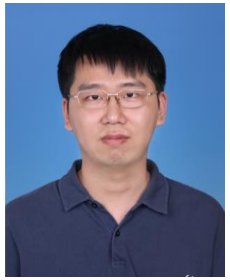
Joint Modeling in Deep Recommender Systems







Pengyue Jia[1] Jingtong Gao[1] Yejing Wang[1] Yuhao Wang[1] Xiaopeng Li[1] Qidong Liu[1] Yichao Wang[2] Bo Chen[2] Huifeng Guo[2] Ruiming Tang[2]

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

CityU AML Lab

Huawei Noah's Ark Lab

Agenda

Introduction

Yejing Wang

Joint Modeling in RS

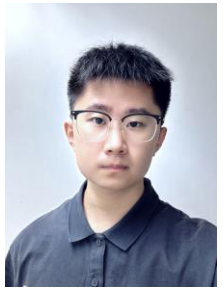
Pengyue Jia

Multi-task Recommendation

Yuhao Wang

Multi-scenario Recommendation

Pengyue Jia

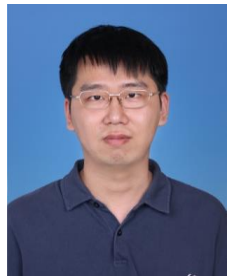


Xiaopeng Li

Multi-behavior Recommendation

Jingtong Gao

Multi-modal Recommendation



Qidong Liu

Conclusion

Future Work

Yichao Wang

Agenda

Introduction

Yejing Wang

Joint Modeling in RS

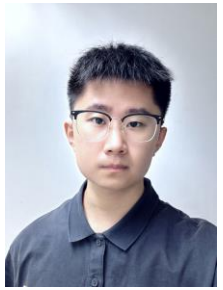
Pengyue Jia

Multi-task Recommendation

Yuhao Wang

Multi-scenario Recommendation

Pengyue Jia

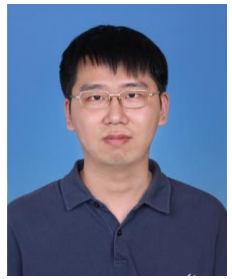


Xiaopeng Li

Multi-behavior Recommendation

Jingtong Gao

Multi-modal Recommendation



Qidong Liu

Conclusion

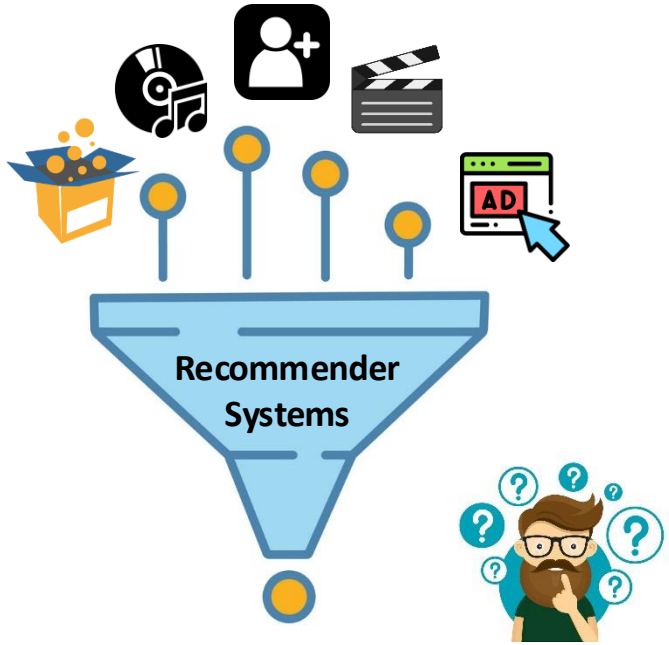
Future Work

Yichao Wang

Recommender Systems

Age of Information Explosion

Information overload



Recommend item X to user

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

Recommender Systems

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

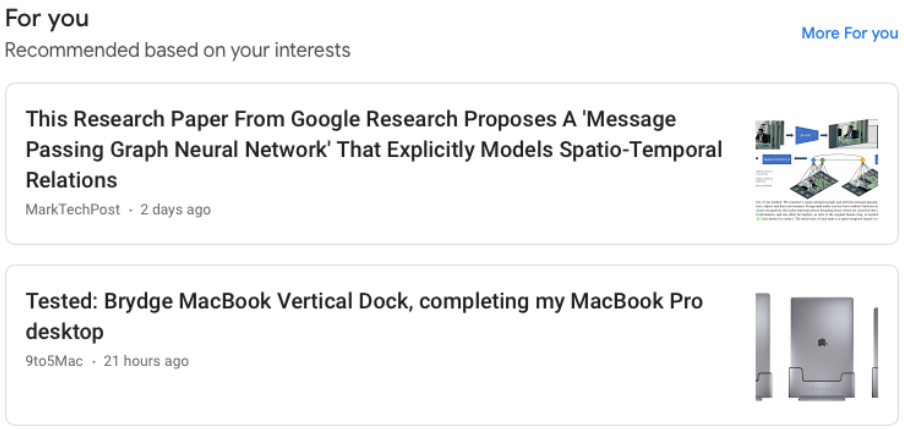
amazon ebay 淘宝网 Taobao.com JD.COM

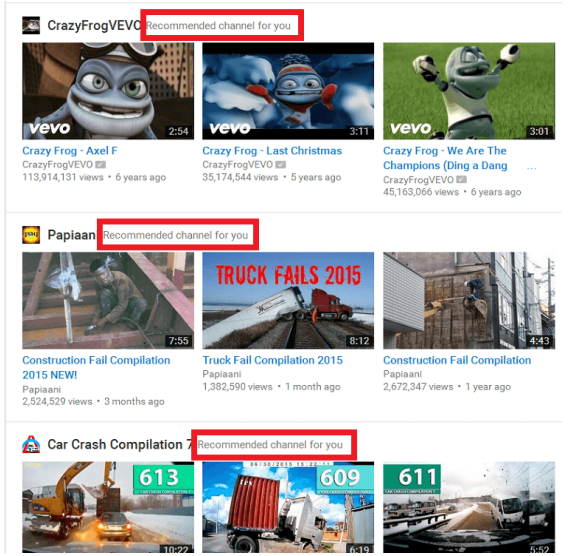
Product Recommendation

Recommender Systems

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

News/Video/Image Recommendation





Recommender Systems

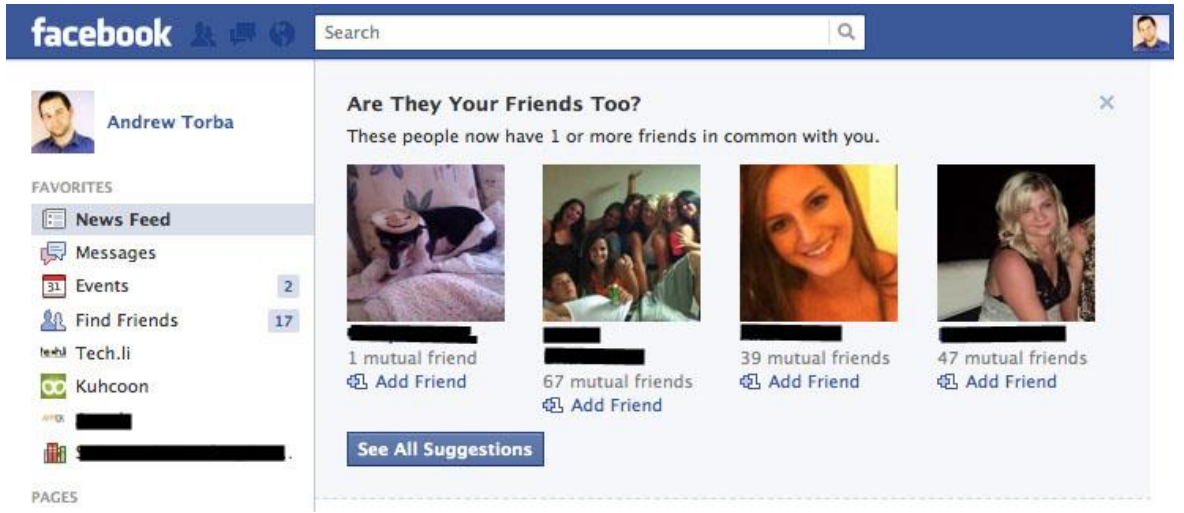
- Recommendation has been widely applied in online services
  - **E-commerce, Content Sharing, Social Networking, etc.**

facebook

新浪微博 weibo.com

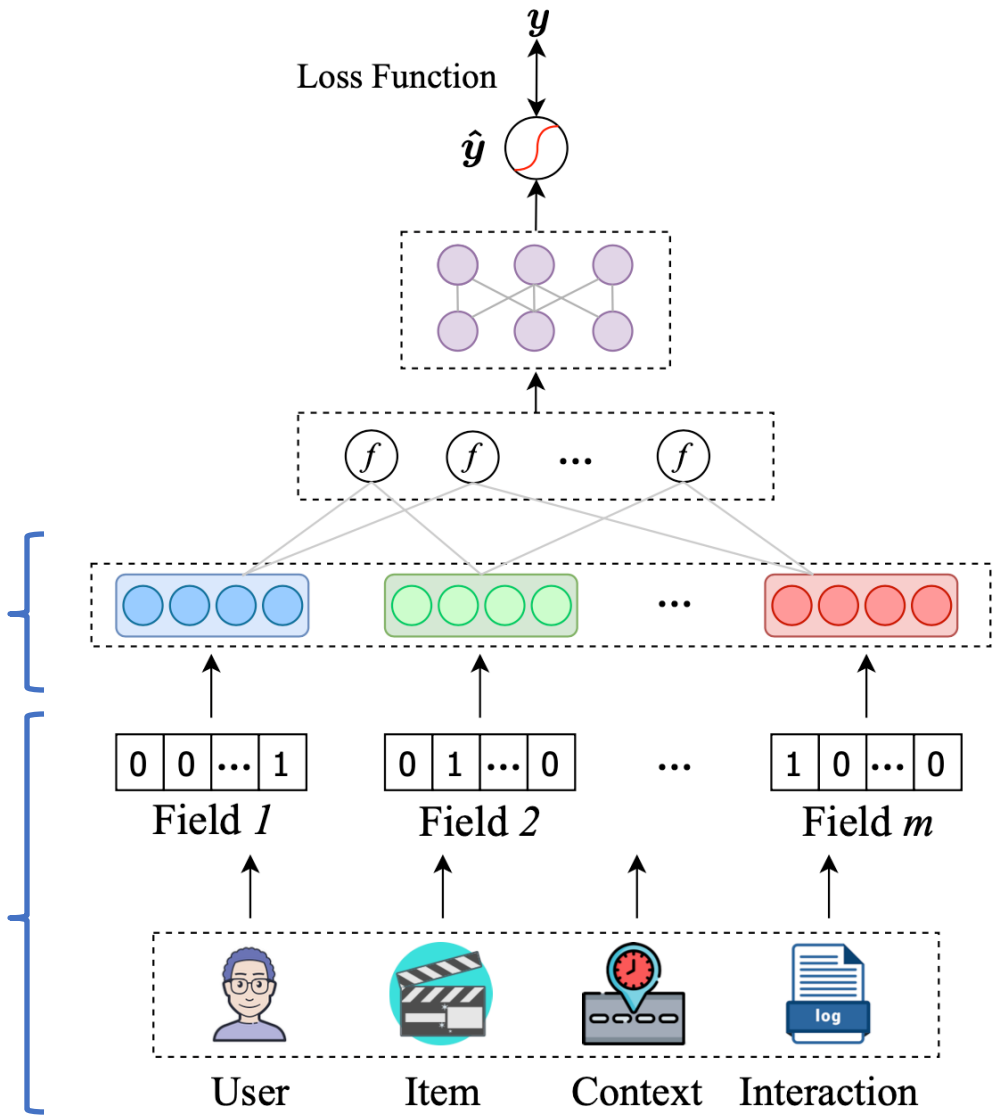
LinkedIn

Friend Recommendation



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

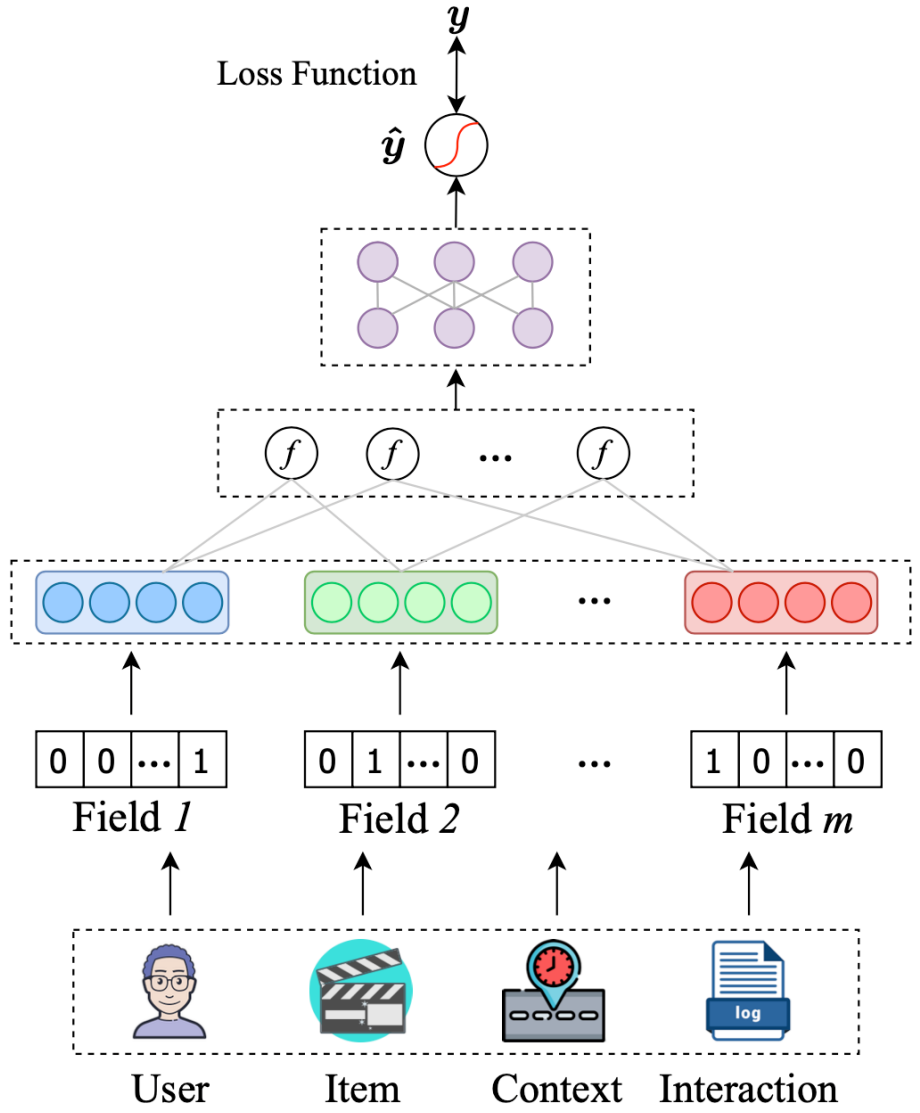
Deep Recommender Architecture

Output Layer

BCE, BPR, MSE

Feature Interaction Layer

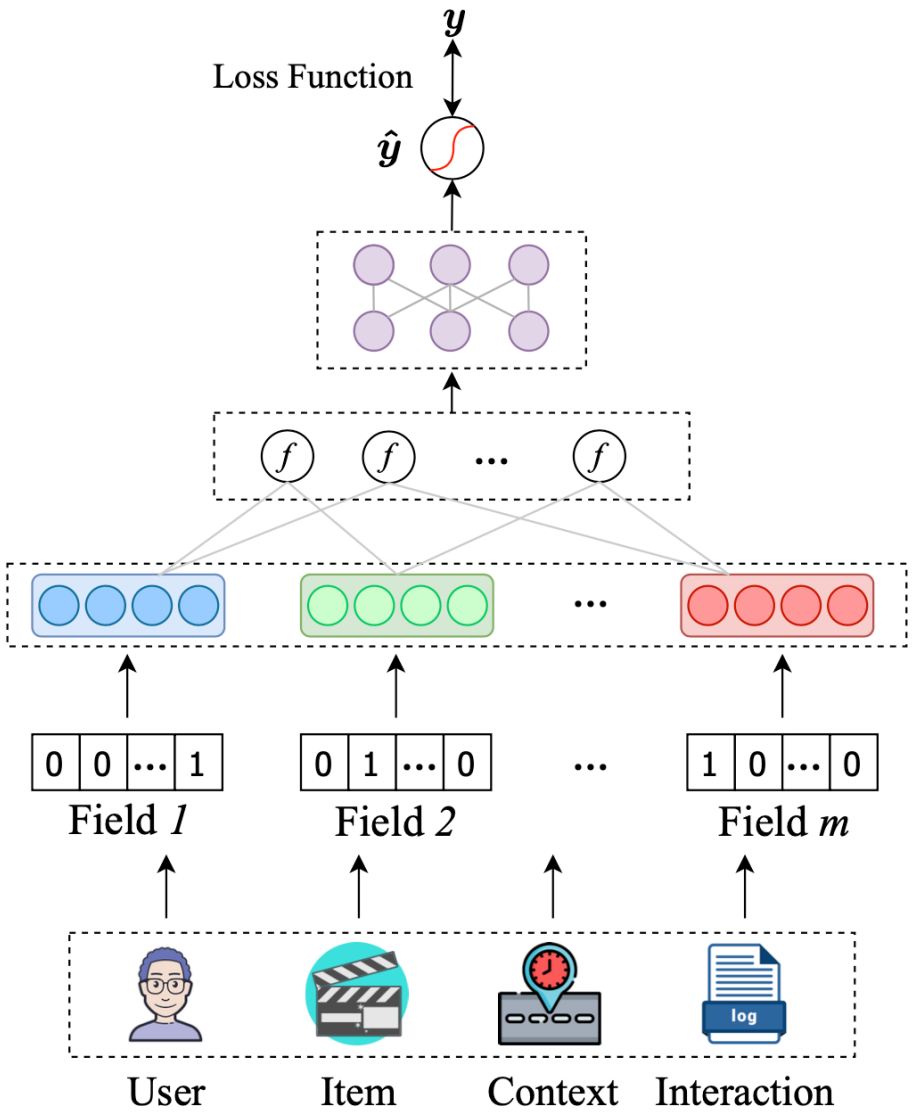
Pooling, convolution, and the number of layers, inner product, outer product, convolution, etc.



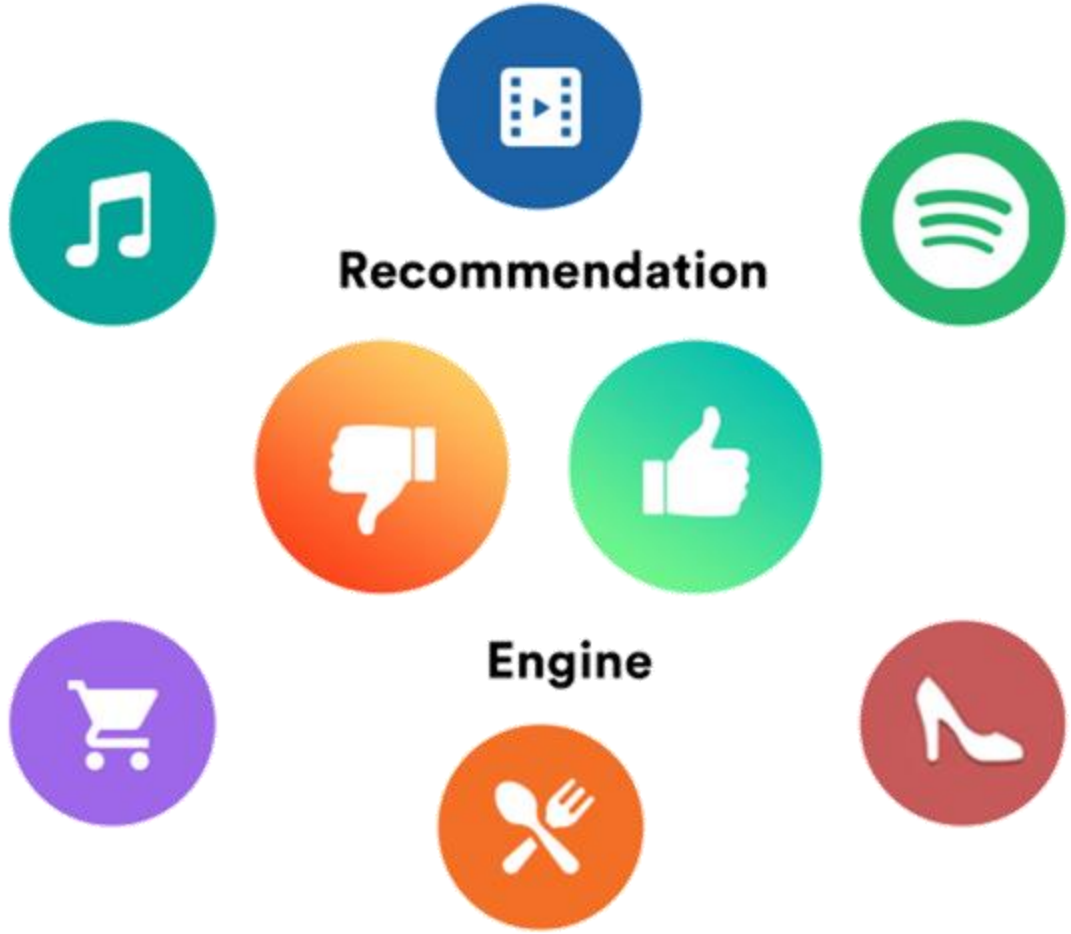
System Design

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

Deep Recommender Architecture



V.S.



Agenda

Introduction

Yejing Wang

Joint Modeling in RS

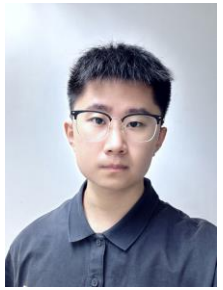
Pengyue Jia

Multi-task Recommendation

Yuhao Wang

Multi-scenario Recommendation

Pengyue Jia

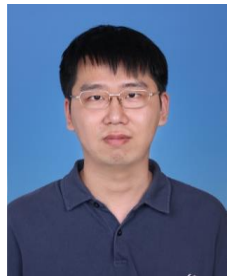


Xiaopeng Li

Multi-behavior Recommendation

Jingtong Gao

Multi-modal Recommendation



Qidong Liu

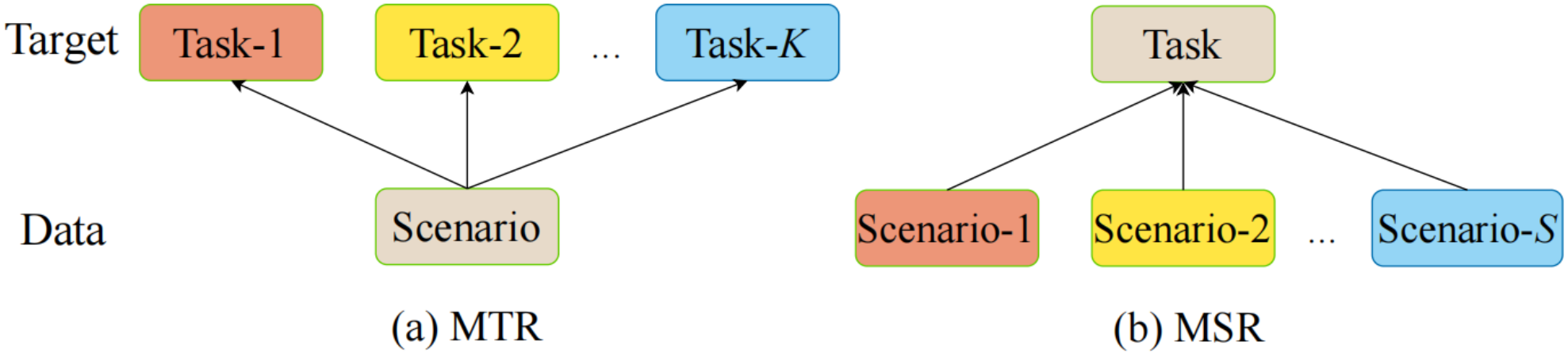
Conclusion

Future Work

Yichao Wang

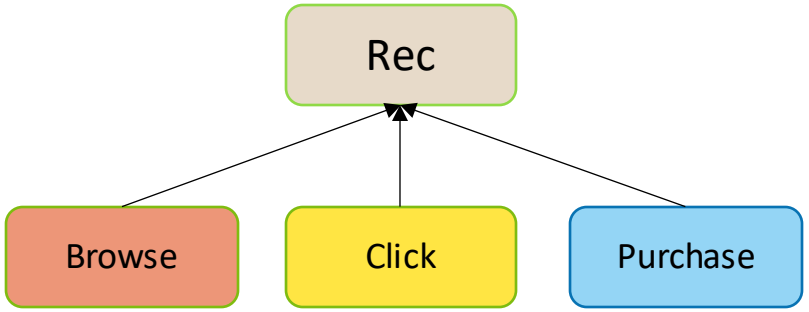
Joint Modeling in Recommendations

- 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)**

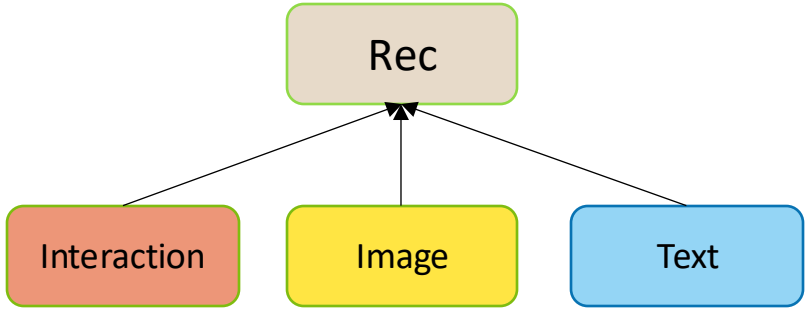


Joint Modeling in Recommendations

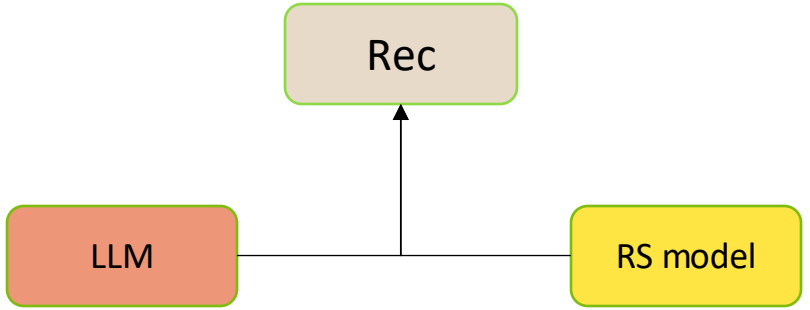
- More joint modeling methods:
  - **Multi-behavior recommendation**
  - **Multi-modal recommendation**
  - **Large language model-based recommendation**



Multi-behavior recommendation



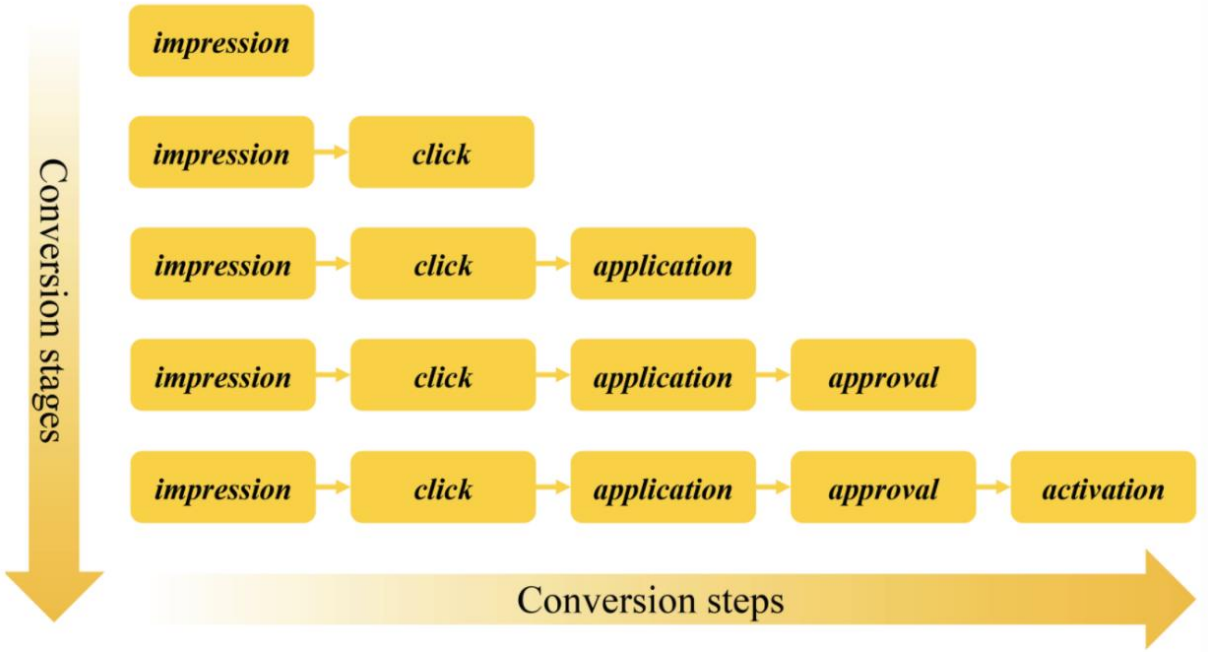
Multi-modal recommendation



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 …**



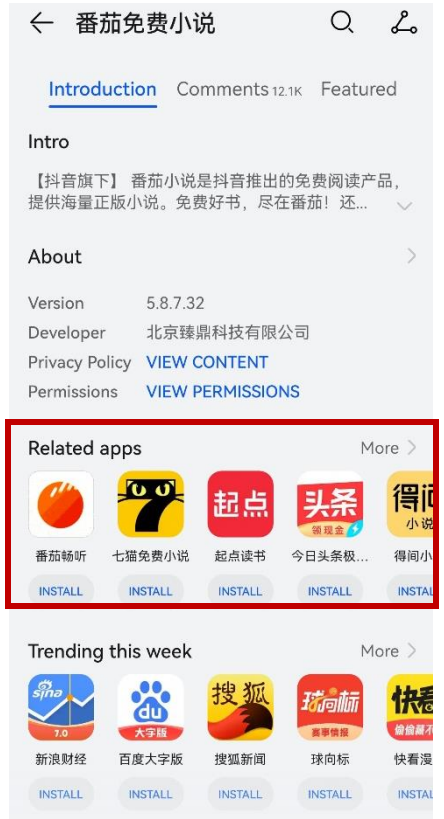
How to extract useful information from other tasks ?

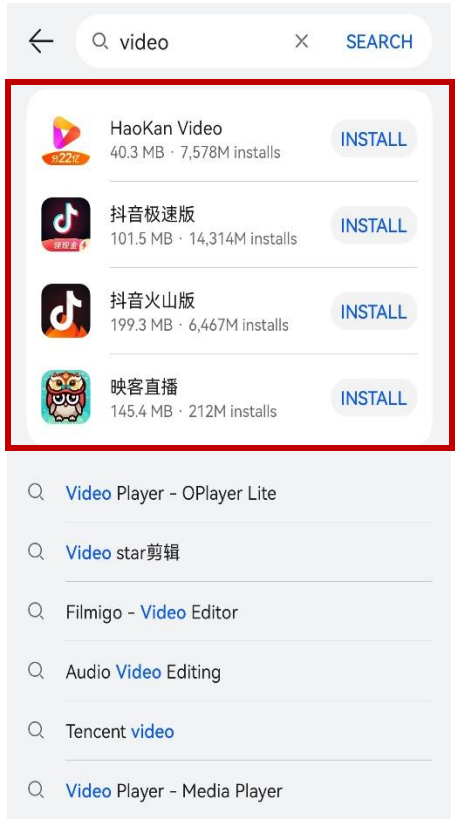
How to capture task dependences and resolve the sparsity issue ?

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

Why Joint Modeling ?

