

Joint Modeling in Deep Recommender Systems

Pengyue Jia¹Jingtong Gao¹Yejing Wang¹Yuhao Wang¹Xiaopeng Li¹Qidong Liu¹Yichao Wang²Bo Chen²Huifeng Guo²Ruiming Tang²

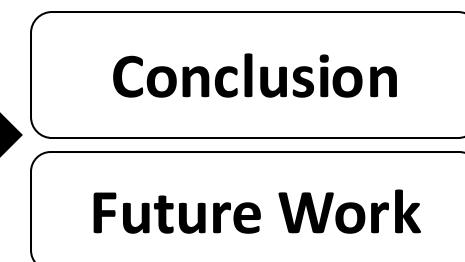
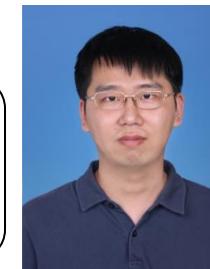
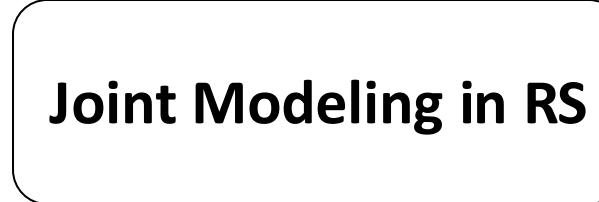
¹City University of Hong Kong, ²Huawei Noah's Ark Lab



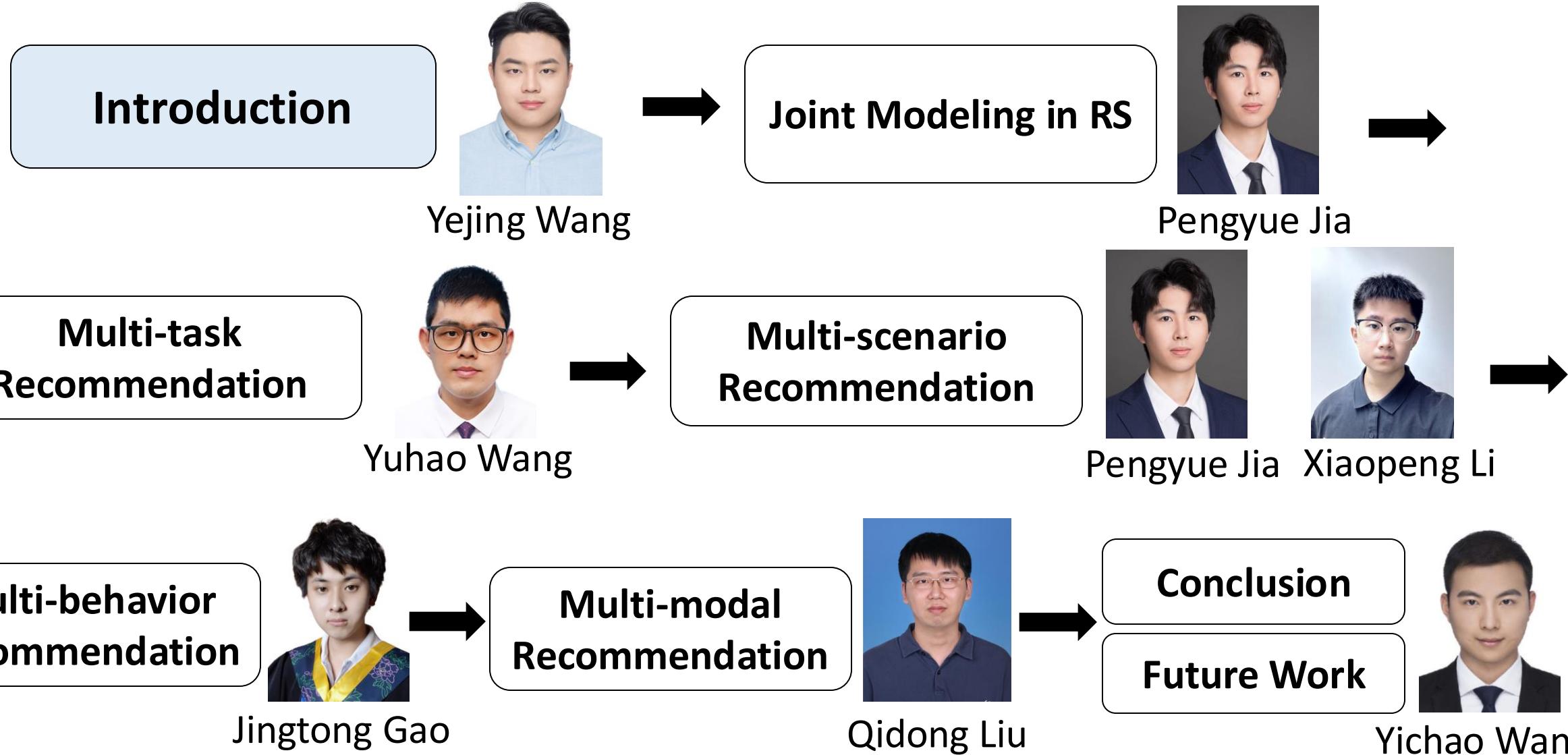
CityU AML Lab



Huawei Noah's Ark Lab



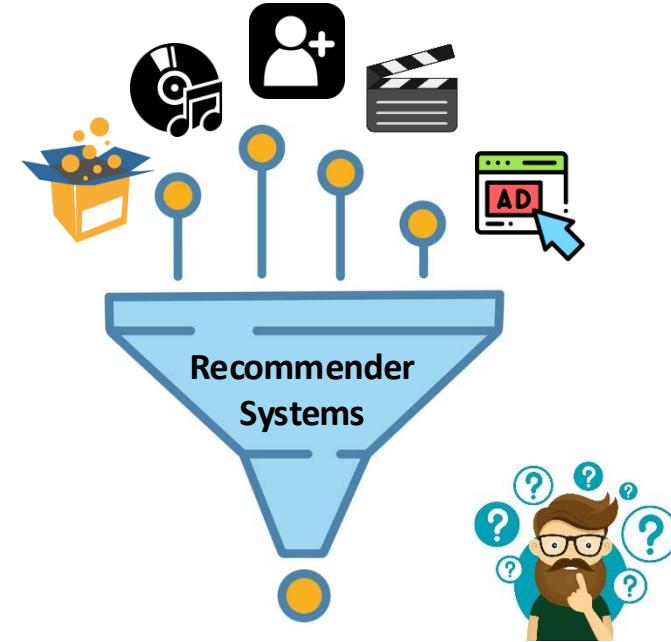
Yichao Wang



Age of Information Explosion



Information overload



Recommend item X to user

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

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

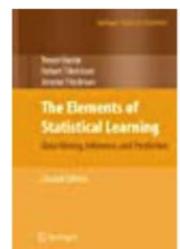


Product Recommendation

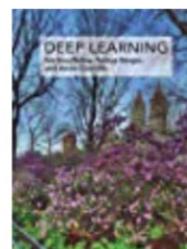
Frequently bought together



+



+



A

B

C

Total price: \$208.9

Add all three to Cart

Add all three to List

Recommender Systems



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



News/Video/Image Recommendation

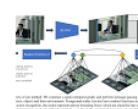
For you

Recommended based on your interests

This Research Paper From Google Research Proposes A 'Message Passing Graph Neural Network' That Explicitly Models Spatio-Temporal Relations

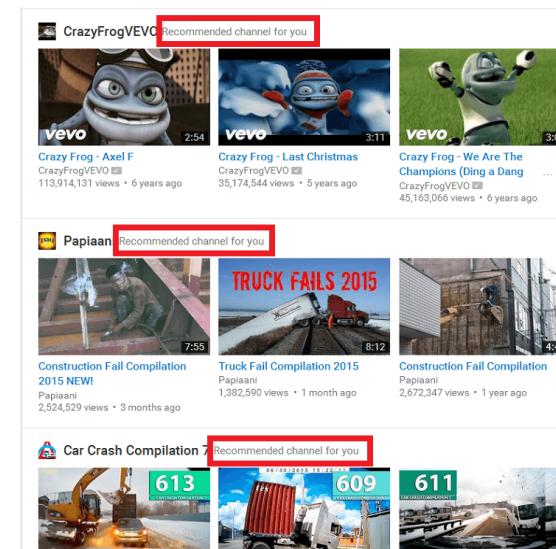
MarkTechPost · 2 days ago

More For you



Tested: Brydge MacBook Vertical Dock, completing my MacBook Pro desktop

9to5Mac · 21 hours ago



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

facebook



LinkedIn



Friend Recommendation

The screenshot shows a Facebook user profile for Andrew Torba. On the left, there's a sidebar with 'FAVORITES' and links to 'News Feed', 'Messages', 'Events', 'Find Friends', 'Tech.li', 'Kuhcoon', and several redacted links. On the right, a modal window titled 'Are They Your Friends Too?' displays four profiles with their mutual friend counts and 'Add Friend' buttons. The profiles are: one with 1 mutual friend, one with 67 mutual friends, one with 39 mutual friends, and one with 47 mutual friends. A 'See All Suggestions' button is at the bottom of the modal.

Mutual Friends	User Profile
1	[Profile Picture]
67	[Profile Picture]
39	[Profile Picture]
47	[Profile Picture]

See All Suggestions

Deep Recommender Architecture

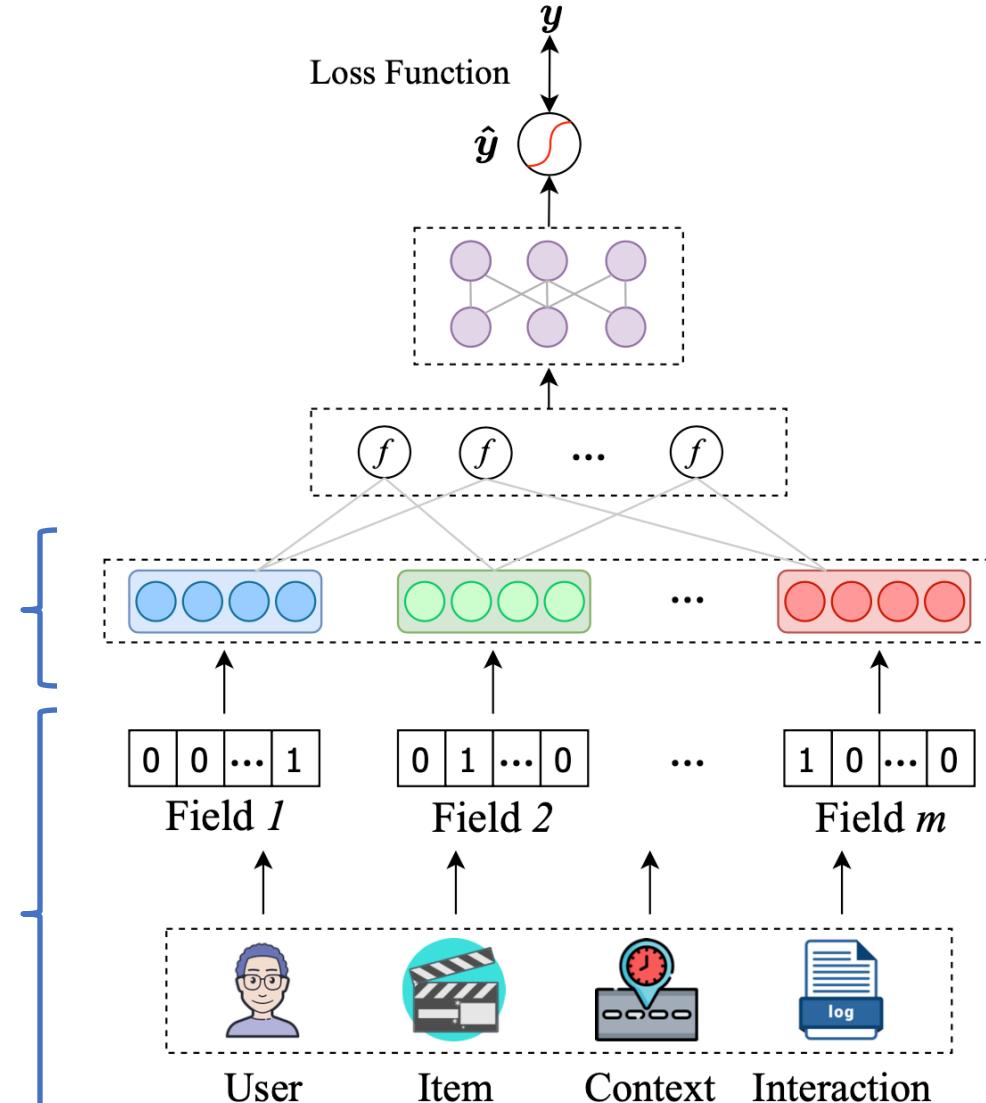


➤ Advantages

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

Feature Embedding Layer
High/low-frequency features
embedding sizes

Input Layer
Feature selection

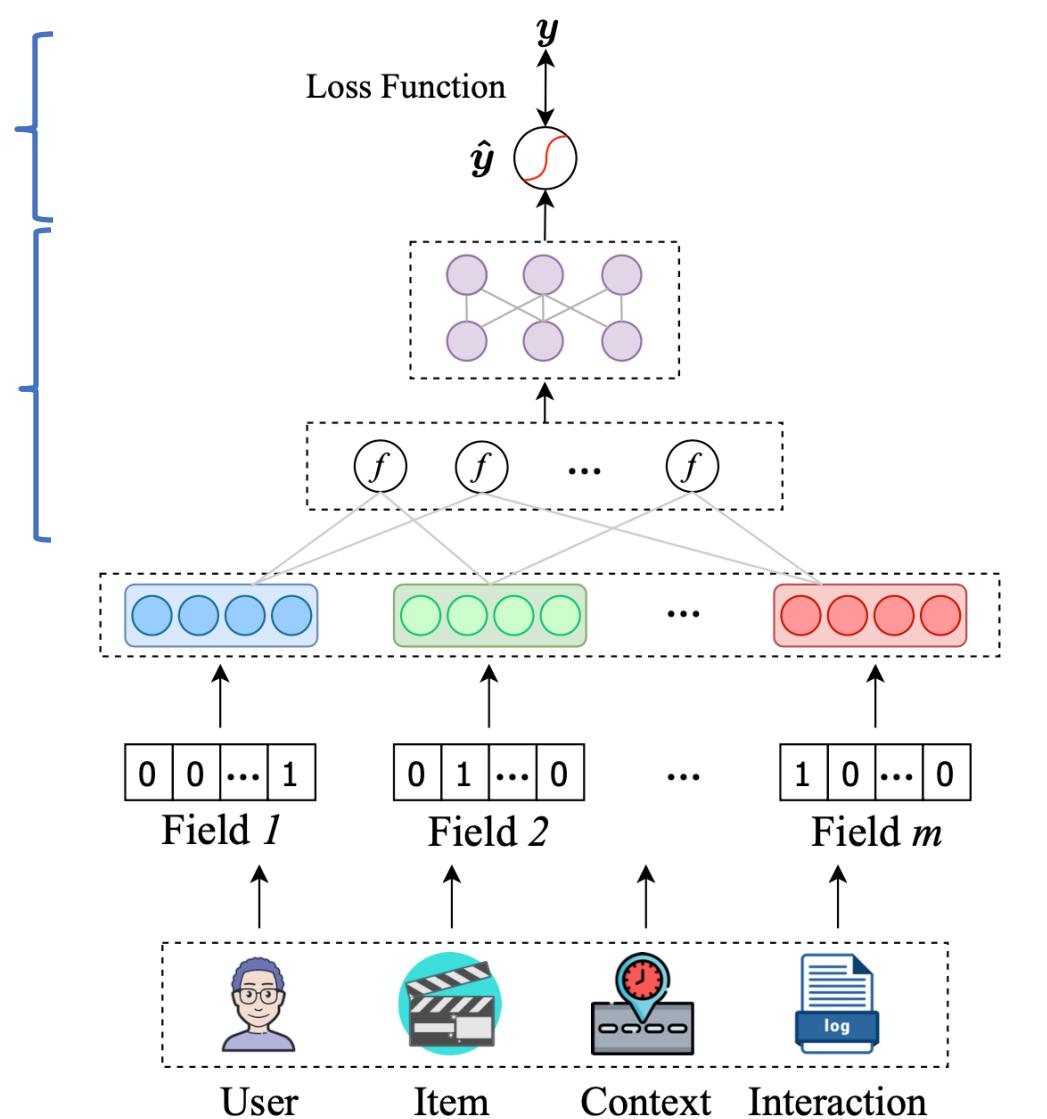


Deep Recommender Architecture



Output Layer
BCE, BPR, MSE

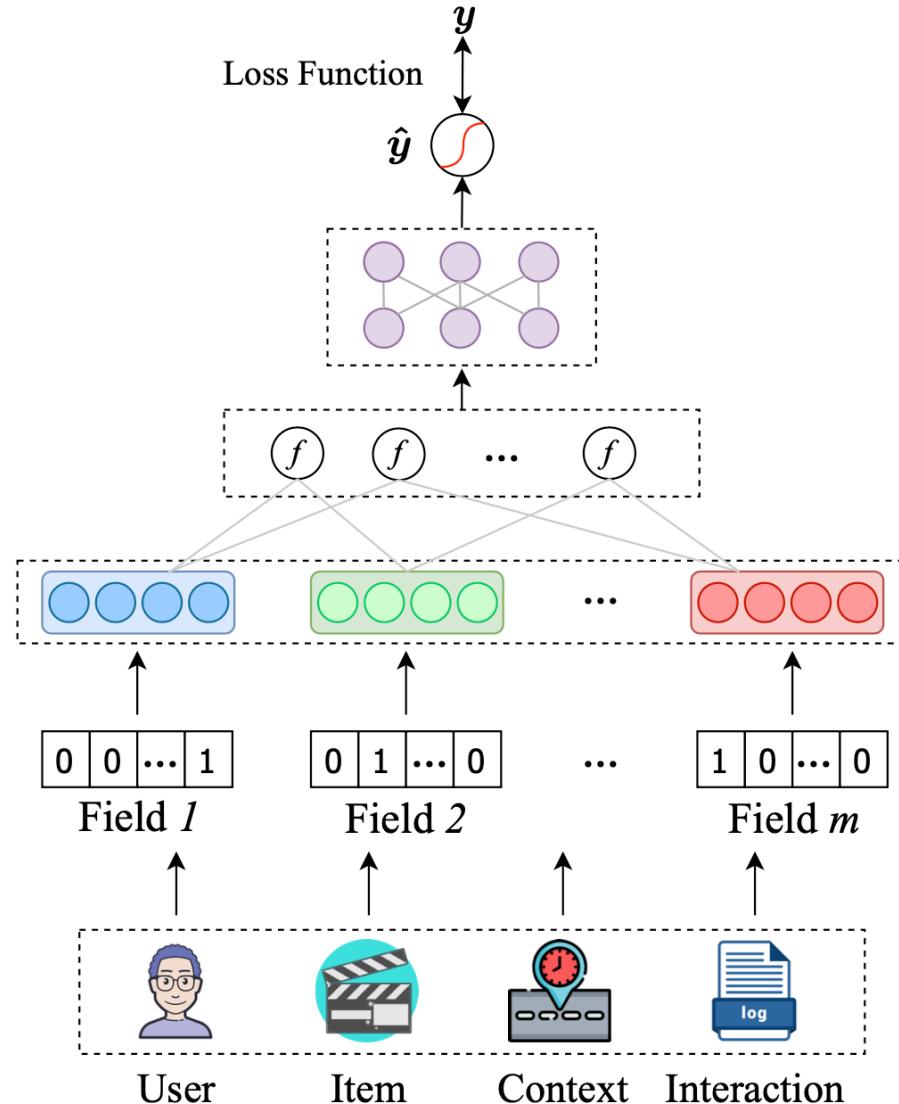
Feature Interaction Layer
Pooling, convolution, and the
number of layers, inner product,
outer product, convolution, etc.



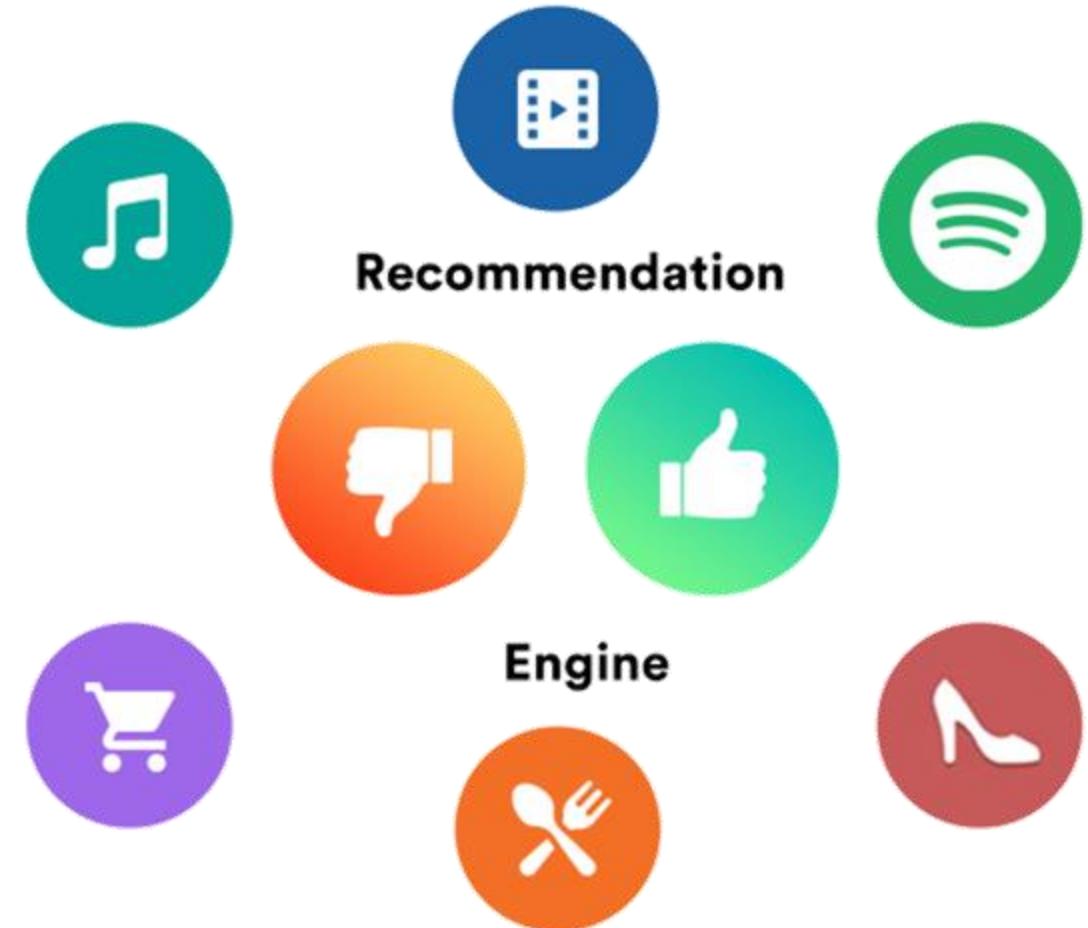
System Design

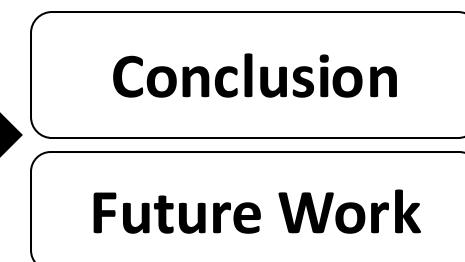
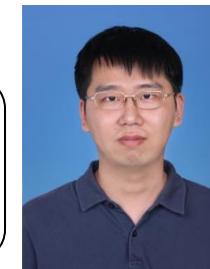
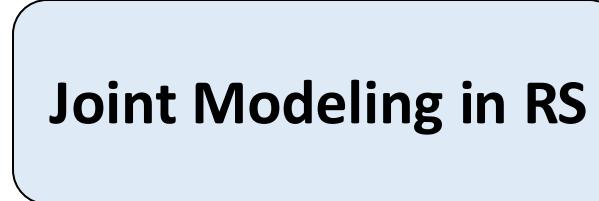
Hardware infrastructure,
data pipeline, information
transfer, implementation,
deployment, optimization,
evaluation, etc.

Deep Recommender Architecture



V.S.



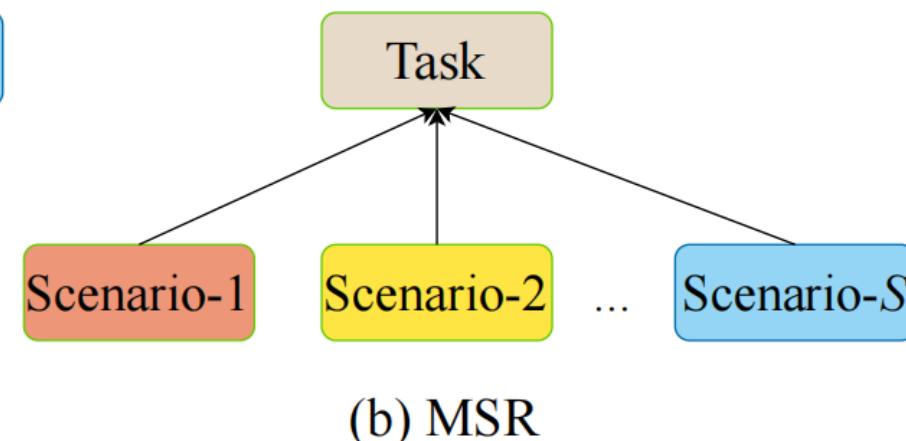
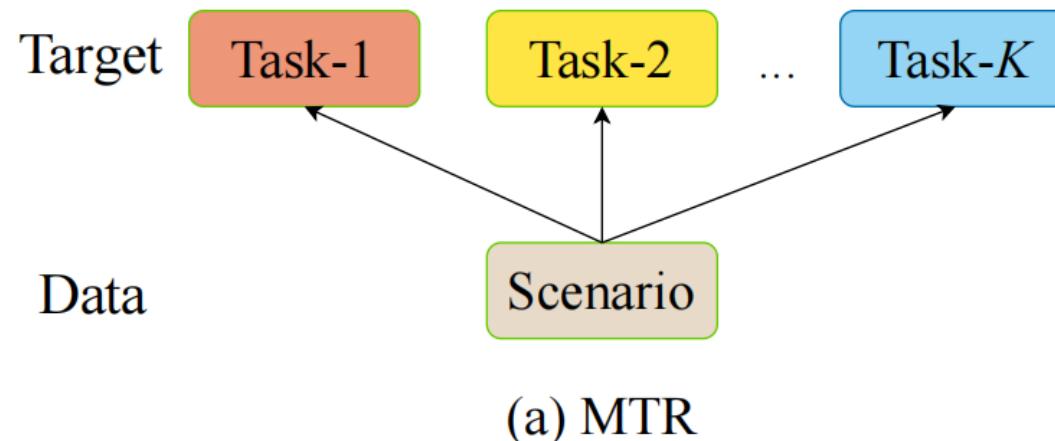


Yichao Wang

Joint Modeling in Recommendations



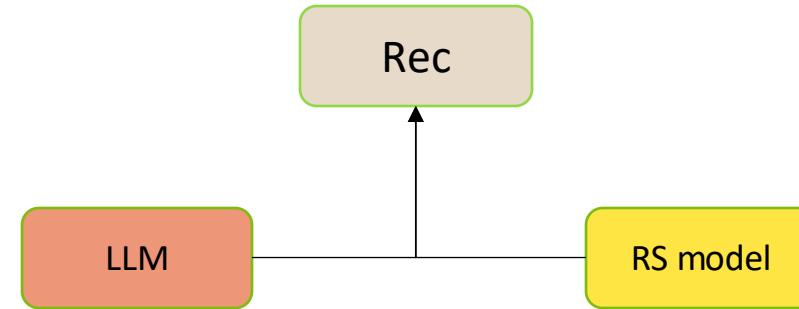
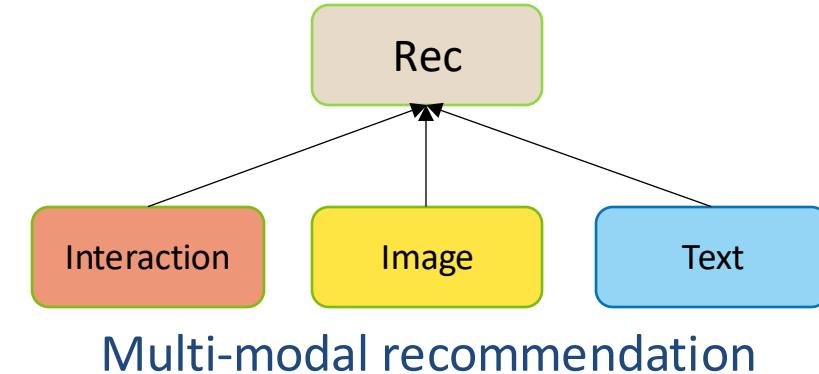
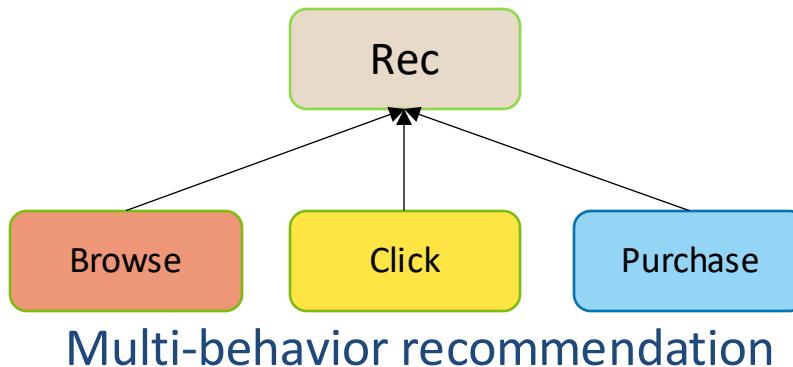
- Handling the inter-dependency between users and items under more complex circumstances
- Advantages
 - One model for several situations
 - Performance improvement caused by information sharing in different situations
- Two typical representatives:
 - Multi-task recommendation (MTR)
 - Multi-scenario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



Joint Modeling in Recommendations



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



Large language model-based recommendation

Why Joint Modeling ?



➤ Multi-Task Recommendation:

- Independent tasks: Comments, repost, likes, bookmarks
- Multi-stage conversion tasks: click, application, approval, activation ...

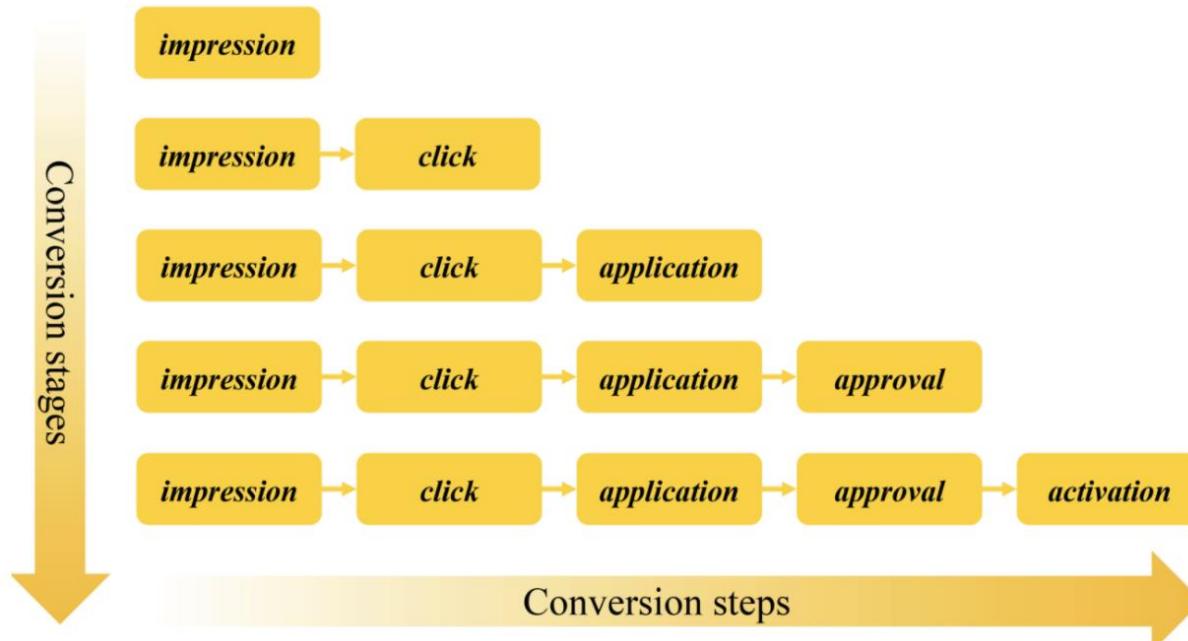


10:20 · 2023/7/31 · 15.3K Views

8 Retweets 1 Quote 13 Likes 3 Bookmarks



How to extract useful information from other tasks ?

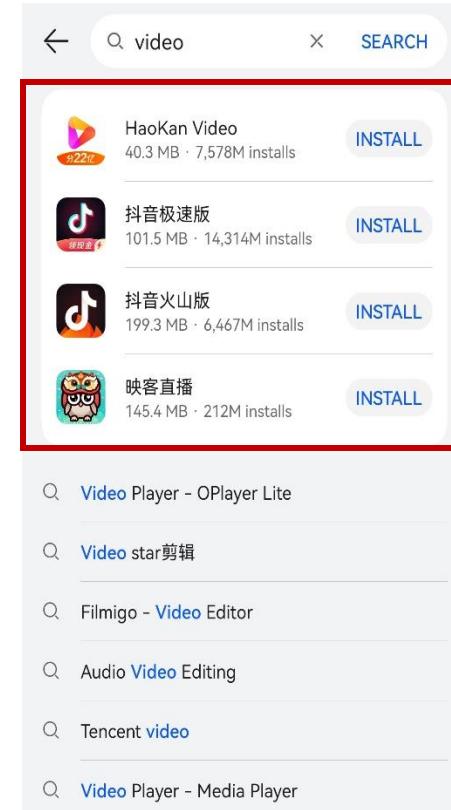


How to capture task dependences and resolve the sparsity issue ?

Why Joint Modeling ?



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

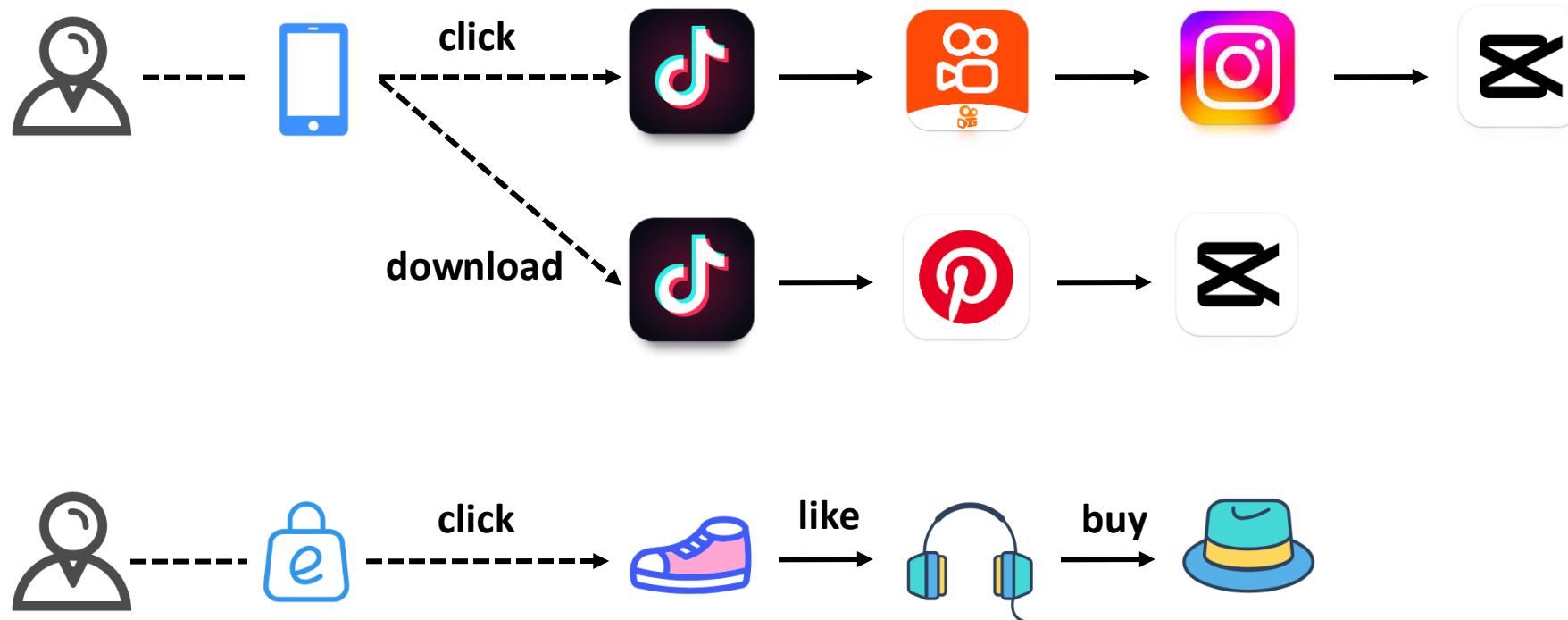


How to extract more comprehensive user portrait from interactions in different scenarios, and make recommendations based on the characteristics of the current scenario ?

Why Joint Modeling ?



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

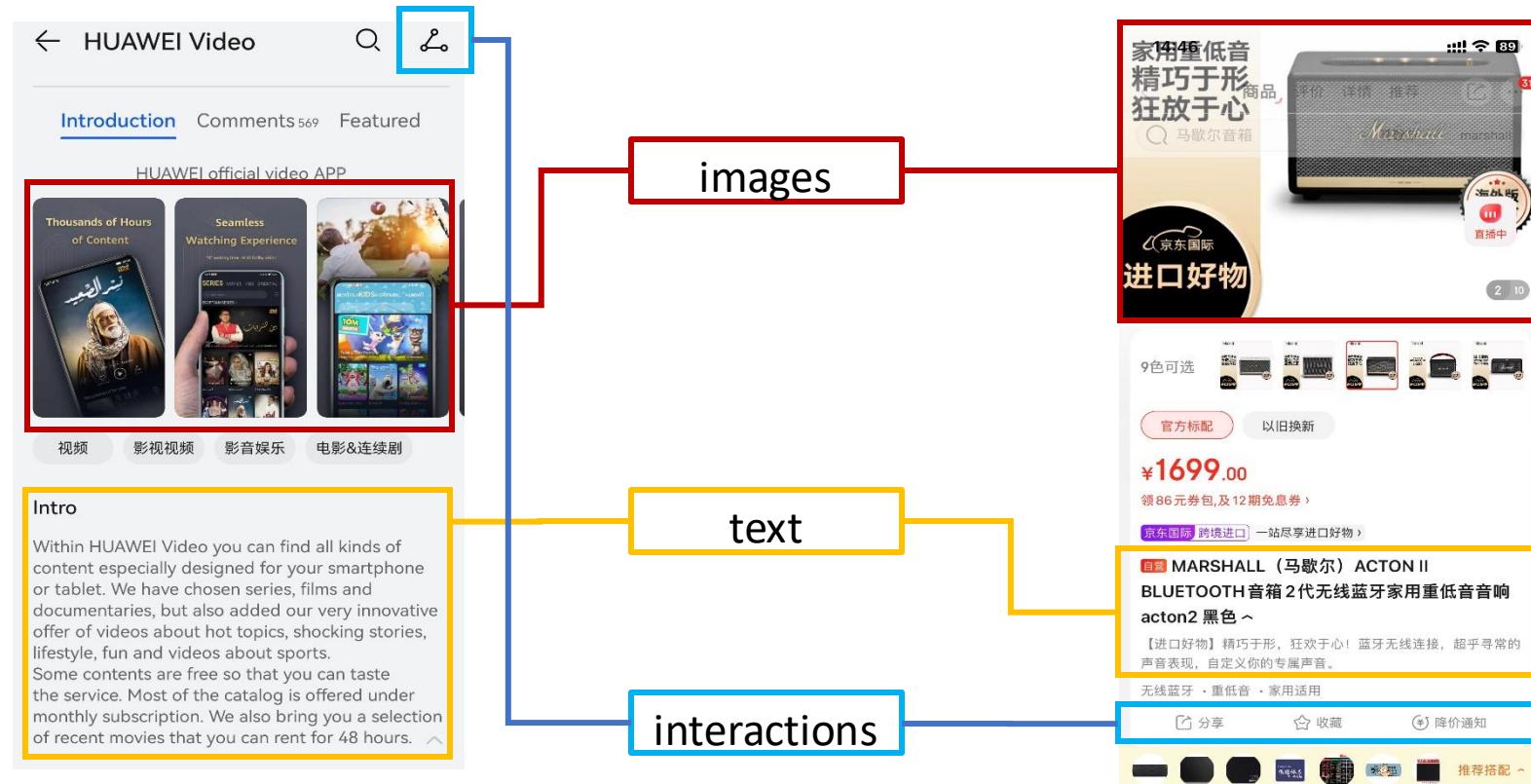


How to learn the relationship between different type of behaviors ?

Why Joint Modeling ?



➤ Multi-Modal Modeling: user interactions, images, text ...

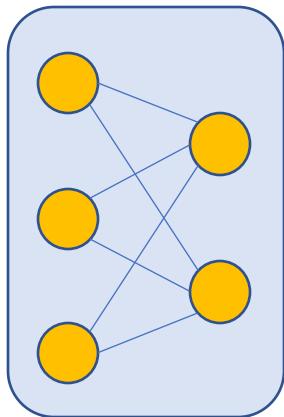


How to extract and align data from different modalities ?

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



Multi-Scenario



Task/scenario adaption

Multi-Task



$$wL(E^{\text{Merge}}, \theta^{sh}, \theta^t, \theta^s)$$

Joint Modeling



Representation extraction



Multi-Behavior



Multi-Modal

$$E^{\text{Merge}} = U(E, E^B, E^M)$$

Multi-behavior

$$E^B = G(H_1, H_2, \dots, H_N)$$

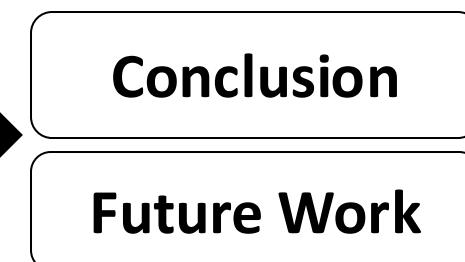
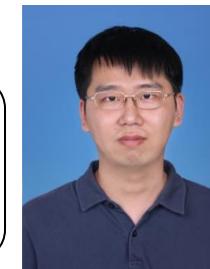
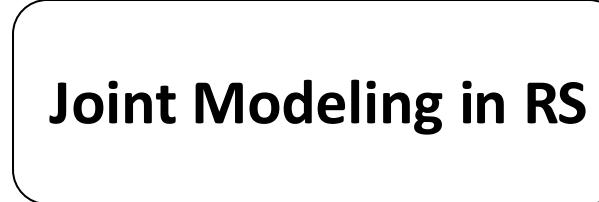
Multi-modal

$$wL(E^{\text{Merge}}, \theta^{sh}, \theta^t, \theta^s)$$

Multi-scenario

$$wL(E^{\text{Merge}}, \theta^{sh}, \theta^t, \theta^s)$$

Multi-task



Yichao Wang

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

- 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 Modeling



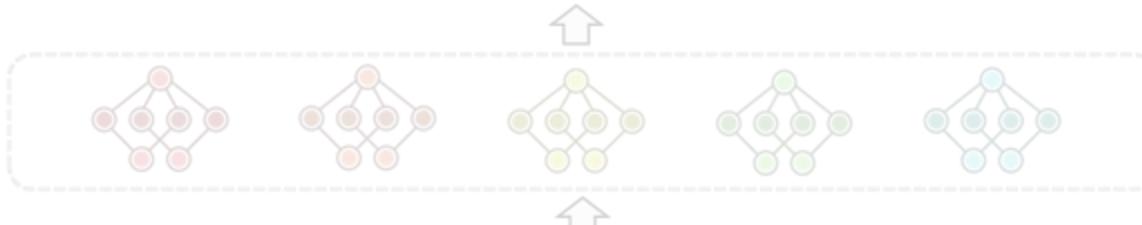
Multi-Scenario



Multi-Task



Task/scenario adaption

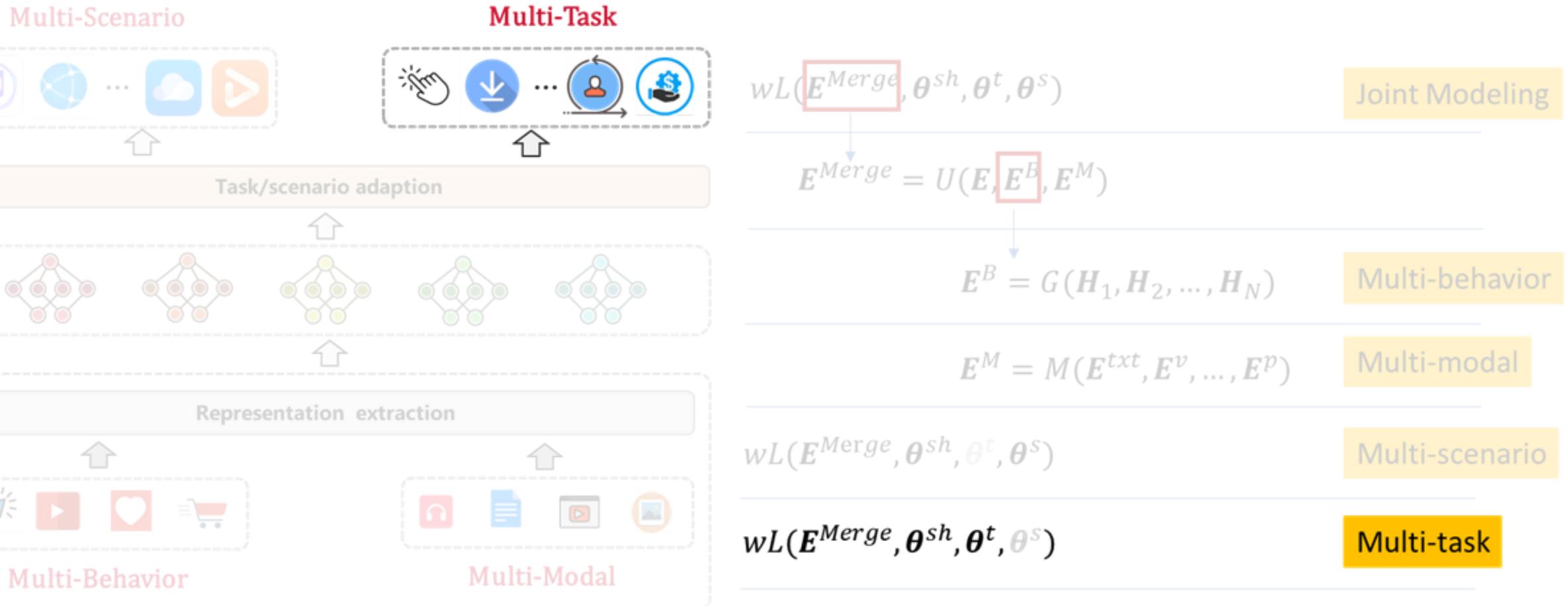


Representation extraction



Multi-Behavior

Multi-Modal



Joint Modeling

Multi-behavior

Multi-modal

Multi-scenario

Multi-task

➤ Problem:

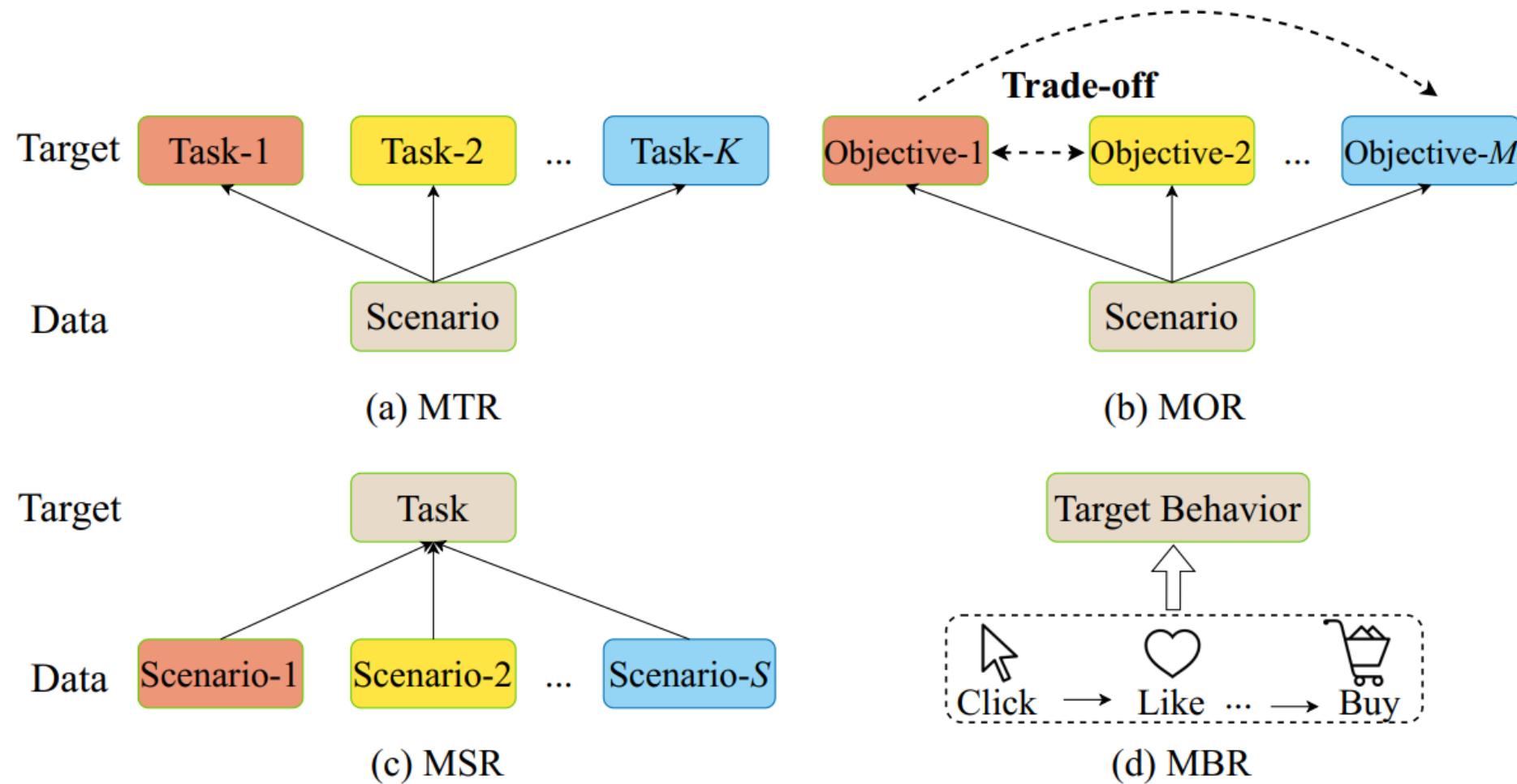
- Learning MTL model with task-specific parameters $(\theta^1, \dots, \theta^K)$ and shared parameter θ^s , which outputs the K task-wise predictions

➤ Optimization problem:

$$\arg \min_{\{\theta^1, \dots, \theta^K\}} \mathcal{L}(\theta^s, \theta^1, \dots, \theta^K) = \arg \min_{\{\theta^1, \dots, \theta^K\}} \sum_{k=1}^K \omega^k L^k(\theta^s, \theta^k)$$

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

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

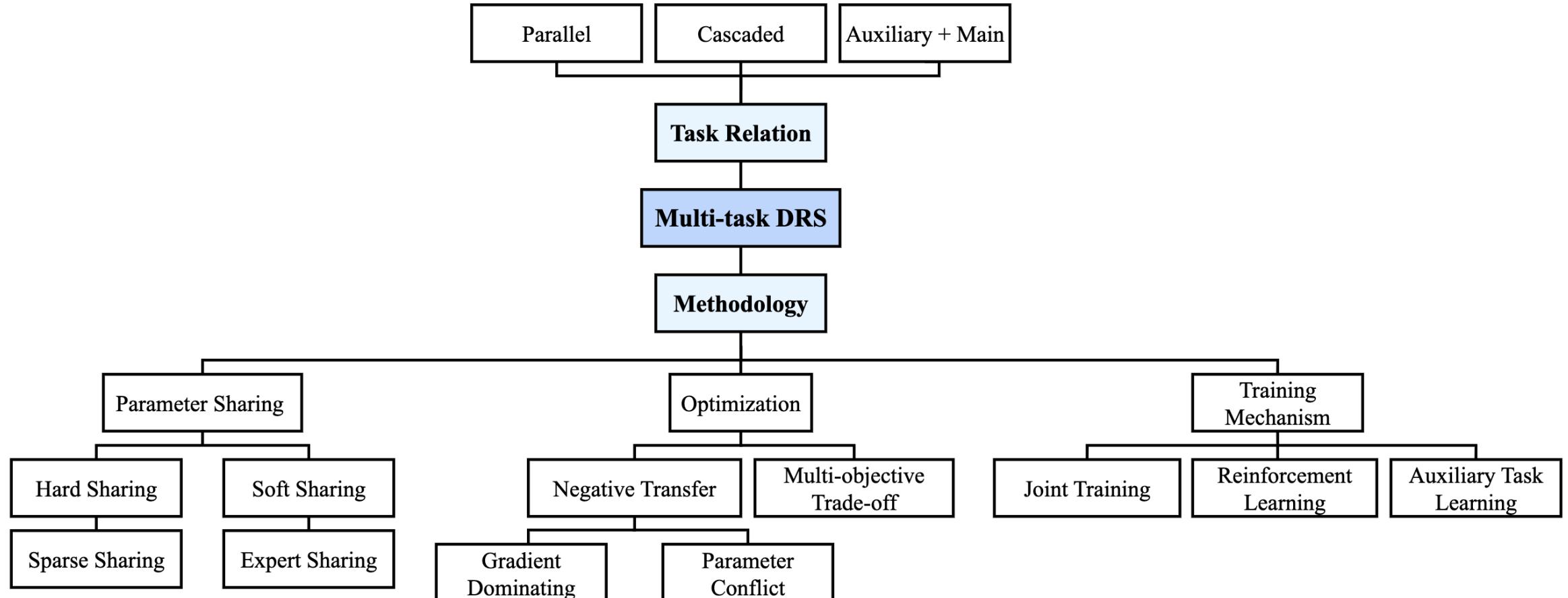


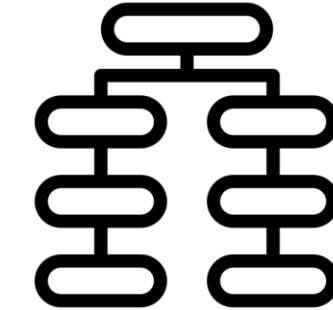
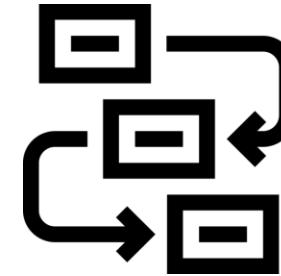
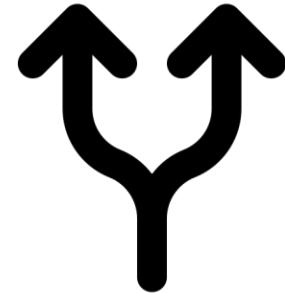
Comparison with CV & NLP



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

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

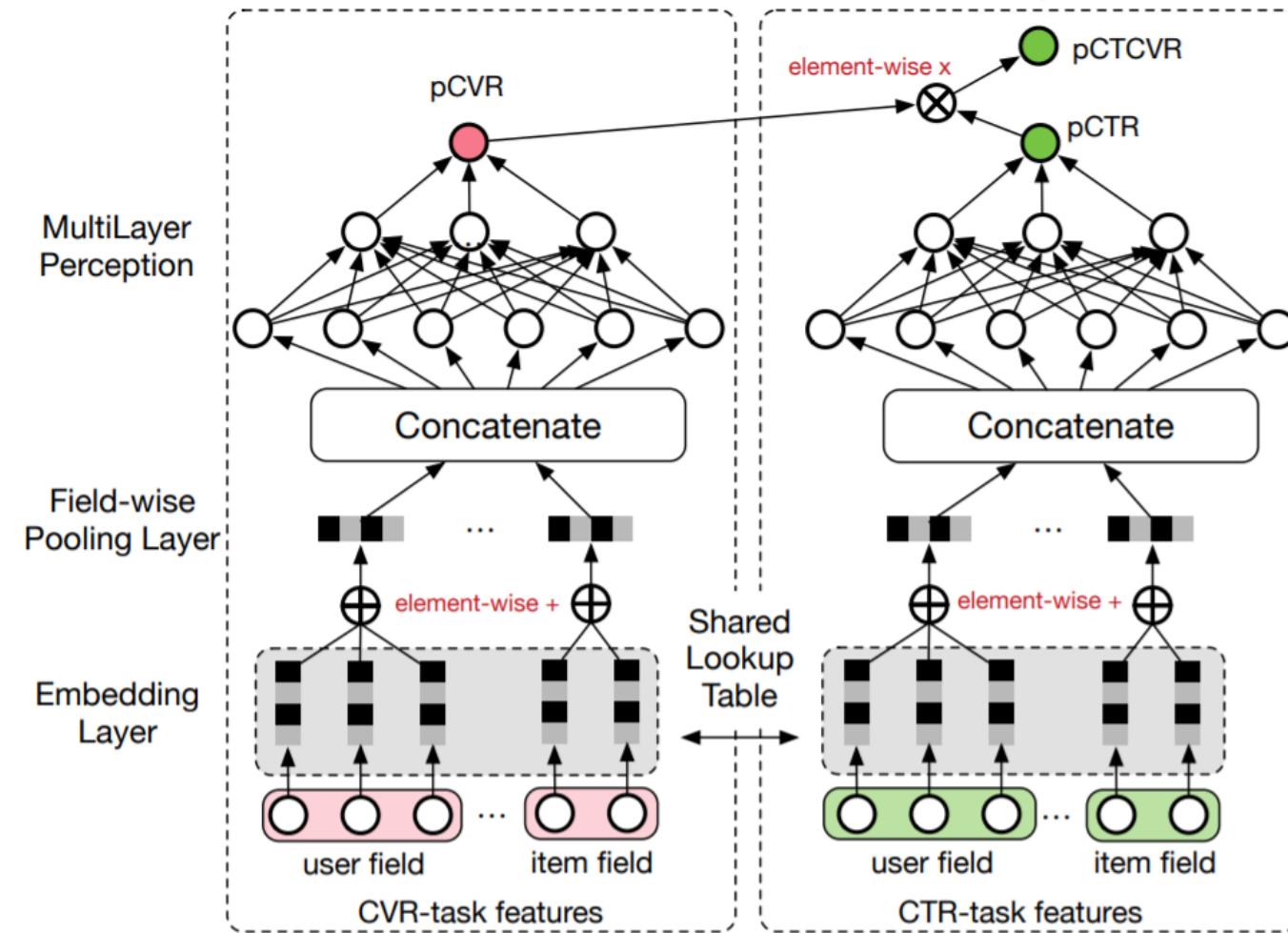
- Cascaded task relationship: **sequential dependency**
- Computation of current task depends on **previous ones**
 - E.g. CTCVR = CTR × CVR
- General formulation:

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

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

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

SSB: Sample Selection Bias DS: Data Sparsity



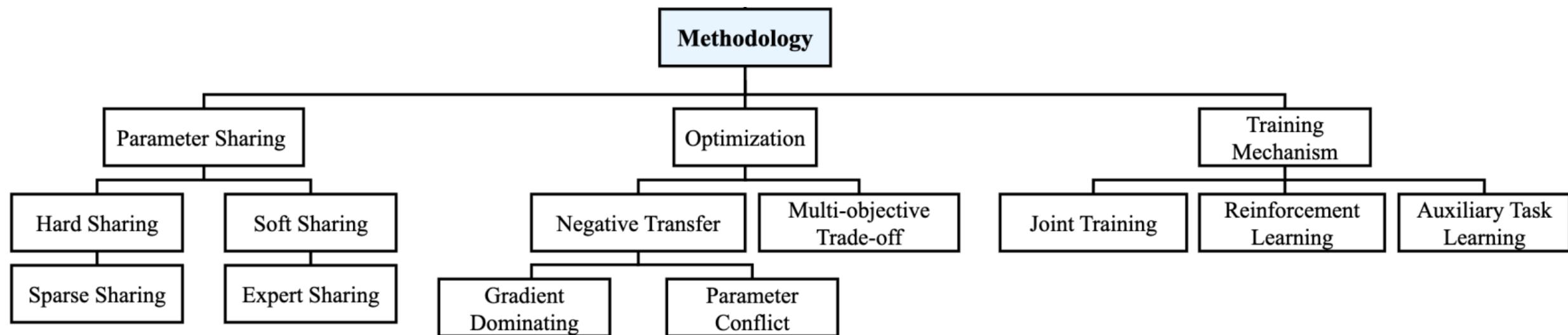
- A task specified as the main task while associated auxiliary tasks help to improve performance
- Probability estimation for main task  the probability of auxiliary tasks
- Provide richer information across entire space

Auxiliary with Main Task

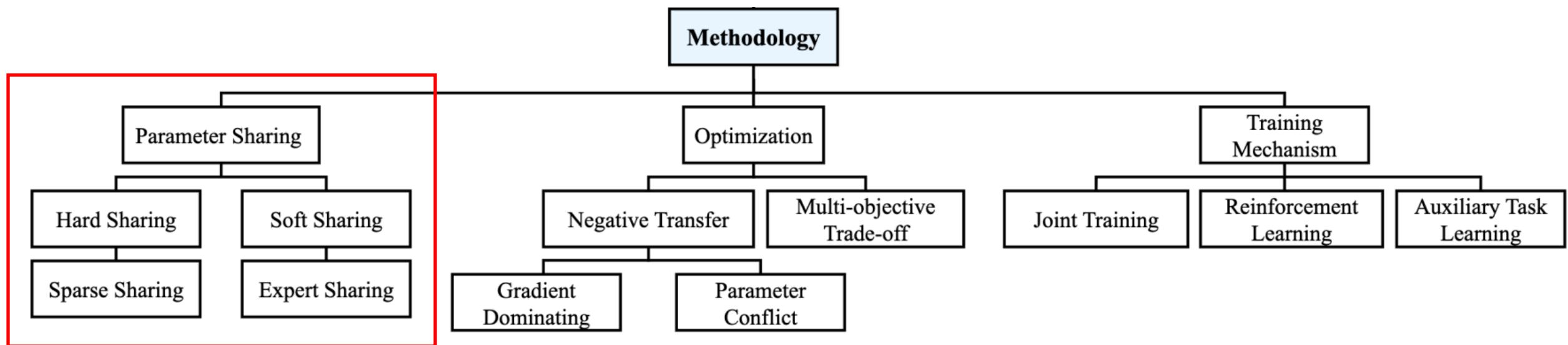


Model	References	Method
ESDF Multi-IPW and Multi-DR DMTL Metabalance	[Wang et al., 2020b] [Zhang et al., 2020] [Zhao et al., 2021] [He et al., 2022]	Adopt the original recommendation tasks as auxiliaries
MTRec PICO MTAE Cross-Distill	[Li et al., 2020a] [Lin et al., 2022] [Yang et al., 2021] [Yang et 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

