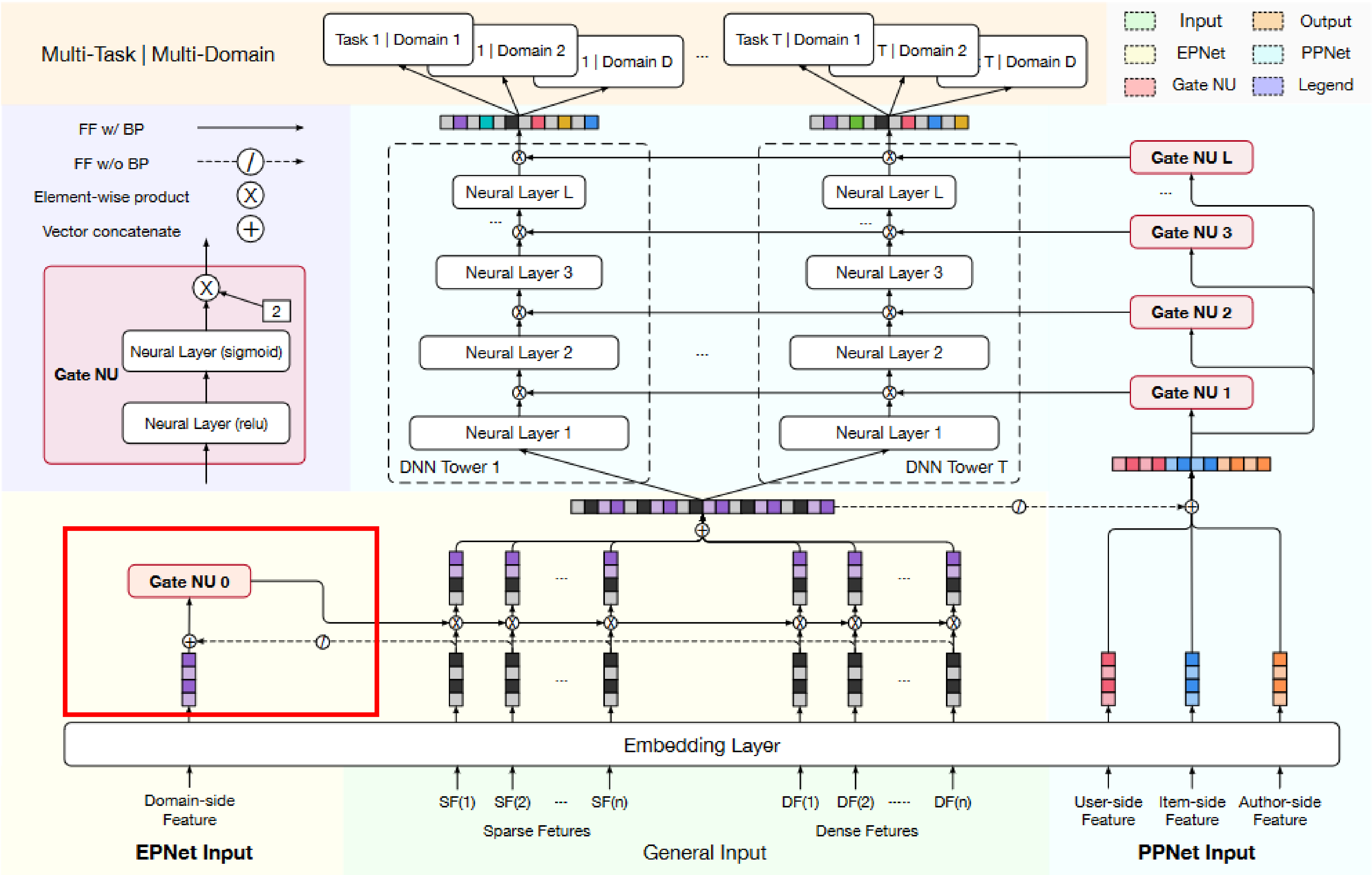


Gate Neural Unit (Gate NU)

$$x_1 = Relu\left(x^{(0)}\mathbf{W}^{(0)} + b^{(0)}\right)$$

$$\boxed{x_2 = \gamma * Sigmoid\left(x^{(1)}\mathbf{W}^{(1)} + b^{(1)}\right), x_2 \in [0, \gamma]}$$

PEPNet Details



EPNet

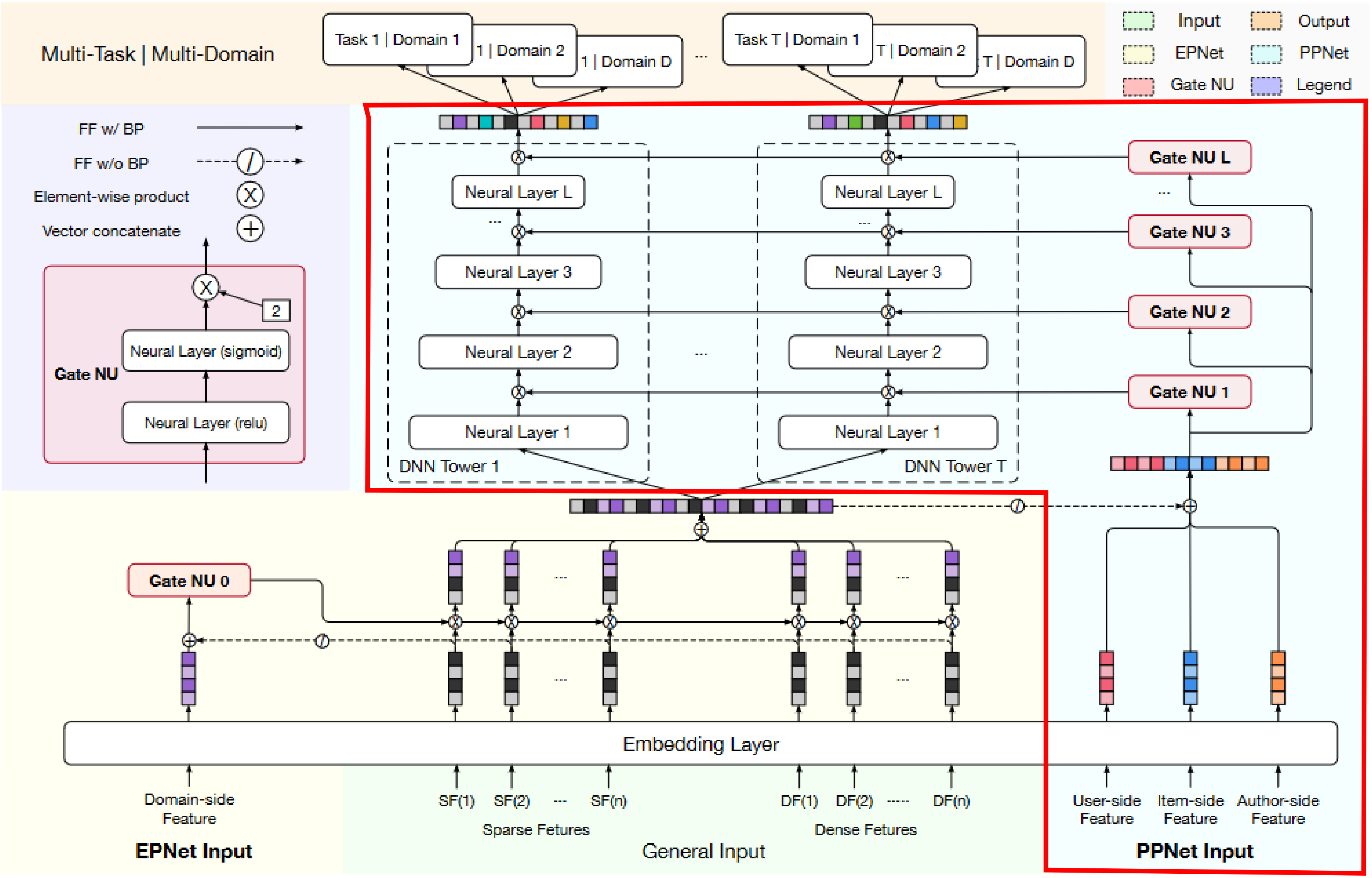
$$\mathbf{E} = E(\mathcal{F}_S) \oplus E(\mathcal{F}_D)$$

Embeddings of sparse features and dense features

$$\delta_{domain} = \mathrm{U}_{ep}(E(\mathcal{F}_d) \oplus (\oslash(\mathbf{E})))$$

$$\mathbf{O}_{ep} = \delta_{domain} \otimes \mathbf{E}$$

PEPNet Details



PPNet

$$0_{prior} = E(uf) \oplus E(if) \oplus E(af)$$

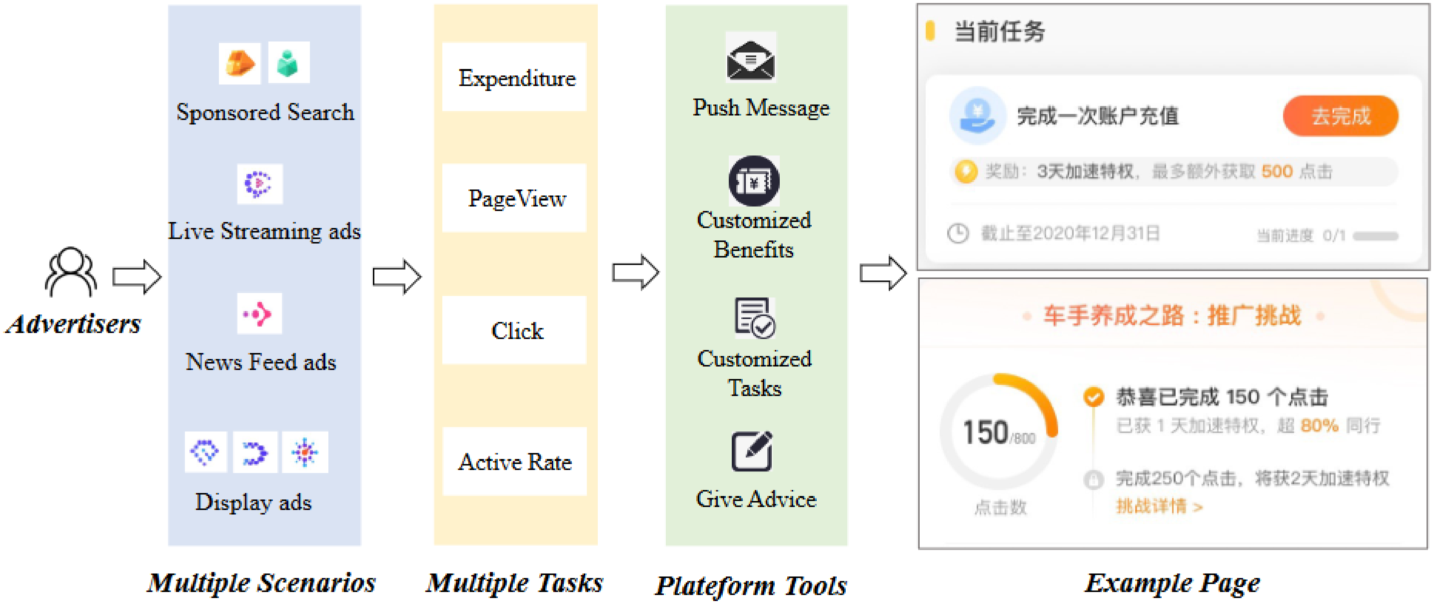
$$\delta_{task} = \mathbf{U}_{pp}(\mathbf{O}_{prior} \oplus (\oslash(\mathbf{O}_{ep})))$$

$$\mathbf{O}_{pp}^{(l)} = \delta_{task}^{(l)} \otimes \mathbf{H}^{(l)},$$

$$\mathbf{H}^{(l+1)} = f(\mathbf{O}_{pp}^{(l)} \mathbf{W}^{(l)} + b^{(l)}), l \in \{1, ..., L\}$$

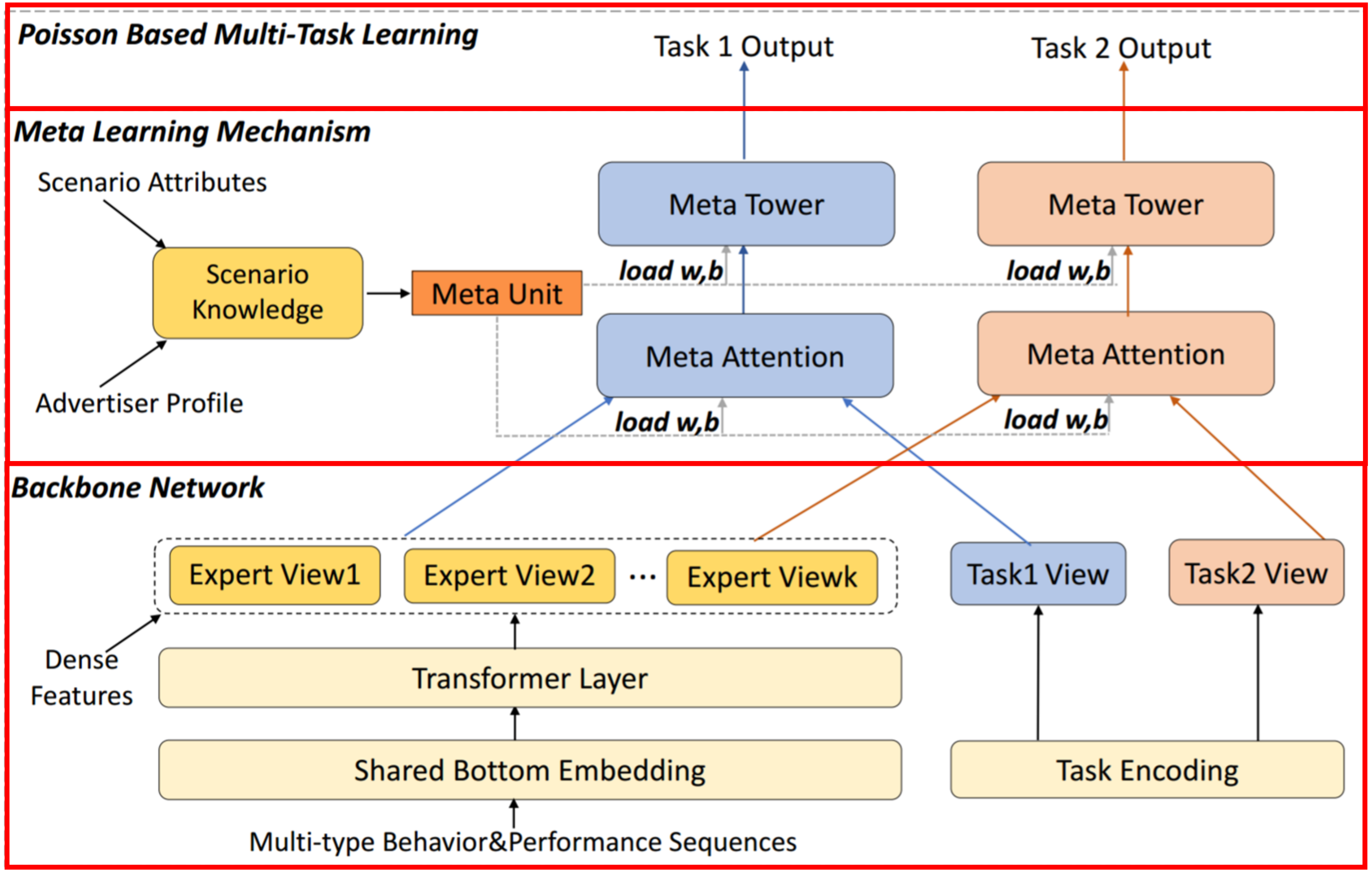
M2M

- Motivation
  - Less attention has been drawn to advertisers
  - Major e-commerce platforms provide multiple marketing scenarios.
- Methods
  - Meta unit
  - Meta attention module
  - Meta tower module