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

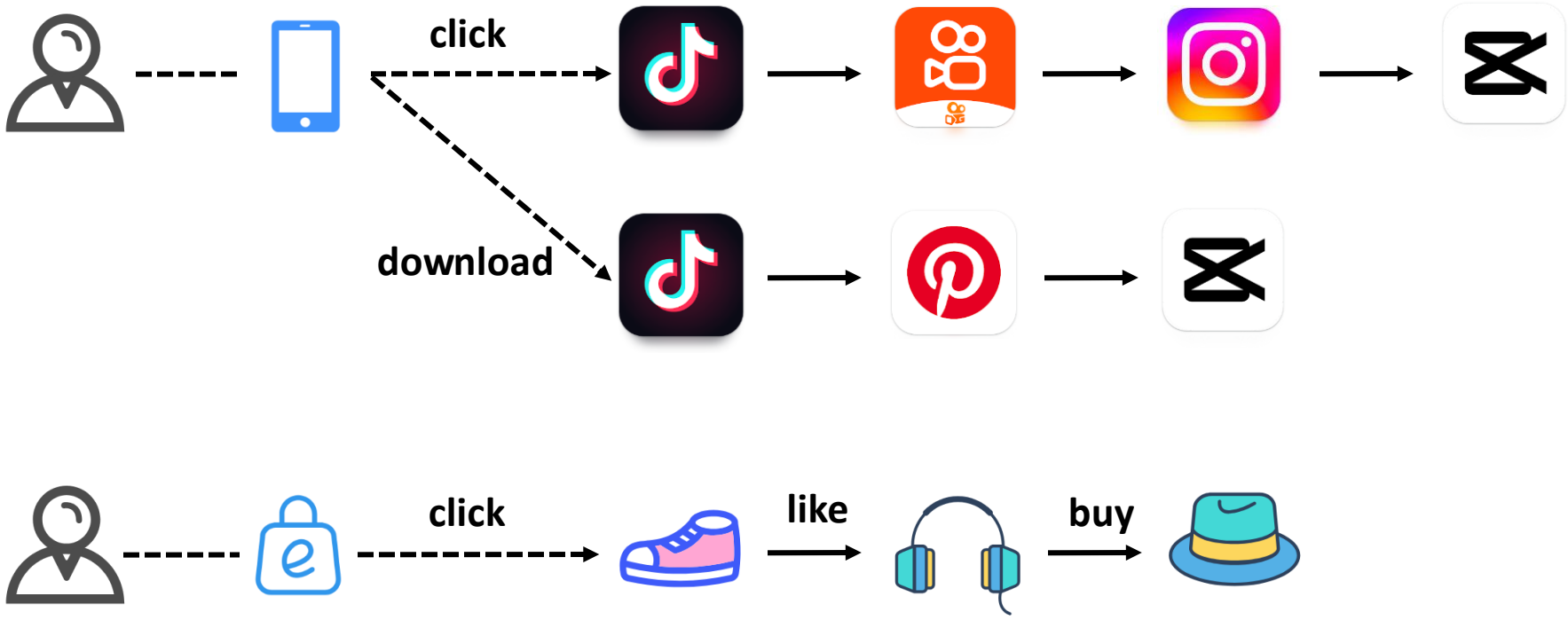




How to extract more comprehensive user portrait from interactions in different scenarios, and make recommendations based on the characteristics of the current scenario ?

Why Joint Modeling ?

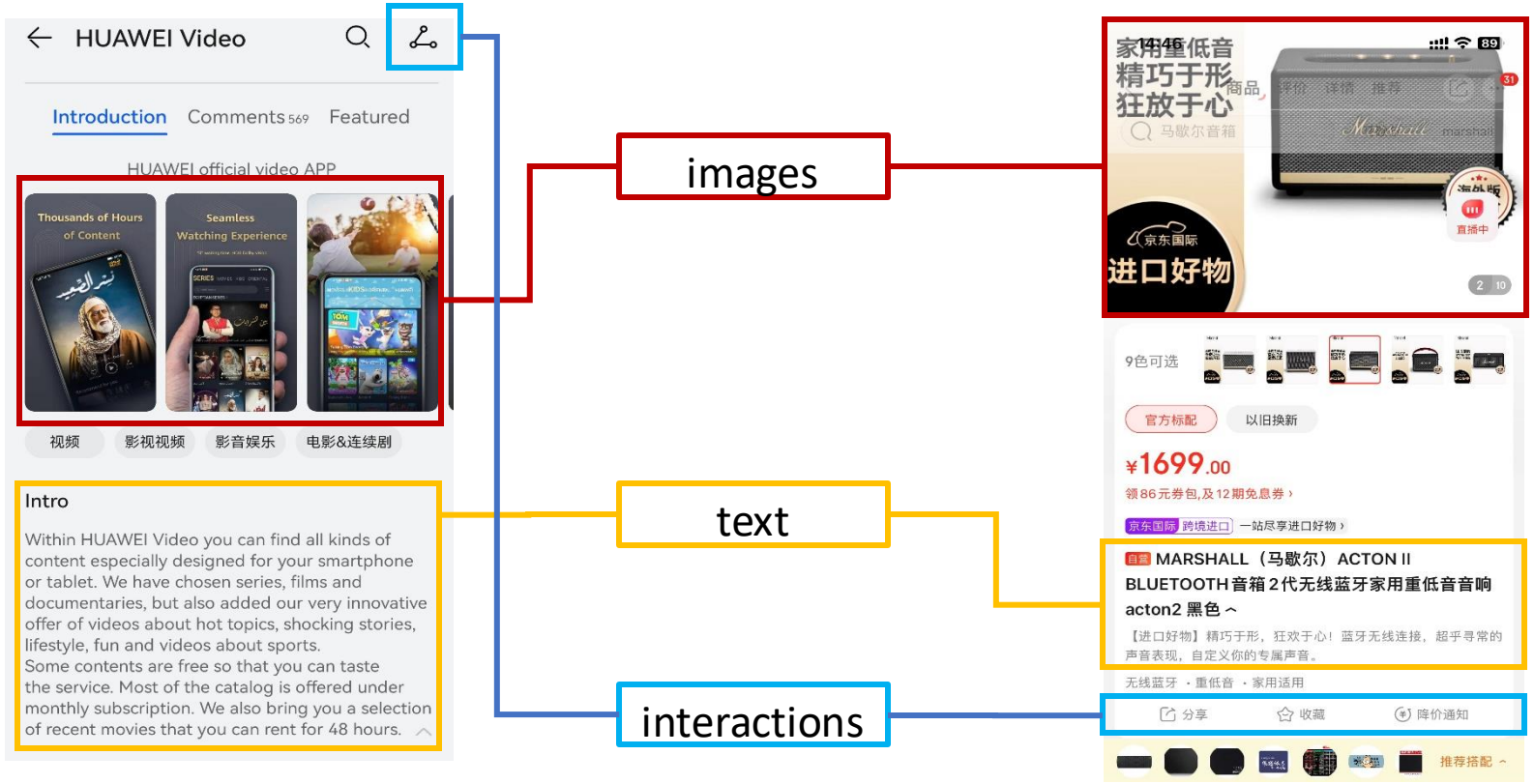
➢Multi-Behavior Modeling: click, download, like, buy



How to learn the relationship between different type of behaviors ?

Why Joint Modeling ?

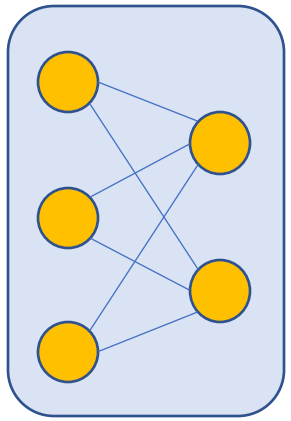
➢Multi-Modal Modeling: user interactions, images, text …



How to extract and align data from different modalities ?

Why Joint Modeling ?

➢Large Language Model-based Recommendation

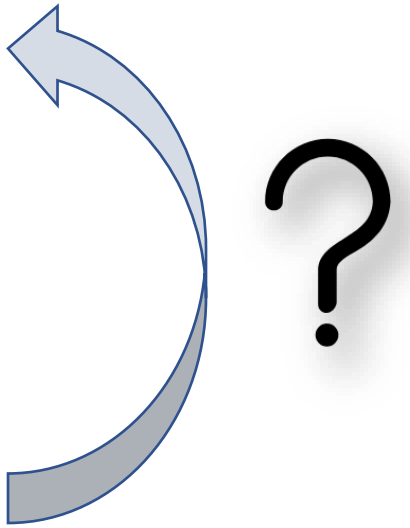


DRS

Trained on labeled data with supervised learning

Collaborative signals

ID-based in-domain collaborative knowledge



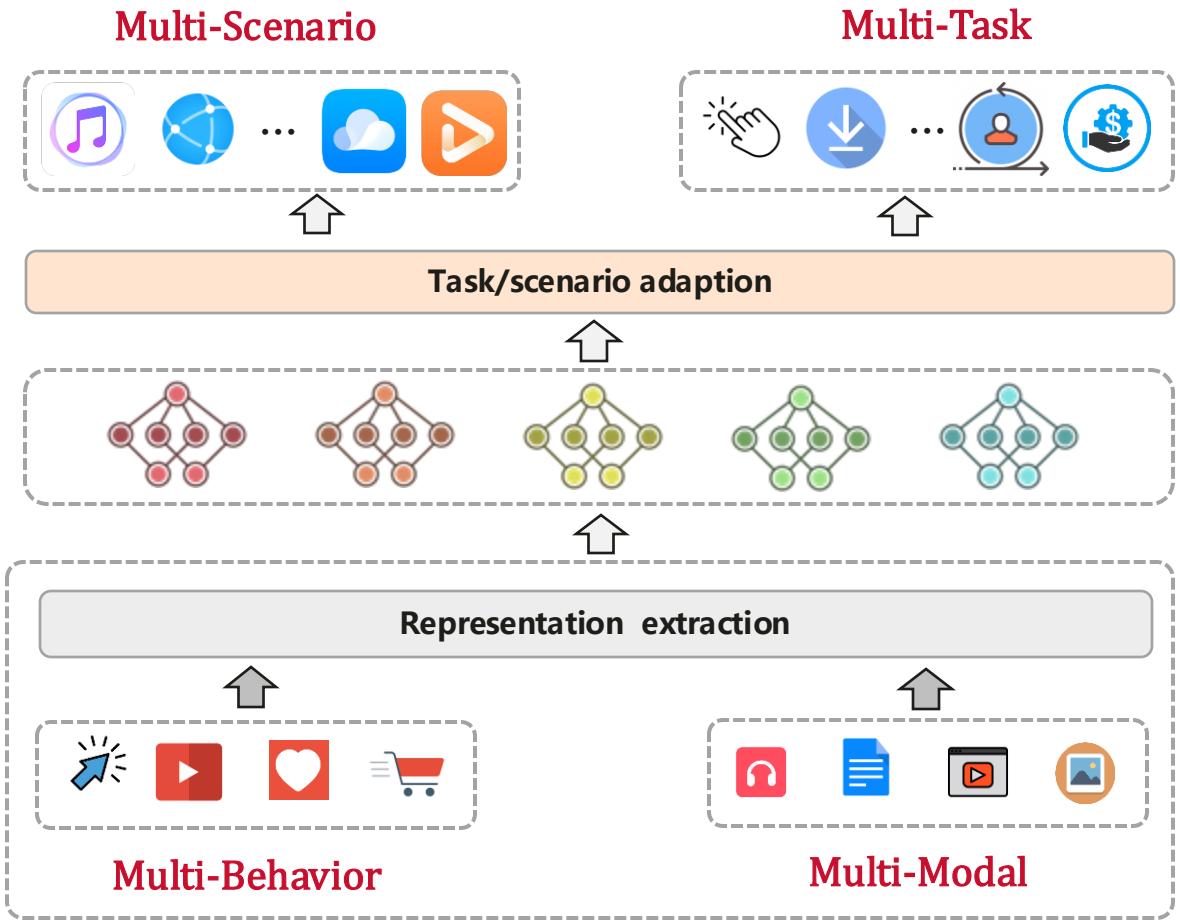
LLM

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

Semantic signals

Generalization, reasoning and open-world knowledge

Relations and Formulations of Joint Modeling



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

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

$$\boldsymbol{E}^{B} = G(\boldsymbol{H}_1, \boldsymbol{H}_2, \dots, \boldsymbol{H}_N)$$ Multi-behavior

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

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

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

Agenda

Introduction

Yejing Wang

Joint Modeling in RS

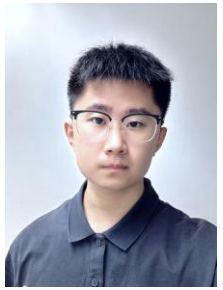
Pengyue Jia

Multi-task Recommendation

Yuhao Wang

Multi-scenario Recommendation

Pengyue Jia

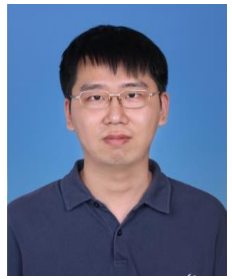


Xiaopeng Li

Multi-behavior Recommendation

Jingtong Gao

Multi-modal Recommendation



Qidong Liu

Conclusion

Future Work

Yichao Wang

MTR & MTDRS

Multi-Task Recommendation (MTR)

Multi-Task Deep Recommender Systems (MTDRS)

➢How

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

➢Why

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

Benefits & Challenges

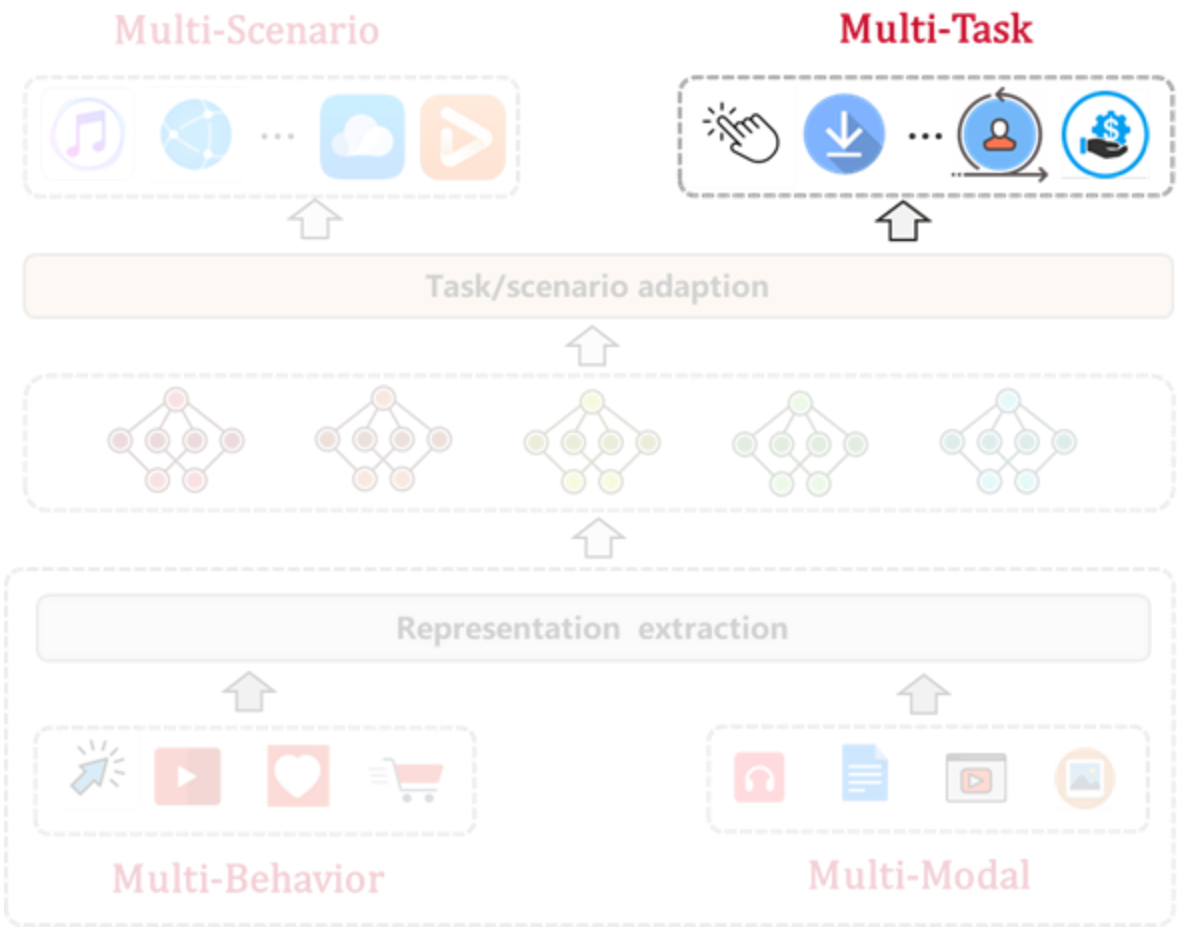
➢ Benefits

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

➢ Challenges

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

Multi-Task Modeling



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

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

$E^{B} = G(H_1, H_2, \ldots, H_N)$ — Multi-behavior

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

$wL(E^{Merge}, \theta^{sh}, \theta^{t}, \theta^{s})$ — Multi-scenario

$wL(E^{Merge}, \theta^{sh}, \theta^{t}, \theta^{s})$ — Multi-task

Formulation

- **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:**

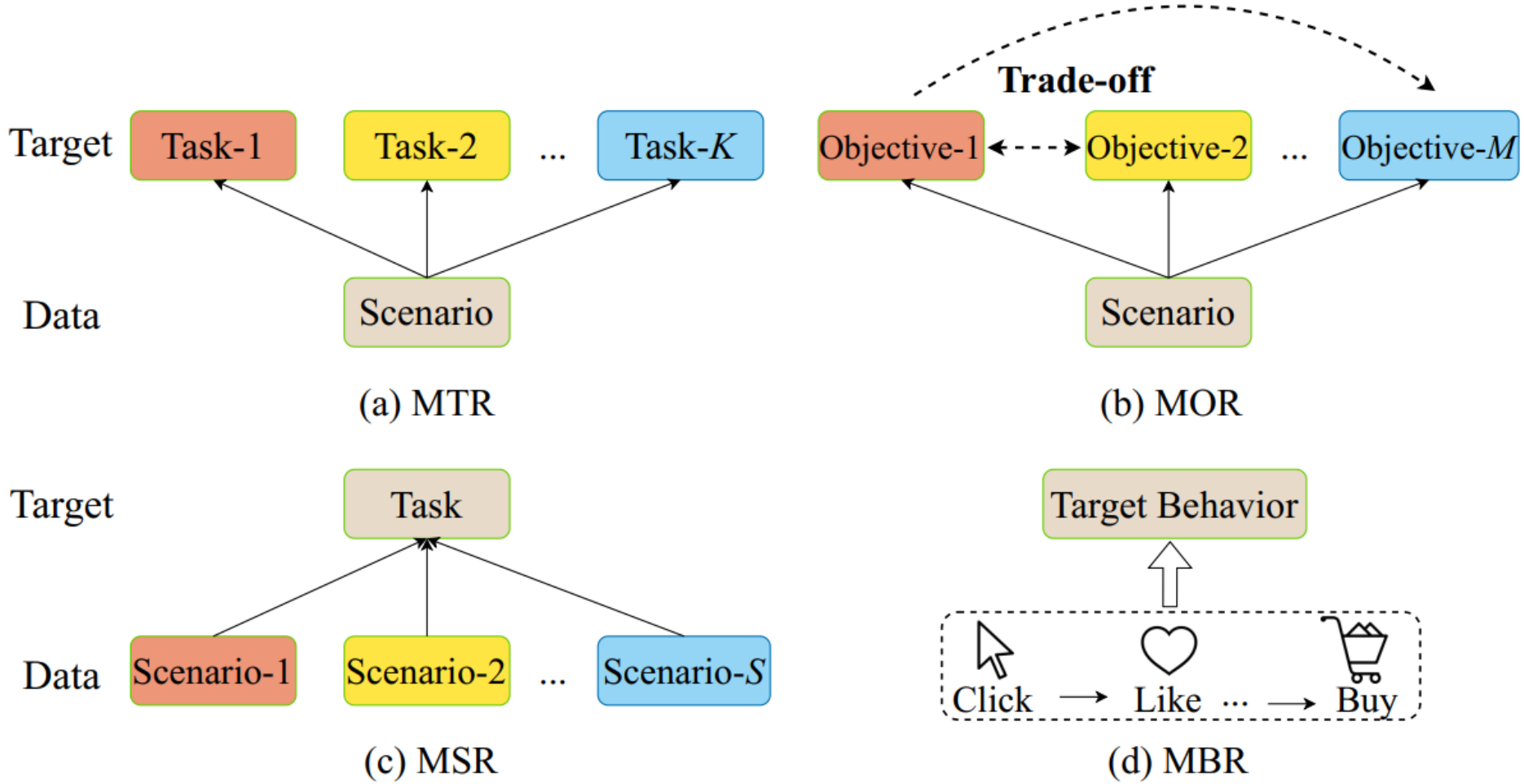
$$\arg\min_{\{\theta^1,\dots,\theta^K\}} \mathcal{L}\left(\theta^s, \theta^1, \cdots, \theta^K\right) = \arg\min_{\{\theta^1,\dots,\theta^K\}} \sum_{k=1}^{K} \omega^k L^k\left(\theta^s, \theta^k\right)$$

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

BCE loss

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

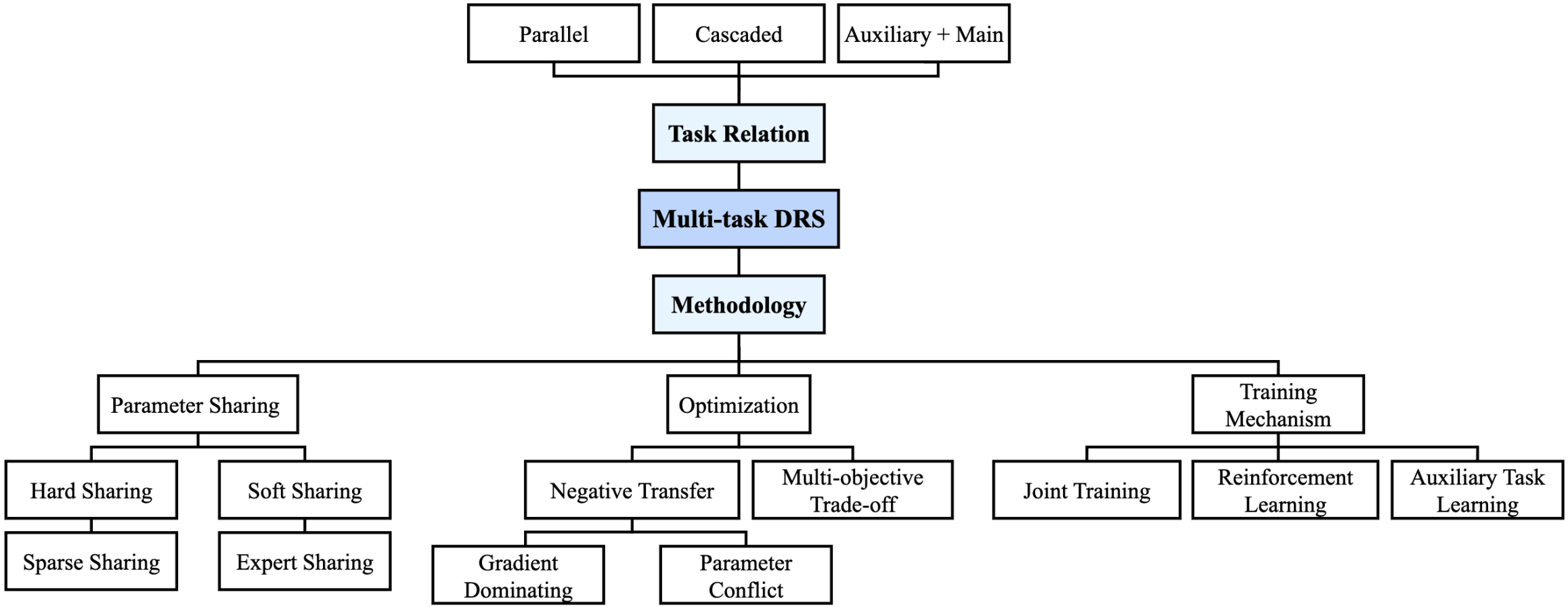
MTR, MOR, MSR, MBR



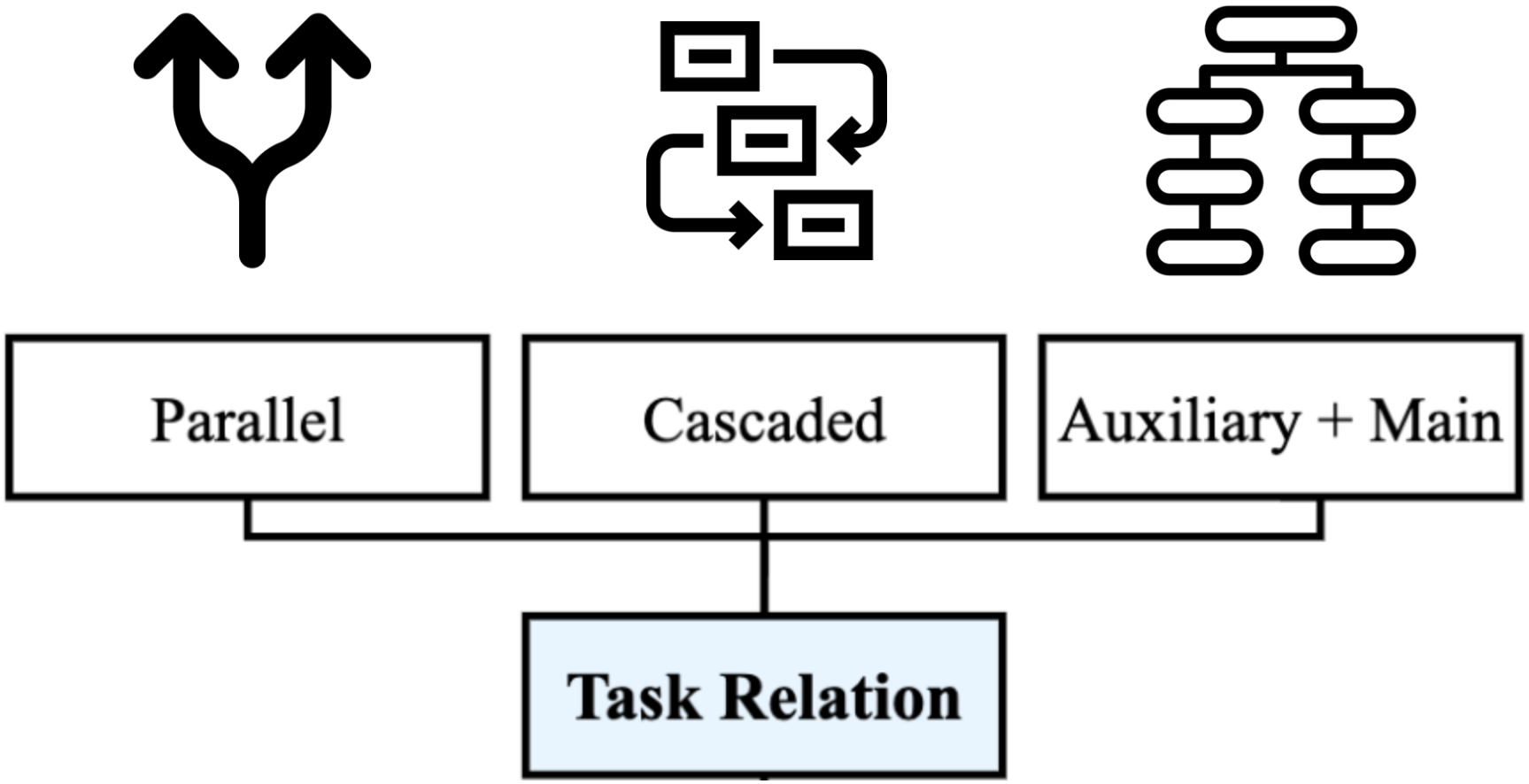
Comparison with CV & NLP

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

Taxonomy


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

Task Relation



Parallel

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

Cascaded

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

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

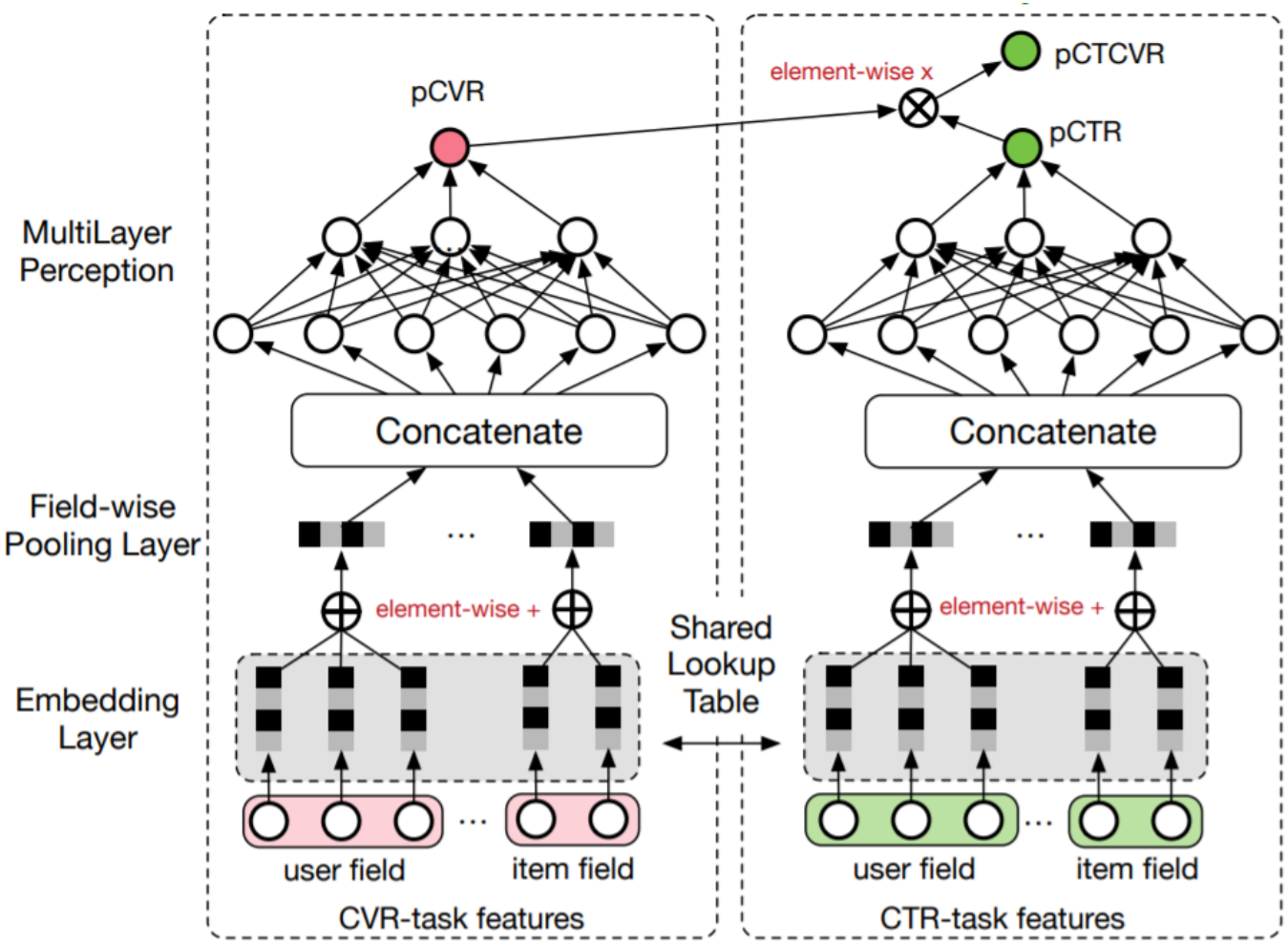
  - ϵ_k: Indicator variable for task k
  - Difference is the probability of the task k not happening while the task $k-1$ is observed