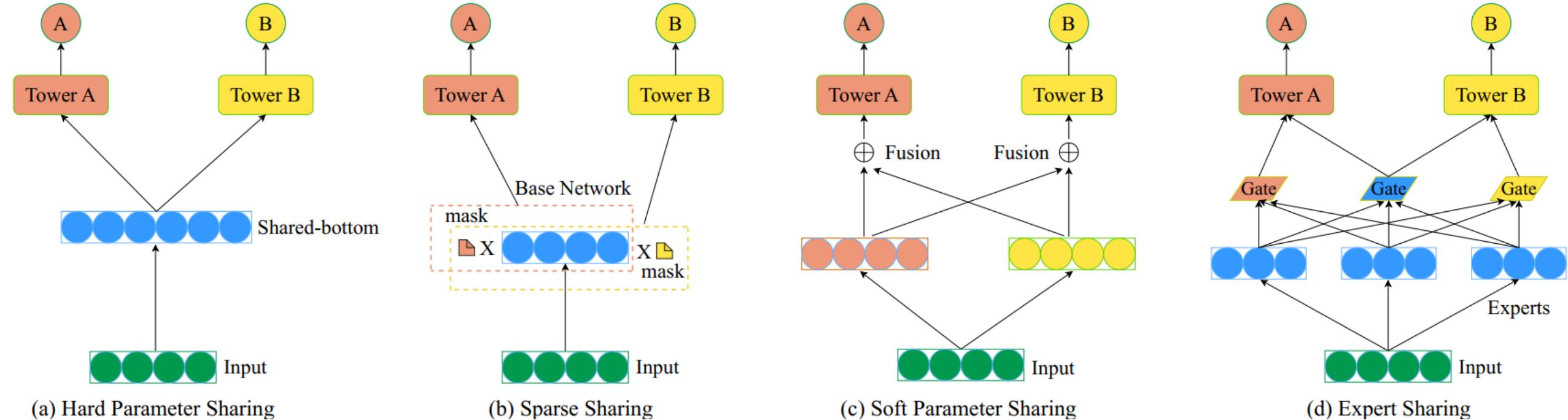
Methodology

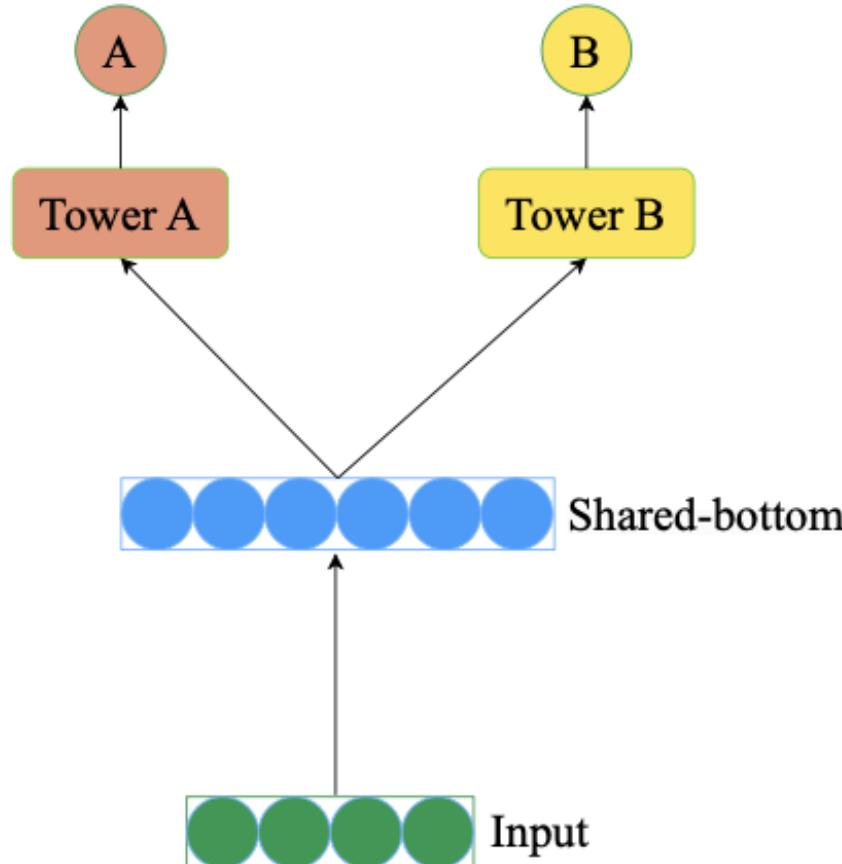


Parameter Sharing



Parameter Sharing

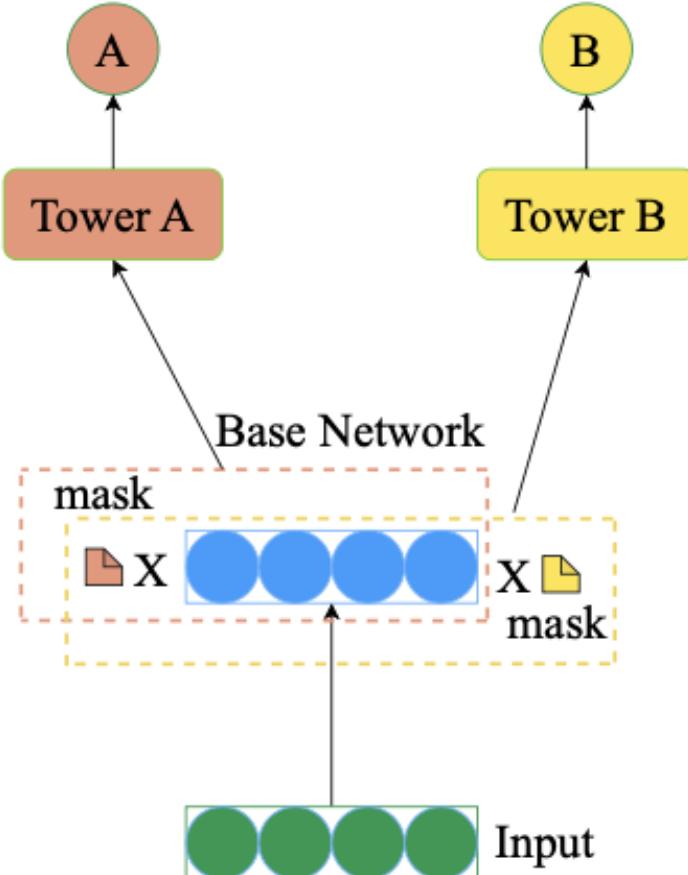




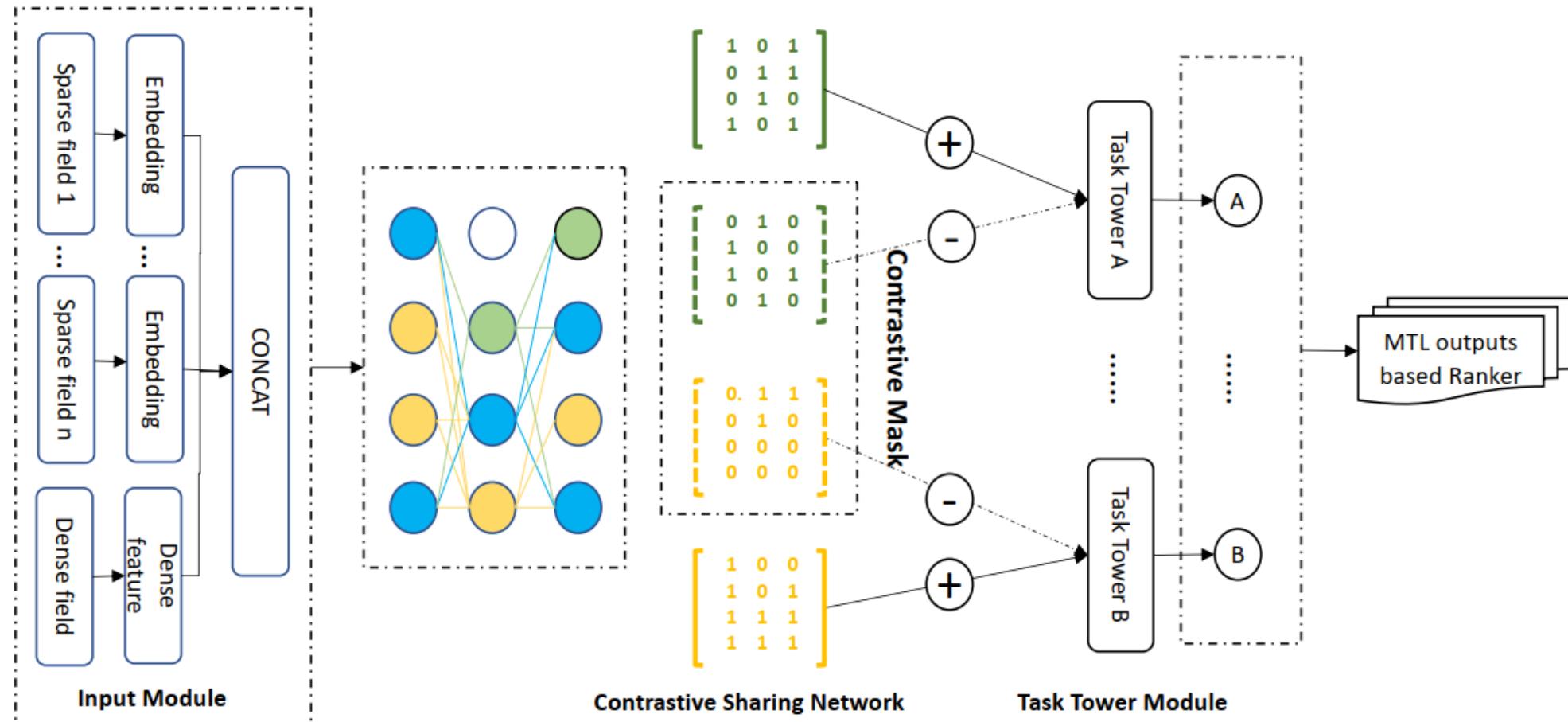
- Shared bottom layers extract the **same** information for different tasks,
- Task-specific top layers are trained individually

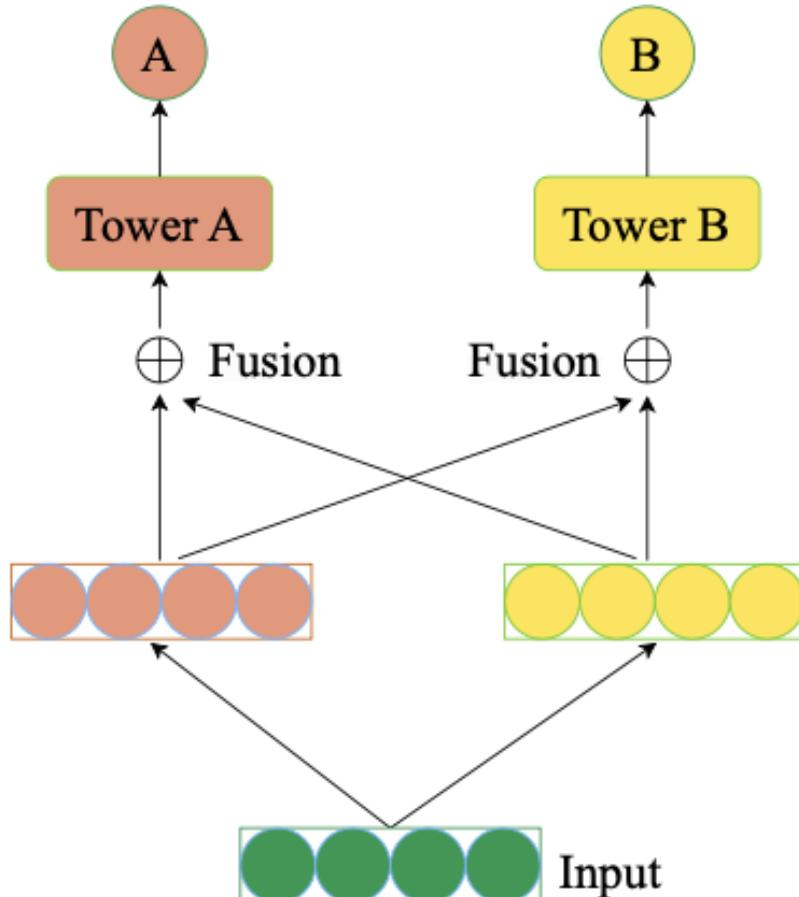
- ✓ Improving computation efficiency and alleviating over-fitting

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

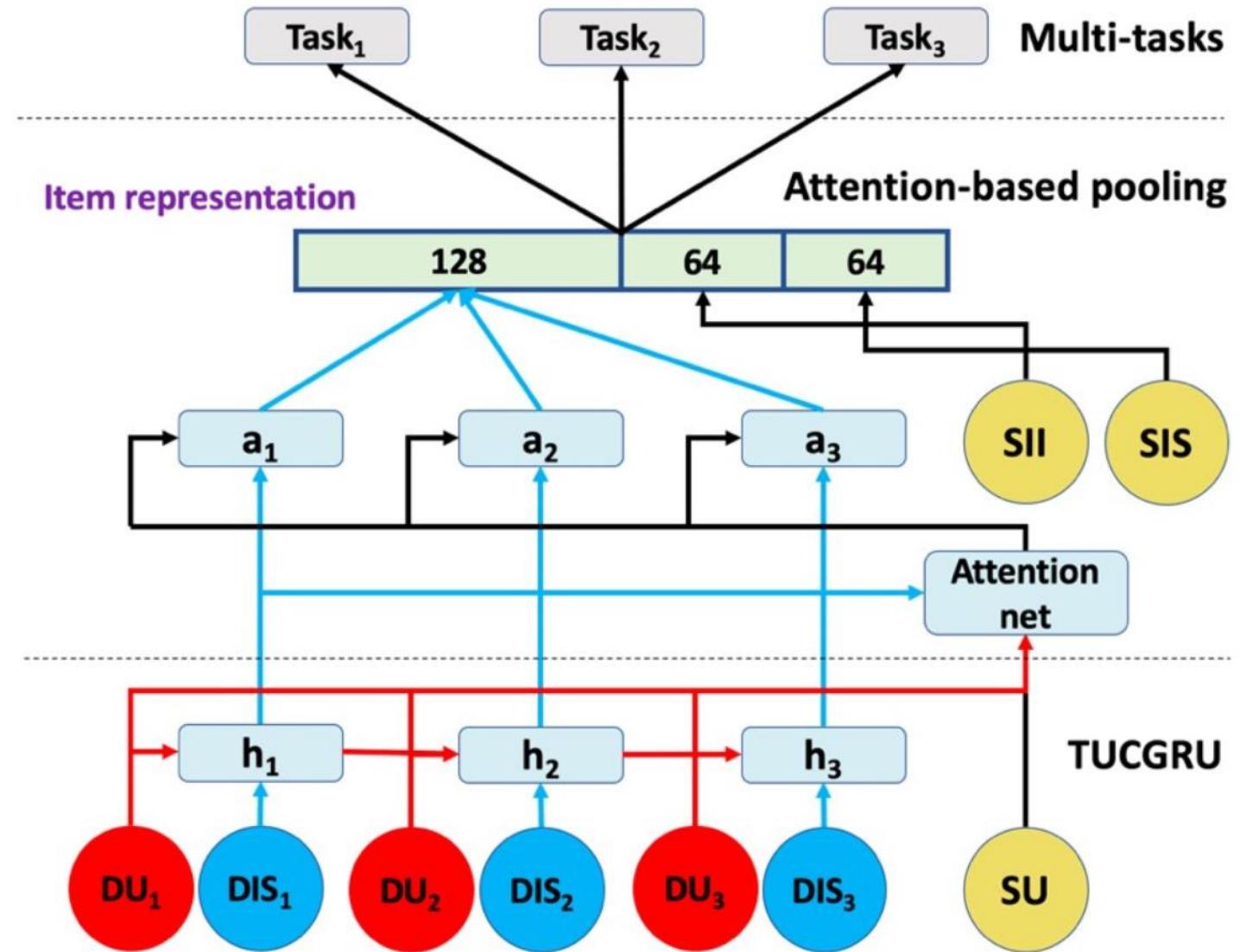


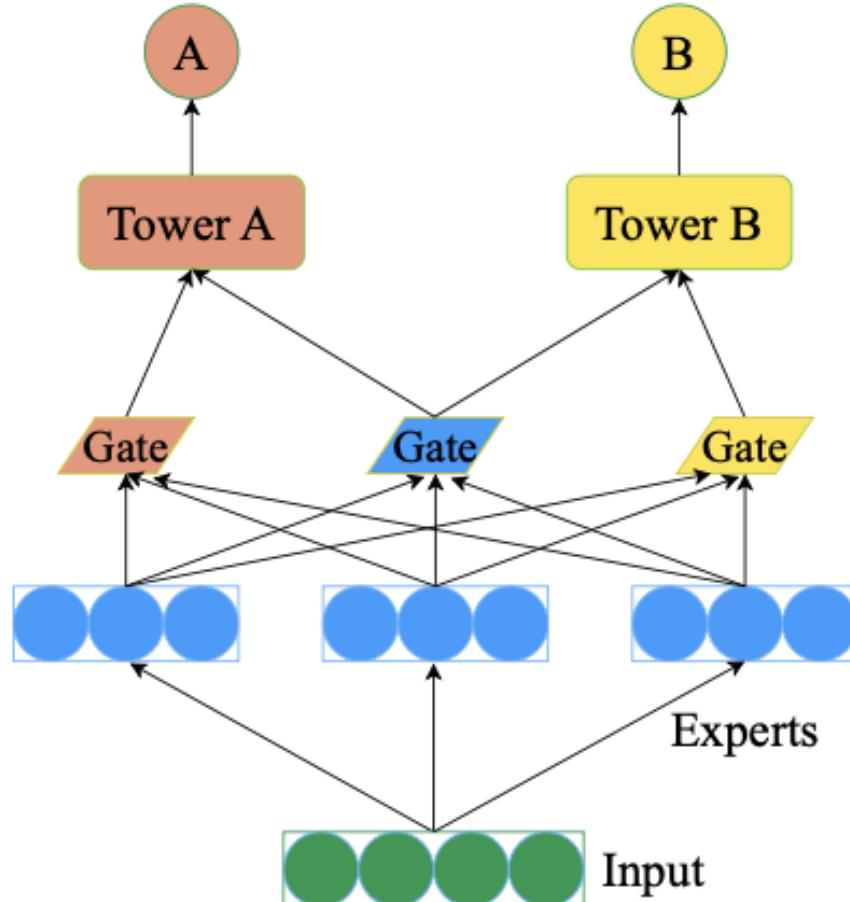
- Extracting **sub-networks** for each task by parameter masks from a base network
 - **Special case of Hard Sharing**
- ✓ Coping with the weakly related tasks flexibly
- ✗ Negative transfer when updating shared parameters



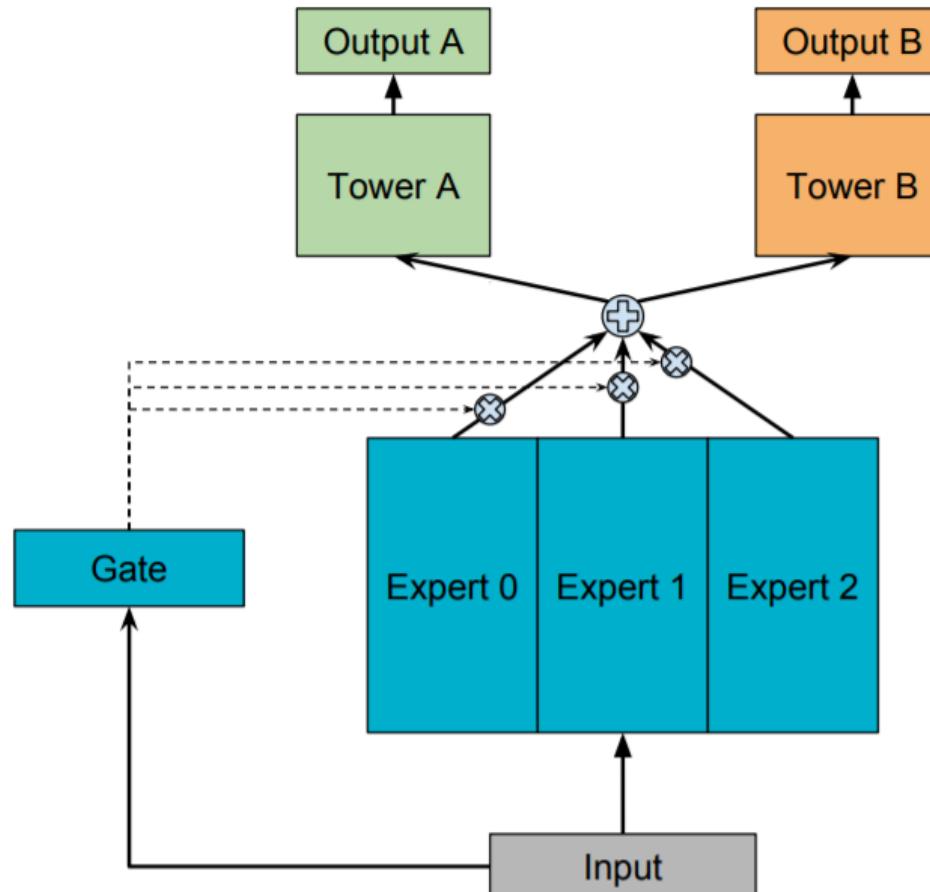


- Building separate models for tasks but the information among tasks is **fused by weights of task relevance**
- ✓ Relatively high **flexibility** in parameter sharing v.s. hard sharing
- ✗ Can not reconcile the flexibility
- ✗ Computation cost of the model

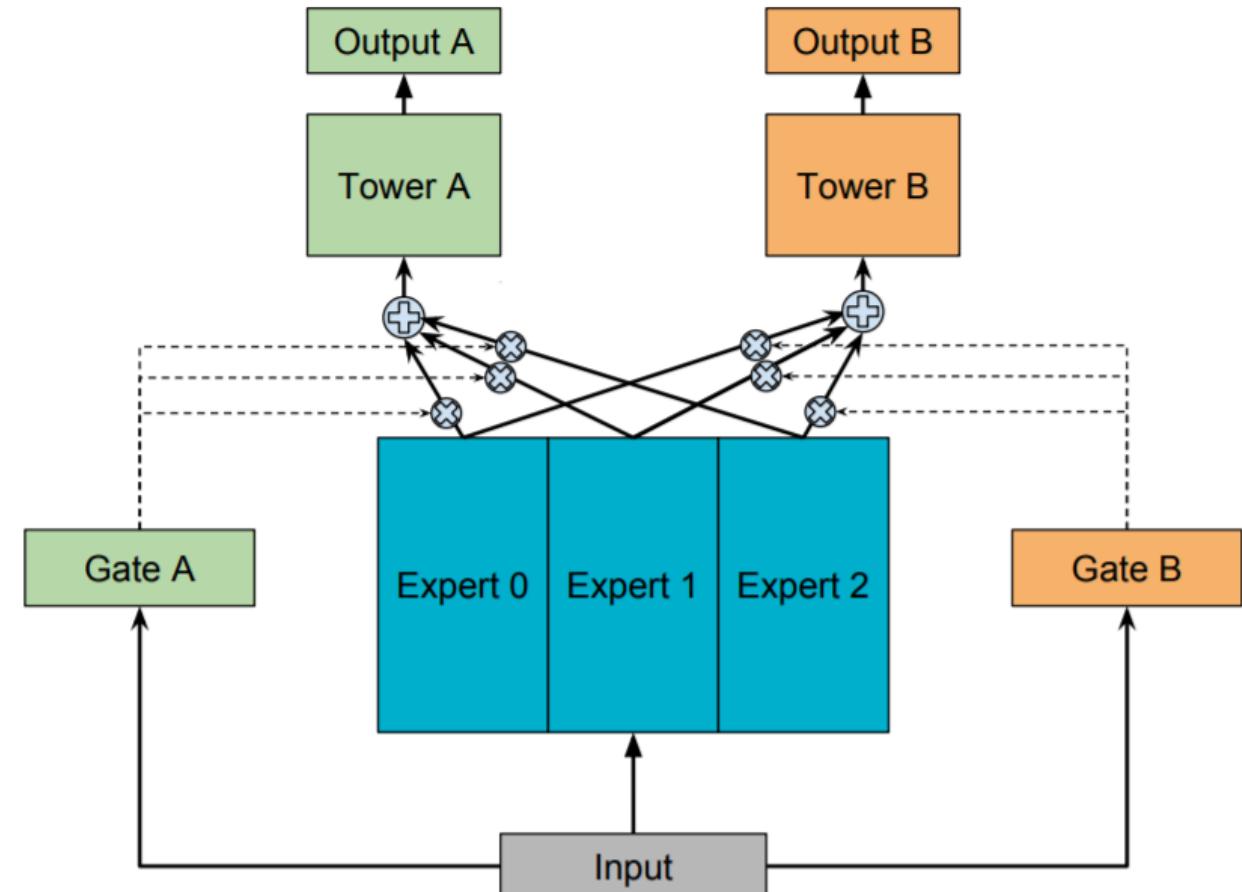




- Employing multiple **expert networks** to extract knowledge from shared bottom
 - Fed into **task-specific** modules like gates
 - Passed into the task-specific tower
- Mainly non-sequential input features
- **Special case** of Soft Sharing



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



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

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

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]
MoME	[Xu et al., 2024]
MoSE	[Qin et al., 2020]



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

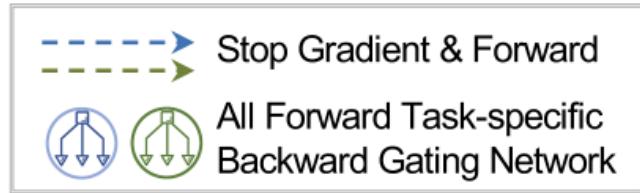
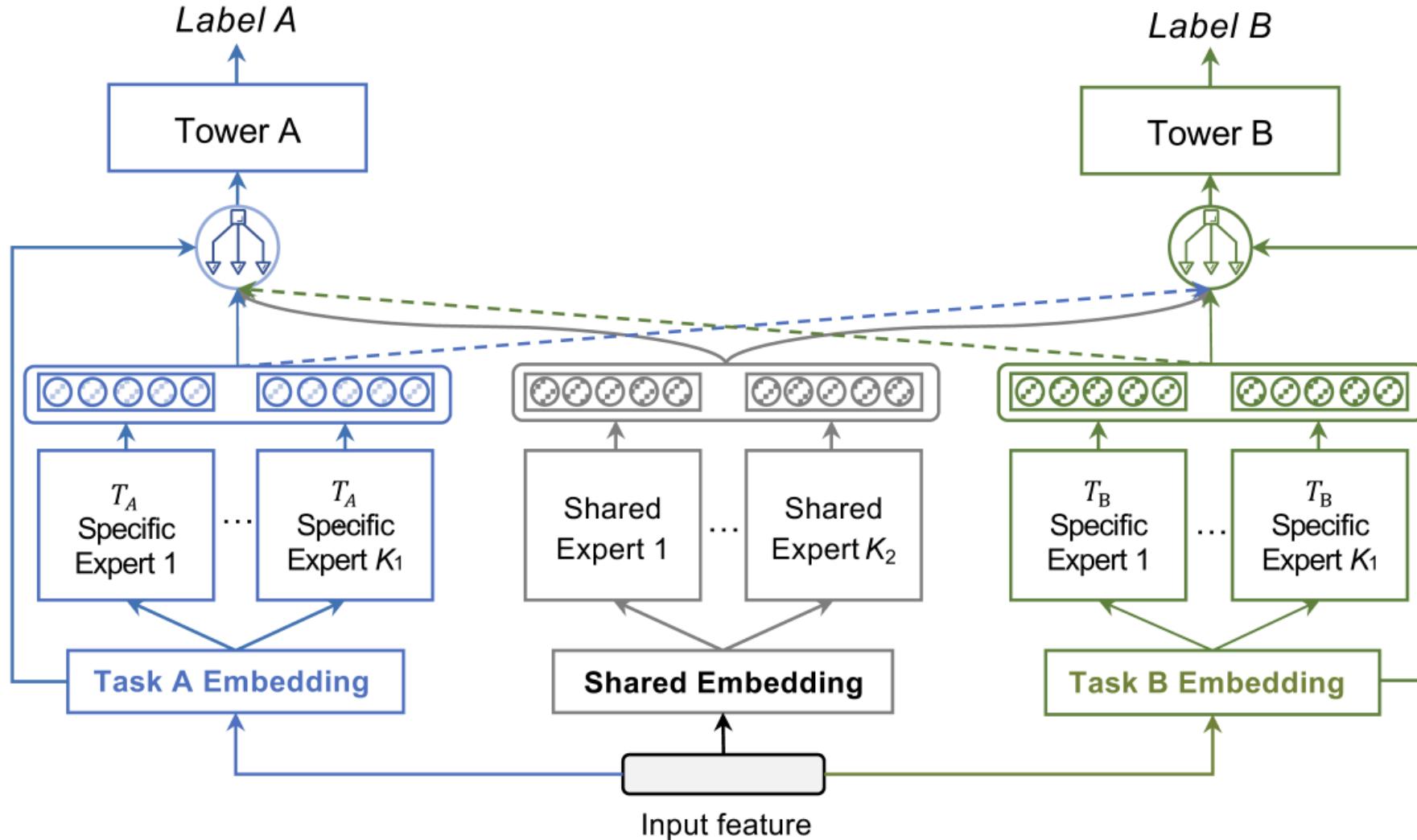


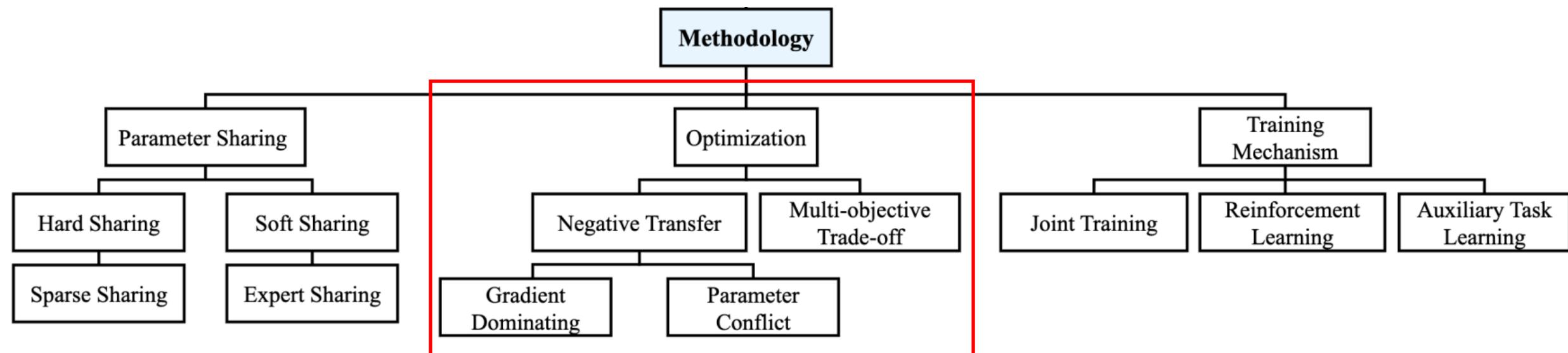
Processing **sequential** input features utilizing LSTM & sequential experts

Special Case: Multi-Embedding Paradigm



STEM (Shared and Task-specific EMbeddings)



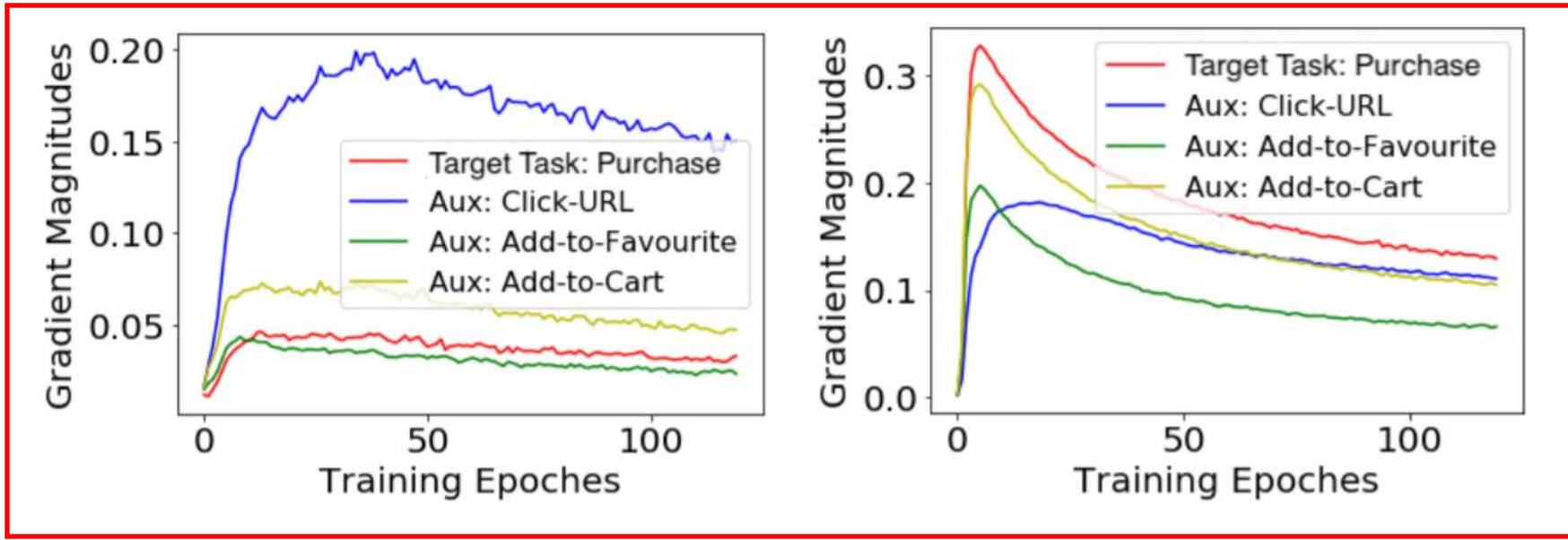


Gradient dominating $\|\nabla_{\theta} L^k(\theta)\|$

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

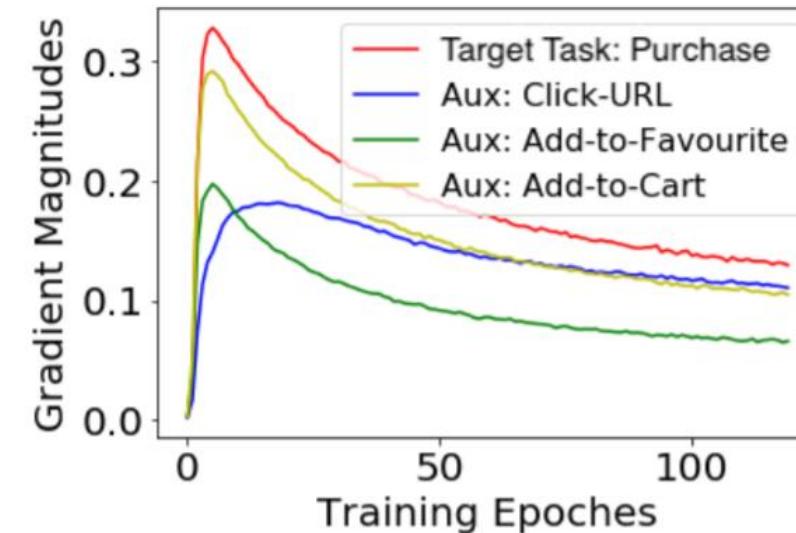
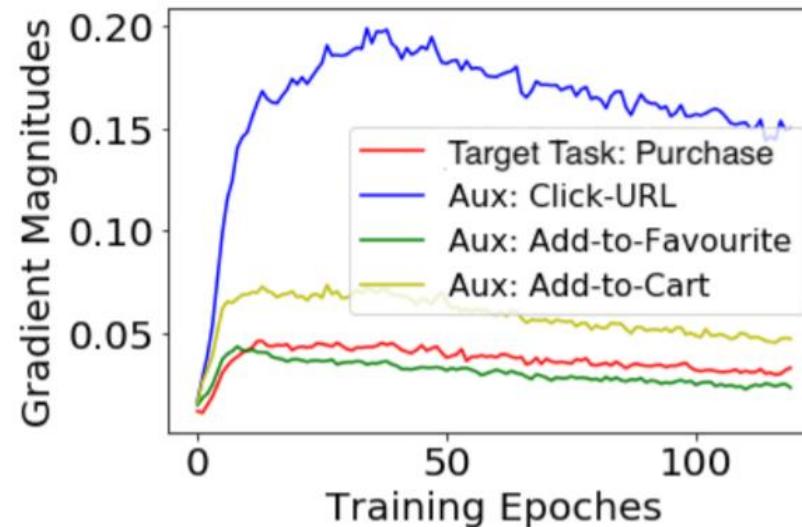
Opposite directions of gradient + - $\nabla_{\theta} L^k(\theta)$

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
GradCraft [Bai et al., 2024]	Adjusting gradient norm and deconflicting global direction through projection and combination



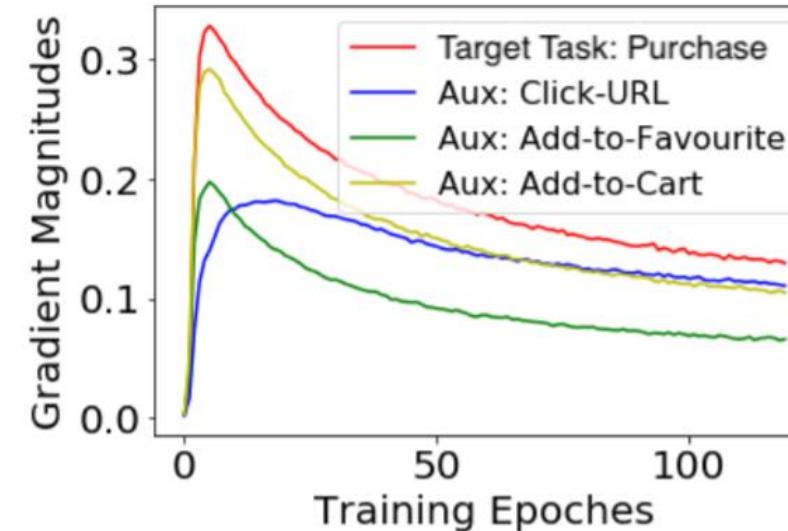
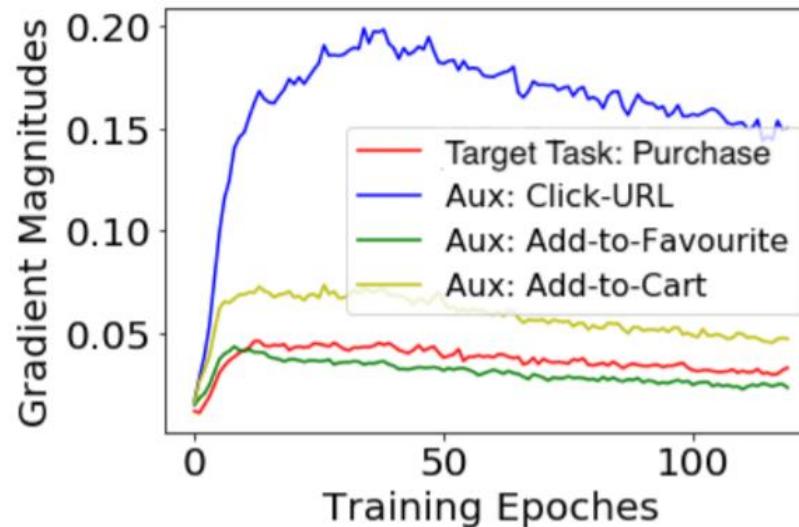
$$\theta^{t+1} = \theta^t - \alpha * \mathbf{G}_{total}^t$$

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



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

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



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

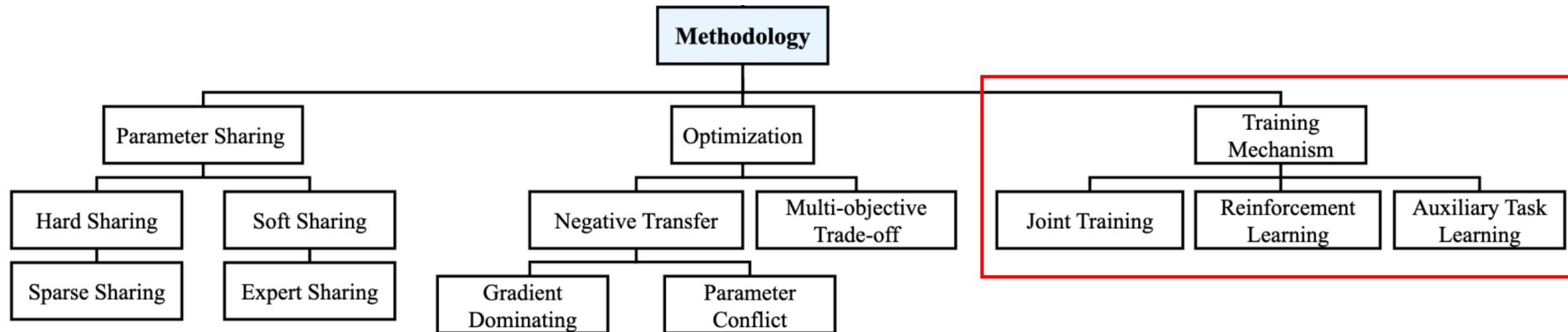
$$\mathbf{G}_{total}^t = \nabla_{\theta} \mathcal{L}_{total}^t = \nabla_{\theta} \mathcal{L}_{tar}^t + \sum_{i=1}^K \nabla_{\theta} \mathcal{L}_{aux,i}^t$$

$$\mathbf{G}_{aux,i}^t \leftarrow (\mathbf{G}_{aux,i}^t * \frac{\|\mathbf{G}_{tar}^t\|}{\|\mathbf{G}_{aux,i}^t\|}) * r + \mathbf{G}_{aux,i}^t * (1 - r)$$


Objectives optimized regardless of the **potential conflict**

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

Training process & Learning strategy



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]

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]

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]

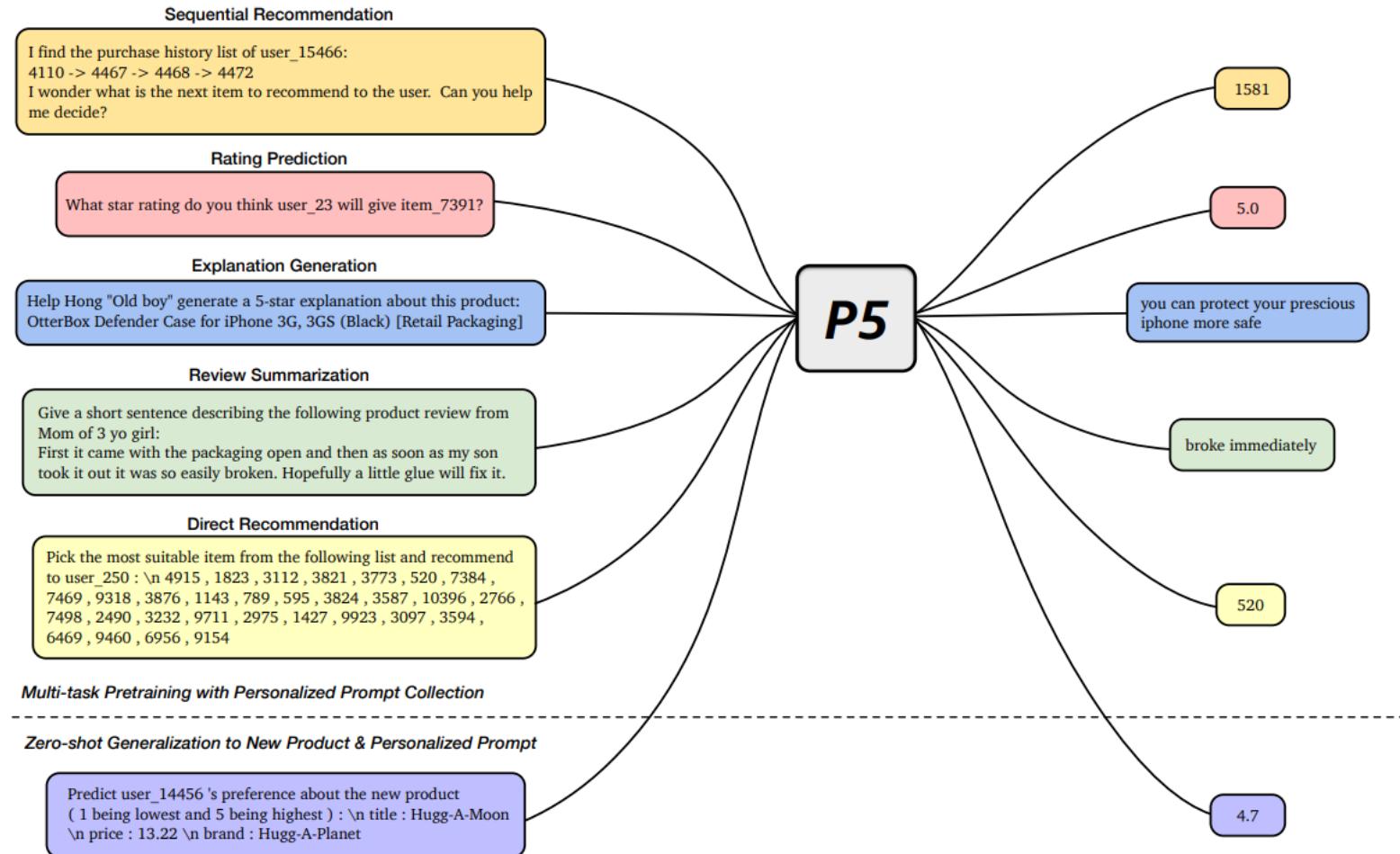
- **E-commerce** : Main focus
- **Advertising**
 - **Utility & Cost**
 - i. MM-DFM [Hou et al., 2021]: Performing multiple conversion prediction tasks in different observation duration
 - ii. MetaHeac [Zhu et al., 2021]: Handling audience expansion tasks on content-based mobile marketing
 - iii. MVKE [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

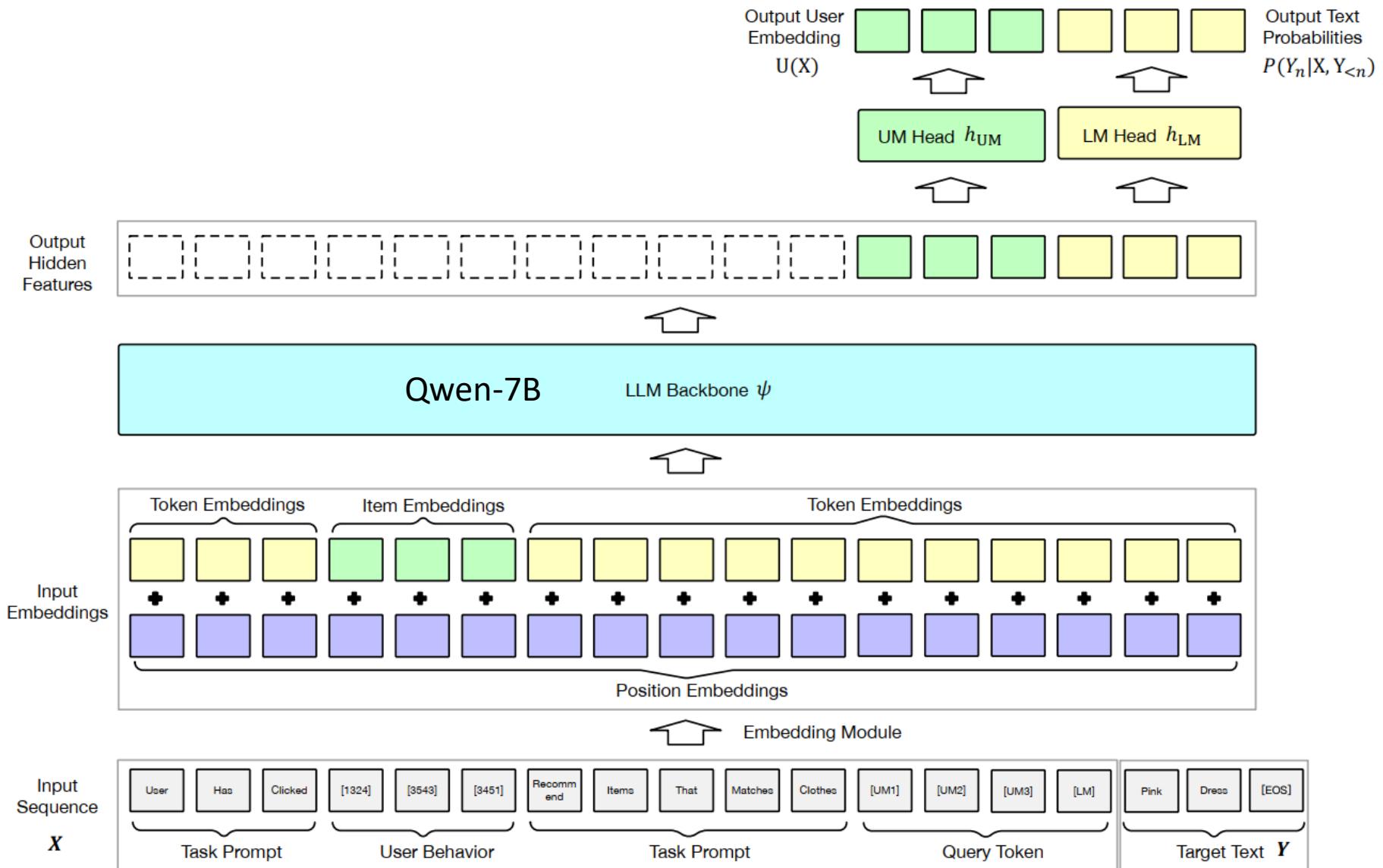
Datasets



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/

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





Prompt Template



Multi-scenario Recommendation: The items the user has recently clicked on are as follows: {USER BEHAVIOR SEQUENCE}. In scenario {SCENE}, please recommend items.

Multi-objective Recommendation: The items the user has recently clicked on are as follows: {USER BEHAVIOR SEQUENCE}. Please find items that the user will {ACTION}.

Long-tail Item Recommendation: The items the user has recently clicked on are as follows: {USER BEHAVIOR SEQUENCE}. Please recommend long-tail items.

Serendipity Recommendation: The items the user has recently clicked on are as follows: {USER BEHAVIOR SEQUENCE}. Please recommend some new item categories.

Long-term Recommendation: The items the user has recently clicked on are as follows: {USER BEHAVIOR SEQUENCE}. Please find items that match the user's long-term interests.

Search Problem: The items the user has recently clicked on are as follows: {USER BEHAVIOR SEQUENCE}. Please recommend items that match {QUERY}.

Inputs: The items the user has recently clicked on are as follows: [7502][8308][8274][8380]. Please recommend items that match *Clothes*. [UM][LM]

Target Text: Swimwear & Beachwear for the Summer;
Casual Dresses for Every Occasion.

Target Items: [3632][1334]

➤ Multi-task Recommendation + Language Model

Model	Setting	Methods
P5	MTR+PLM	Prompt design;SFT
M6-Rec	MTR+PLM	Prompt design;SFT
UniMIND	MTR+PLM	Prompt design;SFT
URM	MTR+LLM	Prompt design;SFT
LUM	MTR+LLM	Next condition-item prediction

Challenges & Future Directions



Topic	Challenge & future direction
Negative Transfer	<ul style="list-style-type: none">• Extra complex inter-task correlation• What, where, and when to transfer to alleviate negative transfer
AutoML	<ul style="list-style-type: none">• Existing models only focus on the parameter sharing routing, while other components and hyper-parameters still under-explored
Explainability	<ul style="list-style-type: none">• Complex task relevance
Task-specific Biases	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• Extra complex inter-task correlation• What, where, and when to transfer to alleviate negative transfer
AutoML	<ul style="list-style-type: none">• Existing models only focus on the parameter sharing routing, while other components and hyper-parameters still under-explored
Explainability	<ul style="list-style-type: none">• Complex task relevance
Task-specific Biases	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• Extra complex inter-task correlation• What, where, and when to transfer to alleviate negative transfer
AutoML	<ul style="list-style-type: none">• Existing models only focus on the parameter sharing routing, while other components and hyper-parameters still under-explored
Explainability	<ul style="list-style-type: none">• Complex task relevance
Task-specific Biases	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• Extra complex inter-task correlation• What, where, and when to transfer to alleviate negative transfer
AutoML	<ul style="list-style-type: none">• Existing models only focus on the parameter sharing routing, while other components and hyper-parameters still under-explored
Explainability	<ul style="list-style-type: none">• Complex task relevance
Task-specific Biases	<ul style="list-style-type: none">• Most existing models only focus on one specific bias• Multiple bias should be tackled in future

- Task relation:
Parallel, Cascaded, Auxiliary with Main

- Methodology:
Parameter Sharing, Optimization, Training Mechanism

<https://arxiv.org/abs/2302.03525>

Multi-Task Deep Recommender Systems: A Survey

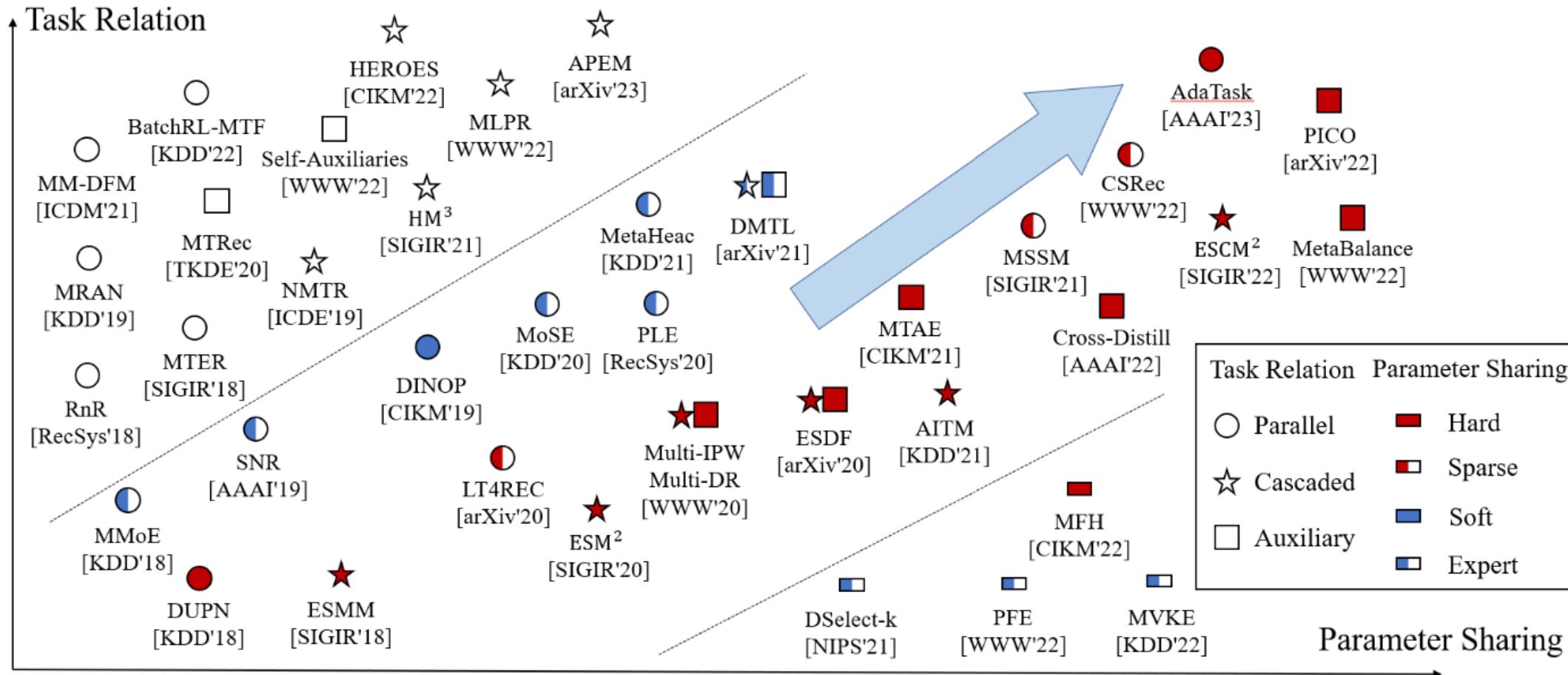
YUHAO WANG*, HA TSZ LAM*, and YI WONG*, City University of Hong Kong

ZIRU LIU, City University of Hong Kong

XIANGYU ZHAO[†], City University of Hong Kong

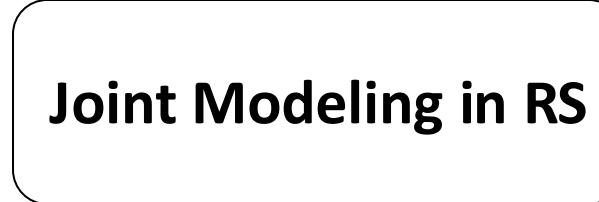
YICHAO WANG, BO CHEN, HUIFENG GUO, and RUIMING TANG[†], Huawei Noah's Ark Lab

Trend of MTDRS





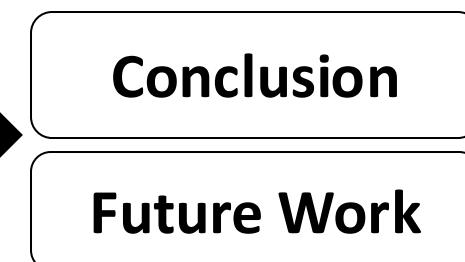
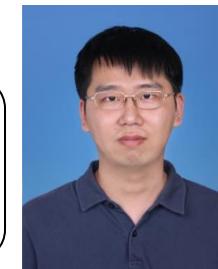
Yejing Wang



Pengyue Jia



Pengyue Jia Xiaopeng Li



Yichao Wang

➤ 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

Multi-Scenario Modeling



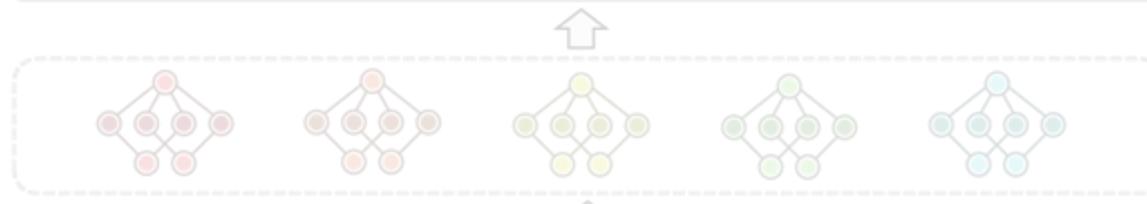
Multi-Scenario



Multi-Task



Task/scenario adaption



Representation extraction



Multi-Behavior

Multi-Modal

$$wL(\mathbf{E}^{Merge}, \theta^{sh}, \theta^t, \theta^s)$$

$$\mathbf{E}^{Merge} = U(\mathbf{E}, \mathbf{E}^B, \mathbf{E}^M)$$

$$\mathbf{E}^B = G(H_1, H_2, \dots, H_N)$$

$$\mathbf{E}^M = M(E^{txt}, E^v, \dots, E^p)$$

$$wL(\mathbf{E}^{Merge}, \theta^{sh}, \theta^t, \theta^s)$$

$$wL(\mathbf{E}^{Merge}, \theta^{sh}, \theta^t, \theta^s)$$

Joint Modeling

Multi-behavior

Multi-modal

Multi-scenario

Multi-task

θ^{sh} : shared parameters across scenarios
 θ^s : scenarios parameters of modeling

➤ What is Scenario?

- Homepage, Searching page, Detailed page ...
- Food, Leisure and entertainment, ...
- Usually refers to different business scenarios

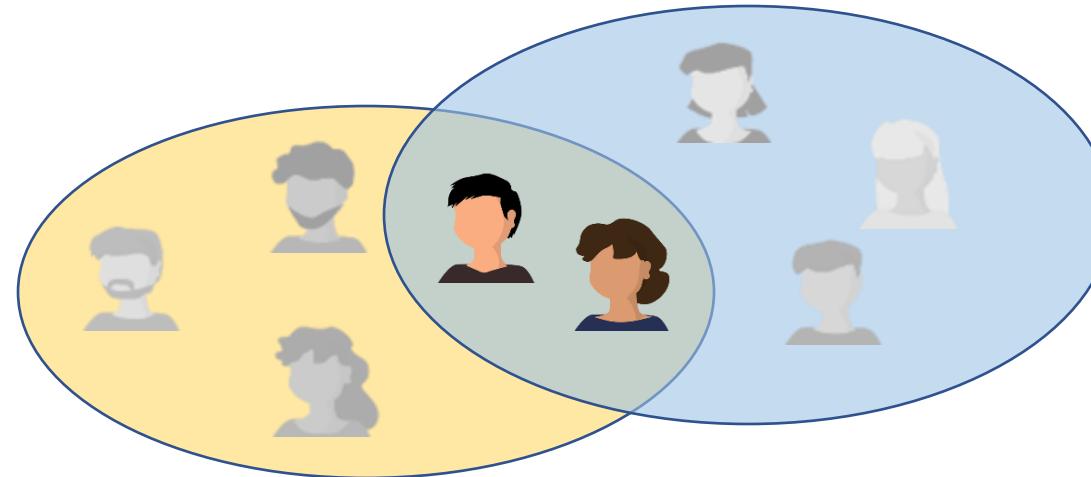
➤ Scenario and Domain?