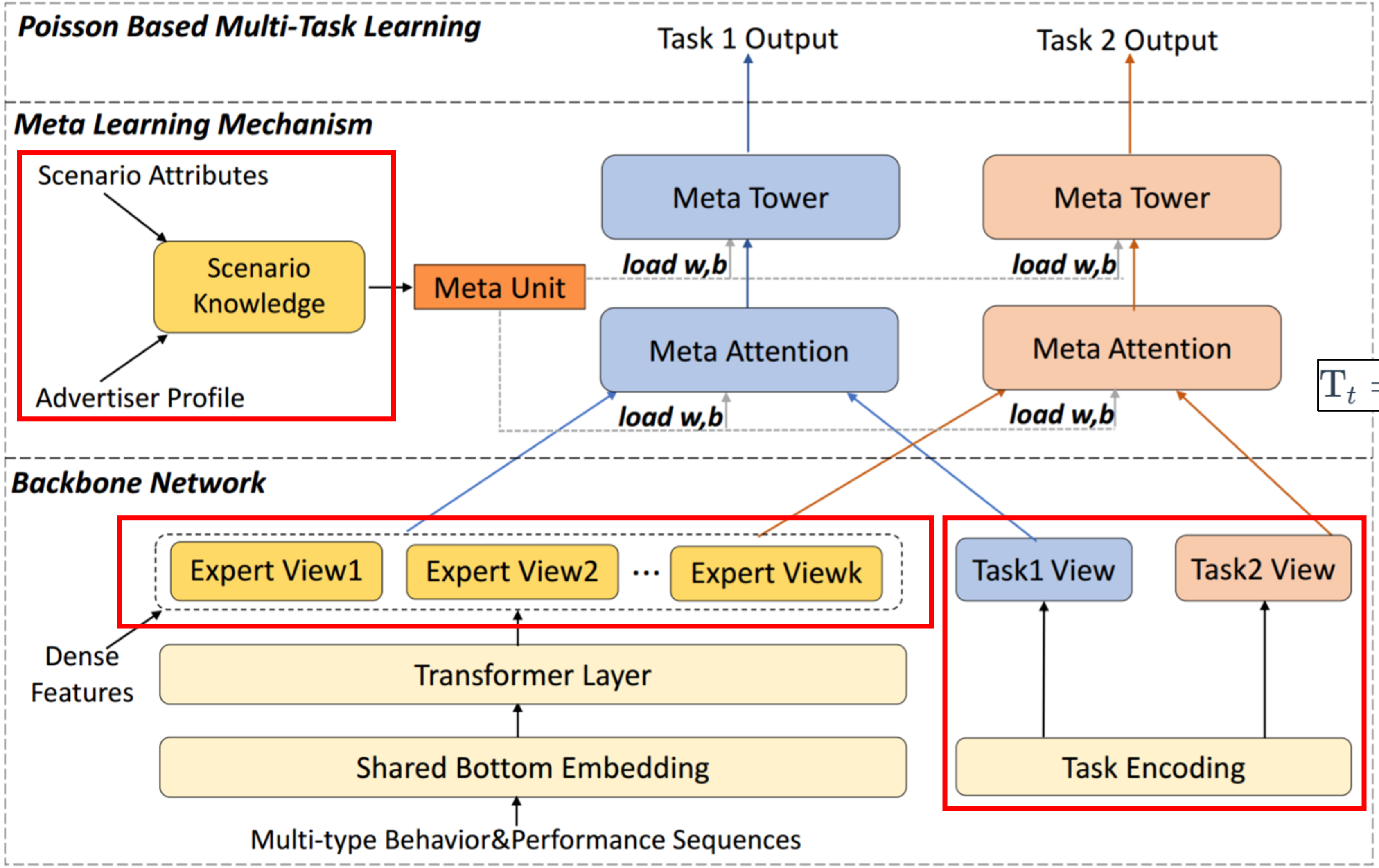


Leaving No One Behind: A Multi-Scenario Multi-Task Meta Learning Approach for Advertiser Modeling. WSDM 2022.

M2M Overview



M2M Details



Backbone Network

Expert View Representation

$$\mathrm{E_i} = f_{MLP}(\mathbf{F}), \forall i \in 1, 2, .., k$$

F is the output of transformer layer

Task View Representation

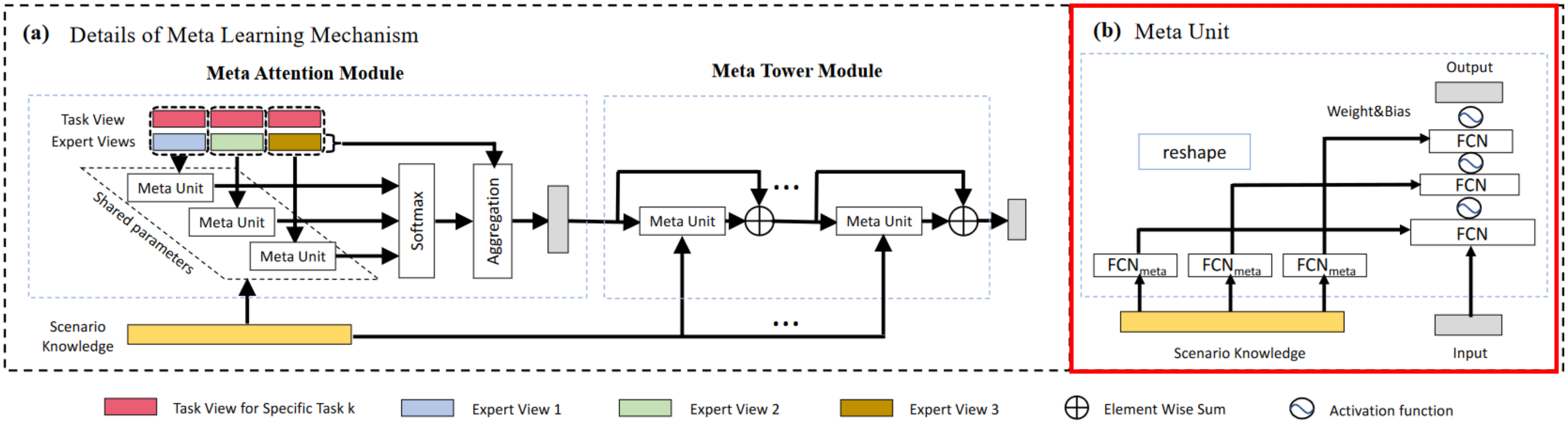
$$\mathrm{T}_t = f_{MLP}(Embedding(t)), \forall t \in 1, 2, .., m$$

Scenario Knowledge Representation

$$\tilde{\mathrm{S}} = f_{MLP}(\mathrm{S}, \Lambda)$$

Meta Unit

$$h_{output} = h^K = Meta(h_{input})$$



M2M Details

Meta Attention Module

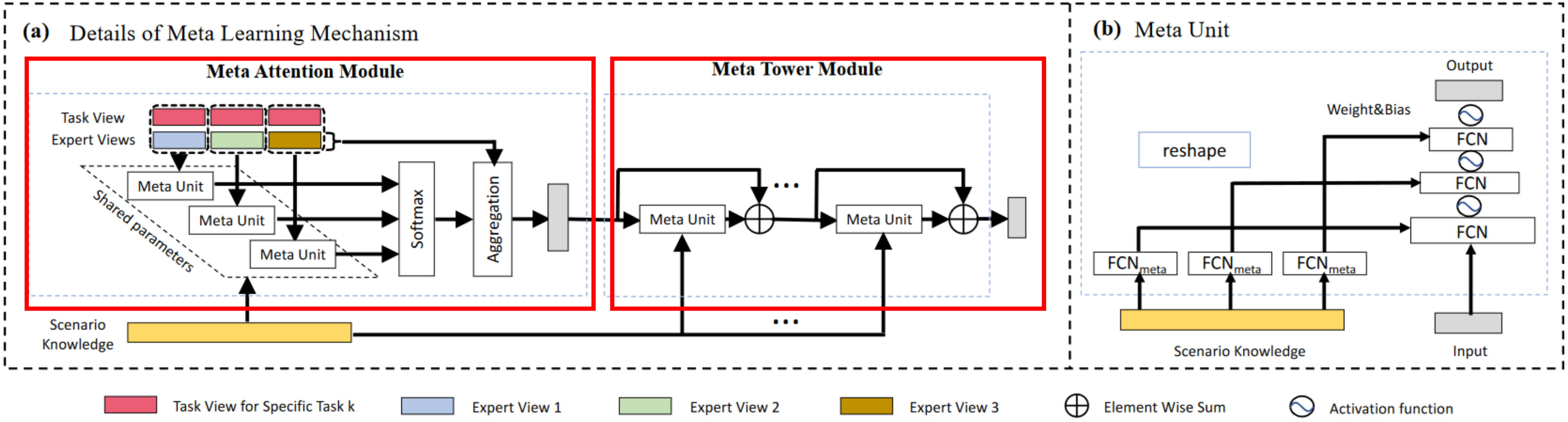
$$a_{t_i} = \mathrm{v}^T Meta_t([\mathrm{E_i} \parallel \mathrm{T}_t])$$

$$\alpha_{t_i} = \frac{exp(a_{t_i})}{\sum_{j=1}^{M} exp(a_{t_j})}, \qquad \mathrm{R}_t = \sum_{i=1}^{k} \alpha_{t_i} \mathrm{E}_i$$

Meta Tower Module

$$\mathrm{L}_t^{(0)} = \mathrm{R}_t$$

$$\mathrm{L}_t^{(j)} = \sigma(Meta^{(j-1)}(\mathrm{L}_t^{(j-1)}) + \mathrm{L}_t^{(j-1)}), \forall j \in 1, 2, .., L$$



Summary

➤ **Multi-Scenario Recommendation**

Model	Setting	Methods
STAR	Multi-Scenario	Shared-Specific
SAR-Net	Multi-Scenario	Shared-Specific; Experts
ADI	Multi-Scenario	Shared-Specific
MUSENET	Multi-Scenario	Dynamic Weight
SASS	Multi-Scenario	Dynamic Weight
MTMS	Multi-Scenario & Multi-Task	Two-stage fine-tune
PEPNet	Multi-Scenario & Multi-Task	Dynamic Weight
M2M	Multi-Scenario & Multi-Task	Dynamic Weight; Experts

Future Directions

➢ Multi-Scenario Recommendation

Topic	Challenge & future direction
LLM-based multi-scenario & multi-task modeling	• Design specific prompts for each scenario or tasks • Take the texts to bridge different scenarios or tasks
Robustness	• Scenarios with different available information (multimodal …)
Privacy	• Data need to be shared between different scenarios to build a unified model. Methods to protect user privacy should be proposed.
Fairness and Bias	• The issue of fairness in recommendation scenarios.

Coffee Break

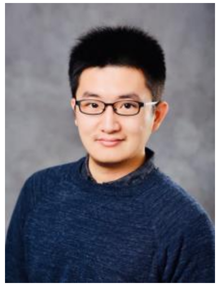
Huawei Noah’s Ark Lab

IJCAI23 Huawei Noah’s Ark Lab Chat Group

Xiangyu Zhao
City University of Hong Kong

Agenda

Introduction



Xiangyu Zhao

Preliminary

Yichao Wang

Multi-task Recommendation

Yuhao Wang

Multi-scenario recommendation

MTR+MSR

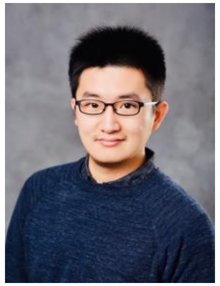
Pengyue Jia

More Joint-learning Methods

Jingtong Gao

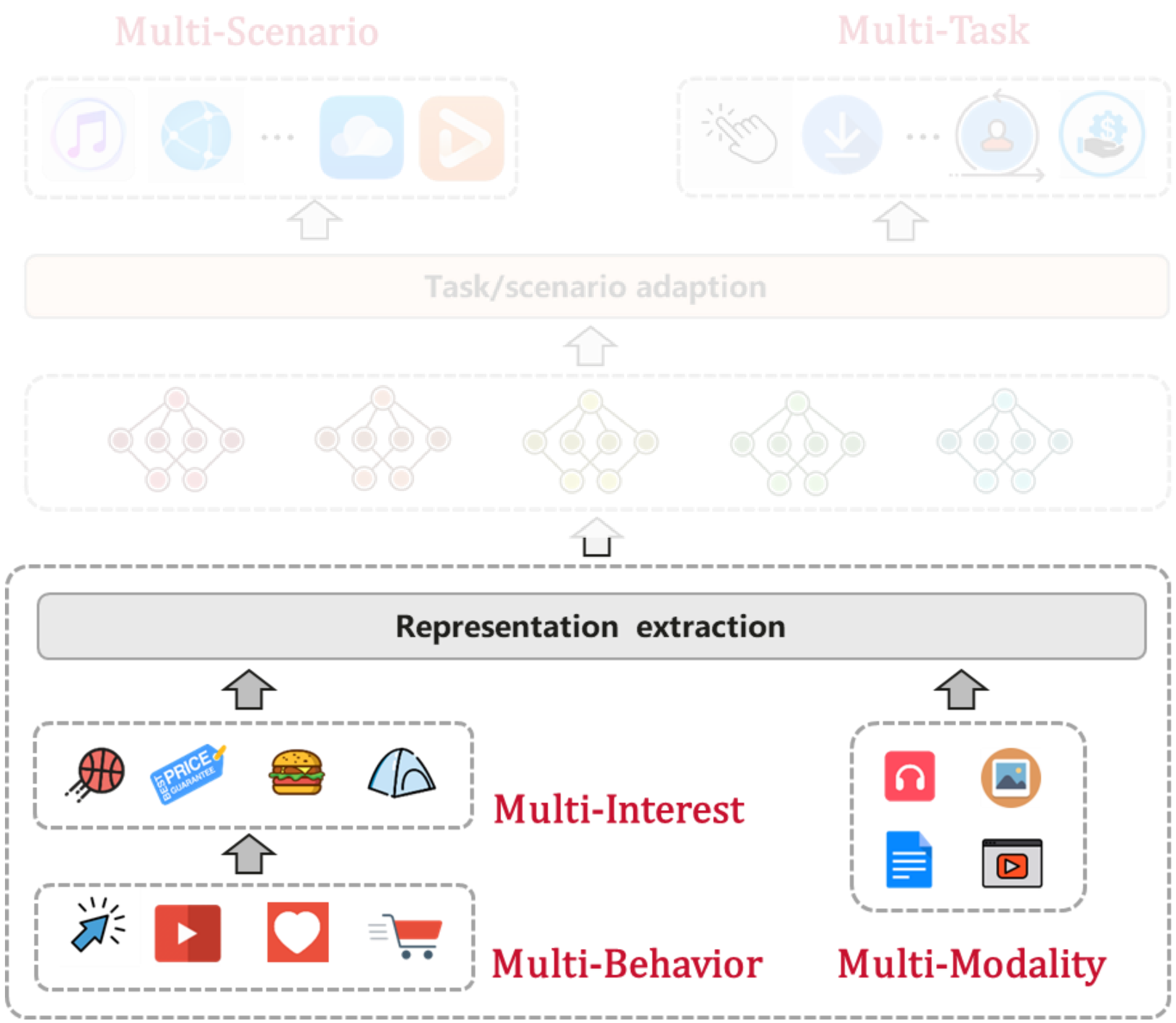
Conclusion

Future Work



Xiangyu Zhao

More Joint-Learning Methods



$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$ Joint Modeling

$E^{Merge} = U(E, E^{Ext}, E^m)$

$E^{Ext} = F(H^{UB})$ **Multi-Interest**

$H^{UB} = G(H_1, H_2, \ldots, H_N)$ **Multi-Behavior**

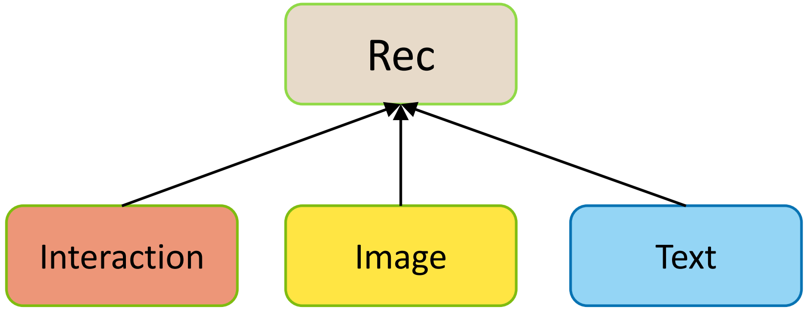
$E^m = M(E^{txt}, E^v, \ldots, E^p)$ **Multi-Modality**