Cascaded

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

SSB: Sample Selection Bias DS: Data Sparsity

ESMM



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

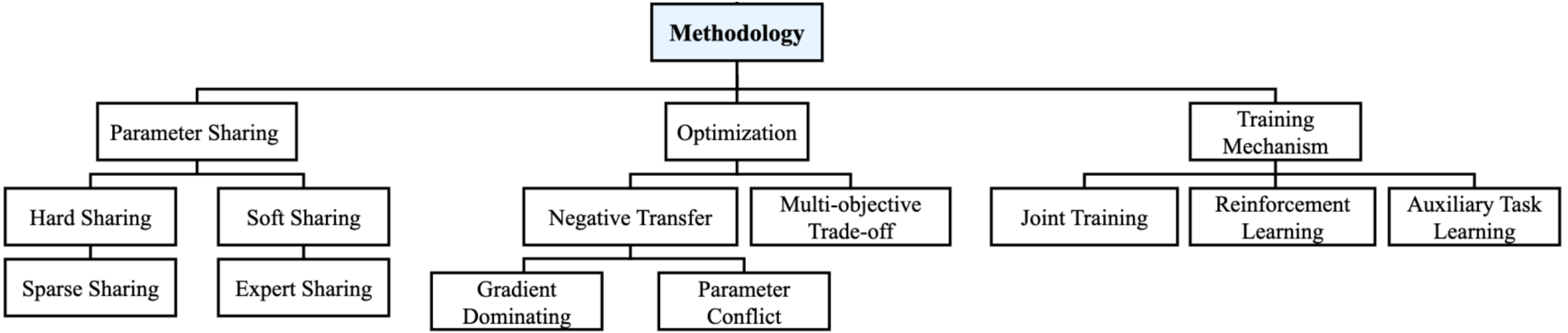
Auxiliary with Main Task

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

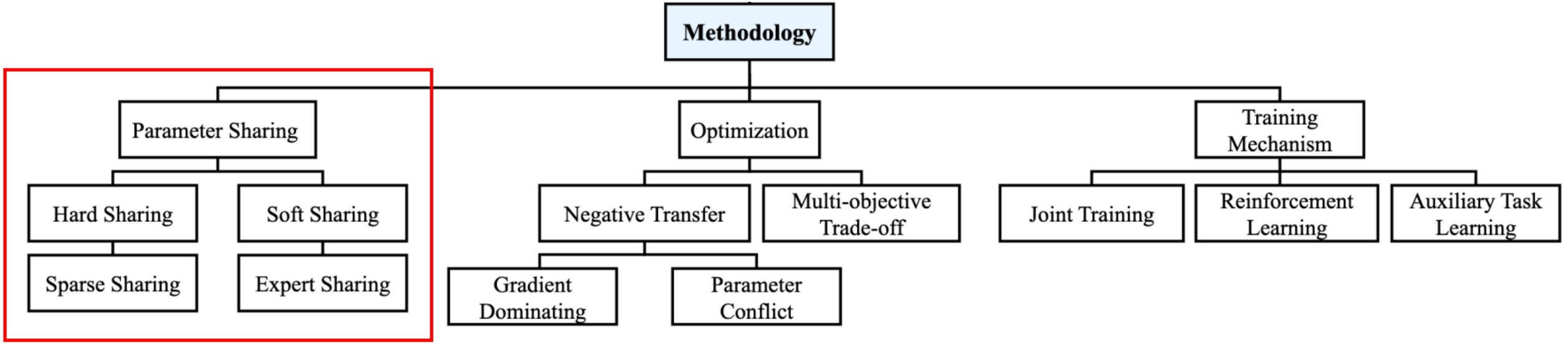
Auxiliary with Main Task

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

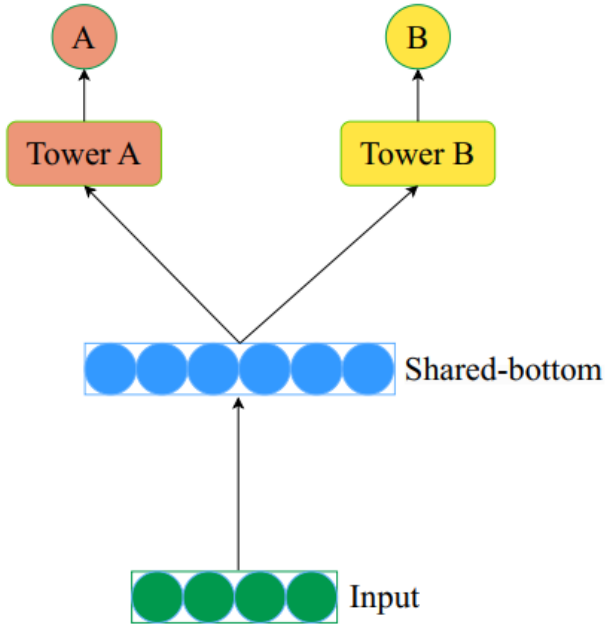
Methodology



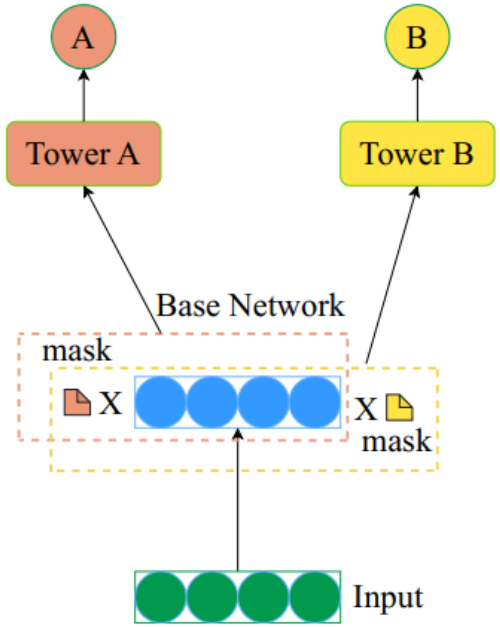
Parameter Sharing



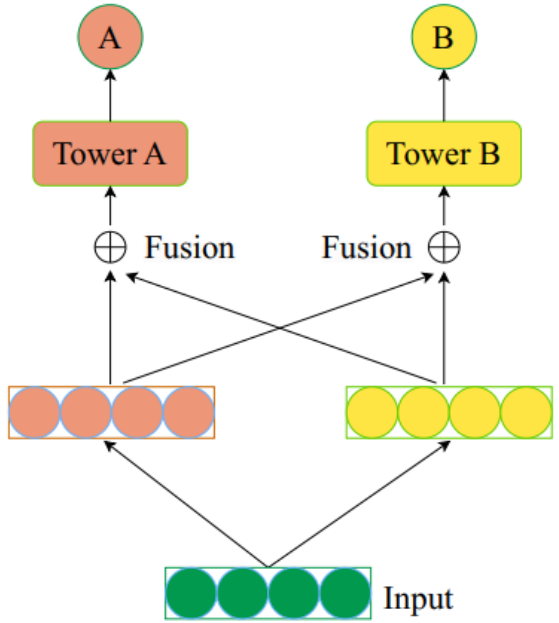
Parameter Sharing



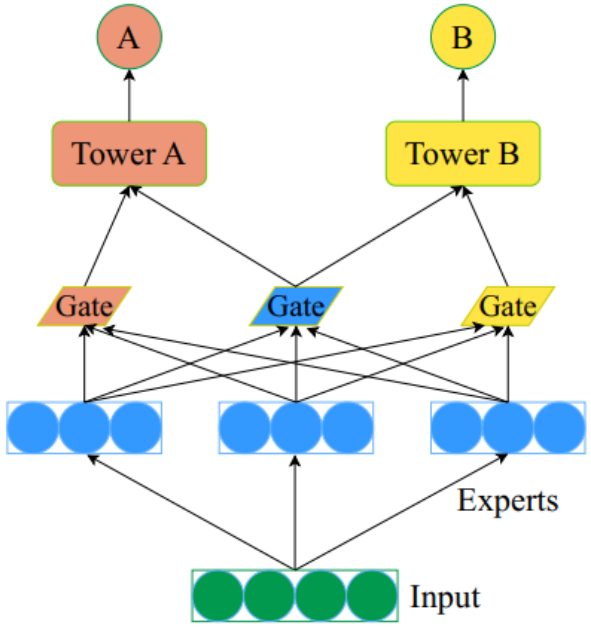
(a) Hard Parameter Sharing



(b) Sparse Sharing

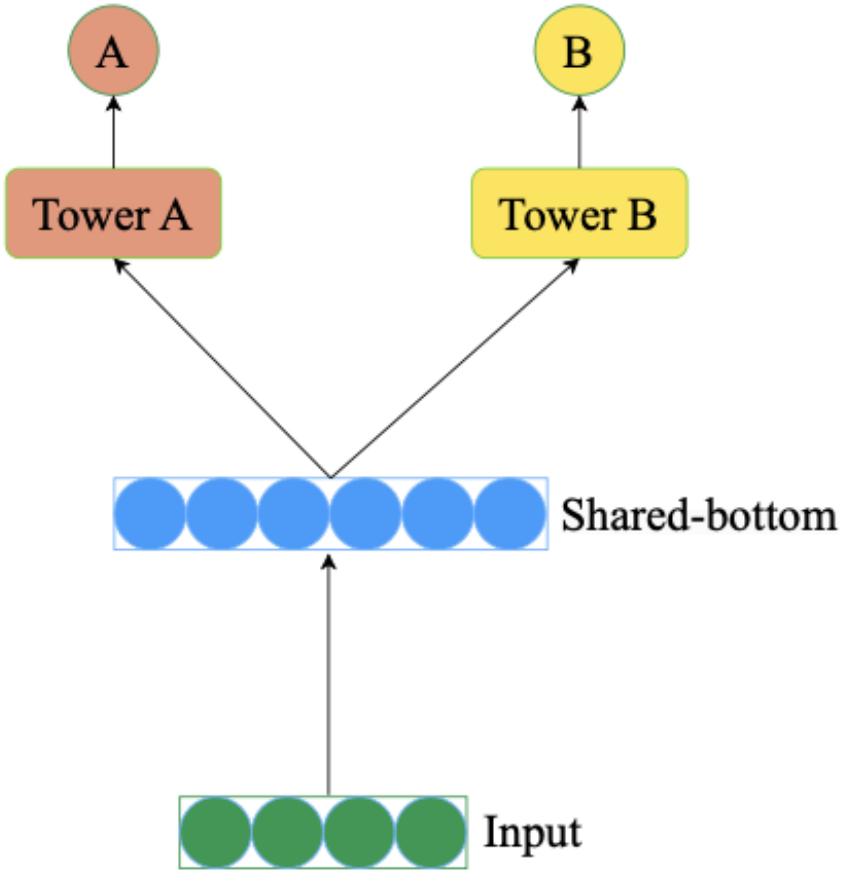


(c) Soft Parameter Sharing



(d) Expert Sharing

Hard Sharing

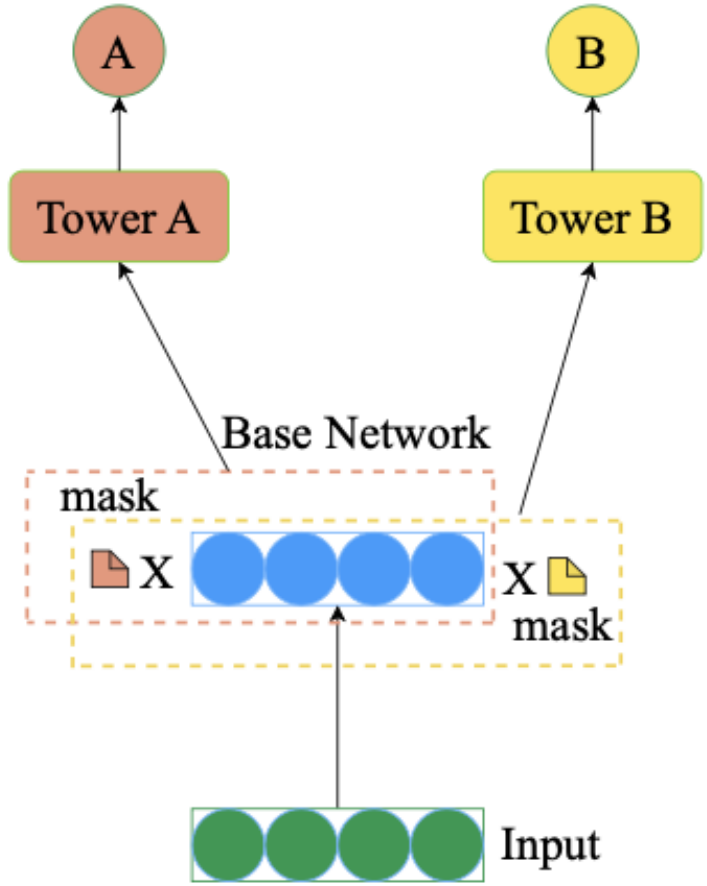


- Shared bottom layers extract the **same** information for different tasks,
- Task-specific top layers are trained individually

✓ Improving computation efficiency and alleviating over-fitting

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

Sparse Sharing

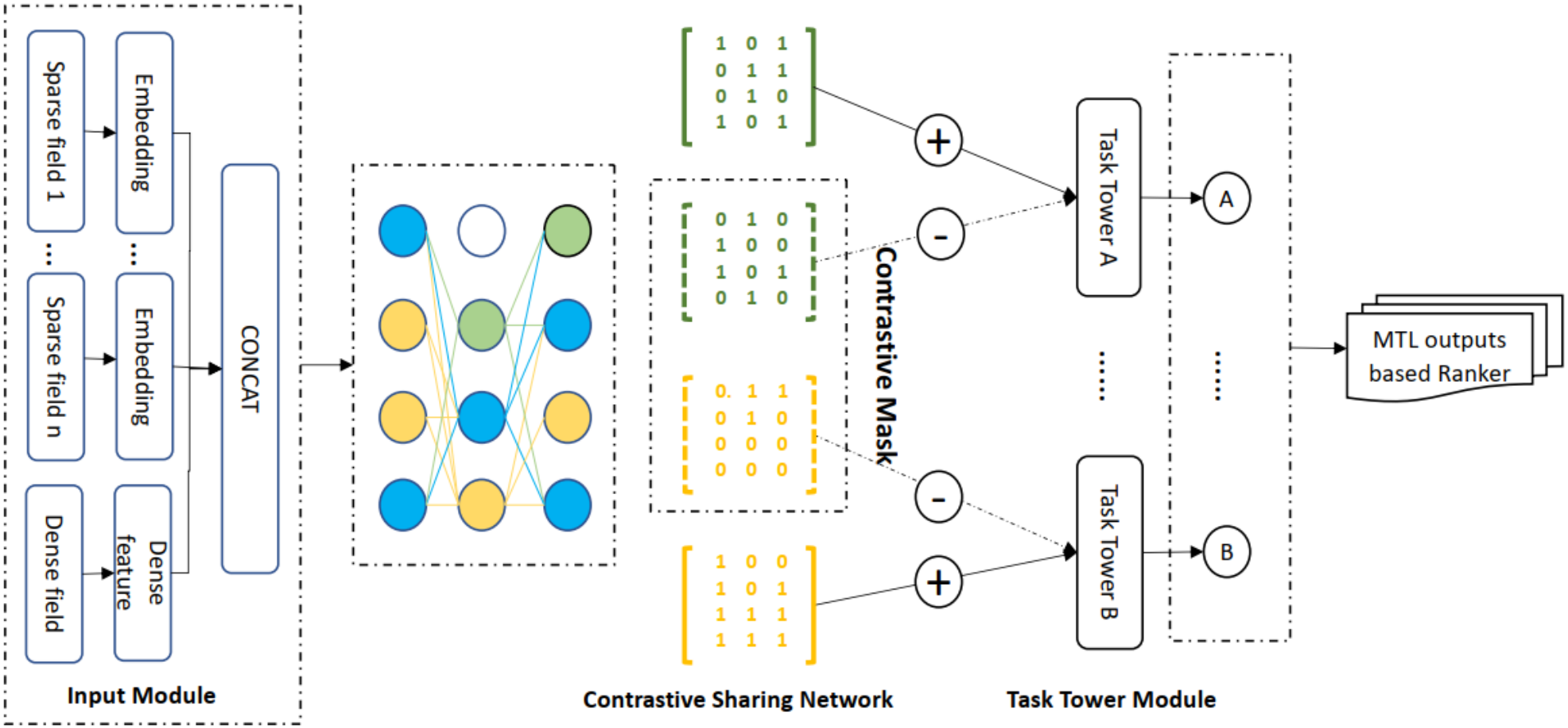


- Extracting **sub-networks** for each task by parameter masks from a base network
  - **Special case of Hard Sharing**

✓ Coping with the weakly related tasks flexibly

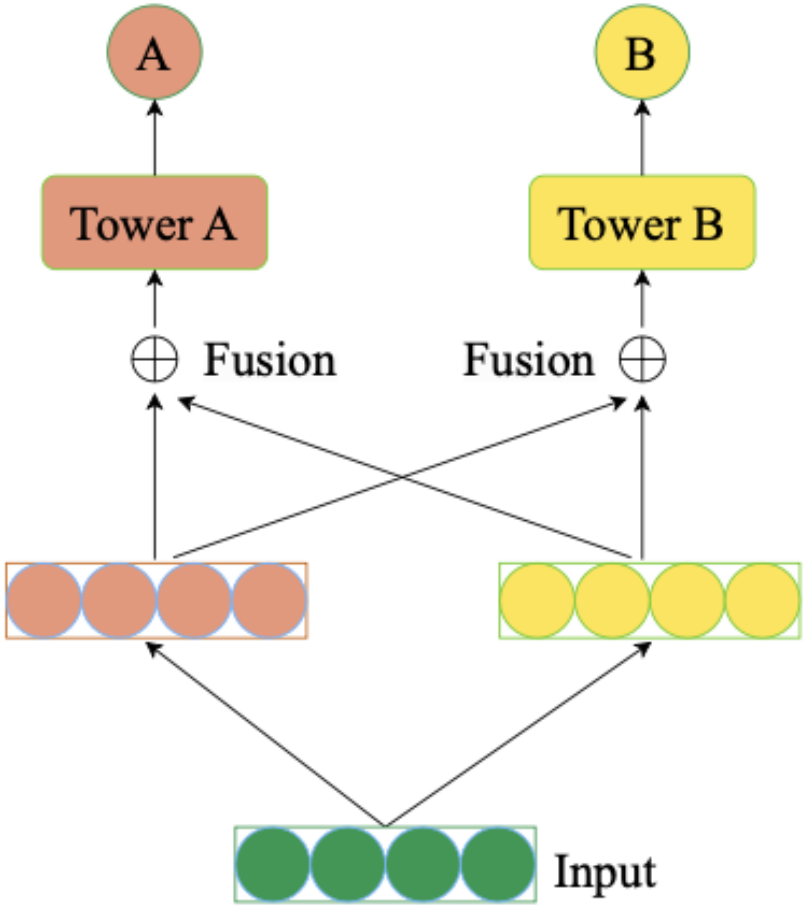
✗ Negative transfer
when updating shared parameters

CSRec



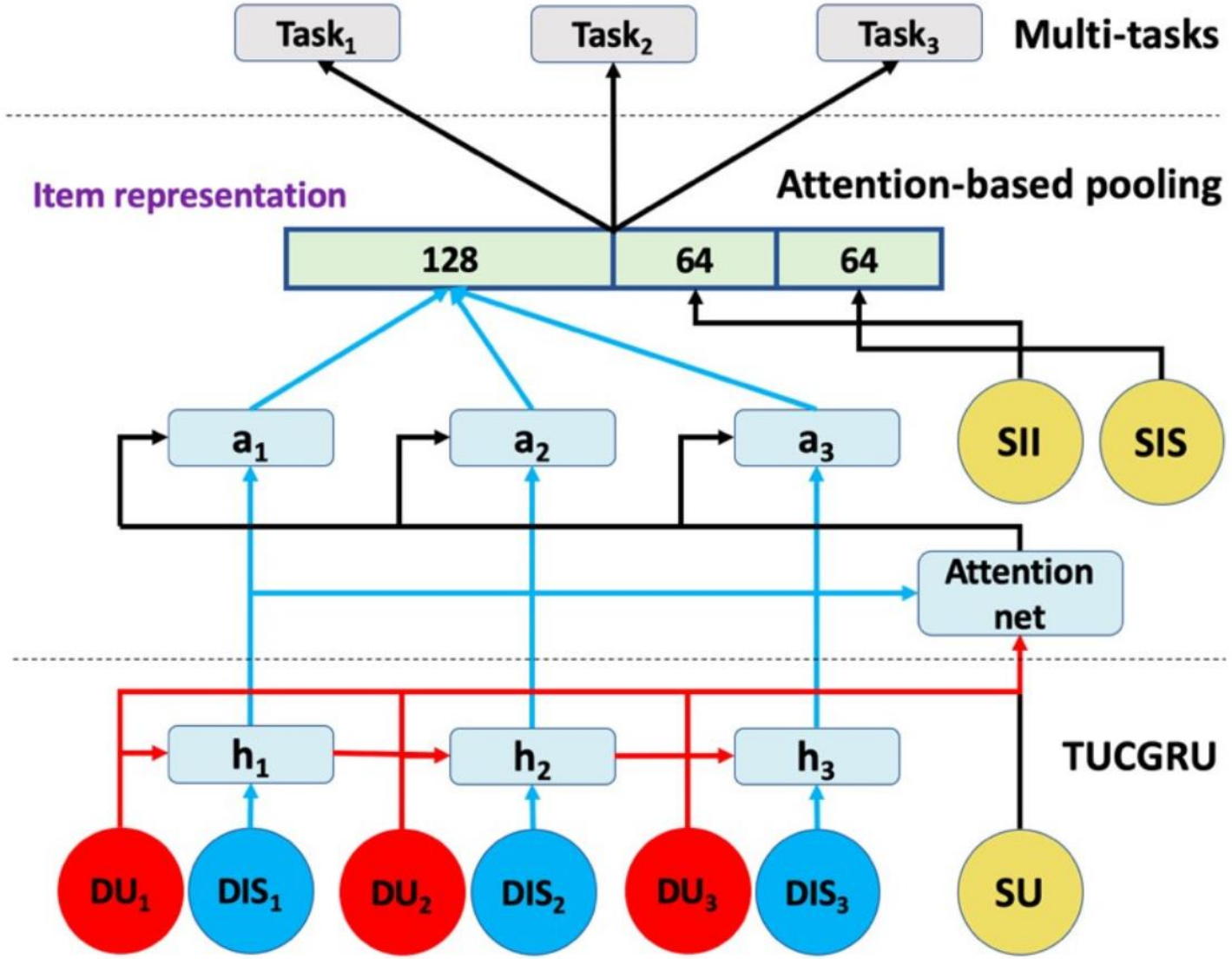
A Contrastive Sharing Model for Multi-task Recommendation. WWW 2022.

Soft Sharing



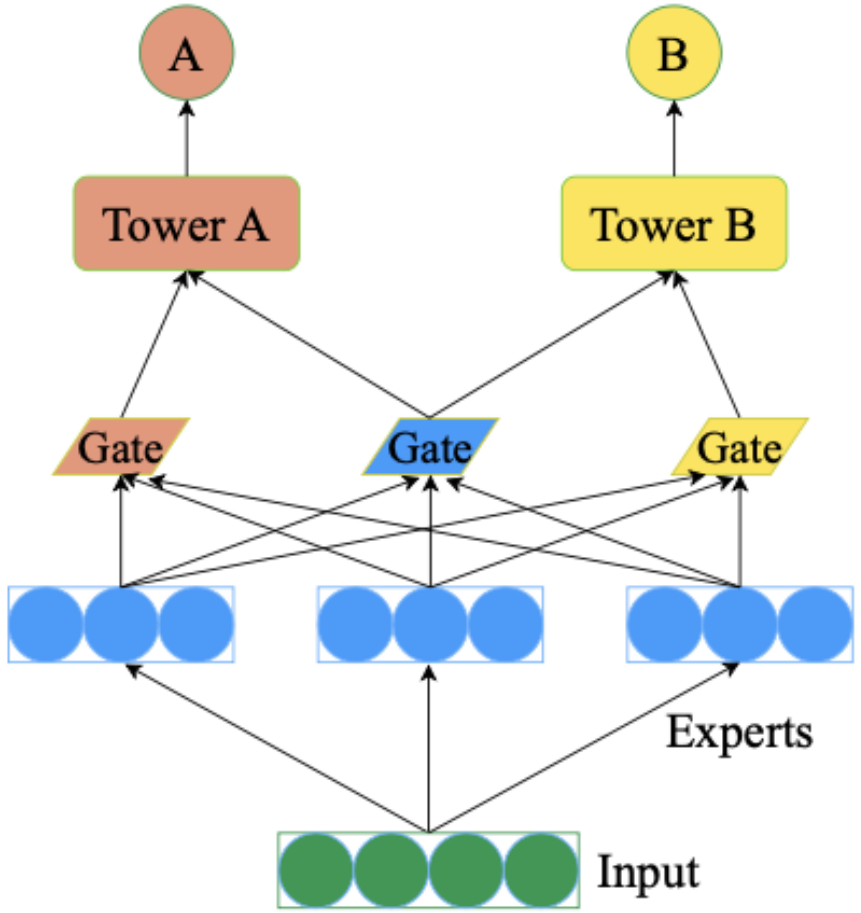
- Building separate models for tasks but the information among tasks is **fused by weights** of task relevance
- ✓ Relatively high **flexibility** in parameter sharing v.s. hard sharing
- ✗ Can not reconcile the flexibility
- ✗ Computation cost of the model

DINOP



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

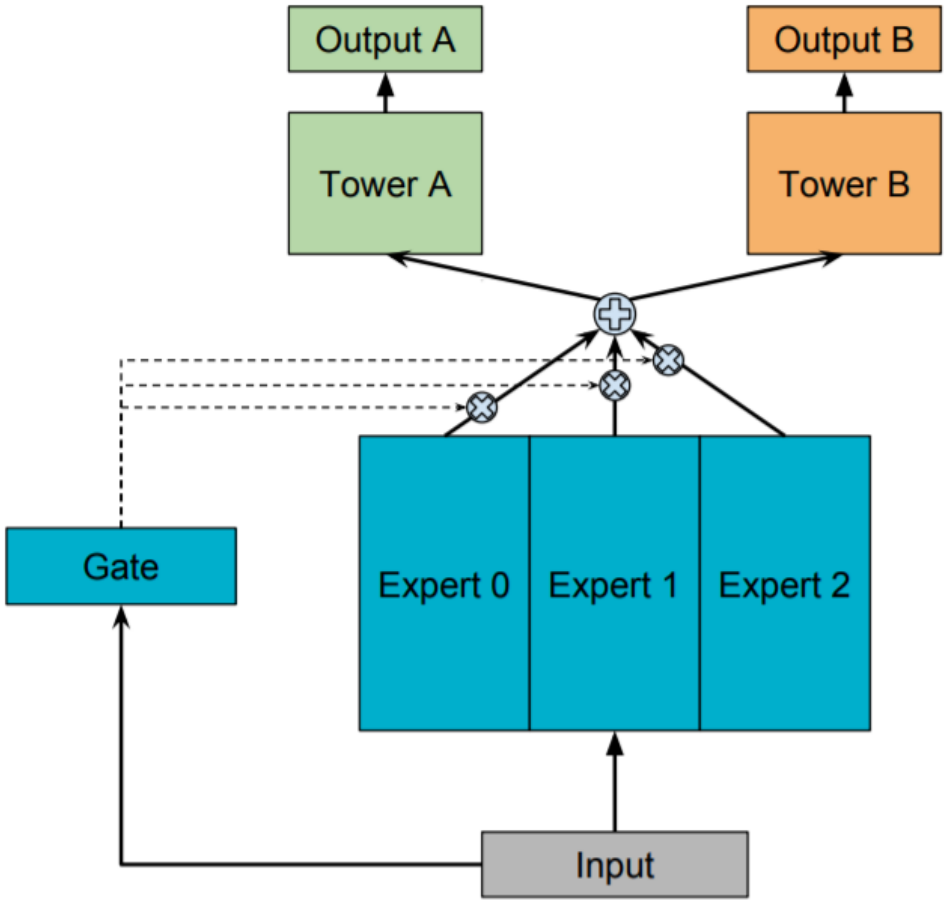
Expert Sharing



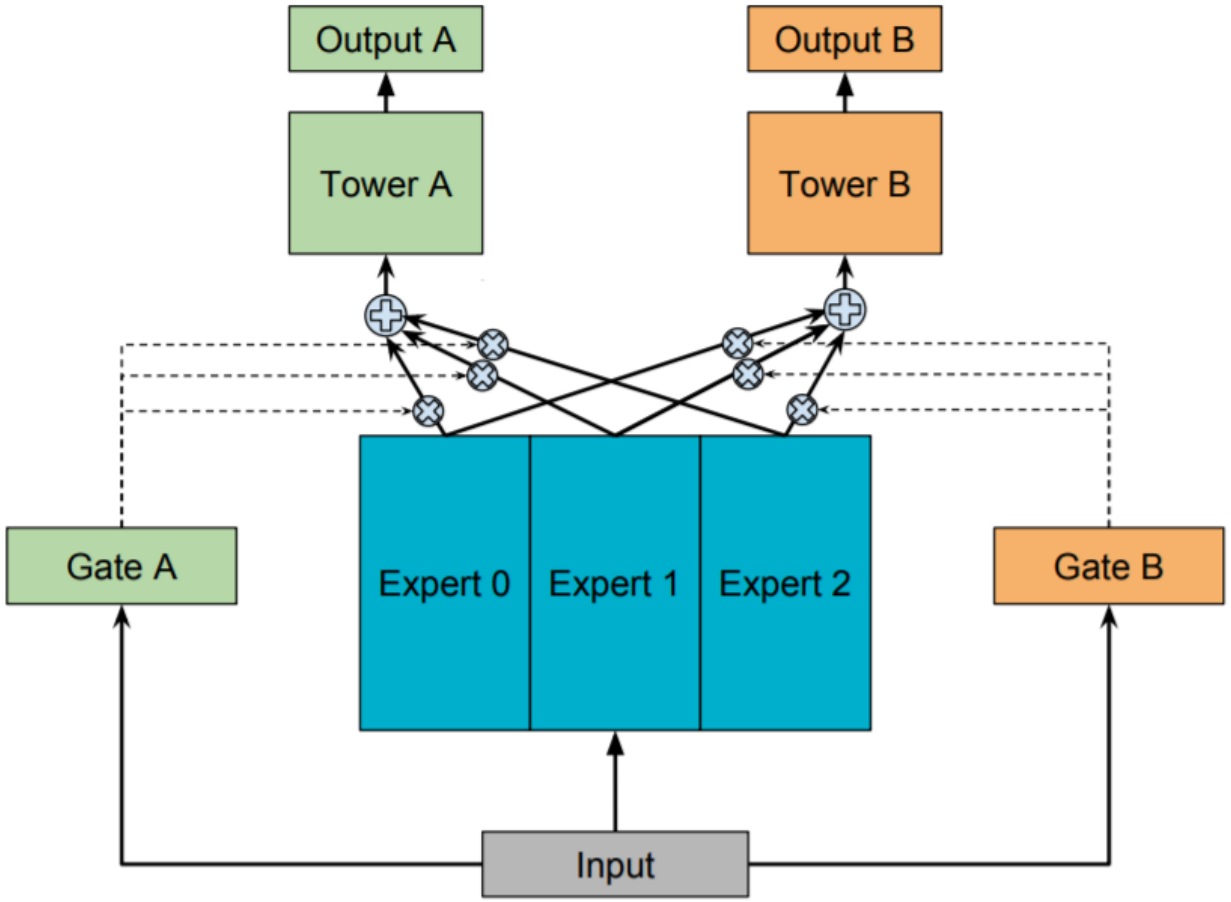
- Employing multiple **expert networks** to extract knowledge from shared bottom
  - → Fed into **task-specific** modules like gates
  - → Passed into the task-specific tower