- Generally do not make a distinction
- The same in this tutorial

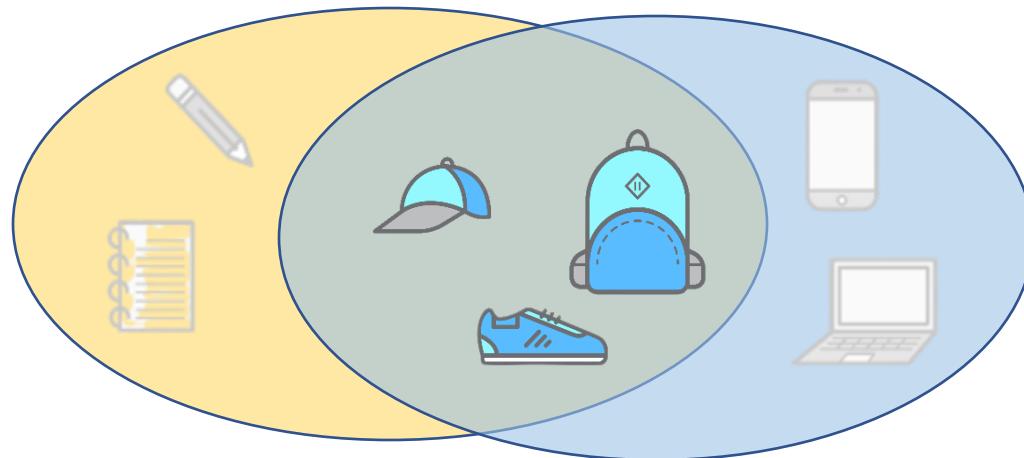
Commonalities and Diversities



- Commonalities
 - User Overlap



- Commonalities
 - Item Overlap



➤ Diversities

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

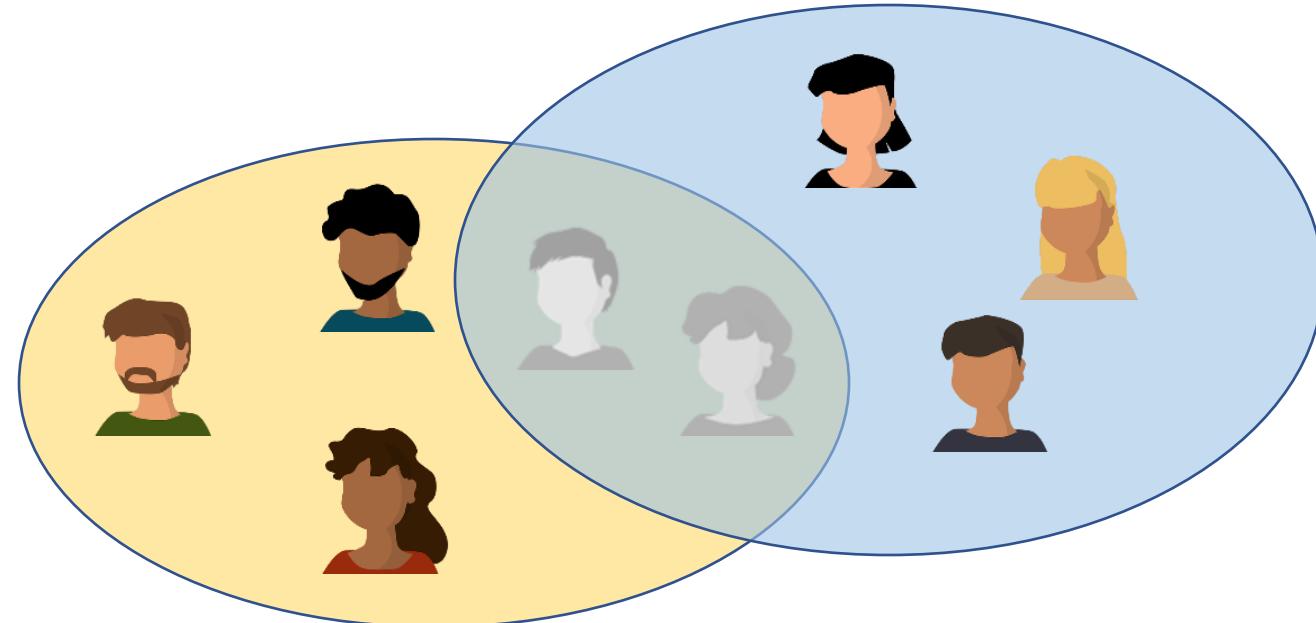
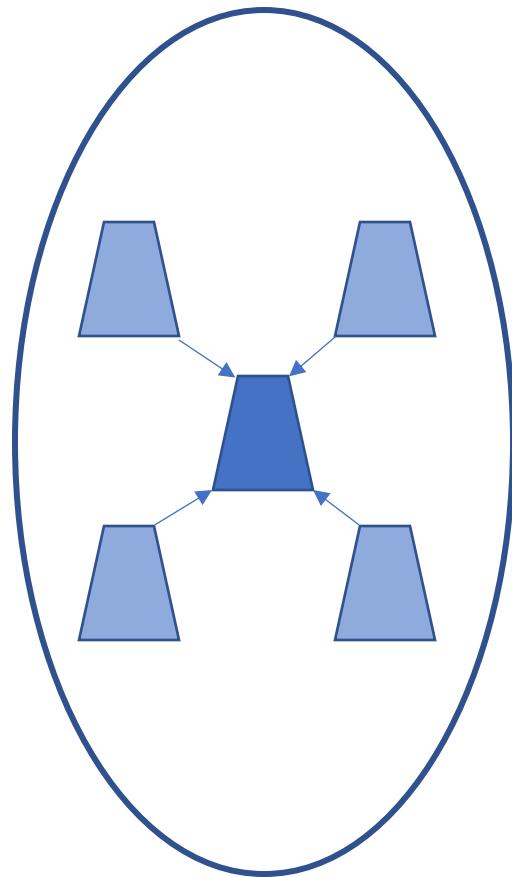
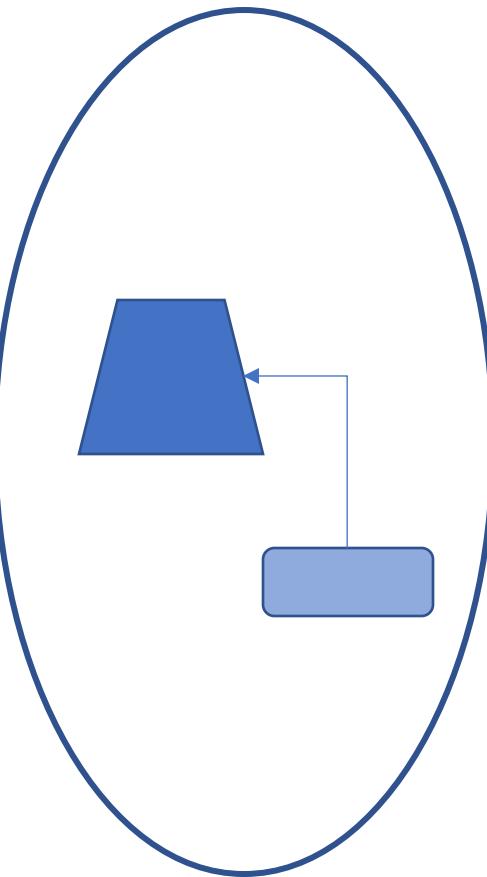


Table of Contents



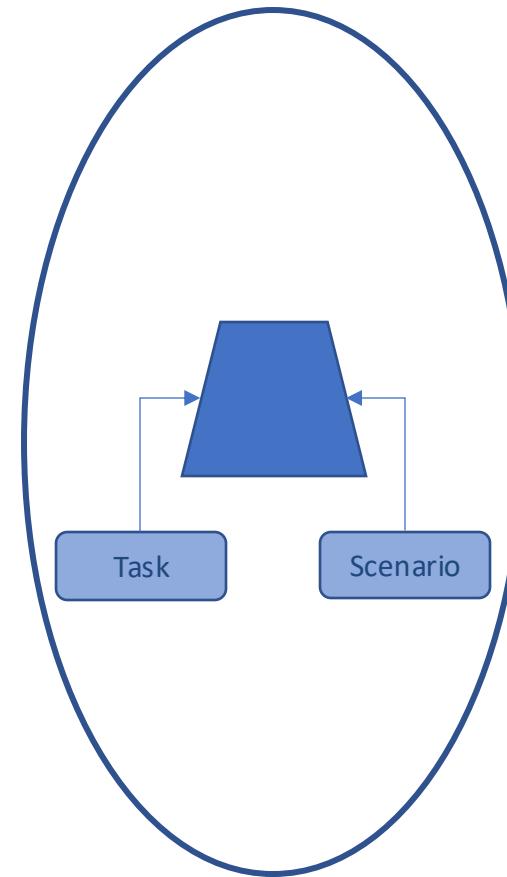
Shared-specific network paradigm

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



Dynamic weight

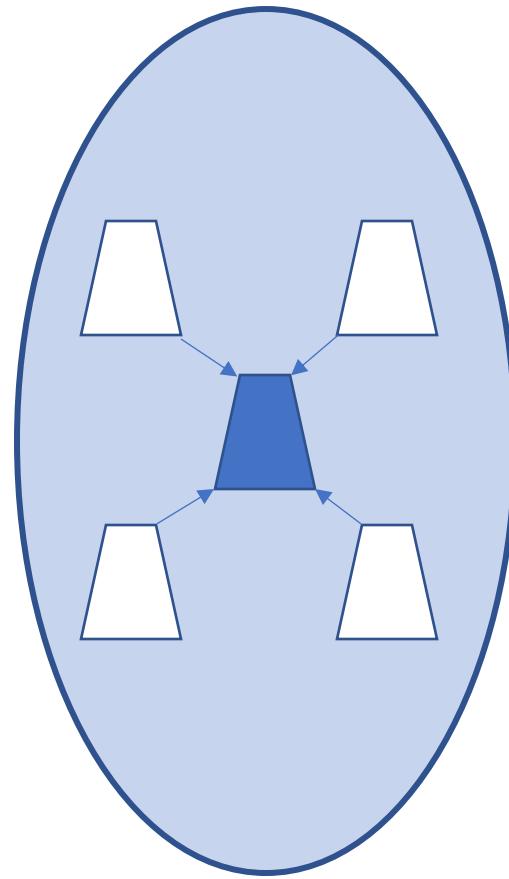
$$wL(E^{Merge}, \theta^{sh}, \theta^t, \theta^s)$$



Multi-Scenario & Multi-Task

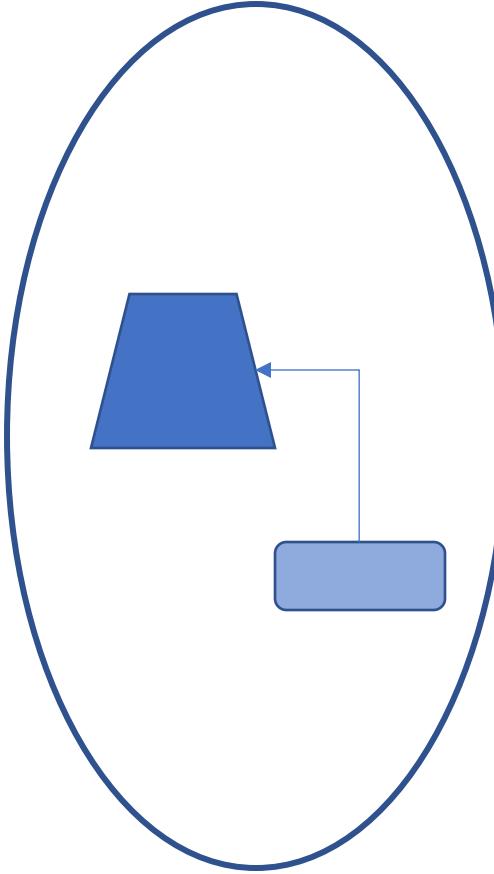
$$wL(E^{Merge}, \theta^{sh}, \theta^t, \theta^s)$$

Table of Contents



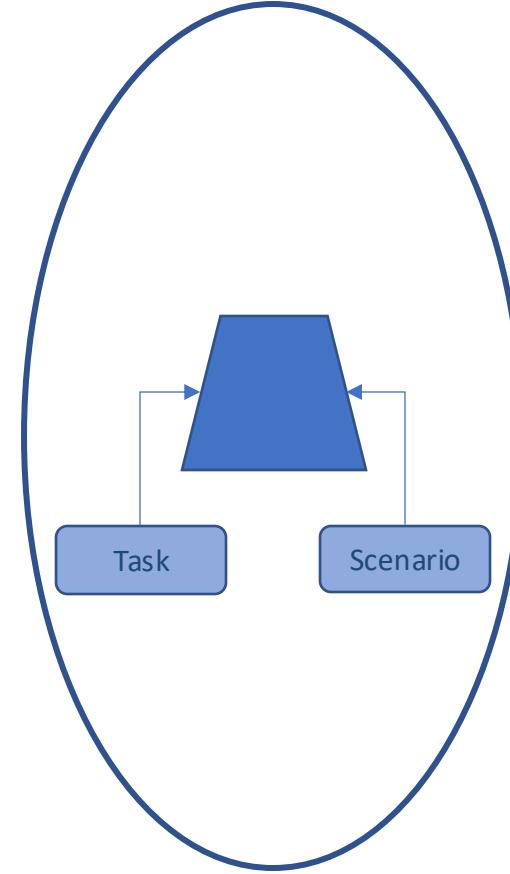
Shared-specific network paradigm

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



Dynamic weight

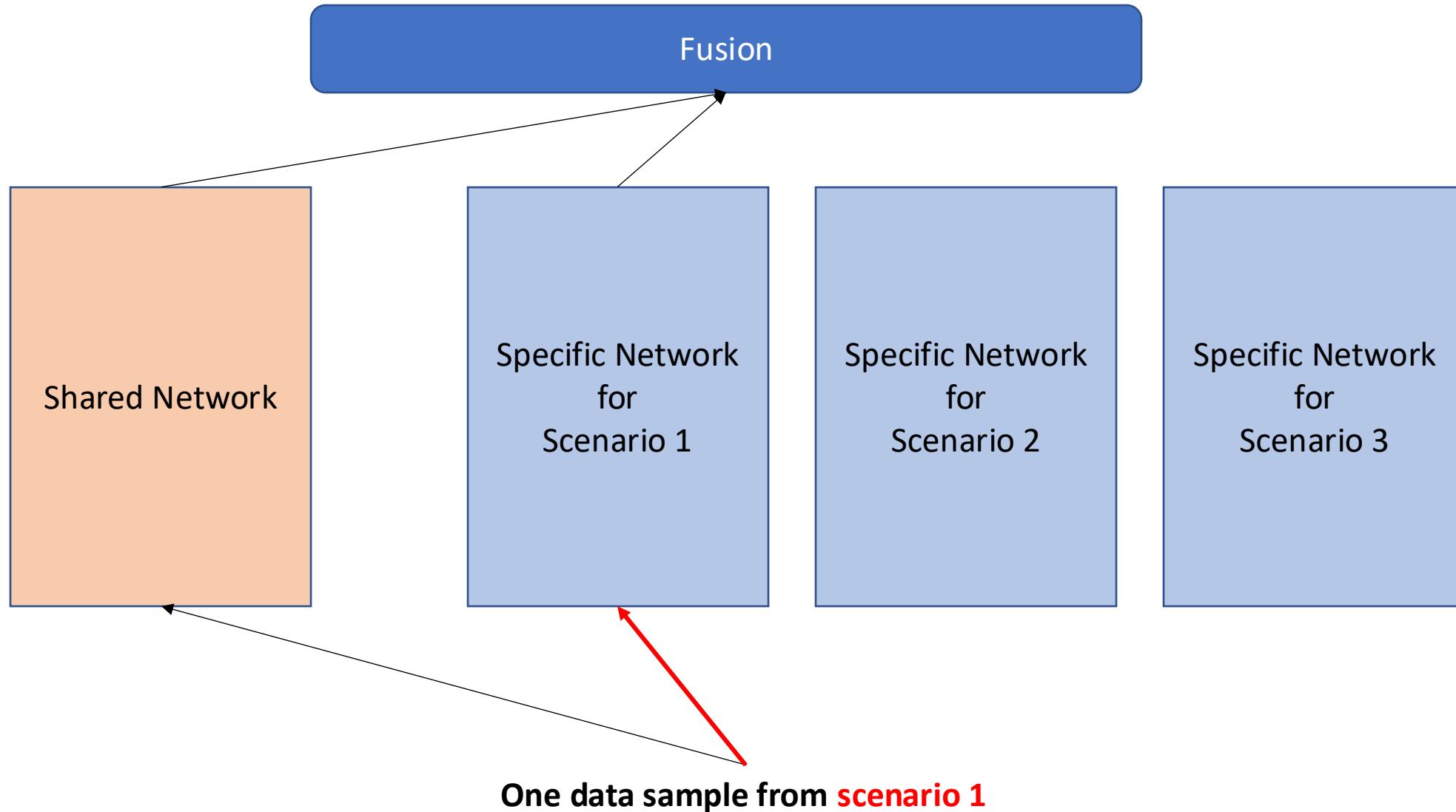
$$wL(E^{Merge}, \theta^{sh}, \theta^t, \theta^s)$$



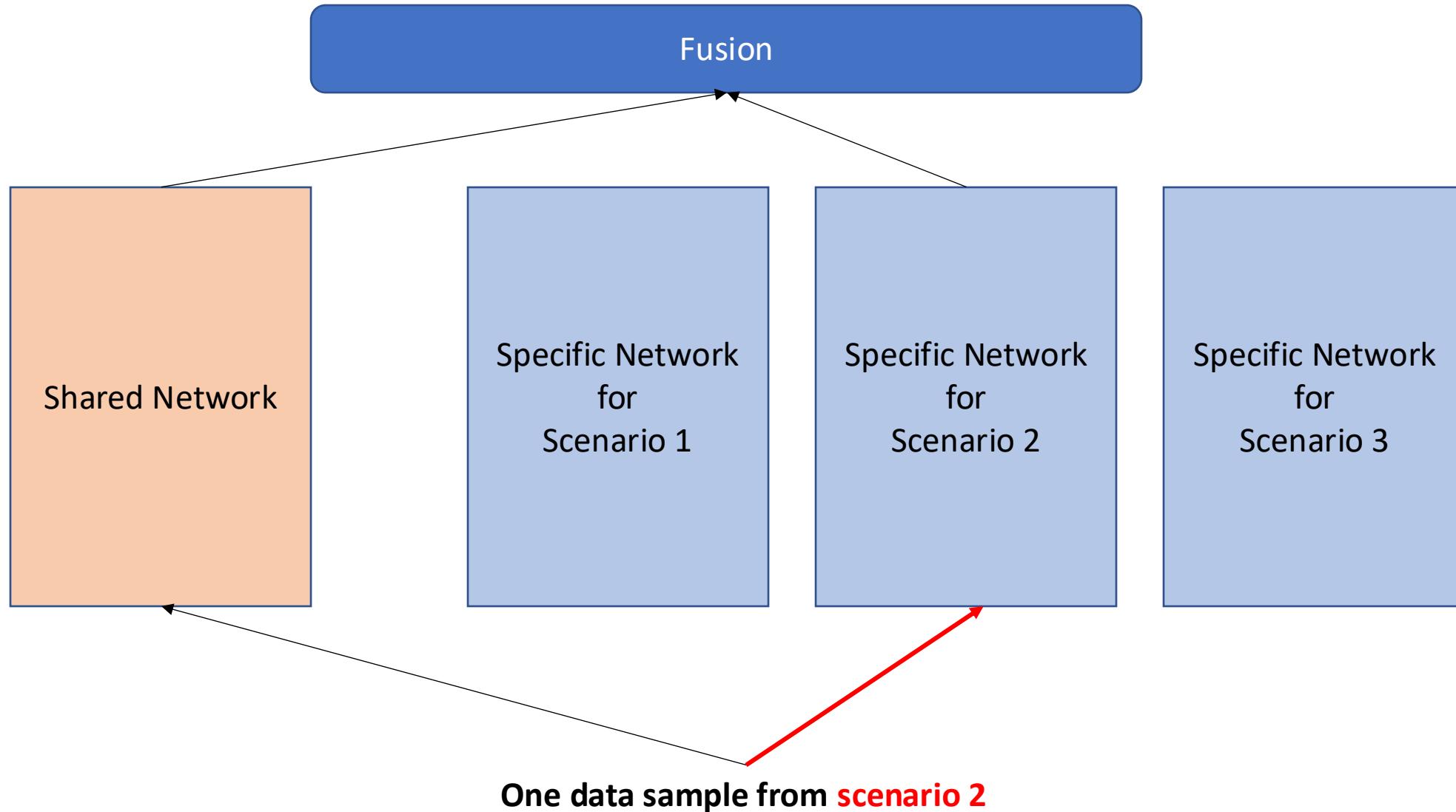
Multi-Scenario & Multi-Task

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

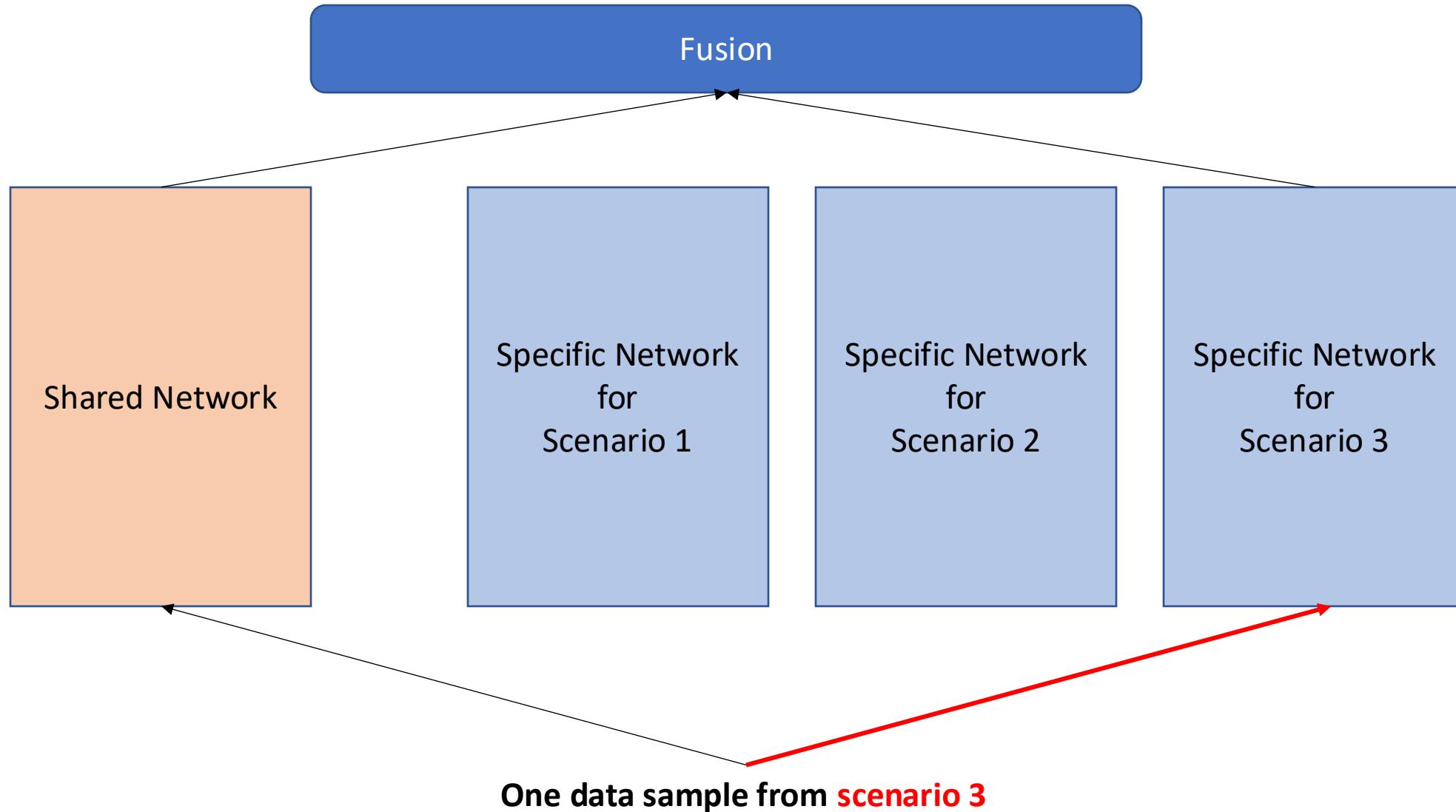
Shared-specific Network Paradigm



Shared-specific Network Paradigm



Shared-specific Network Paradigm



➤ 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

- 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 Topology

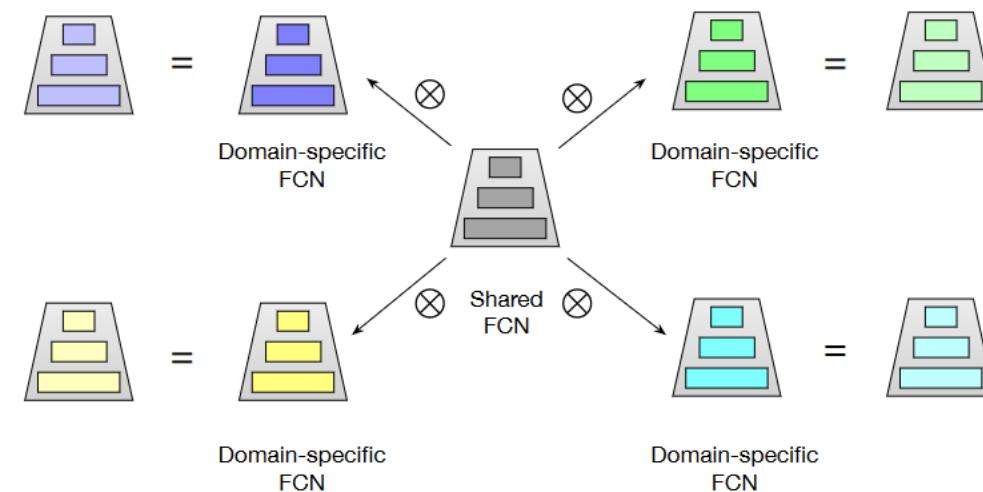
The final weight and bias for p-th domain is obtained by:

$$W_p^* = W_p \otimes W, b_p^* = b_p + b$$

The output for p-th domain is derived by:

$$out_p = \phi((W_p^*)^\top in_p + b_p^*)$$

\otimes element-wise product

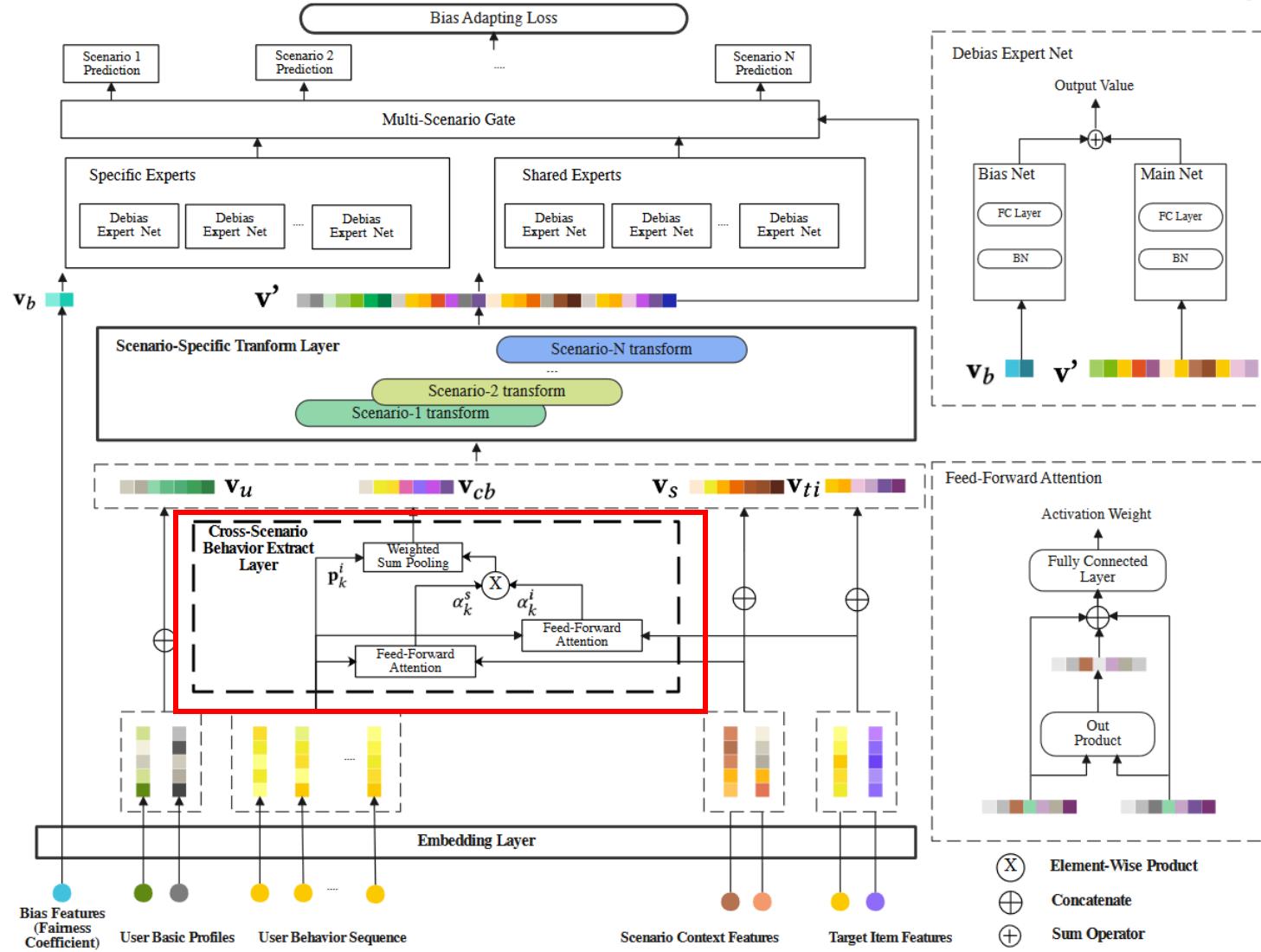


➤ Motivation

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

➤ Target

- Train a unified model to serve all scenarios



Cross-Scenario Behavior Extract Layer

How to aggregate the sequence?

$p(B^i)$ is item behavior sequence

$$p(B^i) = \{p_1^i, p_2^i, \dots, p_{|p(B^i)|}^i\}$$

$$p_k^i = [e_{itemId} || e_{destination} || e_{category} || \dots]$$

$p(B^s)$ is scenario context sequence

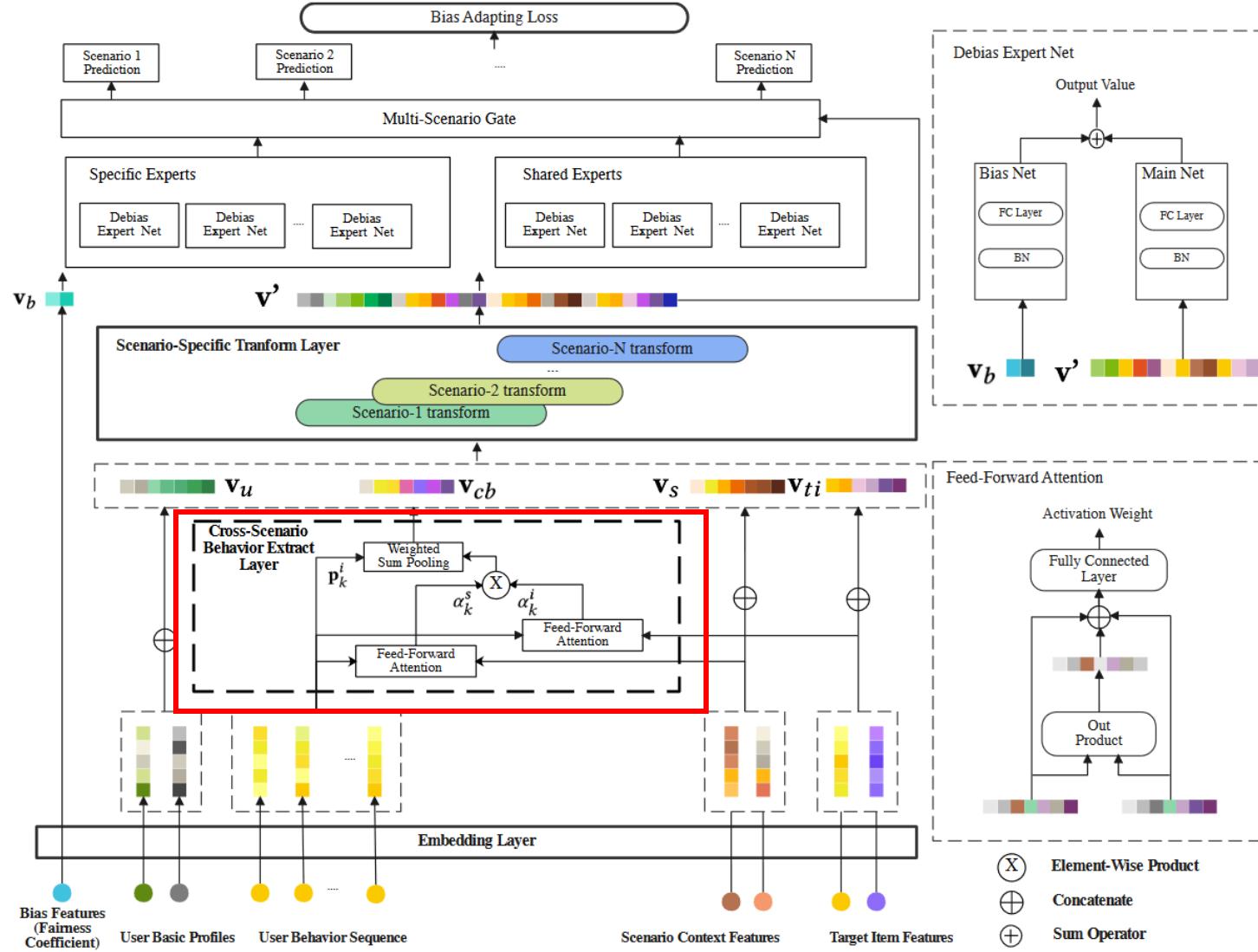
$$p_k^s = [e_{scenarioId} || e_{scenarioType} || e_{behaviorTime} || \dots]$$

$$p(B^s) = \{p_1^s, p_2^s, \dots, p_{|p(B^s)|}^s\}$$

$$\alpha_k^i = \frac{\exp(\psi(p_k^i, p_t^i))}{\sum_{l=1}^{|p(B^i)|} \exp(\psi(p_l^i, p_t^i))},$$

$$\alpha_k^s = \frac{\exp(\psi(p_k^s, p_t^s))}{\sum_{l=1}^{|p(B^s)|} \exp(\psi(p_l^s, p_t^s))},$$

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



Cross-Scenario Behavior Extract Layer

How to aggregate the sequence?

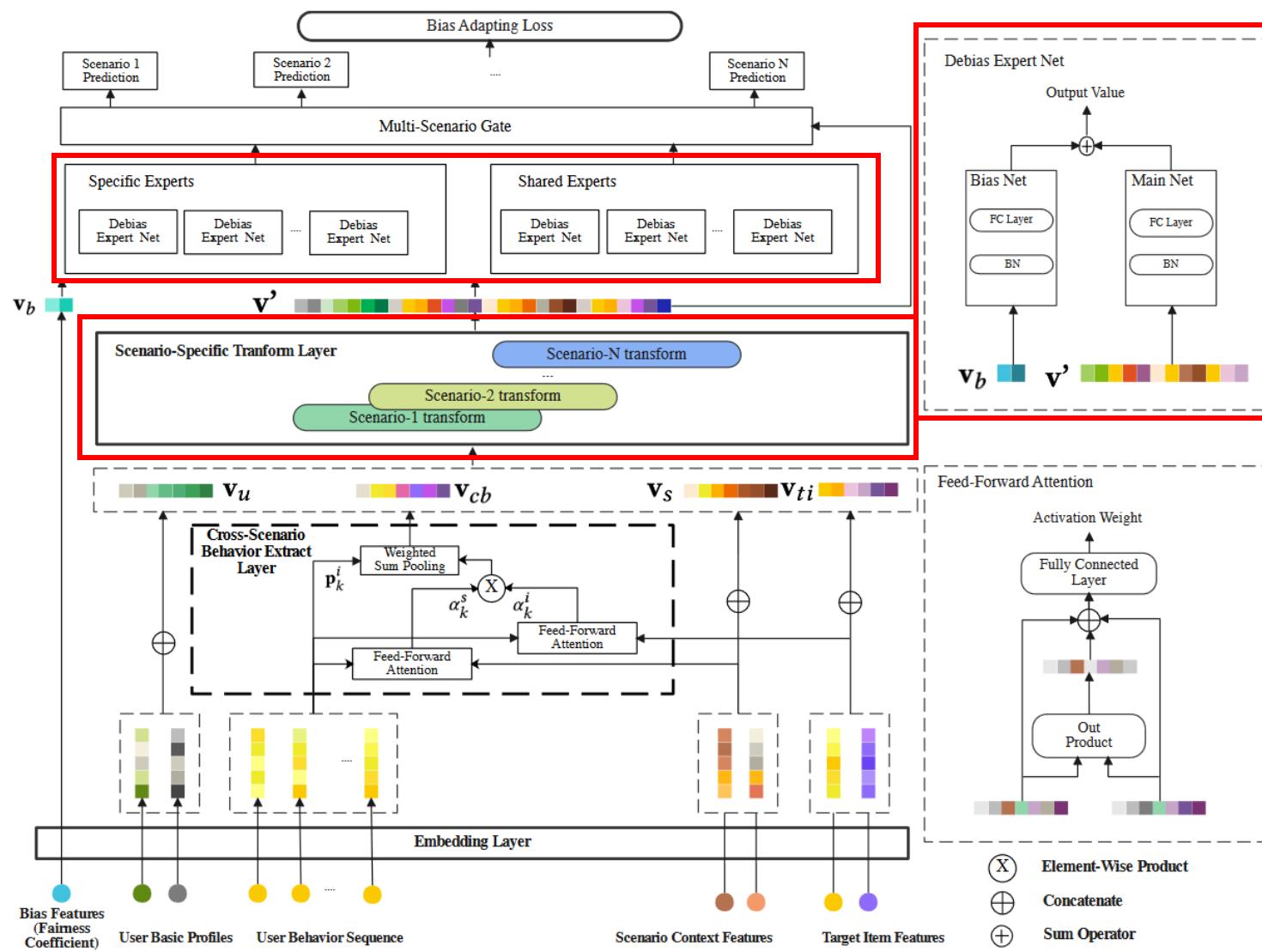
$$\alpha_k^i = \frac{\exp(\psi(p_k^i, p_t^i))}{\sum_{l=1}^{|p(B^i)|} \exp(\psi(p_l^i, p_t^i))},$$

$$\alpha_k^s = \frac{\exp(\psi(p_k^s, p_t^s))}{\sum_{l=1}^{|p(B^s)|} \exp(\psi(p_l^s, p_t^s))},$$

$$p_k^i = [e_{itemId} || e_{destination} || e_{category} || \dots]$$

$$v_{cb} = \sum_{k=1}^t \alpha_k^i * \alpha_k^s * p_k^i$$

- (X) Element-Wise Product
- (⊕) Concatenate
- (+) Sum Operator



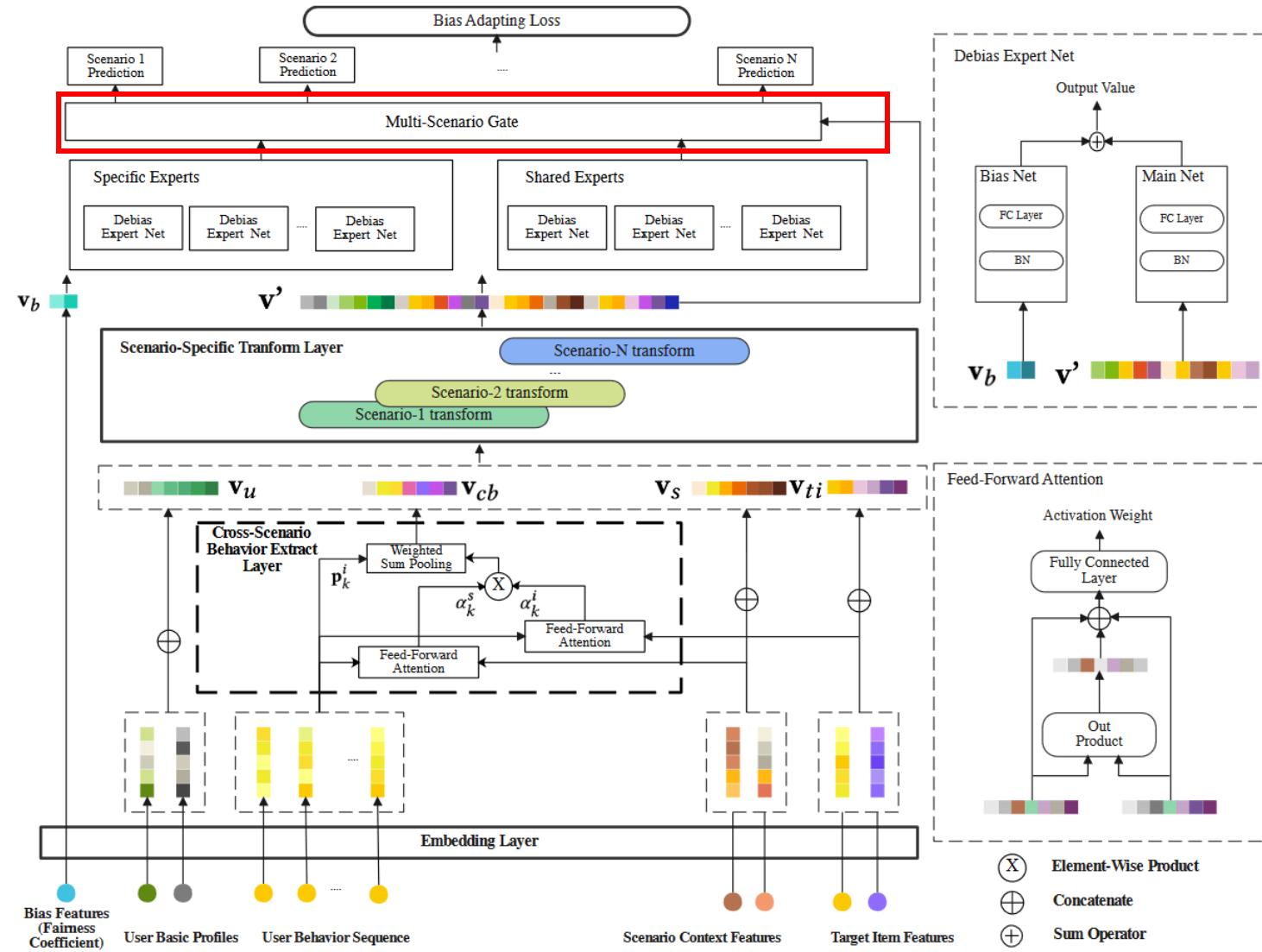
Scenario-Specific Transform Layer

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

Mixture of Debias Experts

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

- (X) Element-Wise Product
- (⊕) Concatenate
- (+) Sum Operator



Multi-Gate Network & Prediction

The output of experts:

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

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

- (X) Element-Wise Product
- (⊕) Concatenate
- (+) Sum Operator

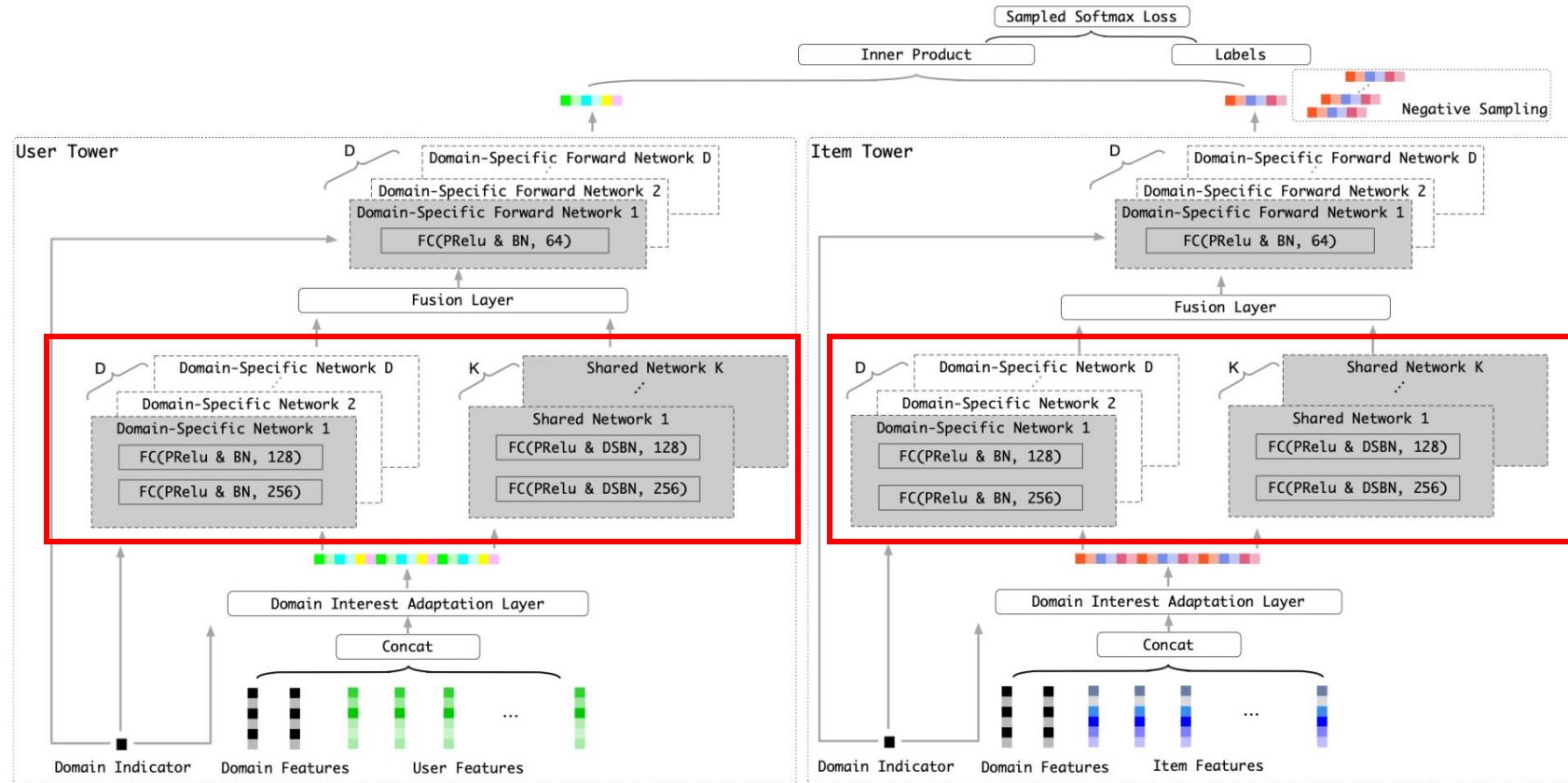
➤ Motivation

- 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

Backbone Network



Shared Network & Domain-Specific Network

$$az_k = \frac{W_{shared}^k(f_{domain}) + b_{shared}^k}{\sum_{n=1}^K (W_{shared}^n(f_{domain}) + b_{shared}^n)}$$

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

$$E_{spec}^{(d)} = MLP_{spec}^{(d)}(\mathbf{F}^{(d)})$$

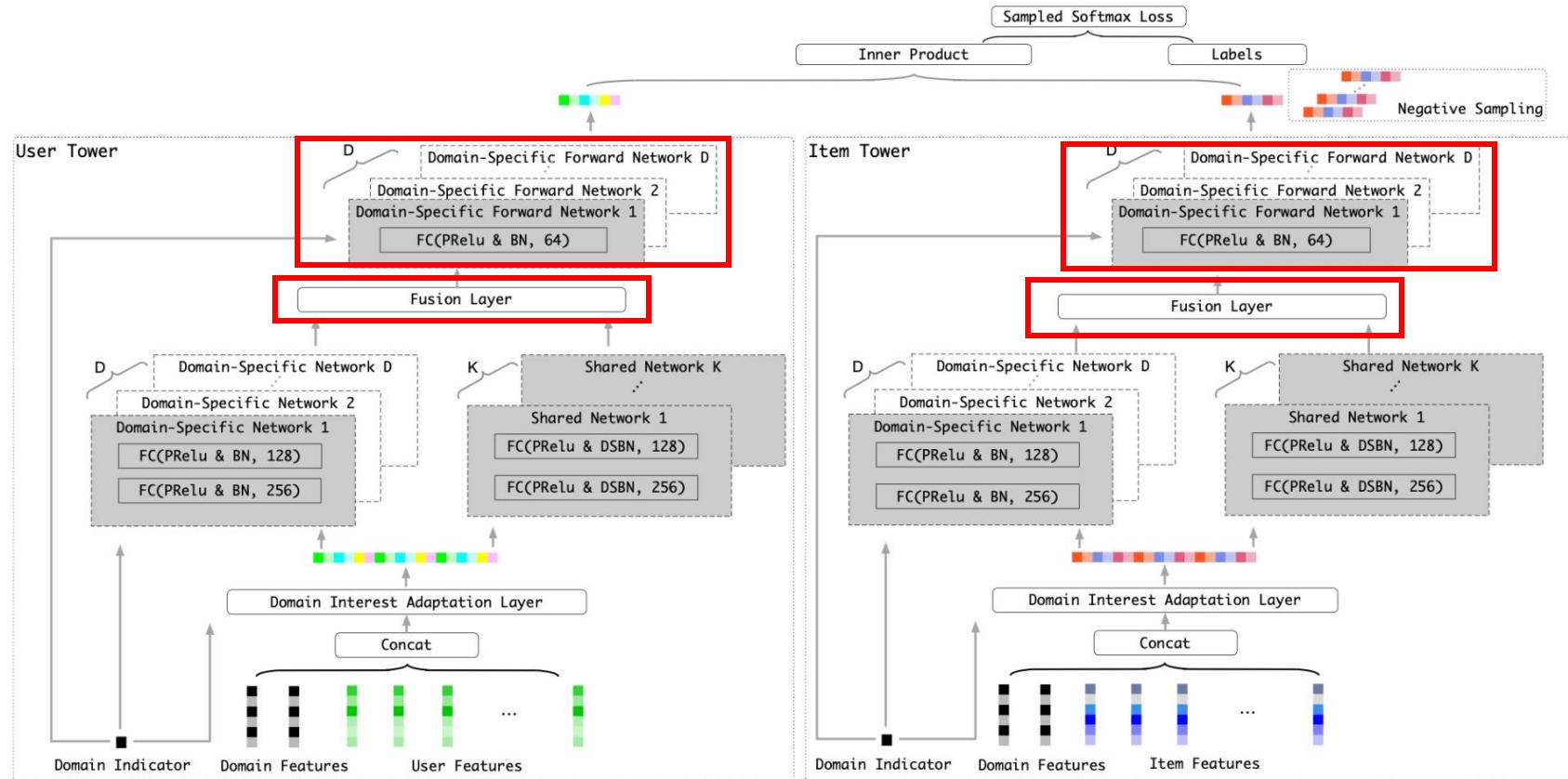
f_{domain} Domain indicator embedding

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

K hyperparameter,
number of Shared Network

D domains, D Domain-Specific Network

Backbone Network



Fusion Layer

$$\beta_1^{(d)} = \sigma(W_{fusion_spec}^{(d)}(f_{domain}))$$

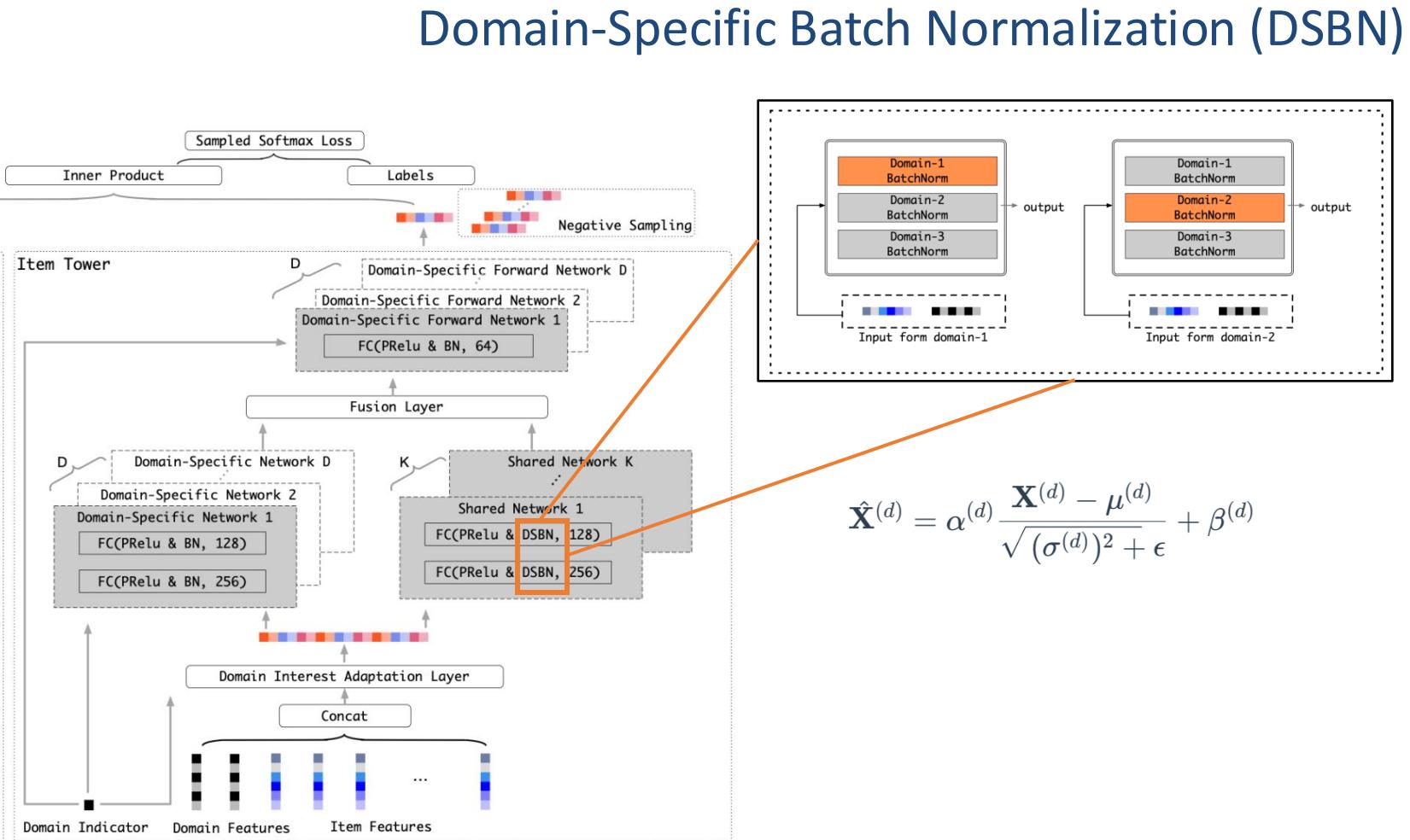
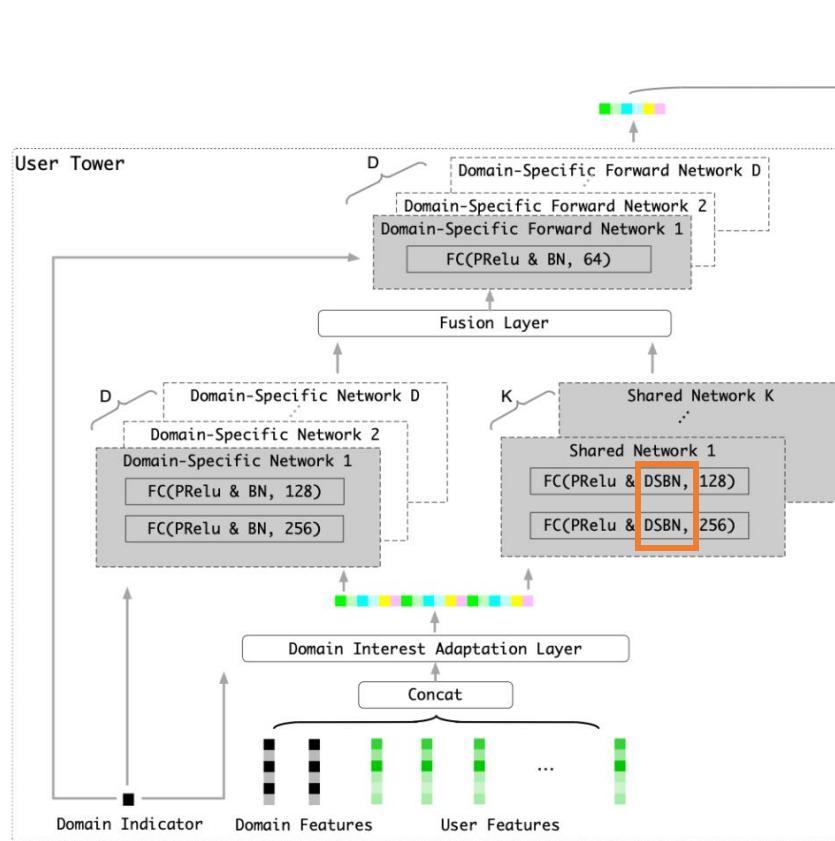
$$\beta_2^{(d)} = \sigma(W_{fusion_shared}^{(d)}(f_{domain}))$$

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

Domain-Specific Forward Network

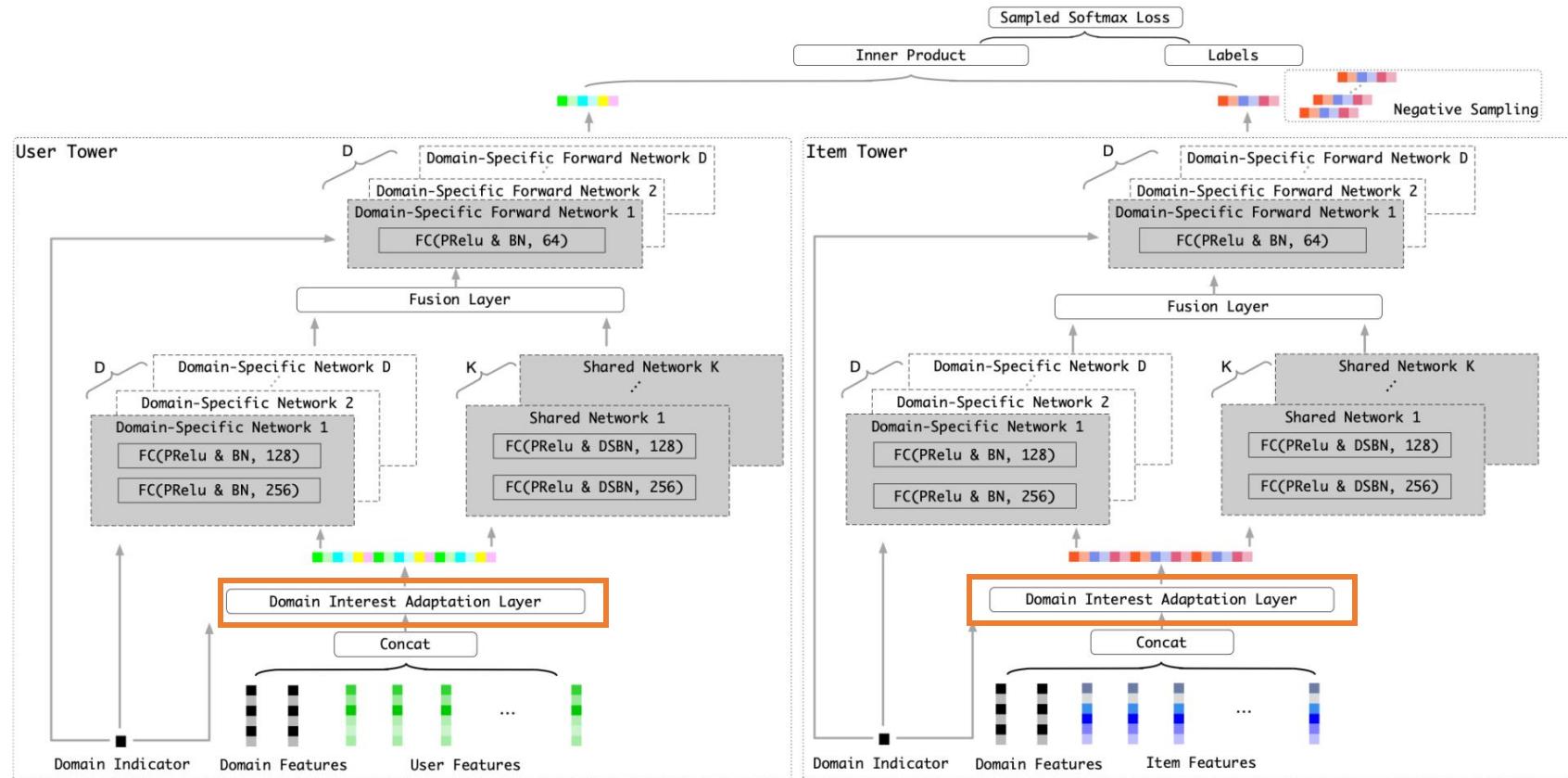
$$E = FC_{forward}^{(d)}(E_{fusion}^{(d)})$$

Domain Adaptation



Domain Adaptation

Domain Interest Adaptation Layer



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

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

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

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

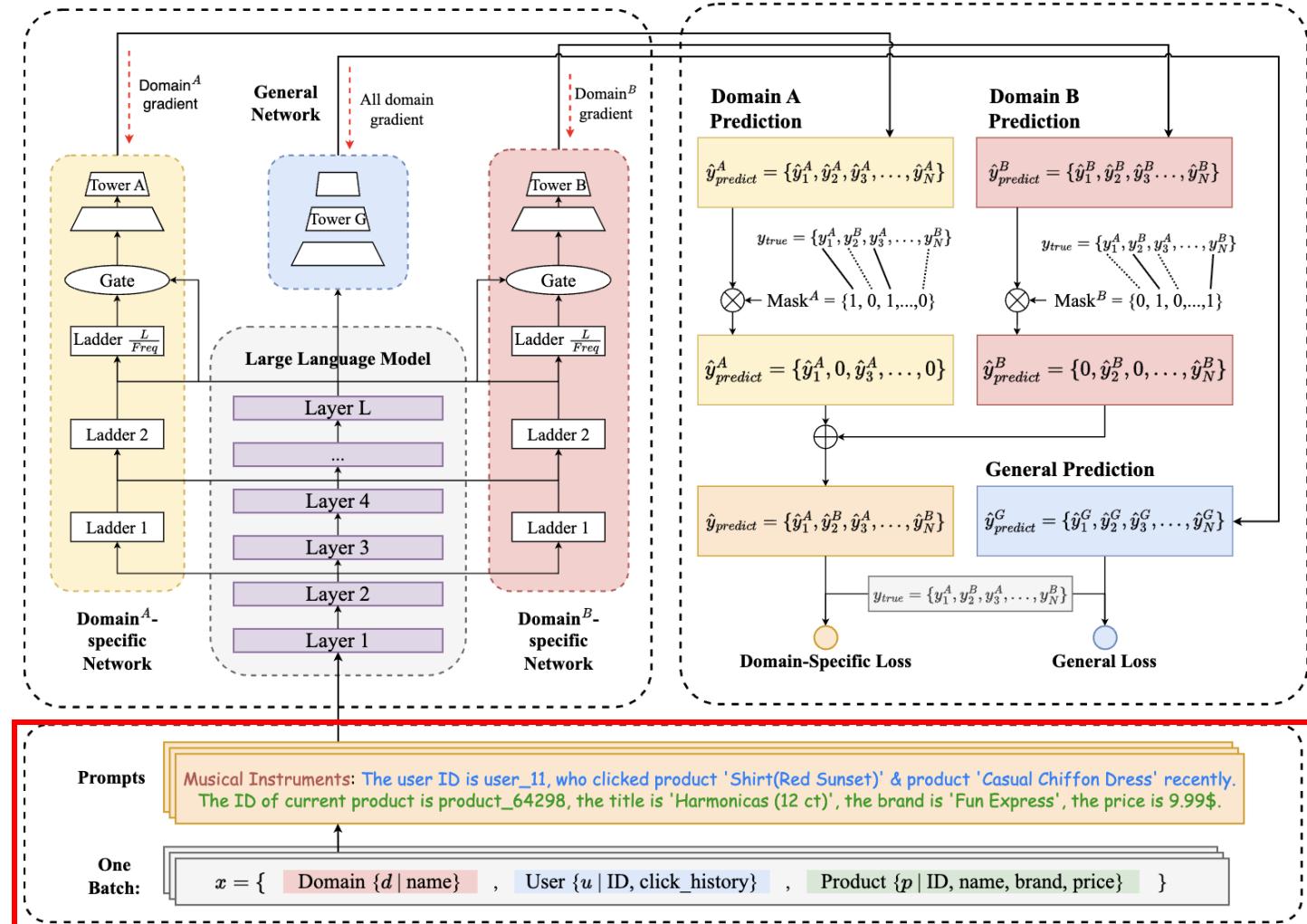
➤ Motivation

- Due to varying data sparsity in different domains, models can easily be dominated by specific domains, leading to “seesaw phenomenon”
- Existing methods are difficult to handle newly added domain

➤ Method

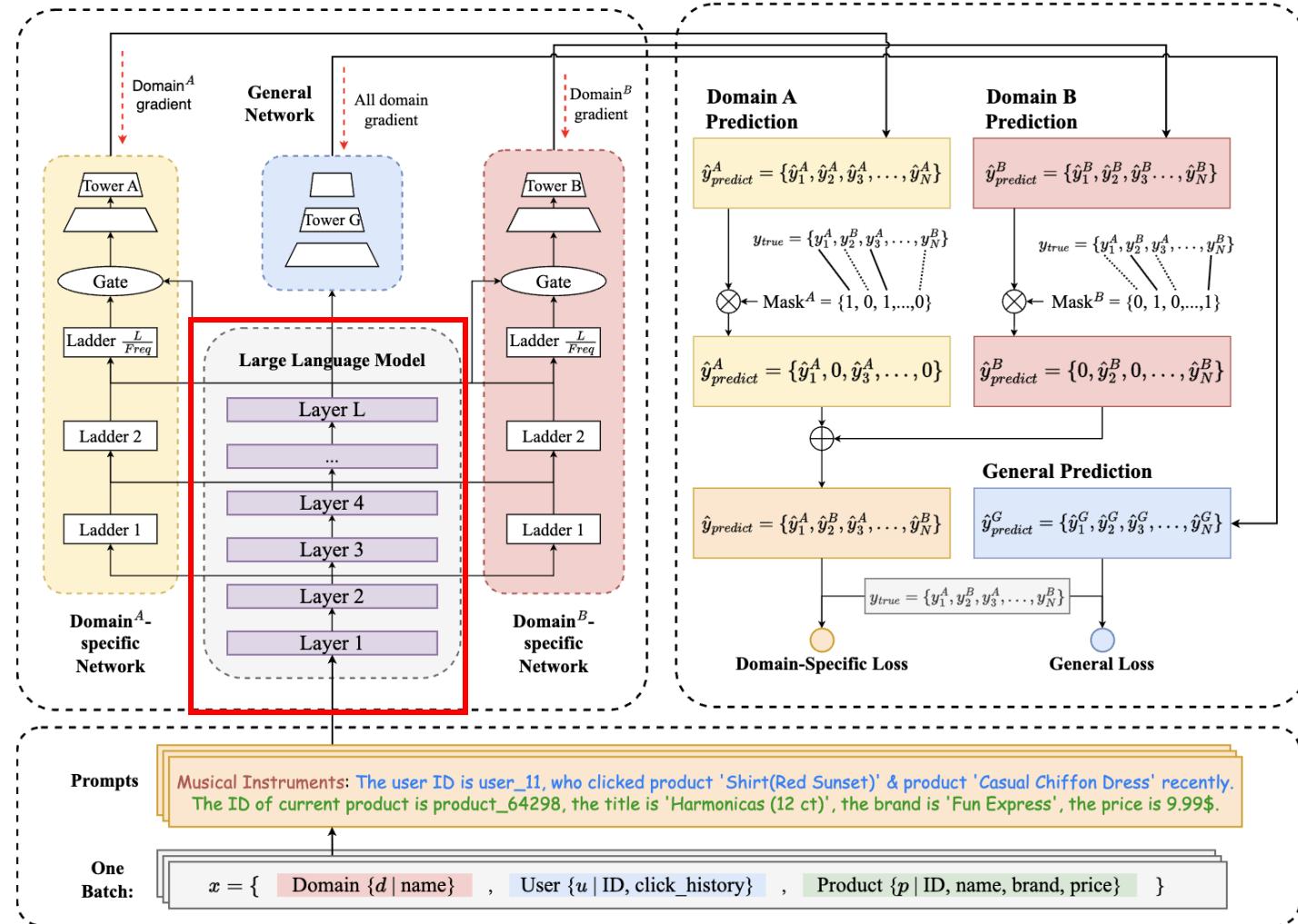
- Leveraging LLM to extract layer-wise representations to capture domain commonalities in order to migrate “seesaw phenomenon”
- Incorporating a pluggable domain-specific network to capture domain characteristics, ensuring scalability to new domains

Uni-CTR Details



- **Prompt-based Semantic Modeling**
- Capture rich semantic information via text-based features
 - Input
 - Domain Context
 - User Information
 - Product Information

Uni-CTR Details



➤ Uni-CTR architecture

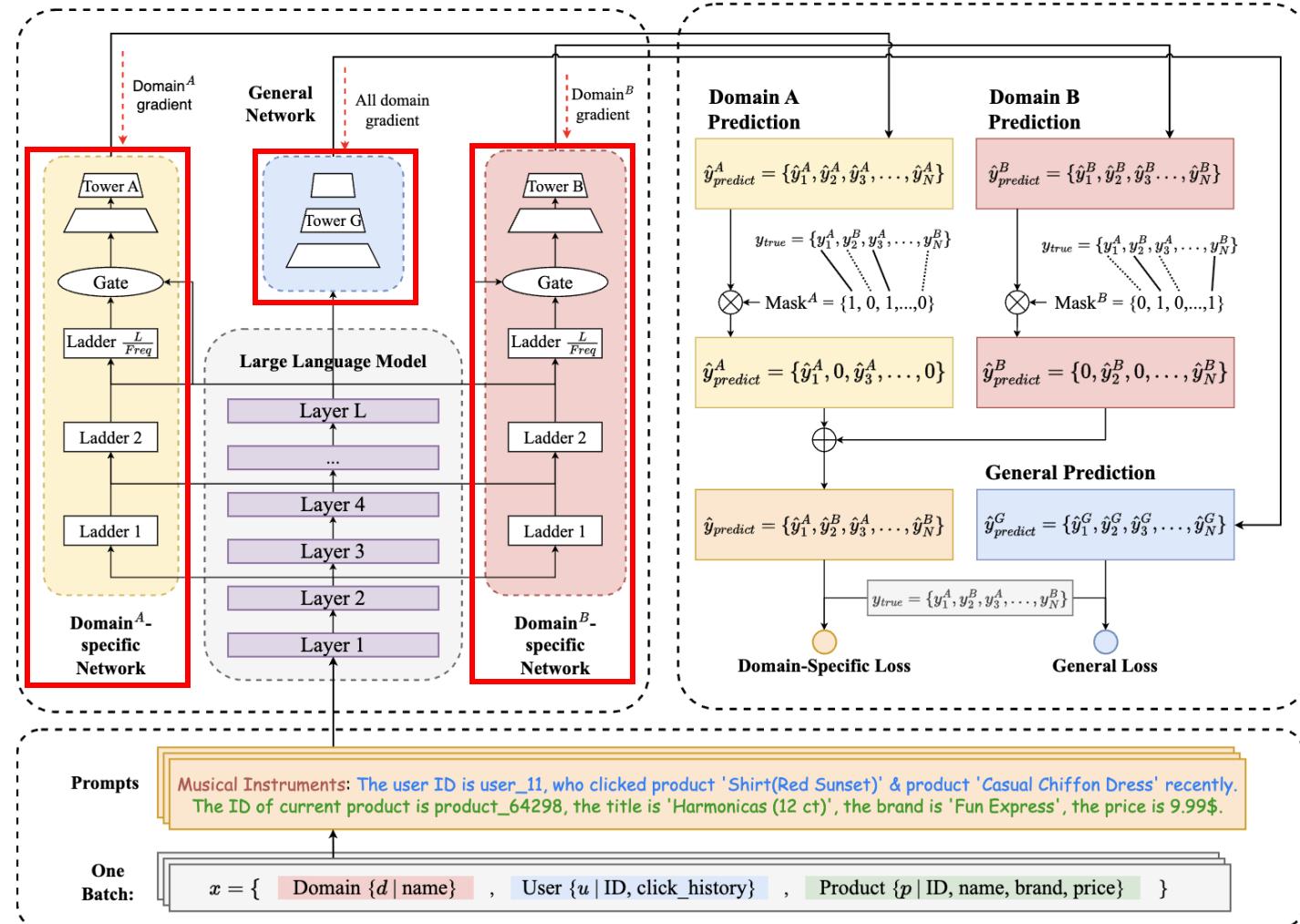
- LLM Backbone

$$\mathbf{x}_{tokens} = \text{Tokenizer}(x_{text}) = \{t_0, t_1, \dots, t_J\},$$

$$\mathbf{h}_0 = \mathbf{E}_{embed}(\mathbf{x}_{tokens}) = \{e_0, e_1, \dots, e_J\}.$$

$$\mathbf{h}_l = \text{Transformer}_l(\mathbf{h}_{l-1}), l \in \{1, 2, \dots, L\},$$