$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$ Multi-Scenario

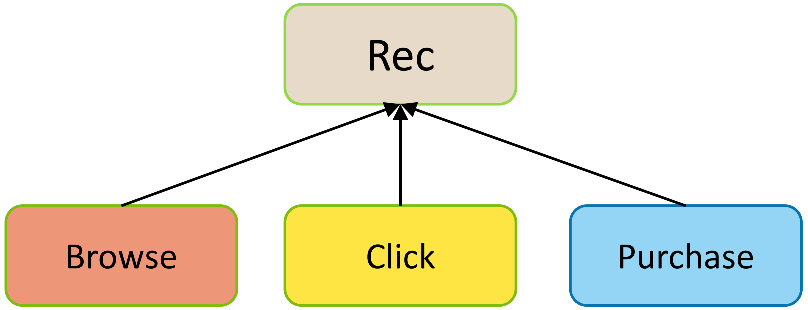
$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$ Multi-Task

More Joint-Learning Methods

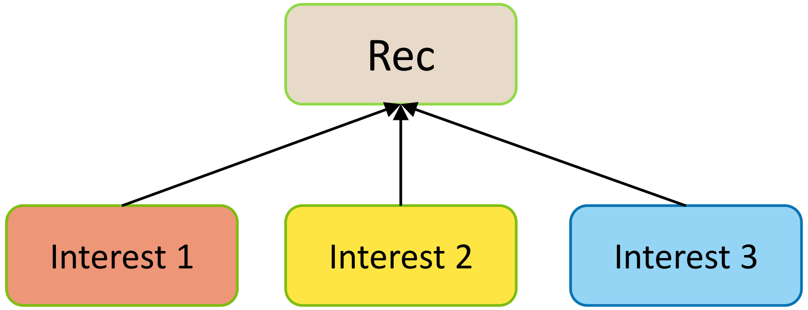
- - Multi-modal recommendation
  - Multi-behavior recommendation
  - Multi-interest recommendation
  - Large language model-based recommendation



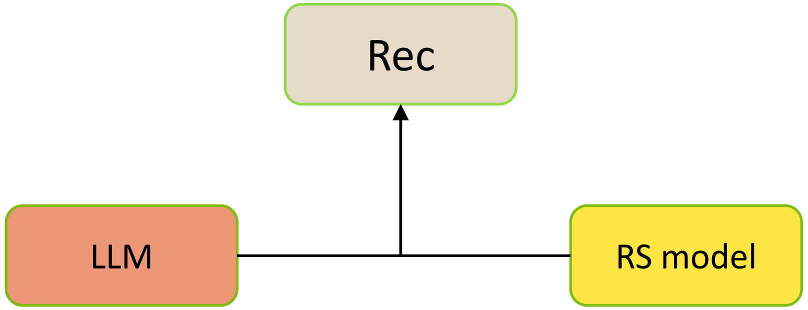
Multi-modal recommendation



Multi-behavior recommendation

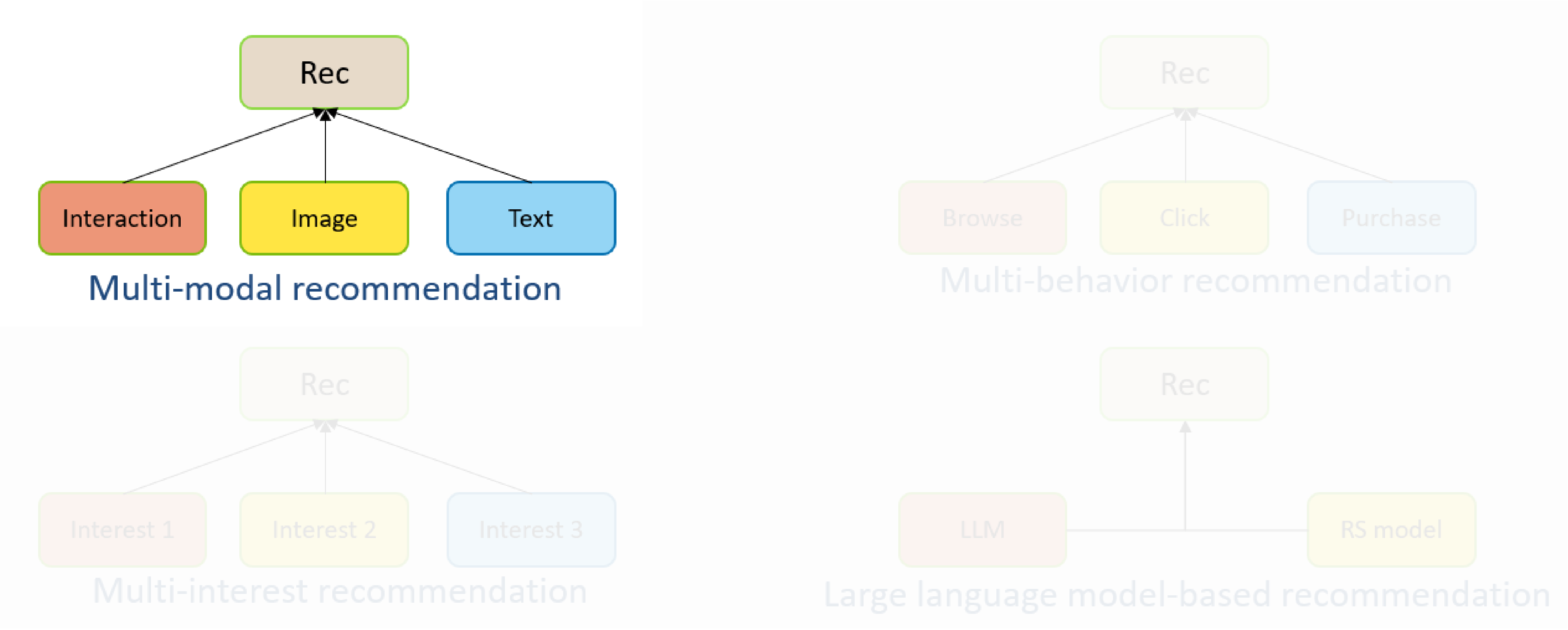


Multi-interest recommendation



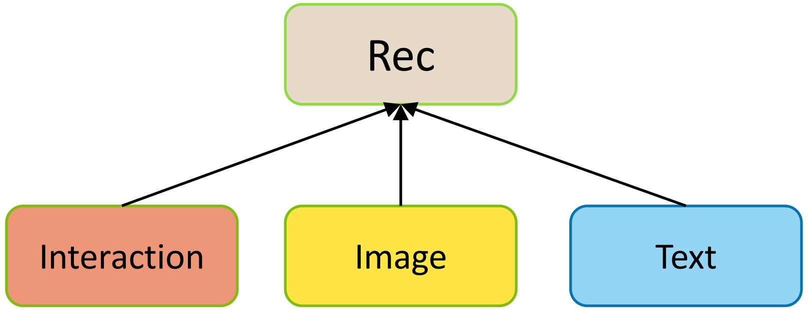
Large language model-based recommendation

Multimodal Recommender Systems (MRS)



Multimodal Recommender Systems (MRS)

- ➢ Using various types of information generated by multimedia applications and services to enhance recommender systems' performance
- ➢ Making use of multimodal features simultaneously, such as image, audio, and text
- ➢ Challenge:
  - Acquisition of different representations -> Modality Encoder
  - Fusion of different modality features -> Feature Interaction
  - Acquisition of representations under the data-sparse condition -> Feature Enhancement
  - Effectiveness and efficiency improvement -> Model Optimization



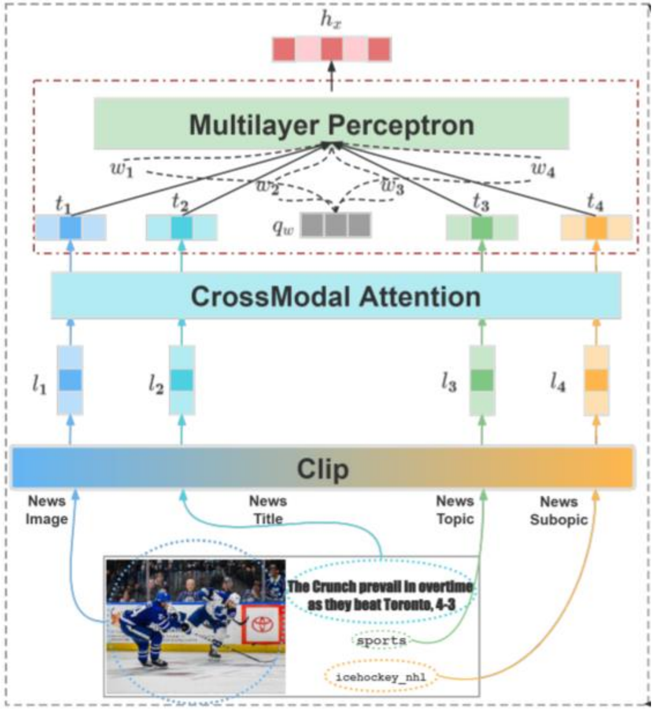
Multi-modal recommendation

Multimodal Recommender Systems: A Survey. arXiv preprint 2023.

Modality Encoder

- Encoding different multimodal features
- Commonly used:
  - Visual: CNN-based, ViT / Transformer-based
  - Textual: Word2Vec, CNN-based, RNN-based, Transformer-based
  - Others: E.g., converting acoustic and video data into text or visual information

Modality	Category
Visual Encoder	CNN ResNet Transformer
Textual Encoder	Word2vec RNN CNN Sentence-transformer Bert
Other Modality Encoder	Published Feature



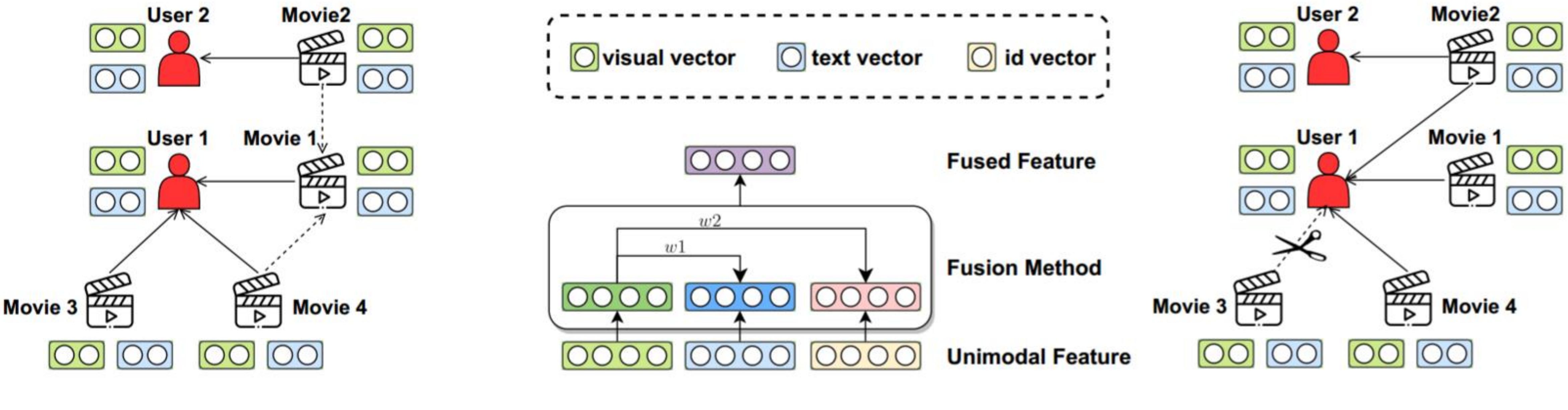
MultiModal Encoder

Example: Multimodal encoder in VLSNR: Clip+ViT

VLSNR: Vision-Linguistics Coordination Time Sequence-aware News Recommendation. arXiv preprint 2022.

Feature Interaction

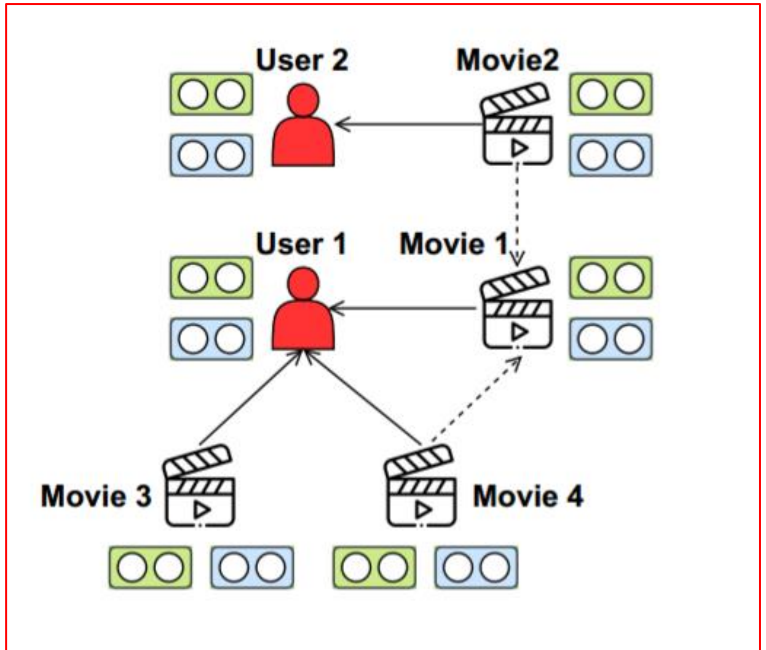
- ➢Connecting different modalities to enhance the model performance
- ➢Three mainly used types: Bridge, Fusion, and Filtration
- ➢These methods are combined and used together in some research



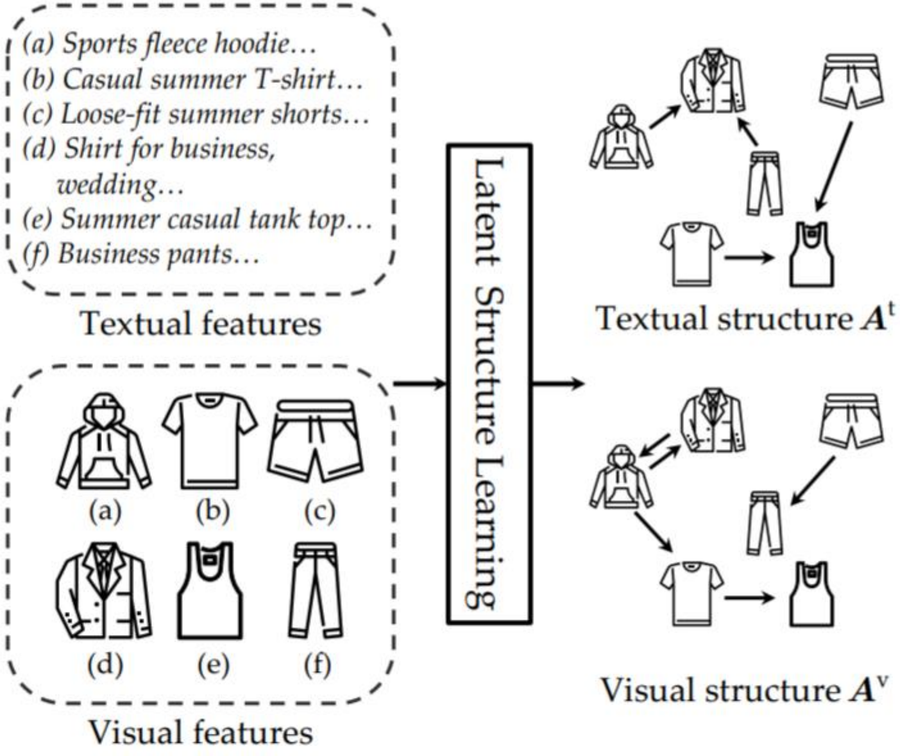
(a) Bridge (b) Fusion (c) Filtration

Feature Interaction: Bridge

- ➢The construction of a multimodal information transfer channel
- ➢Capturing the inter-relationship between users and items
- ➢Form: User-item Graph, Item-item Graph, Knowledge Graph