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

MMoE



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



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

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

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

Expert Sharing

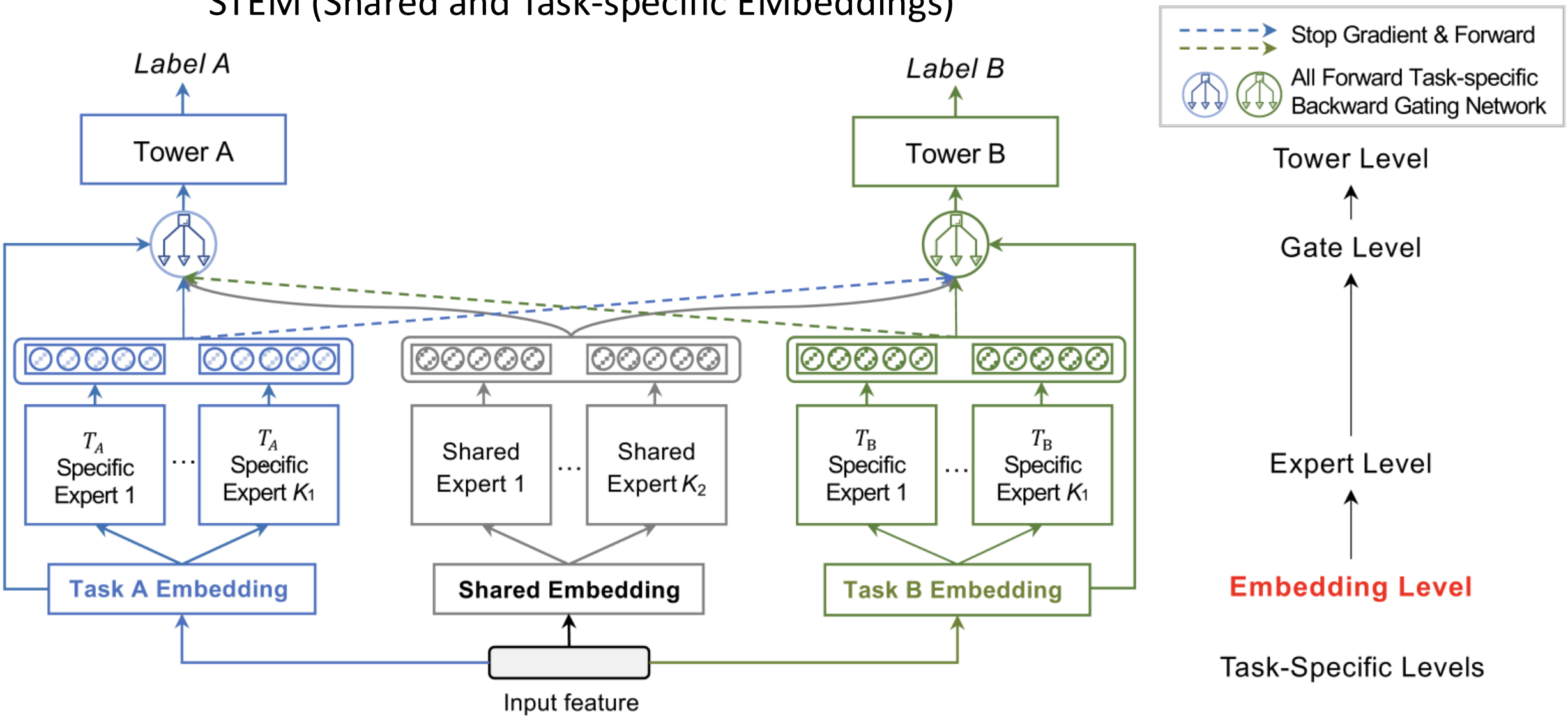
Model	Reference
MMoE	[Ma et al., 2018a]
SNR	[Ma et al., 2019]
PLE	[Tang et al., 2020]
DMTL	[Zhao et al., 2021]
DSelect-k	[Hazimeh et al., 2021]
MetaHeac	[Zhu et al., 2021]
PFE	[Xin et al., 2022]
MVKE	[Xu et al., 2022]
FDN	[Zhou et al., 2023]
MoME	[Xu et al., 2024]
MoSE	[Qin et al., 2020]

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

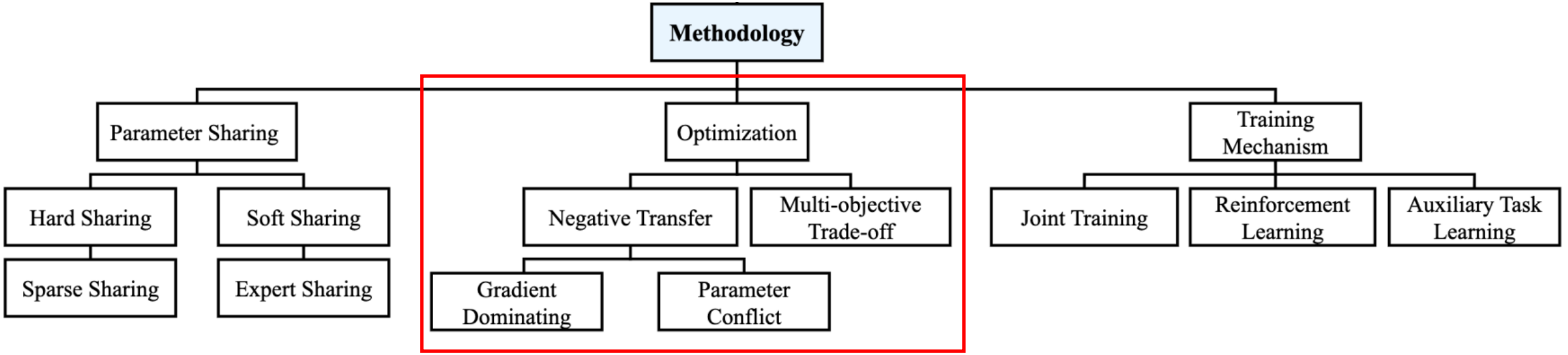
Processing **sequential** input features utilizing LSTM & sequential experts

Special Case: Multi-Embedding Paradigm

STEM (Shared and Task-specific EMbeddings)



Optimization



Negative Transfer

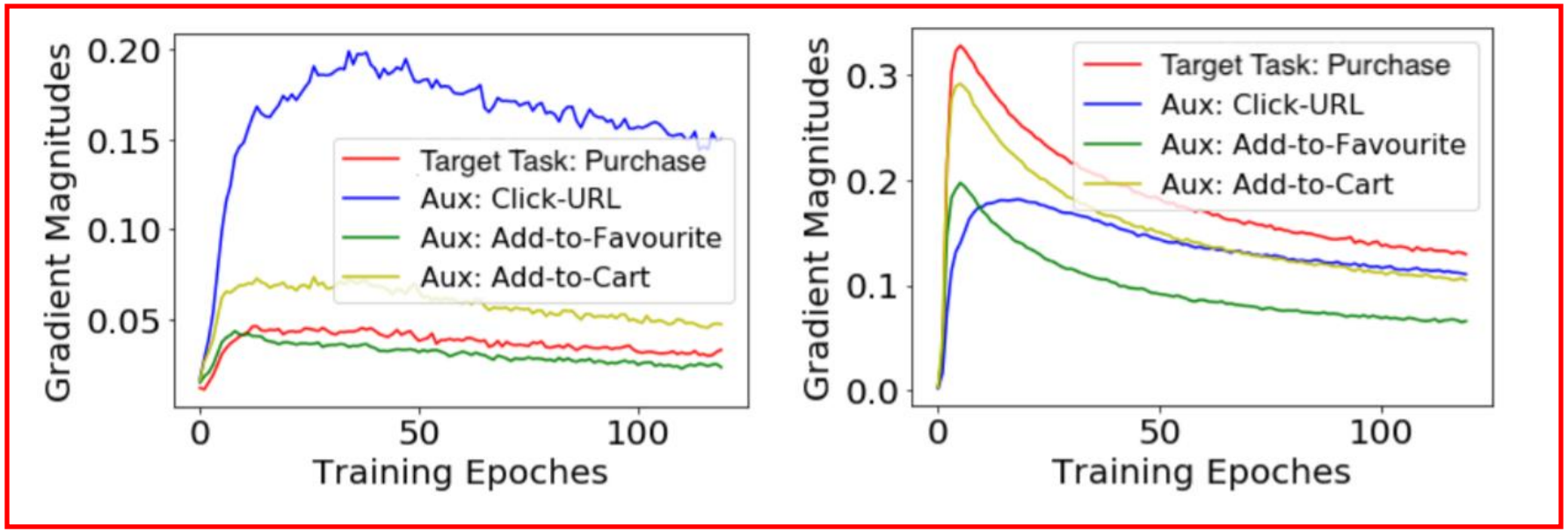
Gradient dominating $\|\nabla_\theta L^k(\theta)\|$

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

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

Works	Approach
PLE [Tang et al., 2020]	Proposing customized gate control (CGC) separating shared and task-specific experts
CSRec [Bai et al., 2022]	Alternating training procedure and contrastive learning on parameter masks to reduce the conflict probability
GradCraft [Bai et al., 2024]	Adjusting gradient norm and deconflicting global direction through projection and combination

MetaBalance

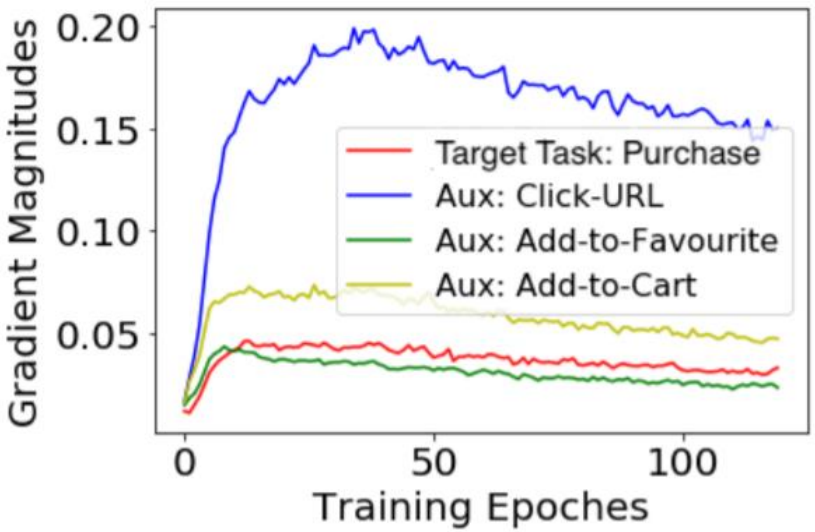


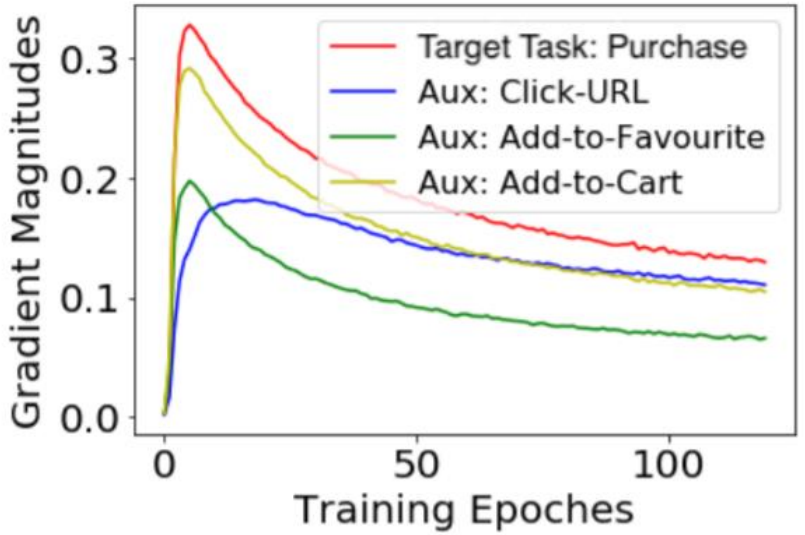
$$\theta^{t+1} = \theta^{t} - \alpha * \mathbf{G}^{t}_{total}$$

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

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

MetaBalance



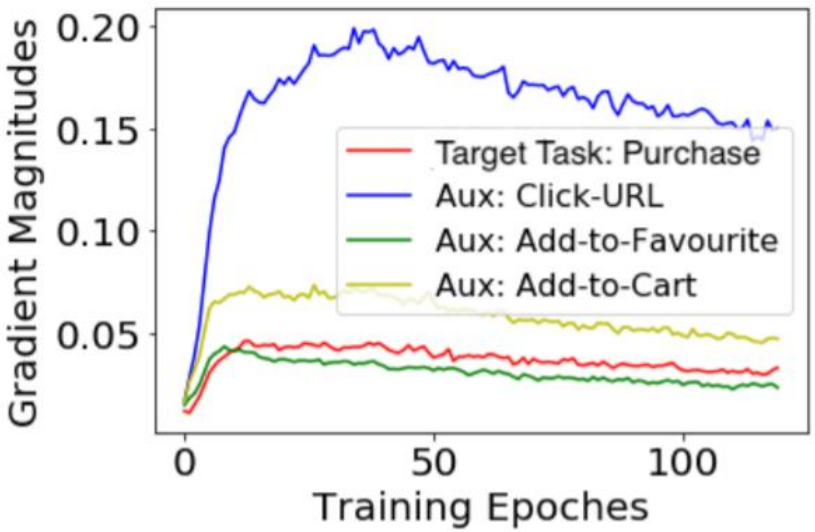


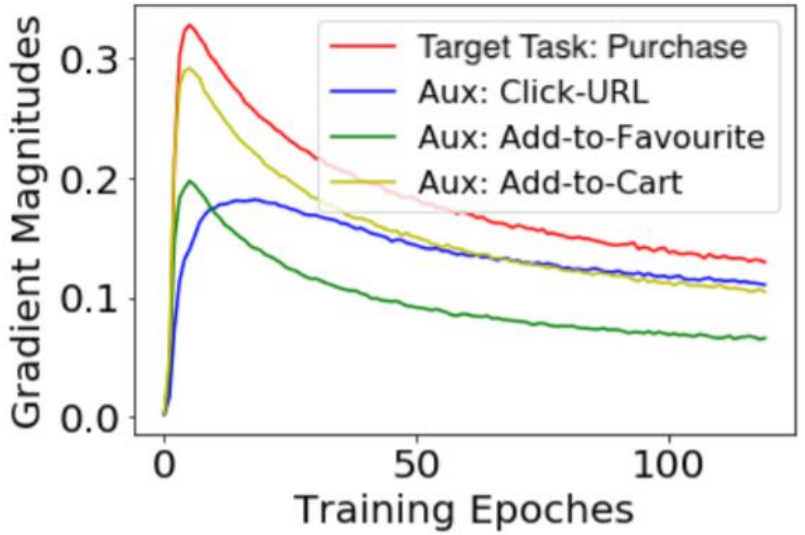
$$\theta^{t+1} = \theta^{t} - \alpha * \mathbf{G}^{t}_{total}$$

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

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

MetaBalance





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

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

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

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

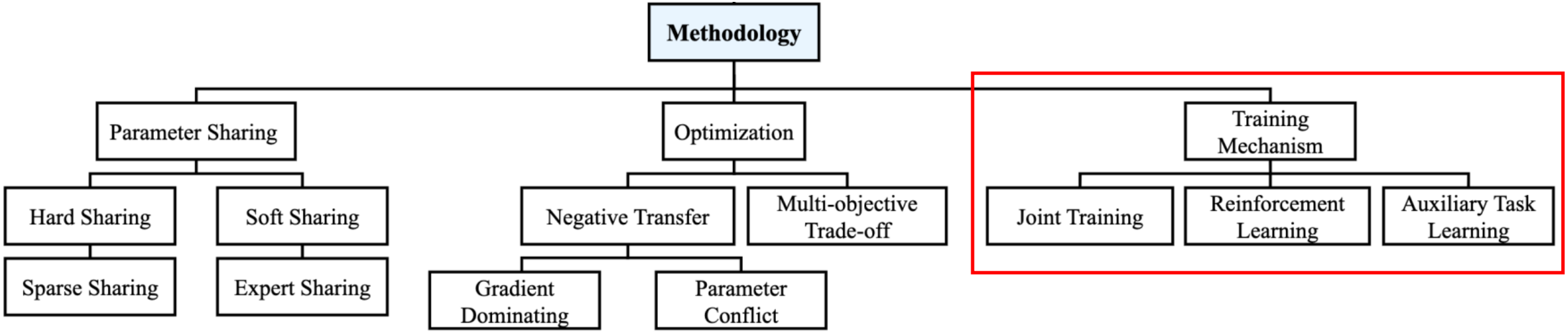
Multi-objective Trade-off

Objectives optimized regardless of the **potential conflict**

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

Training Mechanism

Training process & Learning strategy



Joint Training

Parallel manner

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

Reinforcement Learning

Sequential user behaviors as MDP

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

Auxiliary Task Learning

Joint training & Others

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

Application Fields

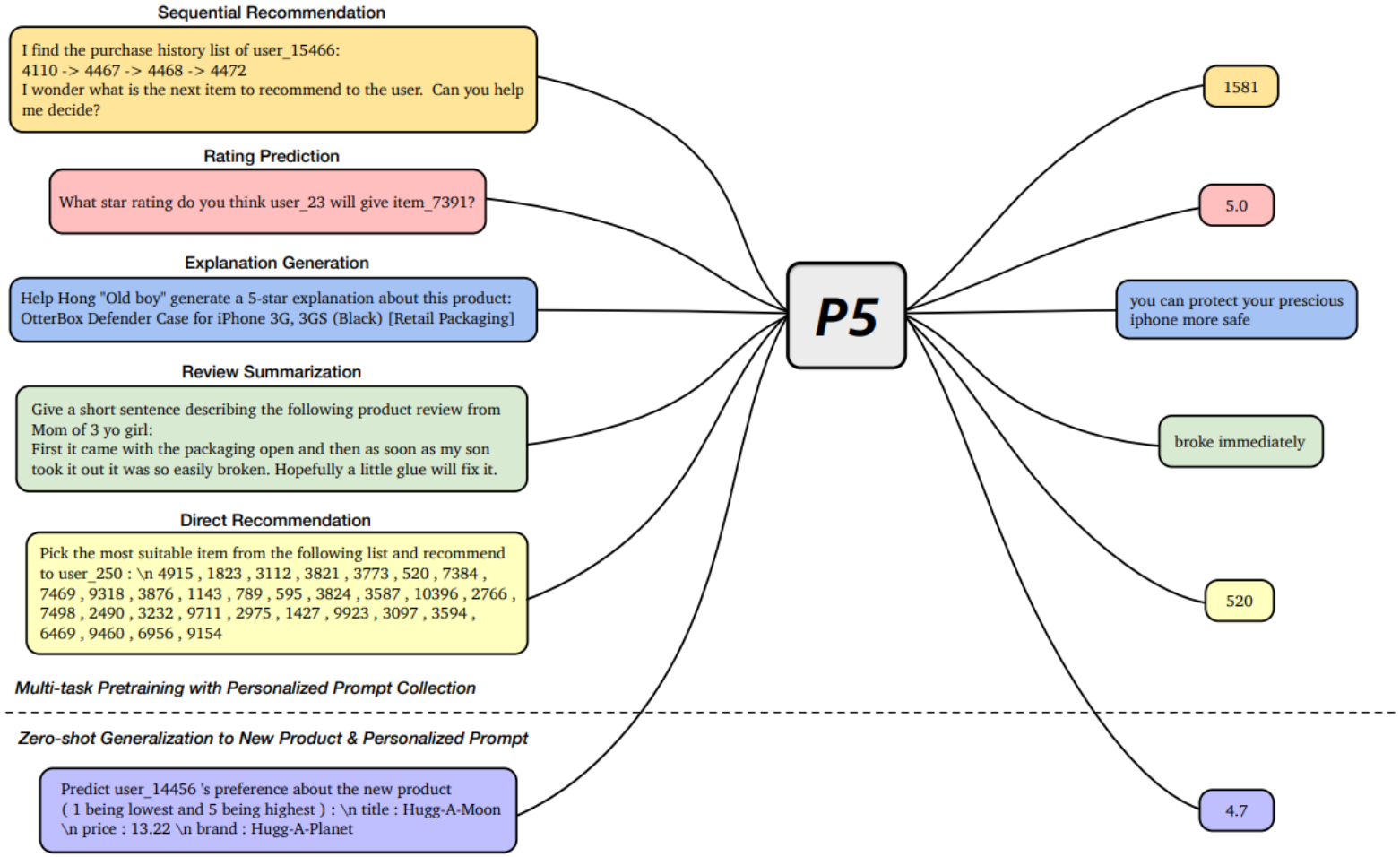
- **E-commerce** : Main focus
- **Advertising**
  - Utility & Cost
    i. MM-DFM [Hou et al., 2021]: Performing multiple conversion prediction tasks in different observation duration
    ii. MetaHeac [Zhu et al., 2021]: Handling audience expansion tasks on content-based mobile marketing
    iii. MVKE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

Datasets

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

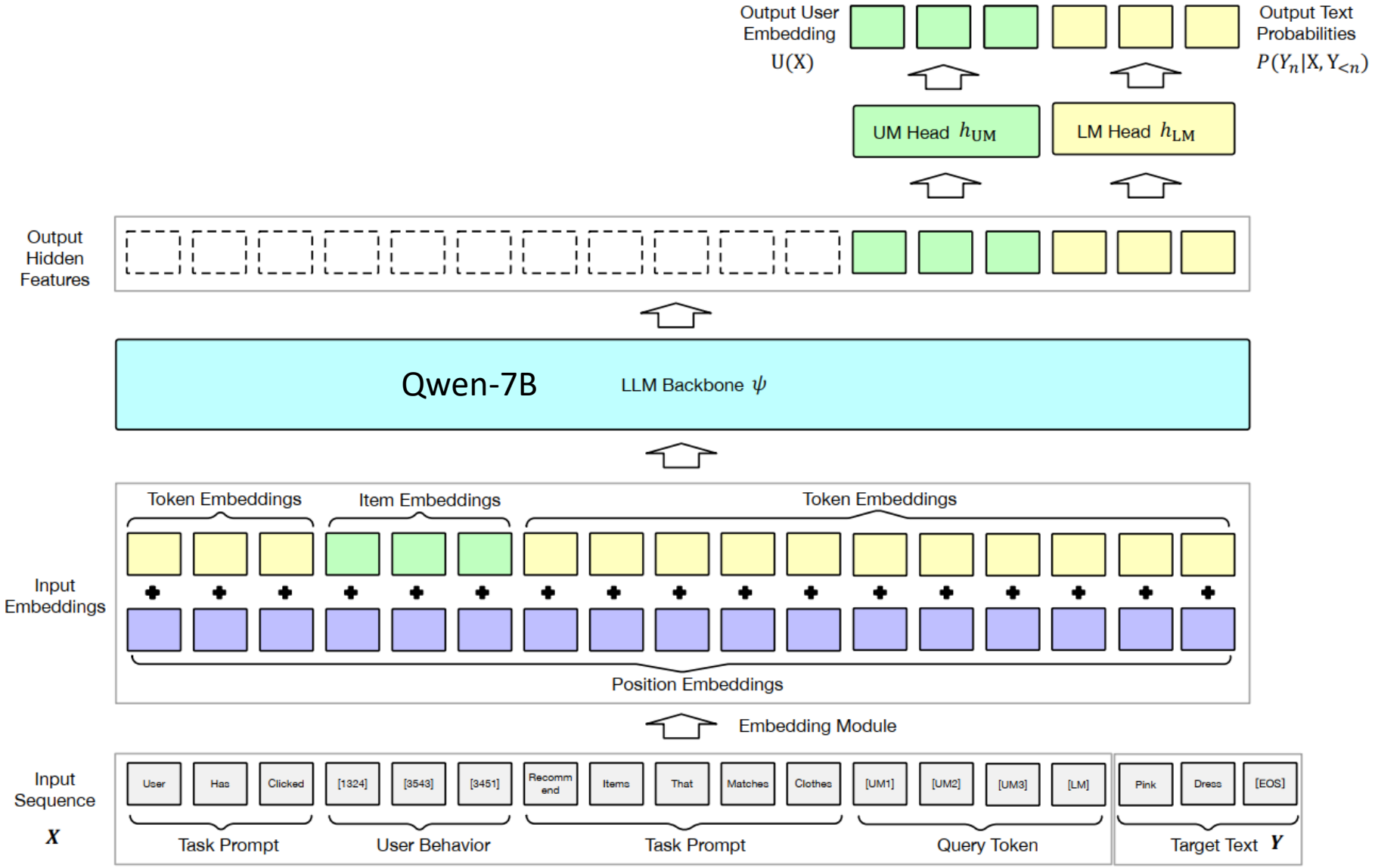
P5

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



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

URM



Large Language Models Are Universal Recommendation Learners. Arxiv 2025.

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]

Summary

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

Challenges & Future Directions

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

Challenges & Future Directions

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

Challenges & Future Directions

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

Conclusion

- Task relation:
  - Parallel, Cascaded, Auxiliary with Main
- Methodology:
  - Parameter Sharing, Optimization, Training Mechanism

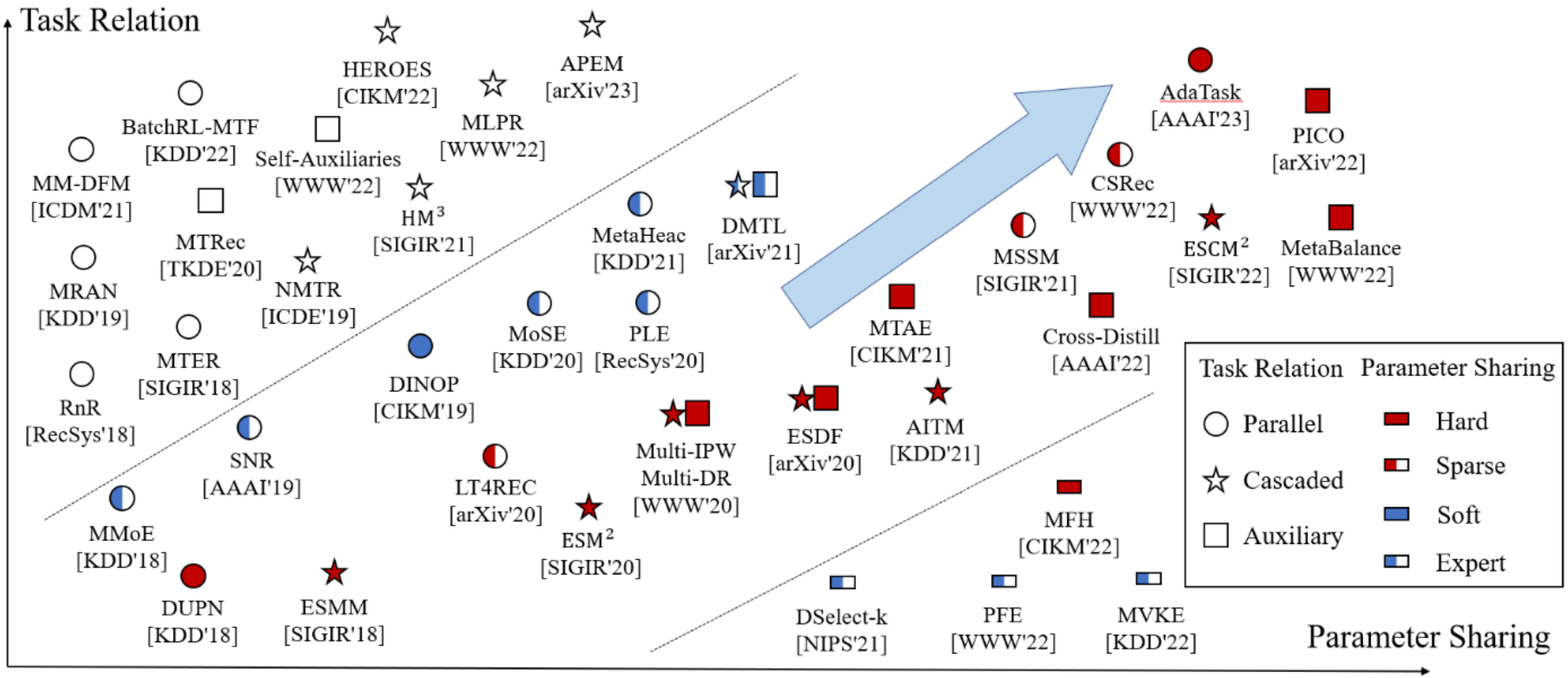
Survey

https://arxiv.org/abs/2302.03525

Multi-Task Deep Recommender Systems: A Survey

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

Trend of MTDRS



Agenda

Introduction

Yejing Wang

Joint Modeling in RS

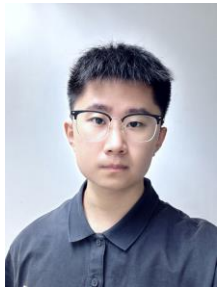
Pengyue Jia

Multi-task Recommendation

Yuhao Wang

Multi-scenario Recommendation

Pengyue Jia

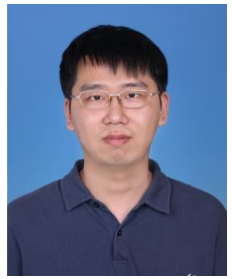


Xiaopeng Li

Multi-behavior Recommendation

Jingtong Gao

Multi-modal Recommendation



Qidong Liu

Conclusion

Future Work

Yichao Wang

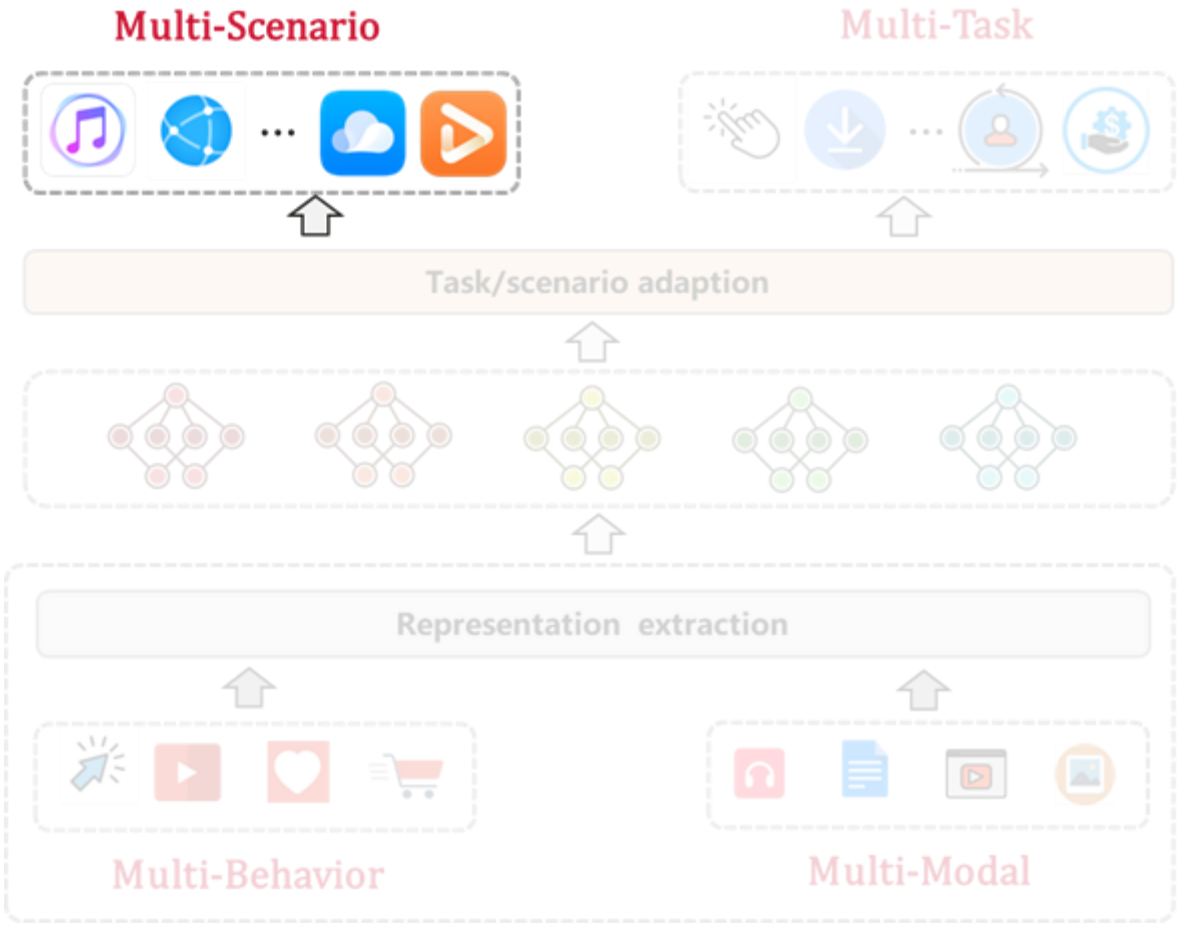
Background

- Multi-Scenario Recommender Systems:
  - By using a unified model to simultaneously model multiple scenarios, the goal of improving the effects of different scenarios at the same time is achieved through information transfer between scenarios.