Uni-CTR Details



➤ Uni-CTR architecture

- LLM Backbone

$$\mathbf{x}_{\text{tokens}} = \text{Tokenizer}(x_{\text{text}}) = \{t_0, t_1, \dots, t_J\},$$

$$\mathbf{h}_0 = \mathbf{E}_{\text{embed}}(\mathbf{x}_{\text{tokens}}) = \{\mathbf{e}_0, \mathbf{e}_1, \dots, \mathbf{e}_J\}.$$

$$\mathbf{h}_l = \text{Transformer}_l(\mathbf{h}_{l-1}), l \in \{1, 2, \dots, L\},$$

- Domain-Specific Network

- Ladder Network
- Gate Net
- Tower Net

- General Network

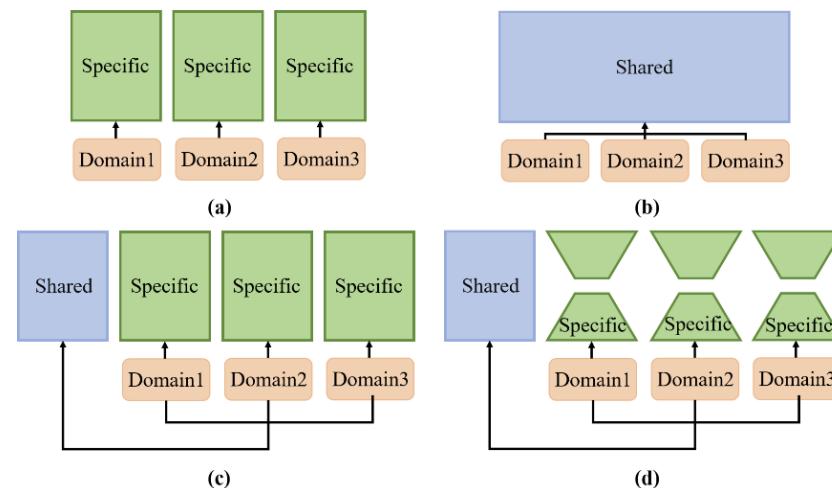
$$\hat{y}^G = \text{MLP}(\mathbf{h}_L; \mathbf{W}_\sigma^G, \mathbf{b}_\sigma^G).$$

➤ Motivation

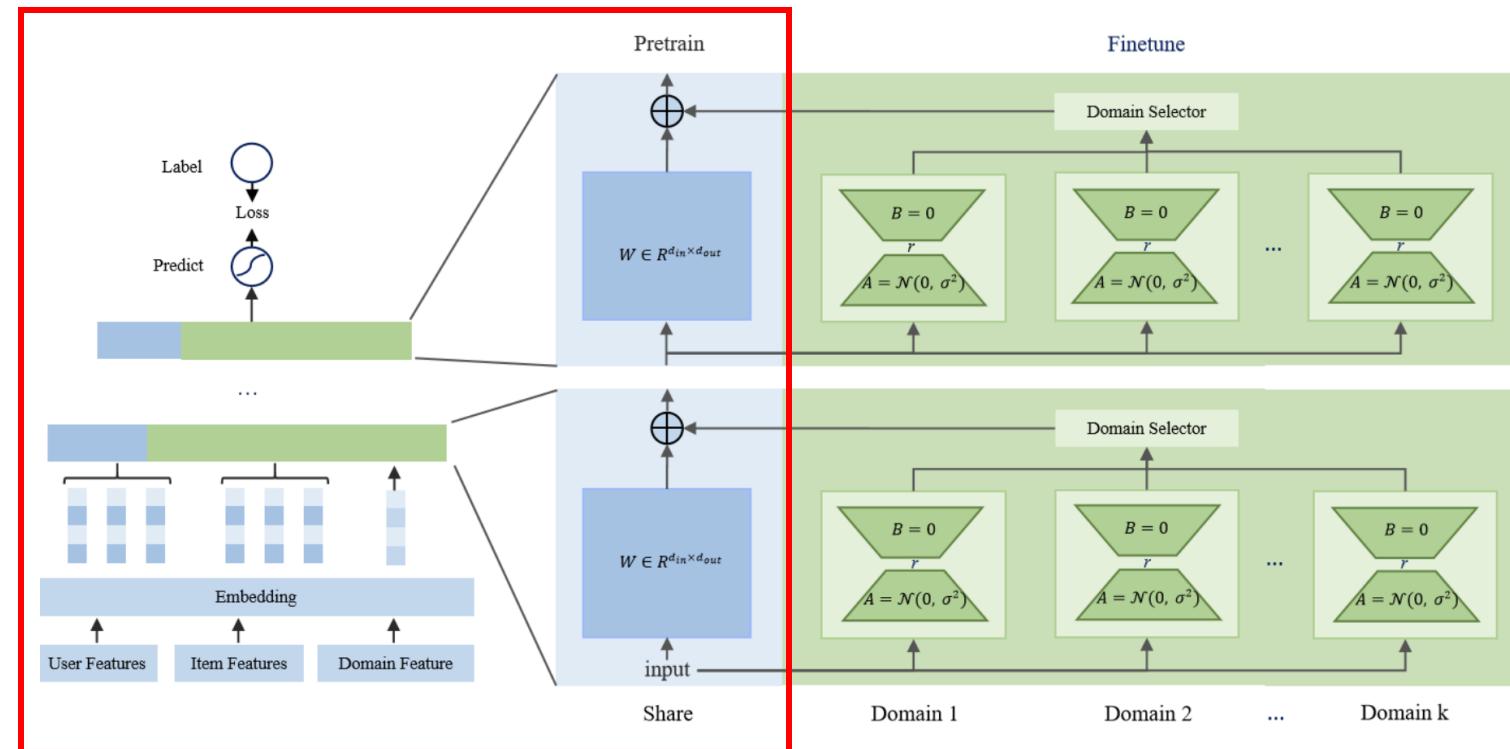
- Suffering from data sparsity and ignoring domain relations
- Failing to capture domain diversity
- Suffering from a sharp increase in model parameters

➤ Method

- Incorporating Low-Rank Adaptor (LoRA) for multi-domain fine-tuning

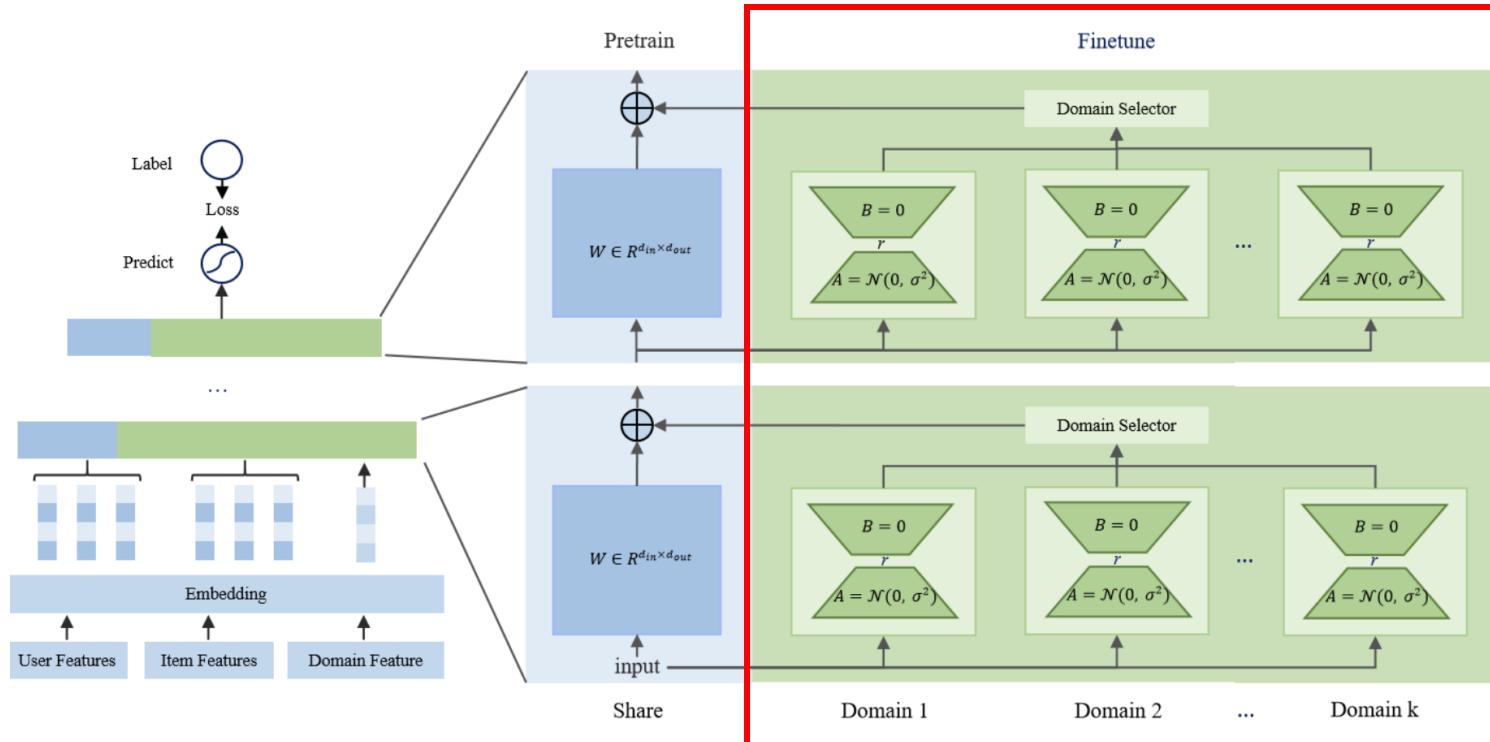


M-LoRA Details



➤ Pre-training

- Large-scale pre-training dataset
- Shared Network and Embedding layer



➤ Pre-training

- Large-scale pre-training dataset
- Shared Network and Embedding layer

➤ Fine-tuning

- LoRA module is integrated in each layer for each domain, including A and B

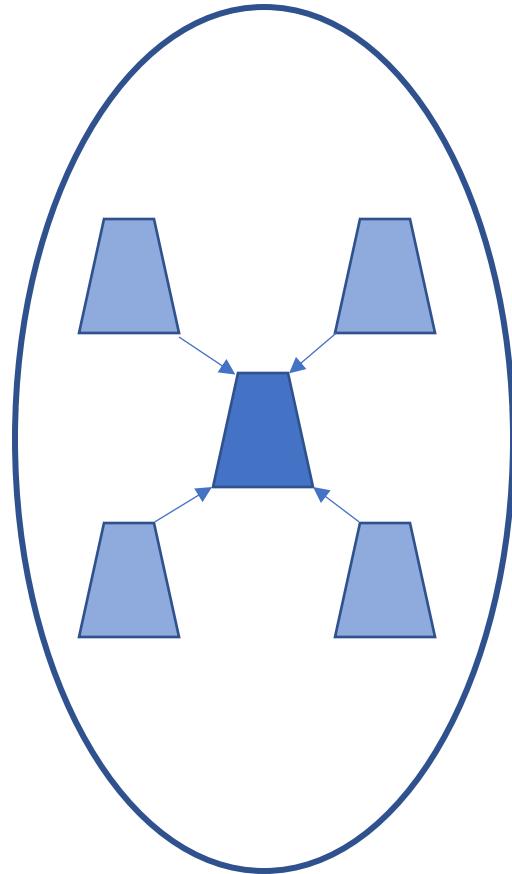
$$\mathbf{h}_t = \mathbf{Wx} + \Delta\mathbf{W}_t = \mathbf{Wx} + \mathbf{B}_t \mathbf{A}_t \mathbf{x},$$

where $\mathbf{B} \in R^{d_{out} \times r}$ and $\mathbf{A} \in R^{r \times d_{in}}$

- r is not fixed for a more flexible representation

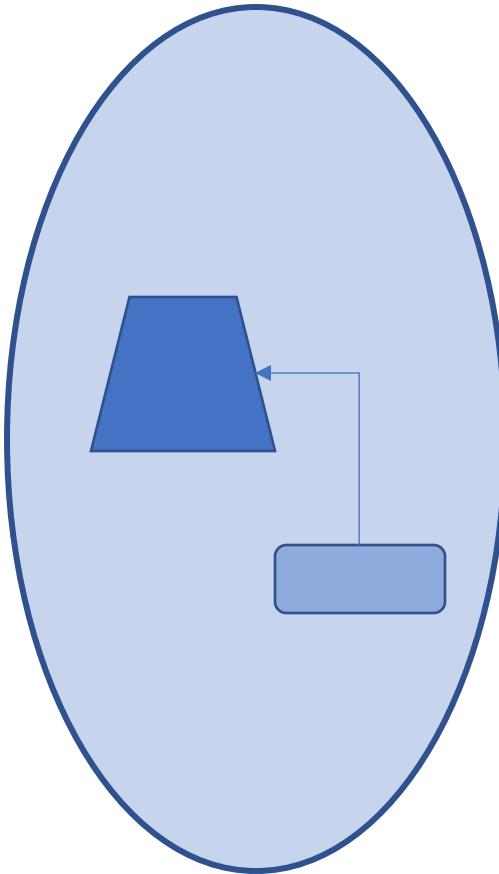
$$r = \max\left(\frac{d_{out}}{\alpha}, 1\right).$$

Table of Contents



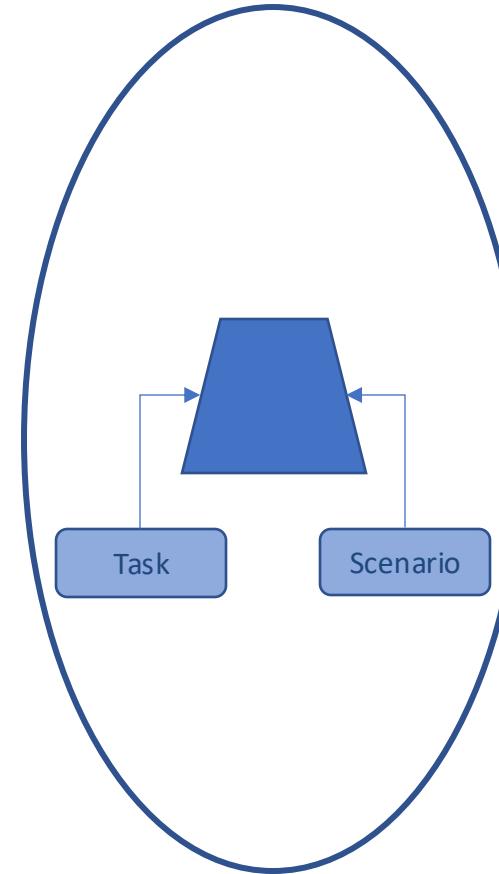
Shared-specific network paradigm

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



Dynamic weight

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



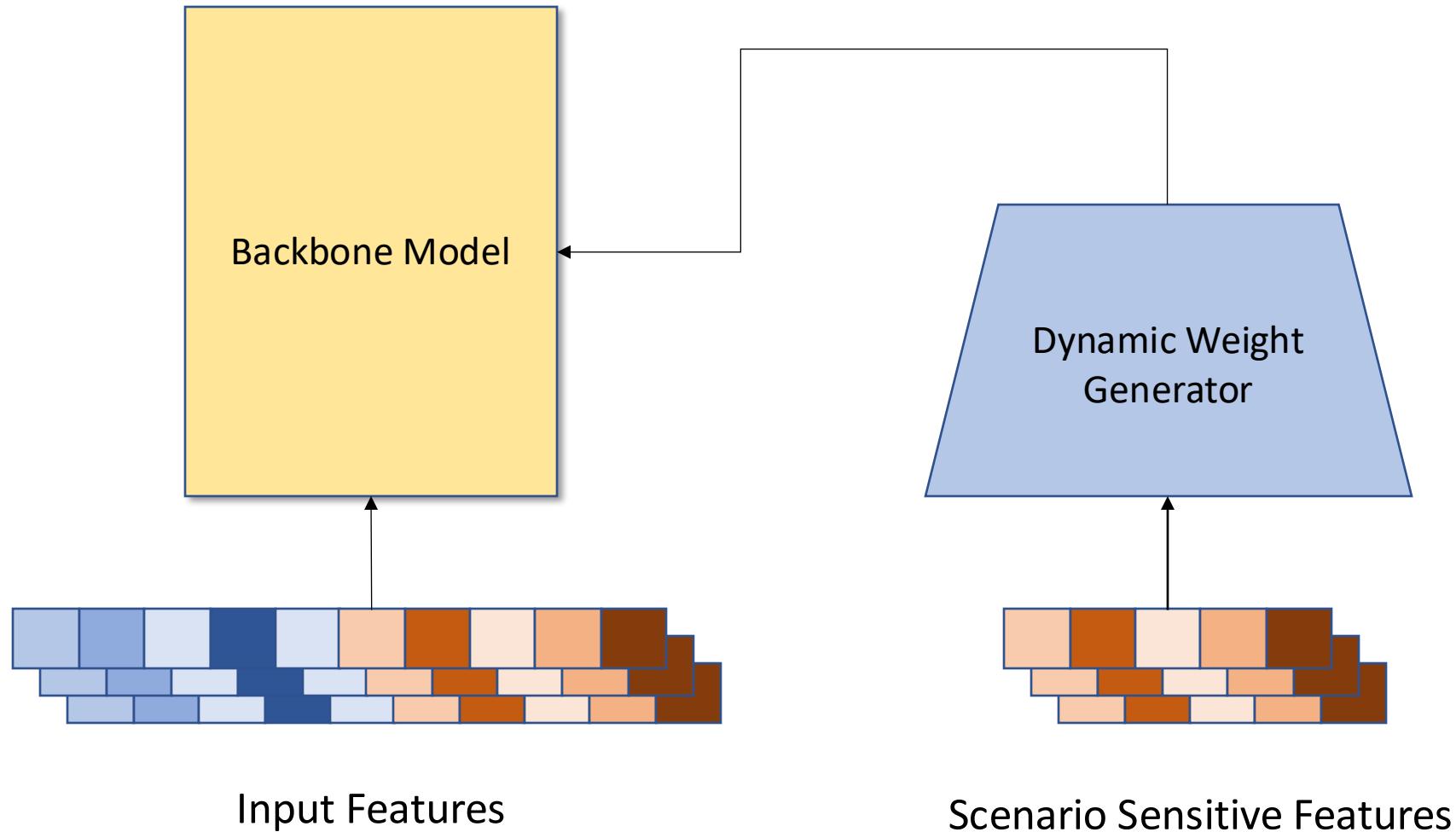
Multi-Scenario & Multi-Task

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

Dynamic Weight



➤ Why Dynamic?

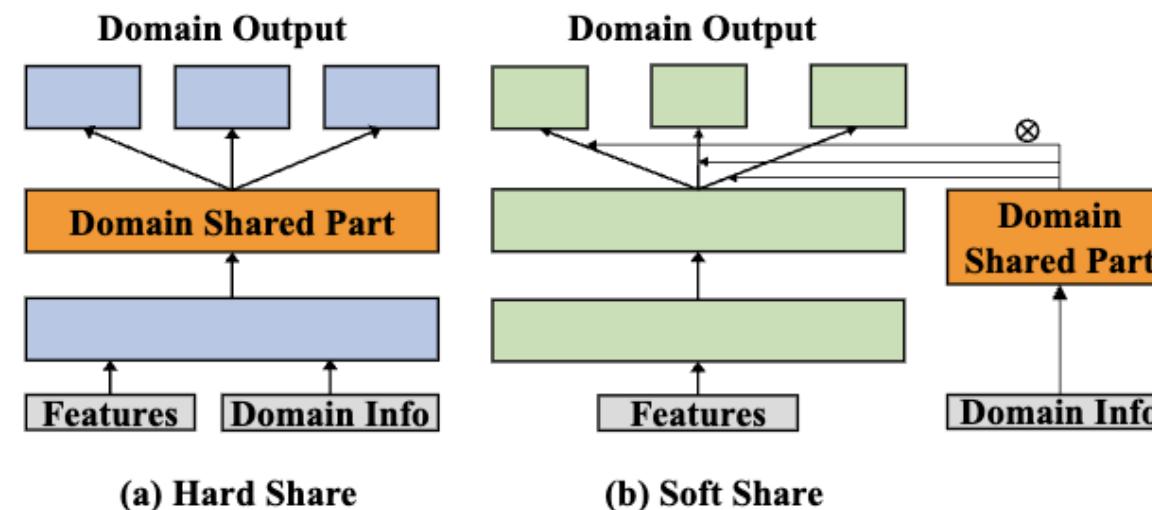


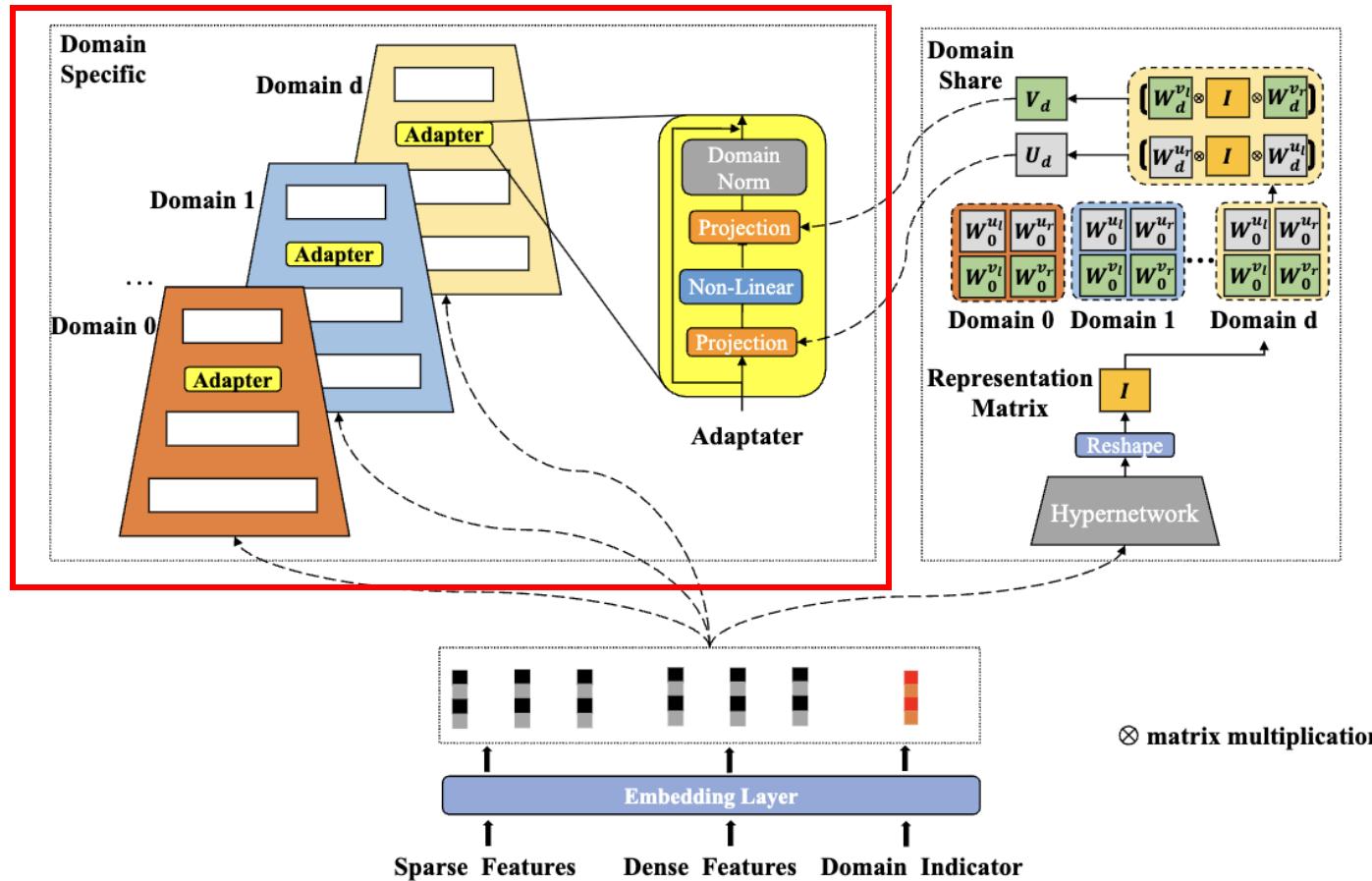
➤ Motivation

- Previous research relies on explicit sharing across different domains
- Static parameters constrain the representation of different domains

➤ Methods

- Adapter for multi-domain dynamic adaptation
- Hyper-net for dynamic generation parameters for adapters





- **Domain-specific adapter**
- $$A_d(x) = DN_d(V_d(\sigma(U_d(x)))) + x$$

$$DN_d = \gamma_d \odot \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta_d$$
- **Domain Shared Hyper-Network**
 - Parameters Generation

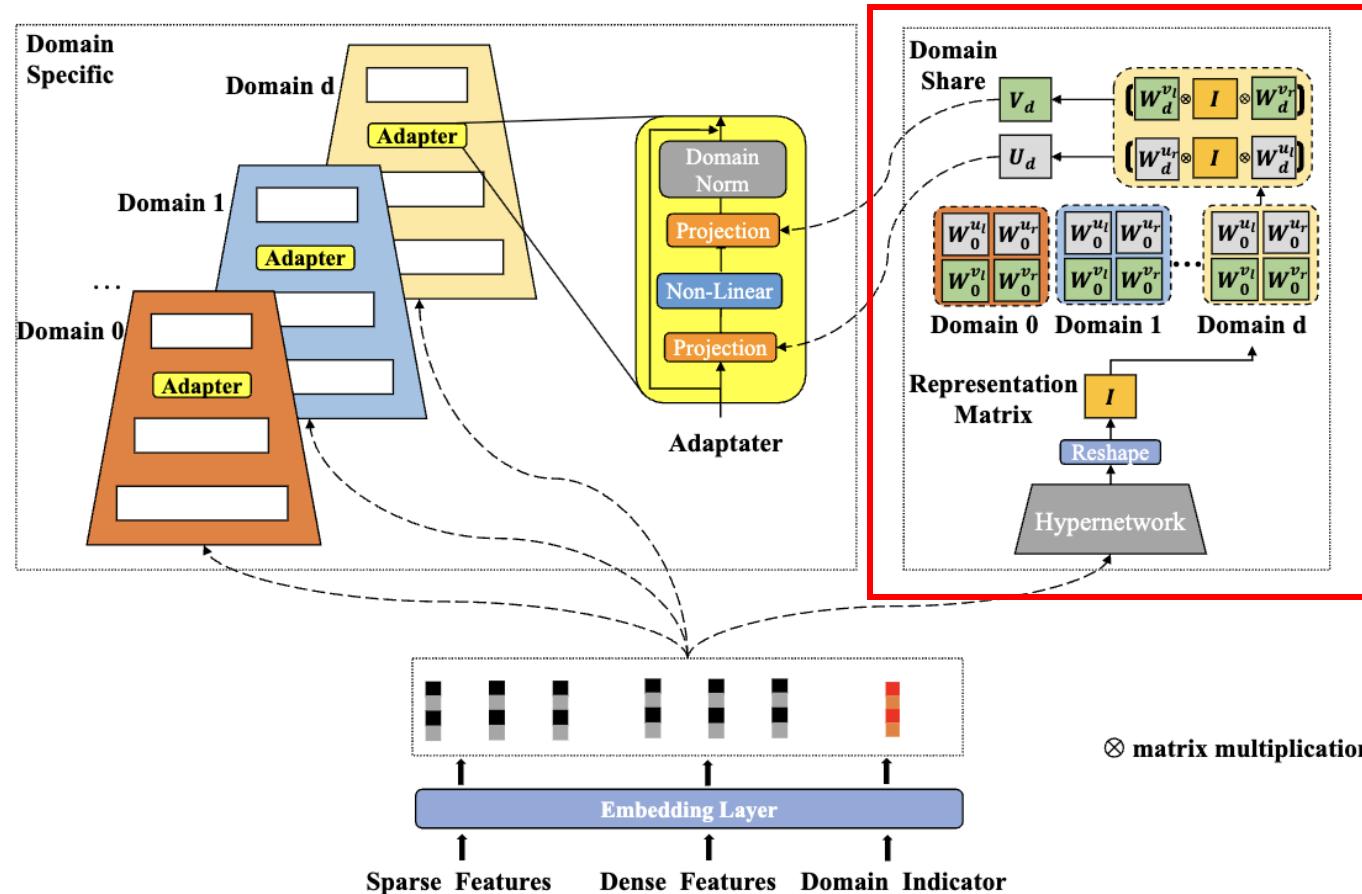
$$\mathbf{h}^i = \mathcal{H}(z^i)$$

$$I^i = \text{reshape}(\mathbf{h}^i)$$

- Low-Rank Decomposition

$$U_d^i = W_d^{u_l} \cdot I^i \cdot W_d^{u_r}$$

$$V_d^i = W_d^{v_l} \cdot I^i \cdot W_d^{v_r}$$



➤ Domain-specific adapter

$$A_d(x) = DN_d(V_d(\sigma(U_d(x)))) + x$$

$$DN_d = \gamma_d \odot \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta_d$$

➤ Domain Shared Hyper-Network

- Parameters Generation

$$\mathbf{h}^i = \mathcal{H}(z^i)$$

$$I^i = \text{reshape}(\mathbf{h}^i)$$

- Low-Rank Decomposition

$$U_d^i = W_d^{u_l} \cdot I^i \cdot W_d^{u_r}$$

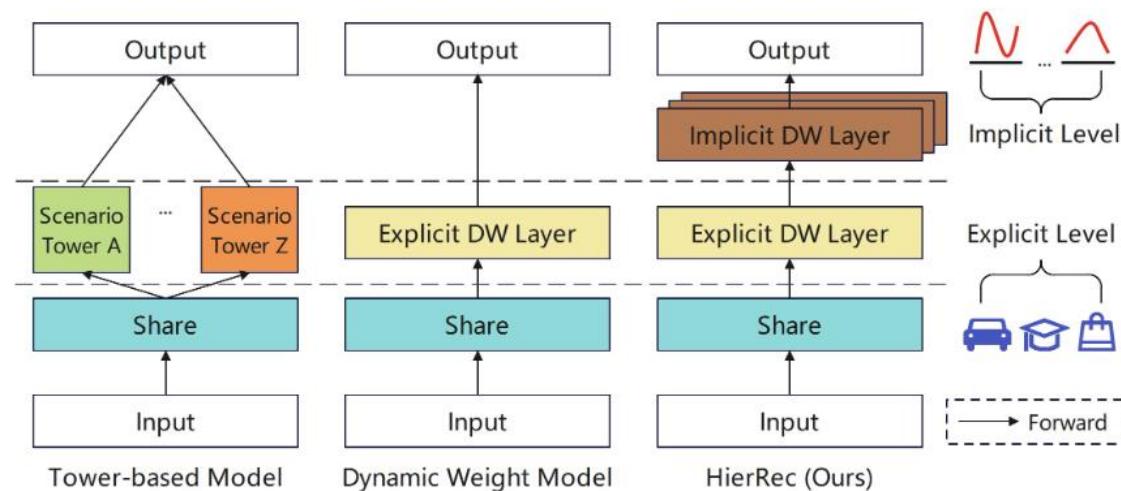
$$V_d^i = W_d^{v_l} \cdot I^i \cdot W_d^{v_r}$$

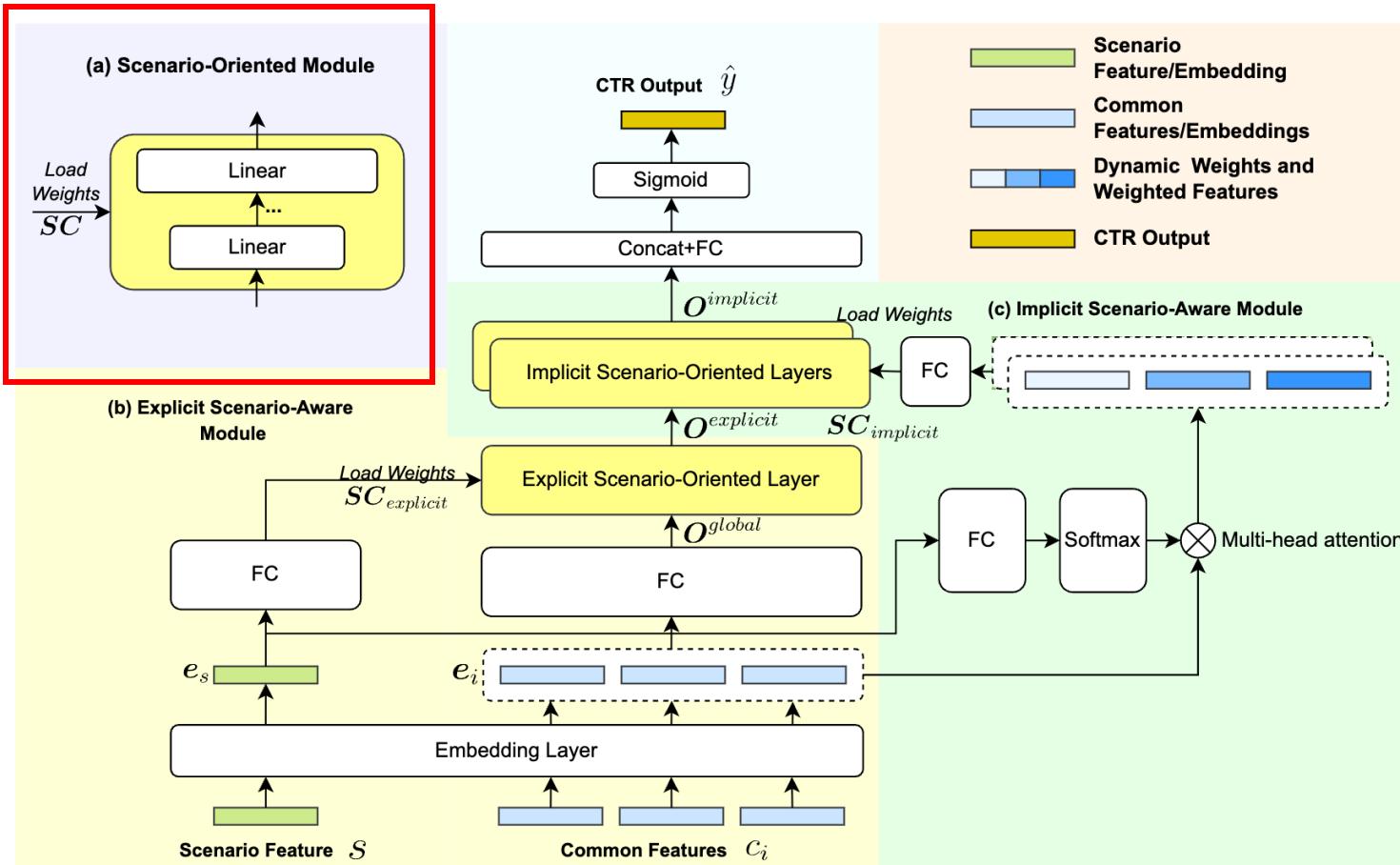
➤ Motivation

- Current multi-scenario models mainly rely on explicit scenario modeling based on manually defined scenario IDs (like ad slots or channels).
- These manual rules are coarse-grained, rigid, and potentially biased, and they ignore internal variations within each scenario

➤ Method

- Propose HierRec that models both explicit and implicit scenarios in a hierarchical and adaptive way

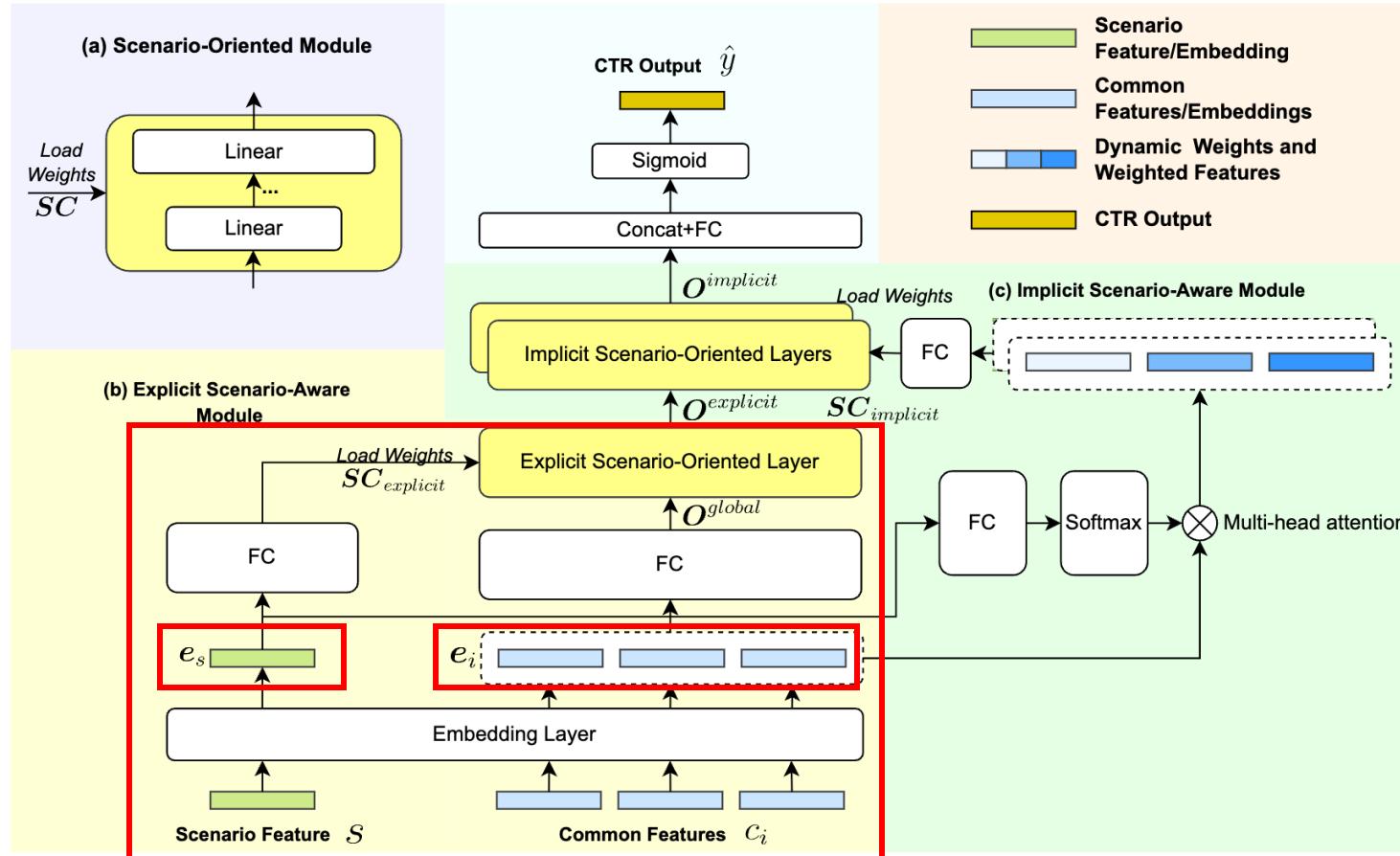




➤ Scenario-Oriented Module

- Adaptively generate parameters depending on scenario condition (SC)

$$\mathbf{W}_l, \mathbf{b}_l = \text{Reshape}(\mathbf{SC})[l] \quad l \in [1, L],$$



➤ Scenario-Oriented Module

- Adaptively generate parameters depending on scenario condition (SC)

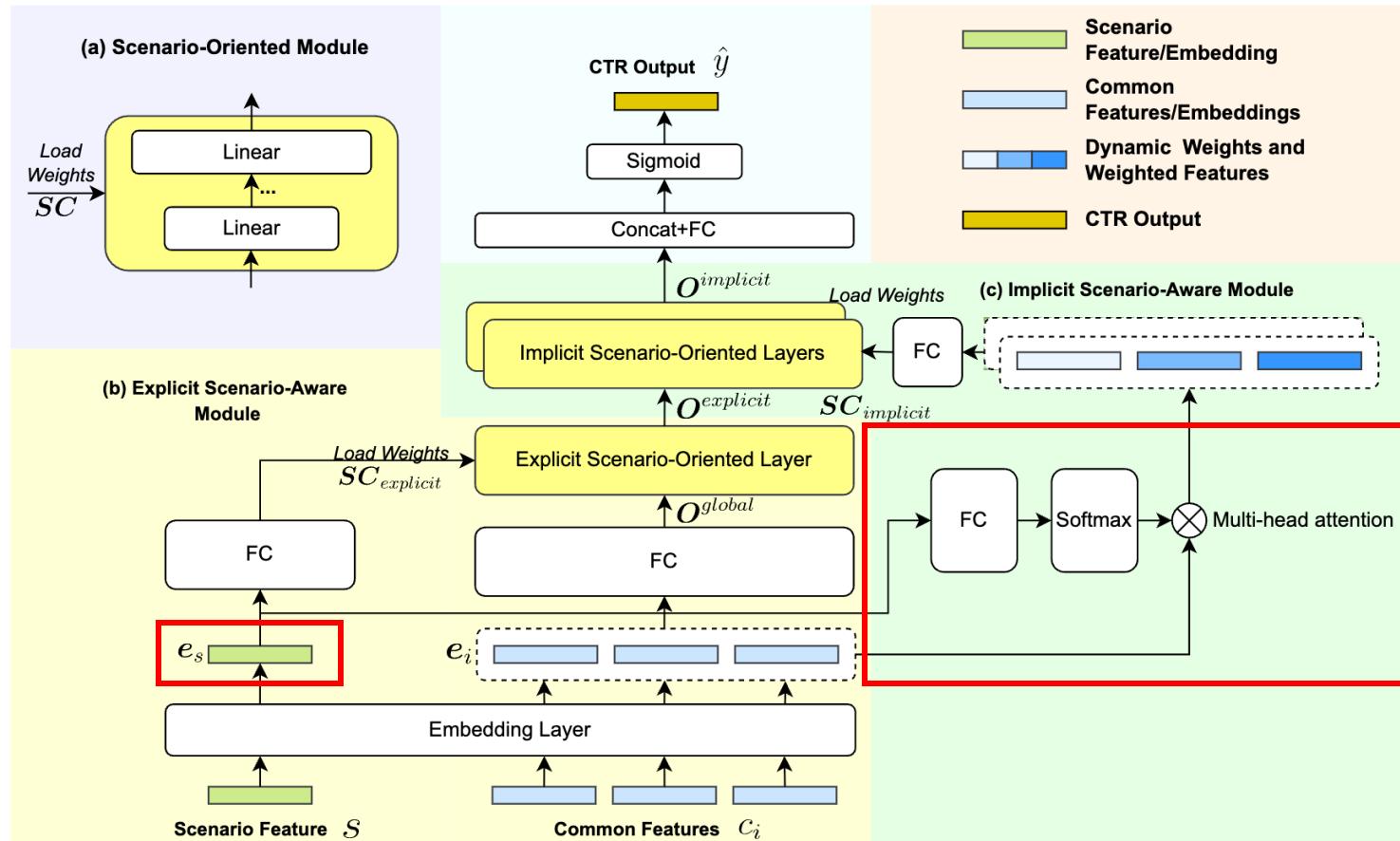
$$\mathbf{W}_l, \mathbf{b}_l = \text{Reshape}(\mathbf{SC})[l] \quad l \in [1, L],$$

➤ Explicit Scenario-Aware Module

- Model coarse-grained explicit scenario information

$$\begin{cases} \mathbf{e}_i = \mathbf{EM}_i \cdot \text{Onehot}(c_i), & i \in [1, I] \\ \mathbf{e}_s = \mathbf{EM}_s \cdot \text{Onehot}(s), \end{cases}$$

$$SC_{explicit} = FC(\mathbf{e}_s)$$

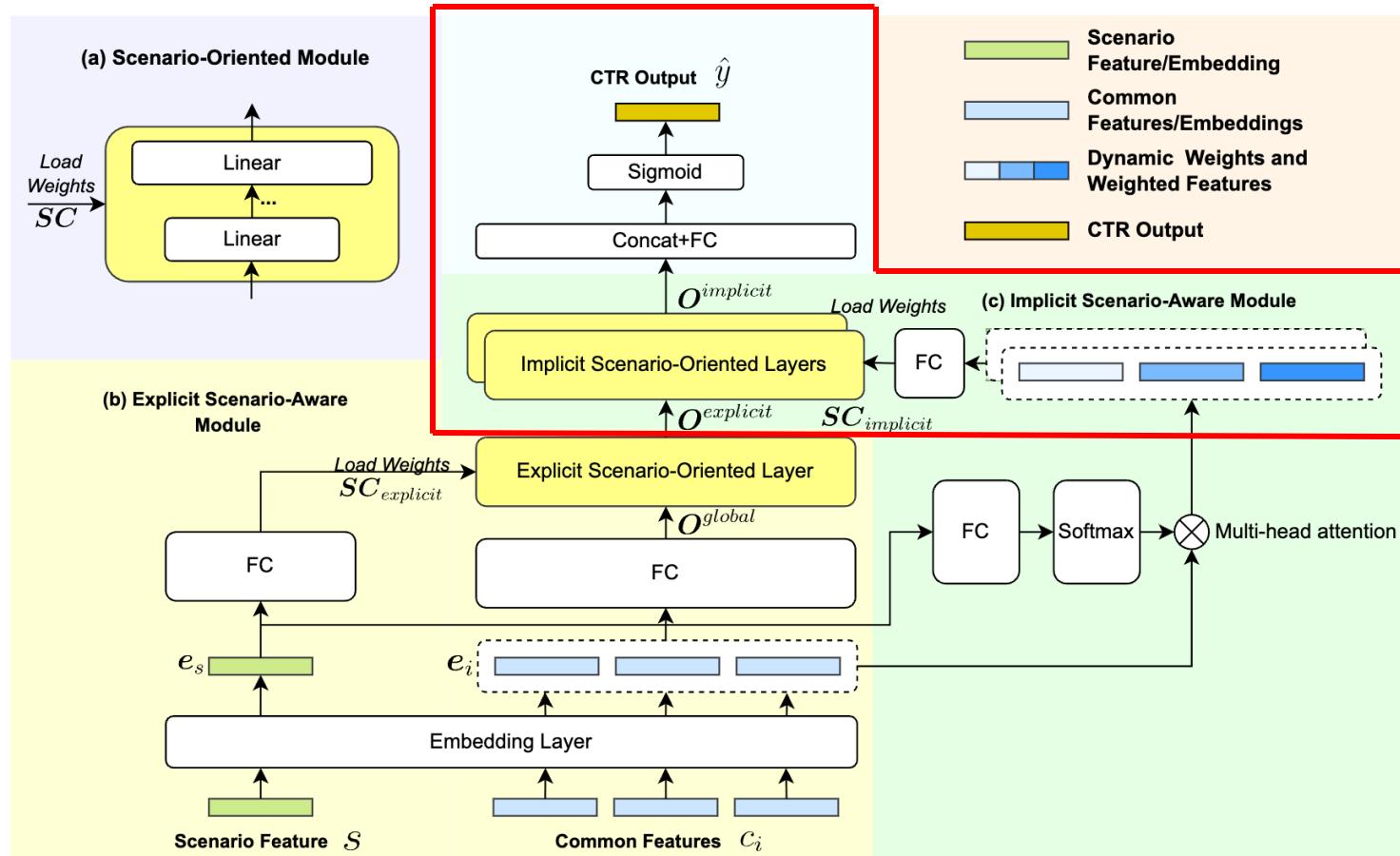


➤ Implicit Scenario-Aware Module

- Model fine-grained implicit scenario information

$$\begin{cases} \mathbf{weight}_{ori} = \text{Reshape}(FC(e_s)) \\ \mathbf{weight}_{norm}[g] = \text{Softmax}(\mathbf{weight}_{ori}[g]), \\ g \in [1, G] \end{cases}$$

$$IE = \mathbf{weight}_{norm} \otimes E_c,$$



➤ Implicit Scenario-Aware Module

- Model fine-grained implicit scenario information

$$\begin{cases} \mathbf{weight}_{ori} = \text{Reshape}(FC(e_s)) \\ \mathbf{weight}_{norm}[g] = \text{Softmax}(\mathbf{weight}_{ori}[g]), \\ g \in [1, G] \end{cases}$$

$$IE = \mathbf{weight}_{norm} \otimes E_c,$$

$$SC_{implicit}[g] = FC(IE[g]), \quad g \in [1, G].$$

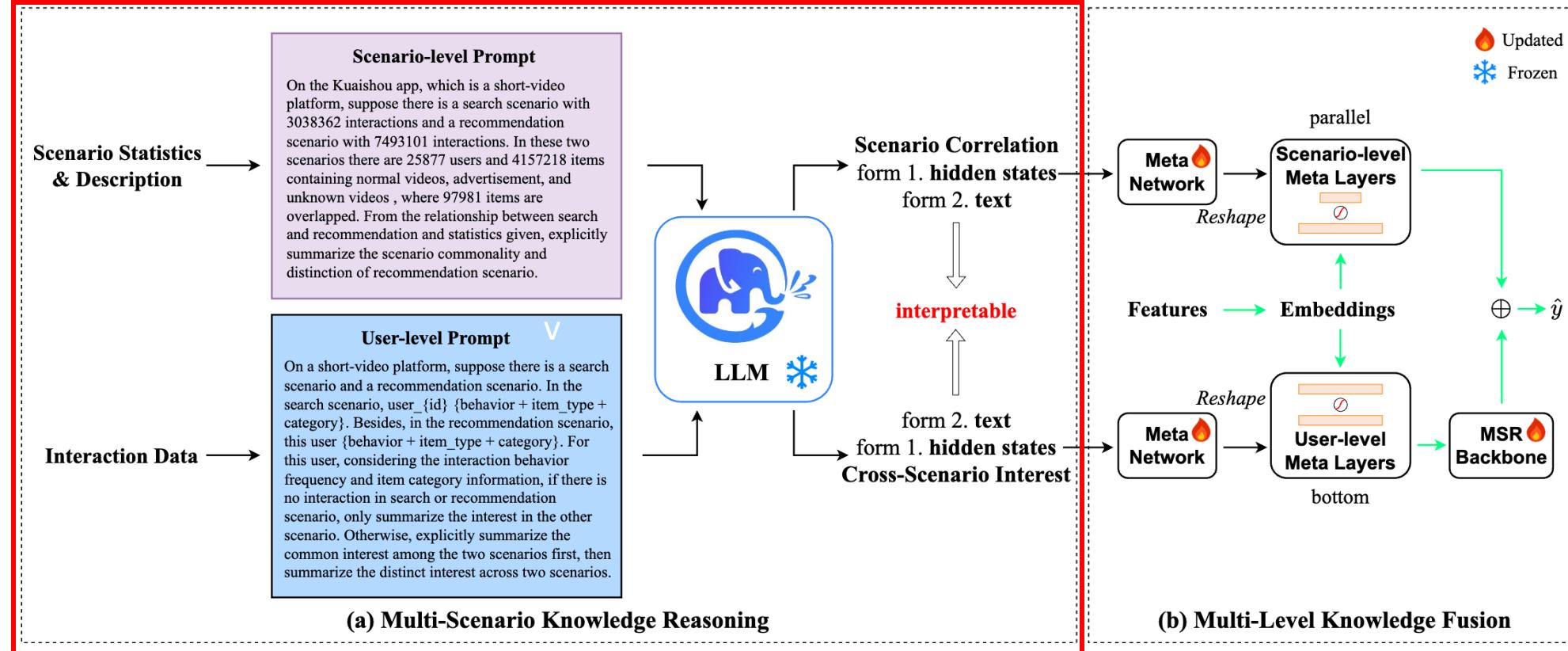
$$\hat{y} = \text{Sigmoid}(FC(\text{Concat}(O_1^{implicit}, \dots, O_G^{implicit}))).$$

➤ Motivation

- Insufficient scenario knowledge is incorporated
- Users' personalized preferences across scenarios tend to be ignored

➤ Method

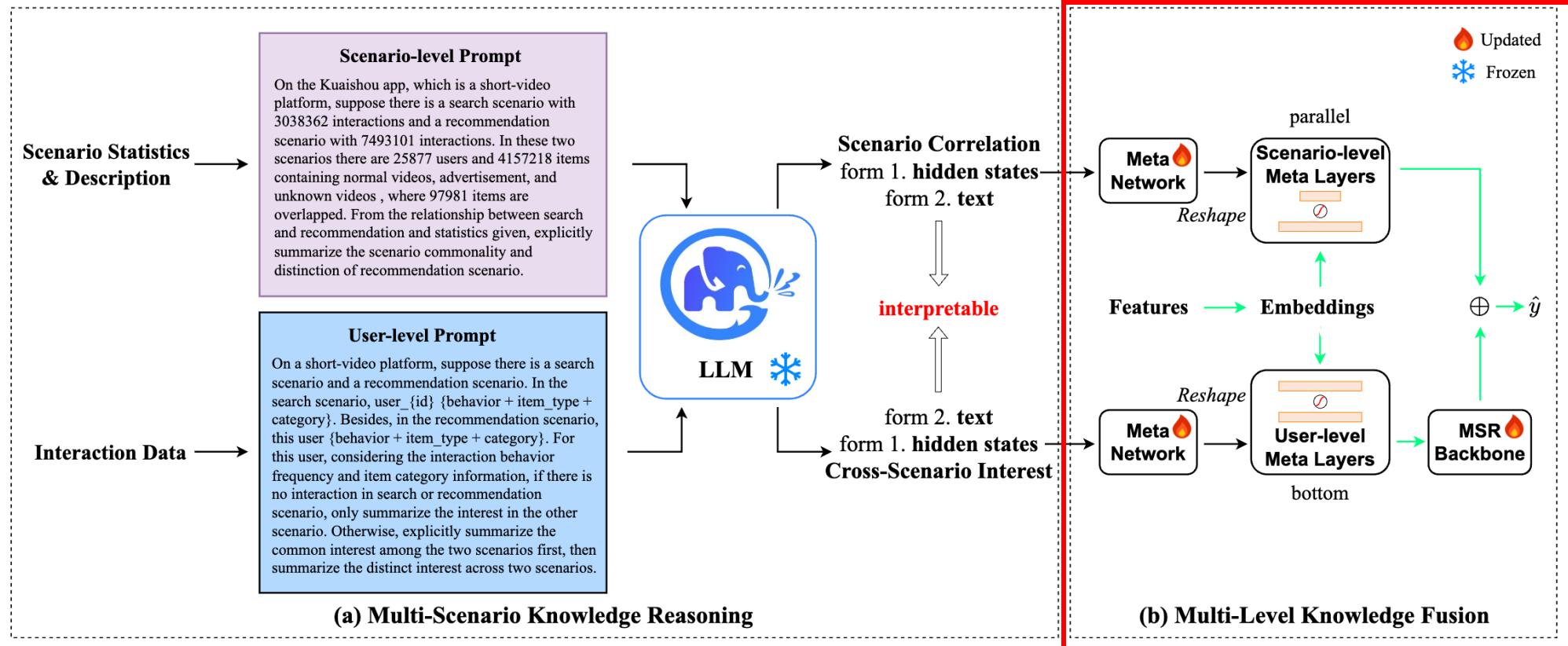
- Resorting to use LLM and hierarchical meta-networks
- Using LLM to grasp cross-scenario correlation and personalized preferences
- Using meta-network as a bridge connecting the semantic space in LLM and recommendation space in the multi-scenario backbone model



➤ Multi-scenario Knowledge Reasoning

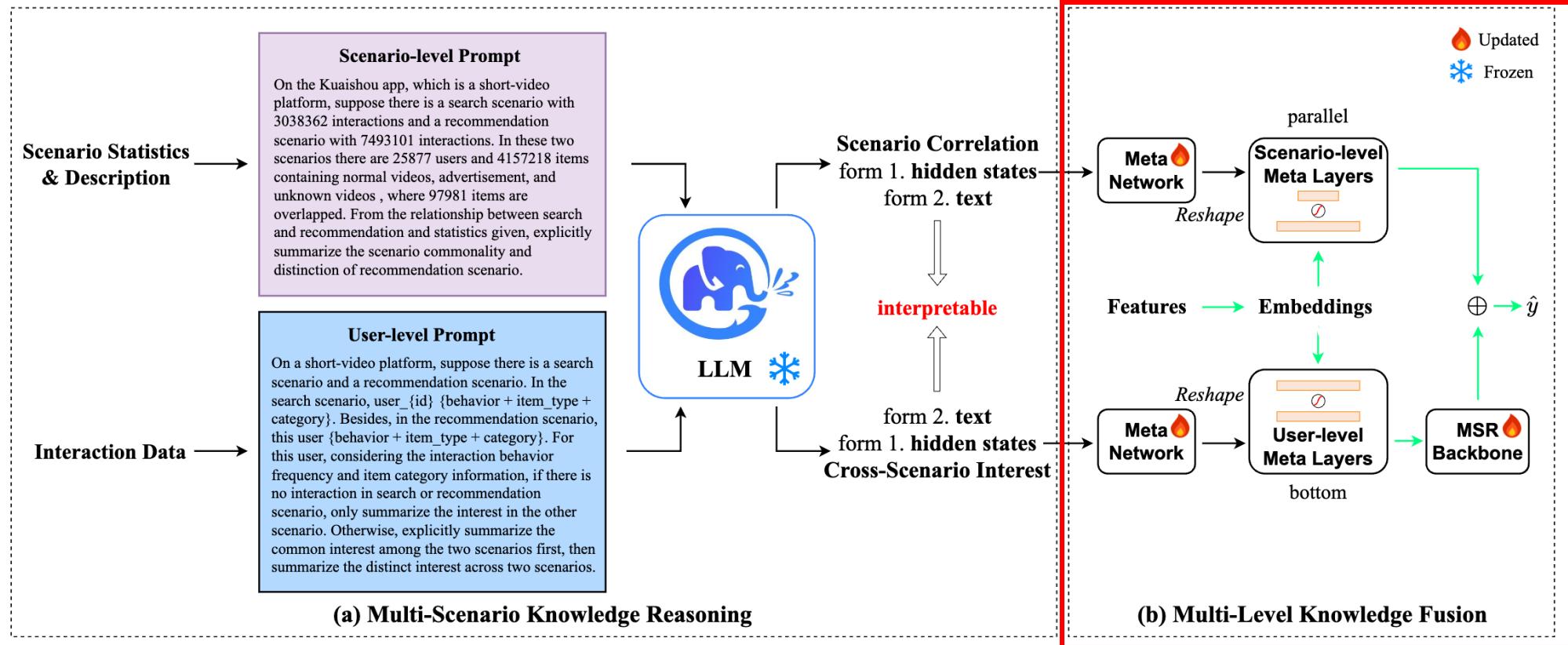
- Scenario-level prompt construction
- User-level prompt construction

LLM4MSR Details



- Multi-Level Knowledge Fusion
 - Meta-net generates meta layers to fuse the scenario- and user-level knowledge

$$\begin{aligned}
 \mathbf{h}_{mw}, \mathbf{h}_{mb} &= \text{Meta Network}(\mathbf{h}_{LLM}), \\
 \mathbf{W}_l^{(i)} &= \text{Reshape}(\mathbf{h}_{mw}) \\
 \mathbf{b}_l^{(i)} &= \text{Reshape}(\mathbf{h}_{mb}), i \in \{1, 2, \dots, K\}
 \end{aligned}$$

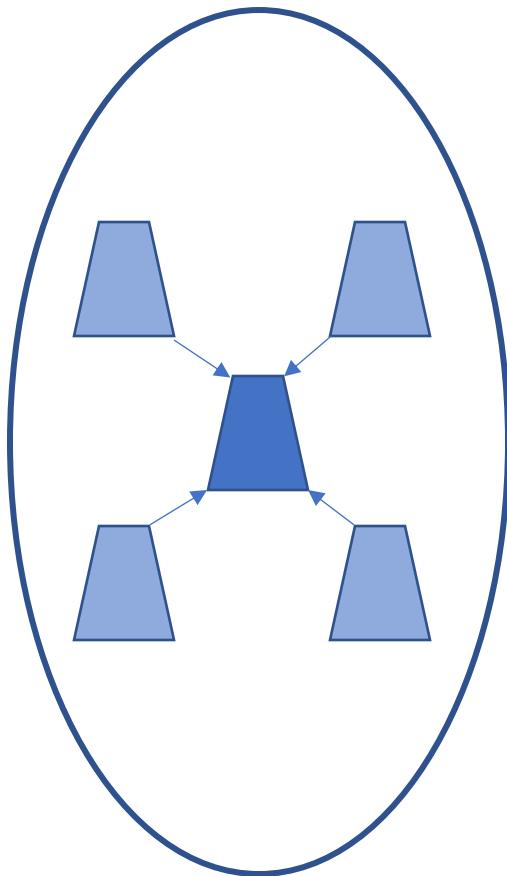


➤ Multi-Level Knowledge Fusion

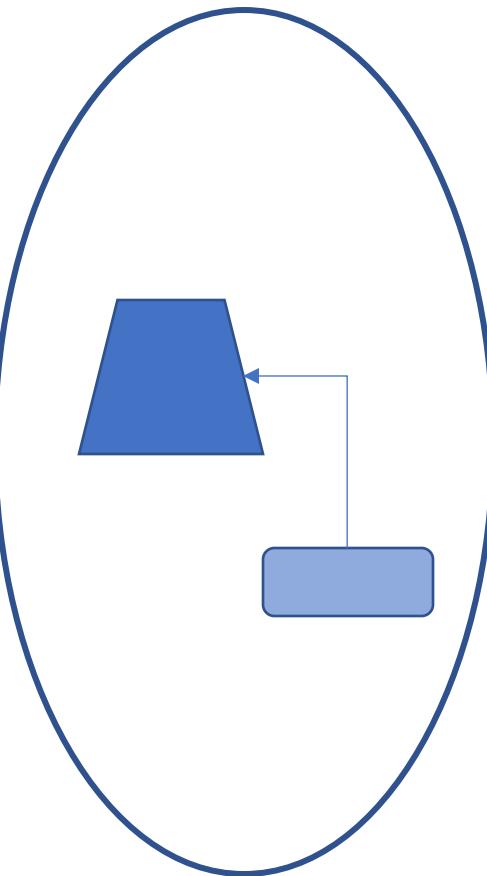
- Prediction $\mathbf{h}^{(i)} = \sigma(\mathbf{W}_l^{(i)} \mathbf{h}^{(i-1)} + \mathbf{b}_l^{(i)}), i \in \{1, 2, \dots, K\}$
 $\mathbf{h} = \text{MSR}(\mathbf{h}_u^{(K)}),$

$$\hat{y} = \sigma'(\alpha \cdot \mathbf{h}_s^{(K)} + (1 - \alpha) \cdot \mathbf{h}),$$

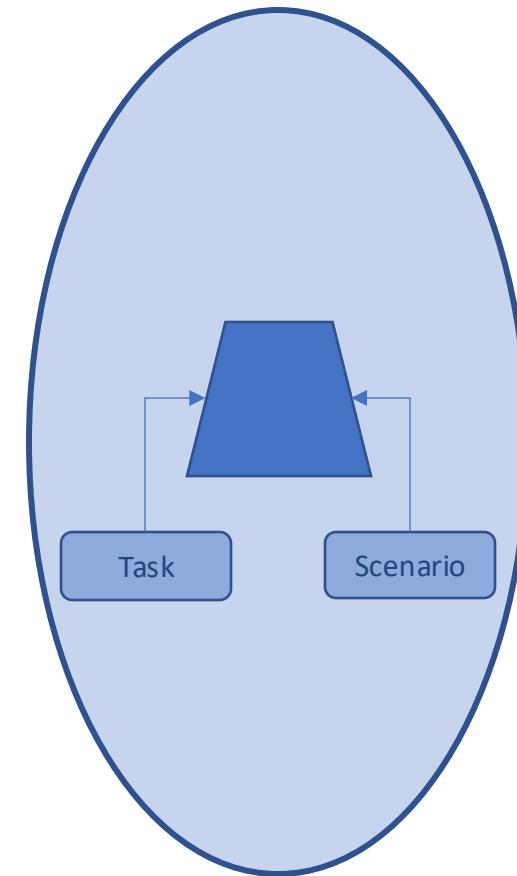
Table of Contents



Shared-specific network paradigm
 $wL(E^{Merge}, \Theta, \Theta^t, (\Theta^{shared}, \Theta^{specific}))$

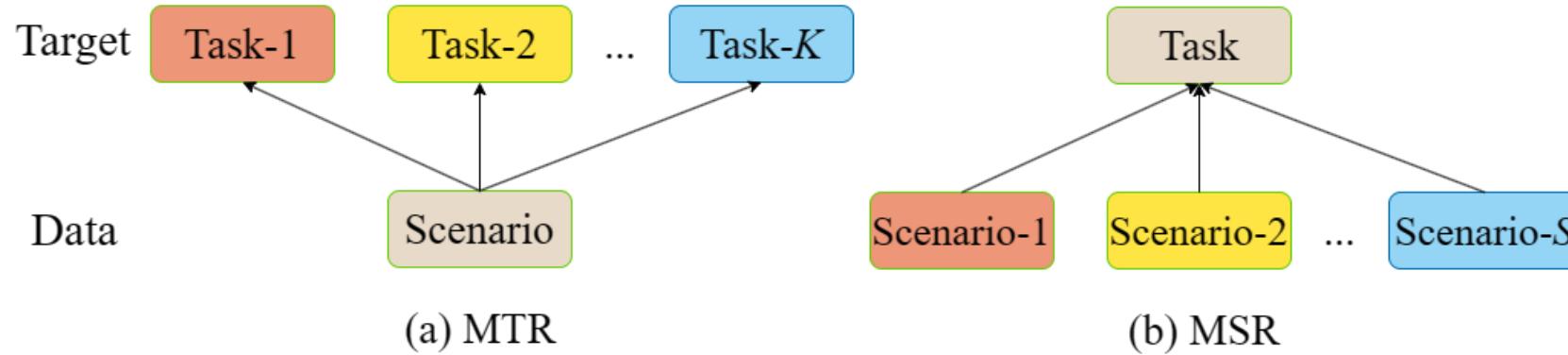


Dynamic weight
 $wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$



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

Multi-Scenario & Multi-Task Studies

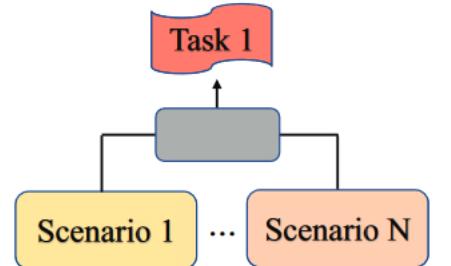


➤ Target

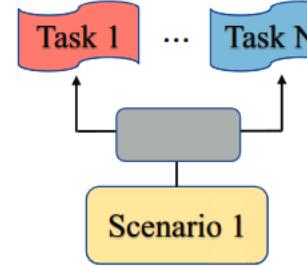
- Develop a unified framework that could realize both MSL and MTL requirements