(a) Bridge

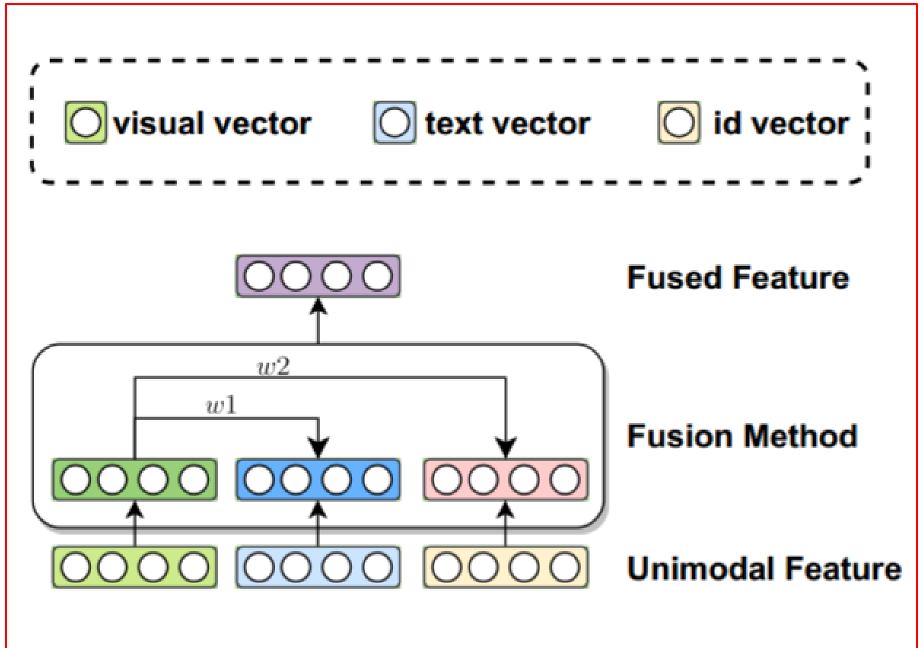


Example: item-item graph in MICRO

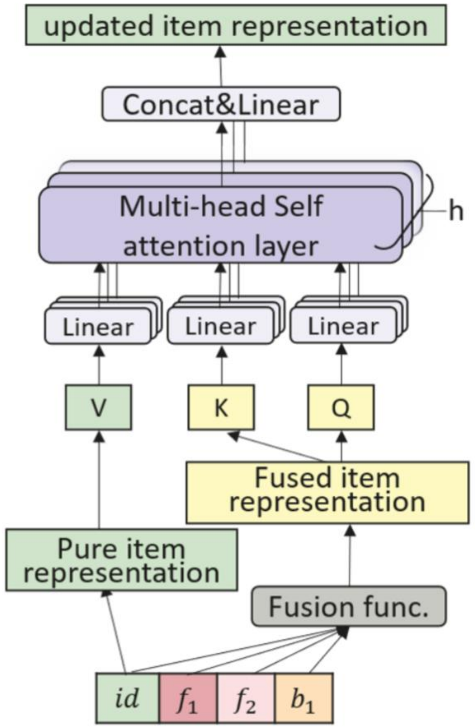
Latent structure mining with contrastive modality fusion for multimedia recommendation. TKDE 2022.

Feature Interaction: Fusion

- ➢Aiming at combining various preferences in modalities
- ➢Concerning more about the multimodal intrarelationships of items
- ➢The attention mechanism is the most widely used feature fusion method



(b) Fusion



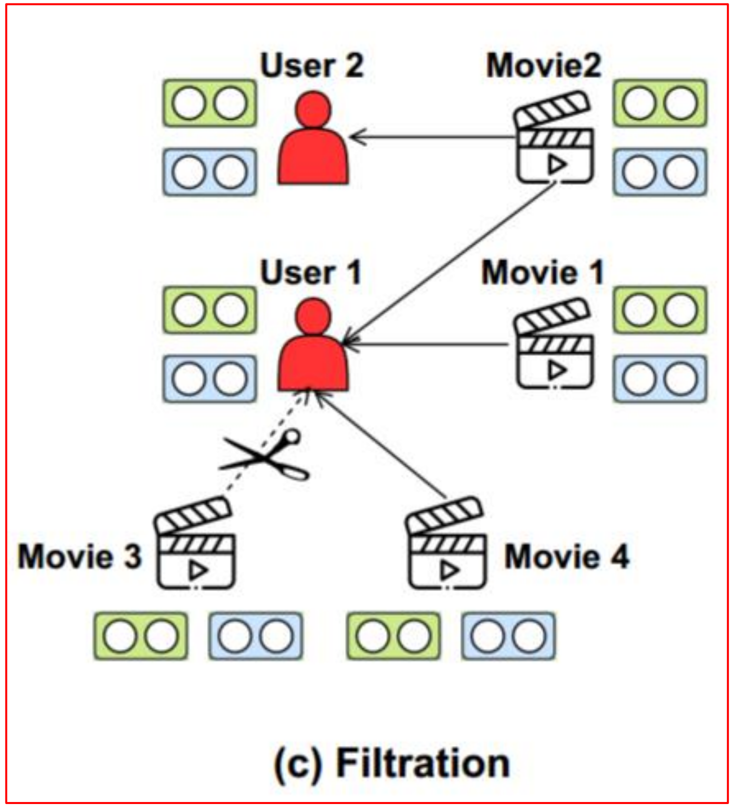
Example: Noninvasive feature fusion in NOVA

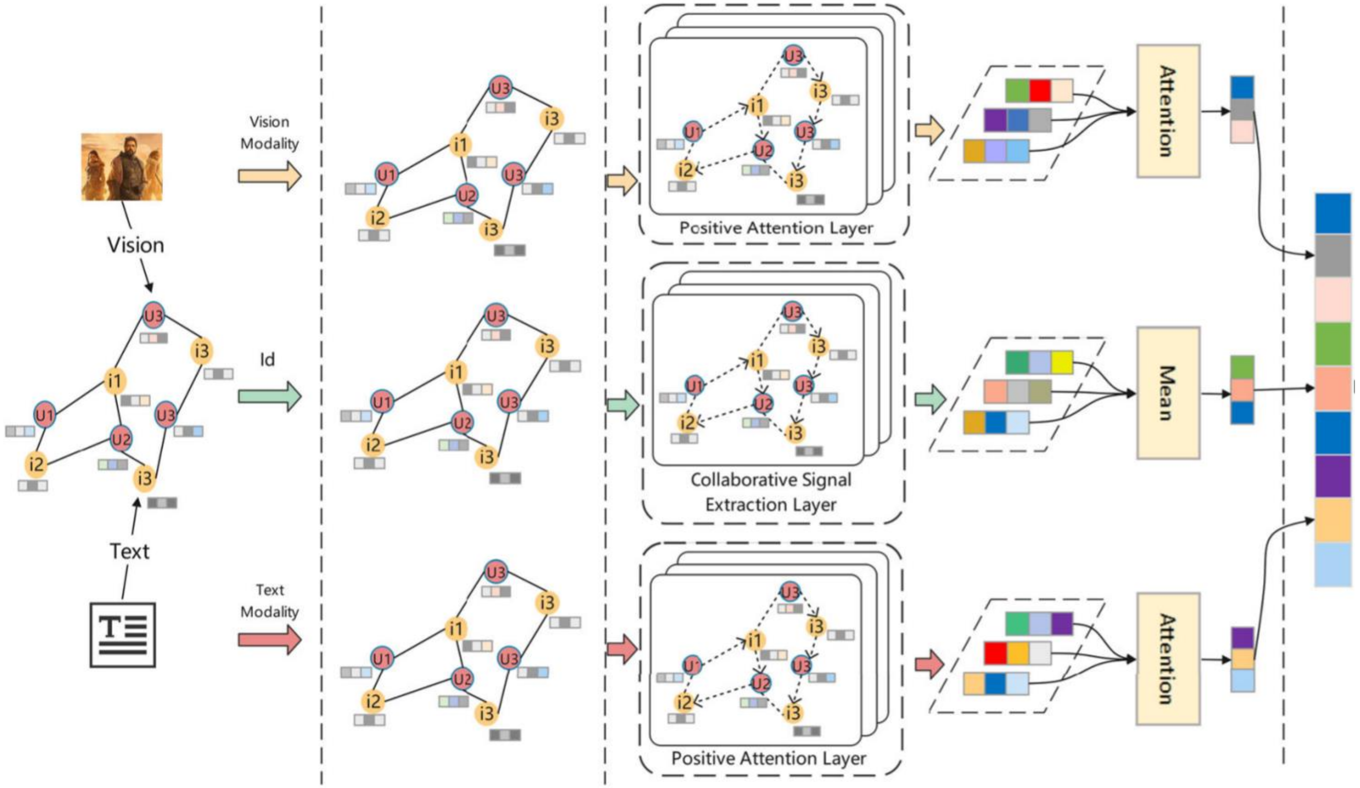
Noninvasive self-attention for side information fusion in sequential recommendation. AAAI 2021.

Feature Interaction: Filtration

➢Aiming at filtering out noisy data (data that is unrelated to user preferences)

➢This step could be done for modality features, or the feature interactions



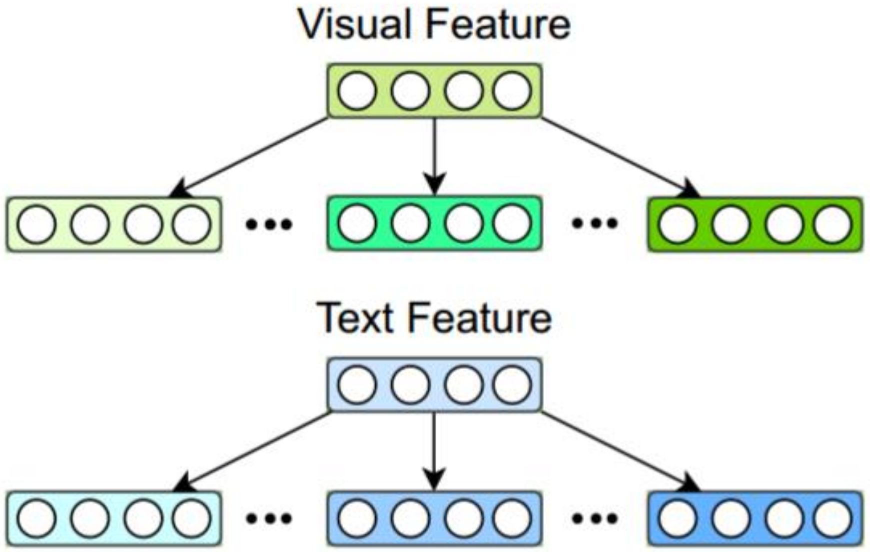


Example: interaction denoising with an active attention mechanism in PMGCRN

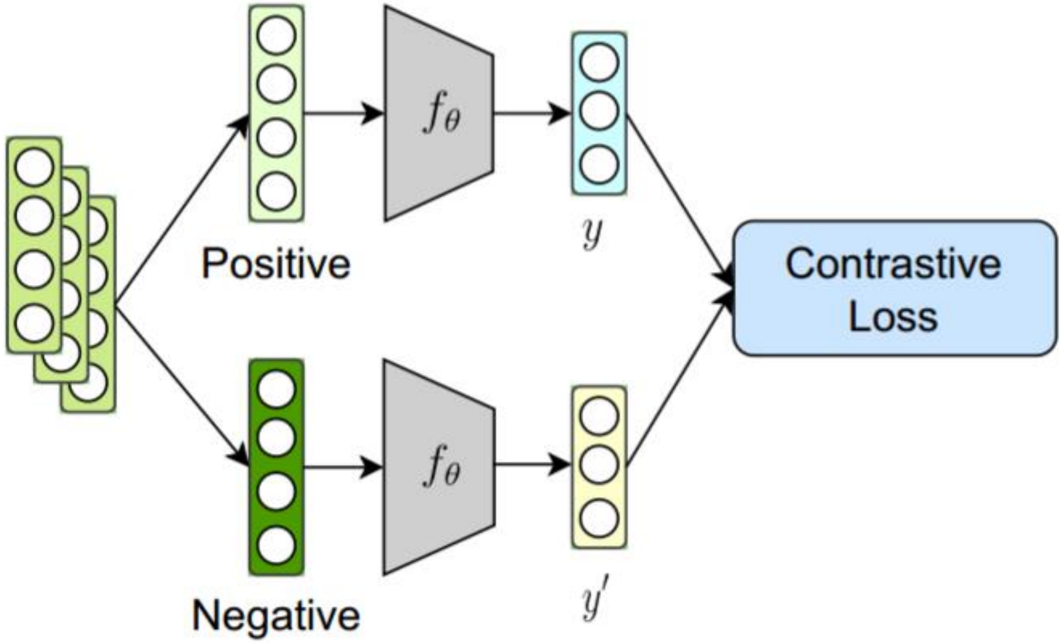
Preference-corrected multimodal graph convolutional recommendation network. Applied Intelligence 2023.

Feature Enhancement

- Different modalities of the same object have unique and common semantic information
- The recommendation performance and generalization of MRS can be significantly improved if the unique and common characteristics can be distinguished
- Methods: Disentangled Representation Learning, Contrastive Learning