- Importance:
  - Time/Memory efficiency; Maintenance cost
  - Accuracy

- Classification on Methods:
  - Shared-Specific network paradigm
  - Dynamic weight
  - Multi-scenario & Multi-task recommendation

Multi-Scenario Modeling



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

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

$E^{B} = G(H_1, H_2, \ldots, H_N)$ — Multi-behavior

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

$wL(\boldsymbol{E}^{Merge}, \boldsymbol{\theta}^{sh}, \theta^{t}, \boldsymbol{\theta}^{s})$ — Multi-scenario

$wL(E^{Merge}, \theta^{sh}, \theta^{t}, \theta^{s})$ — Multi-task

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

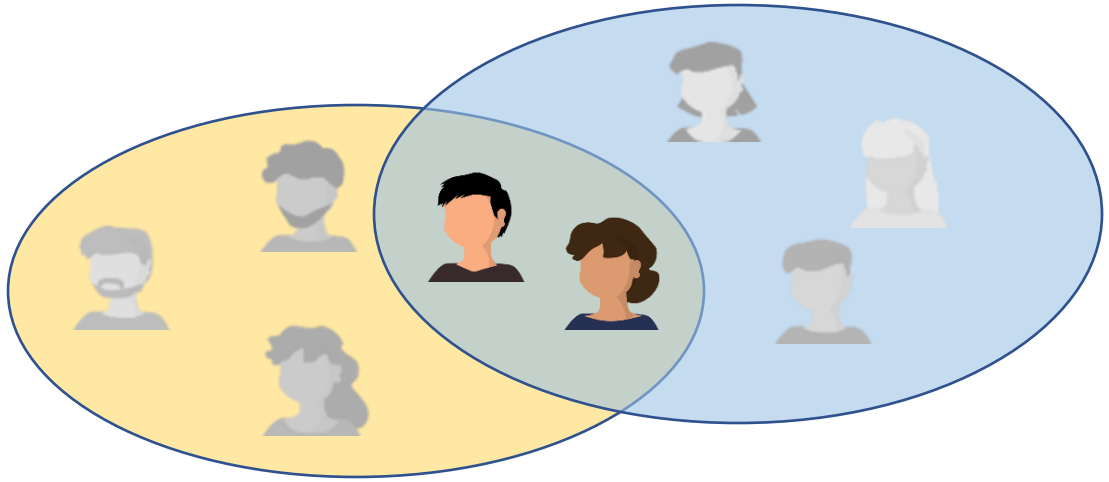
Recommendation Scenarios

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

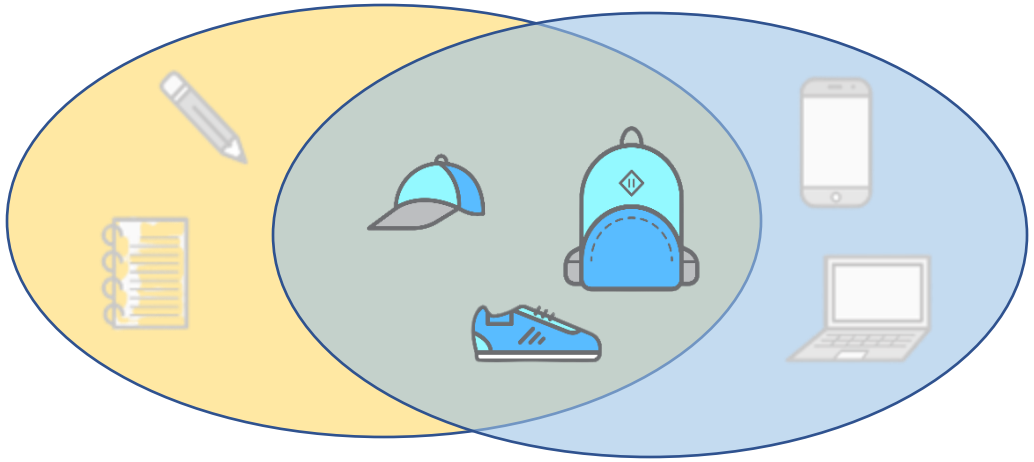
- Scenario and Domain?
  - Generally do not make a distinction
  - The same in this tutorial

Commonalities and Diversities

- Commonalities
  - User Overlap



- Commonalities
  - Item Overlap



Commonalities and Diversities

➢Diversities

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

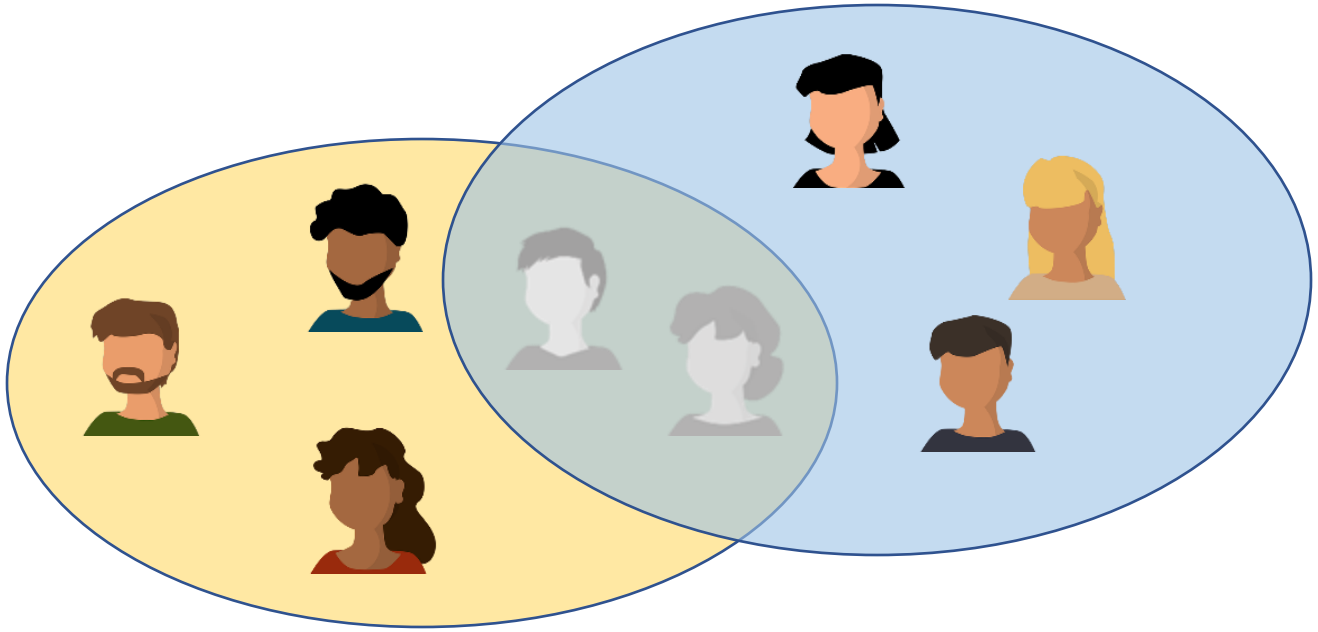
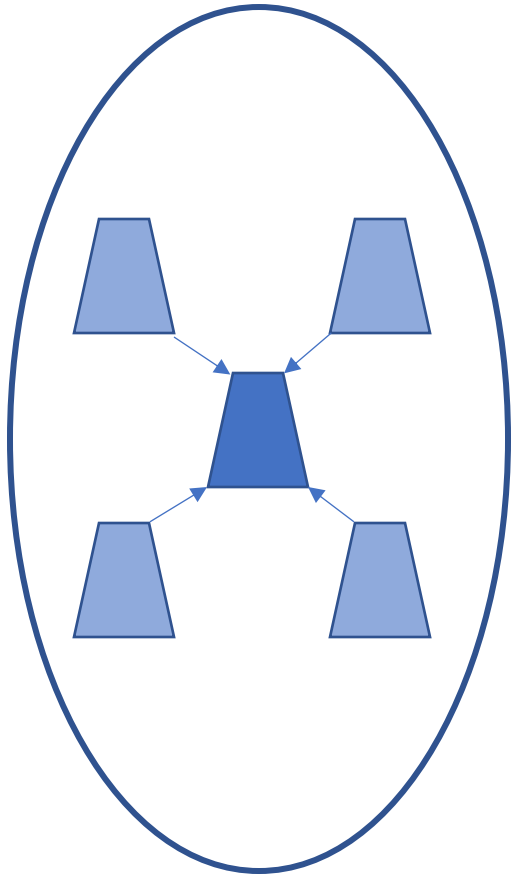
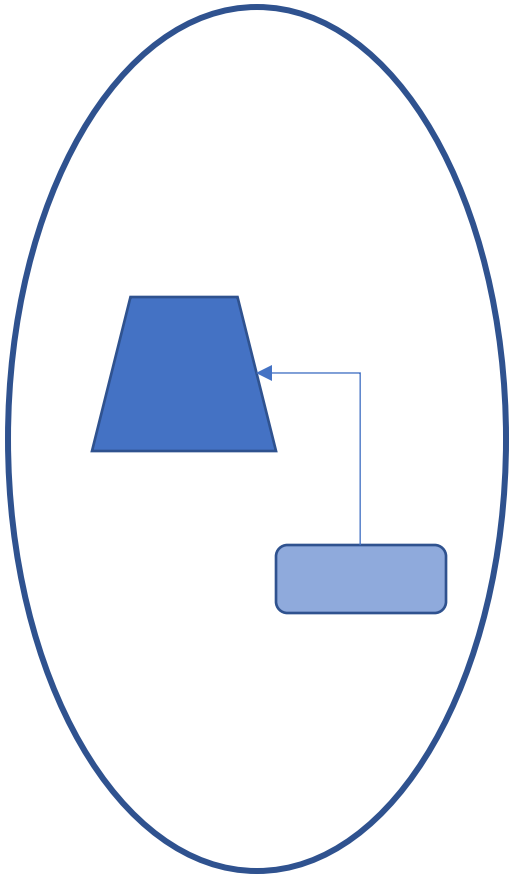


Table of Contents



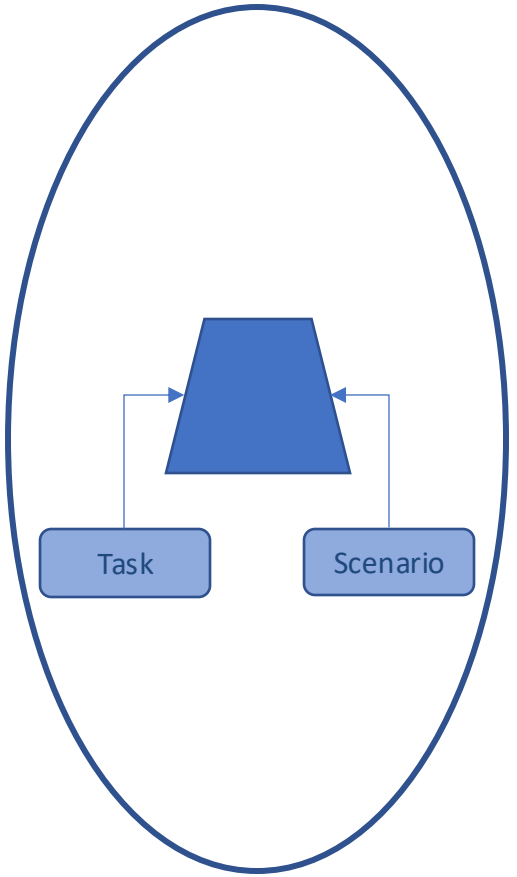
Shared-specific network paradigm

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



Dynamic weight

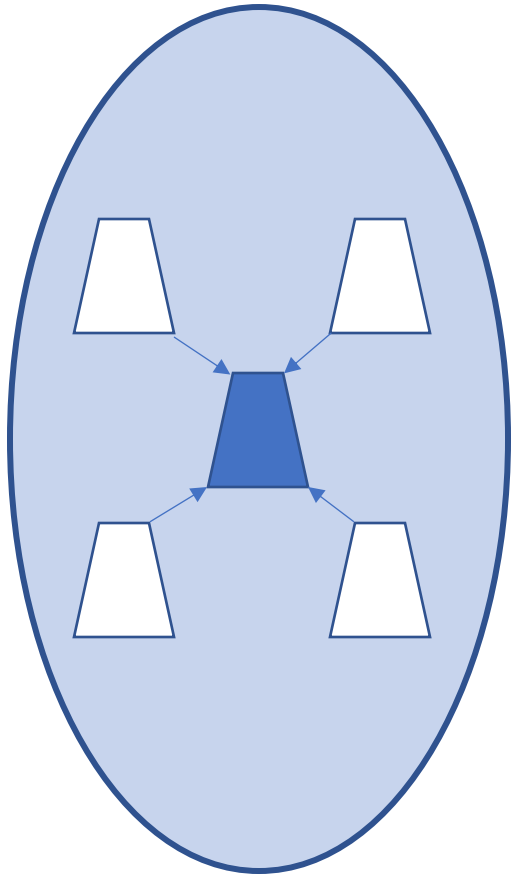
$wL(E^{Merge}, \theta^{sh}, \theta^{t}, \boldsymbol{\theta}^{s})$



Multi-Scenario & Multi-Task

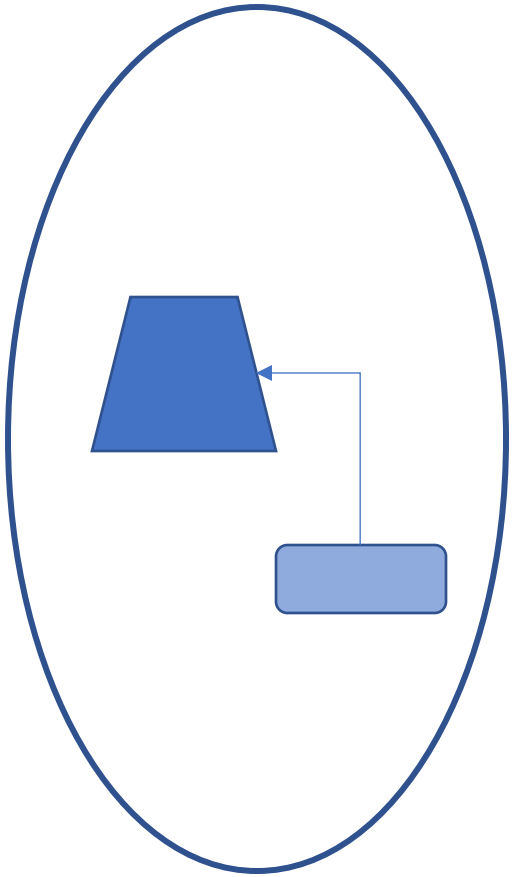
$wL(E^{Merge}, \theta^{sh}, \boldsymbol{\theta}^{t}, \boldsymbol{\theta}^{s})$

Table of Contents



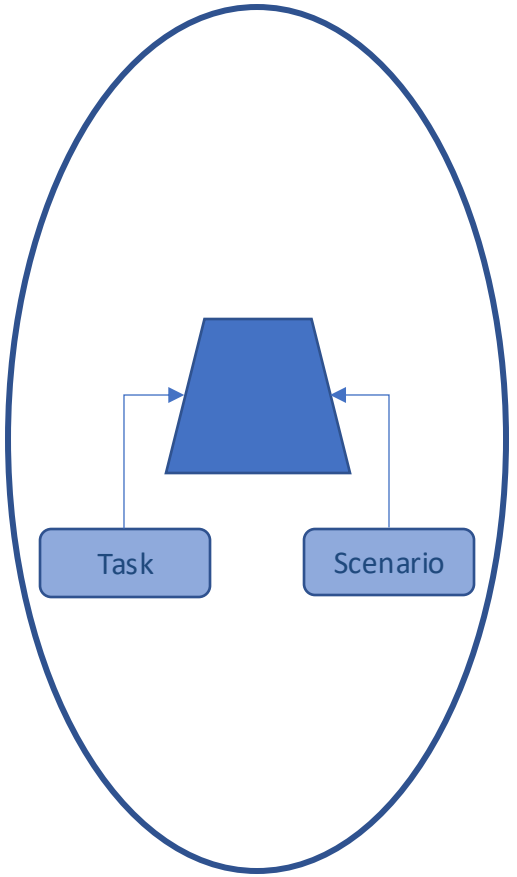
Shared-specific network paradigm

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



Dynamic weight

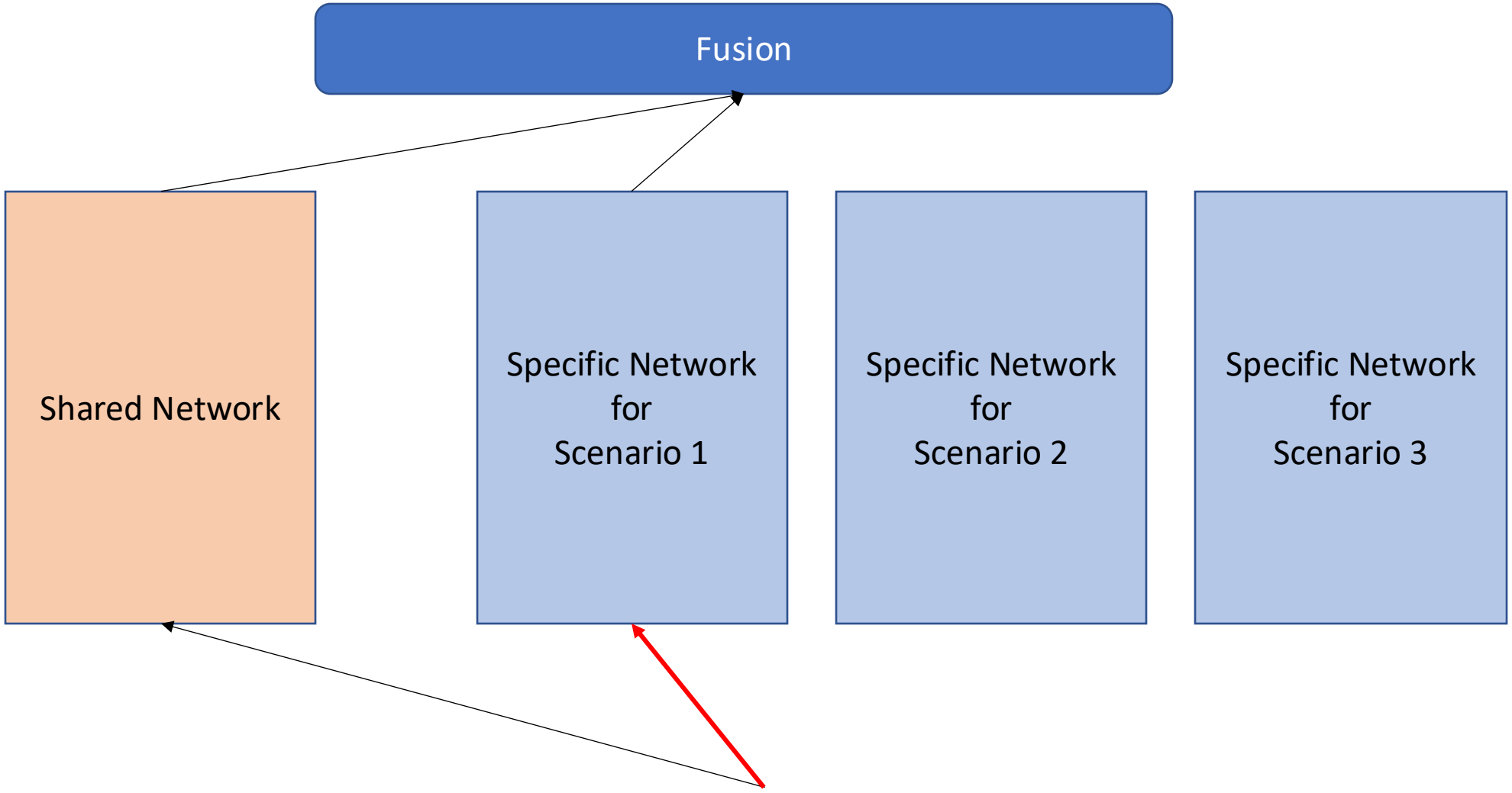
$wL(E^{Merge}, \theta^{sh}, \theta^{t}, \boldsymbol{\theta}^{s})$



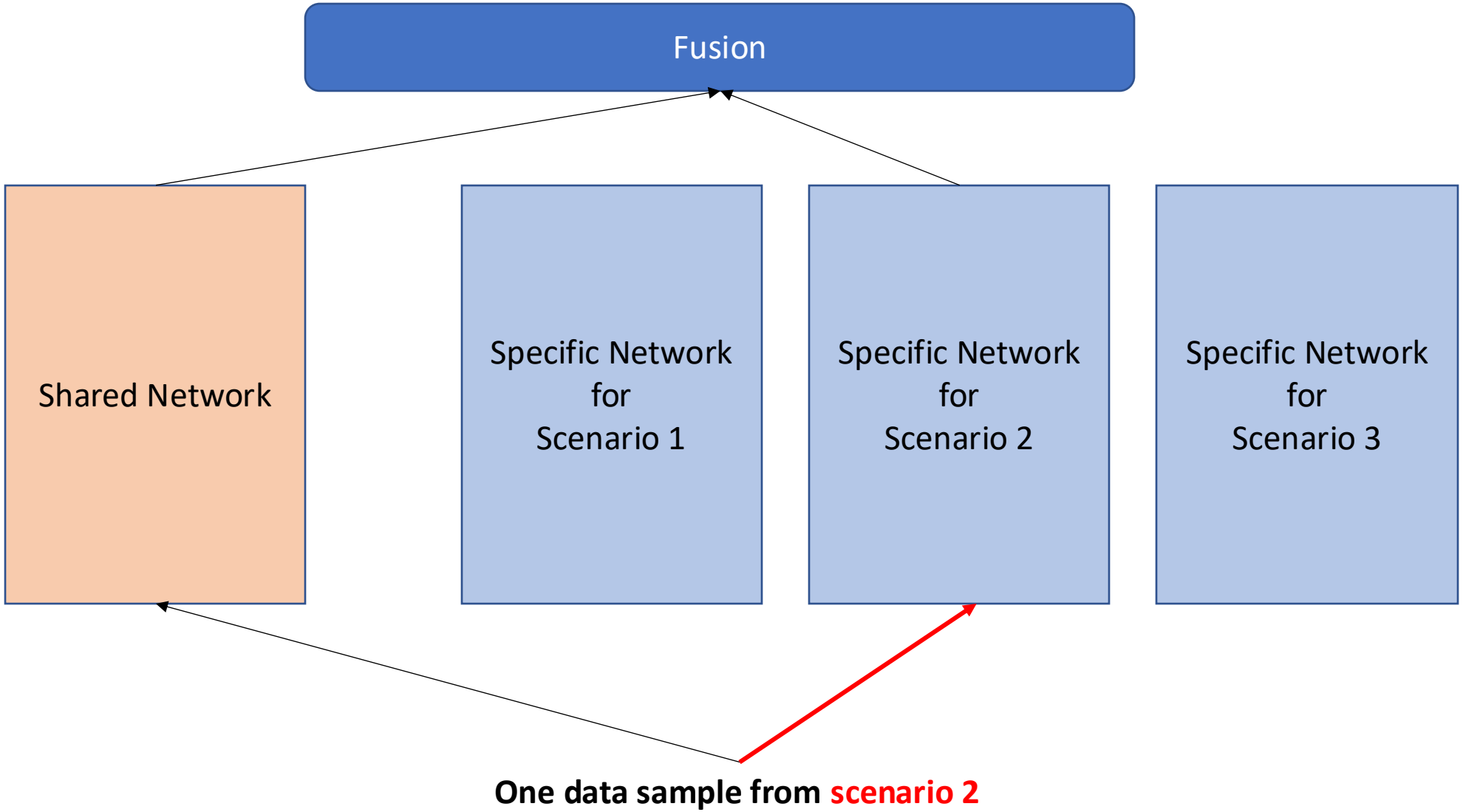
Multi-Scenario & Multi-Task

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

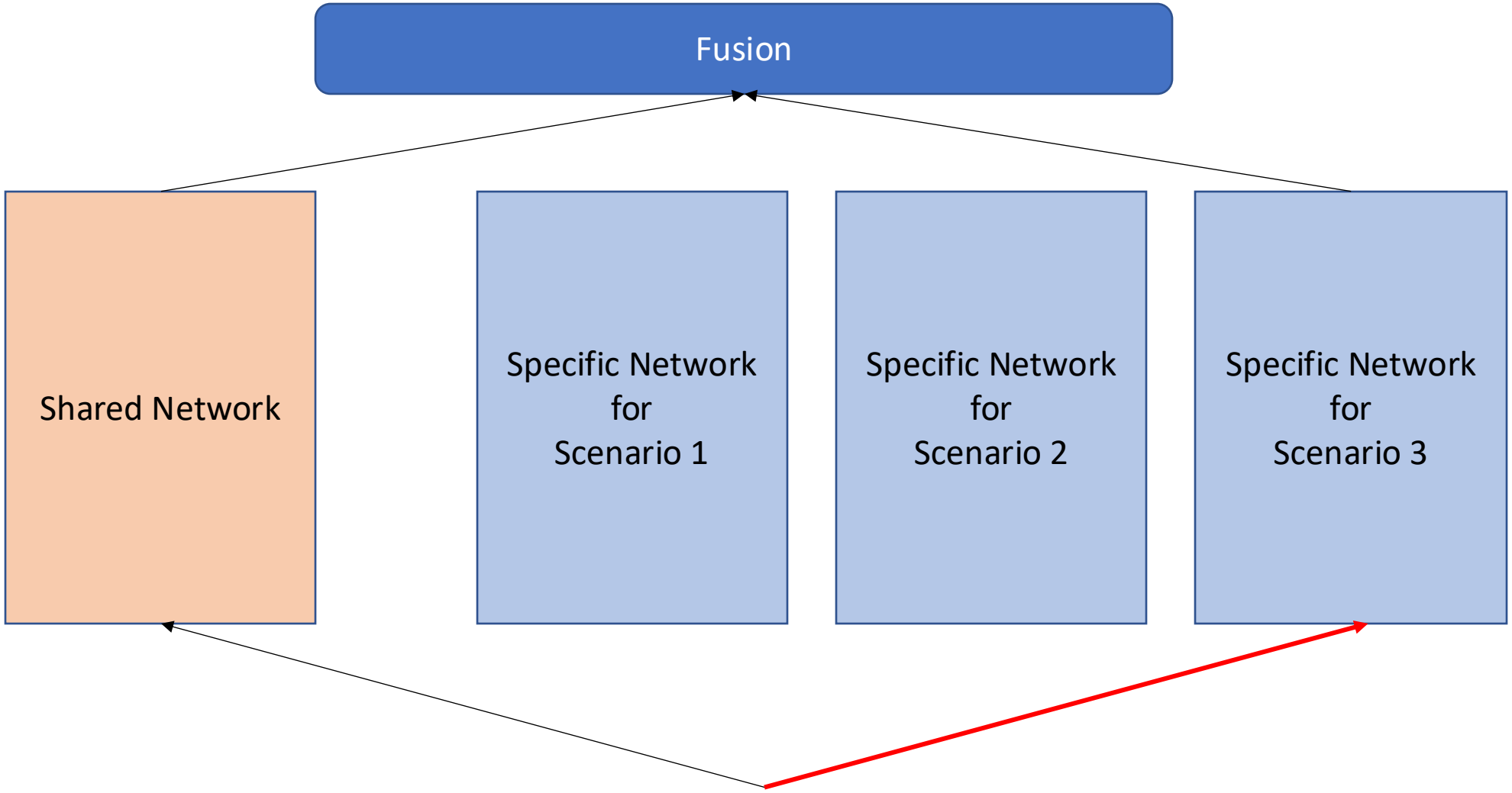
Shared-specific Network Paradigm



Shared-specific Network Paradigm



Shared-specific Network Paradigm



STAR

- Motivation:
  - Training individual models for each domain → does not fully use the data from all domains
  - Data across domains owns commonalities and characteristics
- Target:
  - Use a single model to serve multiple domains simultaneously
  - Shared network → commonalities
  - Specific network → characteristics
- Methods:
  - Partitioned Normalization
  - STAR Topology
  - Auxiliary Network

Banner

Guess What You Like

One Model to Serve All: Star Topology Adaptive Recommender for Multi-Domain CTR Prediction. CIKM 2021.

STAR Details

- Partitioned Normalization (PN)
- Training

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

- Testing

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

Compared to BN

- Batch Normalization (BN)
- Training

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

- Testing

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

STAR Details

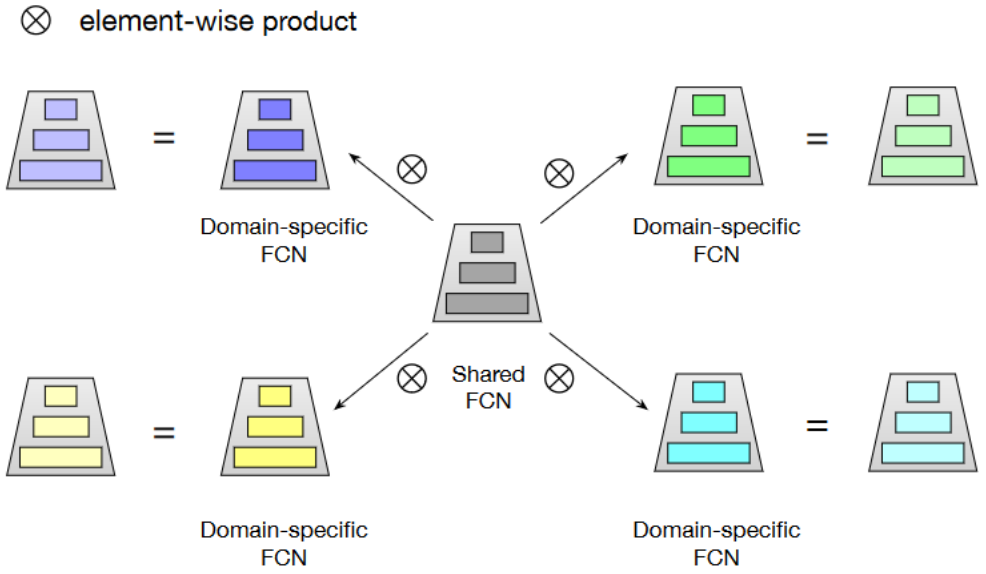
STAR Topology

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

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

The output for p-th domain is derived by:

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



SAR-Net

➢Motivation

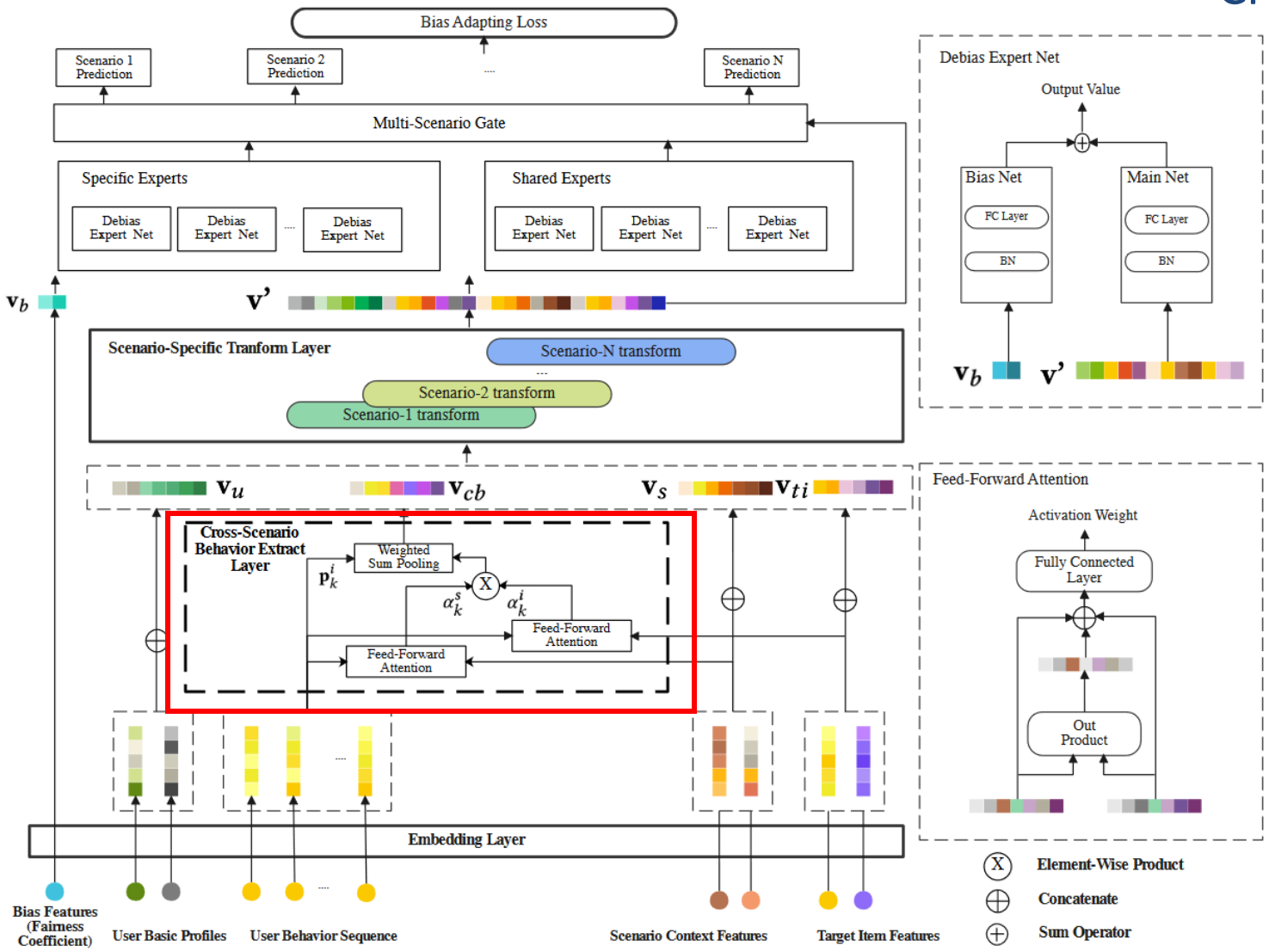
- Traffic characteristics of different scenarios are significantly different (individual data scale or representative topic)

➢Target

- Train a unified model to serve all scenarios

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

SAR-Net Details



Cross-Scenario Behavior Extract Layer

How to aggregate the sequence?