➤ Methods

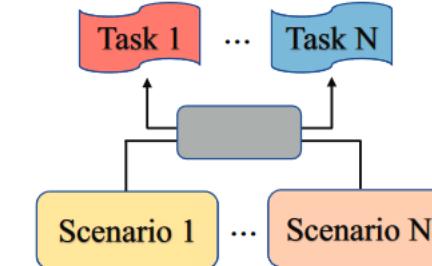
- Propose AESM², a flexible hierarchical structure where the multi-task layers are stacked over the multi-scenario layers
- General expert selection algorithm



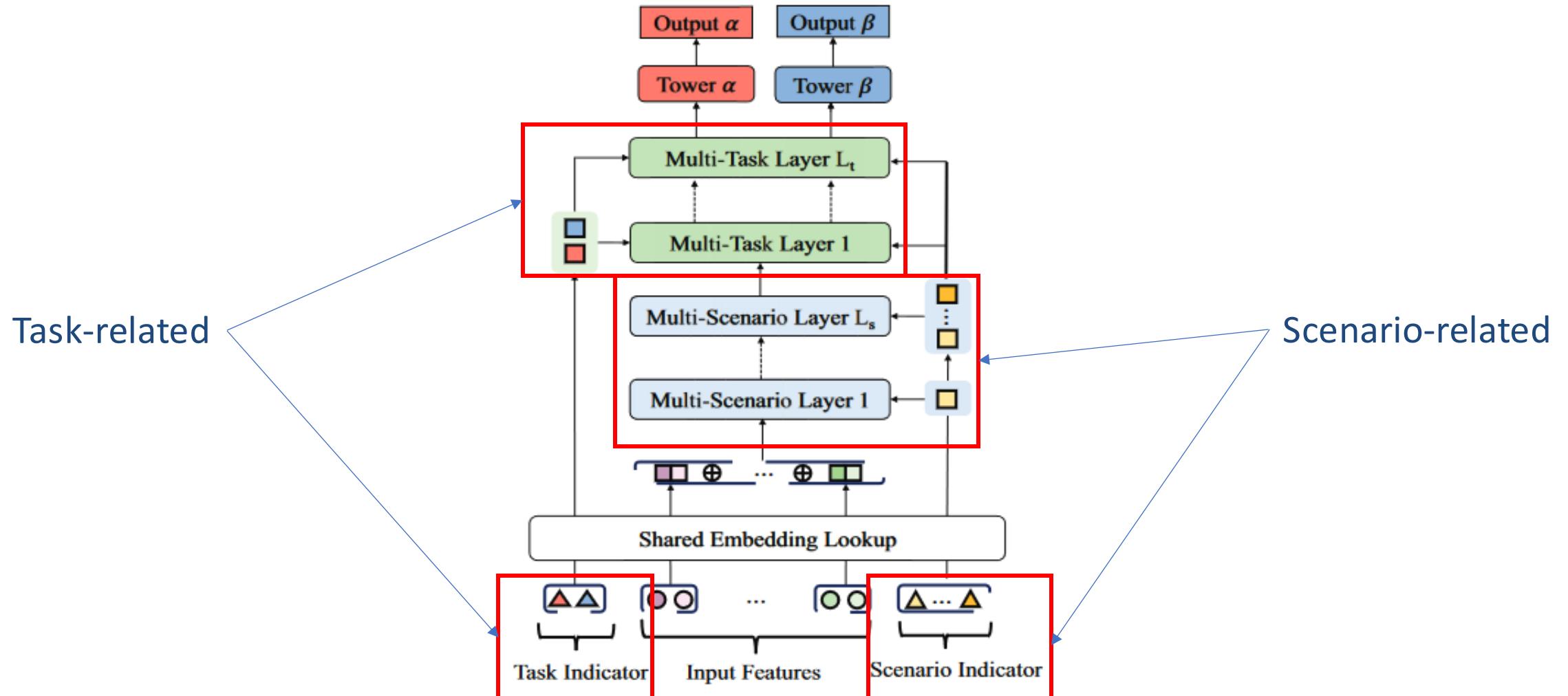
(a) MSL



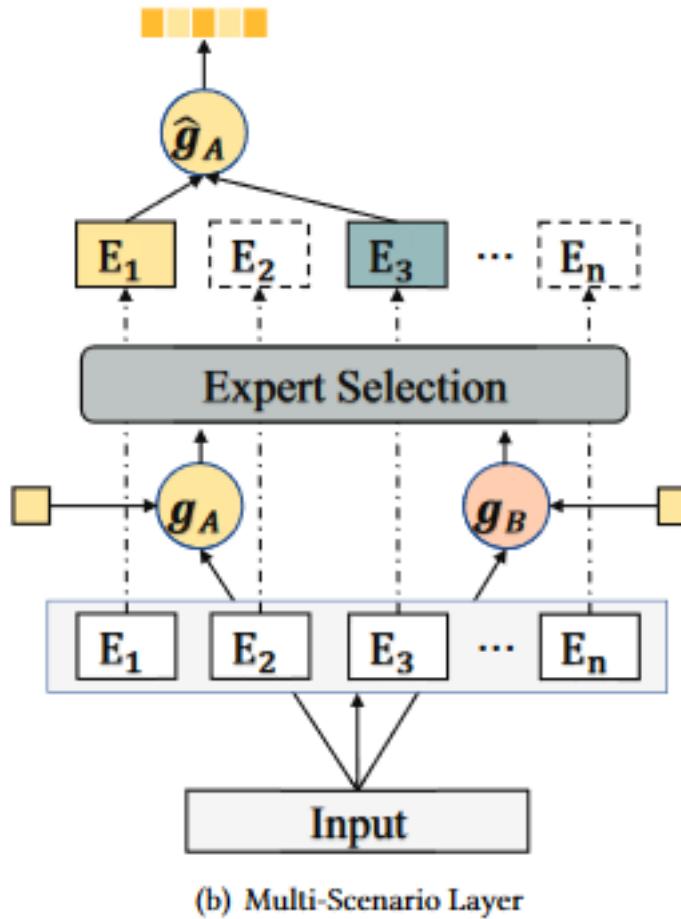
(b) MTL



(c) Both MSL & MTL



Multi-Scenario Layer



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

$$\mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_m]$$

$$\mathbf{g}_j = \mathbf{S}_j[\mathbf{x}, \mathbf{s}] + \eta_j$$

$$\tilde{\mathbf{G}} = \text{softmax}(\mathbf{G})$$

- Expert selection

$$\mathcal{E}_{sp} = \text{TopK}(h_1^p, \dots, h_n^p)$$

$$h_k^p = -KL(\mathbf{p}_j, \tilde{\mathbf{G}}[k, :])$$

$$\mathcal{E}_{sh} = \text{TopK}(h_1^q, \dots, h_n^q)$$

$$h_k^q = -KL(\mathbf{q}_j, \tilde{\mathbf{G}}[k, :])$$

$$\mathbf{p}_j (\text{e.g., } [1, \dots, 0])$$

$$\mathbf{q}_j = [1/m, \dots, 1/m]$$

Specific

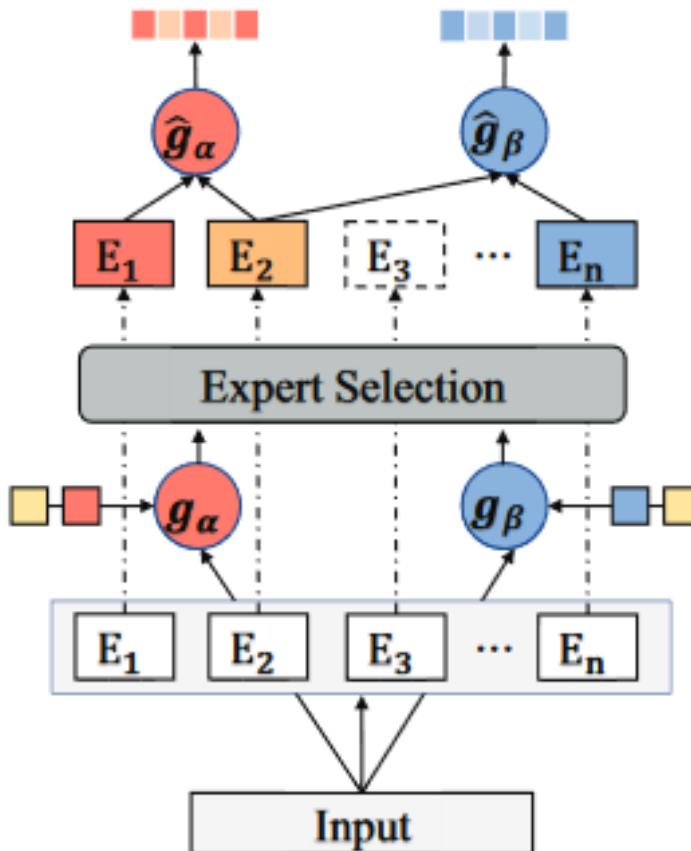
Shared

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

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

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:



$$g_k = T_k[x, s, t_k] + \eta_k$$

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

- Output layer

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

➤ Motivation

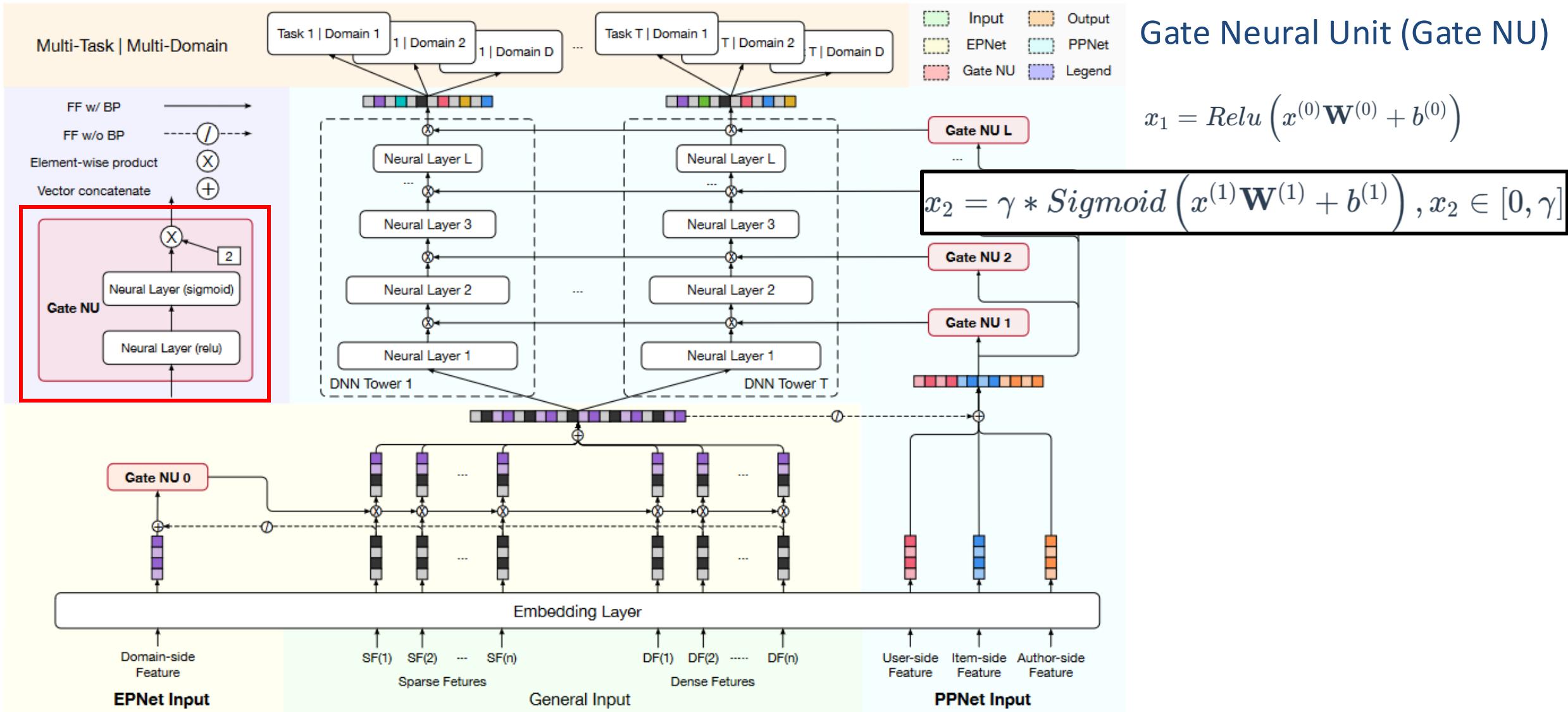
- The imperfectly double seesaw phenomenon
- More accurate personalization estimates can alleviate the imperfectly double seesaw problem

➤ Target

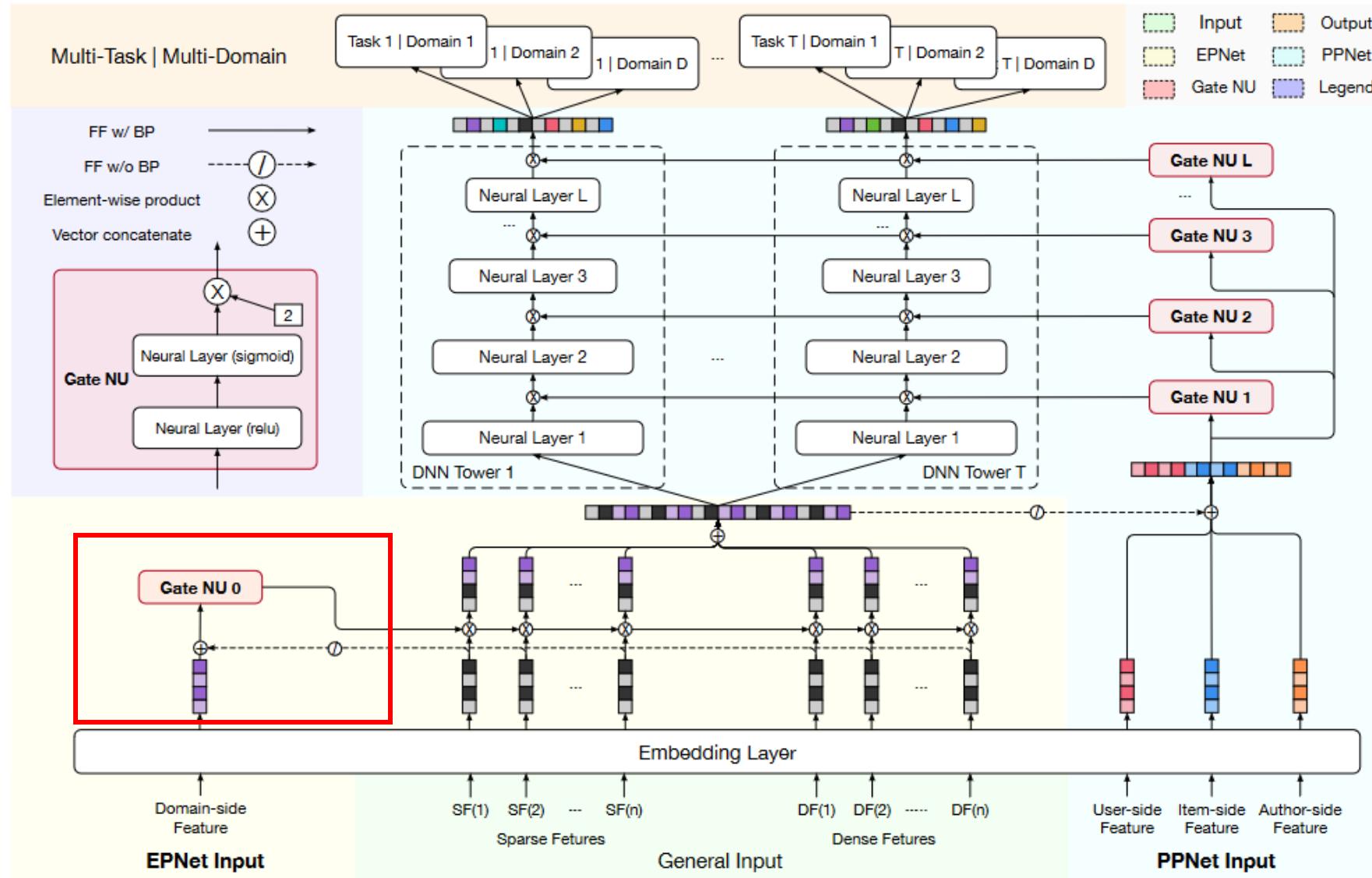
- Jointly model multi-domain and multi-task
- an efficient, low-cost deployment and plug-and-play method that can be injected in any network.



PEPNet Details



PEPNet Details



EPNet

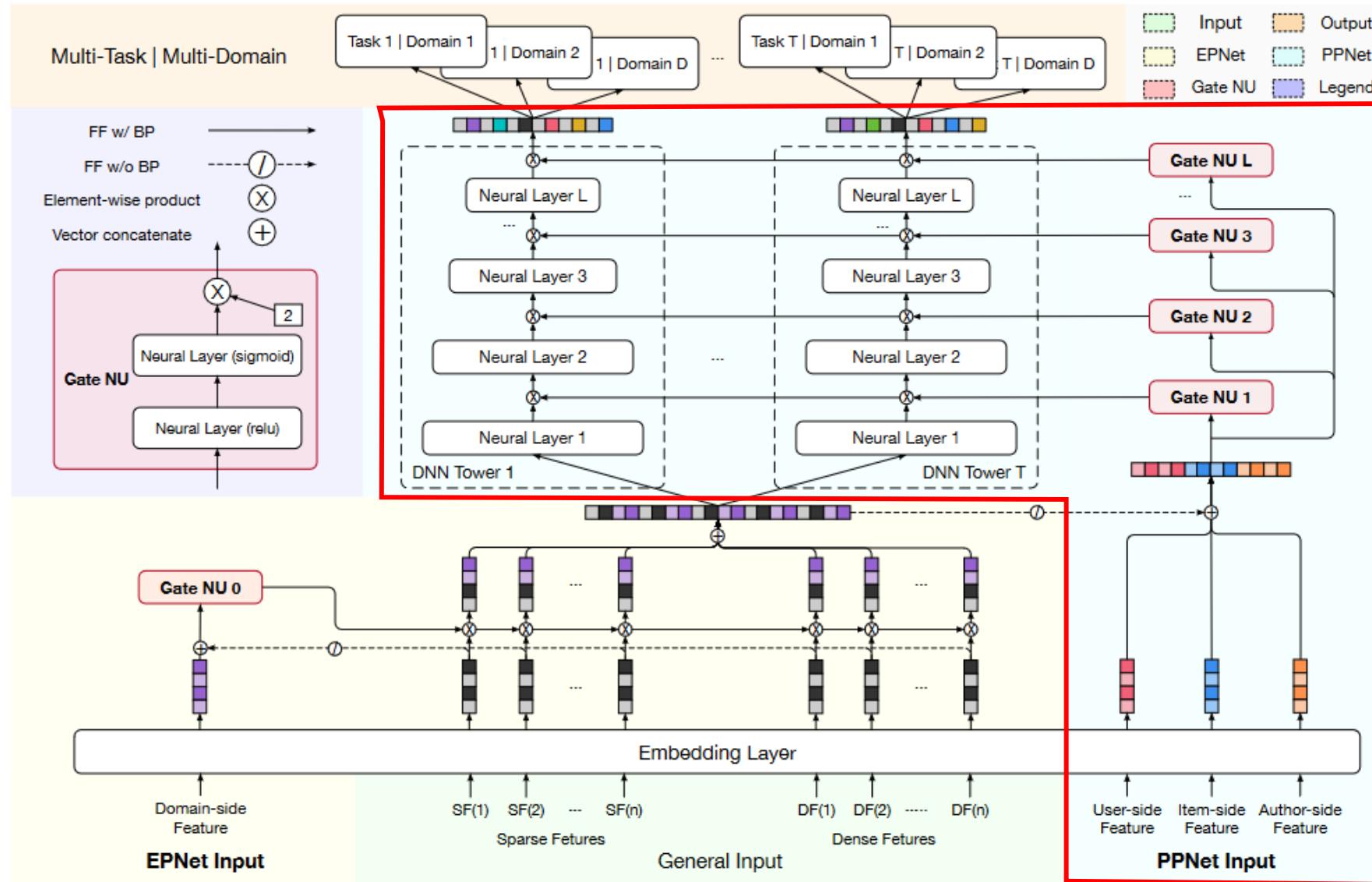
$$\mathbf{E} = E(\mathcal{F}_S) \oplus E(\mathcal{F}_D)$$

Embeddings of sparse features and dense features

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

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

PEPNet Details



PPNet

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

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

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

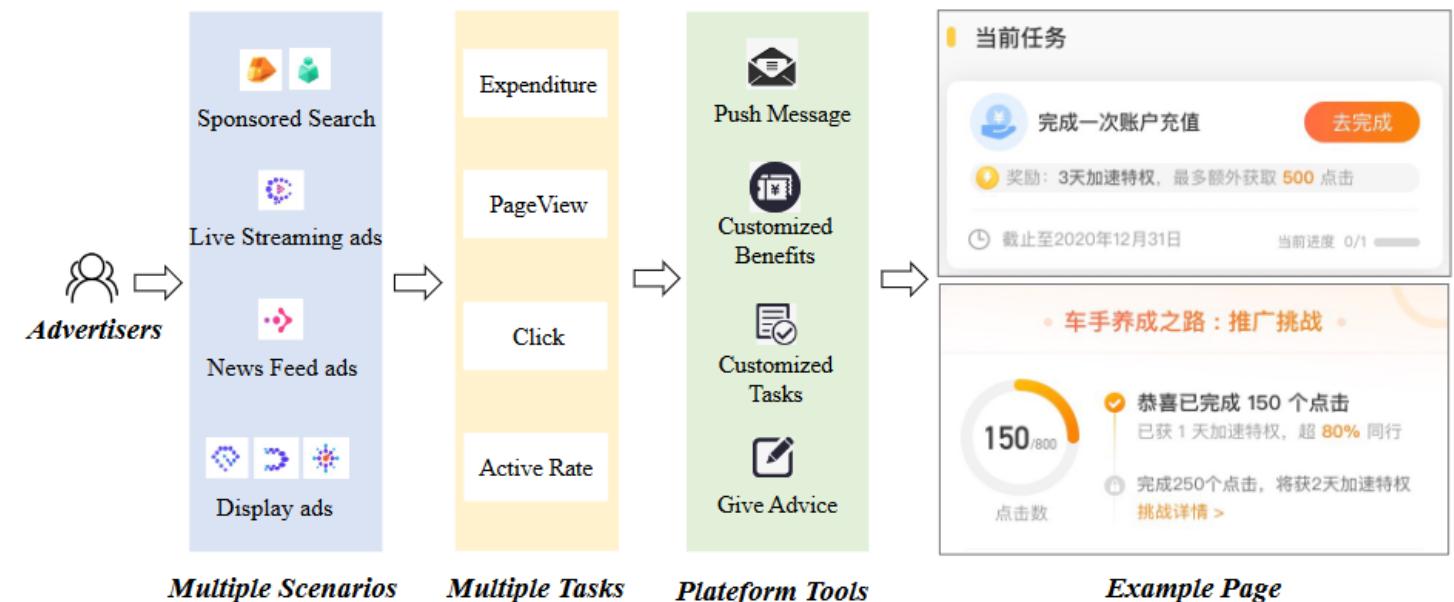
$$\mathbf{H}^{(l+1)} = f(\mathbf{O}_{pp}^{(l)} \mathbf{W}^{(l)} + b^{(l)}), l \in \{1, \dots, L\}$$

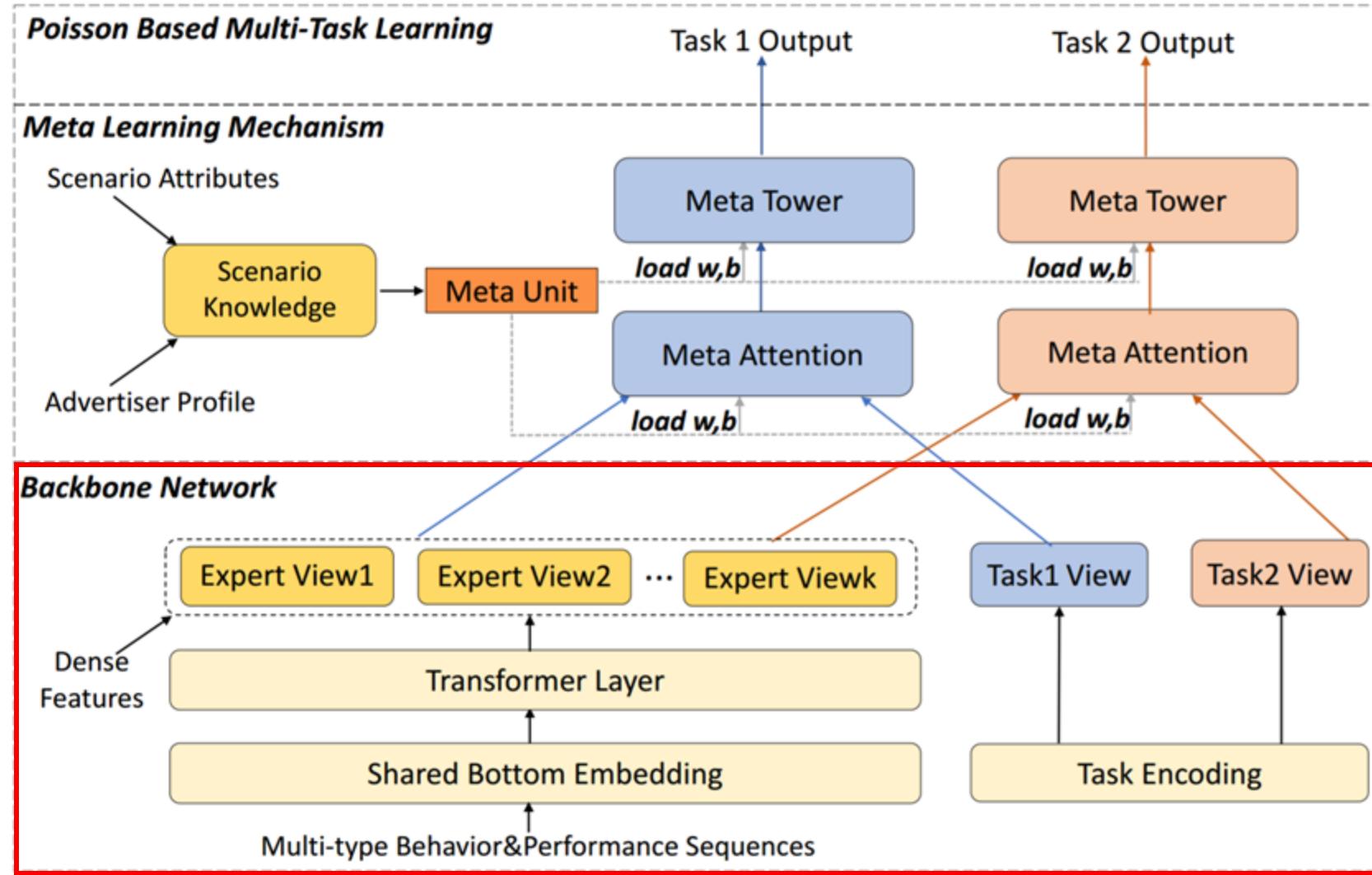
➤ Motivation

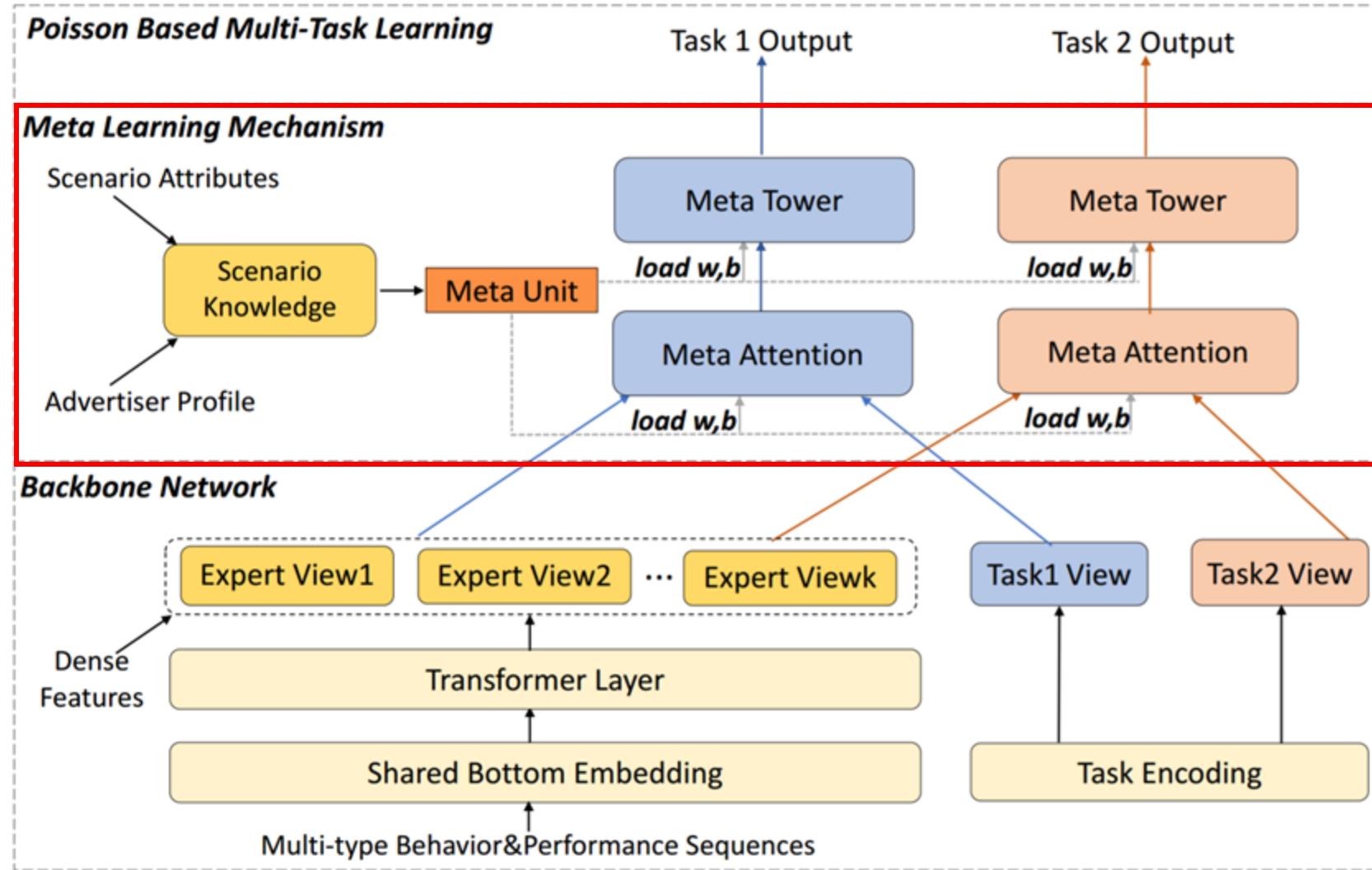
- Less attention has been drawn to advertisers
- Major e-commerce platforms provide multiple marketing scenarios.

➤ Methods

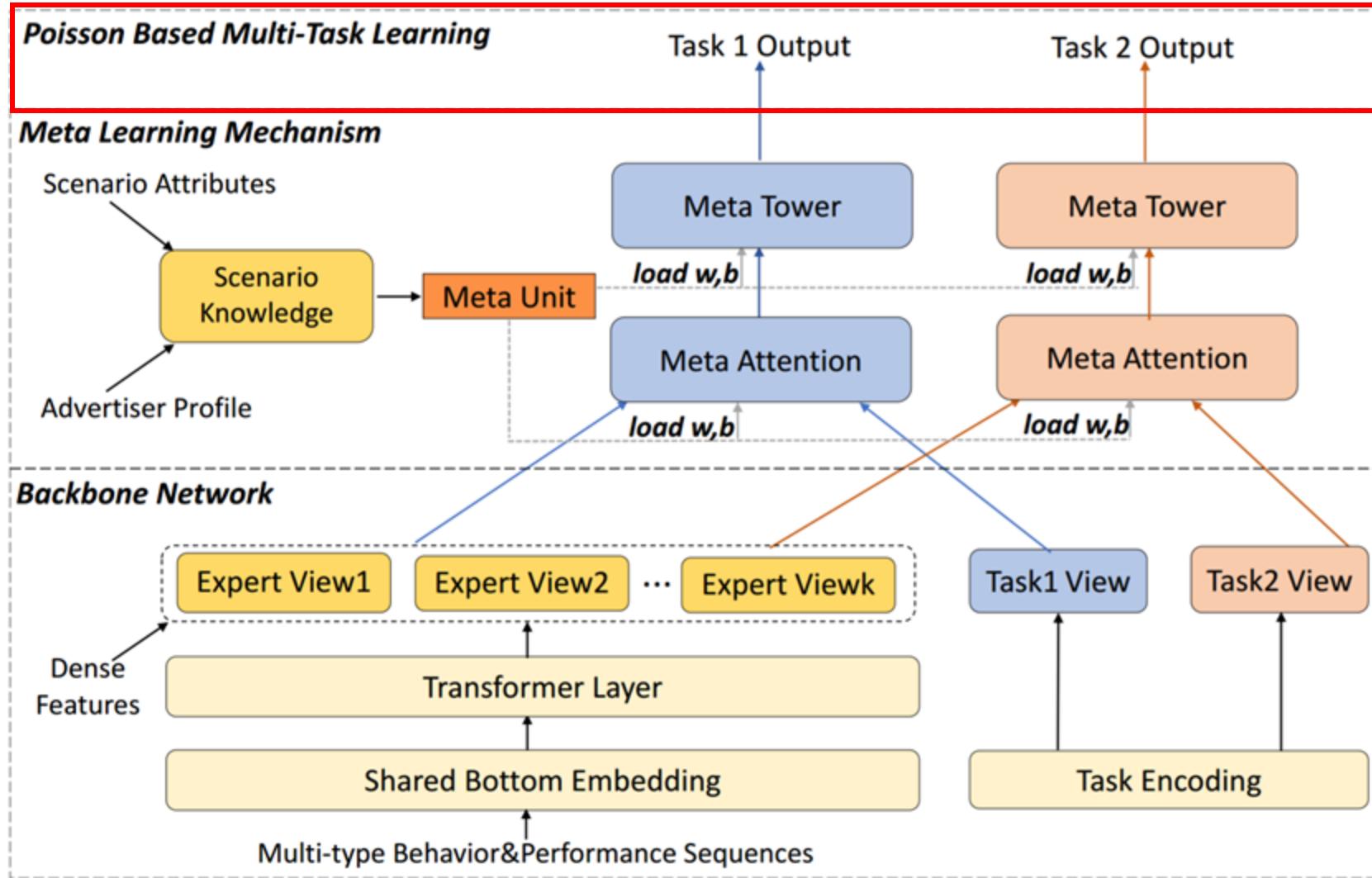
- Meta unit
- Meta attention module
- Meta tower module

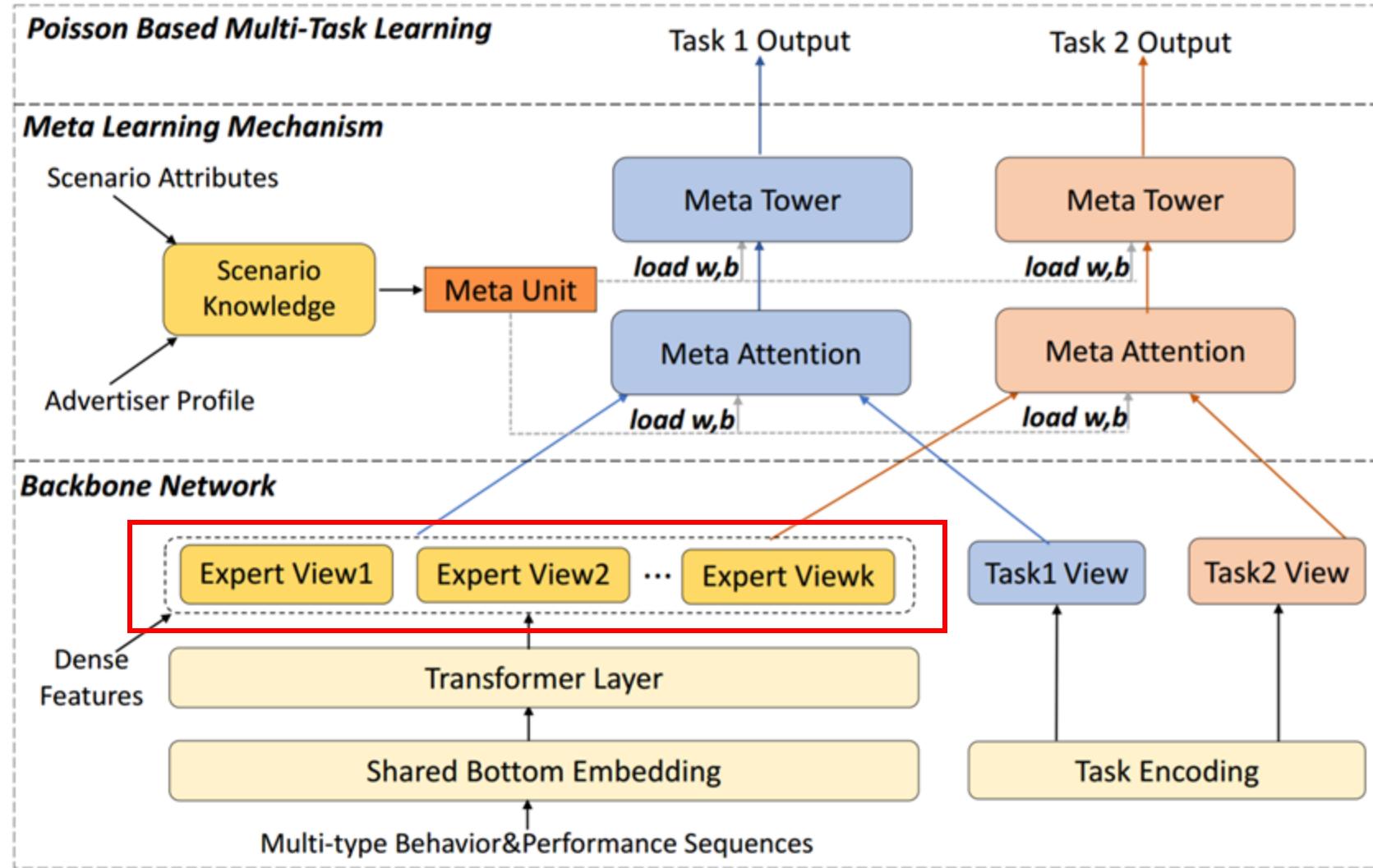






M2M Overview

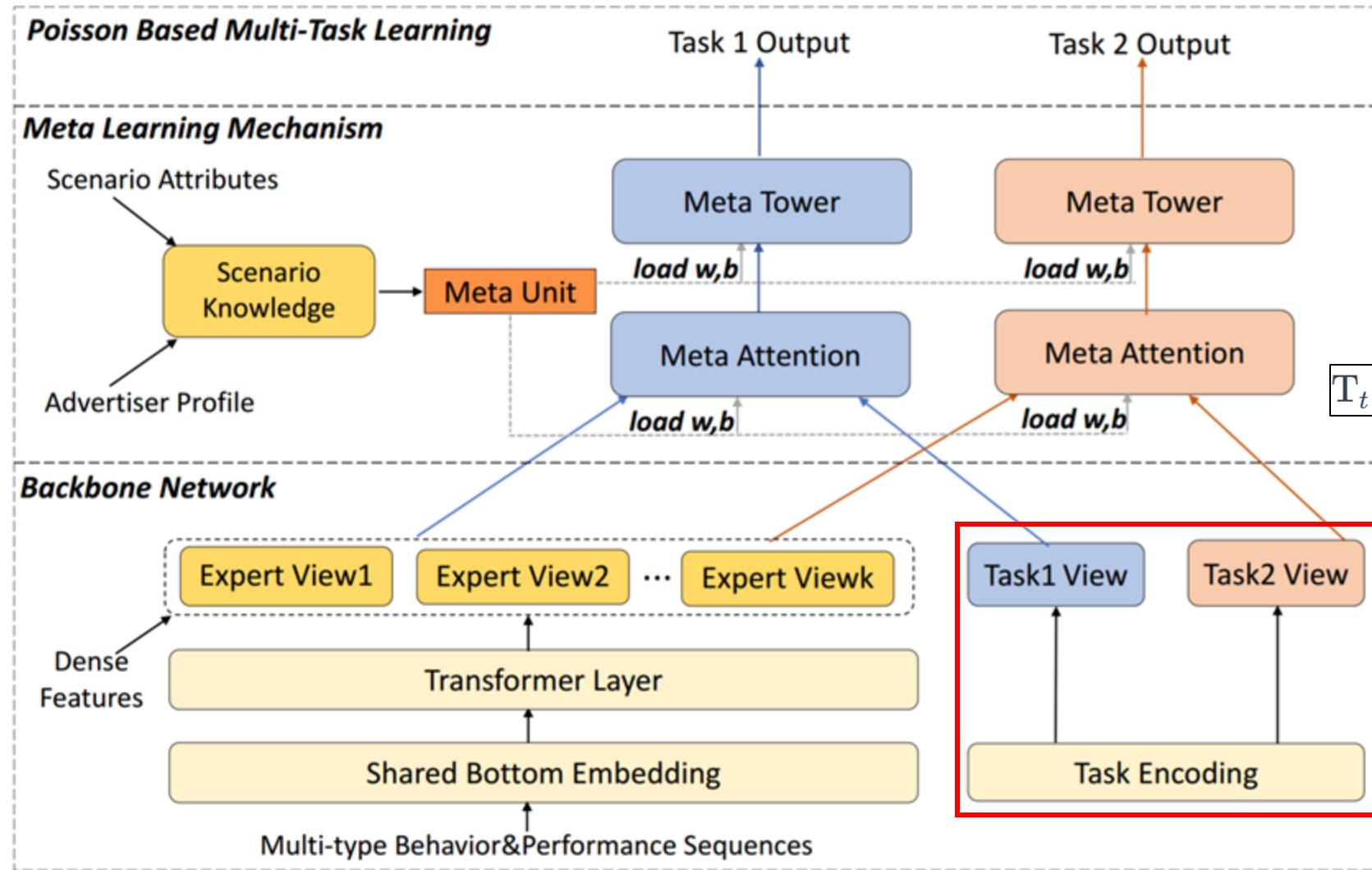




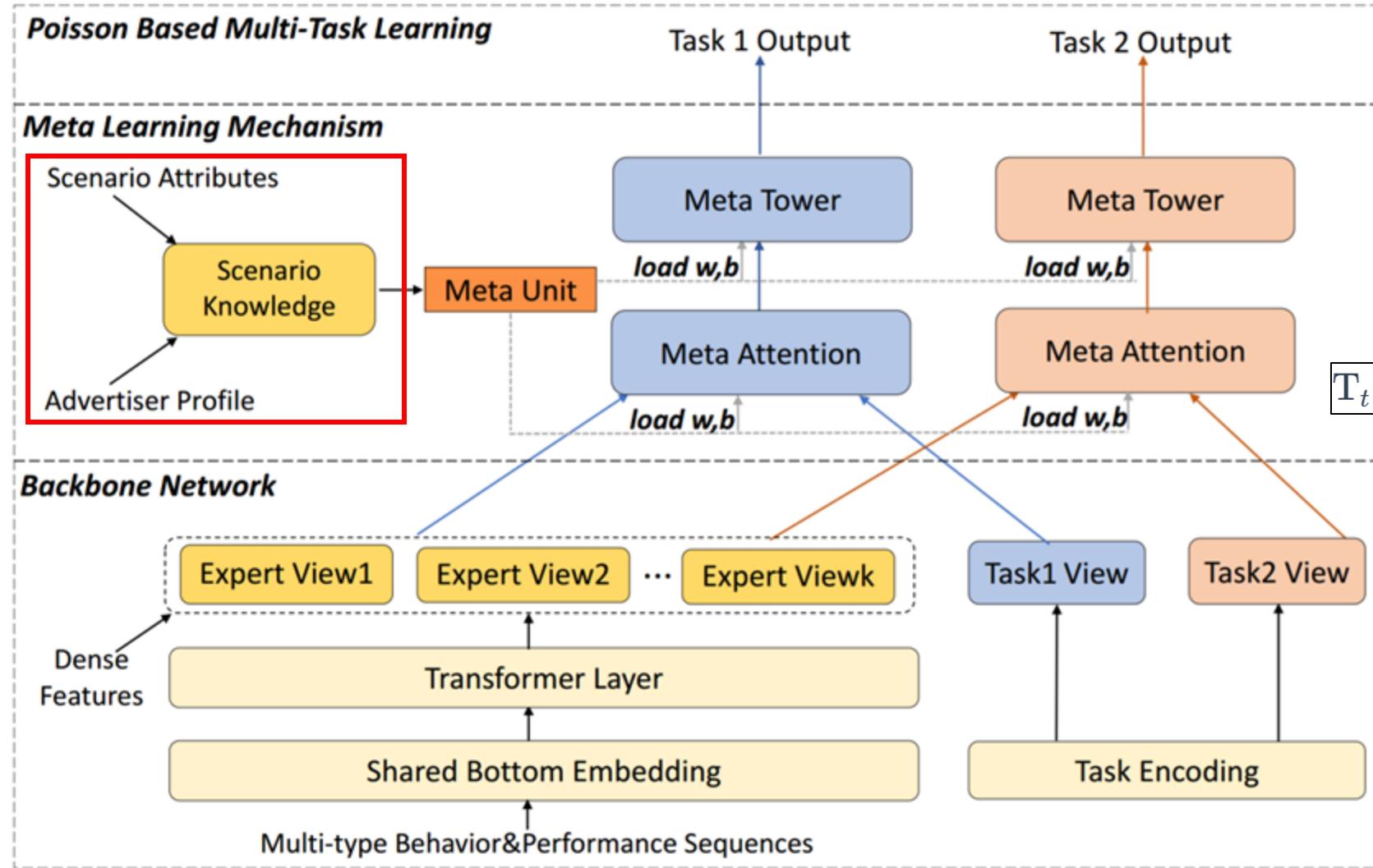
Backbone Network
Expert View Representation

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

F is the output of transformer layer



M2M Details



Backbone Network
Expert View Representation

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

F is the output of transformer layer

Task View Representation

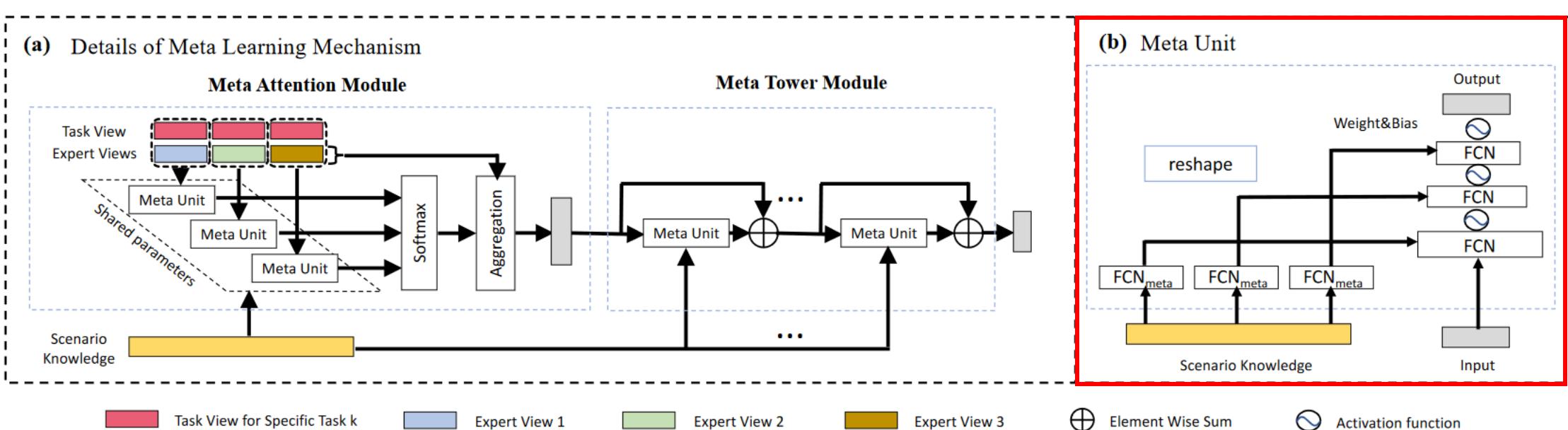
$$T_t = f_{MLP}(Embedding(t)), \forall t \in 1, 2, \dots, m$$

Scenario Knowledge Representation

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

Meta Unit

$$\mathbf{h}_{output} = \mathbf{h}^K = Meta(\mathbf{h}_{input})$$

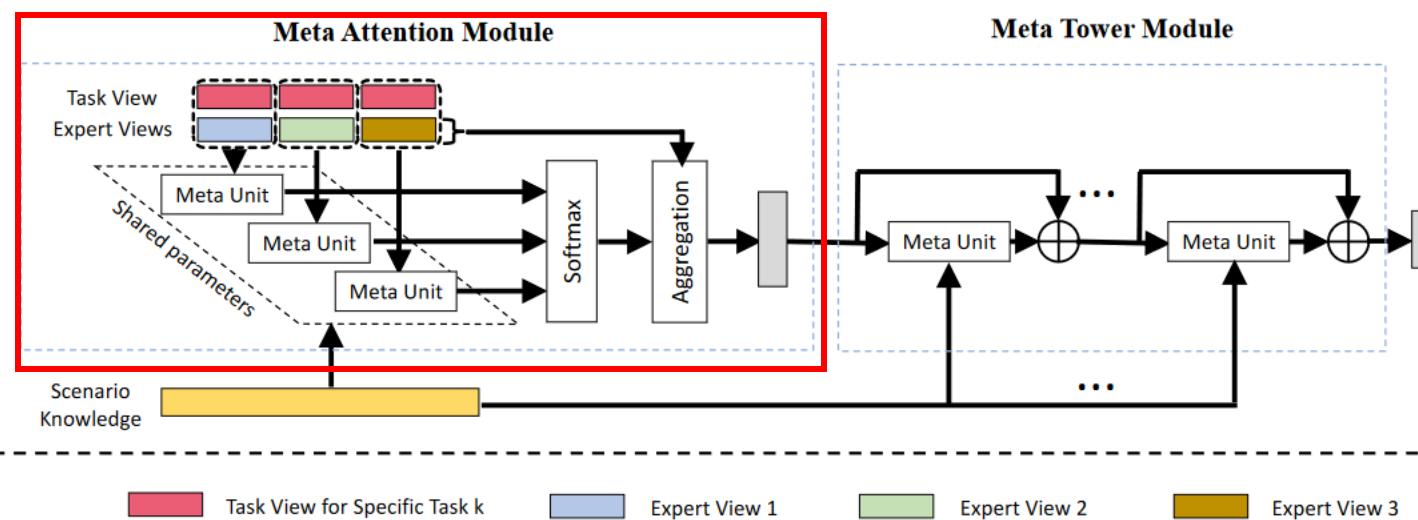


Meta Attention Module

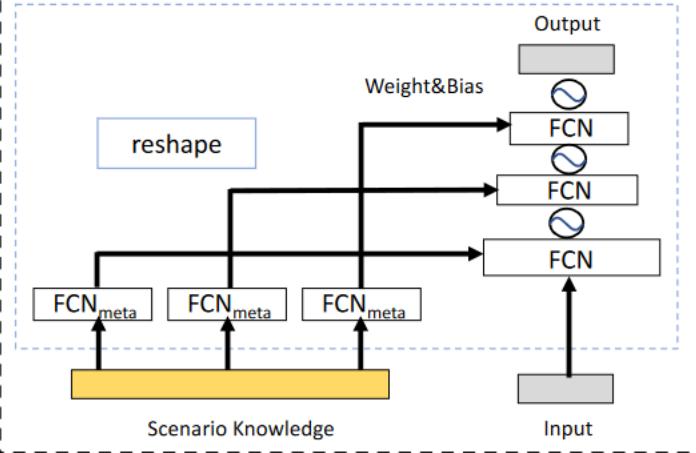
$$a_{t_i} = \mathbf{v}^T \text{Meta}_t([\mathbf{E}_i \parallel \mathbf{T}_t])$$

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

(a) Details of Meta Learning Mechanism



(b) Meta Unit



Meta Attention Module

$$a_{t_i} = \mathbf{v}^T \text{Meta}_t([\mathbf{E}_i \| \mathbf{T}_t])$$

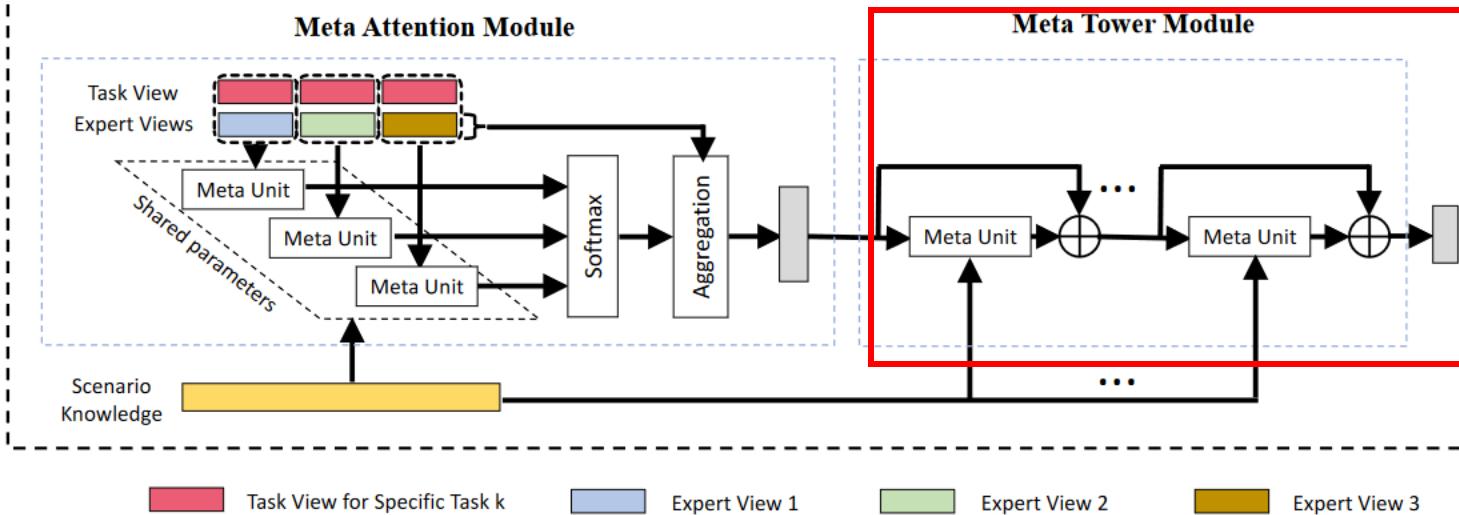
$$\alpha_{t_i} = \frac{\exp(a_{t_i})}{\sum_{j=1}^M \exp(a_{t_j})}, \quad \mathbf{R}_t = \sum_{i=1}^k \alpha_{t_i} \mathbf{E}_i$$

Meta Tower Module

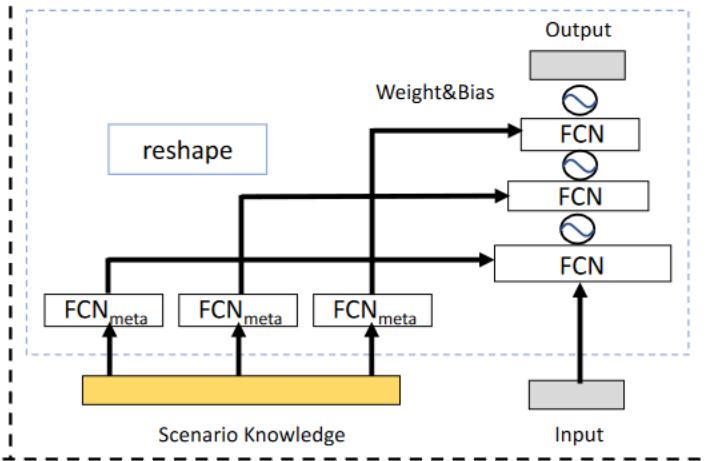
$$\mathbf{L}_t^{(0)} = \mathbf{R}_t$$

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

(a) Details of Meta Learning Mechanism



(b) Meta Unit

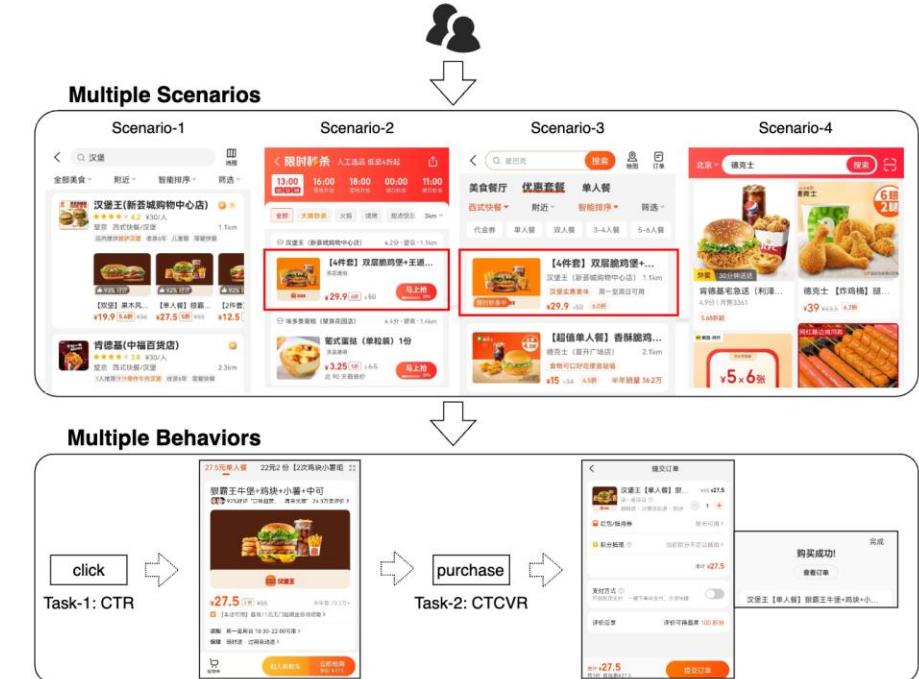


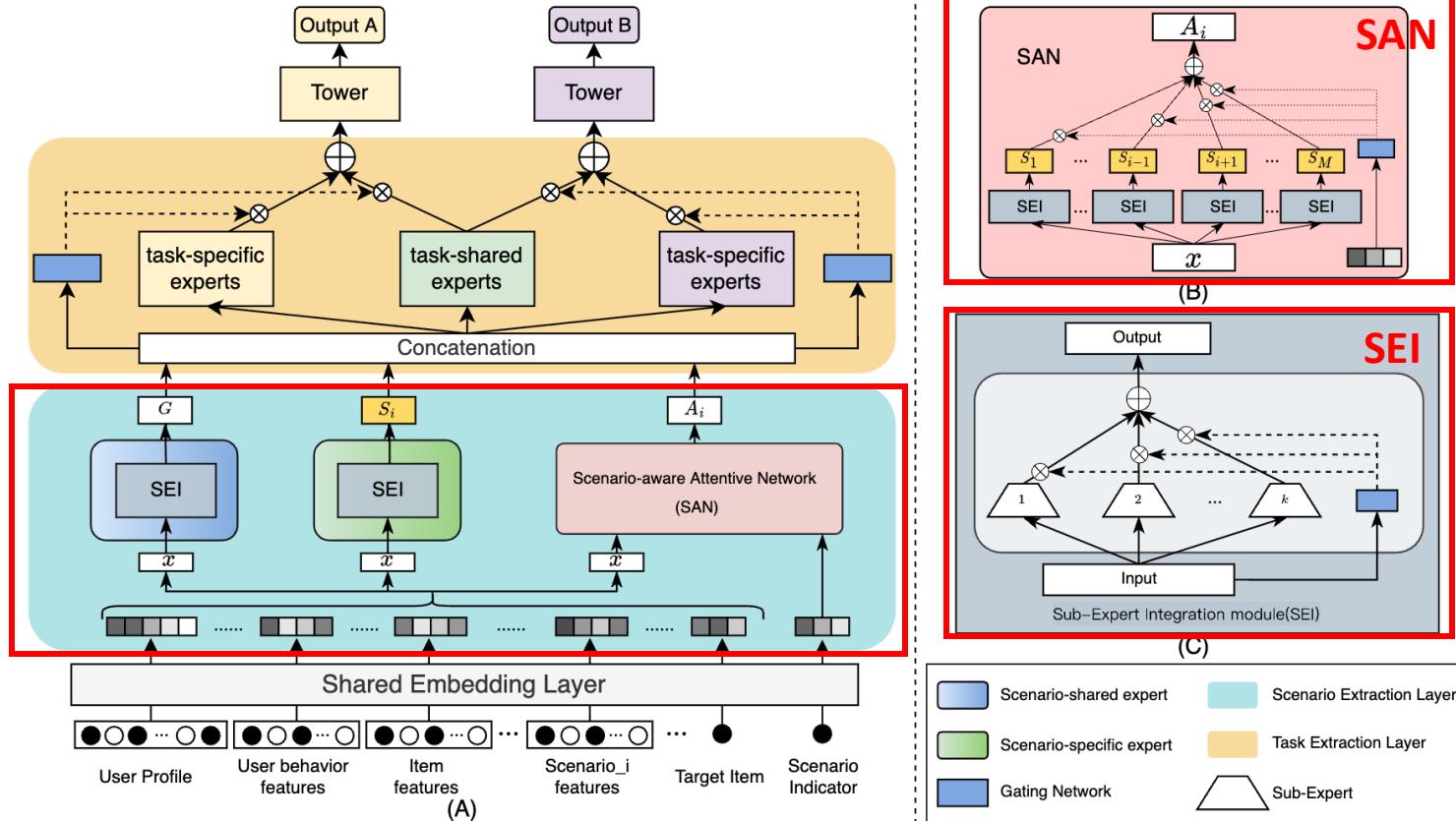
➤ Motivation

- Multi-scenario and multi-task (CTR, CTCVR) optimization

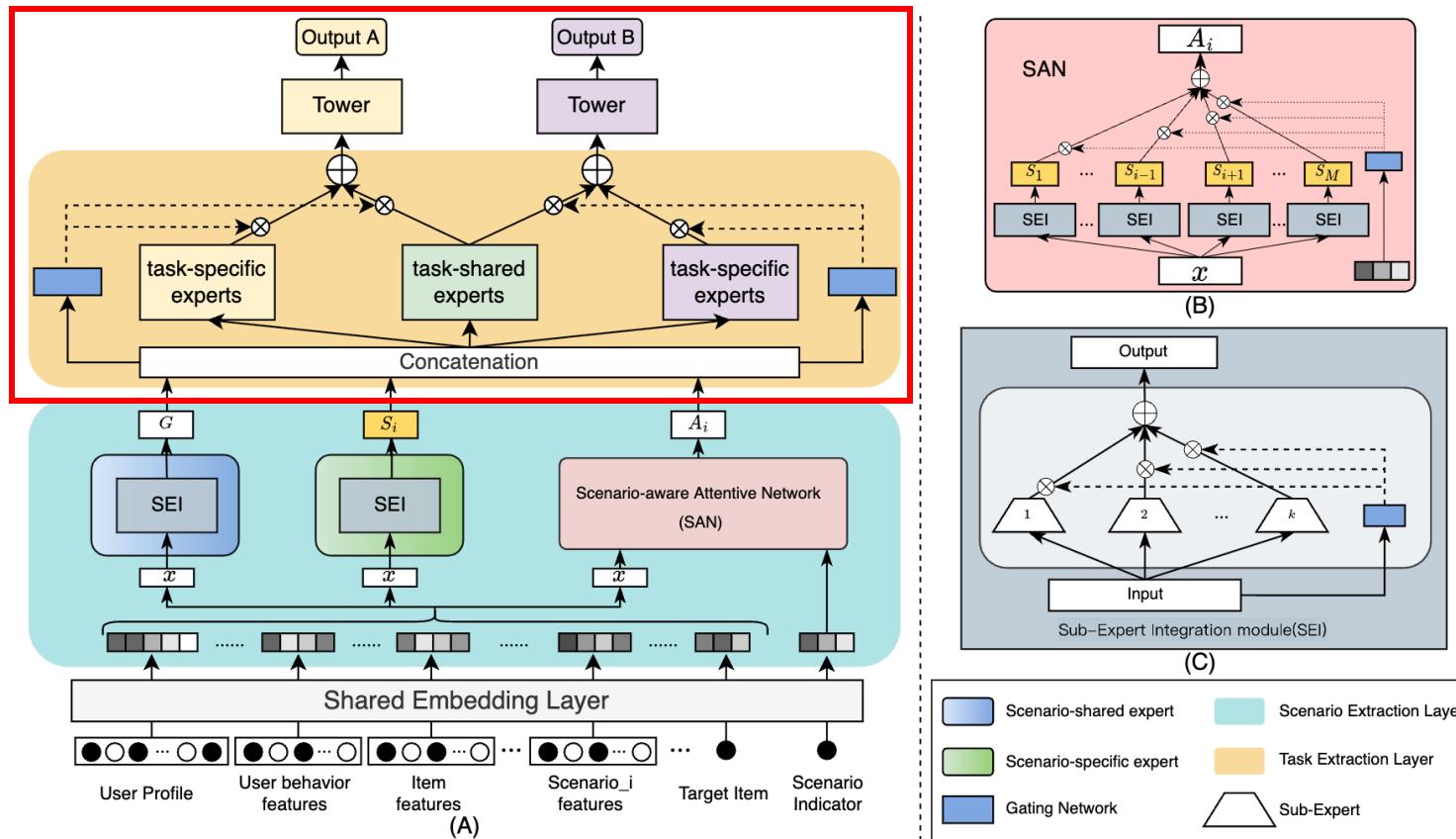
➤ Methods

- Proposing Hierarchical information extraction Network (HiNet) for multi-scenario & multi-task
- Scenario Extraction Layer: Sharing information among scenarios and extracting scenario-specific characteristic
- Task Extraction Layer: Resolving negative transfer problem in multi-task learning





- **Scenario Extraction Layer**
- Scenario-shared expert network (SEI)
 - Scenario-specific expert network (SEI)
 - Scenario-aware attentive network (SAN)



➤ Scenario Extraction Layer

- Scenario-shared expert network (SEI)
- Scenario-specific expert network (SEI)
- Scenario-aware attentive network (SAN)

➤ Task Extraction Layer

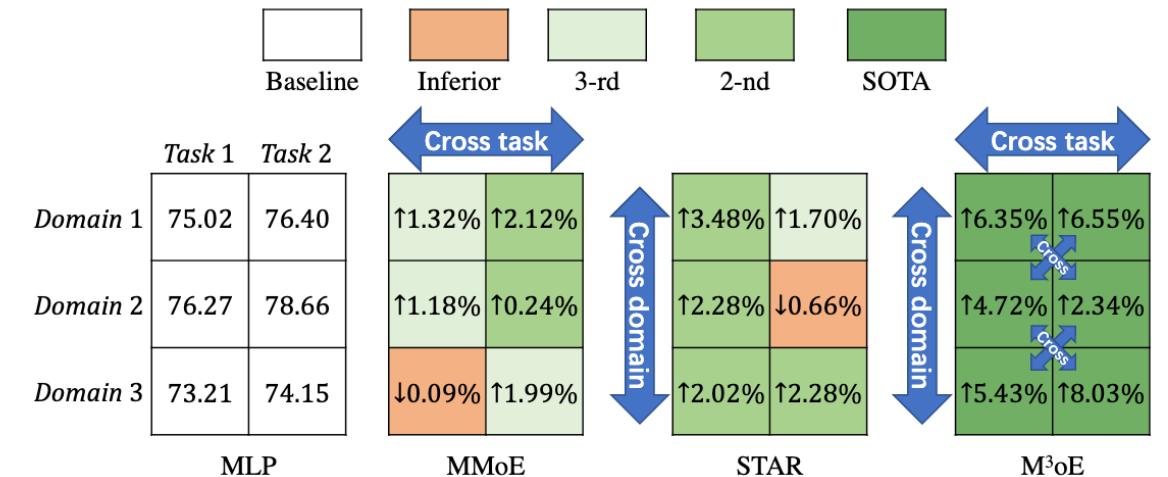
- Task-shared expert networks
- Task-specific expert networks

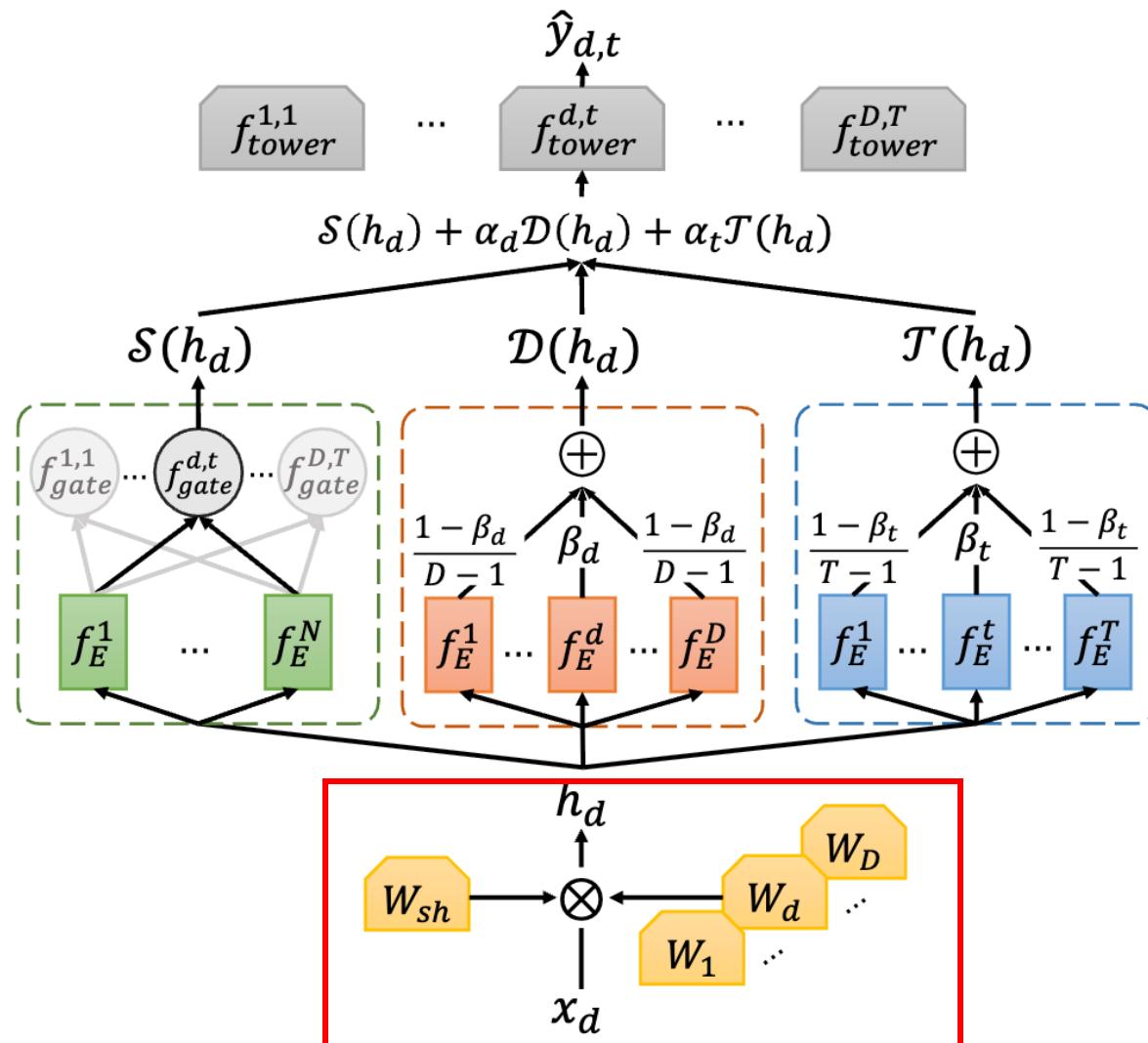
➤ Motivation

- Multi-Domain Mult-Task (MDMT) seesaw problem
- The same multi-domain information transfer method may not generalize to different tasks
- The same multi-task optimization balancing strategy may not generalize to different domains.