(a) Disentangled Representation Learning

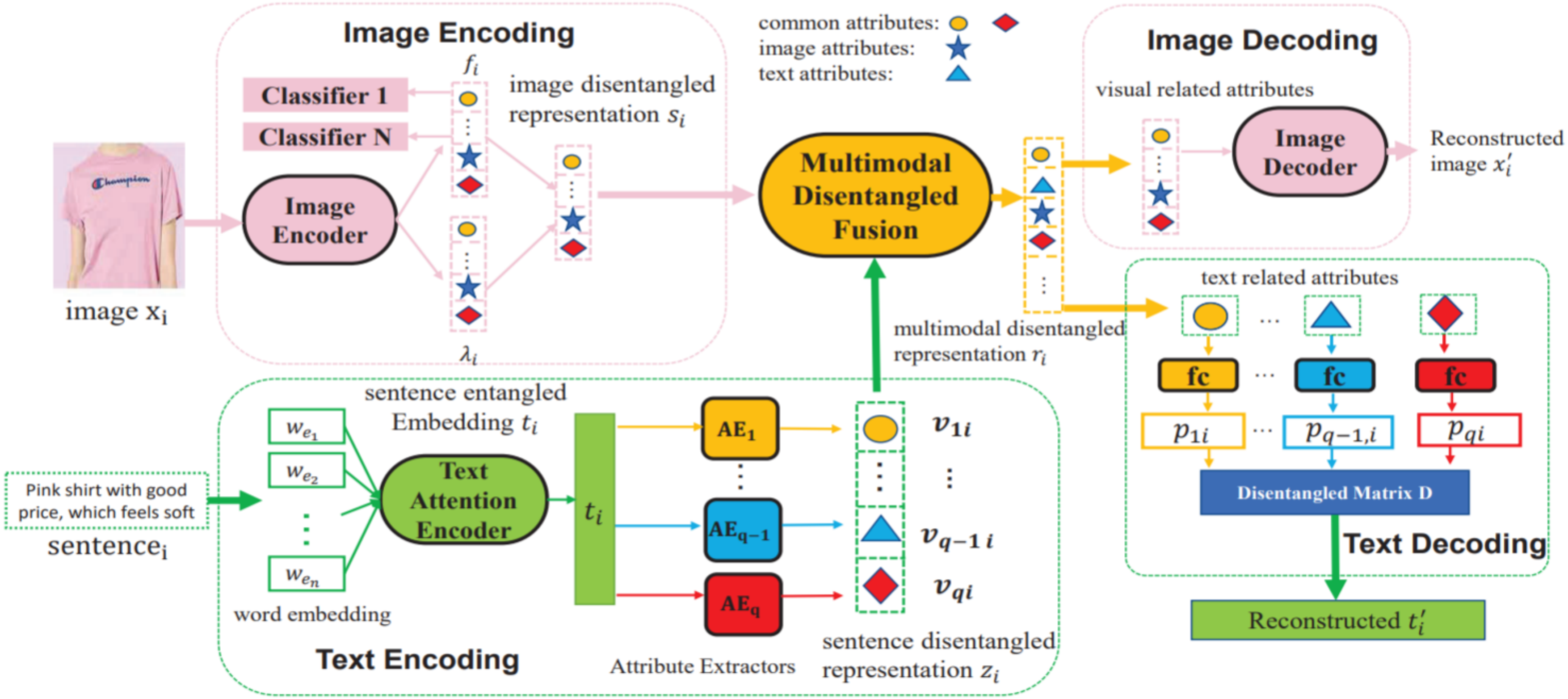


(b) Contrastive Learning

Disentangled Representation Learning

➢MDR: multimodal disentangled recommendation

-> fuse representations that have the similar meaning



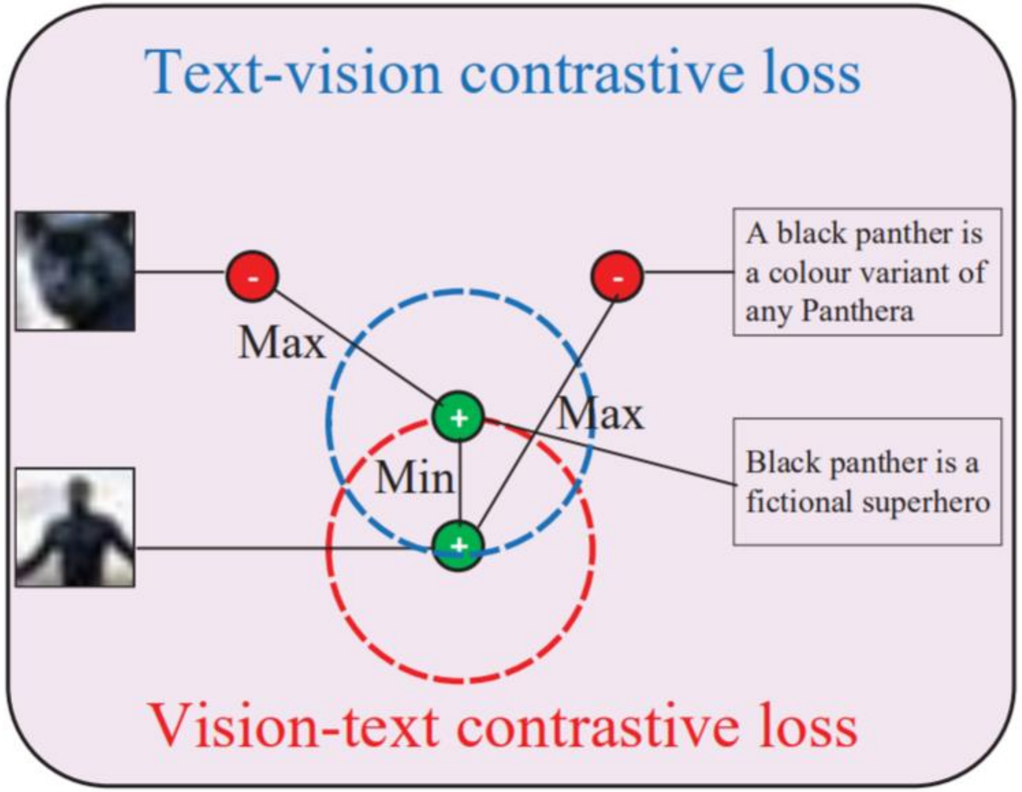
Example: MDR for multimodal disentangled recommendation

Multimodal disentangled representation for recommendation. ICME 2021.

Contrastive Learning

➢GHMFC: contrastive learning modules with two loss functions (text2image and image2text)

-> Learning similar semantic knowledge

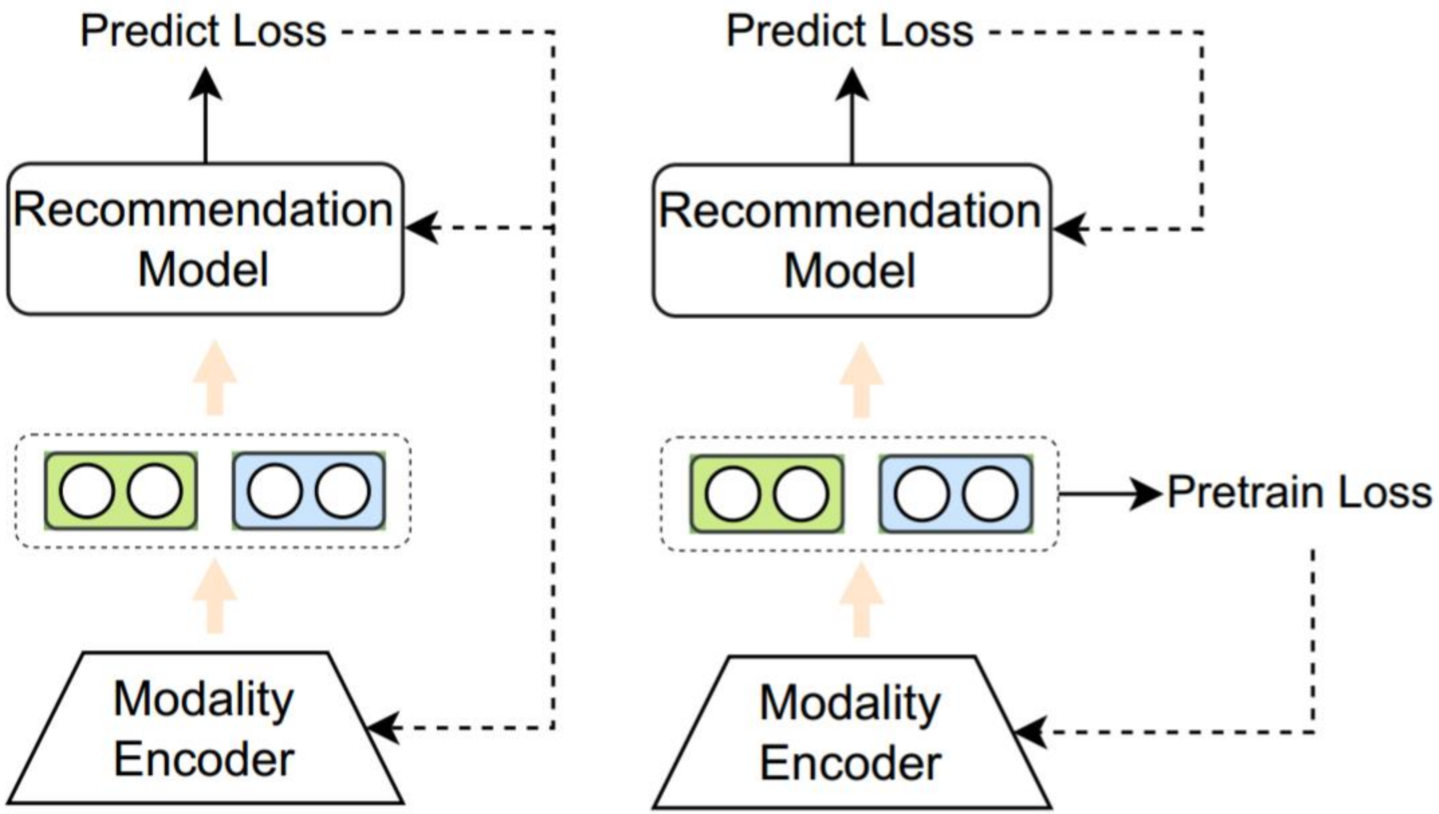


Example: Contrastive loss of GHMFC

Multimodal entity linking with gated hierarchical fusion and contrastive training. SIGIR 2022.

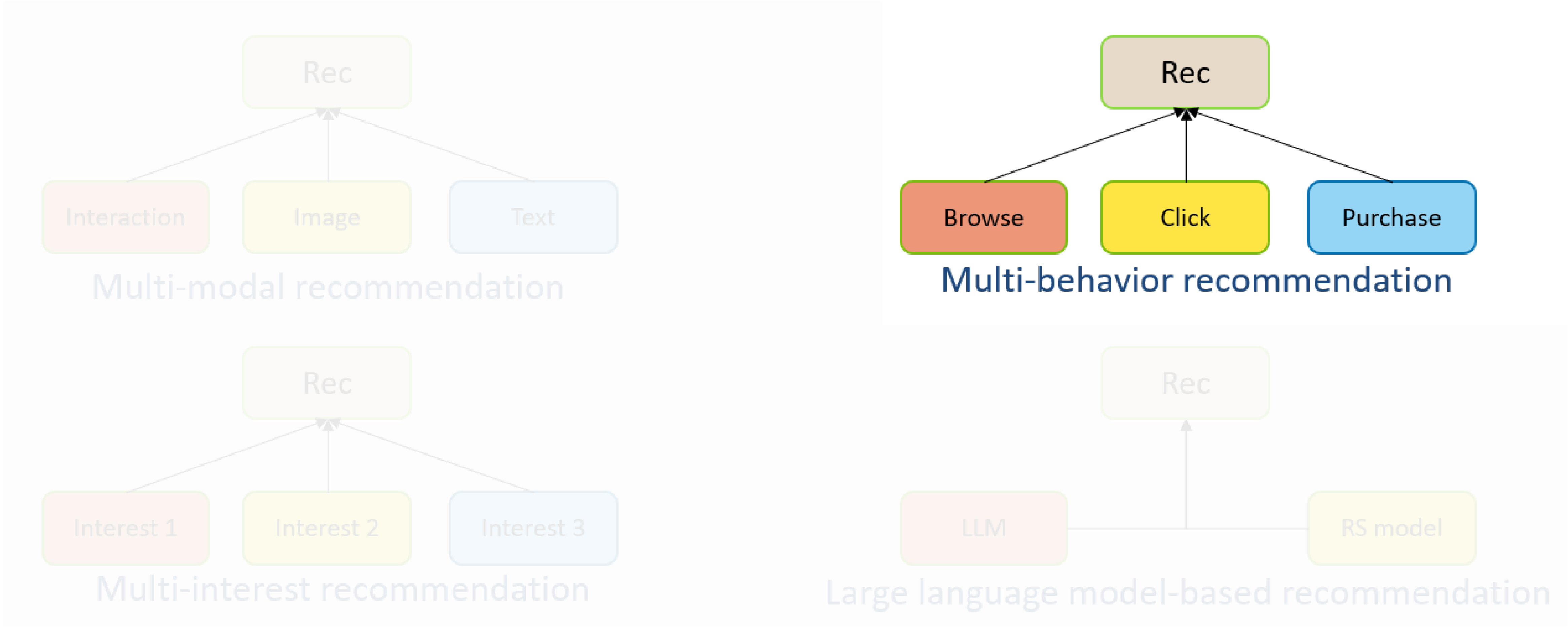
Model Optimization

- The computational requirements are greatly increased with multimodal information
- Training strategies: End-to-end training (with pre-trained encoder), Two-step training



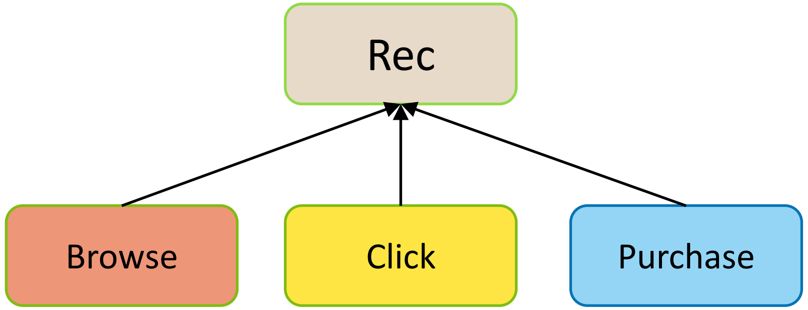
(a) End-to-end Training (b) Two-step Training

Multi-Behavior Modeling



Multi-Behavior Modeling

- ➢Understanding behavior patterns and behavior correlations at a fine-grained granularity
- ➢Explicitly considering the different behavior types as they convey subtle differences in user interest modeling



Multi-behavior recommendation

A Survey on User Behavior Modeling in Recommender Systems. arxiv preprint 2023.

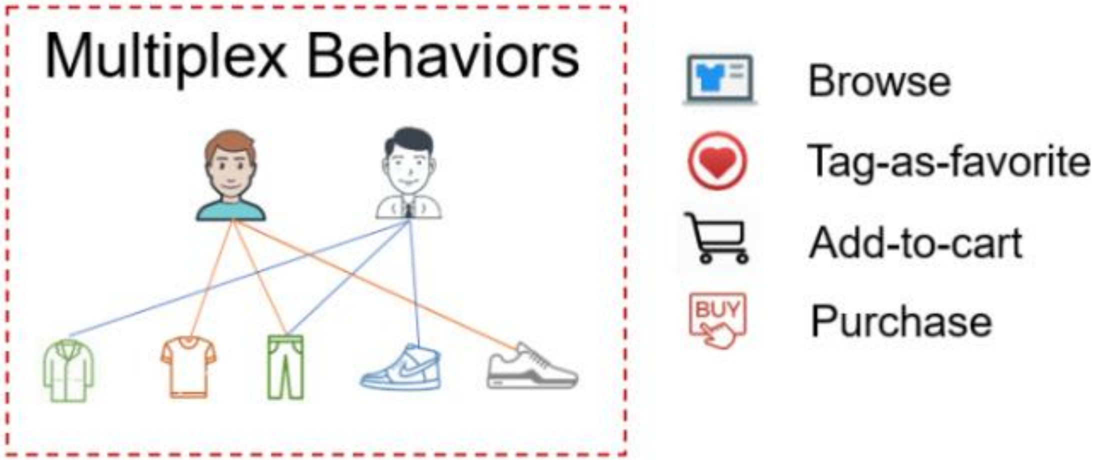
Behavior Type Definition

- An open question
- Roughly three categories:
  - Macro behaviors: interaction with different items

    E.g. user 1 interact with item 1, then item 22, then item 81.
  - Micro behaviors: actions taken on this item

    E.g. click, add to cart,…
  - Behaviors from different domains or scenarios

    E.g. Same behavior in two domains => different behaviors (highlight the distinctions)

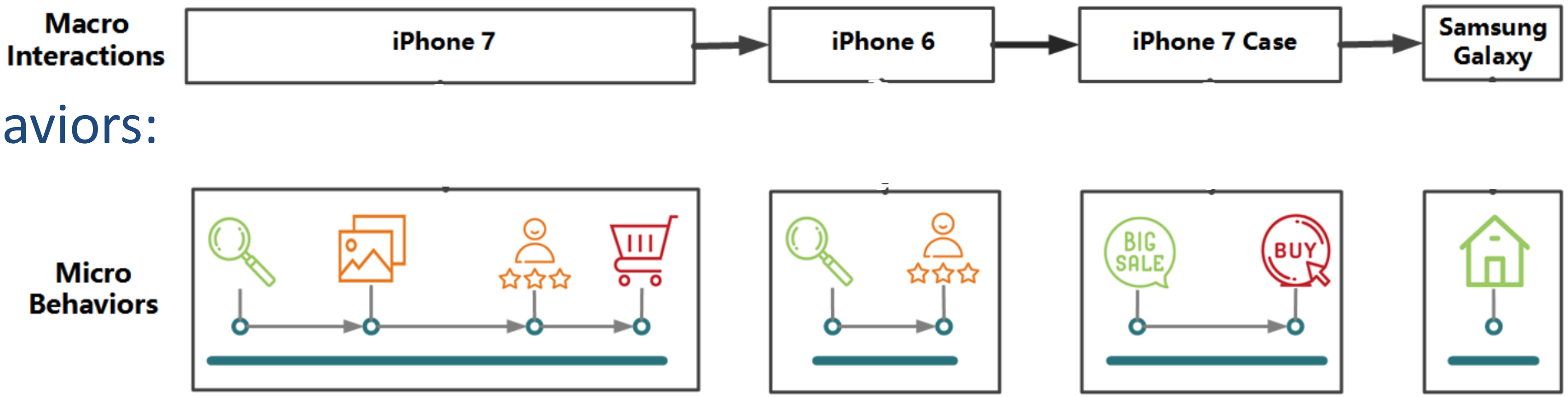


A Survey on User Behavior Modeling in Recommender Systems. arxiv preprint 2023.
Graph Meta Network for Multi-Behavior Recommendation. SIGIR 2021.