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

$$\mathbf{p}(B^i) = \{\mathbf{p}_1^i, \mathbf{p}_2^i, \cdots, \mathbf{p}_{|\mathbf{p}(B^i)|}^i\}$$

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

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

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

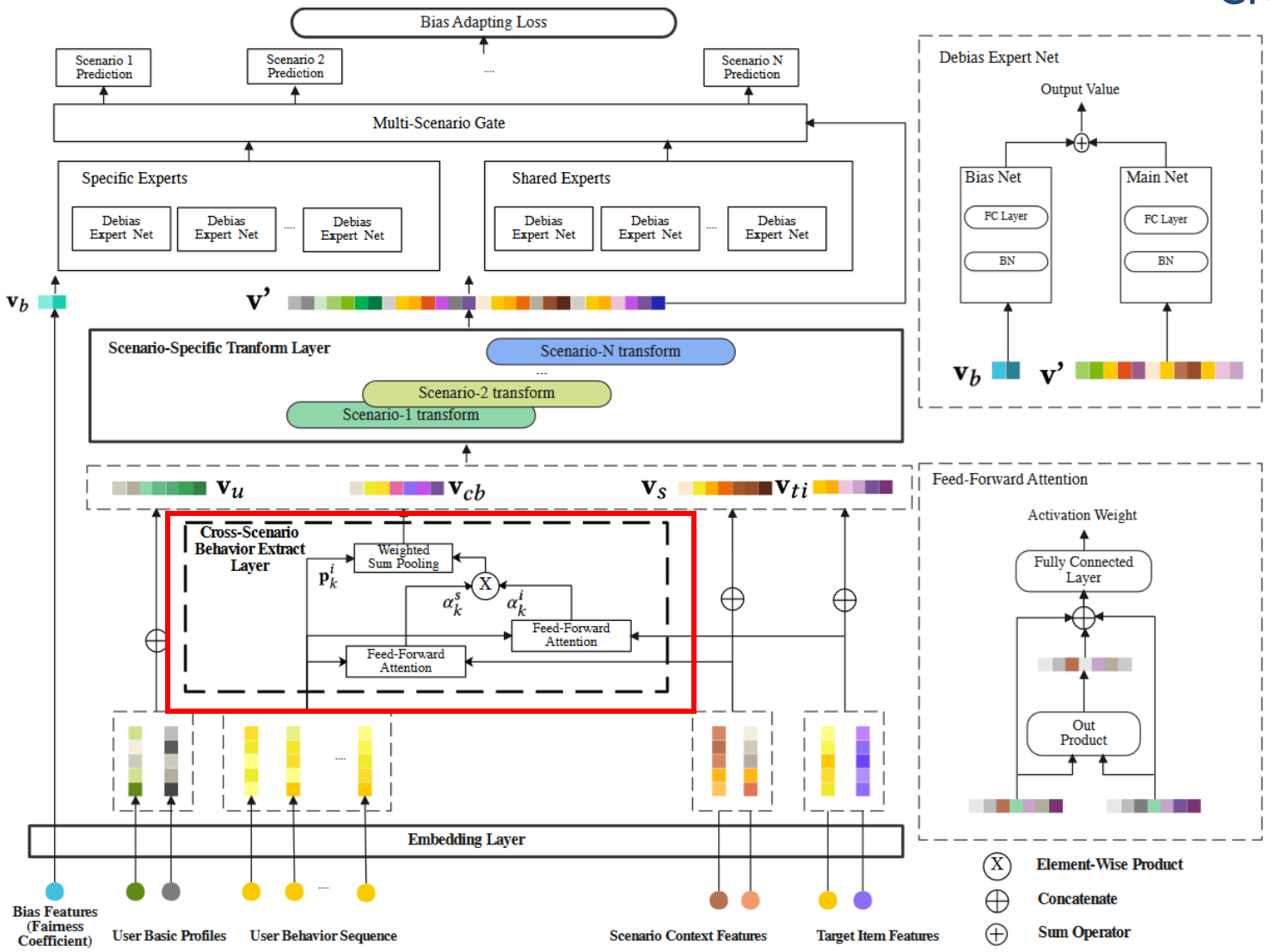
$$\mathbf{p}(B^s) = \{\mathbf{p}_1^s, \mathbf{p}_2^s, \cdots, \mathbf{p}_{|\mathbf{p}(B^s)|}^i\}$$

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

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

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

SAR-Net Details



Cross-Scenario Behavior Extract Layer

How to aggregate the sequence?

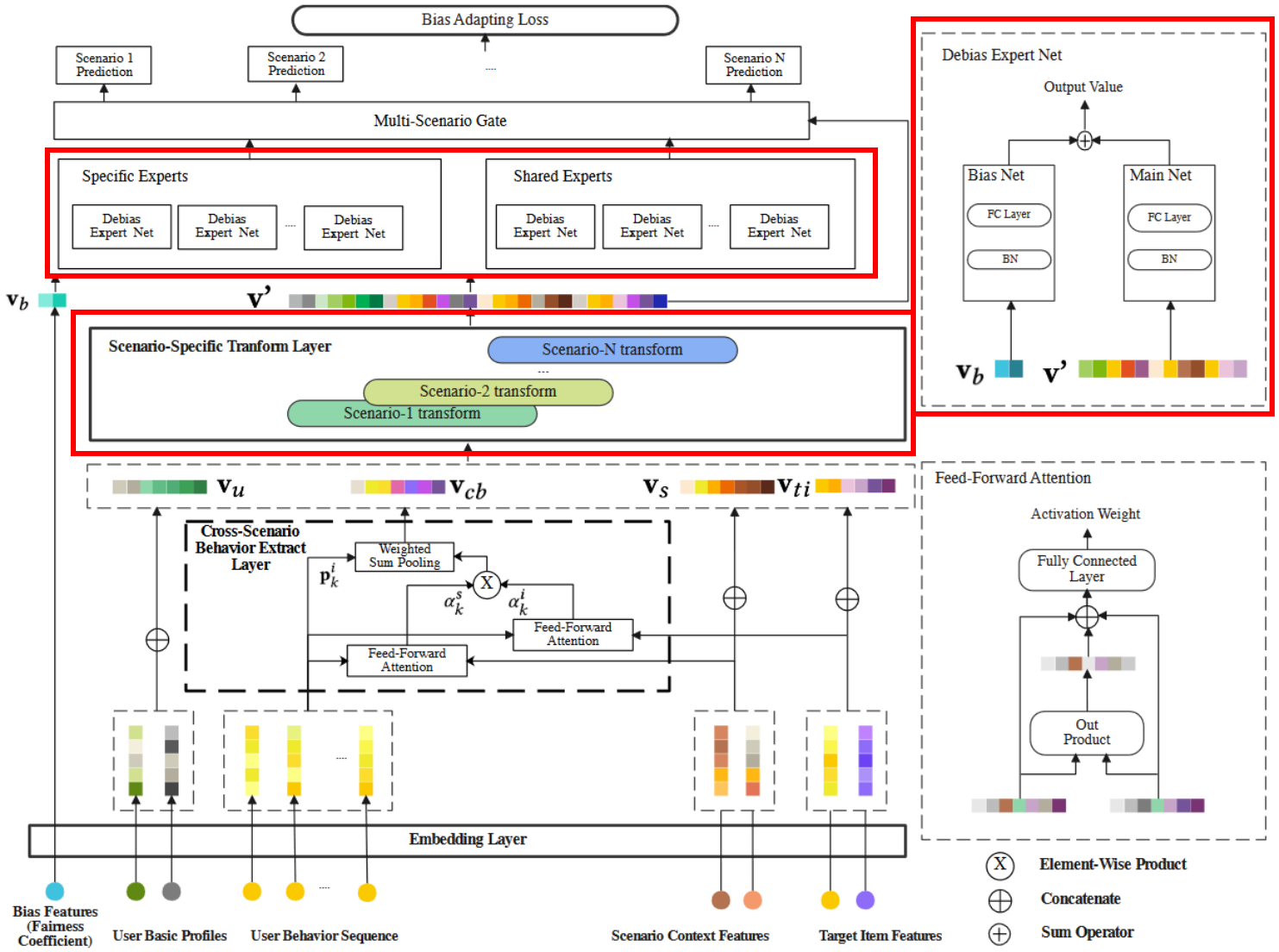
$$\alpha_k^i = \frac{\exp(\psi(\mathbf{p}_k^i, \mathbf{p}_t^i))}{\sum_{l=1}^{|\mathbf{p}(B^i)|} \exp(\psi(\mathbf{p}_l^i, \mathbf{p}_t^i))},$$

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

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

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

SAR-Net



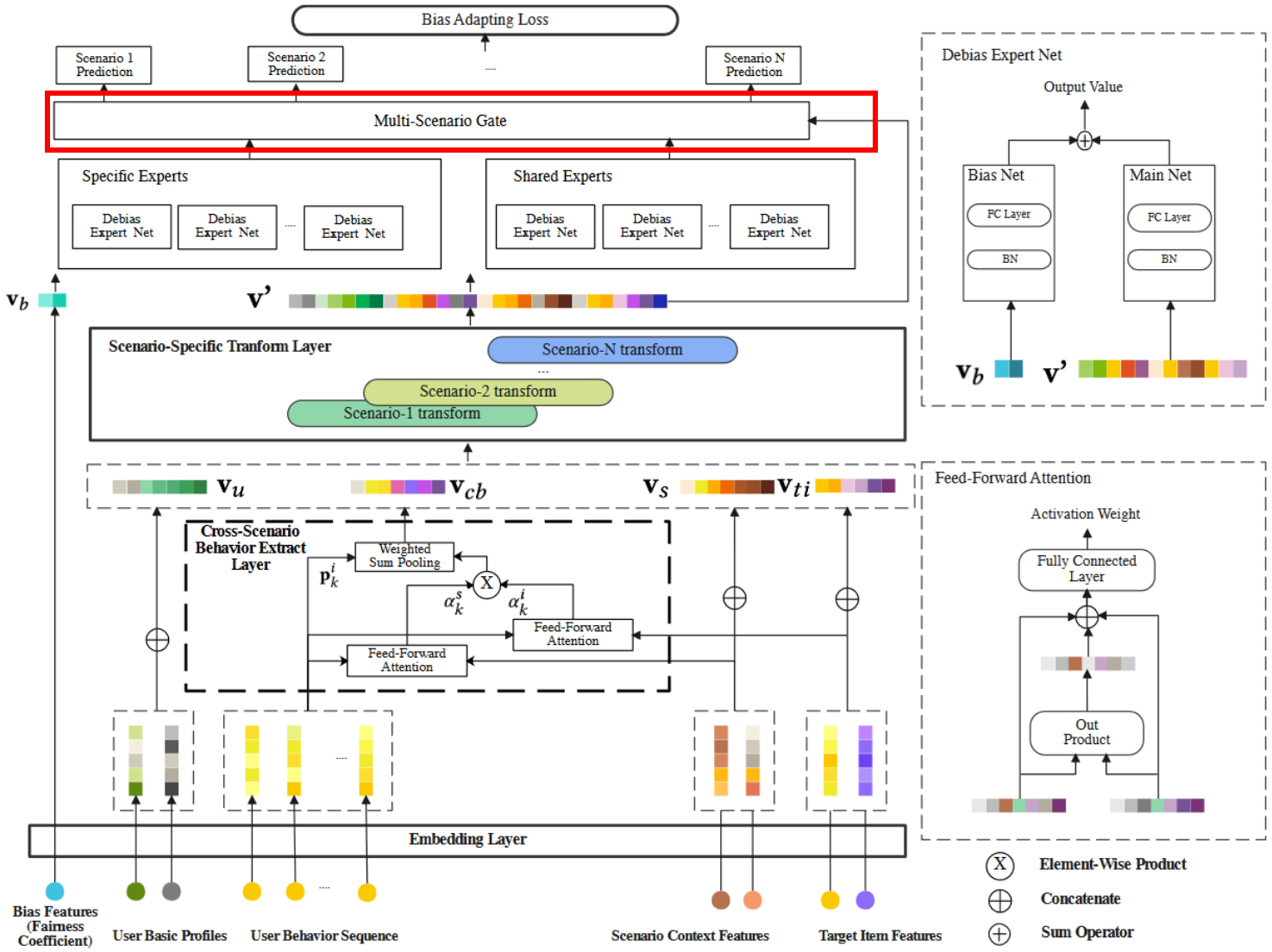
Scenario-Specific Transform Layer

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

Mixture of Debias Experts

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

SAR-Net



Multi-Gate Network & Prediction

The output of experts:

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

Final predicted score of scenario k

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

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

ADI

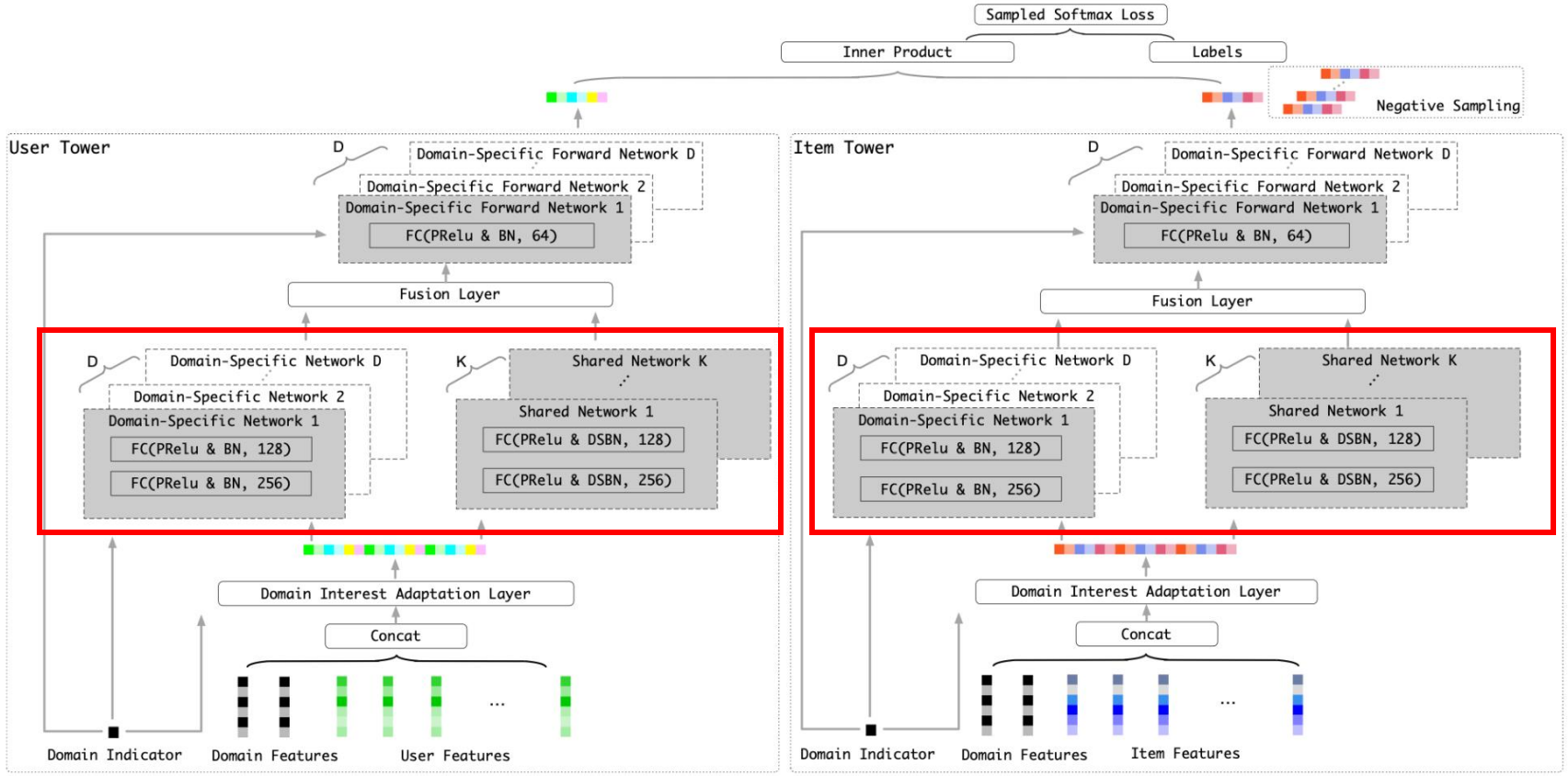
- Motivation
  - Separate model for each scenario, ignoring the cross-domain overlapping of user groups and items
  - One shared model trained on mix data, model performance may decrease when different domains conflict

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

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

ADI Details

Backbone Network



Shared Network & Domain-Specific Network

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

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

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

f_{domain} Domain indicator embedding

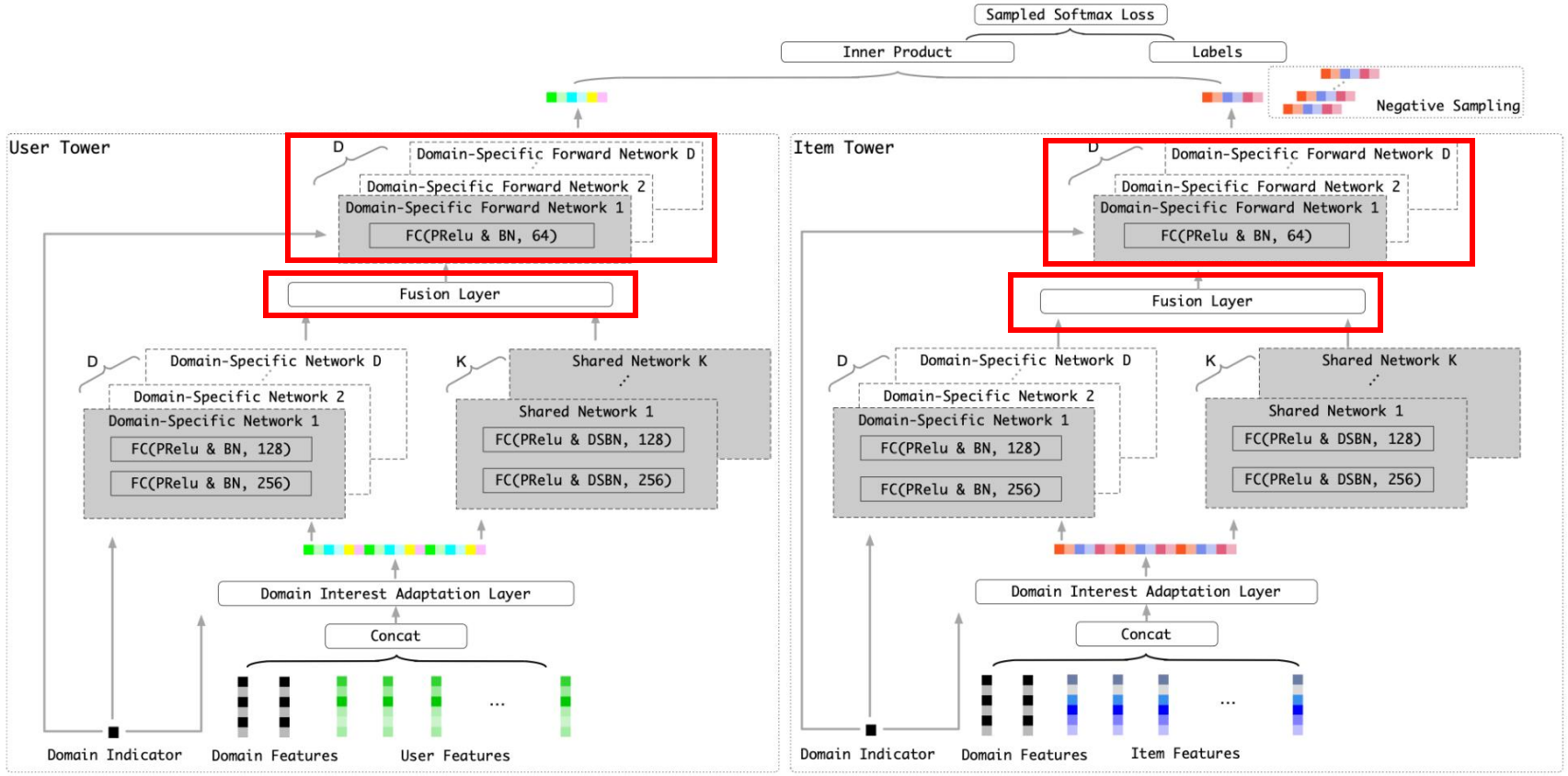
$\mathbf{F}^{(d)}$ Data from domain d

K hyperparameter,
number of Shared Network

D domains, D Domain-Specific Network

ADI Details

Backbone Network



Fusion Layer

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

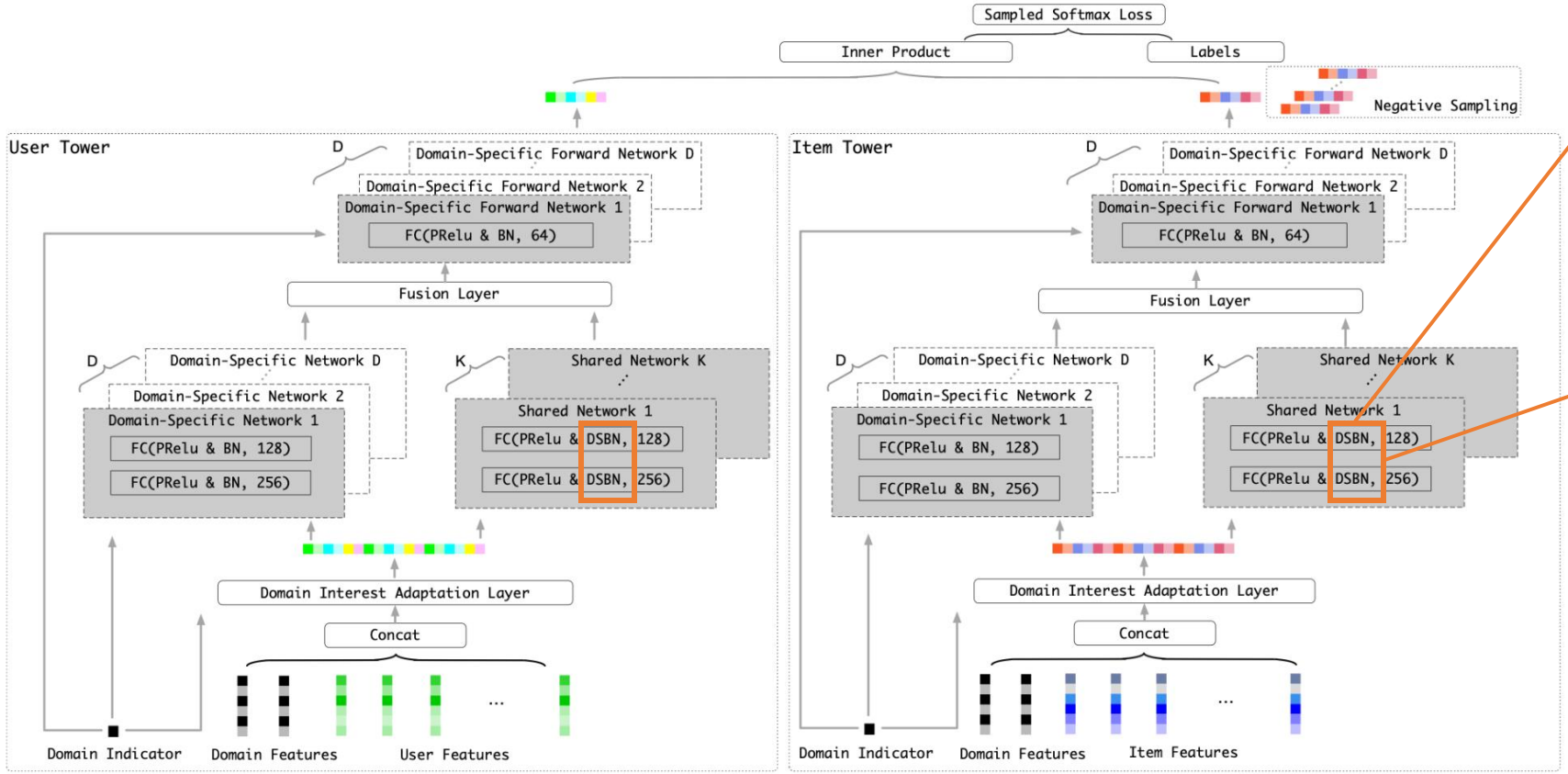
$$E_{fusion}^{(d)} = concat(\beta_1^{(d)} E_{spec}^{(d)} \mid \beta_1^{(d)} E_{spec}^{(d)} \odot \beta_2^{(d)} E_{shared} \mid \beta_2^{(d)} E_{shared})$$

Domain-Specific Forward Network

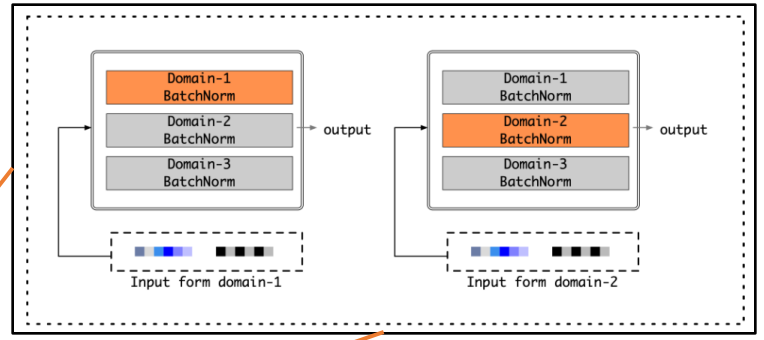
$$E = FC_{forward}^{(d)}(E_{fusion}^{(d)})$$

ADI Details

Domain Adaptation



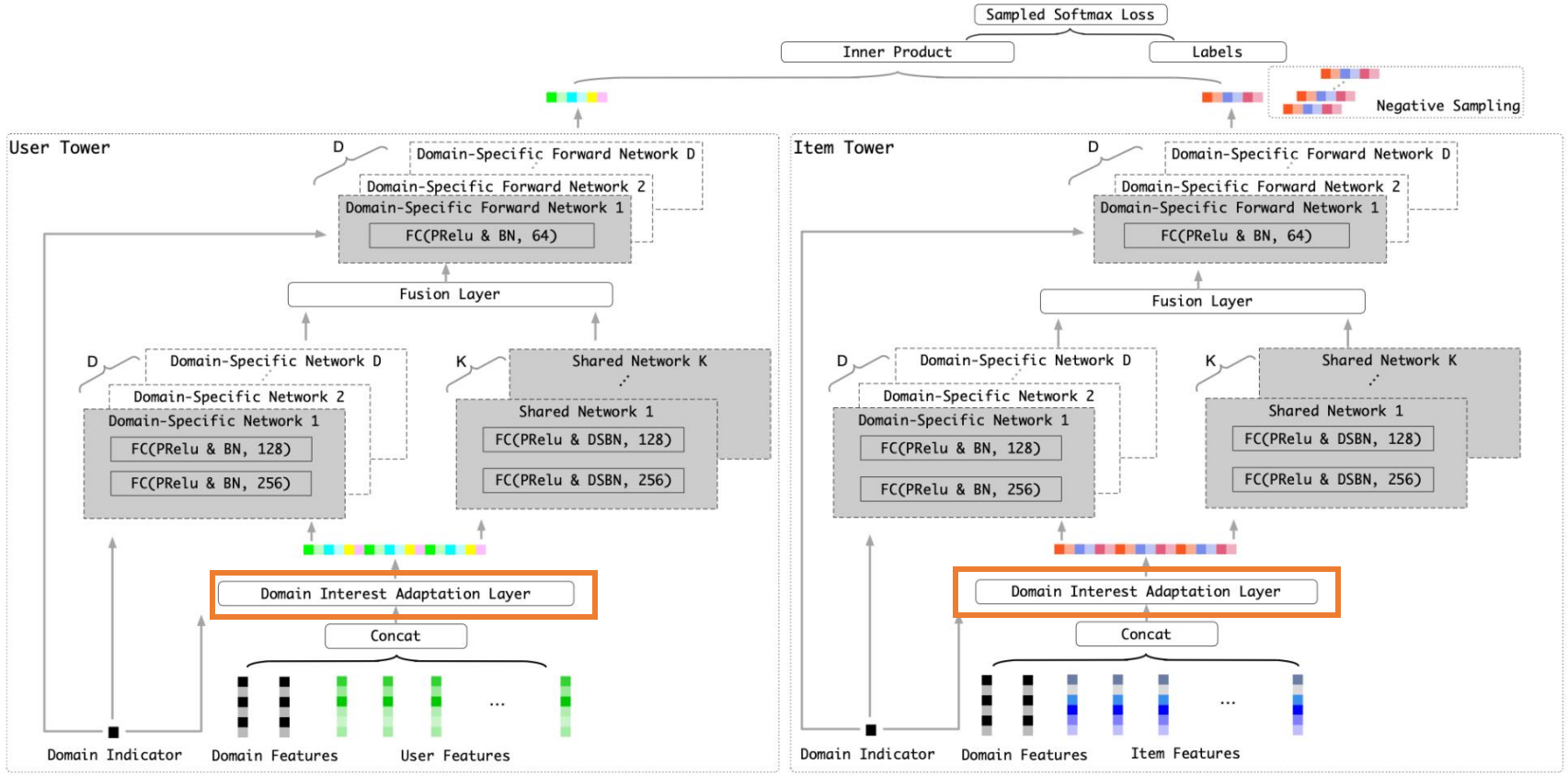
Domain-Specific Batch Normalization (DSBN)