➤ Methods

- Domain representation extraction layer
- Multi-view expert learning layer
- MDMT objective prediction layer



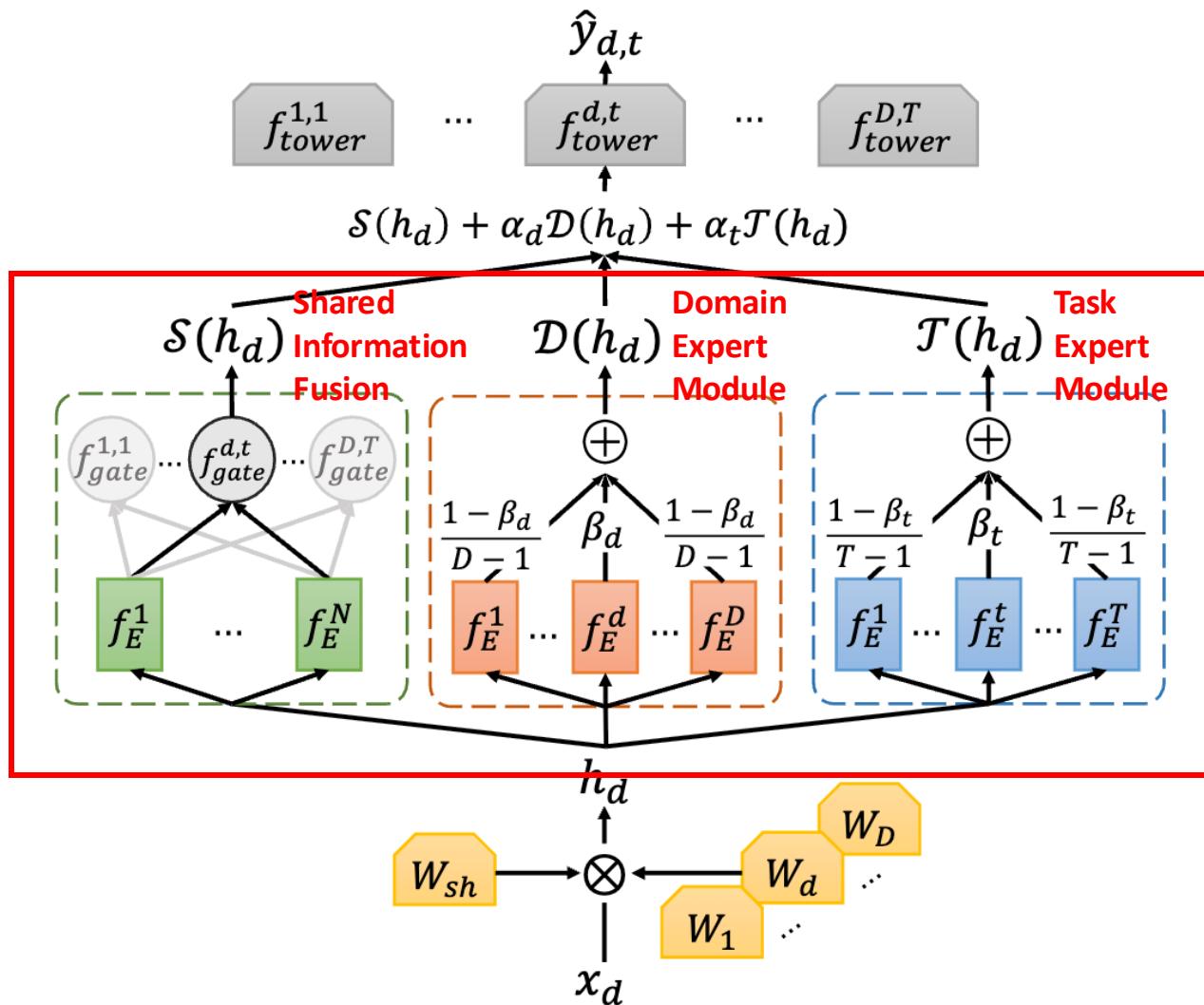


➤ Domain Representation Extraction Layer

$$\widehat{\mathbf{W}}_d = \mathbf{W}_d \otimes \mathbf{W}_{sh}$$

$$f_{DR}(x_d) = \widehat{\mathbf{W}}_d x_d + \mathbf{b}_d + \mathbf{b}_{sh}$$

$$\mathbf{h}_d = \mathbf{W}_c f_{DR}(x_d) + \mathbf{b}_c + f_{DA}(x_d)$$



➤ Domain Representation Extraction Layer

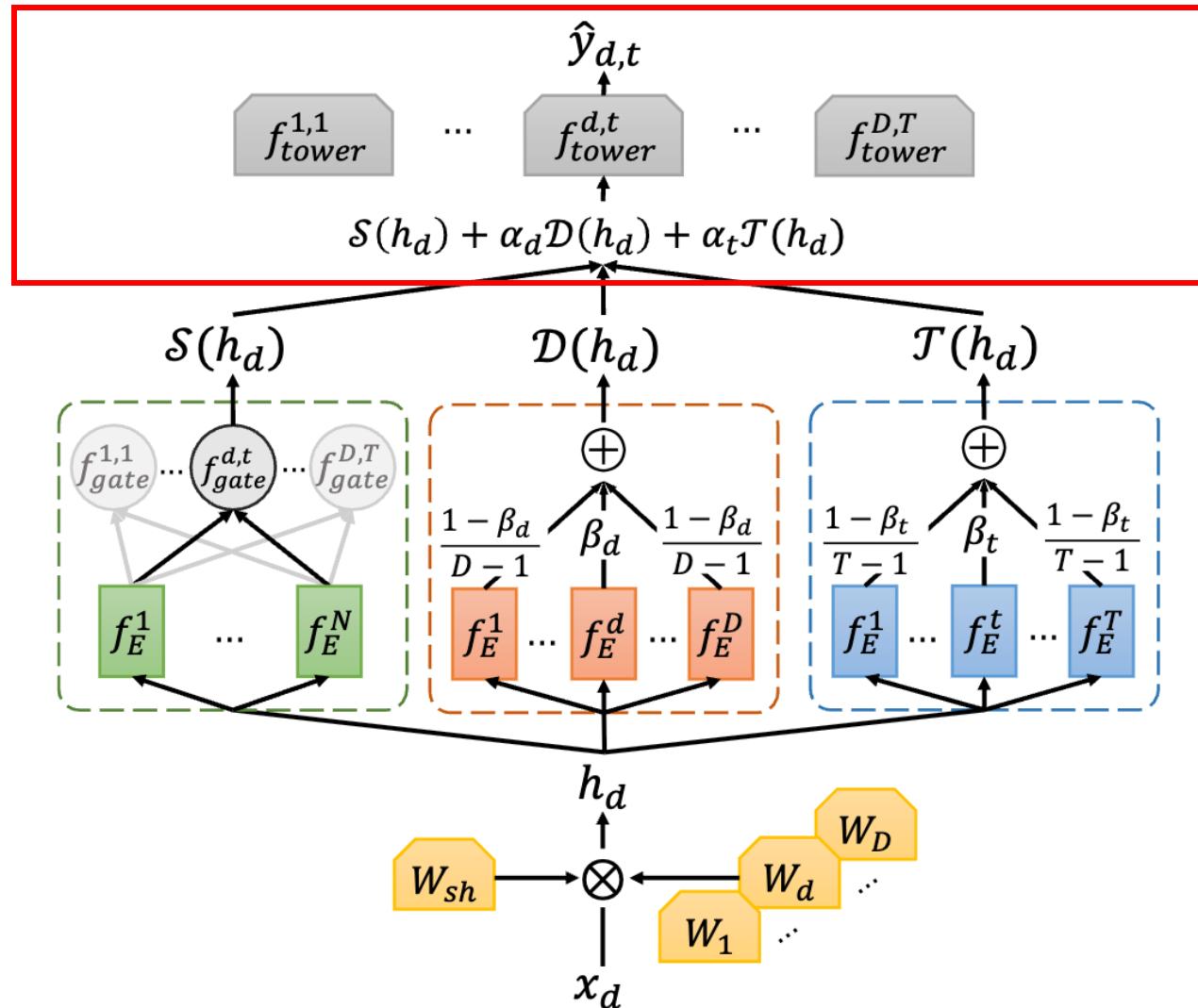
$$\widehat{\mathbf{W}}_d = \mathbf{W}_d \otimes \mathbf{W}_{sh}$$

$$f_{DR}(x_d) = \widehat{\mathbf{W}}_d x_d + \mathbf{b}_d + \mathbf{b}_{sh}$$

$$\mathbf{h}_d = \mathbf{W}_c f_{DR}(x_d) + \mathbf{b}_c + f_{DA}(x_d)$$

➤ Multi-View Expert Learning Layer

- Shared Information Fusion
- Domain Expert Module
- Task Expert Module



➤ Domain Representation Extraction Layer

$$\widehat{\mathbf{W}}_d = \mathbf{W}_d \otimes \mathbf{W}_{sh}$$

$$f_{\text{DR}}(\mathbf{x}_d) = \widehat{\mathbf{W}}_d \mathbf{x}_d + \mathbf{b}_d + \mathbf{b}_{sh}$$

$$\mathbf{h}_d = \mathbf{W}_c f_{\text{DR}}(\mathbf{x}_d) + \mathbf{b}_c + f_{\text{DA}}(\mathbf{x}_d)$$

➤ Multi-View Expert Learning Layer

- Shared Information Fusion
- Domain Expert Module
- Task Expert Module

➤ Multi-View Representation Balancing

$$\bar{h}_d = S(h_d) + \alpha_d \cdot T(h_d) + \alpha_t \cdot D(h_d)$$

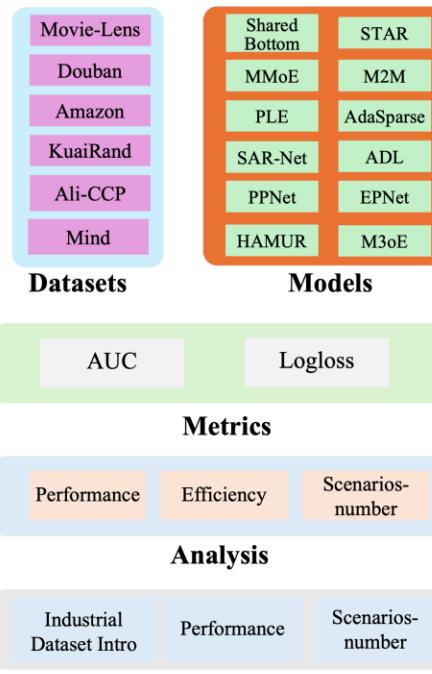
$$f_{\text{tower}}^{d,t}(\bar{h}_d) = W_{d,t}^2 \text{ReLU}(W_{d,t}^1 \bar{h}_d + b_{d,t}^1) + b_{d,t}^2$$

$$\hat{y}_{d,t} = \text{Sigmoid}(f_{\text{tower}}^{d,t}(\bar{h}_d))$$

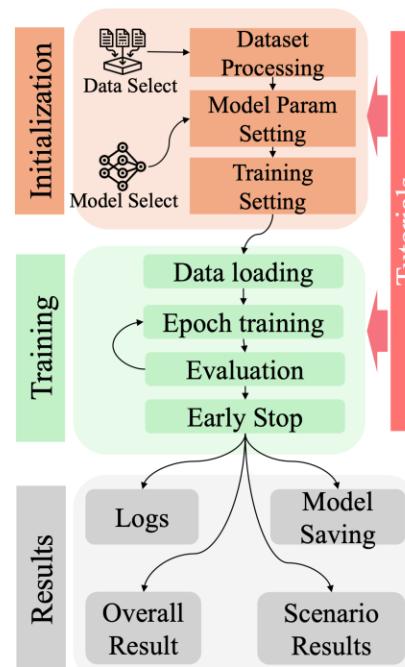


Scenario - Wise Rec

Benchmark for Multi-Scenario Recommendation



Benchmark Overview



Overall Pipeline



Paper Repo



GitHub Repo

➤ Highlight

- A comprehensive benchmark exclusively for MSR
- Complete data loading, training, and evaluation process
- Providing 6 datasets and 12 MSR models
- Comprepresentative analysis and step-by-step tutorial

➤ Multi-Scenario Recommendation

Model	Setting	Methods	Model	Setting	Methods
STAR	Multi-Scenario	Shared-Specific	LLM4MSR	Multi-Scenario	Dynamic Weight; LLMs
SAR-Net	Multi-Scenario	Shared-Specific; Experts	AESM2	Multi-Scenario & Multi-Task	Experts
ADI	Multi-Scenario	Shared-Specific	PEPNet	Multi-Scenario & Multi-Task	Dynamic Weight
Uni-CTR	Multi-Scenario	Shared-Specific; LLMs	M2M	Multi-Scenario & Multi-Task	Dynamic Weight; Experts
M-LoRA	Multi-Scenario	Shared-Specific; LoRAs	HiNet	Multi-Scenario & Multi-Task	Experts
HAMUR	Multi-Scenario	Dynamic Weight	M3oE	Multi-Scenario & Multi-Task	Experts
HierRec	Multi-Scenario	Dynamic Weight	Scenario-Wise Rec	Multi-Scenario	Benchmark

➤ Multi-Scenario Recommendation

Topic	Challenge & future direction
LLM-based multi-scenario & multi-task modeling	<ul style="list-style-type: none">Explore quantification or compression techniques for handling large-scale scenarios.More fine-grained modeling to bridge semantic gaps between LLM and MSR models.
Robustness	<ul style="list-style-type: none">Scenarios with different available information (multimodal ...)
Privacy	<ul style="list-style-type: none">Data need to be shared between different scenarios to build a unified model. Methods to protect user privacy should be proposed.
Fairness and Bias	<ul style="list-style-type: none">The issue of fairness in recommendation scenarios.

Coffee Break



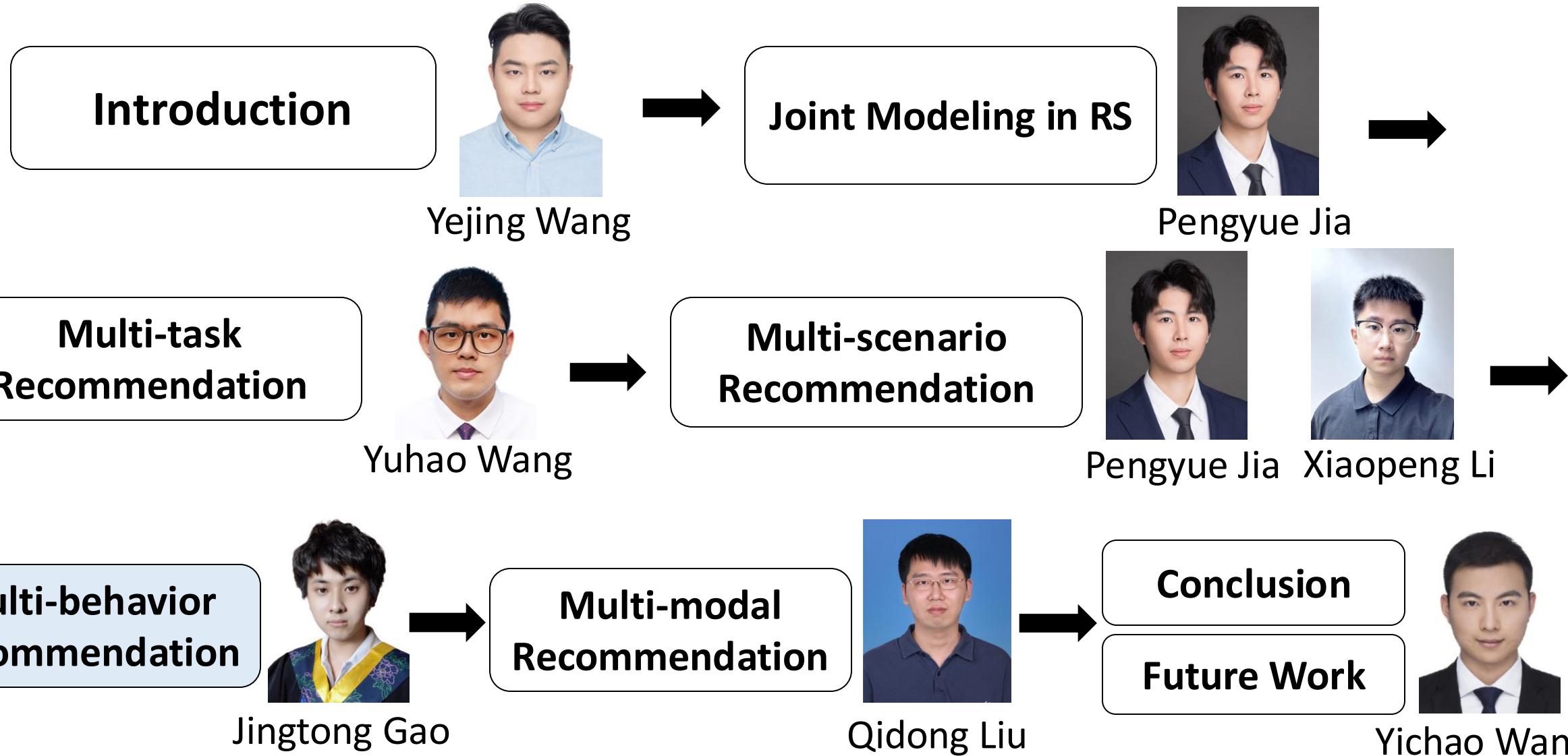
Huawei Noah's Ark Lab



WWW25 Huawei Noah's Ark
Lab Chat Group



AML Lab
CityU



Multi-Behavior Modeling



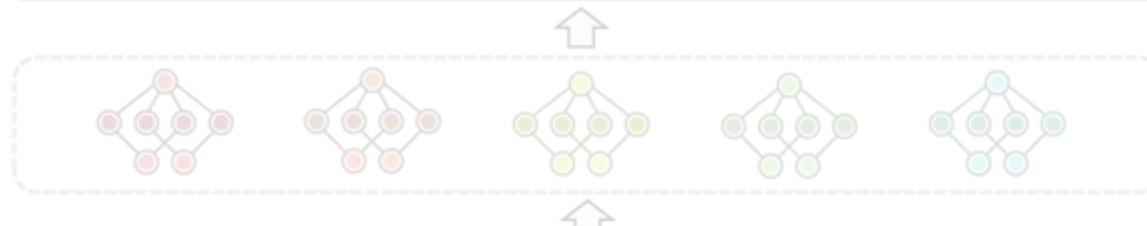
Multi-Scenario



Multi-Task



Task/scenario adaption



Representation extraction



Multi-Behavior



Multi-Modal

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

Joint Modeling

$$E^{Merge} = U(E, E^B, E^M)$$

Multi-behavior

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

Multi-modal

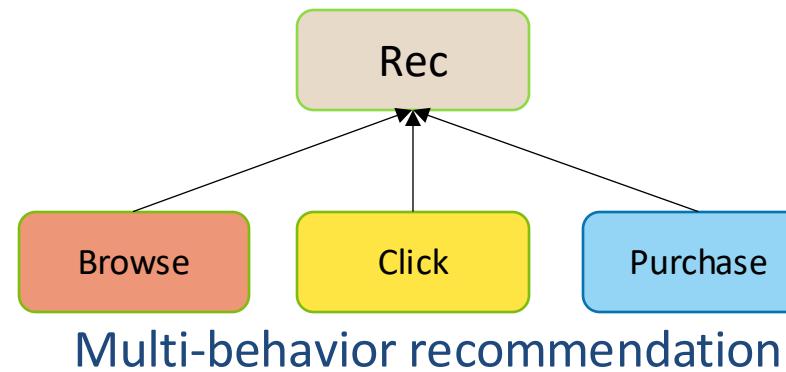
$$wL(E^{Merge}, \theta^{sh}, \theta^t, \theta^s)$$

Multi-scenario

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

Multi-task

- 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



Behavior Type Definition



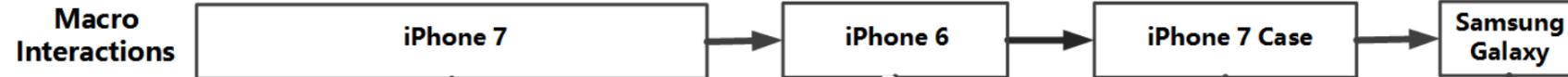
- An open question
- Roughly three categories:
 - Macro behaviors: interaction with different items
E.g. user 1 interact with item 1, then item 22, then item 81.
 - Micro behaviors: actions taken on this item
E.g. click, add to cart,...
 - Behaviors from different domains or scenarios
E.g. Same behavior in two domains => different behaviors (highlight the distinctions)



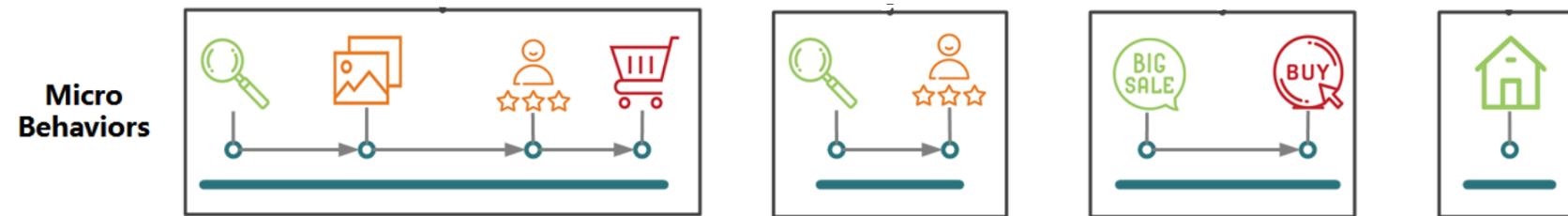
Behavior Type Definition



➤ Macro behaviors:

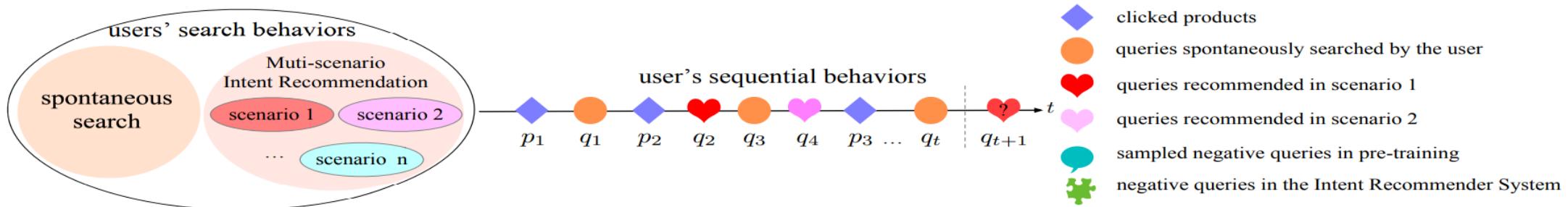


➤ Micro behaviors:

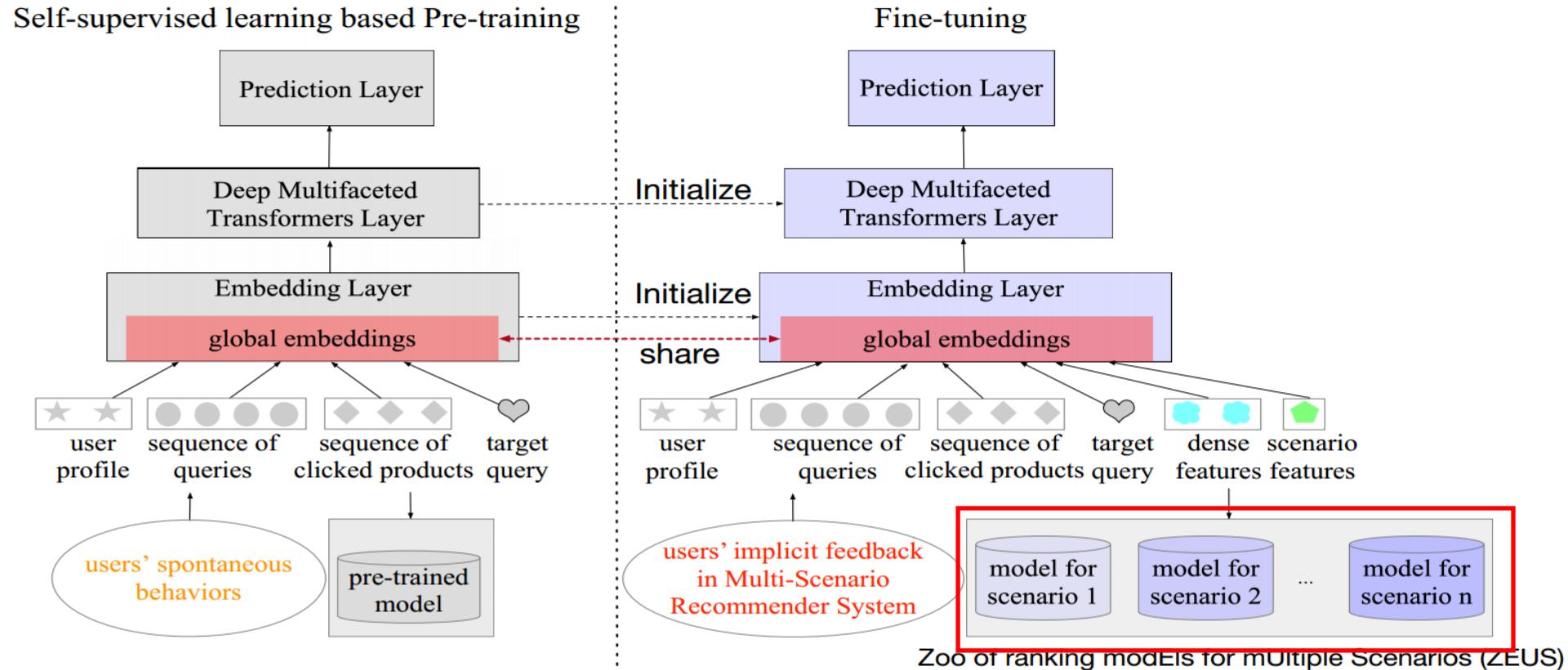


➤ Behaviors from different domains or scenarios

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



➤ Modeling the complicated cross-scenario behavior dependencies

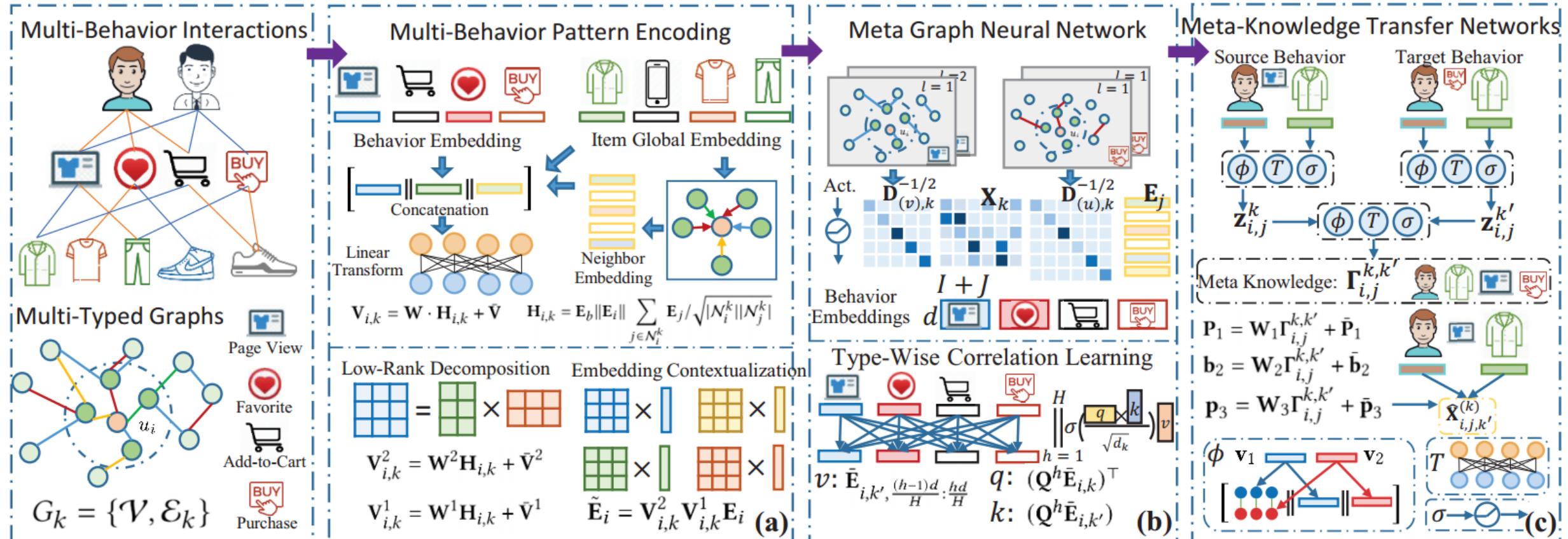


Example: pre-training and fine-tuning of ZEUS

Multi-Behavior Fusion



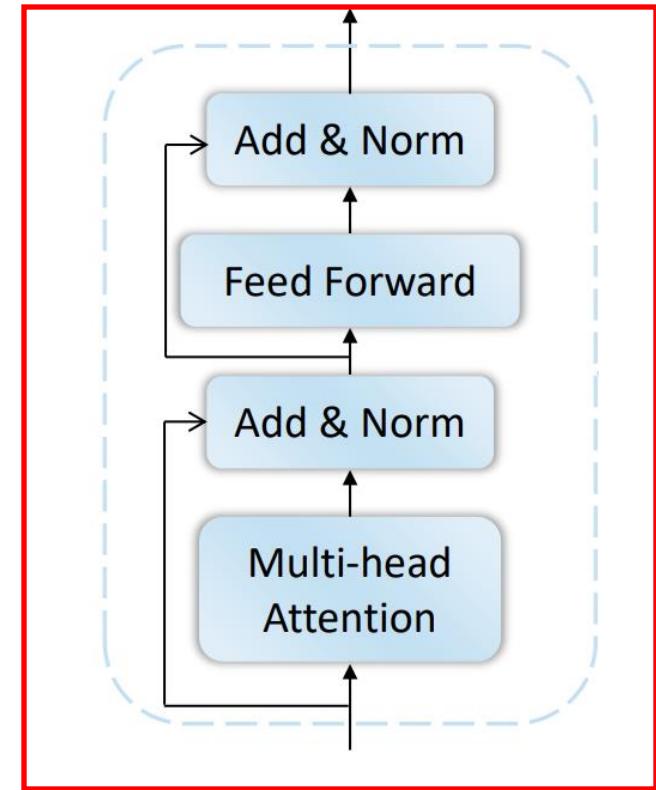
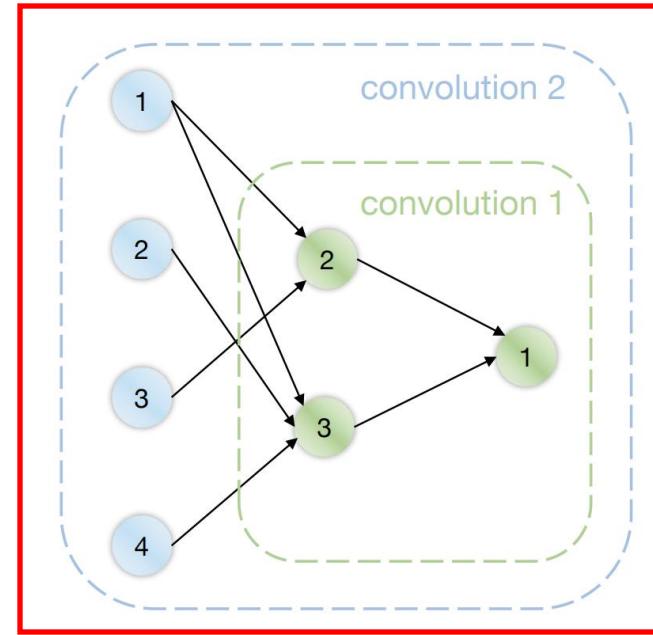
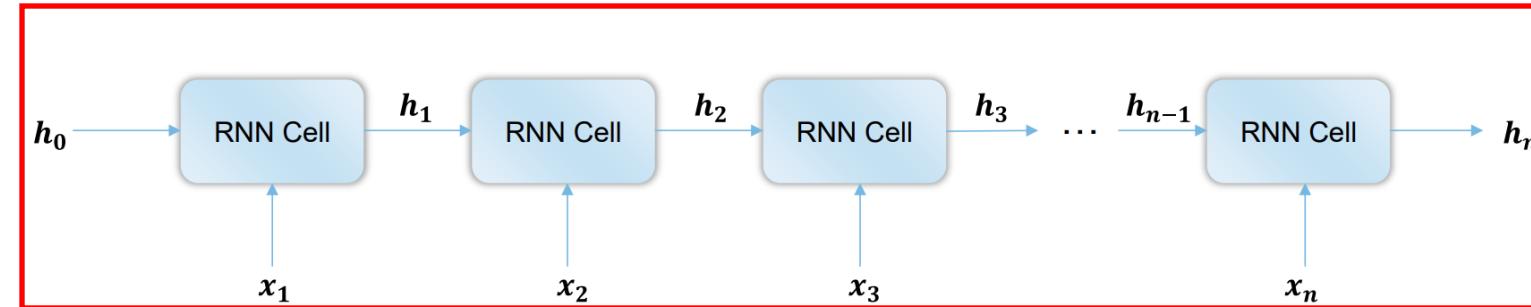
➤ Modeling the complicated cross-type behavior dependencies



Example: MB-GMN

- Sequence modeling of heterogeneous behavioral feedback
 - How to model different behaviors and their feedbacks
- Modeling behavior relations
 - How to capture complicated behavior relations
- Joint long-term and short-term preference modeling with heterogeneous behaviors of users
 - How to combining long-term preference and short-term preferences for better user modeling
- Avoiding noise and bias
 - How to solve the problem brought by noises and bias coming with different behaviors

- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods



- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

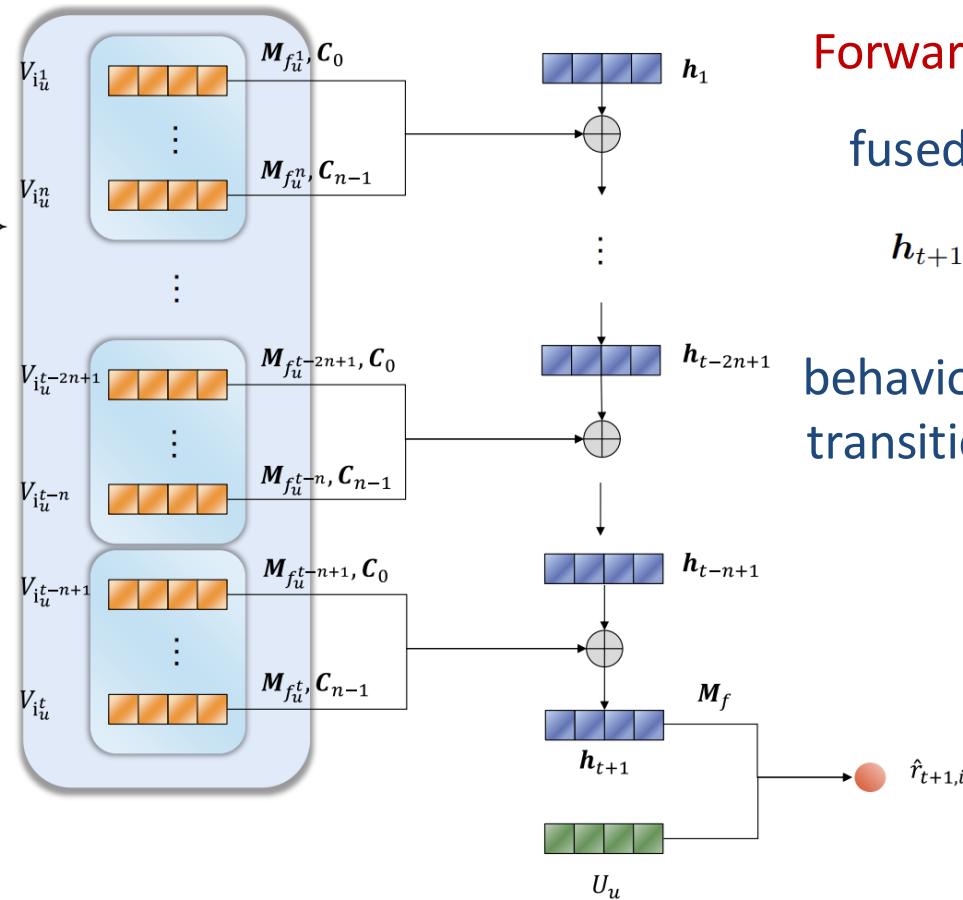
Works	Data Perspective	Model Perspective	Features
RLBL [14]	A sequence of (item, behavior) pairs	Local	Capture the influence of heterogeneous behaviors by utilizing a behavior transition matrix.
RIB [26]	A sequence of (item, behavior) pairs	Local	Leverage GRU and attention mechanism simultaneously.
BINN [22]	A sequence of (item, behavior) pairs	Local	Design the CLSTM and the Bi-CLSTM, where the behavior vector is as context in LSTM.
CBS [63]	Some behavior-specific subsequences of items	Local	Design of models with and without shared parameters for behaviors simultaneously; towards the next-basket recommendation.
DIPN [64]	Some behavior-specific subsequences of items	Local	Leverage GRU and attention mechanism simultaneously; behaviors are specific, including swipe, touch and browse interactive behavior.
HUP [27]	A sequence of (item, behavior) pairs	Local	Design the Behavior-LSTM where adds behavior gate and time gate to the LSTM; leverage attention mechanism; take into account the category of the items.
IARS [28]	A sequence of (item, behavior) pairs	Local	Propose Soft-MGRU (a multi-behavior gated recurrent unit) with sharing parameters between behaviors; leverage attention mechanism; take into account the category of the items.
DeepRec [62]	Some behavior-specific subsequences of items	Local + Global	Utilizing multi-behavior sequence data to make privacy-preserving recommendation.
MBN [65]	Some behavior-specific subsequences of items	Local	The overall Meta-RNN and the separate Behavior-RNN share the learned potential representations by gathering and then scattering; towards the next-basket recommendation.

RNN-based Methods—RLBL



➤ Conducting item side behavior modeling via recurrent log-bilinear model

windowed representations
Item-behavior pairs
 $\{(i_u^{t-n+1}, \hat{f}_u^{t-n+1}), \dots, (i_u^t, f_u^t)\}$



Forwarding steps:

$$h_{t+1} = W_{RLBL} h_{t-n+1} + \sum_{i=0}^{n-1} C_i M_{f_u^{t-i}} V_{i_u^{t-i}}$$

behavior-specific position transition
transition matrix embedding

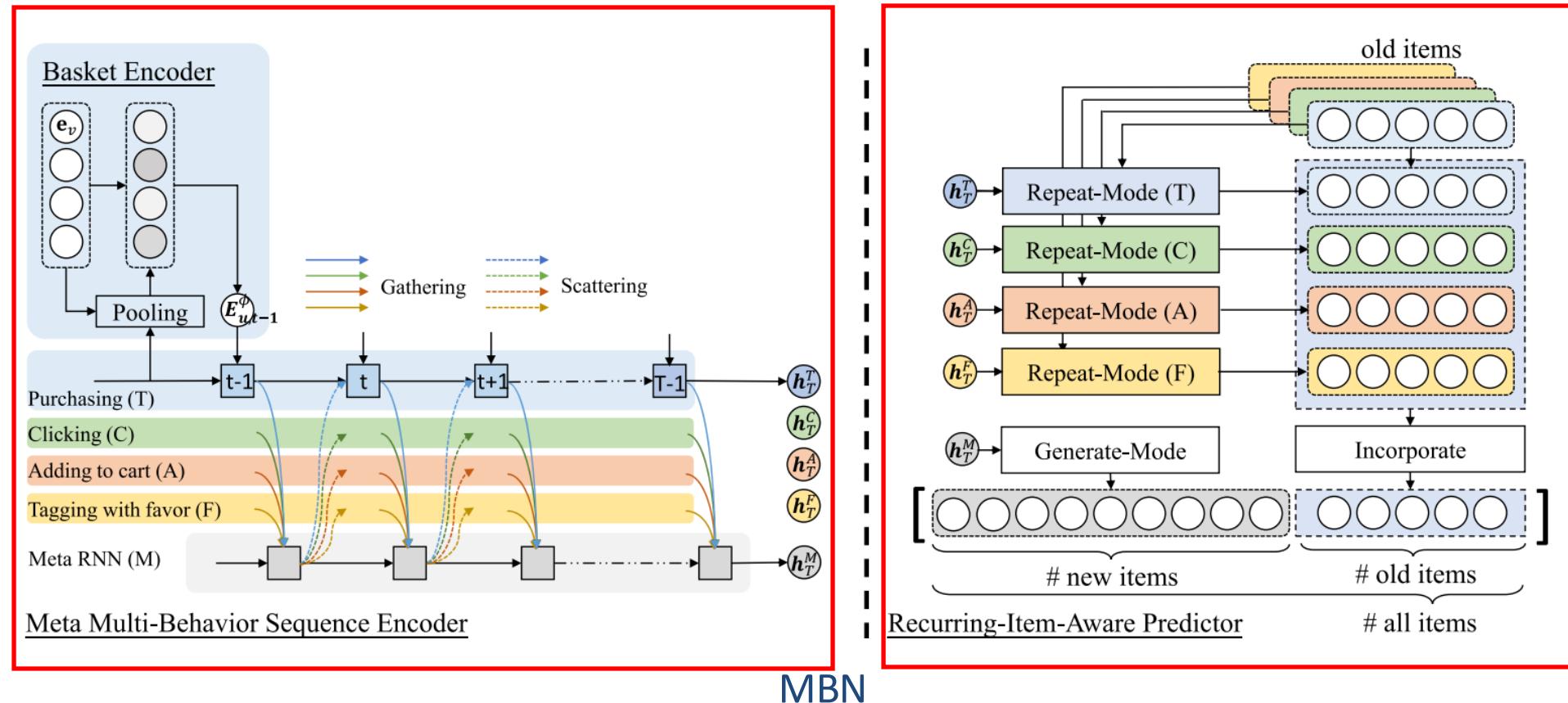
RLBL

RNN-based Methods—MBN



➤ Conducting Next basket recommendation with multi-behavior modeling

- Encoder: Multiple Behavior-RNN and one Meta-RNN for behavior modeling and fusion
- Predictor: Generating next items with both new items and old items



Taxonomy



- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

Methods	Prons	Cons
RNN-based	<ul style="list-style-type: none">• Suitable for sequence problems and can store short-term memories	<ul style="list-style-type: none">• Gradient disappearance & explosion problems• Inefficient in predicting future sequences• Rarely used currently
Graph-based		
Transformer-based		
Others		

- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

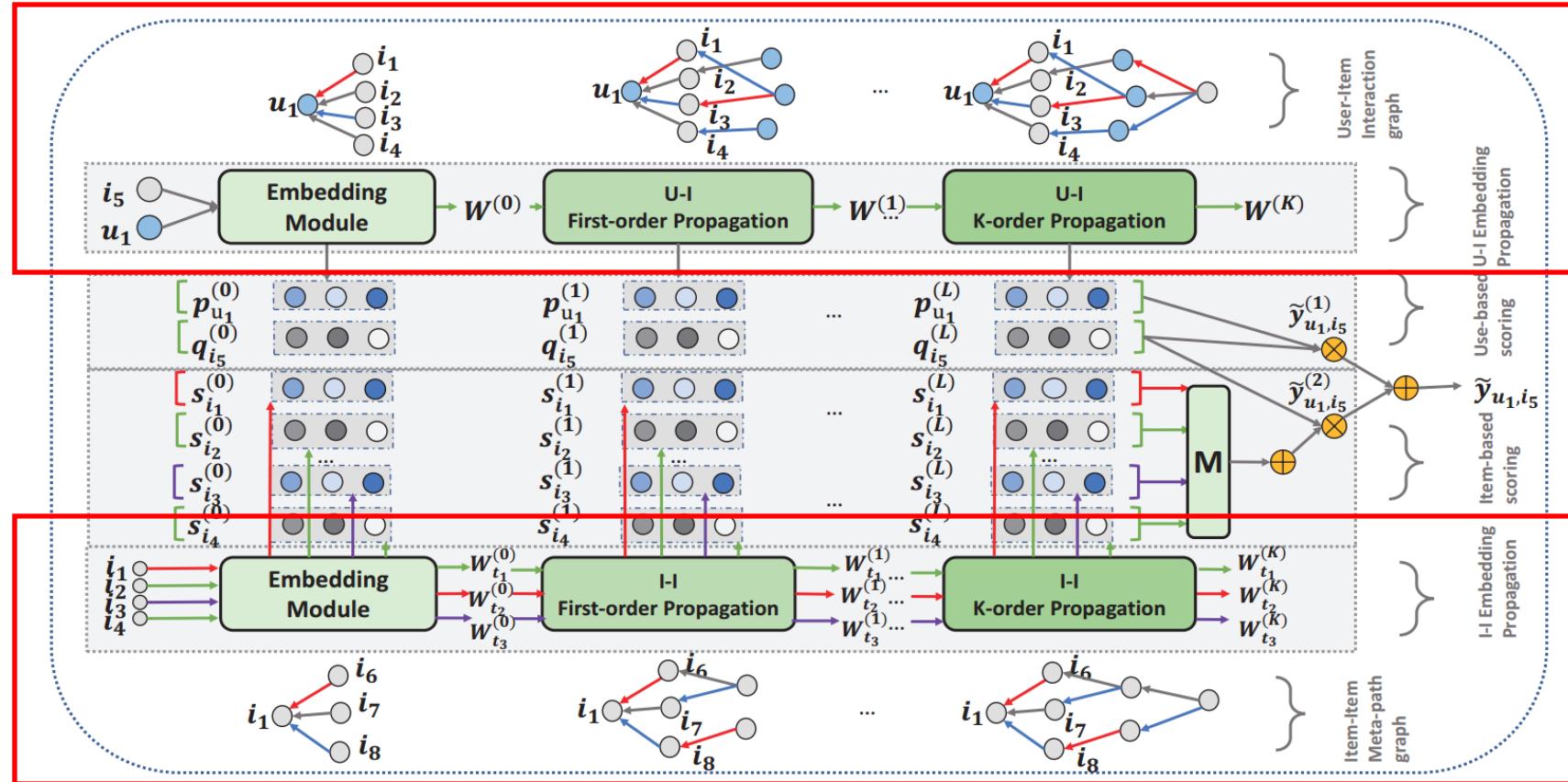
Works	Data Perspective	Model Perspective	Features
MGNN-SPred [24]	Some behavior-specific subsequences of items	Global	Modeling behavior from behavior transition relations, containing homogeneous behavior transitions intra each kind of behavior-specific subsequences.
DMBGN [71]	Some behavior-specific subsequences of items	Global	Focus on the task of voucher redemption rate prediction and model the relationship between multiple behaviors and vouchers effectively.
GPG4HSR [72]	A sequence of (item, behavior) pairs	Local + Global	Learn various behavior transition relations from the global graph and the personalized graph, respectively.
BGNN [73]	Some behavior-specific subsequences of items	Global	Construct directed graphs for different behavior transition (homogeneous and heterogeneous) information.
BA-GNN [74]	Some behavior-specific subsequence of items	Global	Construct directed graphs for different behavior-specific sequences respectively.

Graph-based Methods—MB-GCN



➤ Conducting behavior-aware user-item propagation and item-relevance aware item-item propagation in the user-item graph

U-I Embedding Propagation

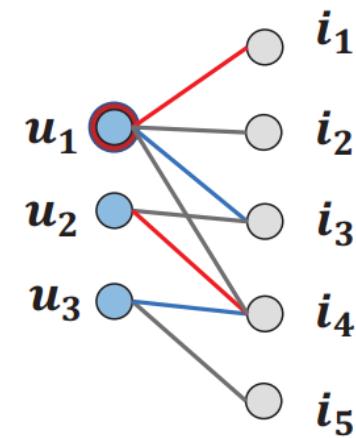


I-I Embedding Propagation

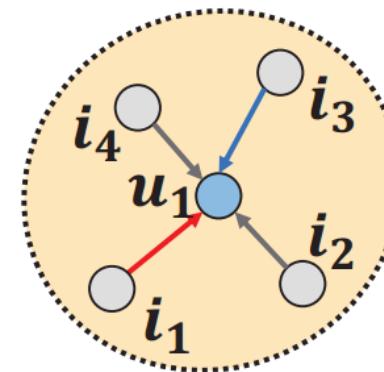
- Existing researches approach this task from two aspects
 - Utilizing multi-behavior data into the sampling process and builds multi-sampling pairs to reinforce the model learning process
 - Tryng to design model to capture multi-behavior information
- Their limitations
 - The strength of multiple types of behaviors is not sufficiently utilized
 - The semantics of multiple types of behaviors are not considered
- Why?
 - The limitations of existing methods lie in the fact that they cannot thoroughly address the above two challenges: modeling user-to-item based strength and item-to-item based semantics of multiple types of behaviors.

➤ Solution

- Constructing a unified heterogeneous graph based on multiple types of behavioral data
- User/item represented as nodes and different types of behaviors represented as multiple types of edges of the graph



(a) U-I Interaction Graph

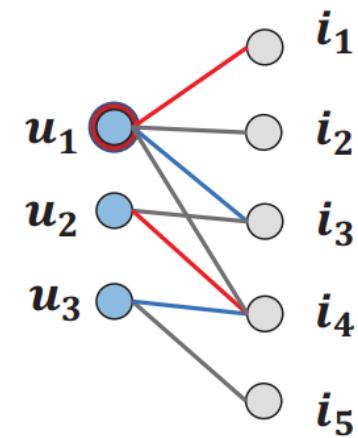


(b) Local Graph of u_1

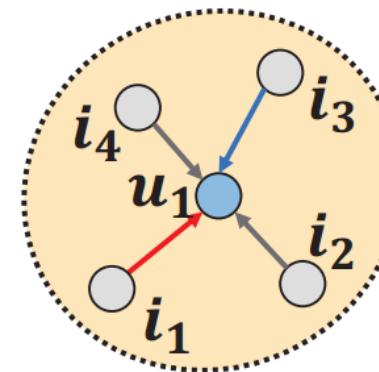
An illustration of the user-item multi-behavior graph

➤ Framework design

- Input: The user-item interaction data of T types of behaviors, $\{Y^1, Y^2, \dots, Y^T\}$
- Output: A recommendation model that estimates the probability that a user u will interact with an item i under the T -th behavior, i.e. target behavior.



(a) U-I Interaction Graph



(b) Local Graph of u_1

An illustration of the user-item multi-behavior graph

➤ User behavior propagation weight calculation

- Different behavior contributes differently to the target behavior
- The importance of each behavior to the target behavior cannot be measured artificially and should be learned by the model itself
- A frequency-based propagation weight

$$\alpha_{ut} = \frac{w_t \cdot n_{ut}}{\sum_{m \in N_r} w_m \cdot n_{um}}$$

Behavior-wised importance weight

Behavior types

User behavior count

User Embedding Propagation



- Neighbour item aggregation based on behavior
 - Items that are interacted under the same behavior reflect user's similar preference strength
 - Items that have the same behavior interaction with user are aggregated together so as to obtain one embedding for each behavior
 - Aggregation function: mean function, mean function with sampling, max pooling, etc.

$$p_{u,t}^{(l)} = \text{aggregate}(q_i^{(l)} | i \in N_t^I(u))$$

Item embedding Items the user interacted under behavior t

➤ Behavior-level Item Propagation for User

- Summing neighbor item aggregation embedding together according to weight
- Going through an encoder matrix to obtain the final neighbor item aggregation for users
- A graph neural network to refine information based on multi-behavior

$$p_u^{(l+1)} = W^{(l)} \cdot \left(\sum_{t \in N_r} \alpha_{ut} p_{u,t}^{(l)} \right)$$

Learned param User embedding at layer l

Propagation weight

Item Embedding Propagation

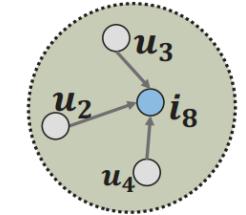
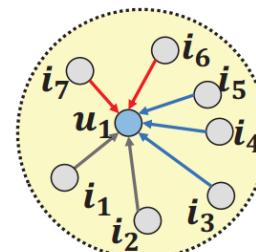


- The feature of the item is static
 - No importance needed. Assuming that different user has the same contribution to item

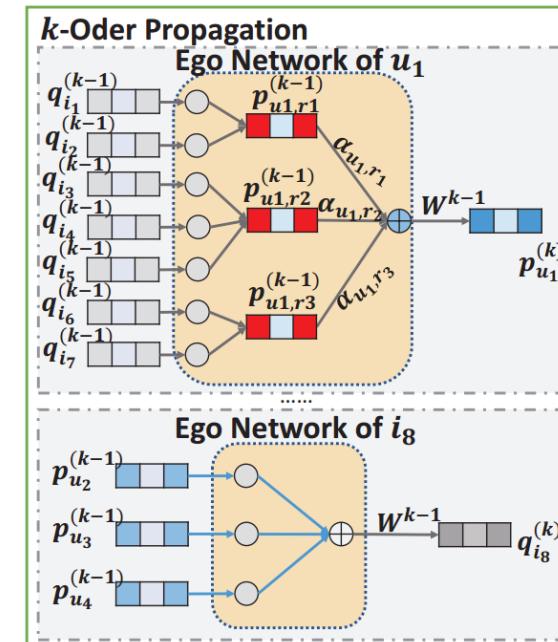
$$q_i^{(l+1)} = W^{(l)} \cdot \text{aggregate}(p_j^{(l)} | j \in N^U(i)),$$

➤ Summary: Behavior-aware User-Item Propagation

- User Embedding Propagation
- Item Embedding Propagation



(a) Local Graph



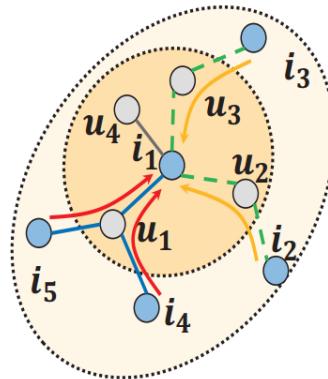
(b) k-order Propagation Process

Item-Relevance Aware I-I Propagation

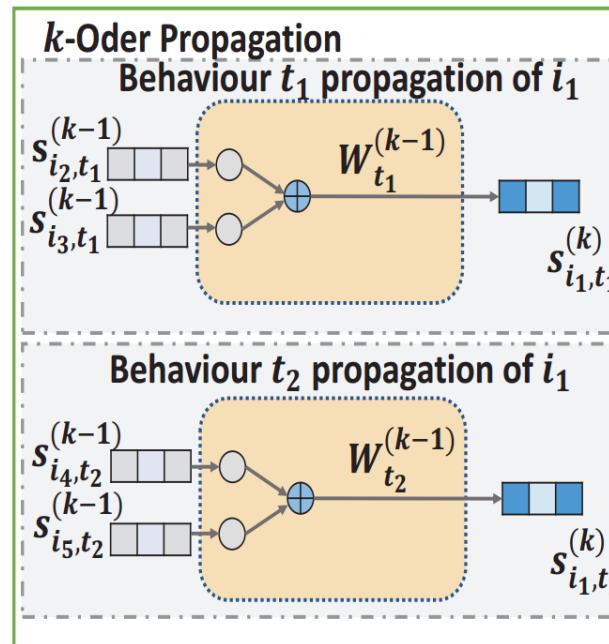


➤ Item information extracting

$$s_{it}^{(l+1)} = W_t^{(l)} \cdot \text{aggregate}(s_{jt}^{(l)} | j \in N_t^I(i)) \quad s_{it}^{(0)} = q_i^{(0)}$$



(a) Local Graph



(b) k-order Propagation Process

➤ Embedding aggregation

$$\begin{aligned} p_u^* &= p_u^{(0)} || \dots || p_u^{(L)}, \\ q_i^* &= q_i^{(0)} || \dots || q_i^{(L)}, \\ s_{it}^* &= s_{it}^{(0)} || \dots || s_{it}^{(L)}, t \in N_r \end{aligned}$$

➤ User-based CF scoring

$$y_1(u, i) = p_u^{*T} \cdot q_i^*$$

➤ Item-based CF scoring

$$y_2(u, i) = \sum_{t \in N_r} \sum_{j \in N_t^I(u)} \frac{s_{jt}^{*T} \cdot M_t \cdot s_{it}^*}{|N_t^I(u)|}$$

➤ Combined Scoring

$$y(u, i) = \lambda \cdot y_1(u, i) + (1 - \lambda) \cdot y_2(u, i)$$

Overall performance



	Method	Recall@10	NDCG@10	Recall@20	NDCG@20	Recall@40	NDCG@40	Recall@80	NDCG@80
One-behavior	MF-BPR	0.02331	0.01306	0.03161	0.01521	0.04239	0.01744	0.05977	0.02049
	NCF	0.02507	0.01472	0.03319	0.01683	0.04502	0.01931	0.06352	0.02252
	GraphSAGE-OB	0.01993	0.01157	0.02521	0.01296	0.03368	0.01474	0.04617	0.01693
	NGCF-OB	0.02608	0.01549	0.03409	0.01757	0.04612	0.02010	0.06415	0.02324
Multi-behavior	MCBPR	0.02299	0.01344	0.03178	0.01558	0.04360	0.01813	0.06190	0.02132
	NMTR	0.02732	0.01445	0.04130	0.01831	0.06391	0.02279	0.09920	0.02891
	GraphSAGE-MB	0.02094	0.01223	0.02805	0.01406	0.03804	0.01616	0.05351	0.01887
	NGCF-MB	0.03076	0.01754	0.04196	0.02042	0.05857	0.02389	0.08408	0.02833
	RGCN	0.01814	0.00955	0.02627	0.01165	0.03877	0.01426	0.05749	0.01750
	MBGCN	0.04006	0.02088	0.05797	0.02548	0.08348	0.03079	0.12091	0.03730
Improvement		30.23%	19.04%	37.04%	24.78%	24.91%	28.88%	8.90%	26.40%

Comparison on Tmall

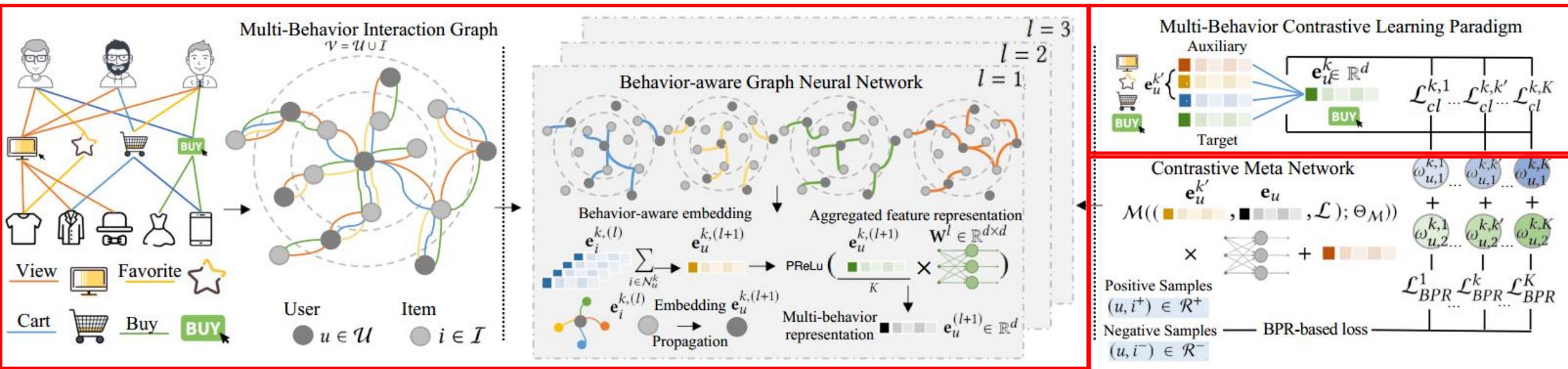
	Method	Recall@10	NDCG@10	Recall@20	NDCG@20	Recall@40	NDCG@40	Recall@80	NDCG@80
One-behavior	MF-BPR	0.03873	0.02286	0.05517	0.02676	0.08984	0.03388	0.14137	0.04258
	NCF	0.04209	0.02394	0.05609	0.02579	0.09118	0.03410	0.15426	0.04022
	GraphSAGE-OB	0.034536	0.01728	0.06907	0.02594	0.11567	0.03547	0.18626	0.04747
	NGCF-OB	0.04112	0.02199	0.06336	0.02755	0.11051	0.03712	0.19524	0.05153
Multi-behavior	MCBPR	0.03914	0.02264	0.04950	0.02525	0.09592	0.03467	0.15422	0.04462
	NMTR	0.03628	0.01901	0.06239	0.02559	0.10683	0.03461	0.18907	0.04855
	GraphSAGE-MB	0.04204	0.02267	0.05862	0.02679	0.09707	0.03451	0.18272	0.04911
	NGCF-MB	0.04241	0.02415	0.06152	0.02893	0.10370	0.03741	0.01771	0.04987
	RGCN	0.04204	0.02051	0.06354	0.02591	0.09859	0.03309	0.16121	0.04363
	MBGCN	0.04825	0.02446	0.07354	0.03077	0.11926	0.04005	0.20201	0.05409
Improvement		13.77%	1.28%	11.76%	3.85%	7.68%	3.30%	6.58%	3.84%

Comparison on Beibei

Graph-based Methods—CML



➤ Conducting contrastive learning among behaviors



$$\mathbf{e}_u^{k,(l+1)} = \sum_{i \in N_u^k} \mathbf{e}_i^{k,(l)}; \quad \mathbf{e}_i^{k,(l+1)} = \sum_{u \in N_i^k} \mathbf{e}_u^{k,(l)}$$

$$\mathcal{L}_{cl}^{k,k'} = \sum_{u \in \mathcal{U}} -\log \frac{\exp(\varphi(\mathbf{e}_u^k, \mathbf{e}_u^{k'})/\tau)}{\sum_{u' \in \mathcal{U}} \exp(\varphi(\mathbf{e}_u^k, \mathbf{e}_{u'}^{k'})/\tau)}$$

Taxonomy



- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

Methods	Prons	Cons
RNN-based	<ul style="list-style-type: none">• Suitable for sequence problems and can store short-term memories	<ul style="list-style-type: none">• Gradient disappearance & explosion problems• Inefficient in predicting future sequences• Rarely used currently
Graph-based	<ul style="list-style-type: none">• Detailed modeling for behavior relations• Improved performance	<ul style="list-style-type: none">• Suffering from low efficiency
Transformer-based		
Others		

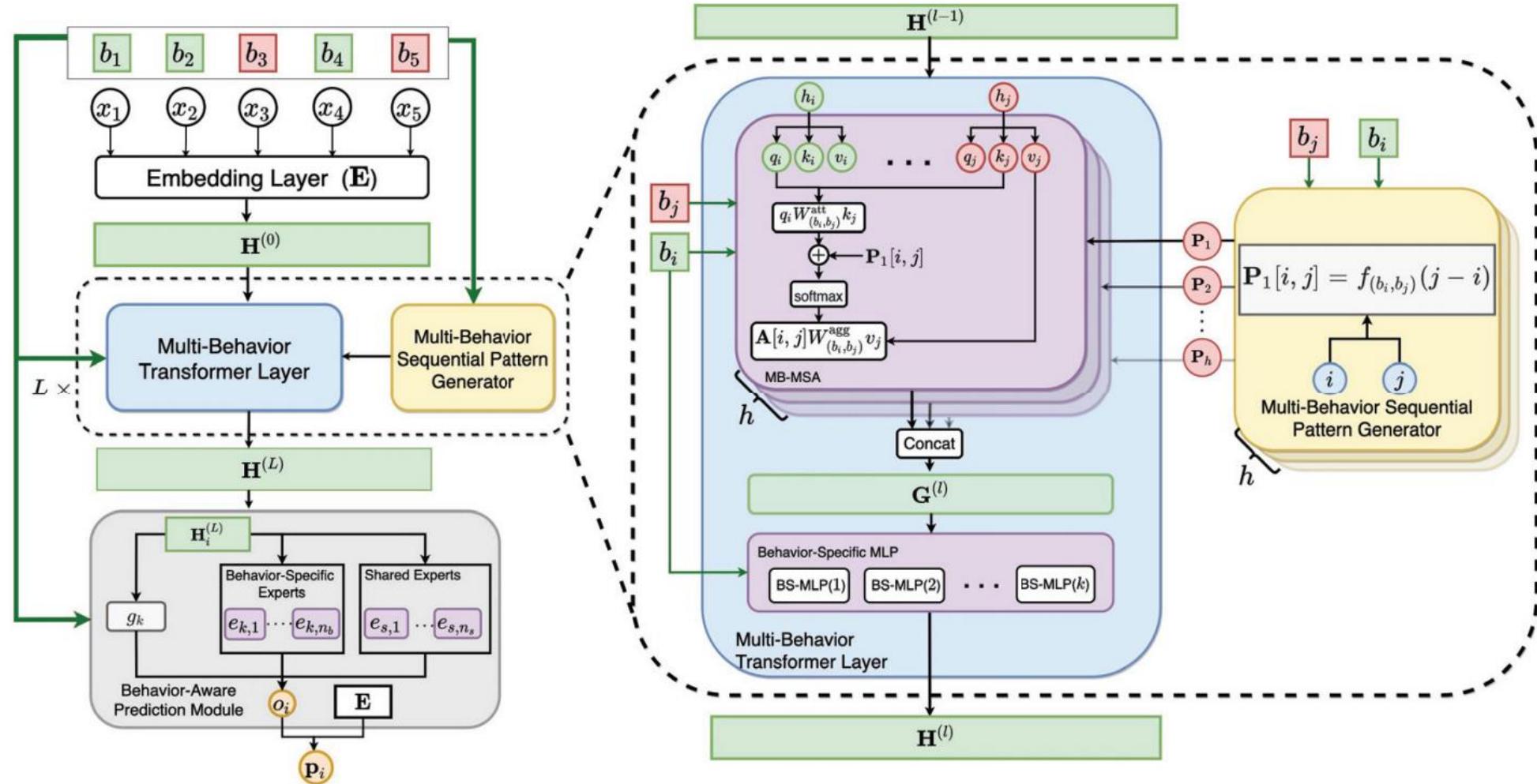
- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

Works	Data Perspective	Model Perspective	Features
DMT [23]	Some behavior-specific subsequences of items	Local + Global	Use target item as query; Consider implicit feedback bias by a bias deep neural network.
DFN [84]	Some behavior-specific subsequences of items	Local + Global	Use target item as query; Consider implicit negative feedback noise by an attention network .
DUMN [88]	Some behavior-specific subsequences of items	Local	Consider implicit feedback noise; Use memory network to obtain the long-term user preference.
FeedRec [25]	Some behavior-specific subsequences of items and a sequence of (item, behavior) pairs	Local + Global	Consider implicit feedback noise by an attention network; Consider multiple patterns of the multi-behavior sequences.
NextIP [86]	Some behavior-specific subsequences of items and a sequence of (item, behavior) pairs	Local + Global	Treat the problem as the item prediction task and the purchase prediction task; Consider multiple patterns of the multi-behavior sequences.
MB-STR [87]	A sequence of (item, behavior) pairs	Local	A novel positional encoding function to model multi-behavior sequence relationships.
FLAG [85]	A behavior-agnostic sequence of items and a sequence of behaviors	Local + Global	Model user's local preference, local intention and global preference simultaneously.