Behavior Type Definition

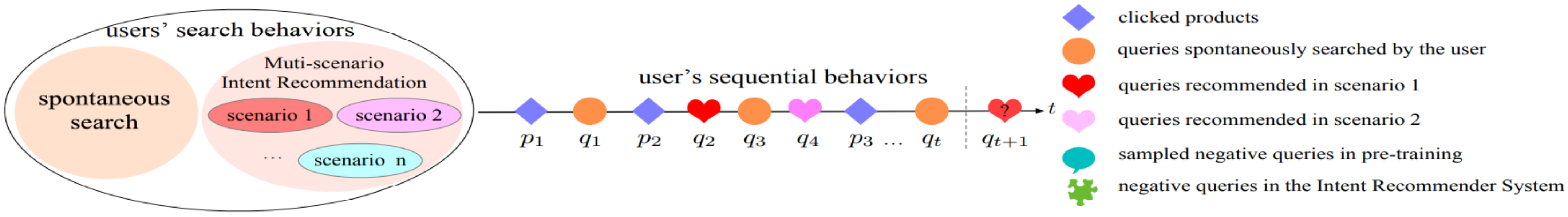
- Macro behaviors:

- Micro behaviors:



- Behaviors from different domains or scenarios

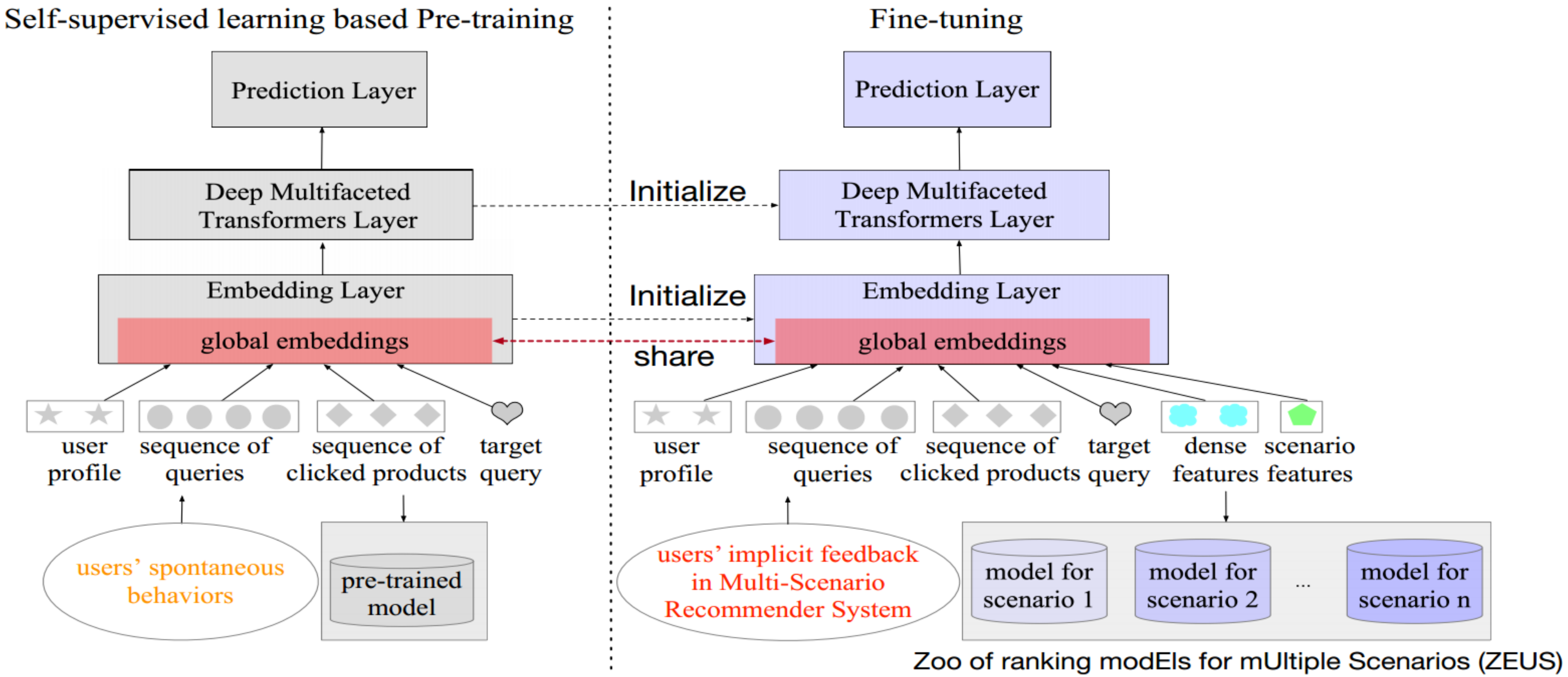
  E.g. Same behavior in two domains => different behaviors (highlight the distinctions)



Micro behaviors: A new perspective in e-commerce recommender systems. WSDM 2018.
Self-Supervised Learning on Users' Spontaneous Behaviors for Multi-Scenario Ranking in E-commerce. CIKM 2021.

Multi-Behavior Fusion

➢ Modeling the complicated cross-scenario behavior dependencies

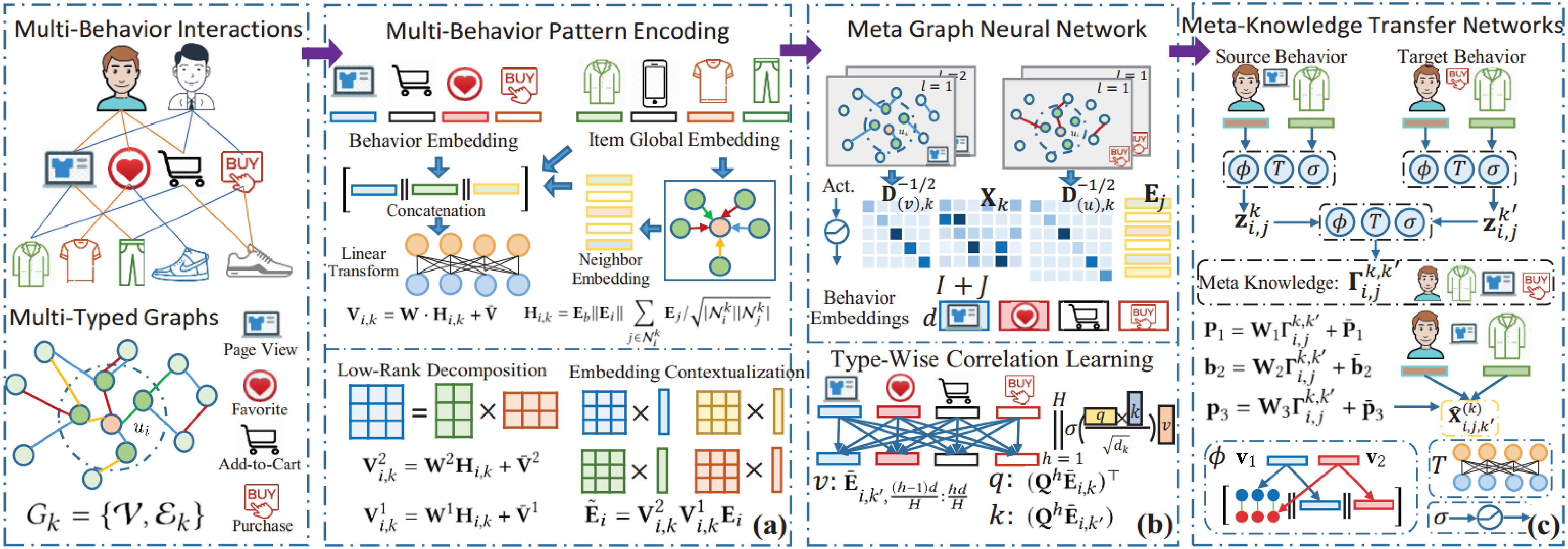


Example: pre-training and fine-tuning of ZEUS

Self-Supervised Learning on Users' Spontaneous Behaviors for Multi-Scenario Ranking in E-commerce. CIKM 2021.

Multi-Behavior Fusion

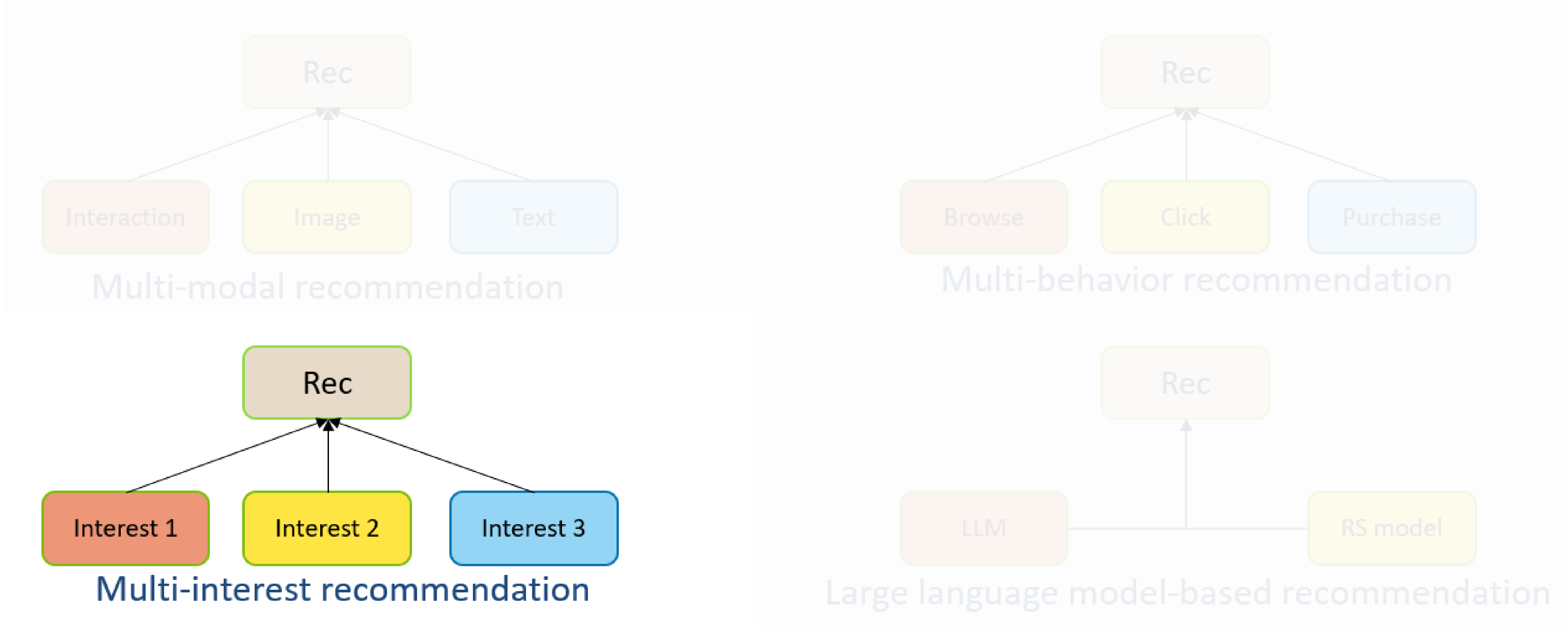
➢Modeling the complicated cross-type behavior dependencies



Example: MB-GMN

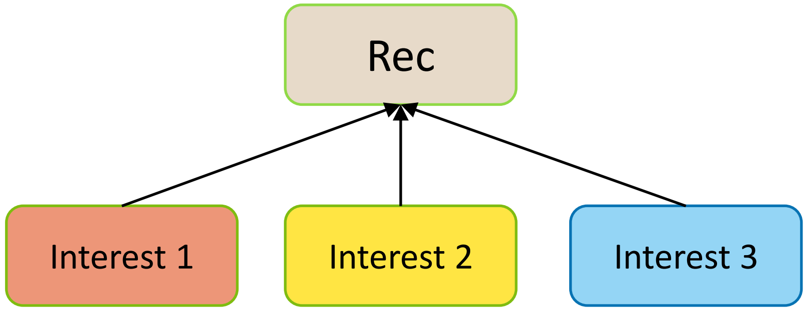
Multi-behavior sequential transformer recommender. SIGIR 2022.

Multi-Interest Recommendation



Multi-Interest Recommendation

- Information cocoon: When a user clicks and buys an item, the platform will only recommend items that are very similar
- Multi-Interest Recommendation: Improving the diversity and discovery of recommendations to better meet user interests

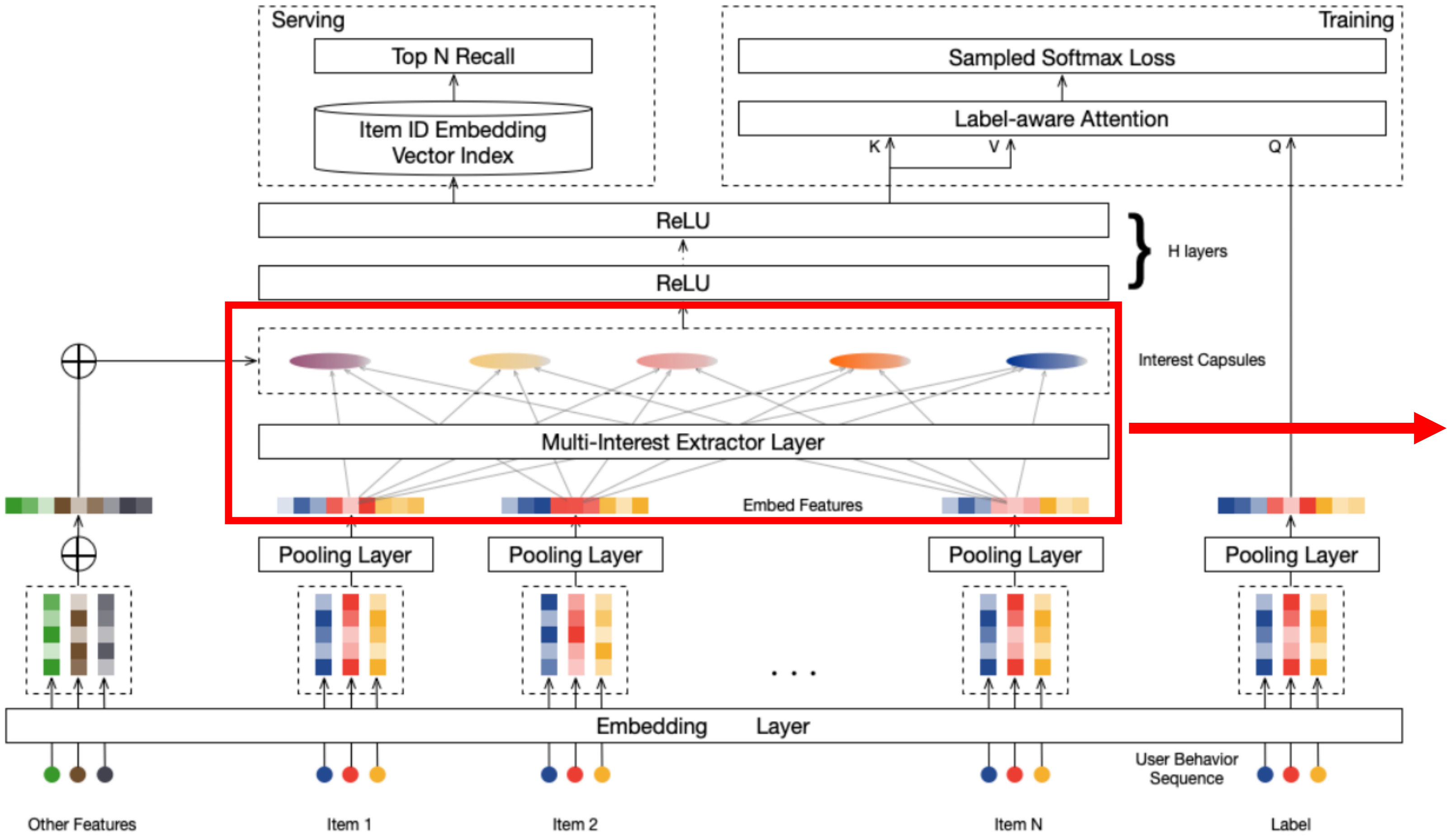


Multi-interest recommendation

MIND

- Mining interests: Interest Capsules (clustering)
- for item i and interest j:

$$b_{ij} = \vec{u}_j^T S \vec{e}_i$$



$$b_{ij} = (\vec{c}_j^h)^T S_{ij} \vec{c}_i^l$$

$$w_{ij} = \frac{\exp b_{ij}}{\sum_{k=1}^{m} \exp b_{ik}}$$

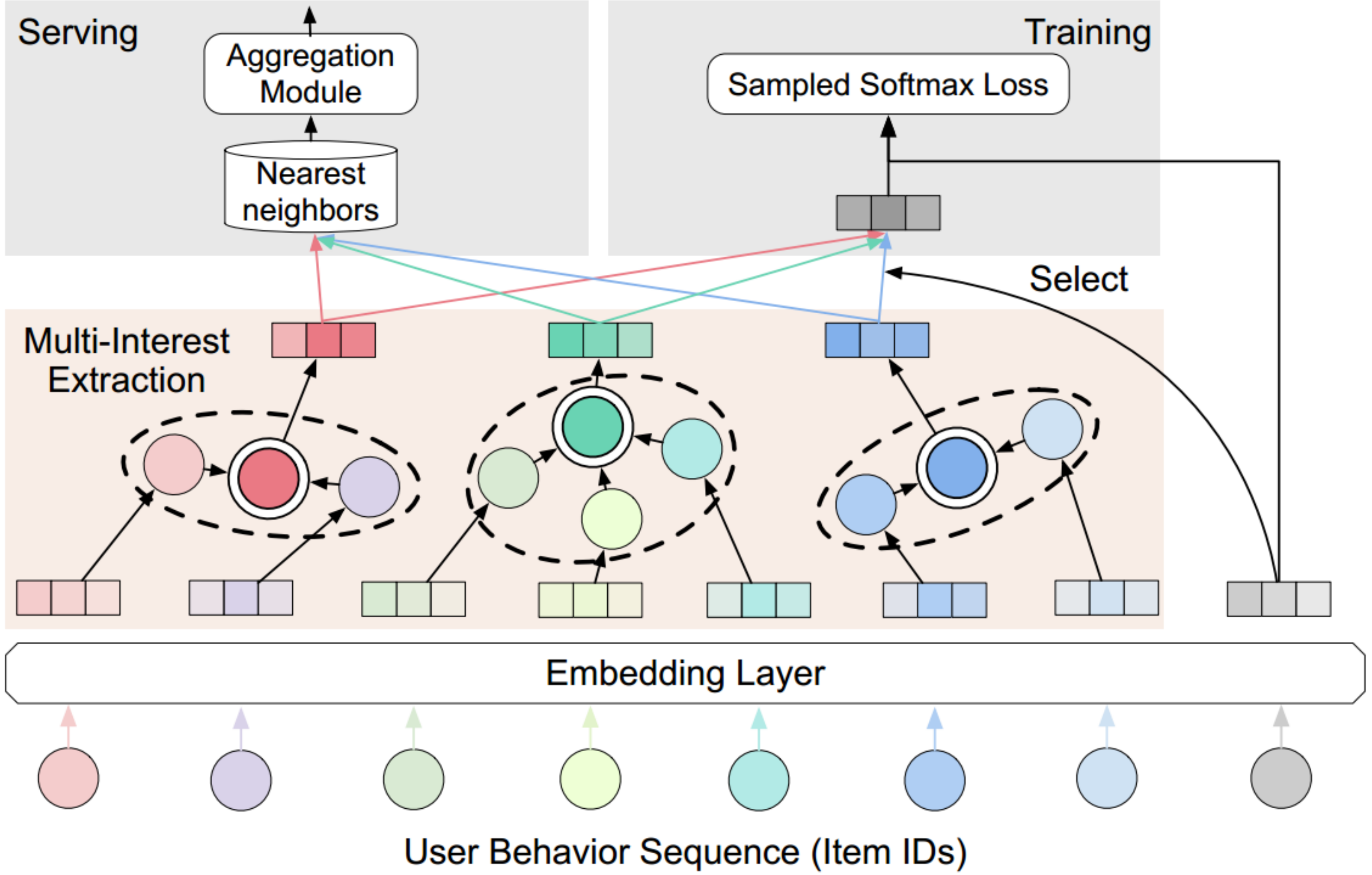
$$\vec{z}_j^h = \sum_{i=1}^{m} w_{ij} S_{ij} \vec{c}_i^l$$

$$\vec{c}_j^h = squash(\vec{z}_j^h) = \frac{\left\|\vec{z}_j^h\right\|^2}{1 + \left\|\vec{z}_j^h\right\|^2} \frac{\vec{z}_j^h}{\left\|\vec{z}_j^h\right\|}$$

Multi-interest network with dynamic routing for recommendation at Tmall. CIKM 2019.

ComiRec

- ➢Mining interests: Interest Capsules (clustering)
- ➢Balancing the accuracy and diversity of the recommendation



Each interest embedding can independently retrieve top-N items based on the inner production proximity.
Total N*Interest candidates

Algorithm 2: Greedy Inference

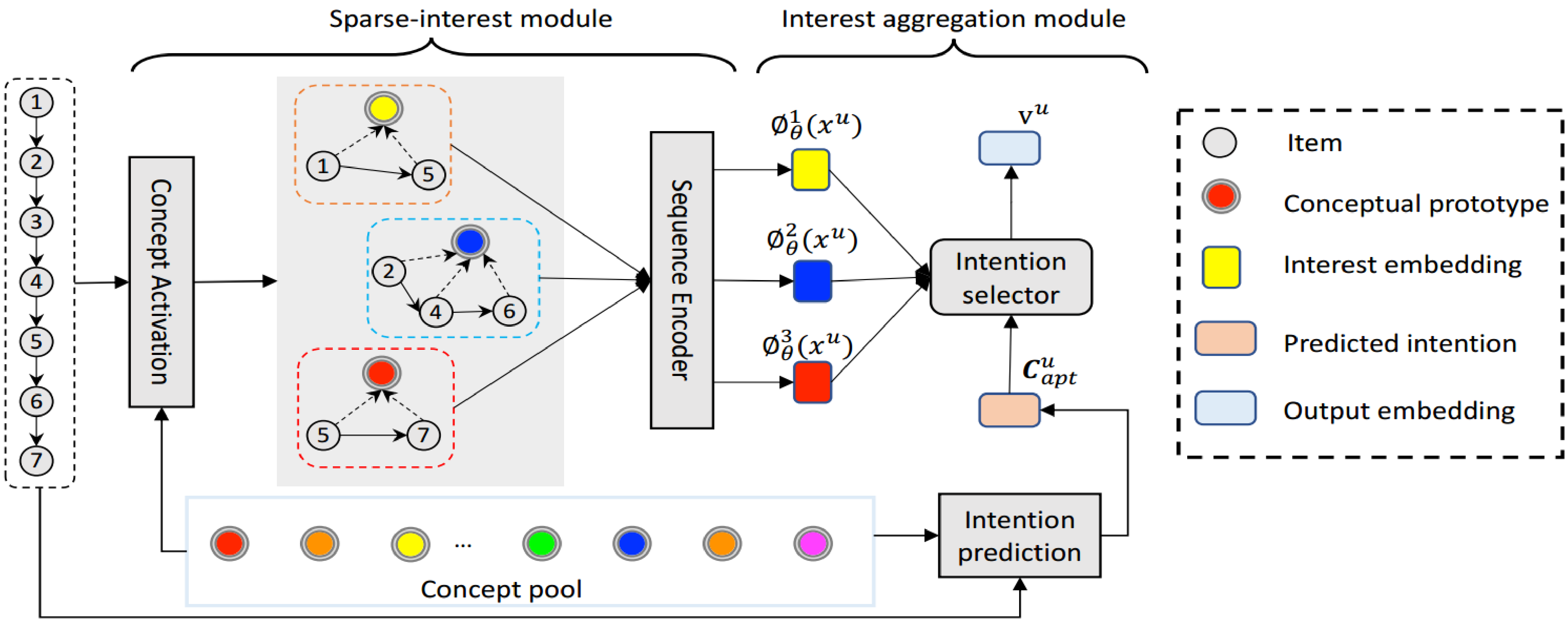
Input: Candidate item set $\mathcal{M}$, number of output items N
Output: Output item set $\mathcal{S}$

1. $\mathcal{S} = \varnothing$
2. **for** $iter = 1, \cdots, N$ **do**
3. $\quad j = \text{argmax}_{i \in \mathcal{M} \setminus \mathcal{S}} \left(f(u, i) + \lambda \sum_{k \in \mathcal{S}} g(i, k) \right)$
4. $\quad \mathcal{S} = \mathcal{S} \cup \{j\}$
5. **return** $\mathcal{S}$

Controllable multi-interest framework for recommendation. KDD 2020.

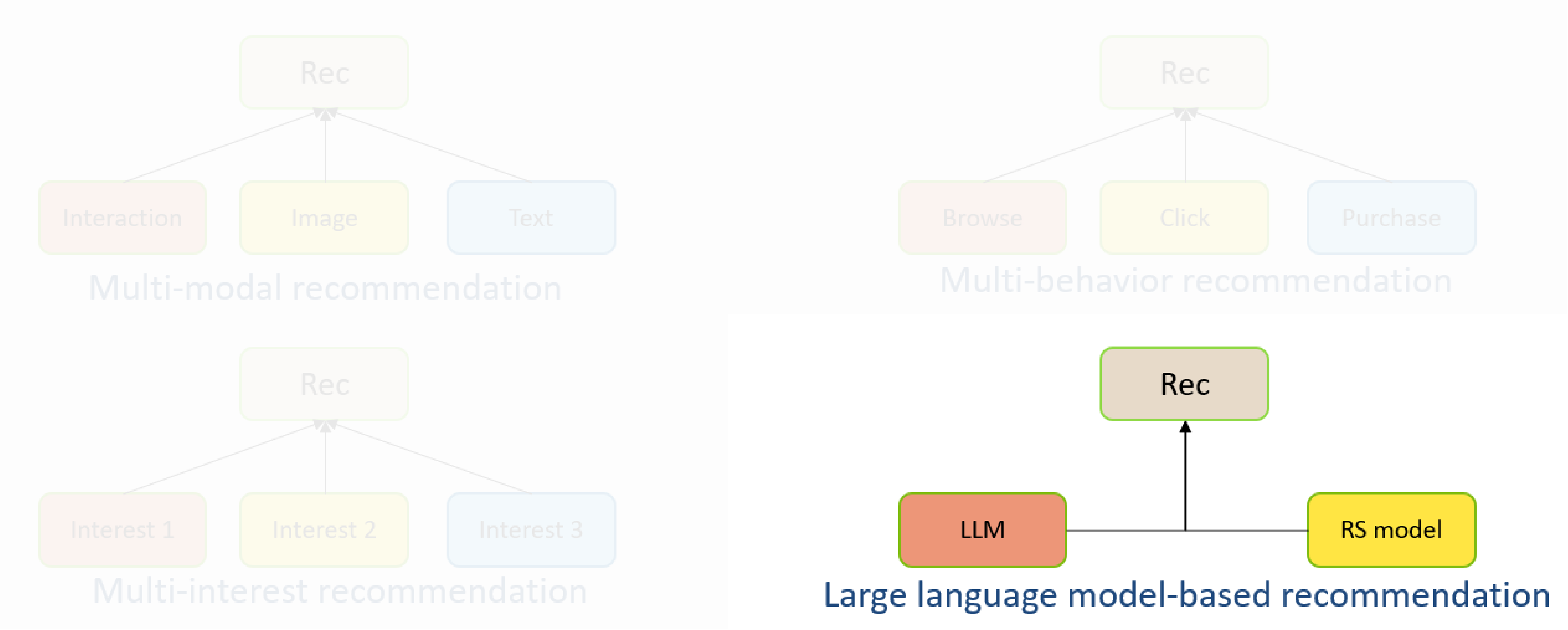
SINE

- ➢Sparse interests: activating different concepts for different input
- ➢Making prediction based on the user intention and activated concepts



Sparse-interest network for sequential recommendation. WSDM 2021.

LLM-based Recommendation

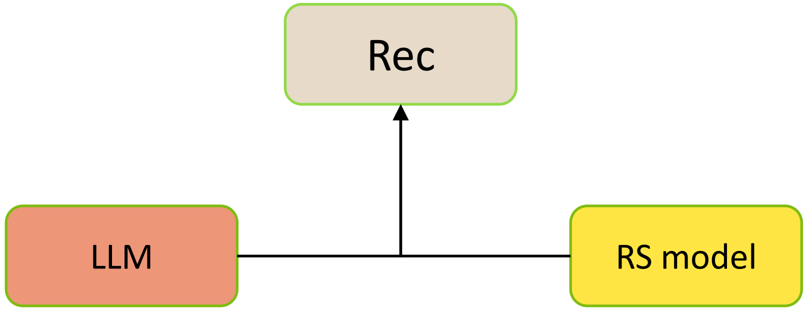


LLM-based Recommendation

➢ Large language model-based recommendation

➢ Two methods:

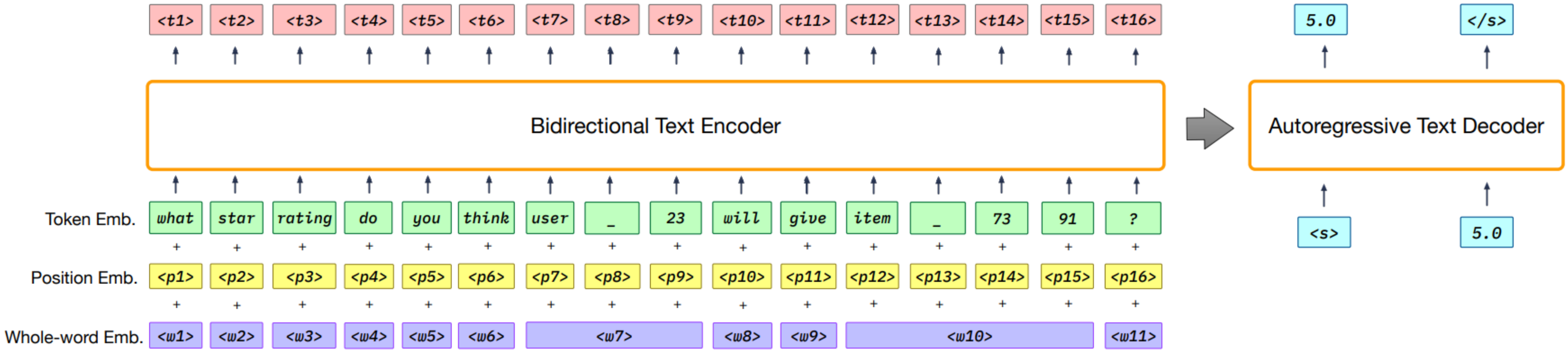
- Fine-tuning
- LLM as a submodule



Large language model-based recommendation

Fine-Tuning

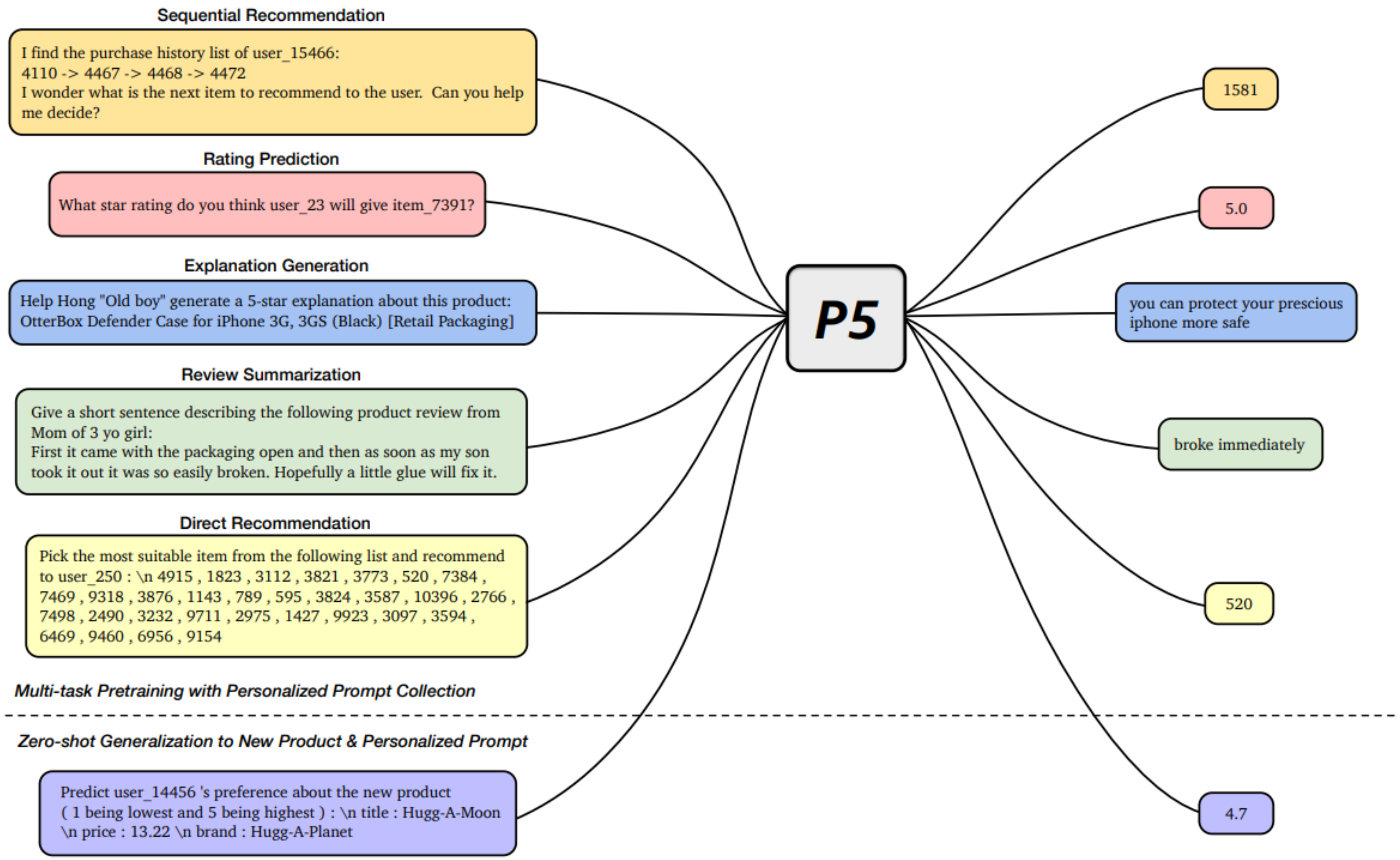
➢P5: a unified recommendation model with pre-trained LLM model T5



Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). RecSys 2022.

Fine-Tuning

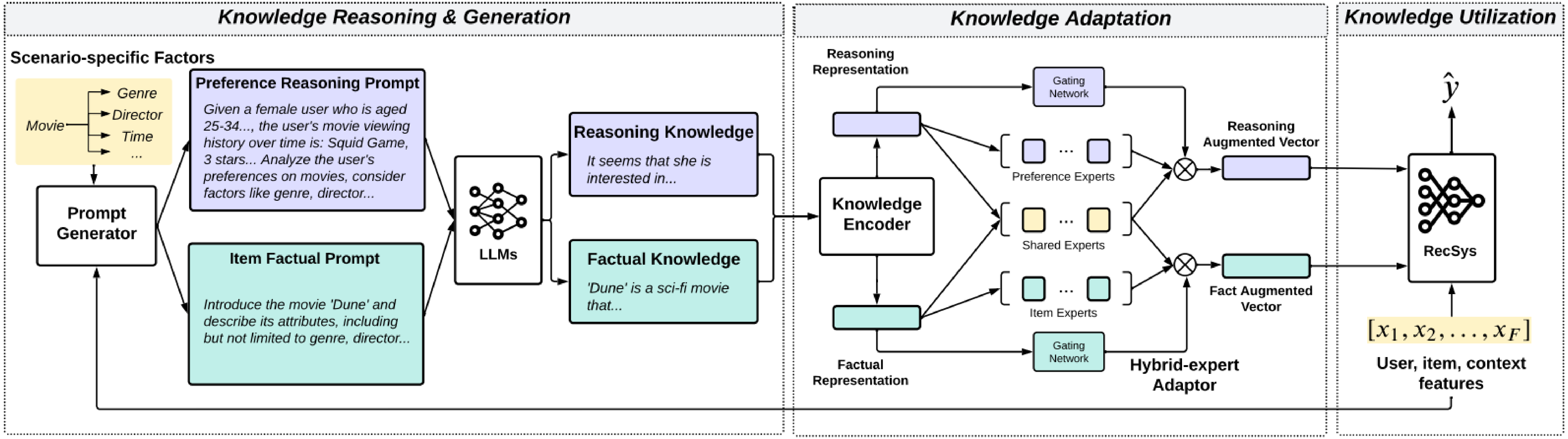
- P5: a unified recommendation model with pre-trained LLM model T5
- Fine-tuning with five commonly used tasks



Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). RecSys 2022.

LLM As a Submodule

- KAR: using LLM as a submodule to obtain more general knowledge
- Knowledge Encoder: NLP-based encoder. E.g. BERT



Towards Open-World Recommendation with Knowledge Augmentation from Large Language Models. arxiv 2023.

Summary

➢ **More extensive joint modeling (Multi Behavior/Interest/Modal, LLM)**
- Fusing heterogeneous information from different **data modalities**
- Acquiring multi-aspect user preferences from different type of **behaviors or interests**
- Introducing open-world knowledge from **large language models**

Models	Type(or other dimension)
MB-GMN	Multi-Behavior
RIB	Multi-Behavior
ZEUS	Multi-Behavior
MIND	Multi-Interest
ComiRec	Multi-Interest
SINE	Multi-Interest
P5	LLM-Based
KAR	LLM-Based

Models	Type(or other dimension)
VLSNR	Multi-Modal
MICRO	Multi-Modal
NOVA	Multi-Modal
PMGCRN	Multi-Modal
MDR	Multi-Modal
GHMFC	Multi-Modal

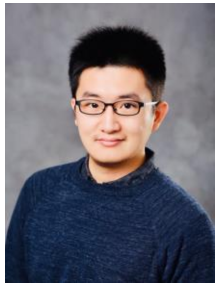
Future Directions

➢ **More extensive joint modeling**

- Joint modeling with all the above methods
- A more comprehensive approach to realize joint modeling with LLM

Agenda

Introduction



Xiangyu Zhao

Preliminary

Yichao Wang

Multi-task Recommendation

Yuhao Wang

Multi-scenario recommendation

MTR+MSR

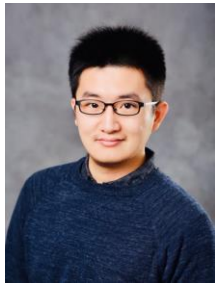
Pengyue Jia

More Joint-learning Methods

Jingtong Gao

Conclusion

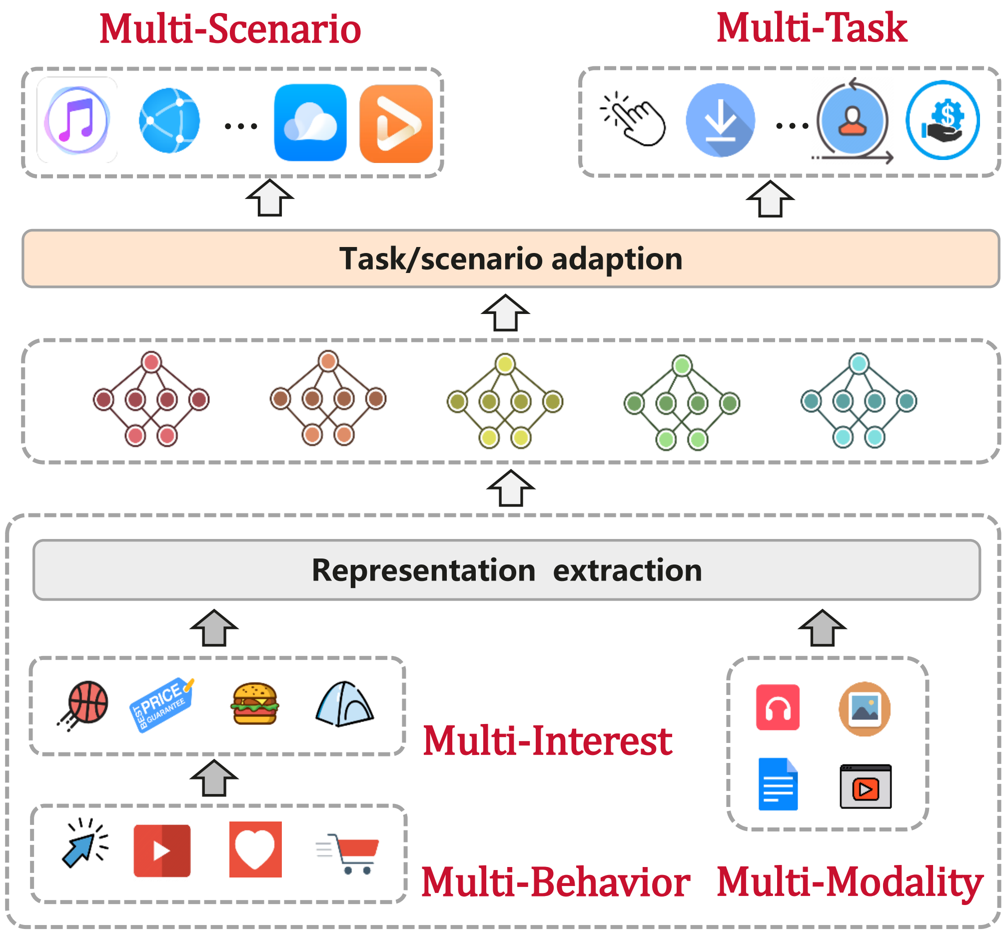
Future Work



Xiangyu Zhao

Conclusion

- ➢Utilizing diverse user feedback signals from **different tasks**
- ➢Extracting commonalities and diversities of user preferences from **different scenarios**
- ➢Fusing heterogeneous information from different **data modalities**
- ➢Acquiring multi-aspect user preferences from different type of **behaviors or interests**
- ➢Introducing open-world knowledge from **large language models**

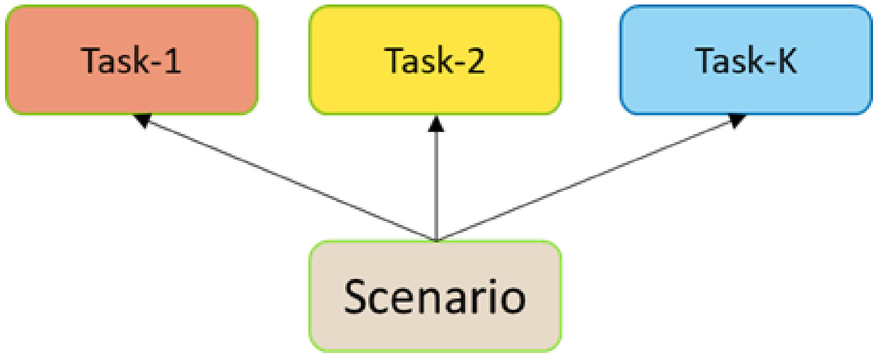


Summary

➢ **Multi-Task Recommendation**

- Task relation:
  Parallel, Cascaded, Auxiliary with Main

- Methodology:
  Parameter Sharing, Optimization, Training Mechanism



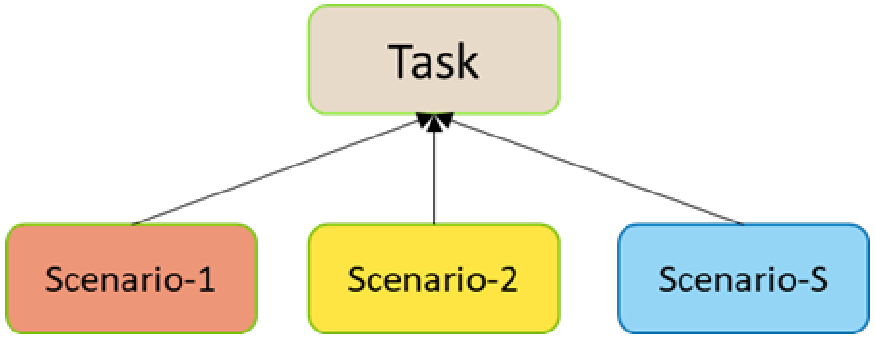
Multi-task

Summary

➢ Multi-Scenario Recommendation

- From the perspective of methods, there are mainly two categories: shared-specific network paradigm, and dynamic weight paradigm.

- Overall, most the work focuses on using one unified model serving multiple scenarios and multiple tasks simultaneously based on knowledge transfer between scenarios or tasks.



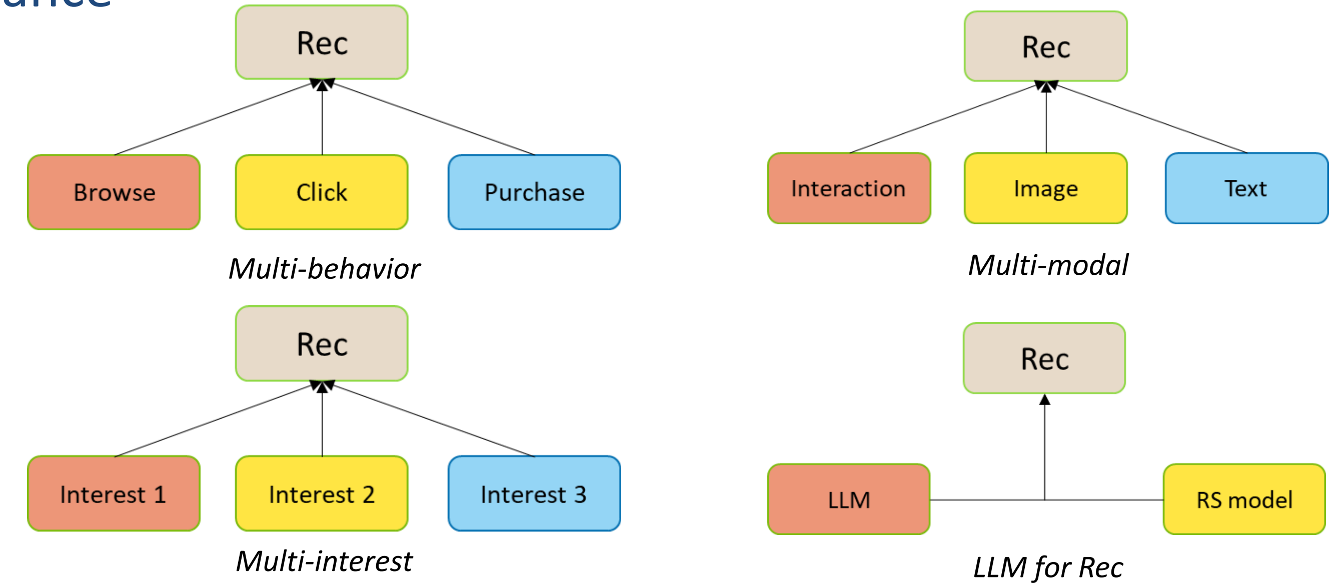
Multi-scenario

Summary

➢ More extensive joint modeling (Multi Behavior/Interest/Modal)

- Multi Behavior/Interest/Modal modeling are joint learning methods focusing on fine-grained modeling of different user/model's aspects

- LLM, as a new effective method for recommendation, could further be combined with recommendation models to jointly learn more universal knowledge to obtain a better performance



Future Directions

- **Multi-Task Recommendation**
  - Negative transfer
  - Task-specific biases

- **Multi-Scenario Recommendation**
  - Robustness
  - Privacy

- **More extensive joint modeling**
  - A more comprehensive approach to realize joint modeling with LLM

- **Ecosystem**
  - Joint modeling with all the above methods
  - More convenient for other researchers to contribute to this field

Q&A

We are hiring !

Huawei Noah's Ark Lab

IJCAI23 Huawei Noah's Ark Lab Chat Group

Xiangyu Zhao

City University of Hong Kong

Multi-Task Deep Recommendation Systems: A Survey.

https://arxiv.org/abs/2302.03525