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

ADI Details

Domain Adaptation



Domain Interest Adaptation Layer

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

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

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

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

Uni-CTR

➢Motivation

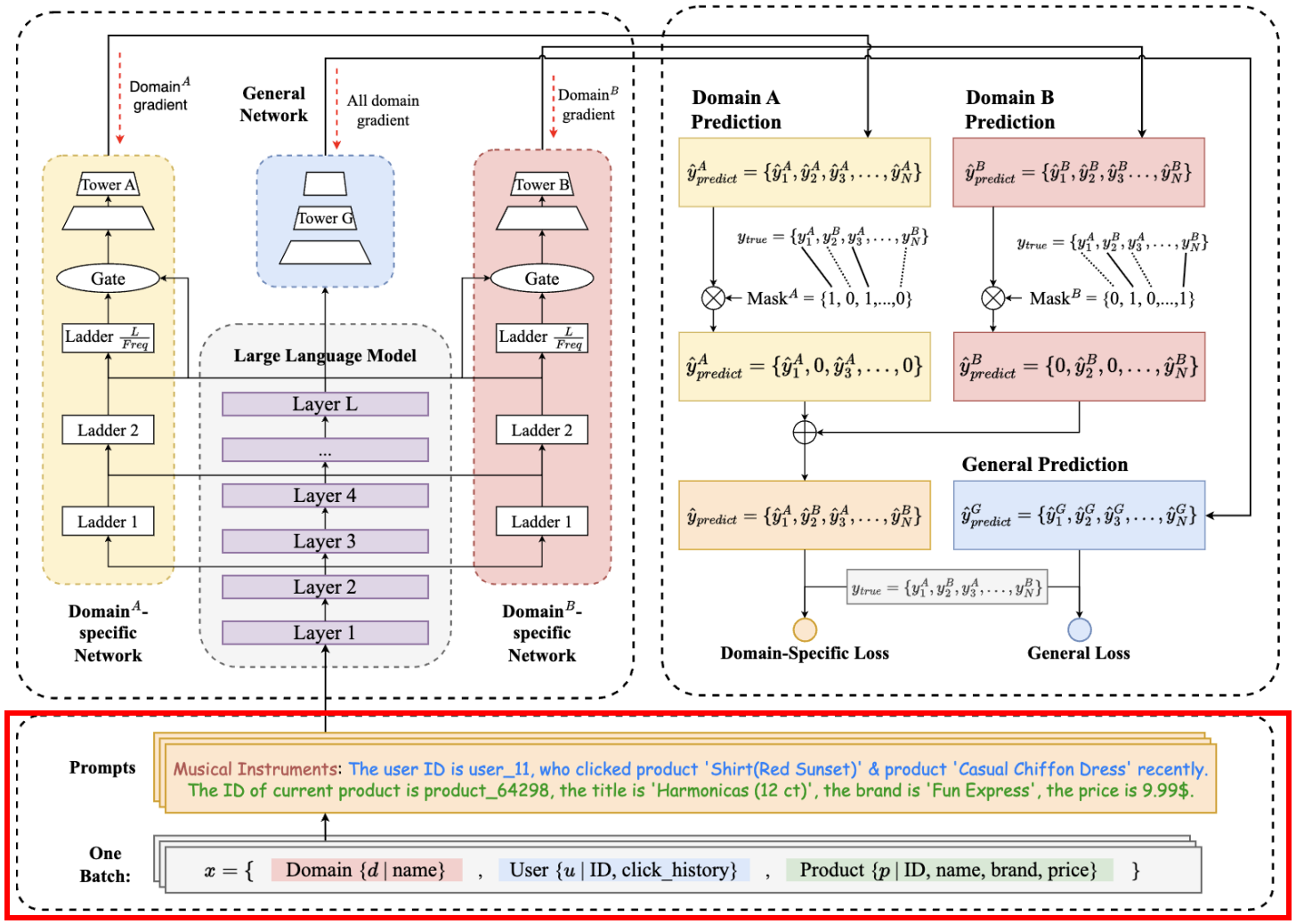
- Due to varying data sparsity in different domains, models can easily be dominated by specific domains, leading to “seesaw phenomenon”
- Existing methods are difficult to handle newly added domain

➢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

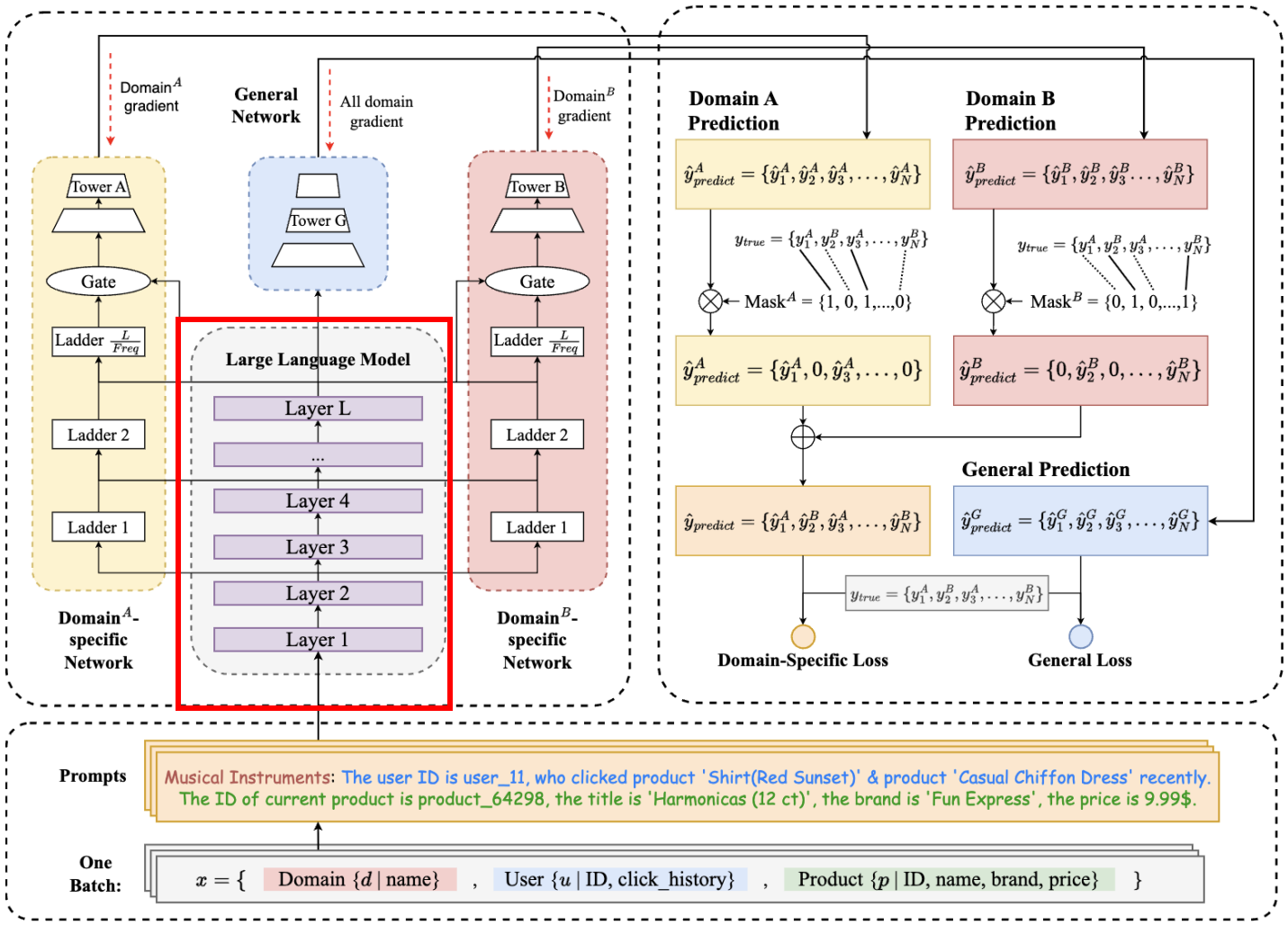
A Unified Framework for Multi-Domain CTR Prediction via Large Language Models. ToIS 24.

Uni-CTR Details



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

Uni-CTR Details



Uni-CTR architecture

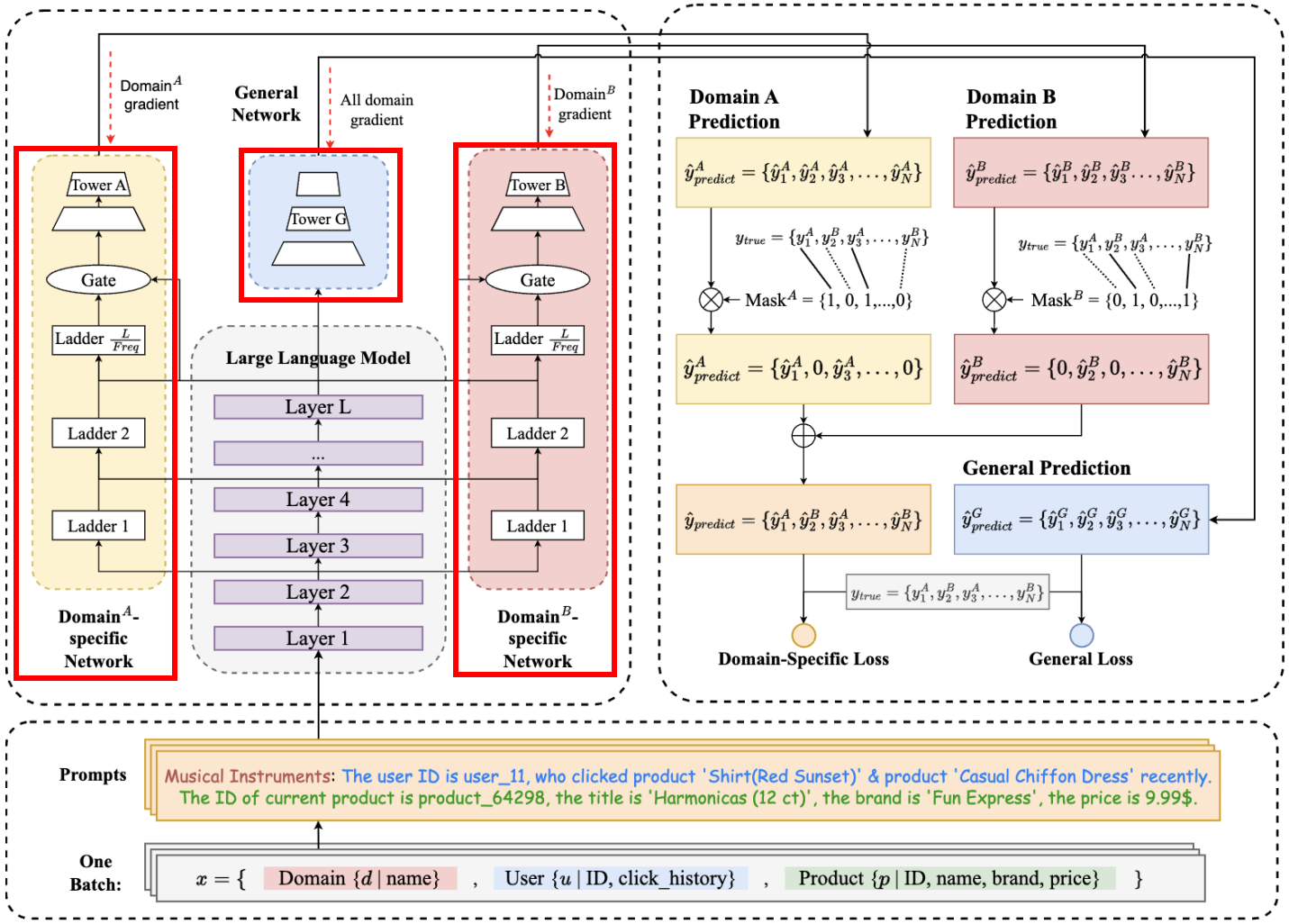
- LLM Backbone

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

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

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

Uni-CTR Details



Uni-CTR architecture

- LLM Backbone

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

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

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

- Domain-Specific Network
  - Ladder Netowork
  - Gate Net
  - Tower Net
- General Network

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

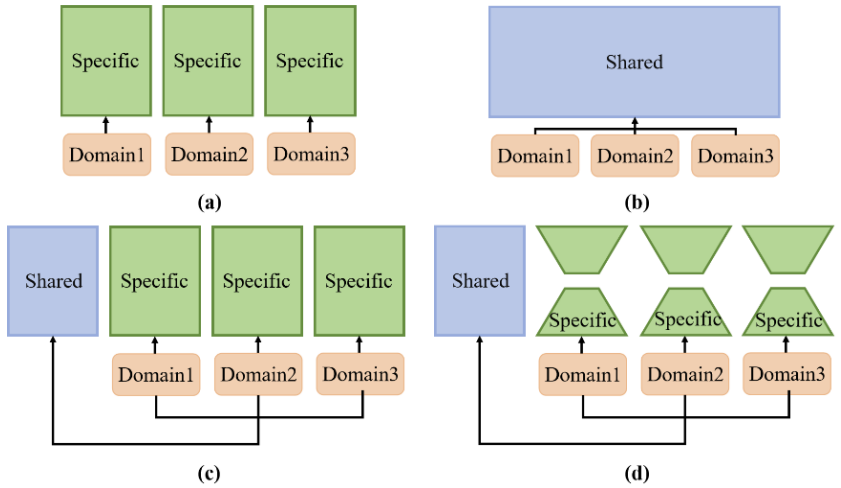
M-LoRA

➢Motivation

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

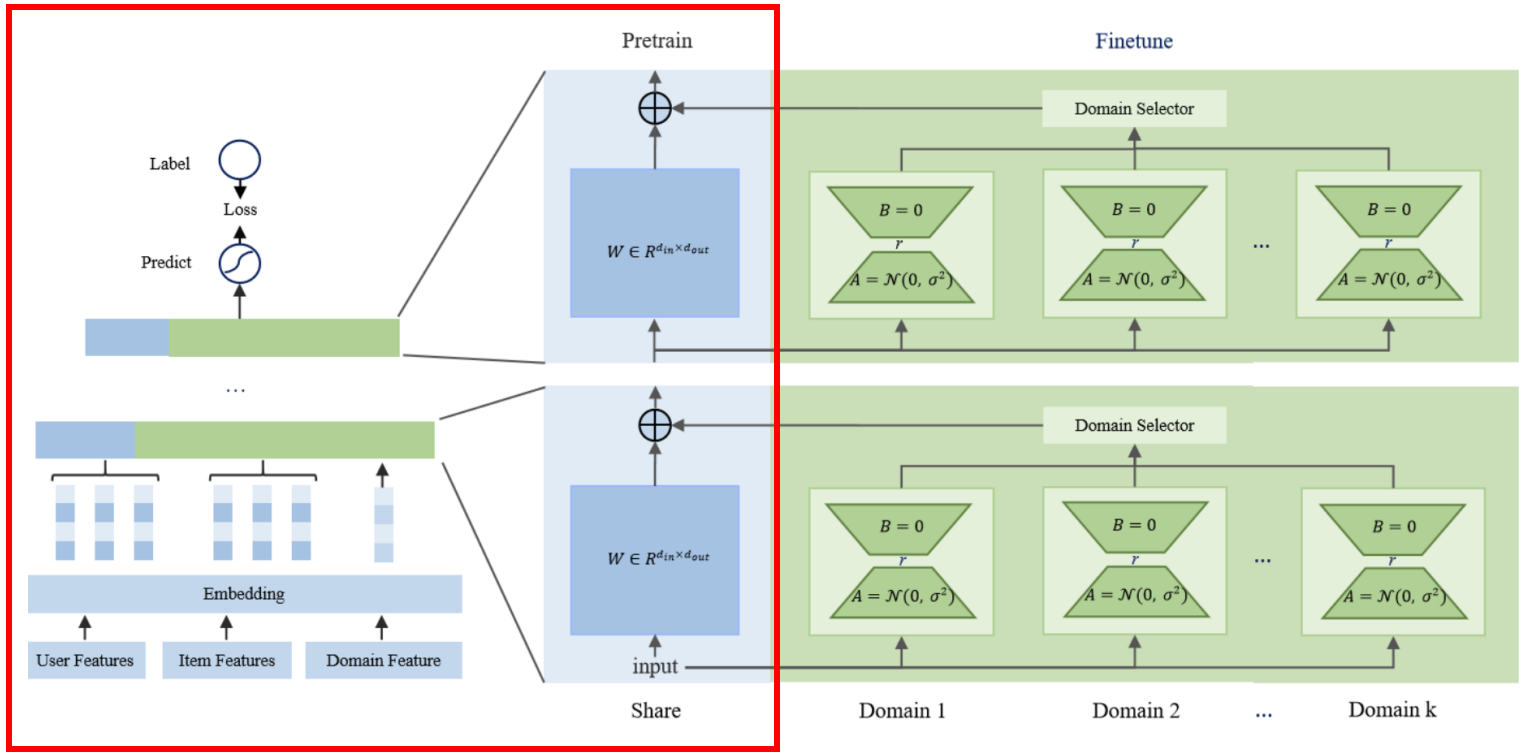
➢Method

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



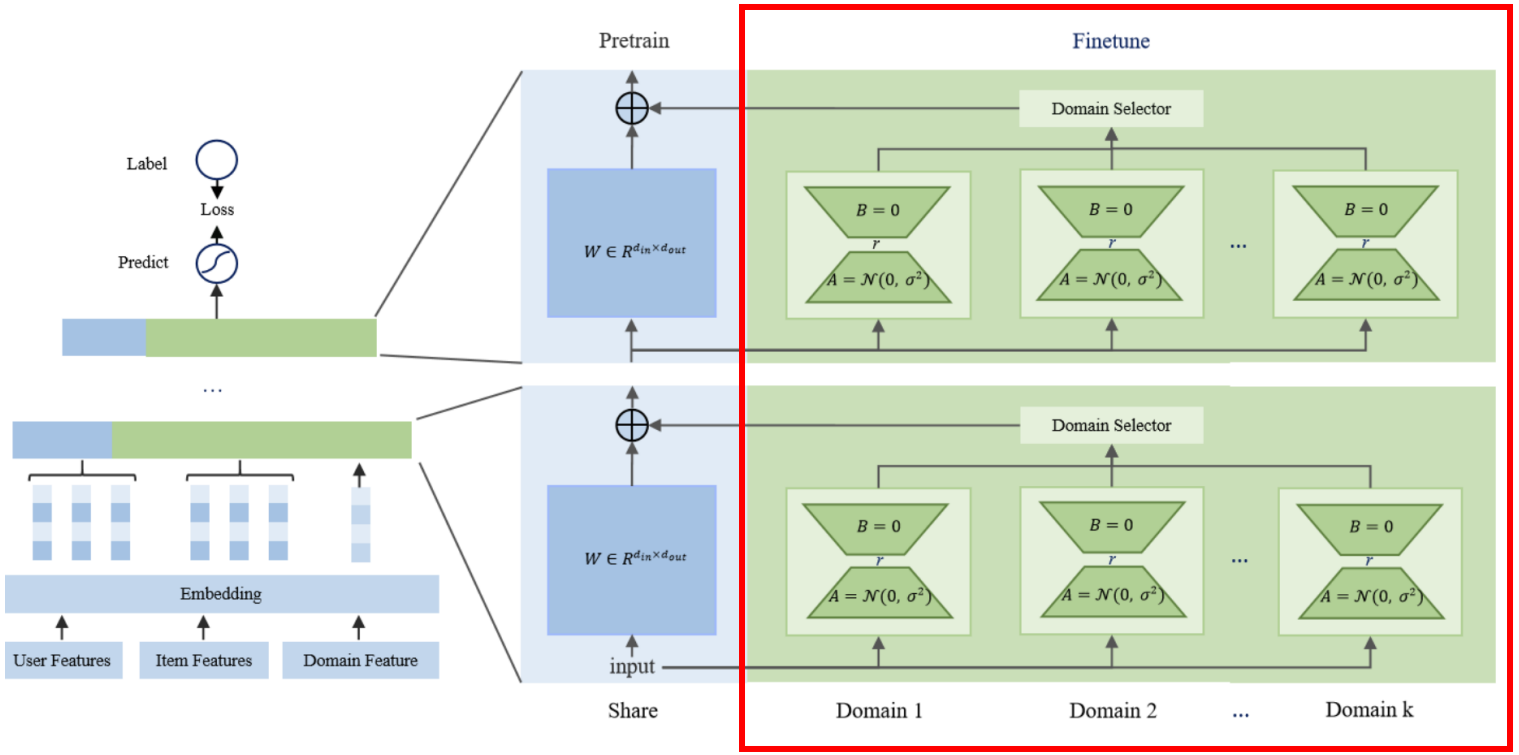
MLoRA: Multi-Domain Low-Rank Adaptive Network for CTR Prediction. RecSys 24.

M-LoRA Details



- Pre-training
  - Lage-scale pre-training dataset
  - Shared Network and Embedding layer

M-LoRA Details



- Pre-training
  - Lage-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

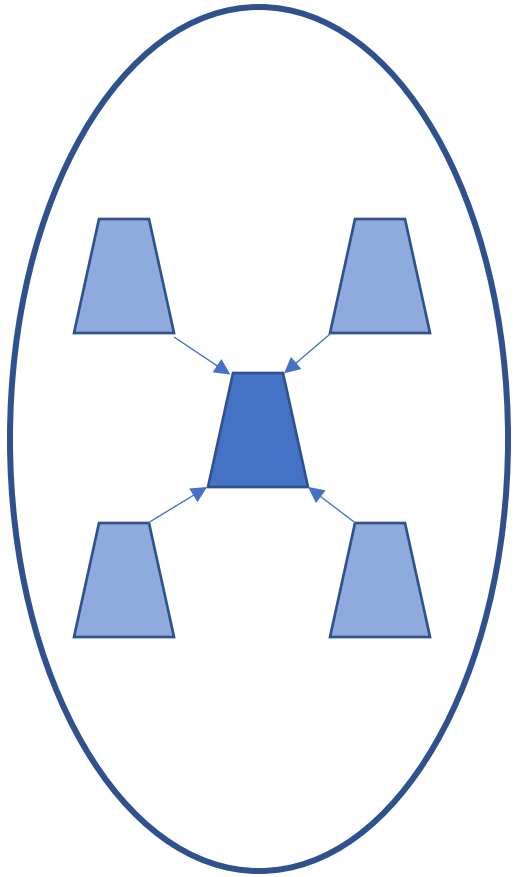
$$\mathbf{h}_t = \mathbf{W}\mathbf{x} + \Delta\mathbf{W}_t = \mathbf{W}\mathbf{x} + \mathbf{B}_t\mathbf{A}_t\mathbf{x},$$

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

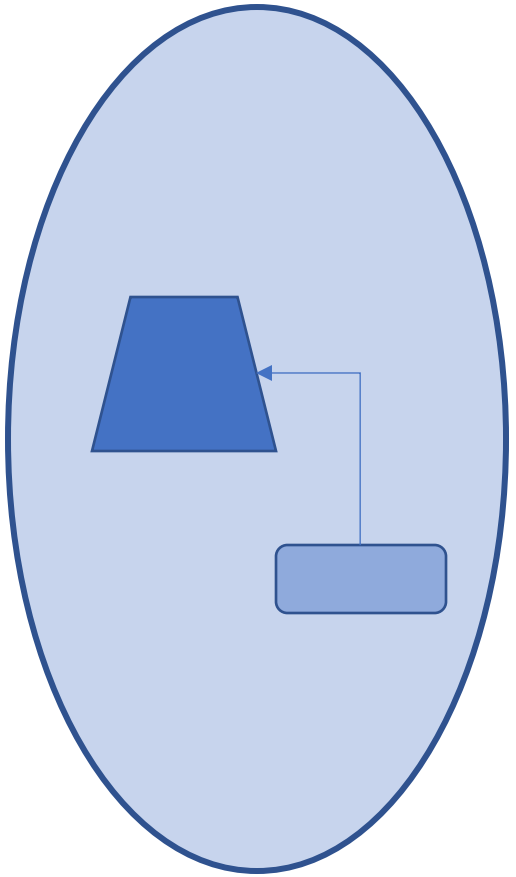
$$r = \max\left(\frac{d_{out}}{\alpha}, 1\right)$$

Table of Contents



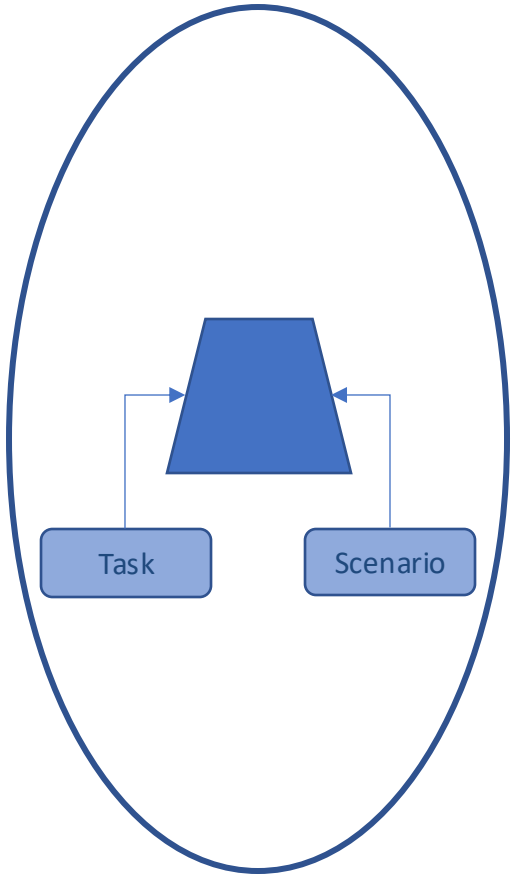
Shared-specific network paradigm

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



Dynamic weight

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

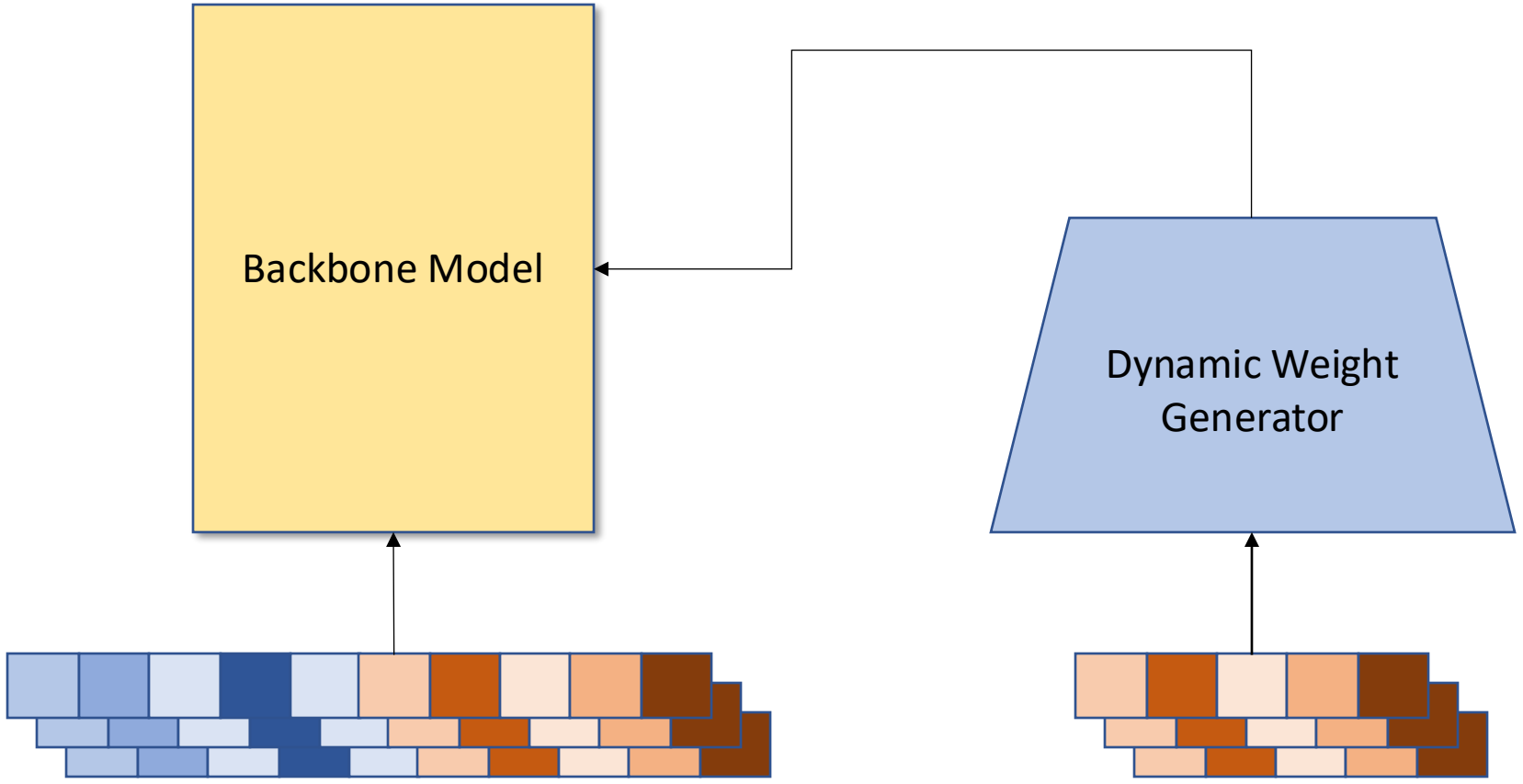


Multi-Scenario & Multi-Task

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

Dynamic Weight

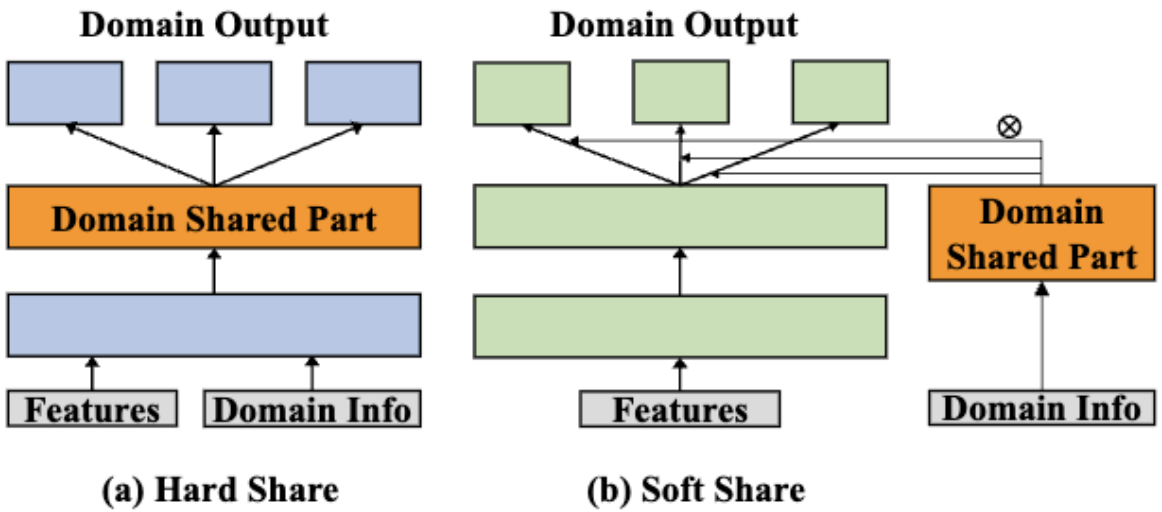
- Why Dynamic?