Transformer-based Methods—MB-STR



➤ Conducting sequential modeling for multiple behaviors



➤ Challenges in modeling of multi-behavior sequential recommendations

- How to model heterogeneous multi-behavior dependencies at the fine-grained item-level
- How to model diverse multi-behavior sequential patterns effectively
- How to effectively mine users' multi-behavior sequence with multi-behavior supervision signals

	Multi-Behavior Modeling	Sequential Information	Behavior-Specific Prediction
MATN [39]	behavior-level	✗	✗
NMTR [9]	behavior-level	✗	✓
MBGCN [19]	behavior-level	✗	✗
MB-GMN [40]	behavior-level	✗	✓
DIPN [13]	behavior-level	fixed single behavior	✓
DMT [11]	behavior-level	fixed single behavior	✓
MB-STR(our)	heterogeneous item-level	diverse multi-behavior	✓

Multi-Behavior Transformer Layer



- Behavior-specific projection
 - n: number of behaviors in a sequence
- Cross behavior similarity
- Sequential pattern injection and softmax
 - P: multi-behavior sequential pattern matrix from MB-SPG
- Cross behavior information aggregation

Algorithm 1: Multi-Behavior Multi-head Self-Attention

Input: $\mathbf{H}^{(l-1)} \in \mathbb{R}^{n \times d}$, $\mathbf{b} \in \mathcal{B}^n$, $\mathbf{P}^{(l)} \in \mathbb{R}^{h \times n \times n}$

Output: $\mathbf{G}^{(l)} \in \mathbb{R}^{n \times d}$

1 **for** head $m = 1$ to h **do**

2 /* Step 1. Behavior-specific projection. */

3 $\mathbf{Q}_m \leftarrow f_{Q_m}(\mathbf{H}^{(l-1)}, \mathbf{b})$

4 $\mathbf{K}_m \leftarrow f_{K_m}(\mathbf{H}^{(l-1)}, \mathbf{b})$

4 $\mathbf{V}_m \leftarrow f_{V_m}(\mathbf{H}^{(l-1)}, \mathbf{b})$

5 /* Step 2. Cross behavior similarity. */

5 **for** $i = 1$ to n , $j = 1$ to n **do**

6 $A_m[i, j] = \frac{\mathbf{Q}_m[i] \mathbf{W}_{(\mathbf{b}[i], \mathbf{b}[j])}^{att} \mathbf{K}_m[j]}{\sqrt{d}}$

7 **end**

8 /* Step 3. Sequential pattern injection and softmax. */

8 $\mathbf{A}_m \leftarrow softmax(\mathbf{A}_m + \mathbf{P}[m])$

9 /* Step 4. Cross behavior information aggregation. */

9 **for** $i = 1$ to n **do**

10 $G_m^{(l)}[i] \leftarrow \sum_j \mathbf{A}_m[i, j] \cdot \mathbf{W}_{(\mathbf{b}[i], \mathbf{b}[j])}^{agg} \cdot \mathbf{V}_m[j]$

11 **end**

12 **end**

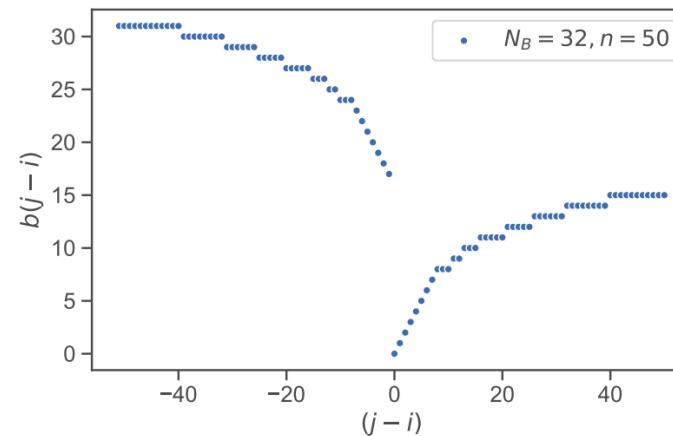
13 $\mathbf{G}^{(l)} \leftarrow \text{Concat}(\mathbf{G}_1^{(l)}, \dots, \mathbf{G}_h^{(l)})$

➤ A designed position encoding function for balanced position pairs

- k : heads
- i, j : behavior position in a sequence

$$P[k, i, j] = f_{(b[i], b[j])}(j - i)$$

$$b(j - i) = \begin{cases} B(j - i) & \text{if } (j - i) \geq 0 \\ B(-(j - i)) + \frac{N_B}{2} & \text{if } (j - i) < 0 \end{cases}$$



➤ Gating & expert

$$o_i = g_k(\mathbf{H}^{(L)}[i])^\top E_k(\mathbf{H}^{(L)}[i]),$$

$$g_k(x) = \text{softmax}(\mathbf{W}_g^k x), \quad k = \mathbf{b}[i]$$

$$E_k(x) = [e_{k,1}(x), e_{k,2}(x), \dots, e_{k,n_b}(x), e_{s,1}(x), e_{s,2}(x), \dots, e_{s,n_s}(x)]$$

$$\mathbf{p}_i(v) = \text{softmax}(o_i \cdot \mathbf{E}^\top)$$

Overall performance



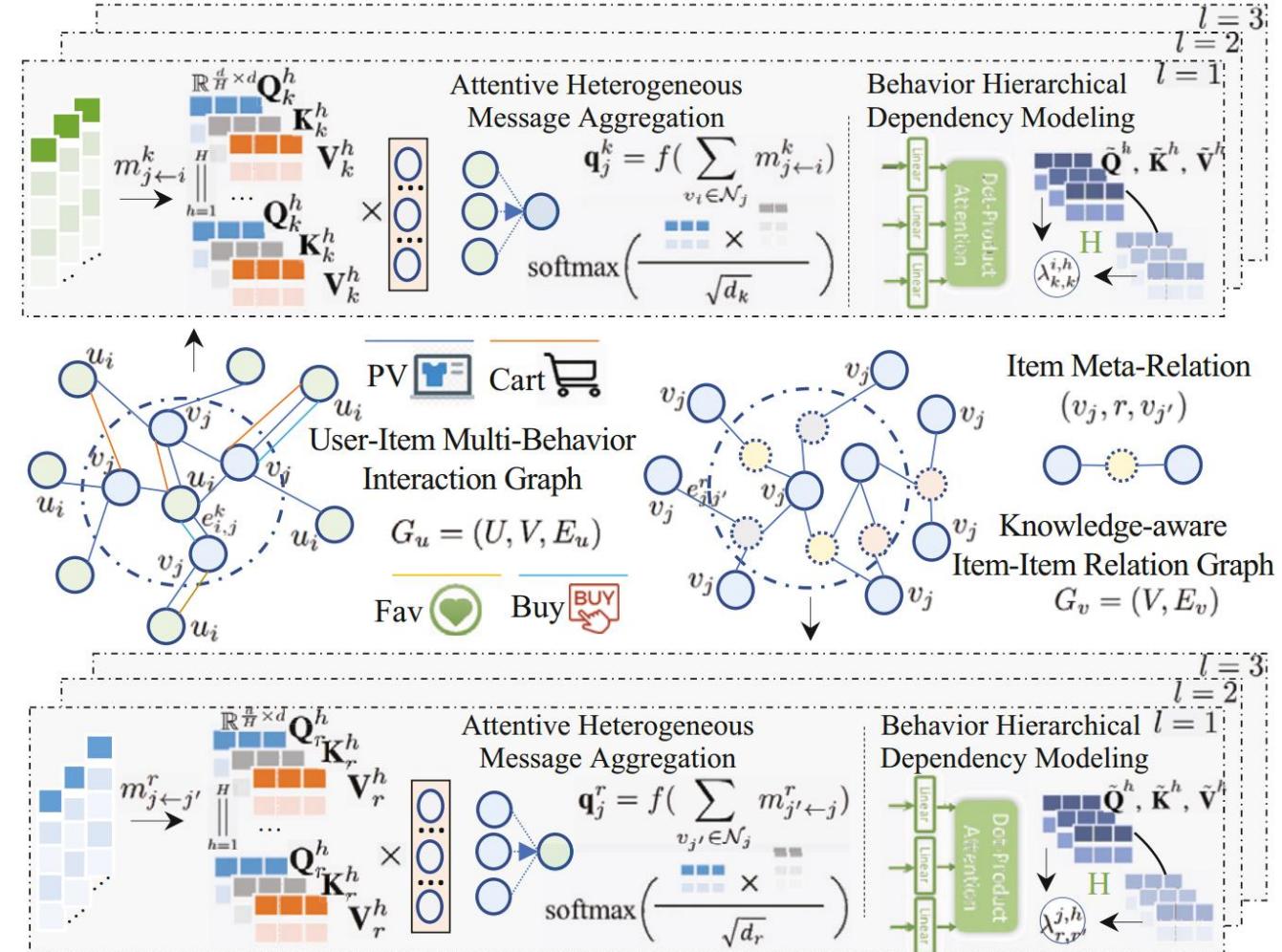
O/S: One/multiple behavior
 NS/S: Non-sequential/sequential

		Dataset	Yelp		Taobao		IJCAI		
		Metrics	HR	NDCG	HR	NDCG	HR	NDCG	
O	NS	MF	0.755	0.481	0.262	0.153	0.285	0.185	
		DMF	0.756	0.485	0.305	0.189	0.392	0.250	
		NGCF	0.789	0.500	0.302	0.185	0.461	0.292	
		LightGCN	0.810	0.513	0.373	0.235	0.443	0.283	
	S	SASRec	0.796	0.504	0.372	0.221	0.597	0.406	
		BERT4Rec	0.816	0.531	0.385	0.234	0.605	0.431	
M	NS	$NGCF_M$	0.793	0.492	0.374	0.221	0.481	0.307	
		$LightGCN_M$	0.872	0.585	0.391	0.243	0.486	0.317	
		NMTR	0.790	0.478	0.332	0.179	0.481	0.304	
		MATN	0.826	0.530	0.354	0.209	0.489	0.309	
		MBGCN	0.796	0.502	0.369	0.222	0.463	0.277	
		MB-GMN	0.87	0.582	0.491	0.300	0.532	0.345	
	S	DIPN	0.791	0.500	0.317	0.178	0.475	0.296	
		$SASRec_M$	0.819	0.531	0.637	0.442	0.795	0.611	
		$BERT4Rec_M$	0.838	0.558	0.675	0.476	0.816	0.632	
		DMT	0.652	0.515	0.666	0.415	0.682	0.513	
Our MB-STR			0.882*	0.624*	0.768*	0.608*	0.879*	0.713*	
Rela, Improv.			1.15%	6.67%	13.78%	27.73%	7.72%	12.82%	

Transformer-based Methods—KHGT



➤ Conducting graph-structured transformer



Taxonomy



- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

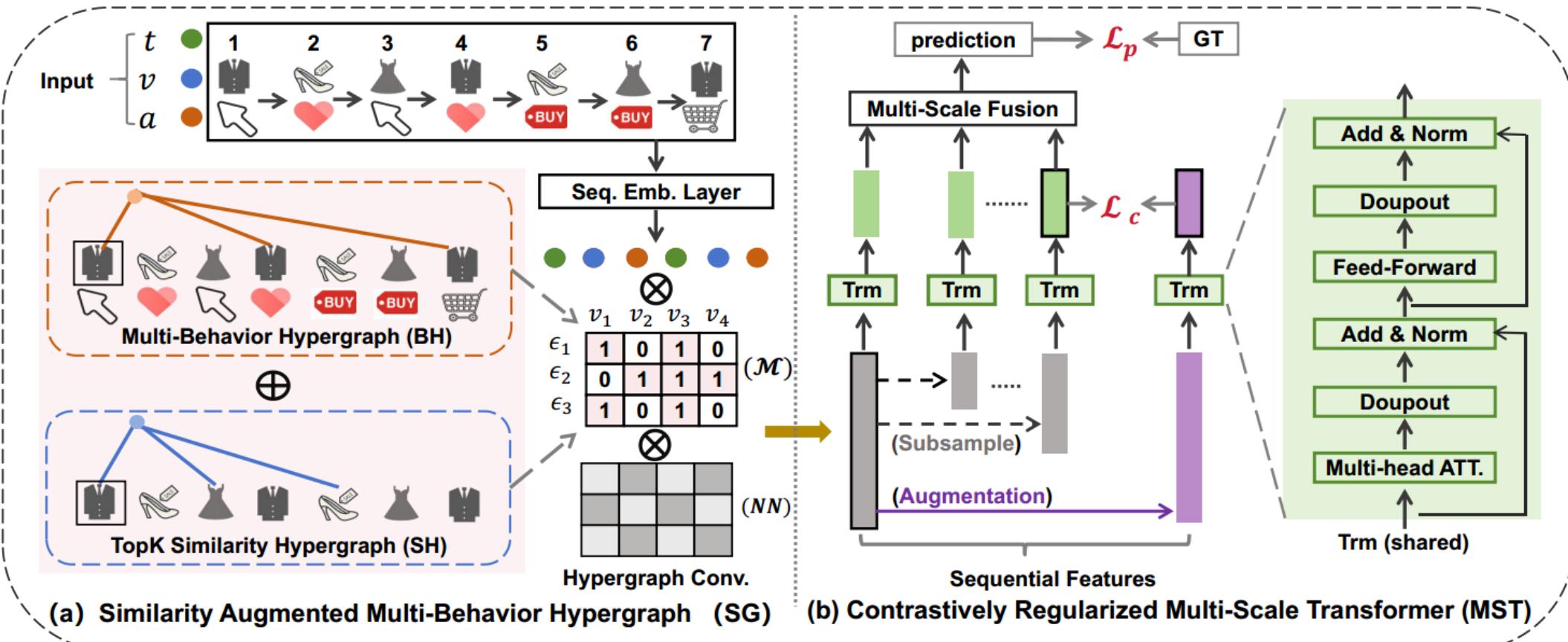
Methods	Prons	Cons
RNN-based	<ul style="list-style-type: none">• Suitable for sequence problems and can store short-term memories	<ul style="list-style-type: none">• Gradient disappearance & explosion problems• Inefficient in predicting future sequences• Rarely used currently
Graph-based	<ul style="list-style-type: none">• Detailed modeling for behavior relations• Improved performance	<ul style="list-style-type: none">• Suffering from low efficiency
Transformer-based	<ul style="list-style-type: none">• Exceptional performance from attention mechanism• Superior parallel computing capabilities• Enhanced ability to capture long-term dependencies• Stronger explanability	\
Others		

- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

Other Methods—SG-MST



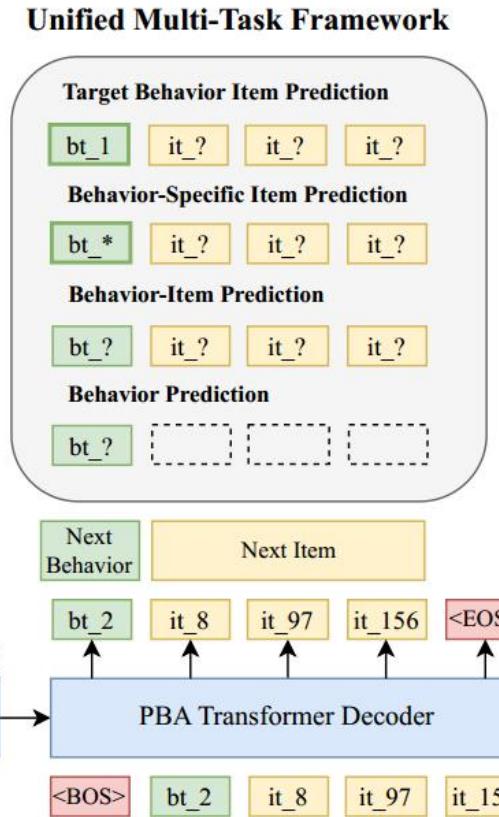
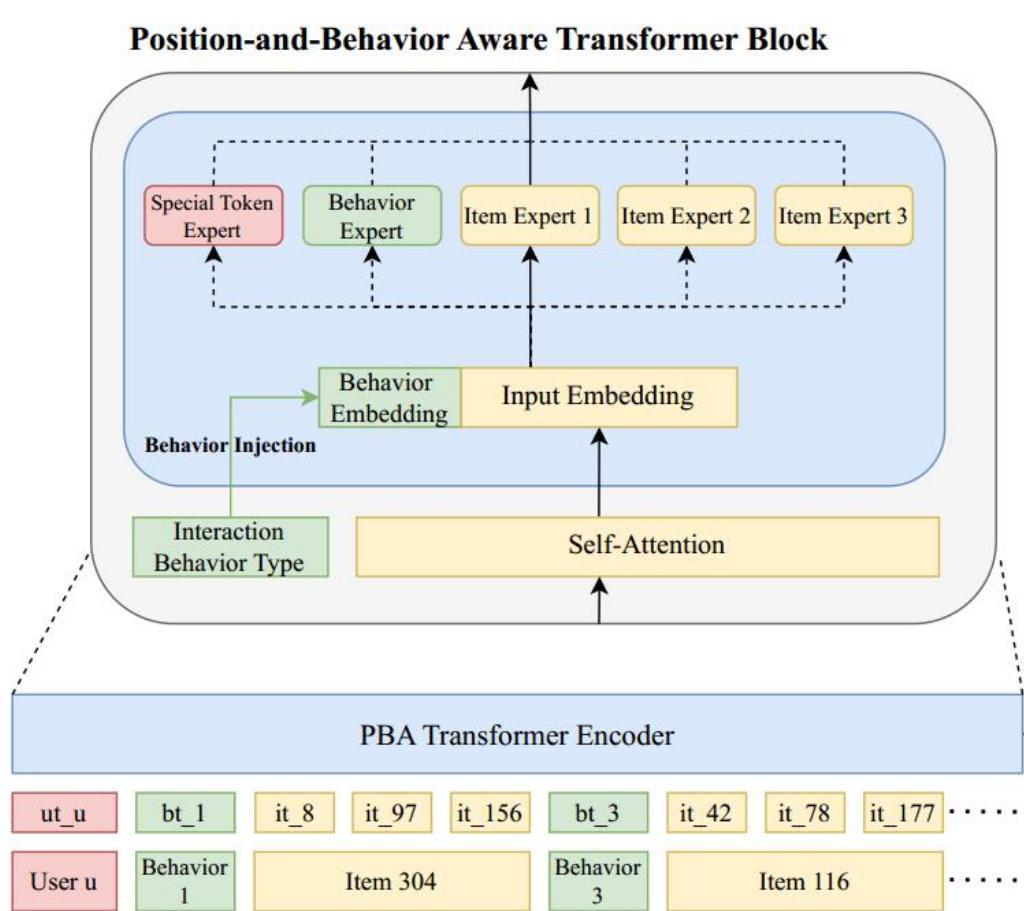
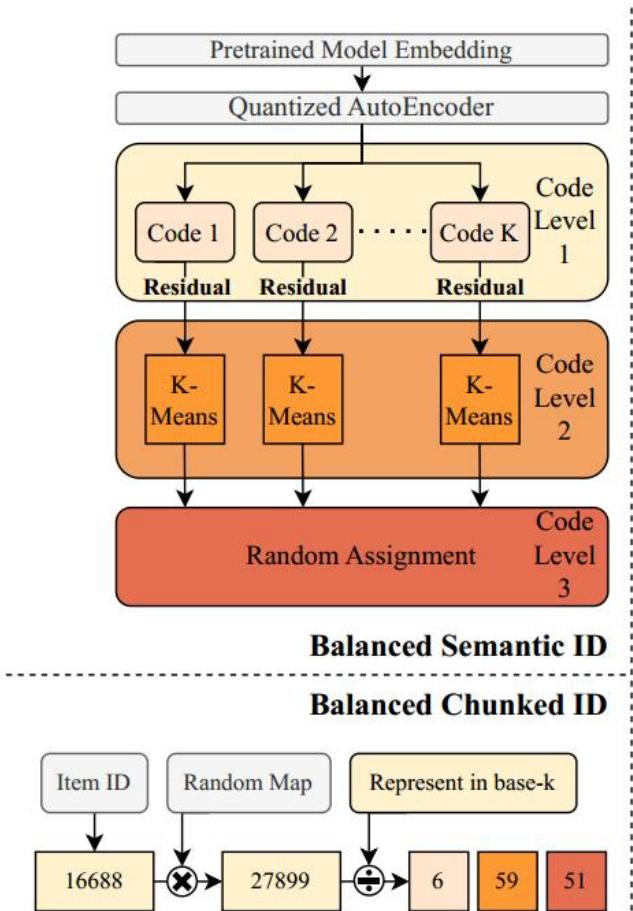
- Integrating a Similarity Augmented Multi-Behavior Hypergraph that captures complex behavior-aware dependencies among items and strengthens connections through item context similarities, producing more informative latent representations



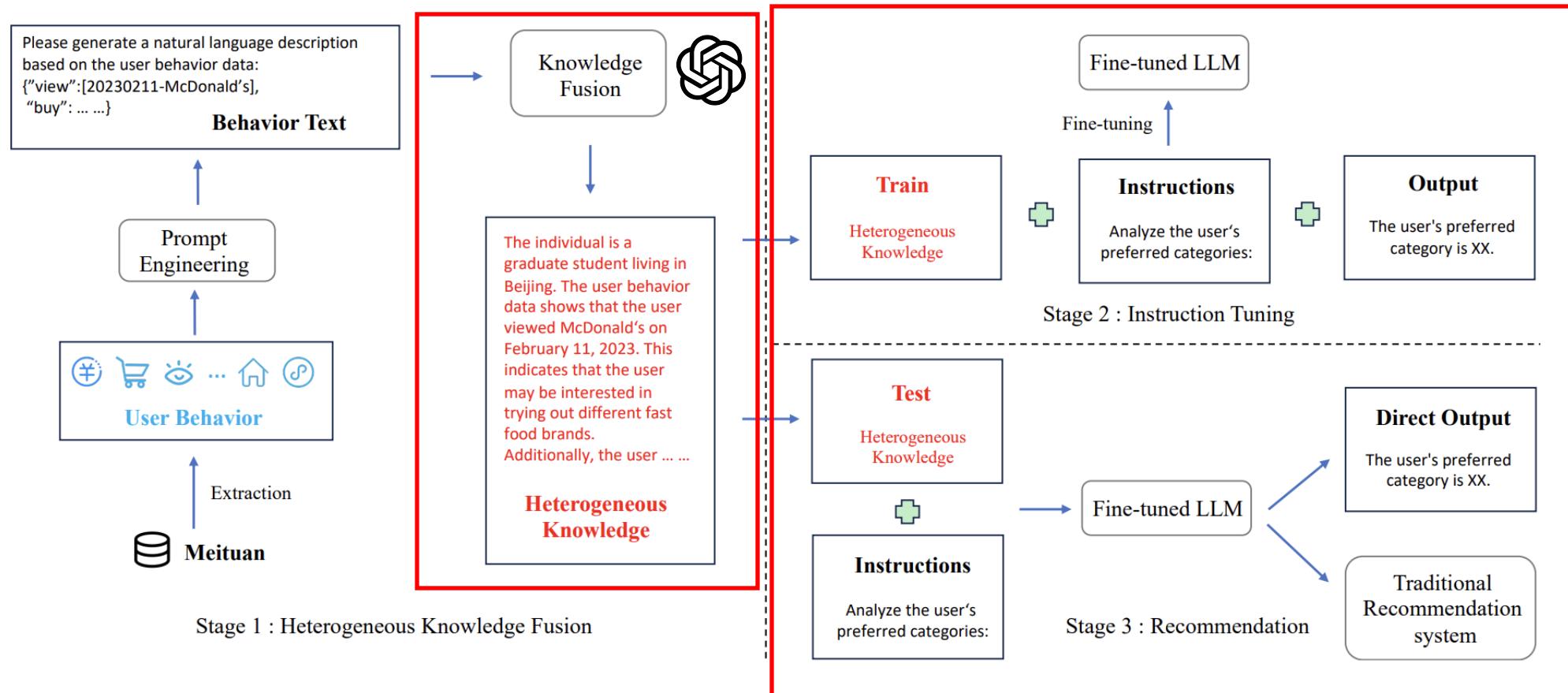
Other Methods—MBGen



➤ Generative modeling for multi-behavior sequential recommendations



➤ Representation modeling for behaviors via LLM



➤ Experiments

- HKFRno-IT: Without instruct tuning
- HKFRno-HKF: Without heterogeneous knowledge fusion

Methods	Category				POIs			
	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
Caser	0.1152	0.1063	0.2147	0.1320	0.0897	0.0770	0.1842	0.1012
BERT4Rec	0.1217	0.1140	0.2196	0.1440	0.0875	0.0744	0.1811	0.0995
P5	0.1416	0.1384	0.2477	0.1589	0.1218	0.1159	0.2187	0.1260
ChatGLM-6B	0.1074	0.1019	0.2038	0.1254	0.0785	0.0720	0.1702	0.0872
<i>HKFR_{no-IT}</i>	0.1241	0.1175	0.2267	0.1415	0.1014	0.0952	0.2050	0.1165
<i>HKFR_{no-HKF}</i>	0.1813	0.1308	0.2825	0.1580	0.1421	0.0975	0.2432	0.1270
HKFR	0.2160	0.1586	0.3007	0.1840	0.1726	0.1243	0.2610	0.1525

Taxonomy



- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

Methods	Prons	Cons
RNN-based	<ul style="list-style-type: none">• Suitable for sequence problems and can store short-term memories	<ul style="list-style-type: none">• Gradient disappearance & explosion problems• Inefficient in predicting future sequences• Rarely used currently
Graph-based	<ul style="list-style-type: none">• Detailed modeling for behavior relations• Improved performance	<ul style="list-style-type: none">• Suffering from low efficiency
Transformer-based	<ul style="list-style-type: none">• Exceptional performance from attention mechanism• Superior parallel computing capabilities• Enhanced ability to capture long-term dependencies• Stronger explanability	\
Others	<ul style="list-style-type: none">• Better modeling structure• Generative modeling• Modeling with LLM	

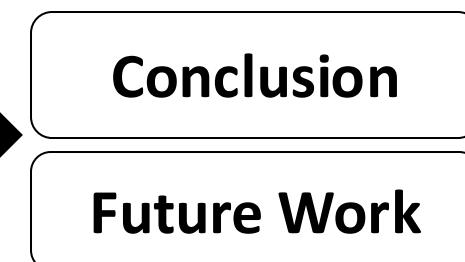
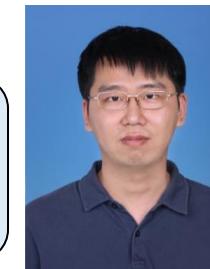
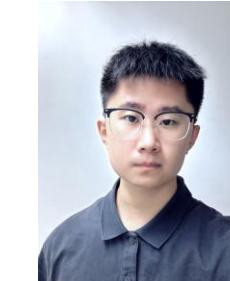
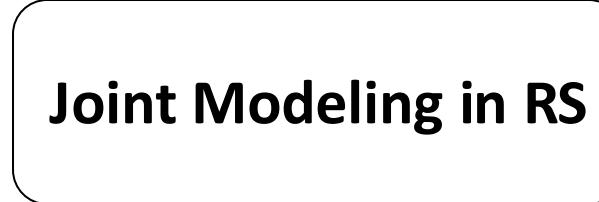
Conclusion



Methods	Prons	Cons
RNN-based	<ul style="list-style-type: none"> Suitable for sequence problems and can store short-term memories 	<ul style="list-style-type: none"> Gradient disappearance & explosion problems Inefficient in predicting future sequences Rarely used currently
Graph-based	<ul style="list-style-type: none"> Detailed modeling for behavior relations Improved performance 	<ul style="list-style-type: none"> Suffering from low efficiency
Transformer-based	<ul style="list-style-type: none"> Exceptional performance from attention mechanism Superior parallel computing capabilities Enhanced ability to capture long-term dependencies Stronger explainability 	\
Others	<ul style="list-style-type: none"> Better modeling structure Generative modeling Modeling with LLM 	

Model	Methods
RLBL	RNN-based
MBN	RNN-based
MB-GCN	Graph-based
CML	Graph-based
MB-STR	Transformer-based
KHGT	Transformer-based
SG-MST	Other
MBGen	Other
HKFR	Other

- **Deeper information fusion**
 - Better representation modeling
- **More efficient learning method**
 - Better modeling for better behavior modeling and lighter computation burden
- **More explainable user representations**
 - Improving explanability
- **Fine-grained modeling with LLM**
 - Conducting fine-grained modeling with LLM



Yichao Wang

Multi-Modal Modeling



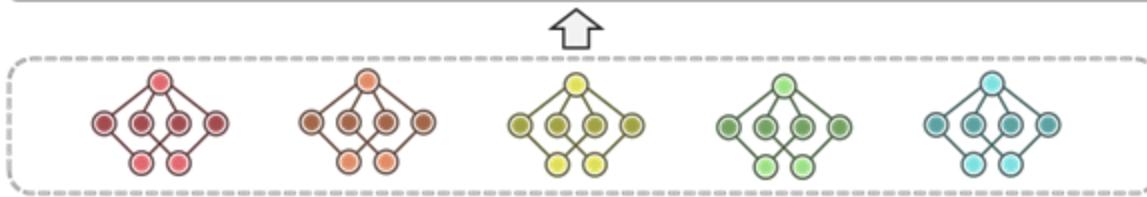
Multi-Scenario



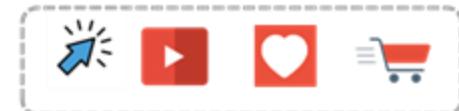
Multi-Task



Task/scenario adaption



Representation extraction



Multi-Behavior



Multi-Modal

$$wL(\mathbf{E}^{Merge}, \boldsymbol{\theta}^{sh}, \boldsymbol{\theta}^t, \boldsymbol{\theta}^s)$$

Joint Modeling

$$\mathbf{E}^{Merge} = U(\mathbf{E}, \mathbf{E}^B, \mathbf{E}^M)$$

Multi-behavior

$$\mathbf{E}^B = G(\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_N)$$

Multi-modal

$$wL(\mathbf{E}^{Merge}, \boldsymbol{\theta}^{sh}, \boldsymbol{\theta}^t, \boldsymbol{\theta}^s)$$

Multi-scenario

$$wL(\mathbf{E}^{Merge}, \boldsymbol{\theta}^{sh}, \boldsymbol{\theta}^t, \boldsymbol{\theta}^s)$$

Multi-task

➤ **Problem:**

- Learning a **unified multimodal representations** for users and items by various raw multimodal features (x^{txt}, x^v, \dots, x^p)

➤ **Optimization problem:**

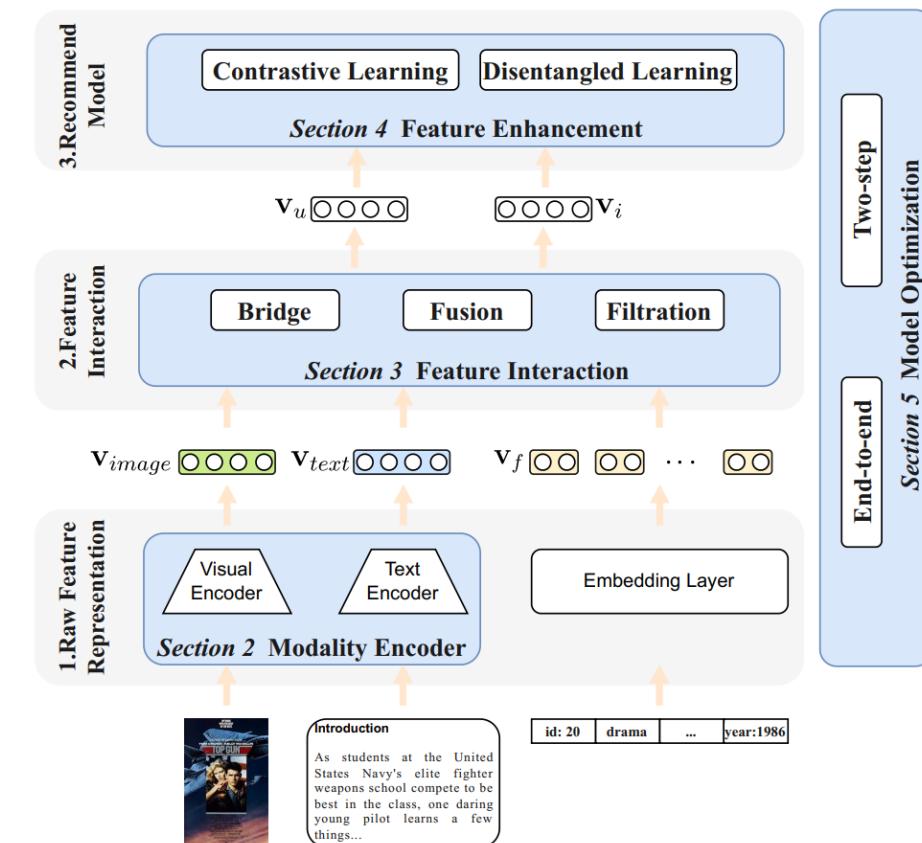
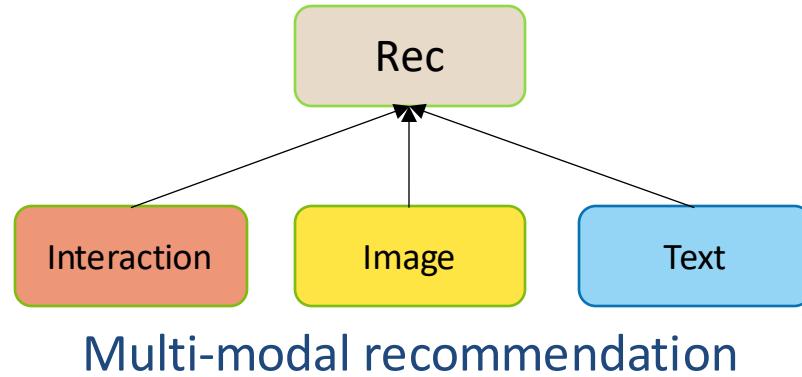
$$E^M = M(E^{txt}, E^v, \dots, E^p) = M(\mathcal{E}_{txt}(x^{txt}), \mathcal{E}_v(x^v), \dots, \mathcal{E}_p(x^p))$$

- $\mathcal{E}_*(\cdot)$: the corresponding modality encoder
- $M(\cdot)$: feature interaction function, enabling combination of various modalities

Multimodal Recommender Systems (MRS)



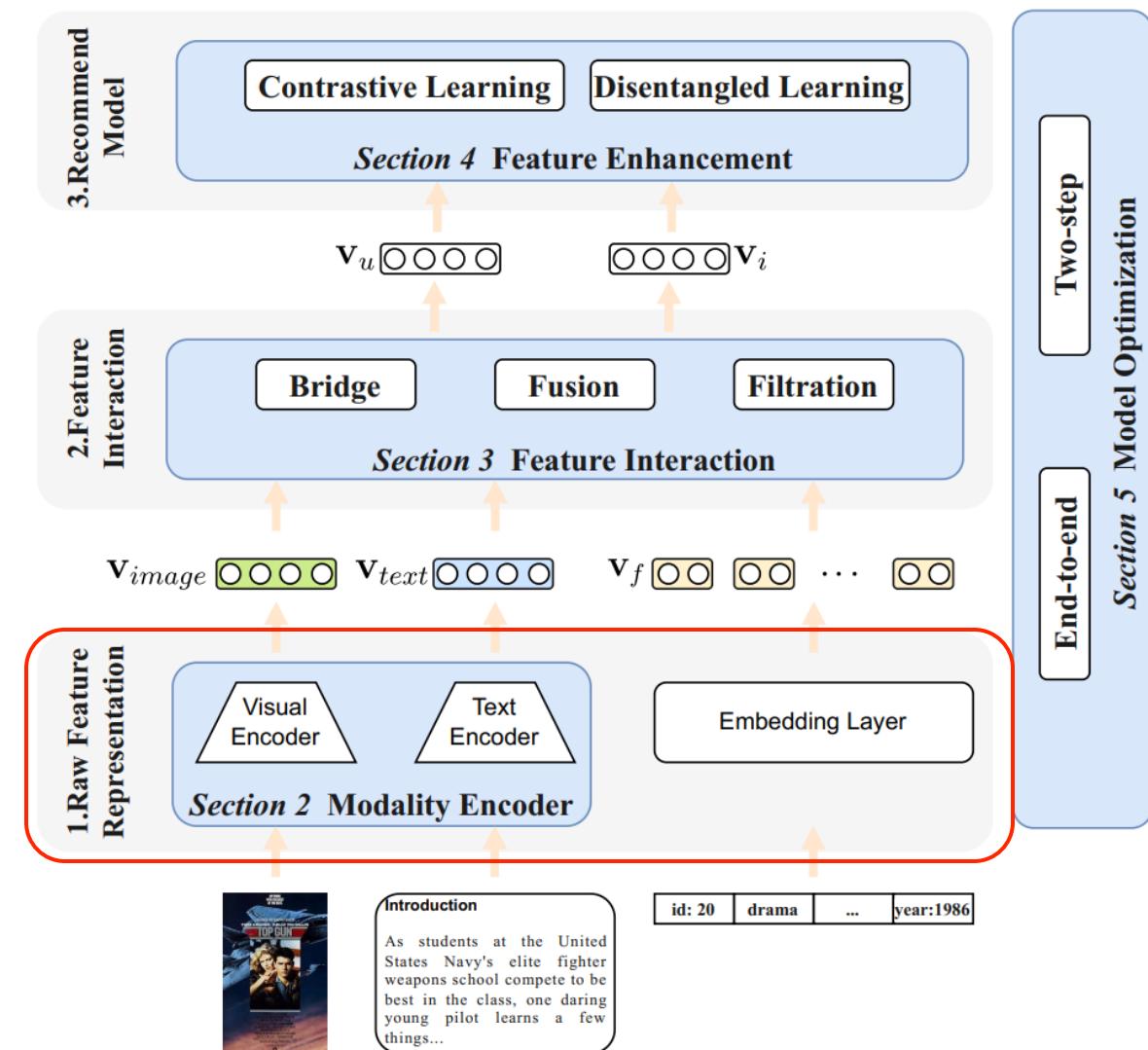
- Using various types of information generated by multimedia applications and services to enhance recommender systems' performance
- Making use of **multimodal features** simultaneously, such as image, audio, and text



Procedures of MRS

➤ Modality Encoder

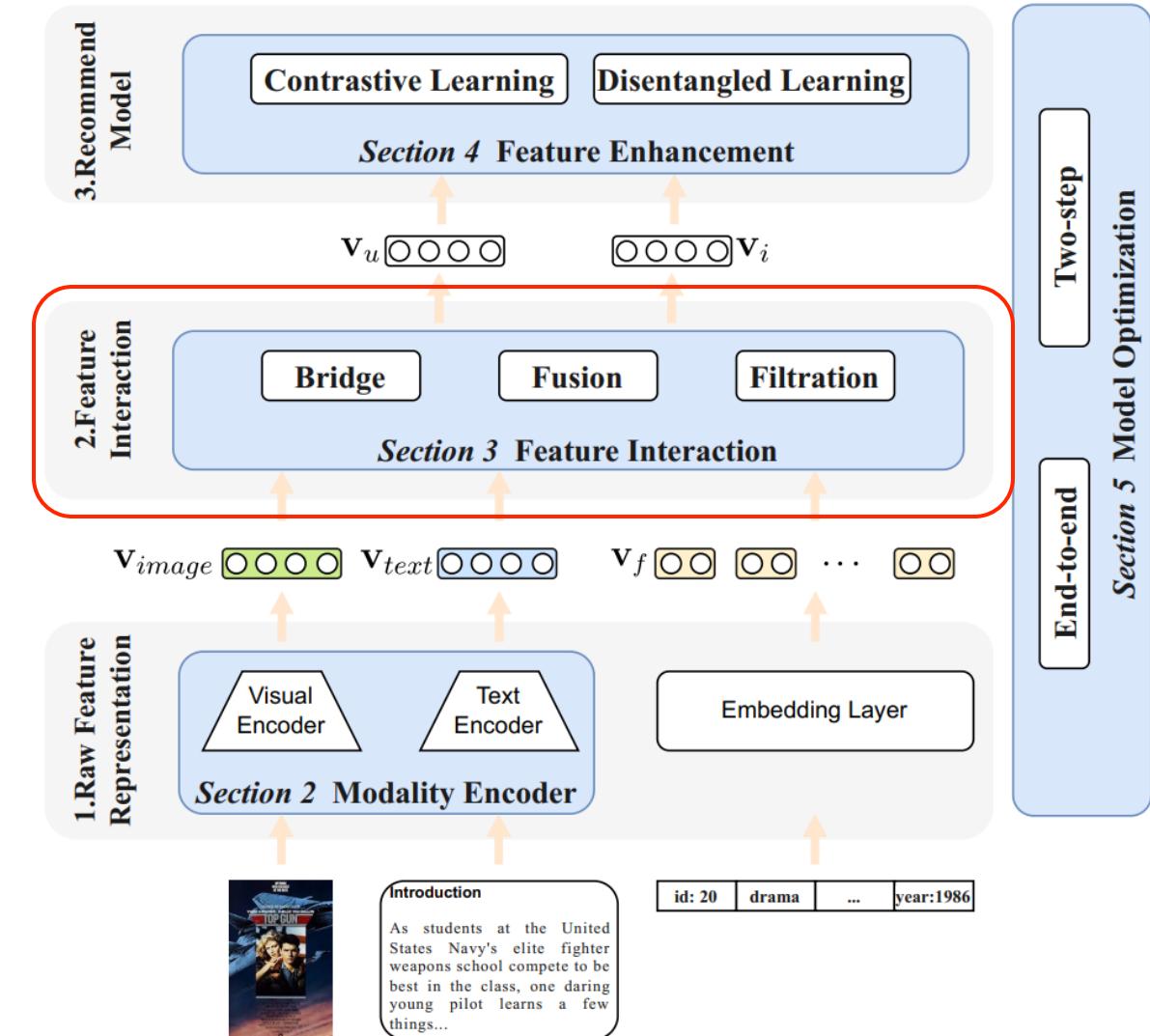
- **Challenge:** how to extract representations from complex raw features
- **Specialty:** various encoders



Challenges



- Modality Encoder
- Feature Interaction
 - **Challenge:** how to fuse the modality features in different semantic spaces
 - **Specialty:** modality alignment and fusion

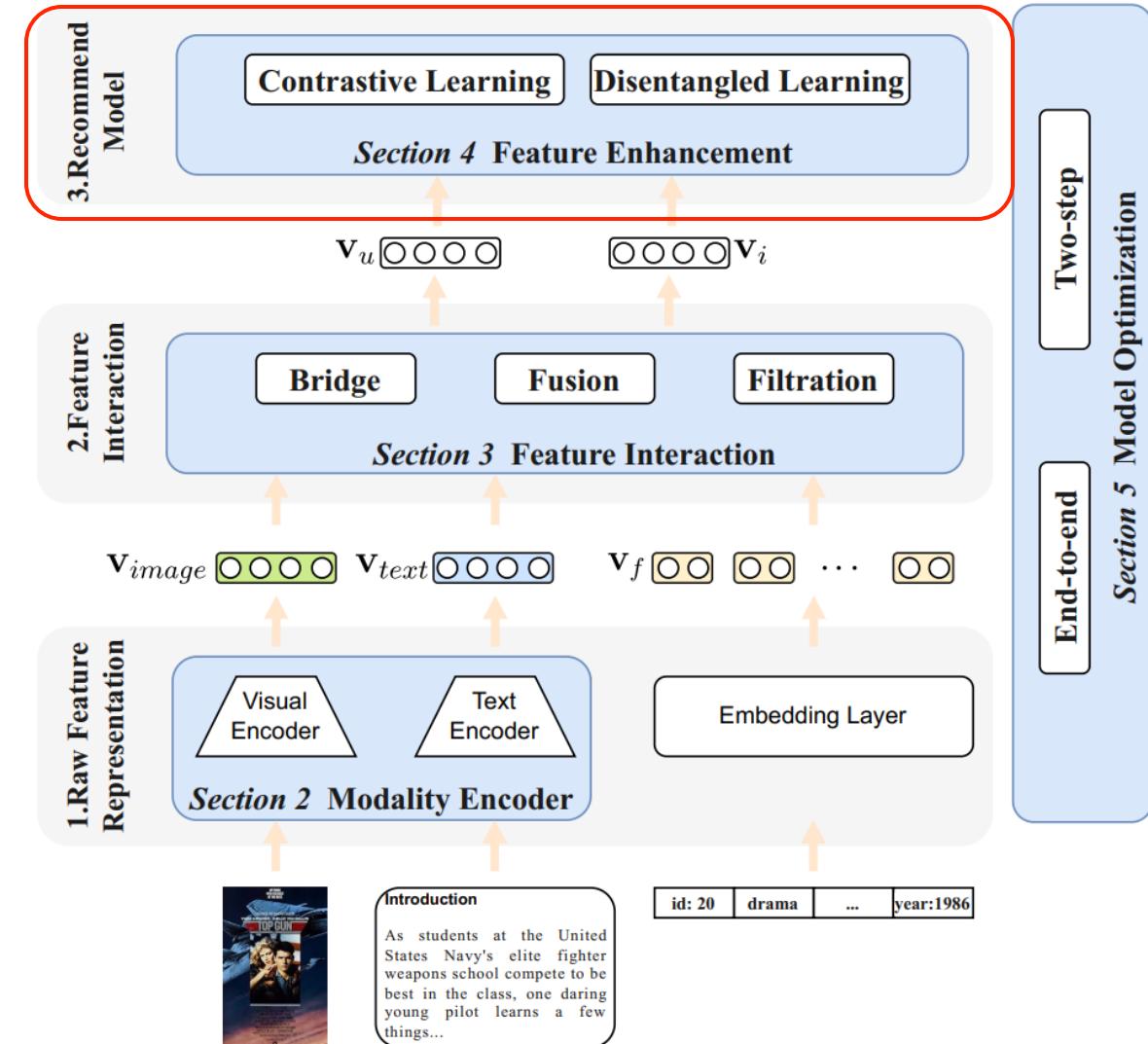


Procedures of MRS

Challenges



- Modality Encoder
- Feature Interaction
- Feature Enhancement
 - **Challenge:** how to get comprehensive representations for recommendation models under the data-sparse condition
 - **Specialty:** multimodal enhancement

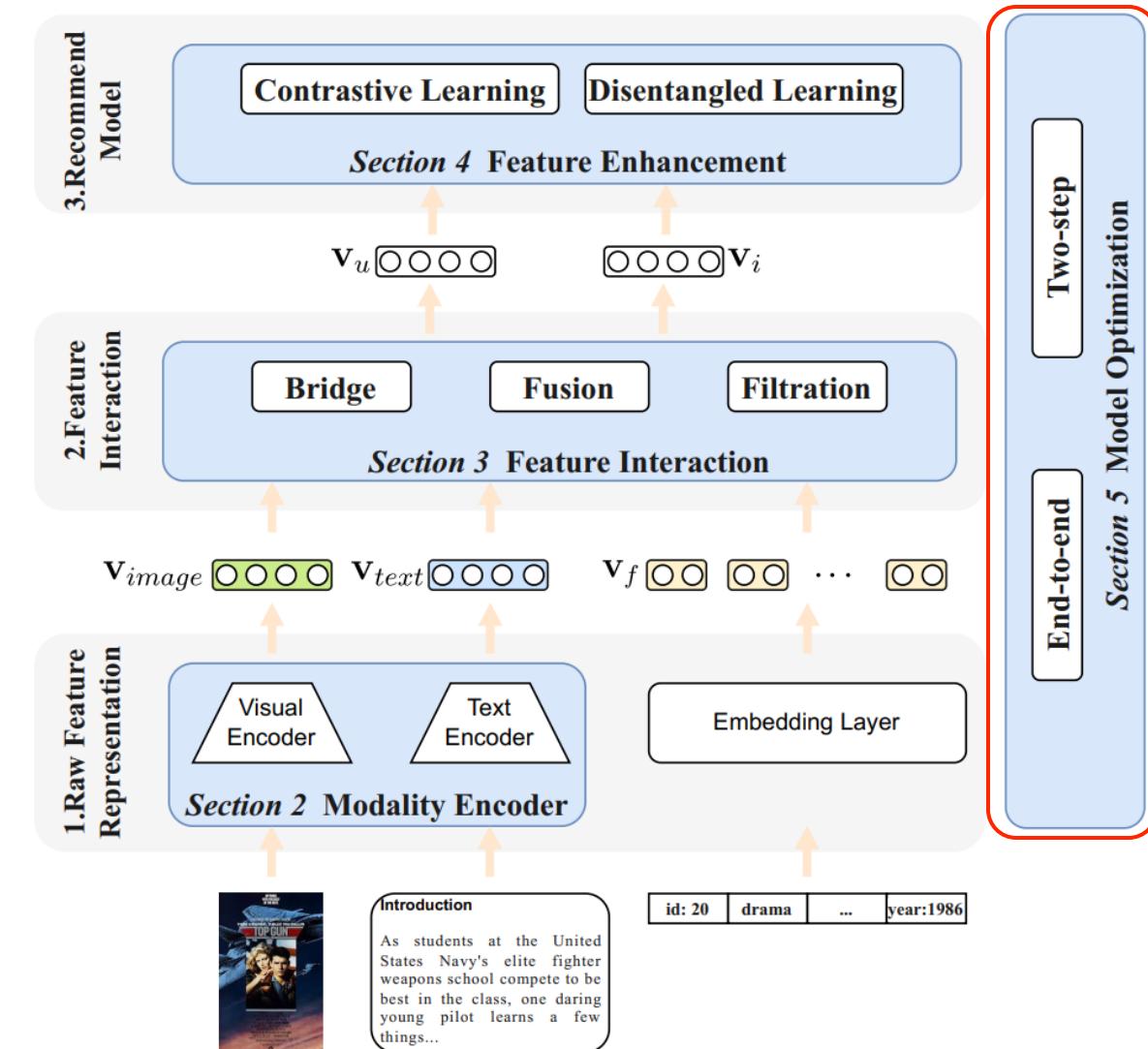


Procedures of MRS

Challenges



- Modality Encoder
- Feature Interaction
- Feature Enhancement
- Model Optimization
 - **Challenge:** how to optimize the lightweight recommendation models and parameterized modality encoder
 - **Specialty:** parameterized modality encoder



Procedures of MRS

Modality Encoder

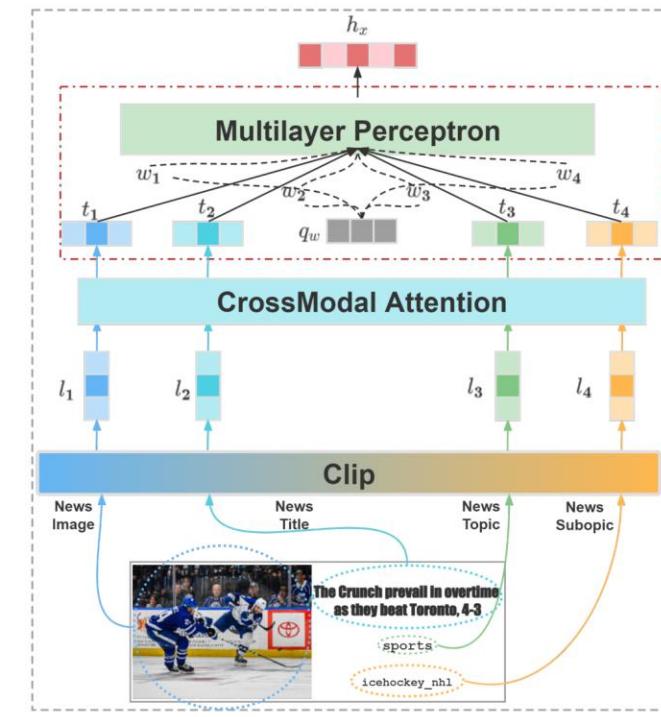


➤ **Target:** encoding different multimodal features

➤ **Taxonomy:**

- **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



Example:
Multimodal encoder
in VLSNR: Clip+ViT

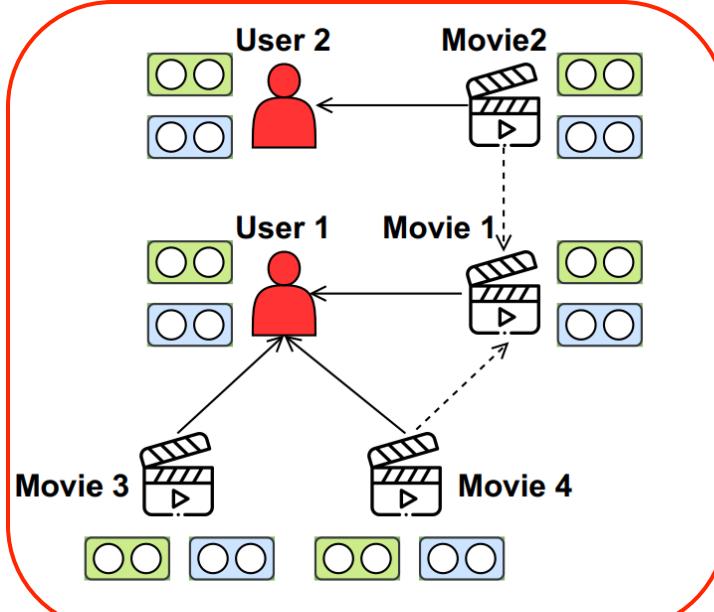
Feature Interaction



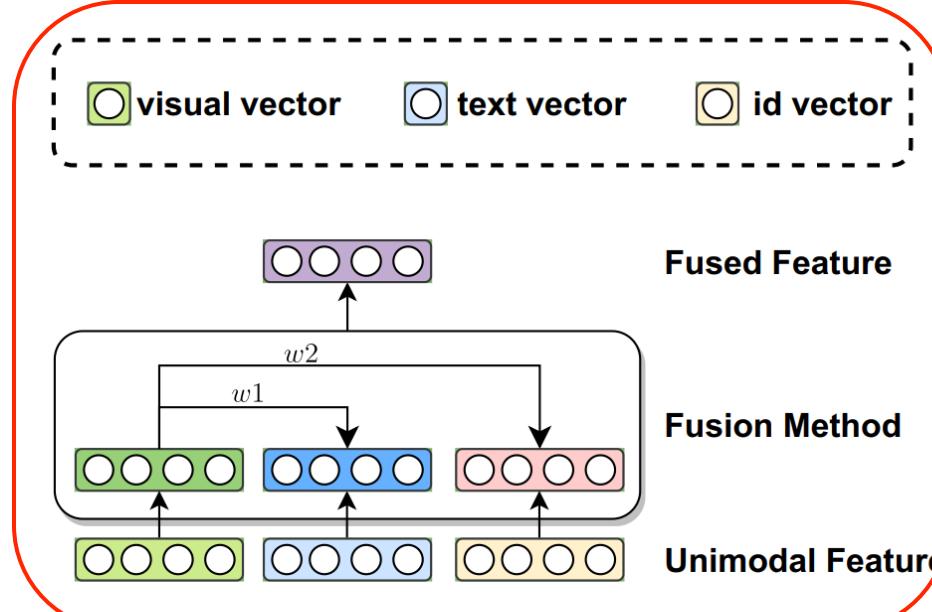
➤ **Target:** connecting various modalities

➤ **Taxonomy:**

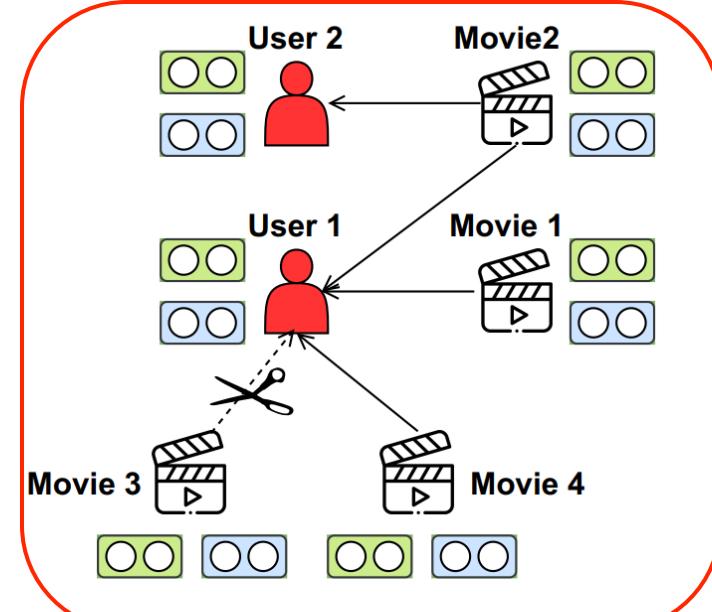
- **Bridge:** capturing **inter-relationship** between users and items considering modalities
- **Fusion:** capturing multimodal **intra-relationships** of items
- **Filtration:** filtering out **noisy data** in interaction graph or multimodal features



(a) Bridge



(b) Fusion



(c) Filtration

Feature Interaction: Bridge

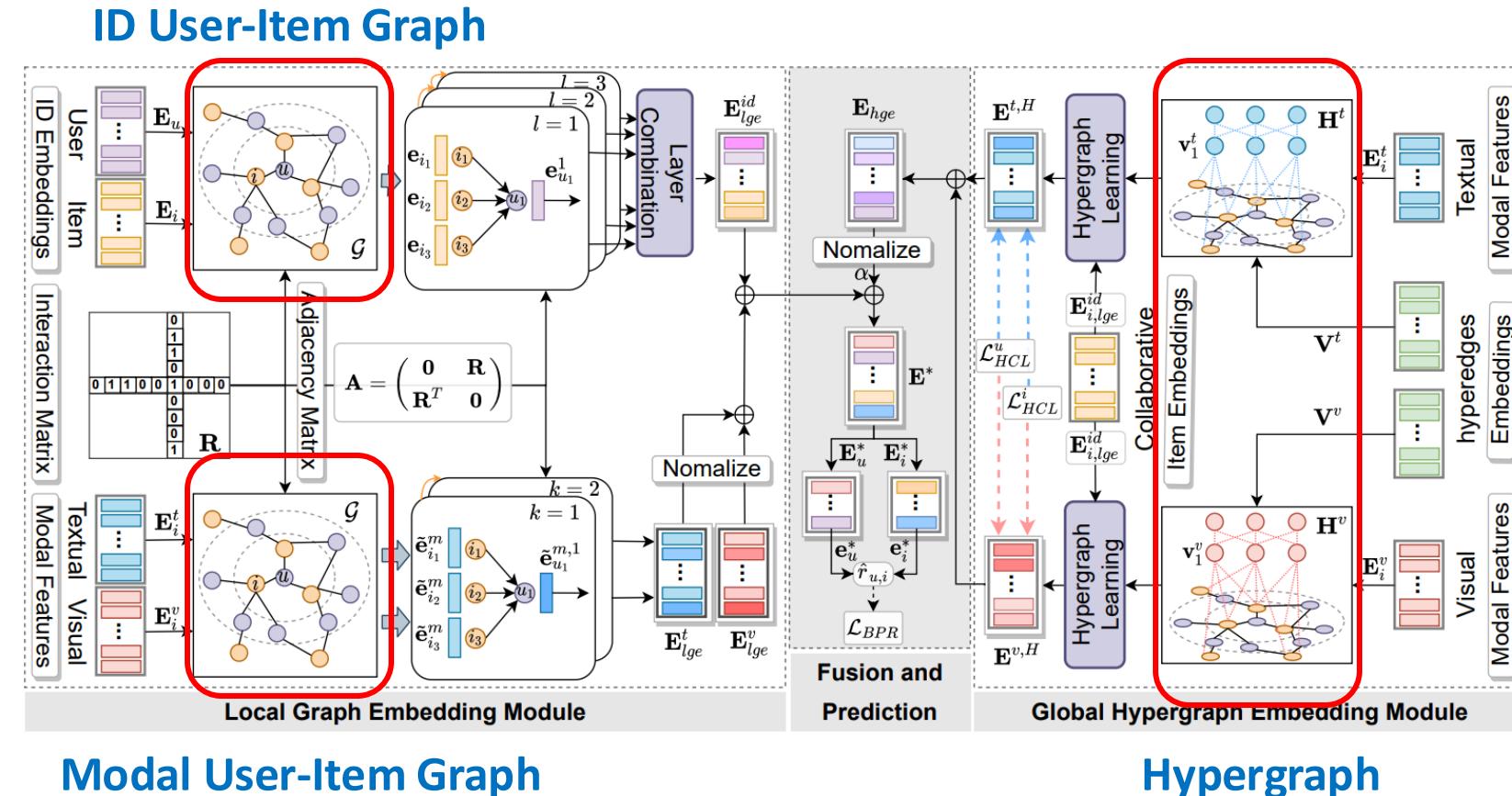


User-Item Graph

- Leveraging the information exchange between users and items to capture user's multimodal preferences

LGMRec (AAAI'24)

- User-Item Graph:** capture local preference
- Hypergraph:** capture global preference
- Aggregating local and global preferences



Feature Interaction: Bridge

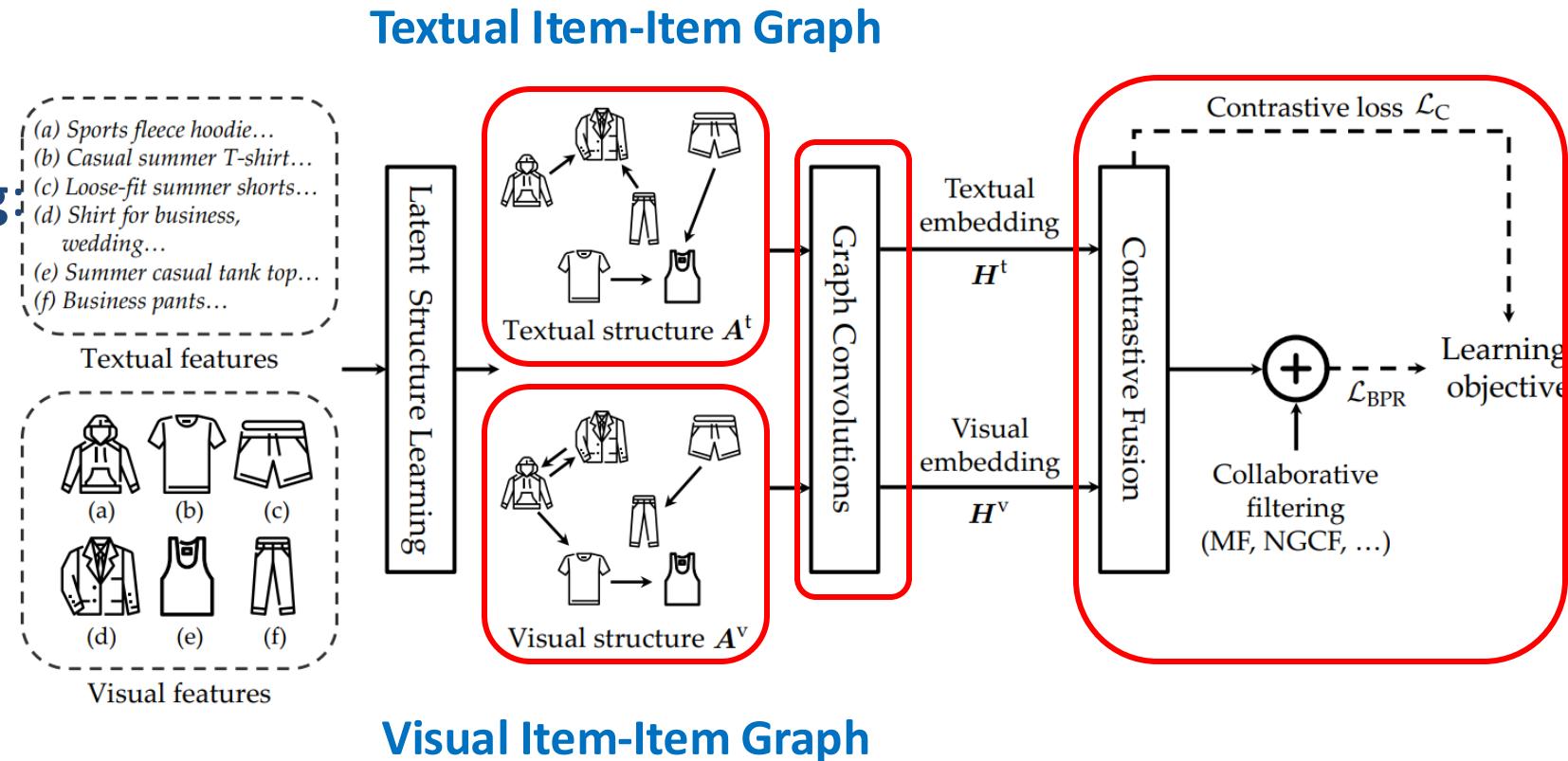


➤ Item-Item Graph

- Capturing latent semantic item-item structures to better learn item representations and improve model performance

➤ MICRO (TKDE'22)

- Latent Structure Learning:** connect items by similarity in each modal
- Graph Convolutions:** encodes two graphs
- Contrastive Fusion:** learn fine-grained multimodal representations



Feature Interaction: Bridge



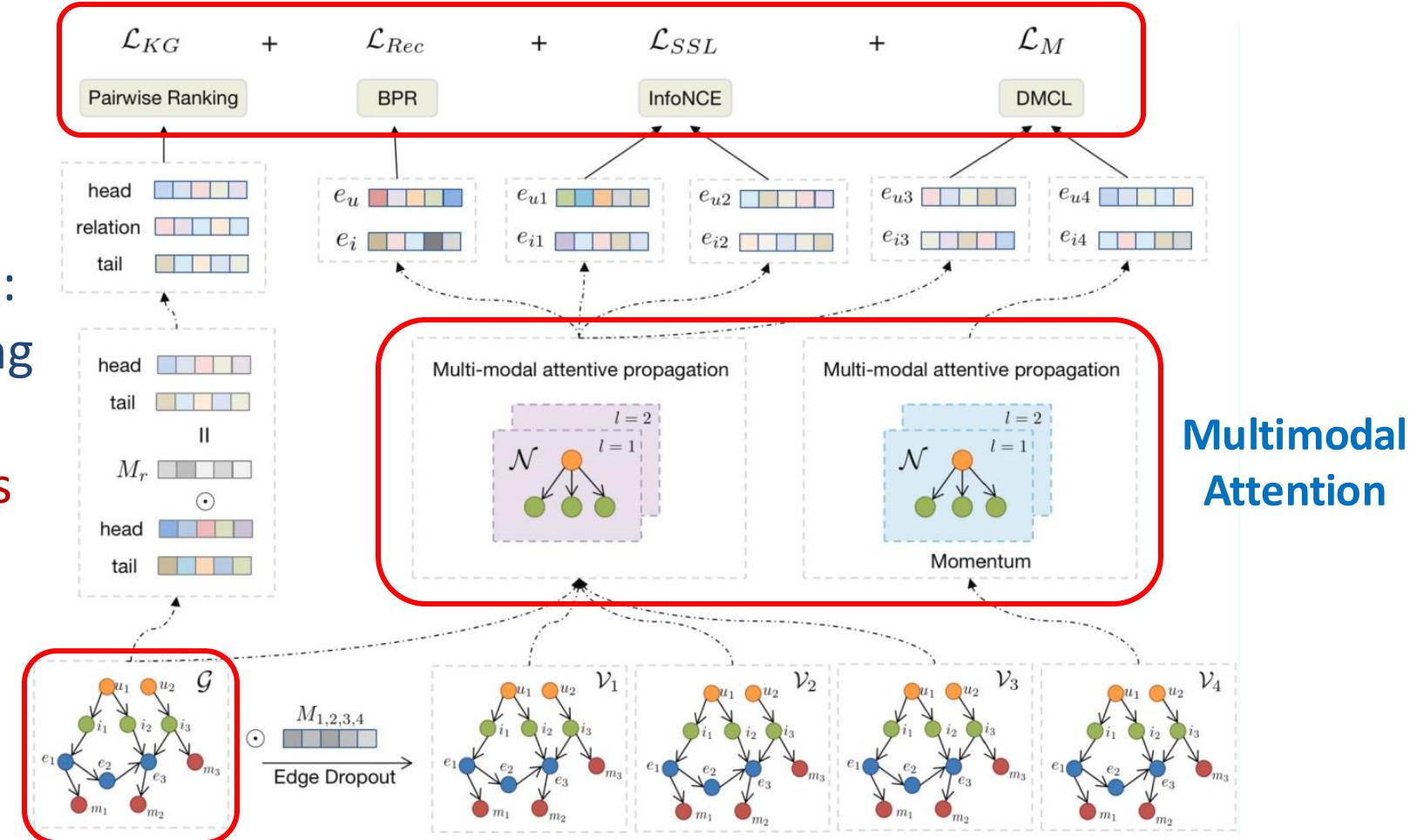
➤ Knowledge Graph

- Utilizing auxiliary information from **multimodal knowledge graph** to learn better multimodal representations

➤ M3KGR (IS'24)

- CLIP+Multimodal Attention:** get **aligned entity embedding**
- Momentum Updating:** increase quality of **negatives**
- Multi-task Learning:** better item representations

Multimodal
KG



Feature Interaction: Fusion

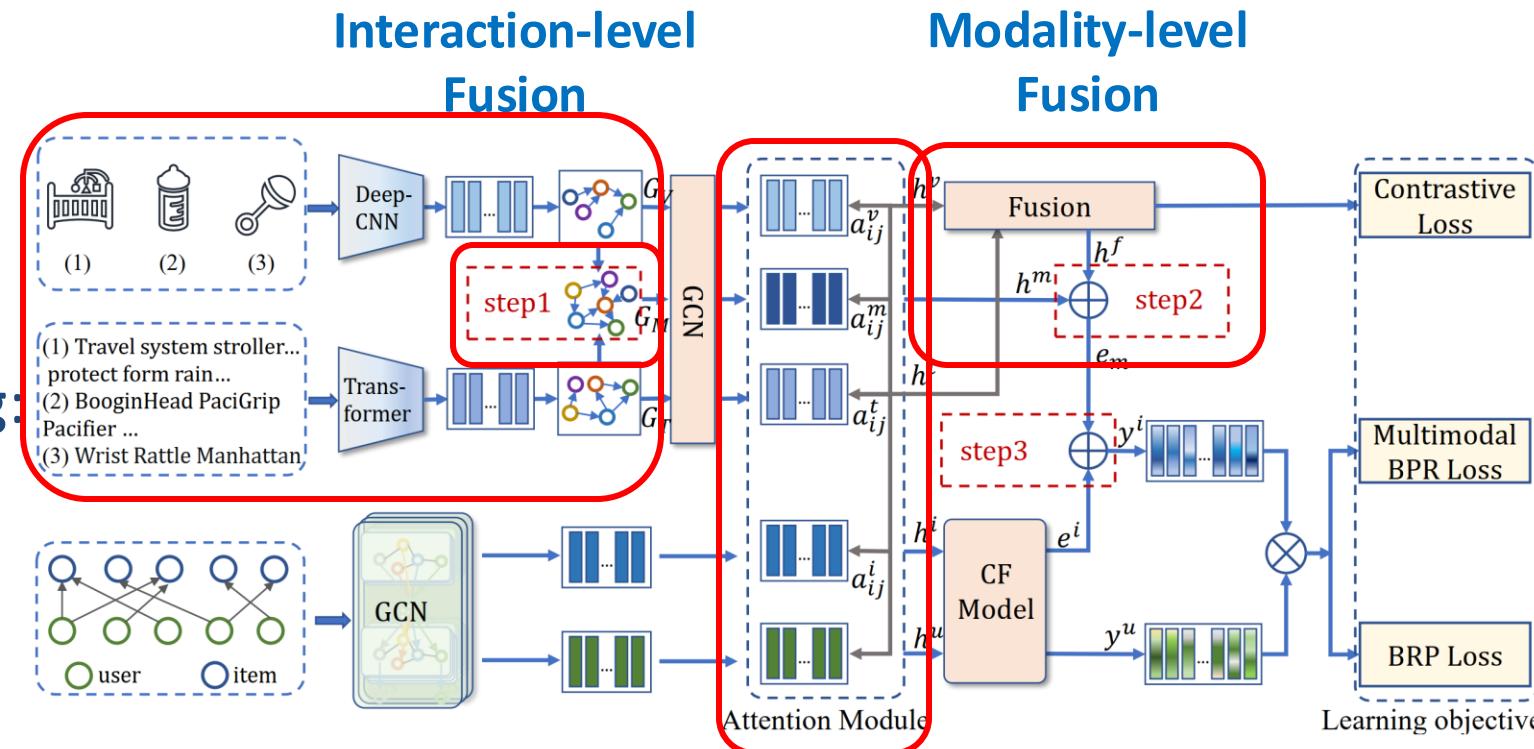


➤ Coarse-grained Attention

- Capturing the multimodal relationships between interactions

➤ TMFUN (SIGIR'23)

- **Graph Construction:** item-item graph for each modal
- **Attention Relationship Mining:** extract user's preferences
- **Multi-step Fusion:** interaction level and modality level



$$G_M = \lambda G_V + (1 - \lambda) G_T$$

Feature Interaction: Fusion



➤ Fine-grained Attention

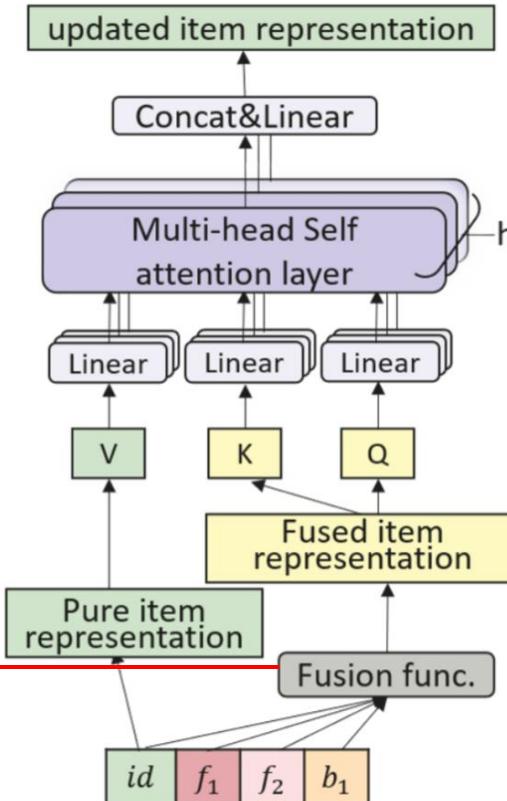
- Capturing the multimodal relationships between modalities

➤ NOVA (AAAI'21)

- **Non-invasive Attention:** Q and K are multimodal feature embeddings, V is the ID embedding
- **Fusion Operations:** gating fusion with trainable coefficients

$$\text{NOVA}(R, R^{(ID)}) = \sigma\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$$\mathcal{F}_{\text{gating}}(f_1, \dots, f_m) = \sum_{i=1}^m G^{(i)} f_i$$
$$G = \sigma(FW^F)$$

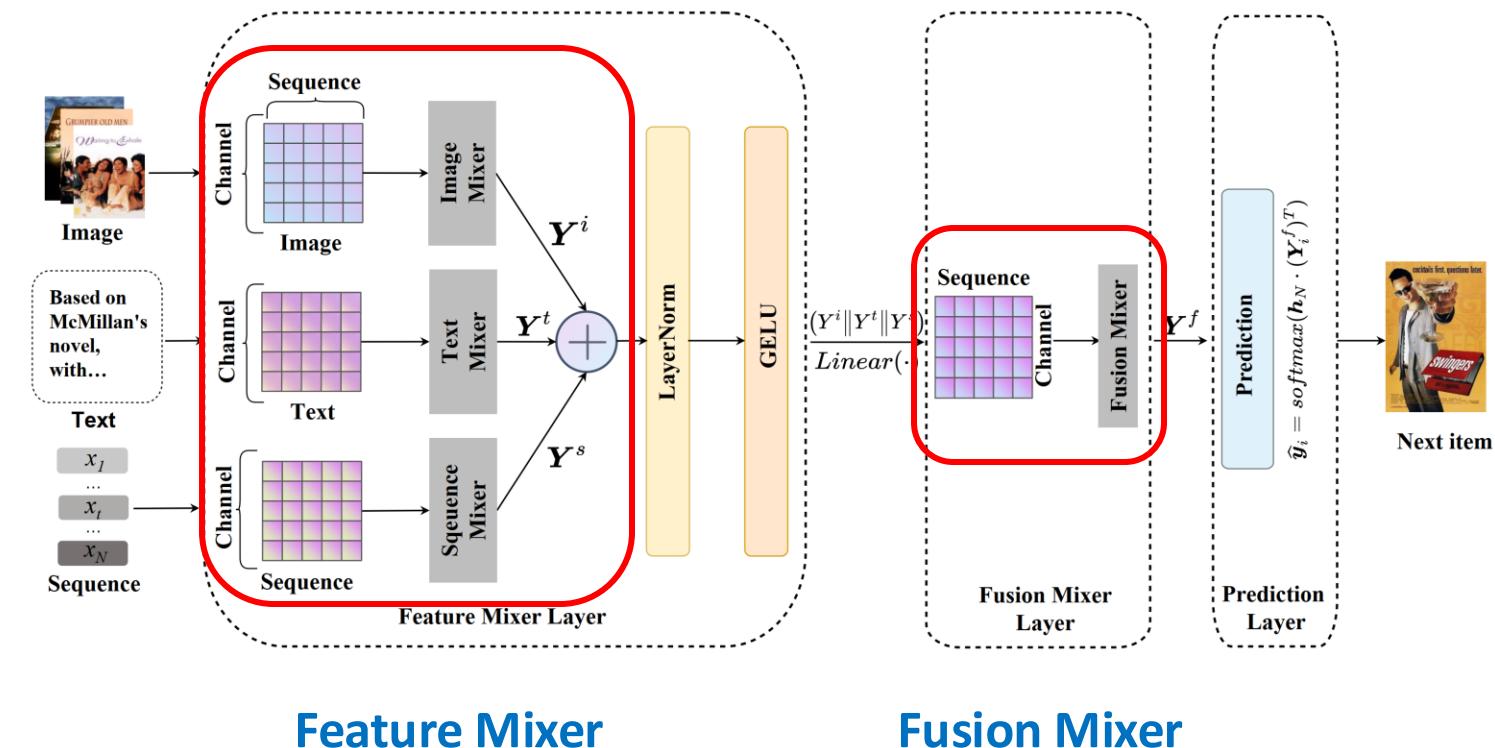


➤ Other Methods

- Applying other simple methods, including concatenation operations, gating mechanism and etc.

➤ MMMLP (WWW'23)

- **Feature Mixer Layer:** learn sequential patterns
- **Fusion Mixer Layer:** fuse various modalities



Feature Mixer

Fusion Mixer

Feature Interaction: Filtration

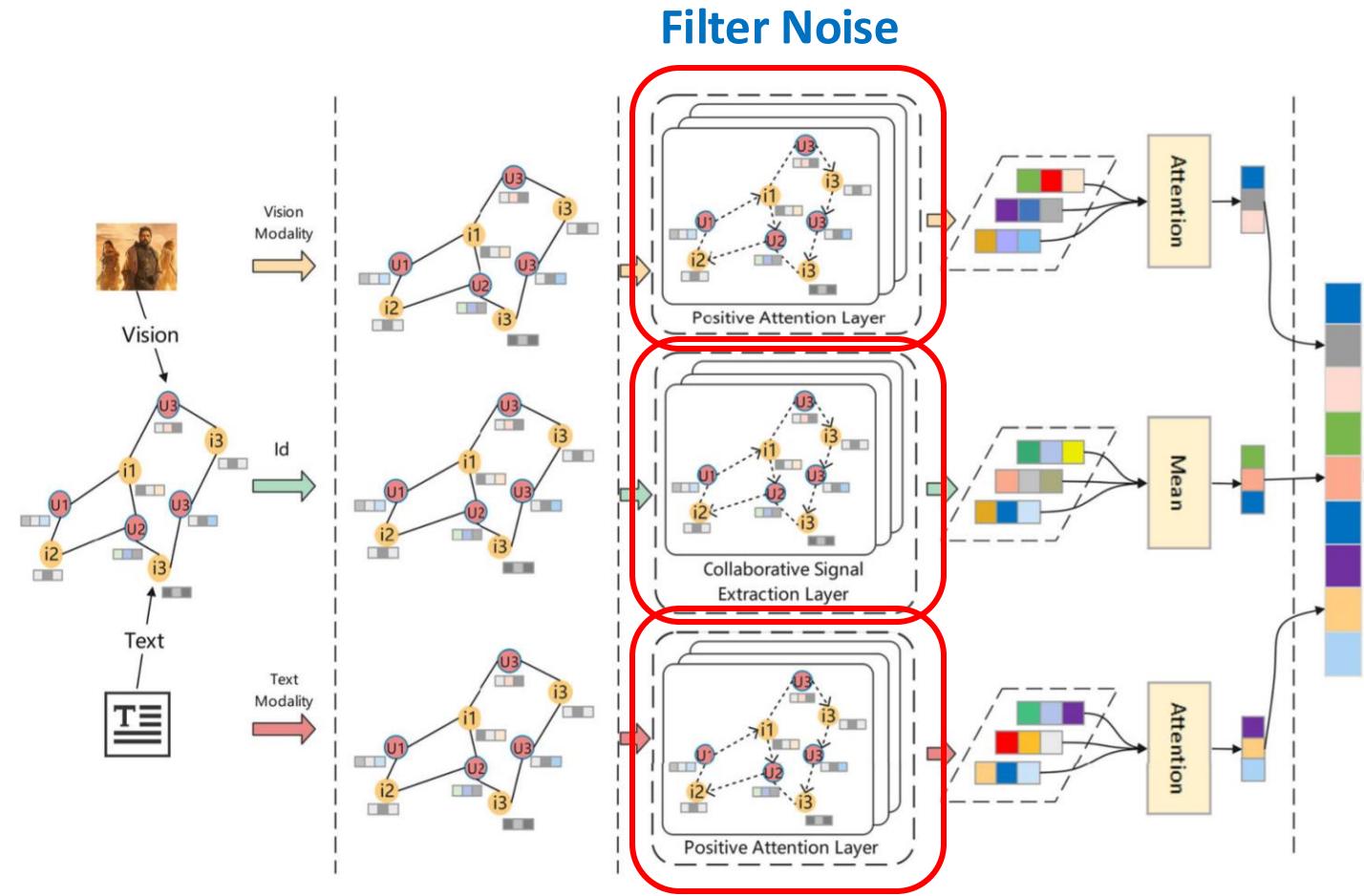


➤ Filtration

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

➤ PMGCRN (APPL INTELL'23)

- **Positive Attention Layer:** remove implicit noisy edges by **node similarity**
- **Collaborative Signal Extraction Layer:** integrate original user-item interaction information



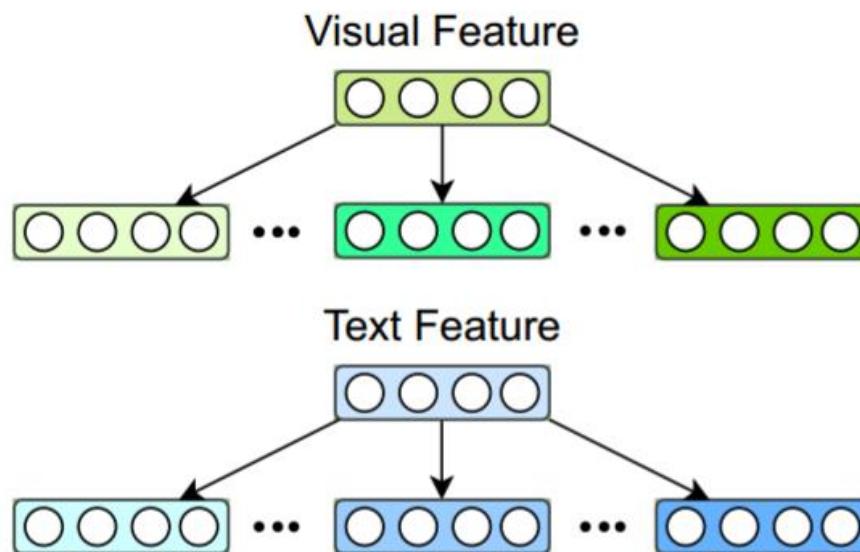
Feature Enhancement



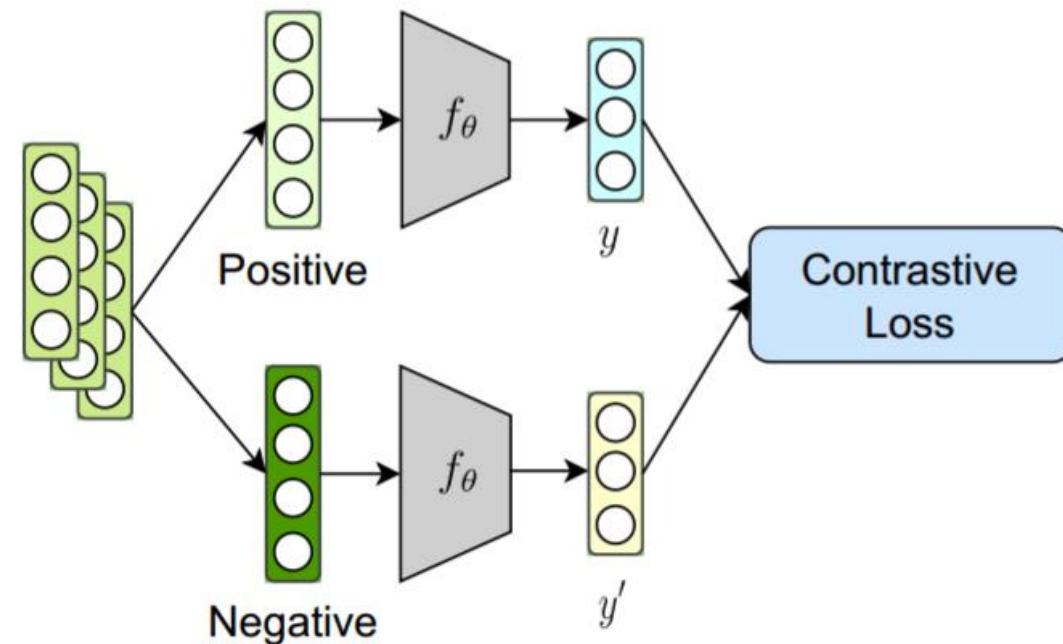
➤ **Target:** distinguishing the unique and common characteristics of the multimodal features to improve the performance and generalization of MRS

➤ **Taxonomy:**

- Disentangled Representation Learning and Contrastive Learning



(a) Disentangled Representation Learning



(b) Contrastive Learning

➤ Disentangled Representation Learning (DRL)

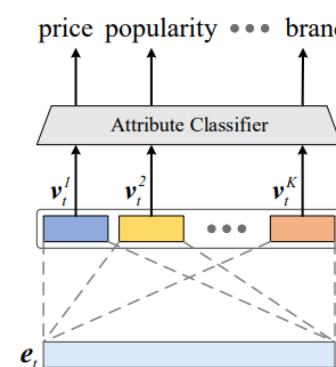
- introducing decomposition learning techniques to **dig out the meticulous factors** in user preference

➤ AD-DRL (MM'24)

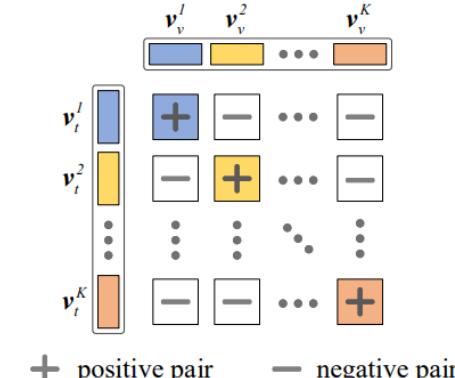
- Low-level Arrtribute-driven DRL:** predicts attributes by **raw modality features** (intra- and inter-disentanglement)
- High-level Arrtribute-driven DRL:** predicts attributes by **fused modality features**

Low-level

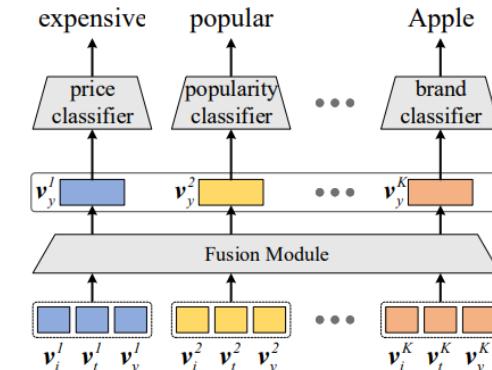
High-level



(a) High-level attribute-driven disentangled representation learning (Intra)



(b) High-level attribute-driven disentangled representation learning (Inter)



(c) Low-level attribute-driven disentangled representation learning

Feature Enhancement



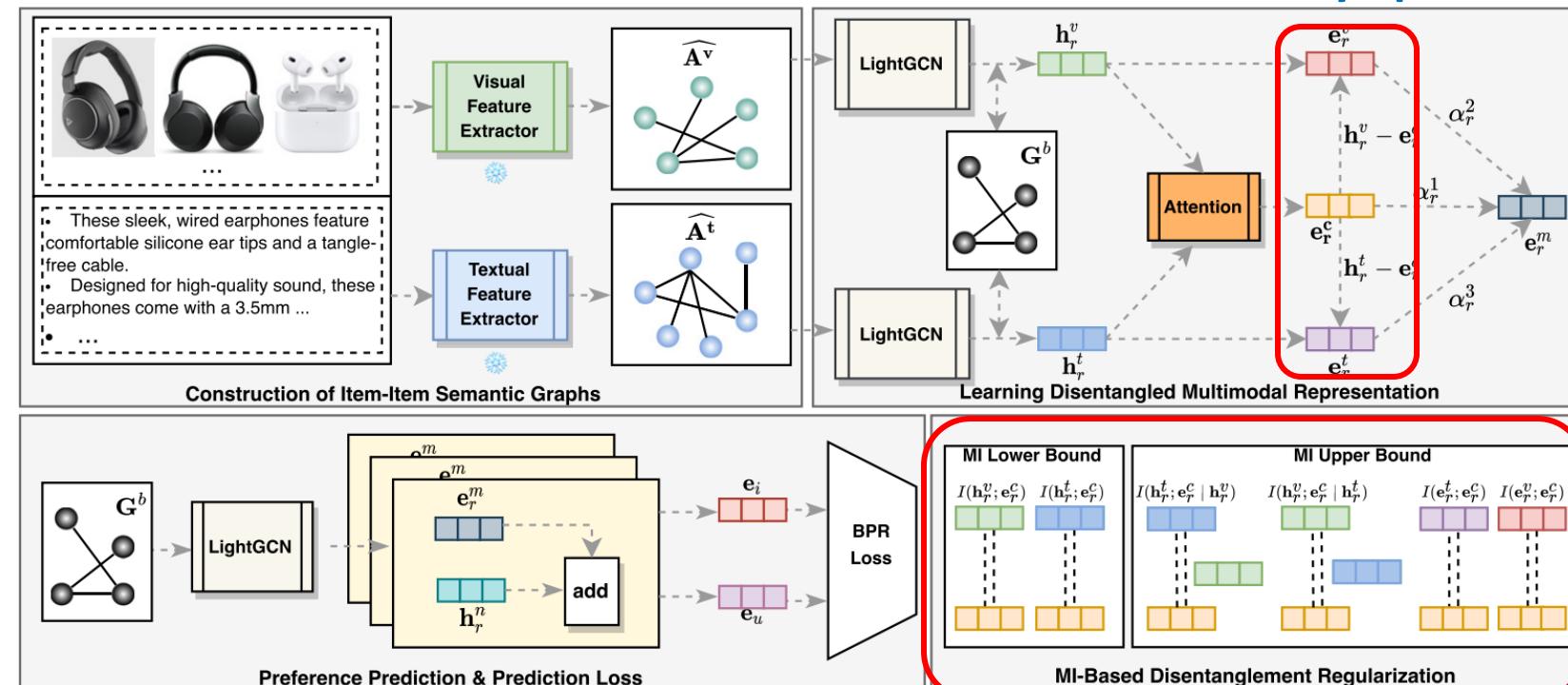
➤ Contrastive Learning (CL)

- minimizing the distance between modalities of one item and maximizing the distance between the modalities of different ones

Modality-shared
Modality-specific

➤ CMDL (TOIS'25)

- Disentangled Multimodal Representation:** modality-invariant part and modality-specific part
- MI-based Disentanglement Regularization:** extended contrastive loss

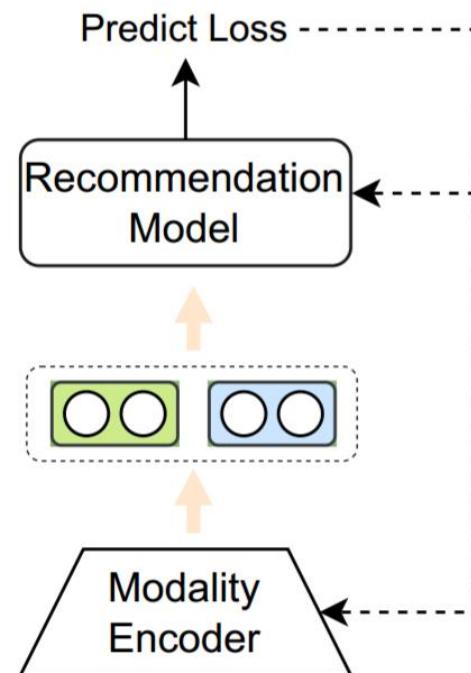


$$\text{Extended CL Loss: } \mathcal{L}^{fub}(X, Y) = \mathbb{E}_{(x,y) \sim p(x,y)} [\hat{f}(x, y)] - \mathbb{E}_{x \sim p(x)} \mathbb{E}_{y \sim p(y)} [\hat{f}(x, y^-)],$$

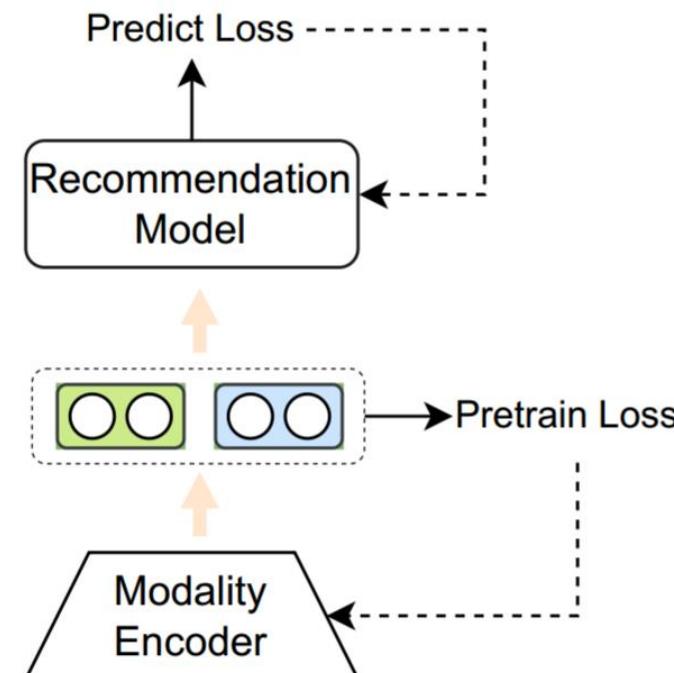
➤ **Target:** meeting the computational requirements for training MRS, including the multimodal encoders and RS model

➤ **Taxonomy:**

- **End-to-end Training and Two-step Training**



(a) End-to-end Training



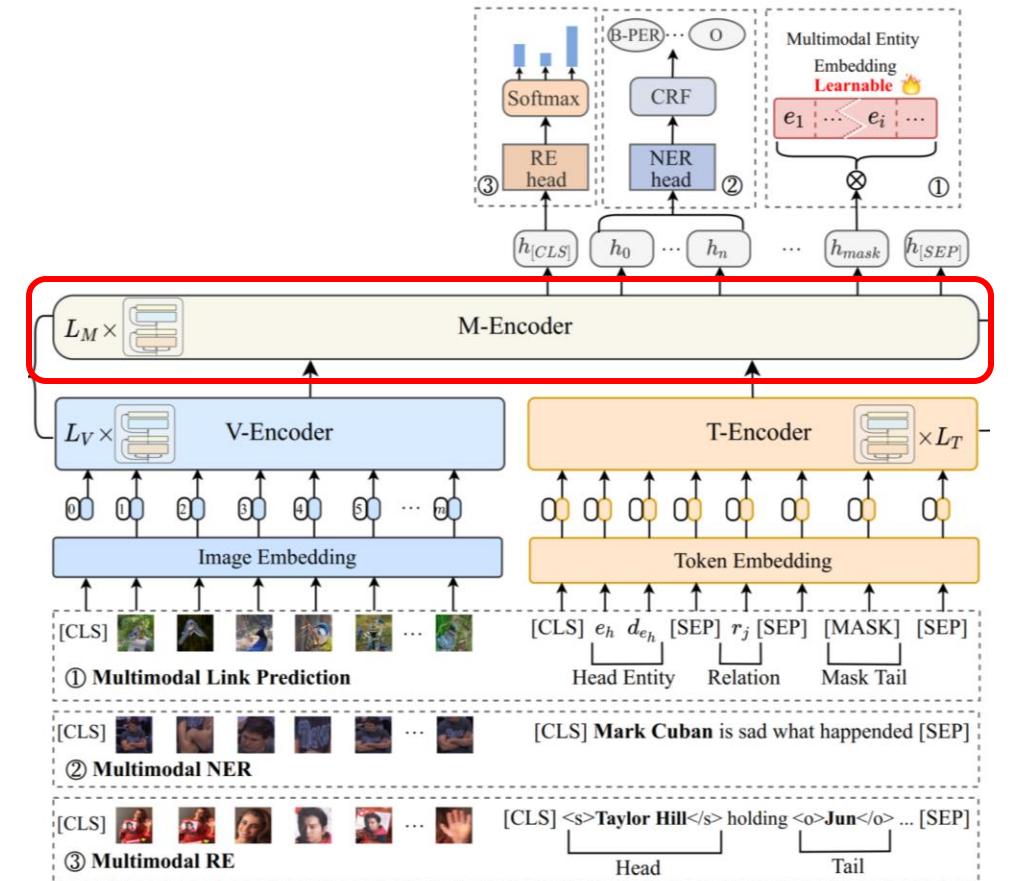
(b) Two-step Training

➤ End-to-end Training

- Training the RS model and modality encoders together, making the encoders better adaptable to RS tasks

➤ MKGformer (SIGIR'22)

- **Modality-specific Parameters:** each modality maintain their own layers, i.e., V-Encoder and T-Encoder
- **Modality-shared Parameters:** both modalities share several layers, i.e., M-Encoder

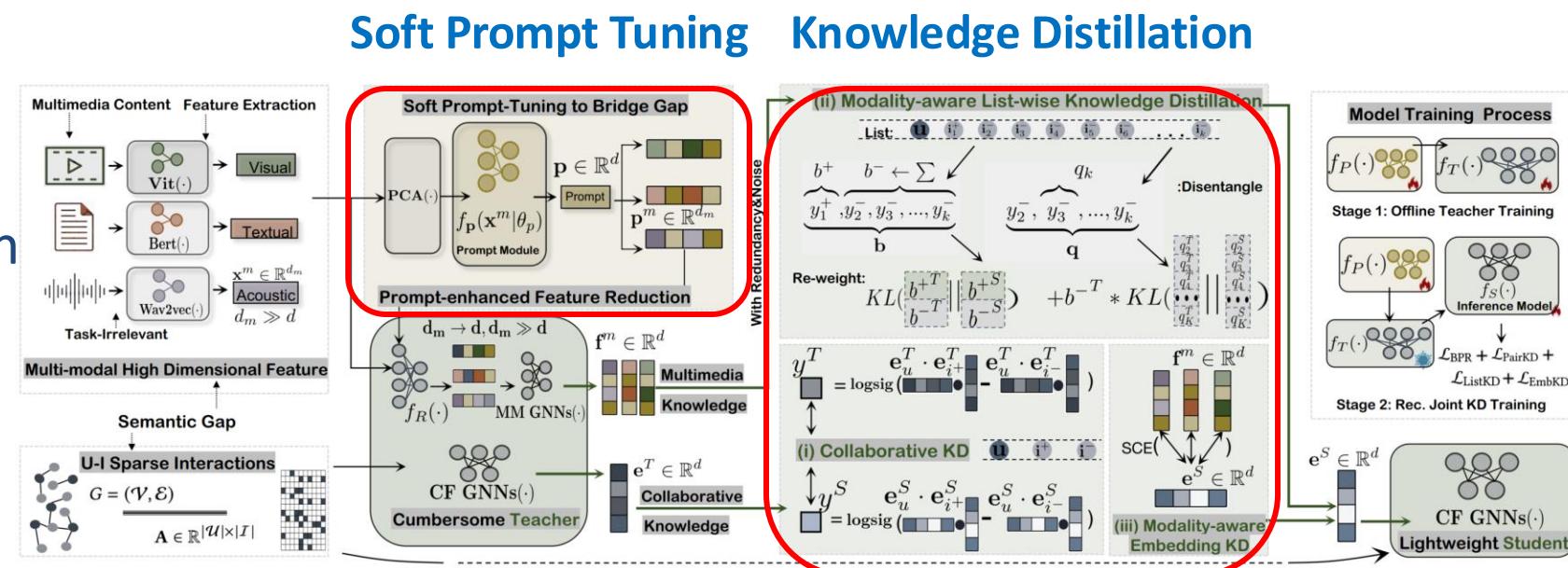


➤ Two-step Training

- Training the RS model and modality encoders separately, making the training process more efficiently

➤ PromptMM (WWW'24)

- **Soft Prompt Tuning:** train the modality adapter for better alignment
- **List-wise Knowledge Distillation:** distill the multimodal knowledge to ID embedding



➤ LLM for Multimodal Recommendation

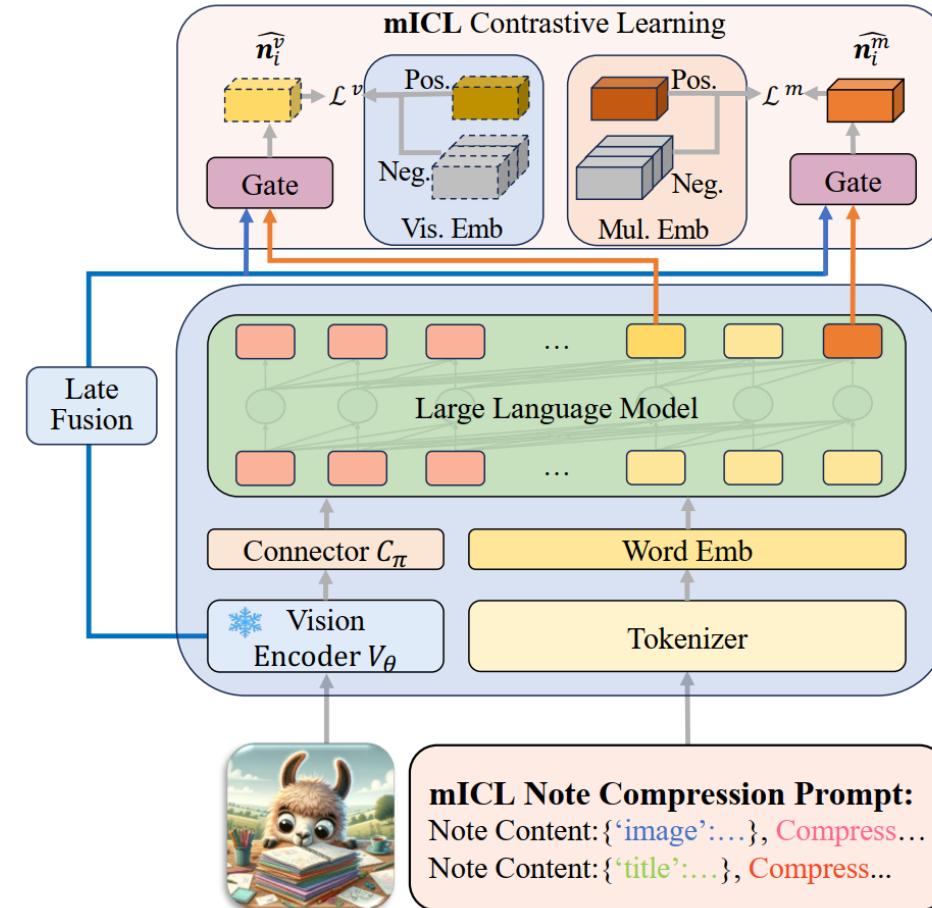
Model	LLM Type	Modality	RS Model
Rec-GPT4V (arXiv'24)	MLLM	Image→Text	LLM
MLLM-MSR (arXiv'24)	MLLM	Image→Text	LLM
Bundle-LLM (arXiv'24)	LLM	Image→Emb	LLM
QARM (arXiv'24)	MLLM	Image→Emb	Conventional RS
NoteLLM-2 (KDD'25)	LLM	Image→Emb	Conventional RS
UniMP (ICLR'25)	LLM	Image→Emb	LLM
UTGRec (arXiv'25)	MLLM	Image→Semantic ID	LLM

➤ Multimodal Large Language Model

- Utilizing the powerful **understanding abilities of MLLM**

➤ NoteLLM-2 (KDD'25)

- Multimodal In-context Learning (mICL):** separates multimodal content into visual and textual components, subsequently compressing the content into **two modality-compressed words**



Applications and Datasets



Data	Field	Modality	Scale	Link
Tiktok	Micro-video	V,T,M,A	726K+	https://paperswithcode.com/dataset/tiktok-dataset
Kwai	Micro-video	V,T,M	1 million+	https://zenodo.org/record/4023390#.Y9YZ6XZBw7c
Movielens + IMDB	Movie	V,T	100k~25m	https://grouplens.org/datasets/movielens/
Douban	Movie,Book,Music	V,T	1 million+	https://github.com/FengZhu-Joey/GA-DTCDR/tree/main/Data
Yelp	POI	V,T,POI	1 million+	https://www.yelp.com/dataset
Amazon	E-commerce	V,T	100 million+	https://cseweb.ucsd.edu/jmcauley/datasets.html#amazon_reviews
Book-Crossings	Book	V,T	1 million+	http://www2.informatik.uni-freiburg.de/cziegler/BX/
Amazon Books	Book	V,T	3 million	https://jmcauley.ucsd.edu/data/amazon/
Amazon Fashion	Fashion	V,T	1 million	https://jmcauley.ucsd.edu/data/amazon/
POG	Fashion	V,T	1 million+	https://drive.google.com/drive/folders/1xFdx5xuNXHGsUVG2VlOhFTXf9S7G5veq
Tianmao	Fashion	V,T	8 million+	https://tianchi.aliyun.com/dataset/43
Taobao	Fashion	V,T	1 million+	https://tianchi.aliyun.com/dataset/52
Tianchi News	News	T	3 million+	https://tianchi.aliyun.com/competition/entrance/531842/introduction
MIND	News	V,T	15 million+	https://msnews.github.io/
Last.FM	Music	V,T,A	186 k+	https://www.heywhale.com/mw/dataset/5cfe0526e727f8002c36b9d9/content
MSD	Music	T,A	48 million+	http://millionsongdataset.com/challenge/

¹ 'V', 'T', 'M', 'A' indicate the visual data, textual data, video data and acoustic data, respectively.

➤ A Universal Solution

- Designing a universal solution with the combinations of the techniques mentioned before

➤ Model Interpretability

- Making the recommended items from MRS interpretable

➤ Computational Complexity

- Shrinking the computational cost and time required by MRS, due to parameterized modality encoder

➤ Privacy

- Protecting user's privacy under condition of affluent multimodal information

➤ Challenges:

Raw feature representation, feature interaction, recommendation

➤ Methodology:

Modality encoders, feature interaction, feature enhancement, model optimization

[CSUR'24] <https://arxiv.org/abs/2302.03883>

Multimodal Recommender Systems: A Survey

QIDONG LIU*, Xi'an Jiaotong University & City University of Hong Kong, China

JIAXI HU*, City University of Hong Kong, China

YUTIAN XIAO*, City University of Hong Kong, China

XIANGYU ZHAO[†], City University of Hong Kong, China

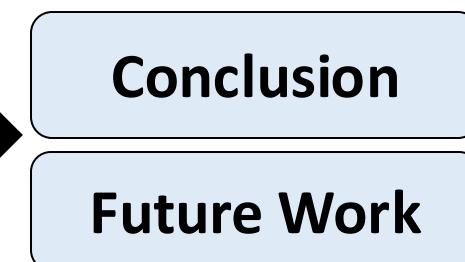
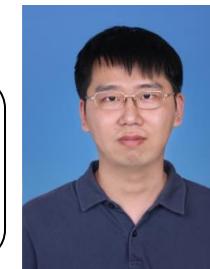
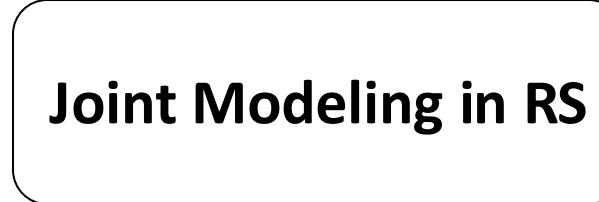
JINGTONG GAO, City University of Hong Kong, China

WANYU WANG, City University of Hong Kong, China

QING LI, The Hong Kong Polytechnic University, China

JILIANG TANG, Michigan State University, USA

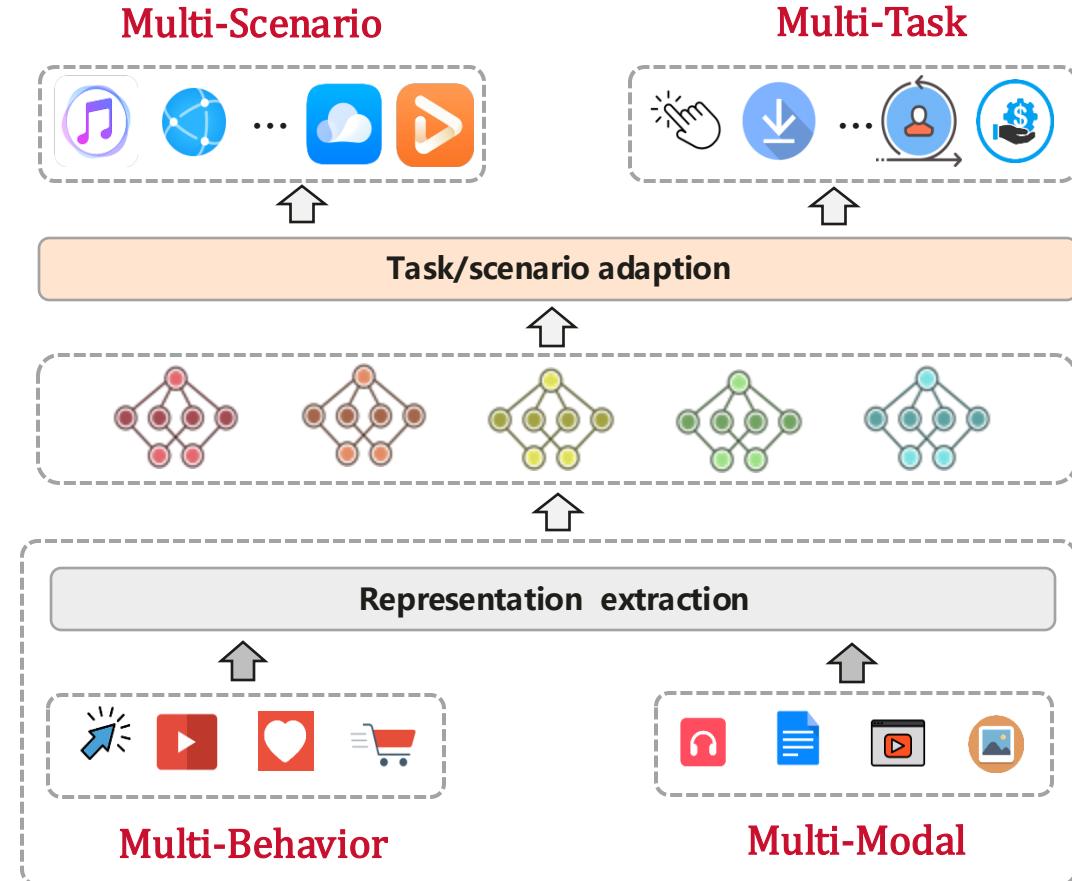




Yichao Wang

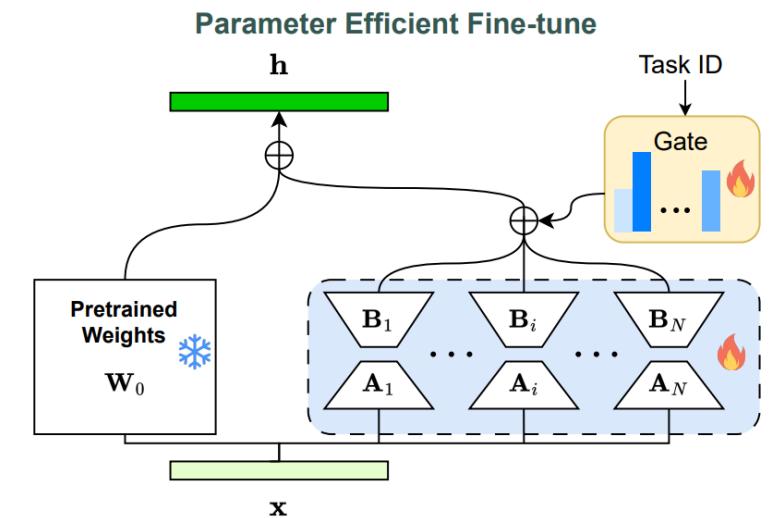
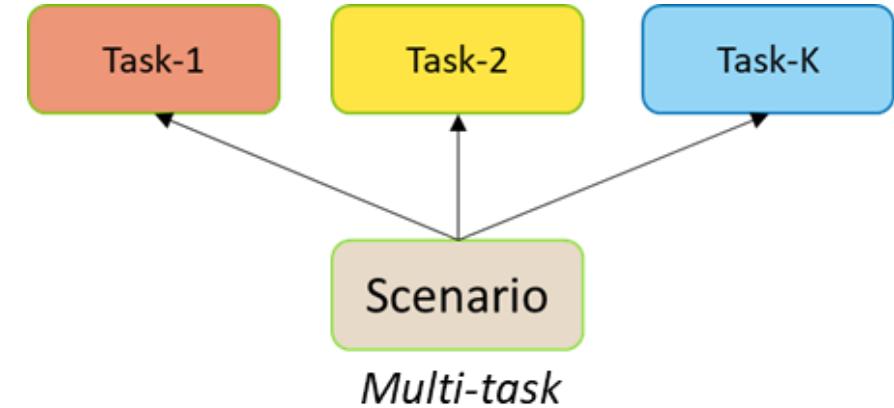
Conclusion

- 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**
- Introducing open-world knowledge from **large language models**



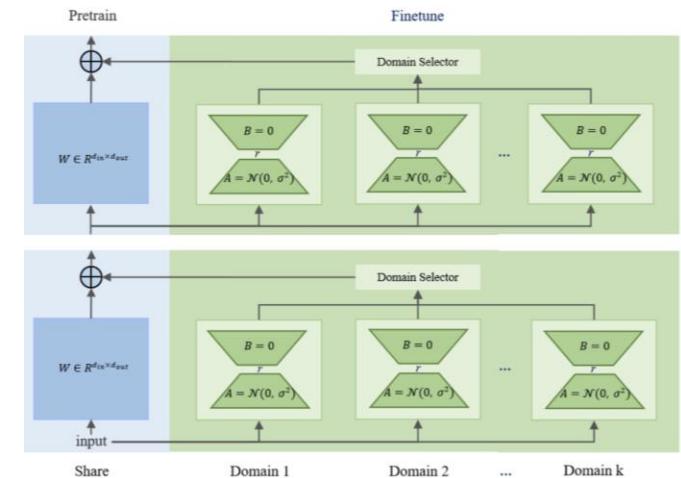
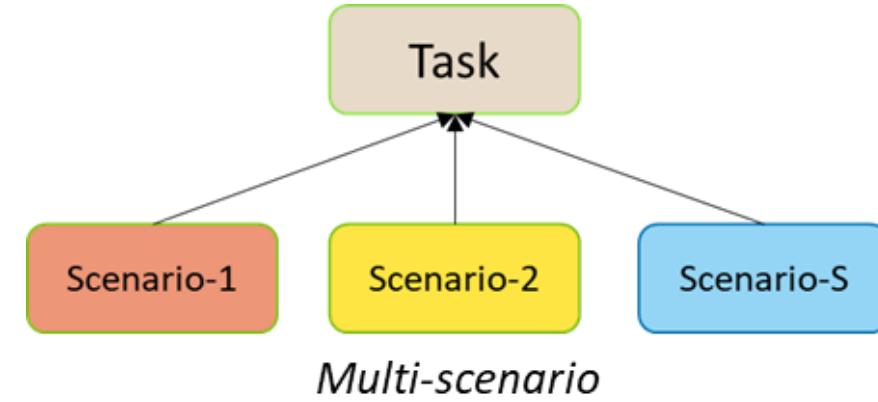
➤ Multi-Task Recommendation

- Task relation:
Parallel, Cascaded, Auxiliary with Main
- Methodology:
Parameter Sharing, Optimization, Training Mechanism
- Trends:
Using LLM Modeling the similarities and differences across tasks with MoE-LoRA to enable task generalization.
e.g., MOELoRA
- Future Direction:
Mitigating negative transfer with LLM's world knowledge
- Take Away:
Designing prompt for different tasks and using LLM backbone as recommender systems



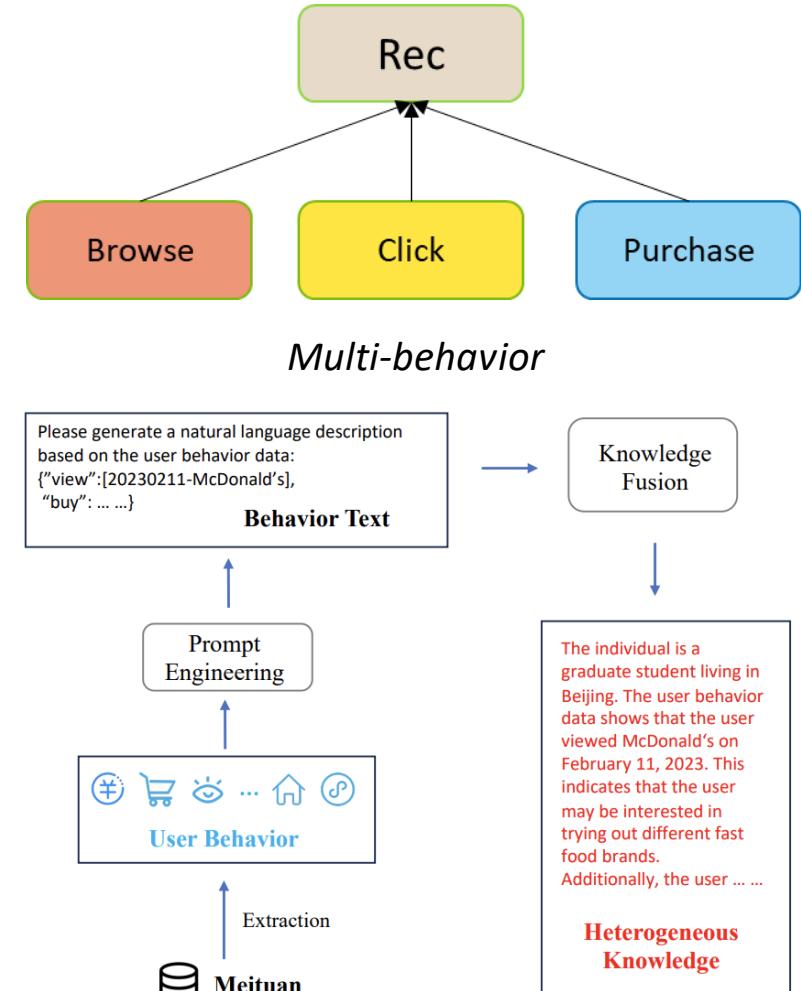
➤ Multi-Scenario Recommendation

- Modeling methods:
shared-specific network paradigm, and dynamic weight paradigm.
- Trends:
Incorporating Low-Rank Adaptor (LoRA) for domain-specific fine-tuning (e.g., M-LoRA); Incorporating LLM's world knowledge in domain understanding (e.g., LLM4MSR)
- Future Direction:
Efficient M-LoRA architecture for large scale scenarios
Scenario recommendation interpretability with LLMs
- Take Away:
Pre-training backbone recommendation model first, then fine-tuning with LoRA for each scenario.



➤ Multi-Behavior Recommendation

- Behavior :
Macro behaviors, Micro behaviors, Behaviors from different domains or scenarios
- Methods:
RNN, Graph, Transformer
- Trends:
Better modeling architecture; Generative Modeling; Modeling with LLM (e.g., knowledge fusion in HKFR)
- Future Direction:
Fine-grained behavior understanding with LLMs
Behavior debias with LLMs
- Take Away:
Using LLMs to extract heterogeneous knowledge, then fusing these knowledge into user modeling.



Stage 1 : Heterogeneous Knowledge Fusion

➤ Multi-Modal Recommendation

- Methods:

- Modality Encoder, Feature Interaction, Feature Enhancement, Model Optimization

- Trends:

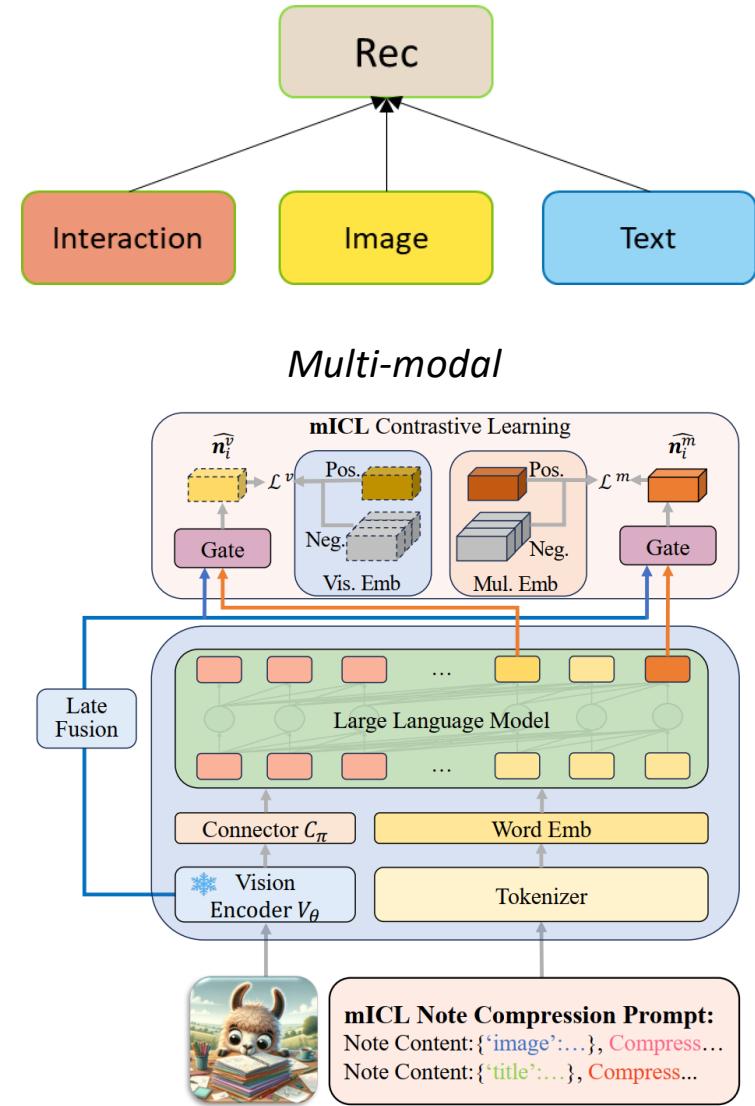
- Utilizing the powerful understanding abilities of MLLM to enhance Multi-Modal Rec (e.g., NoteLLM)

- Future Direction:

- Recommendation representation align with MLLM representation; personalized multi-modal generation recommendation (e.g., image personalized generation)

- Take Away:

- Using MLLM to get embedding for multi-modal data, then fusing these information into generative recommendation or conventional recommender systems.



We are hiring !



Huawei Noah's Ark Lab



**WWW25 Huawei Noah's Ark
Lab Chat Group**



**AML Lab
CityU**

Tutorial Slides

[https://github.com/Applied-Machine-Learning-Lab/Joint-Modeling-in-Deep-Recommender-
Systems-WWW2025/blob/main/pdf/jointmodeling-www2025.pdf](https://github.com/Applied-Machine-Learning-Lab/Joint-Modeling-in-Deep-Recommender-Systems-WWW2025/blob/main/pdf/jointmodeling-www2025.pdf)

[1] Multi-Task Deep Recommendation Systems: A Survey. <https://arxiv.org/abs/2302.03525>

[2] Scenario-Wise Rec: A Multi-Scenario Recommendation Benchmark. <https://arxiv.org/abs/2412.17374>

[3] Multimodal Recommender Systems: A Survey. <https://arxiv.org/abs/2302.03883>