HAMUR

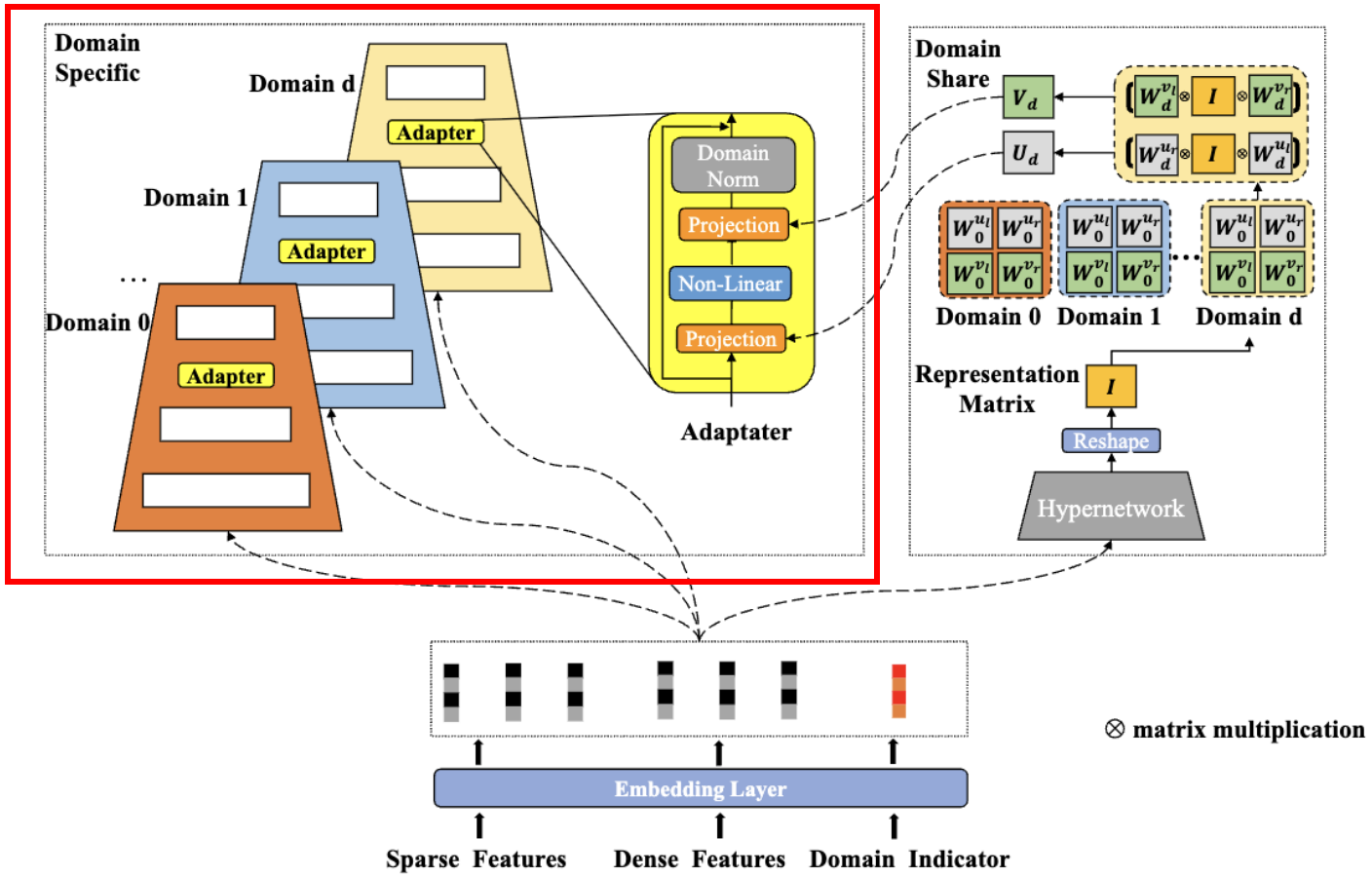
- Motivation
  - Previous research relies on explicit sharing across different domains
  - Static parameters constrain the representation of different domains

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



HAMUR: Hyper Adapter for Multi-Domain Recommendation. CIKM 2023

HAMUR



Domain-specific adapter

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

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

Domain Shared Hyper-Network

- Parameters Generation

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

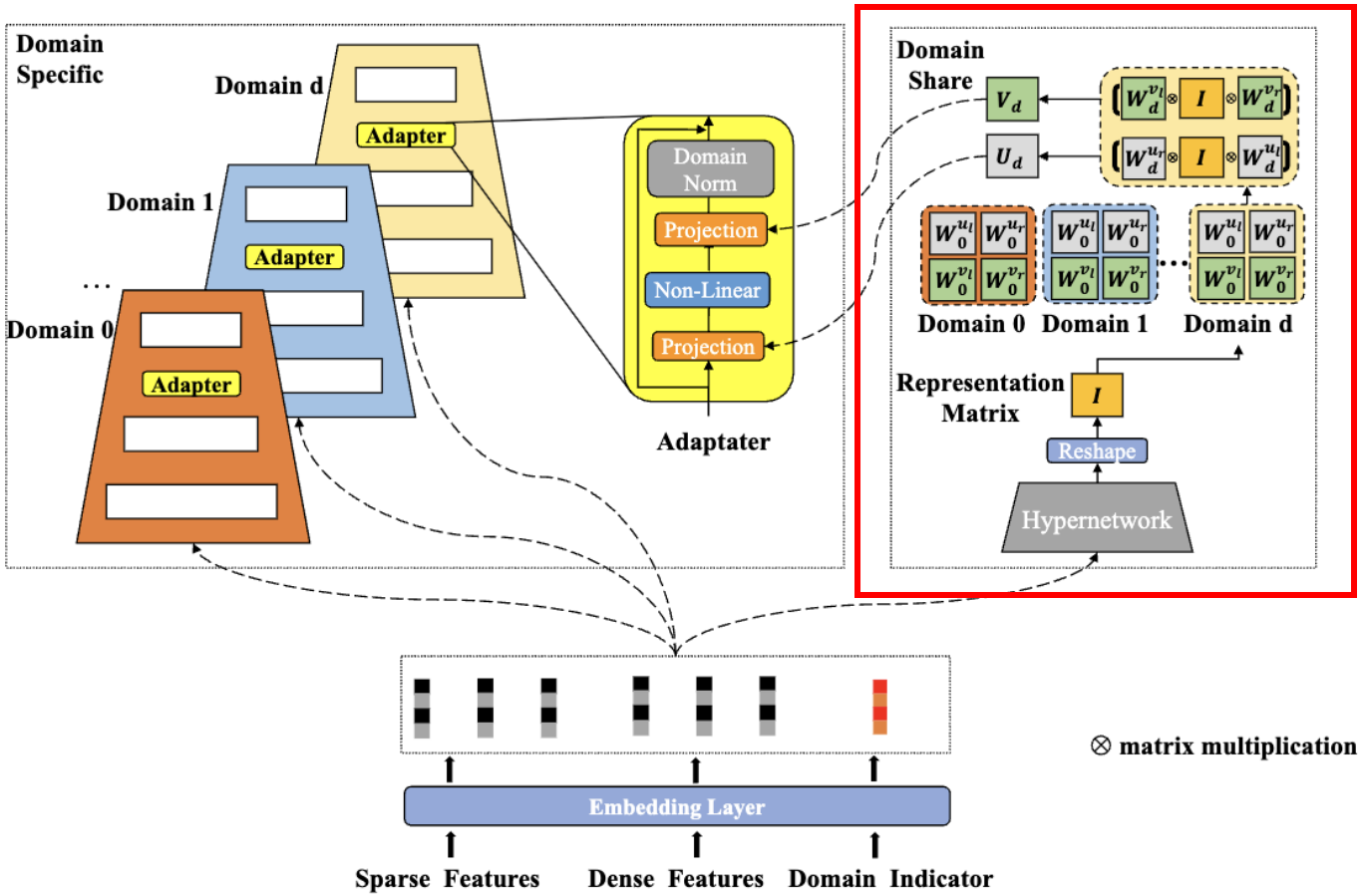
$$I^i = reshape(h^i)$$

- Low-Rank Decomposition

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

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

HAMUR



Domain-specific adapter

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

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

Domain Shared Hyper-Network

- Parameters Generation

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

$$I^i = reshape(h^i)$$

- Low-Rank Decomposition

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

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

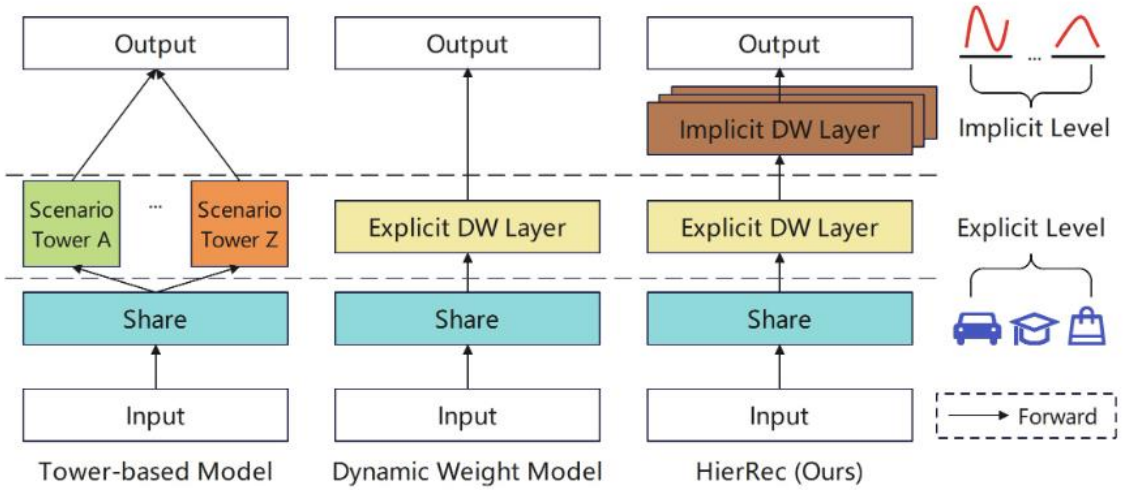
HierRec

➢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

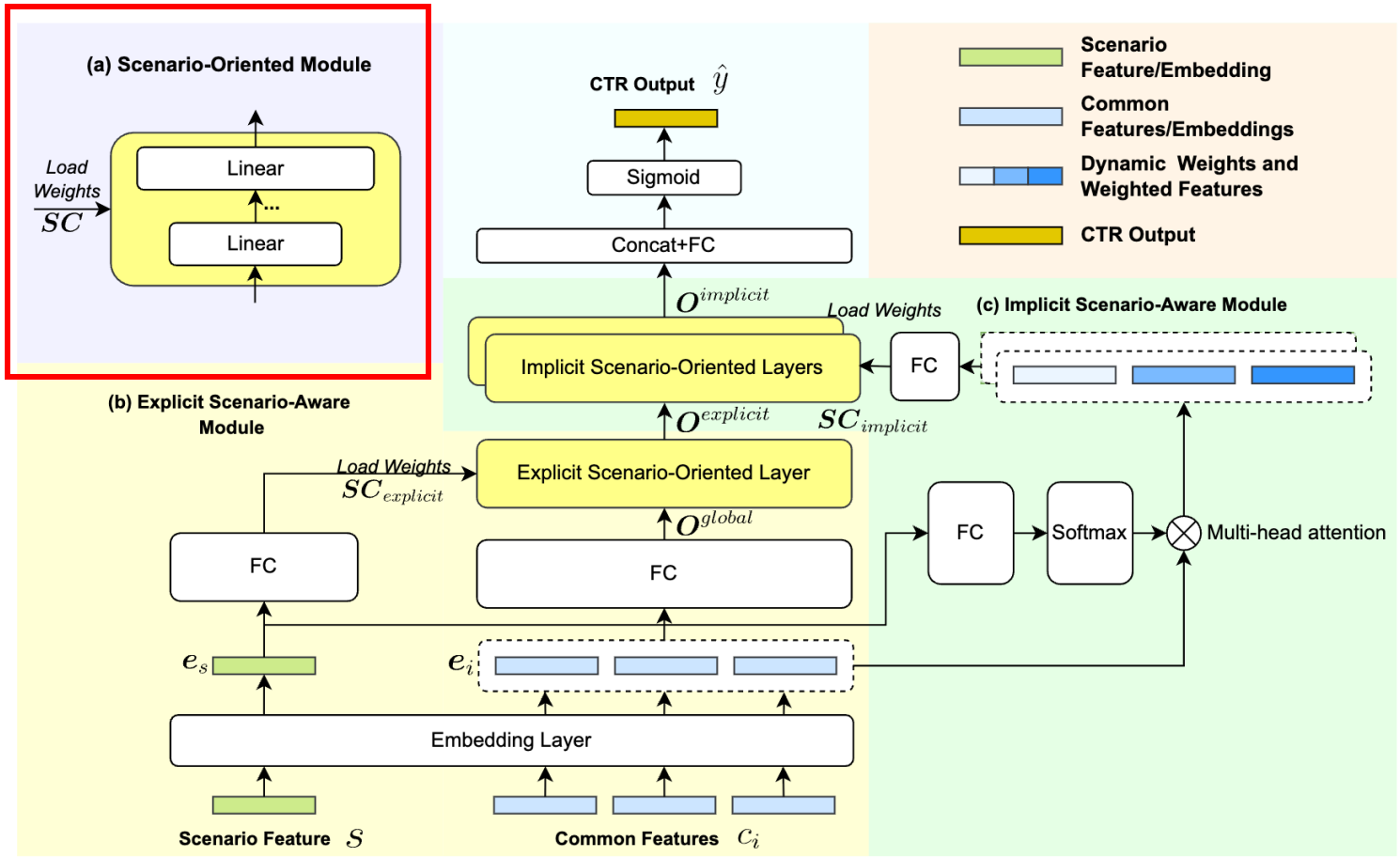
➢Method

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



HierRec: Scenario-Aware Hierarchical Modeling for Multi-scenario Recommendations. CIKM 24.

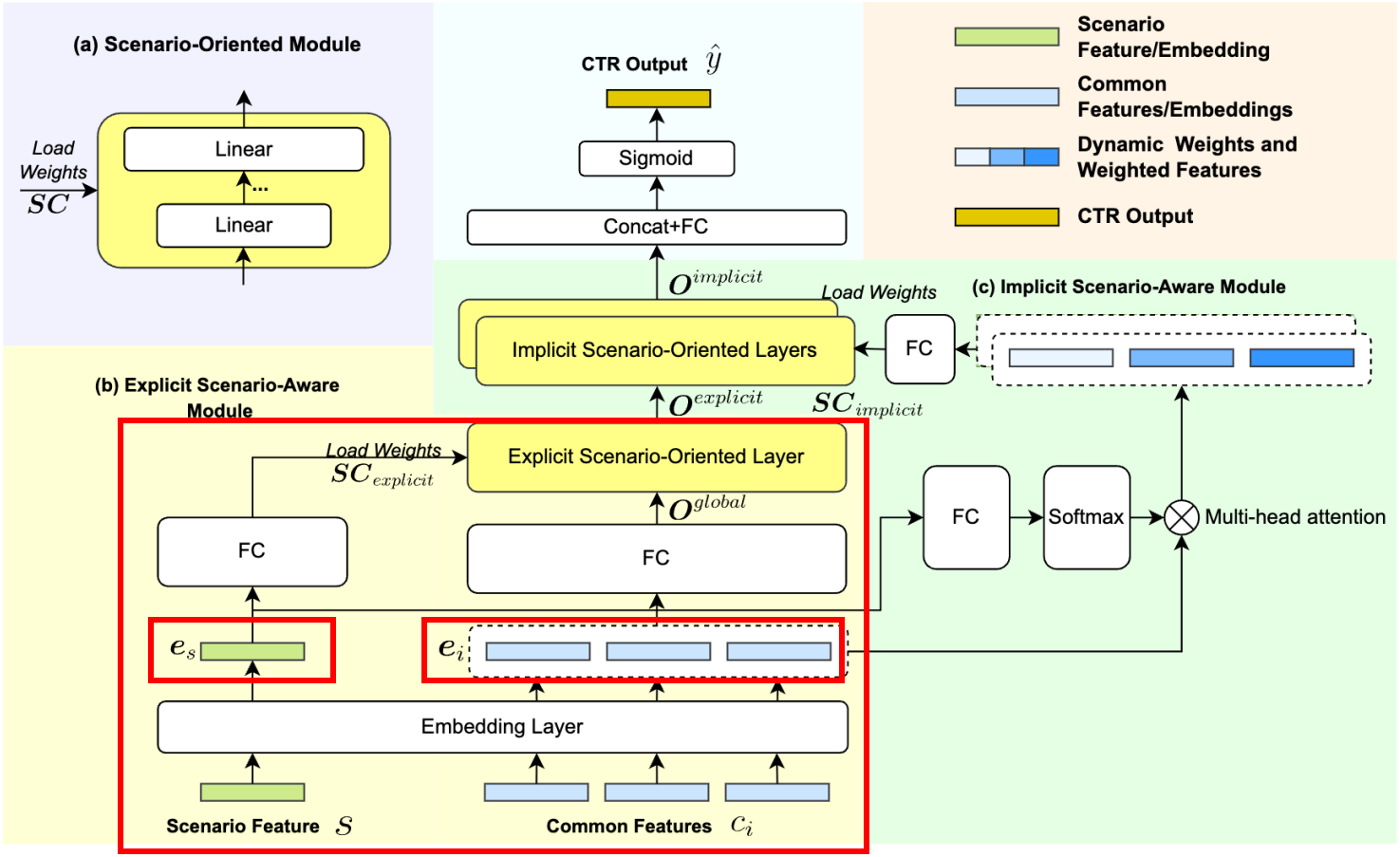
HierRec



- Scenario-Oriented Module
  - Adaptively generate parameters depending on scenario condition (SC)

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

HierRec



- Scenario-Oriented Module
  - Adaptively generate parameters depending on scenario condition (SC)

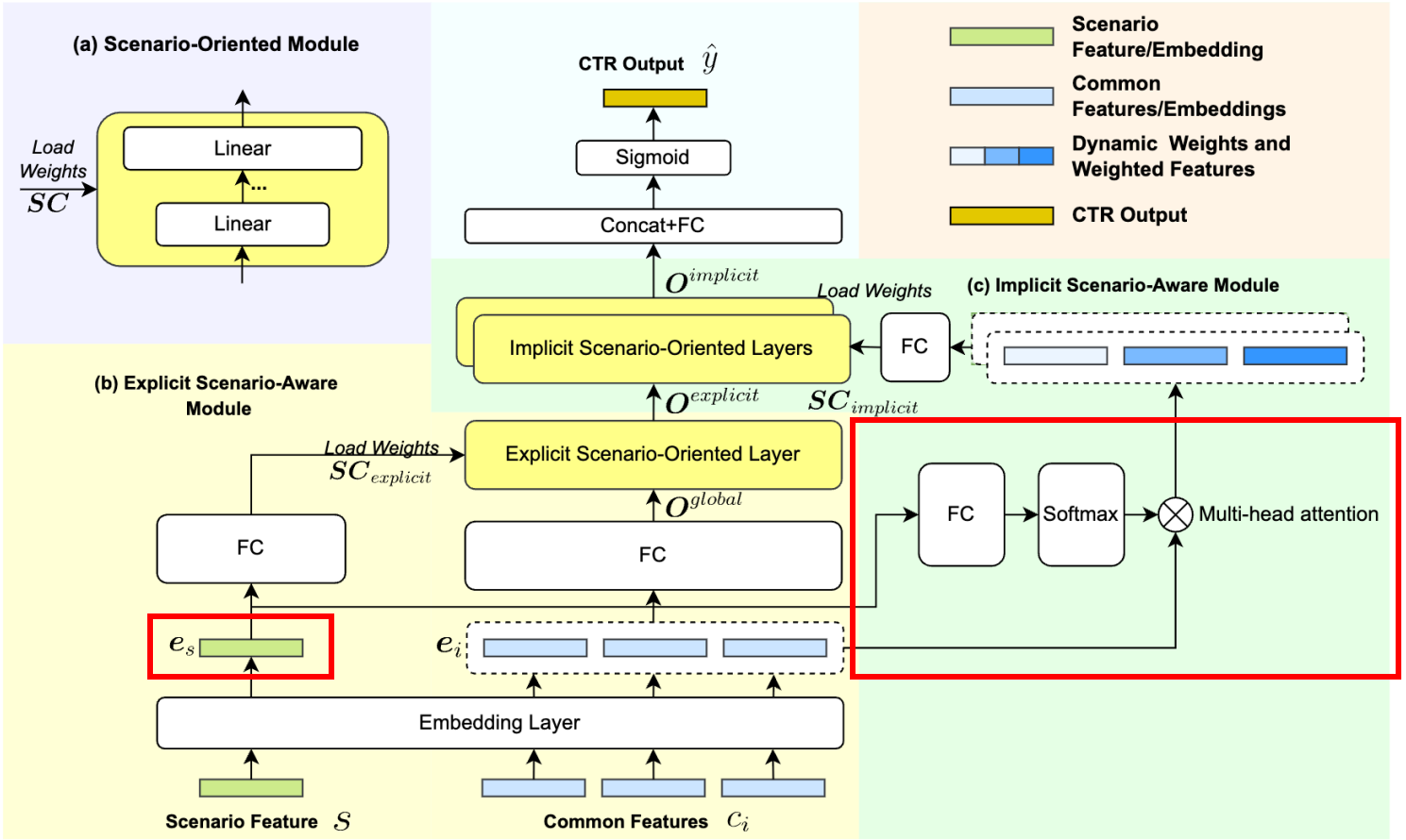
$$\boldsymbol{W}_l, \boldsymbol{b}_l = Reshape(\boldsymbol{SC})[l] \quad l \in [1, L],$$

- Explicit Scenario-Aware Module
  - Model coarse-grained explicit scenario information

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

$$\boldsymbol{SC}_{explicit} = FC(\boldsymbol{e}_s)$$

HierRec



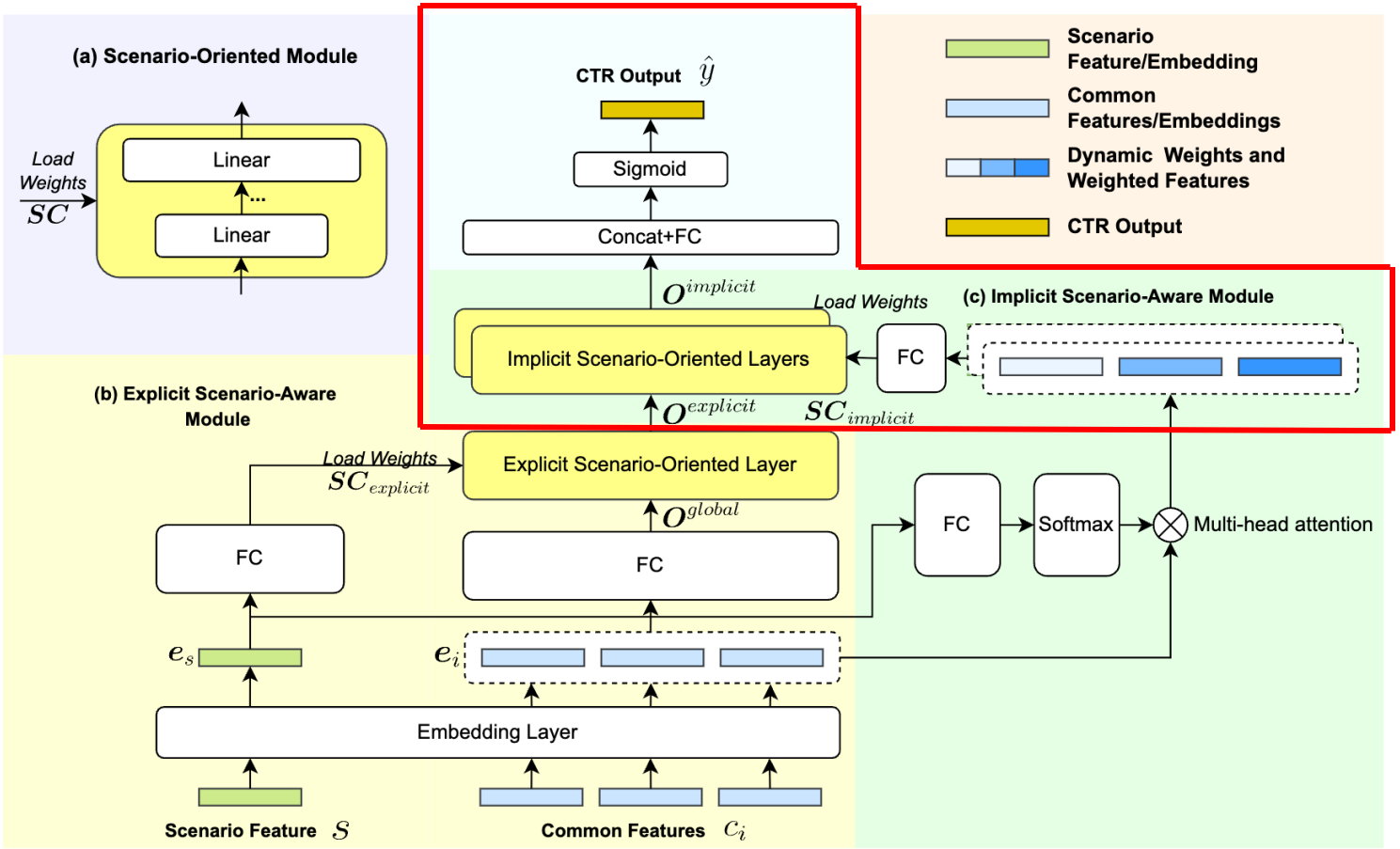
Implicit Scenario-Aware Module

- Model fine-grained implicit scenario information

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

$$\boldsymbol{IE} = \boldsymbol{weight}_{norm} \otimes \boldsymbol{E}_c,$$

HierRec



Implicit Scenario-Aware Module

- Model fine-grained implicit scenario information

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

$$\boldsymbol{IE} = \boldsymbol{weight}_{norm} \otimes \boldsymbol{E}_c,$$

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

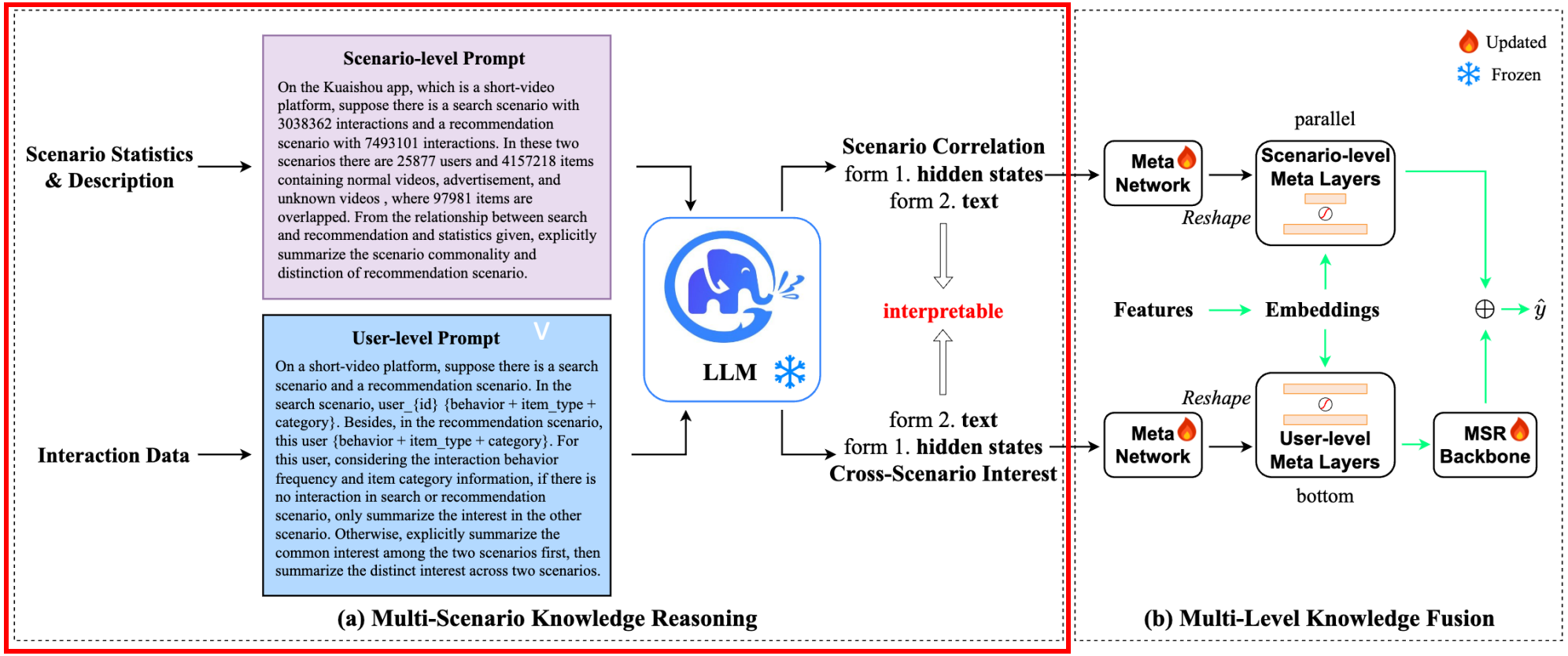
$$\hat{y} = Sigmoid(FC(Concat(\boldsymbol{O}_1^{implicit}, ..., \boldsymbol{O}_G^{implicit}))).$$

LLM4MSR

- 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

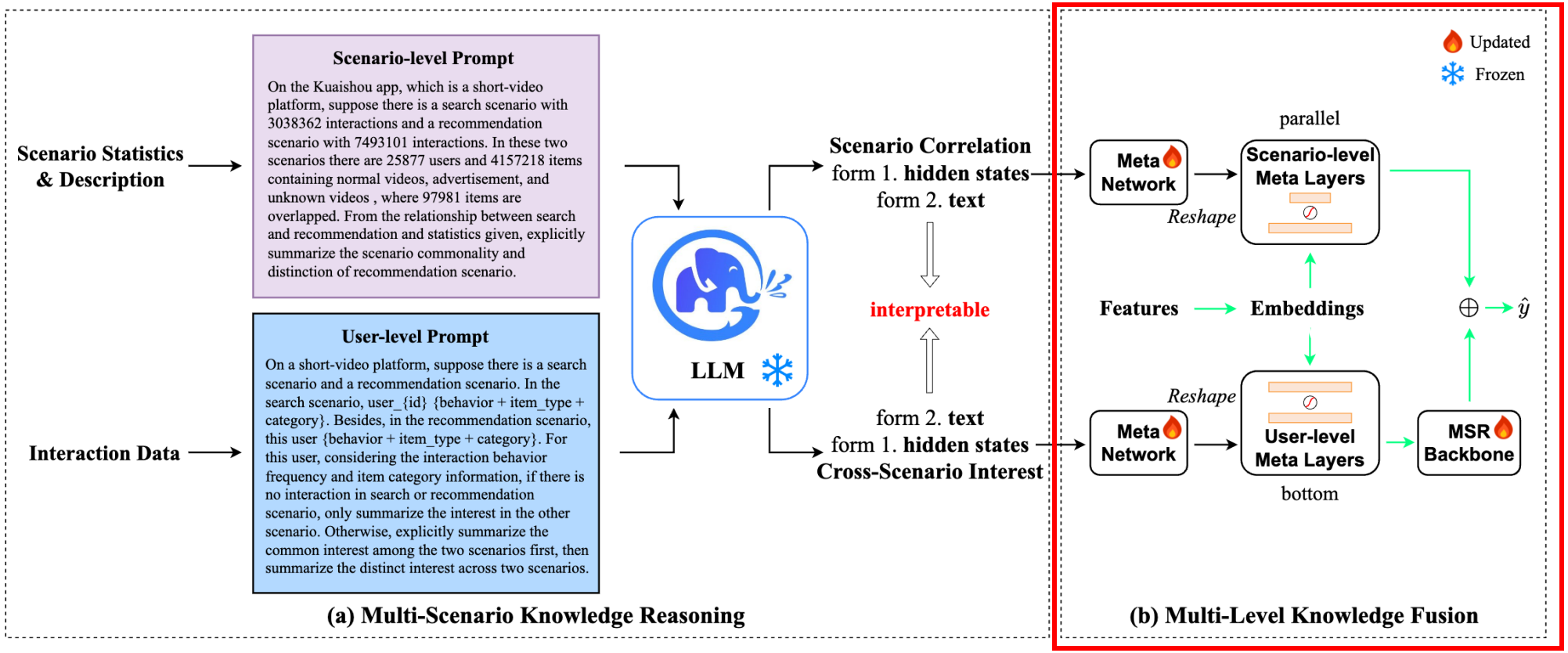
LLM4MSR: An LLM-Enhanced Paradigm for Multi-Scenario Recommendation. CIKM 24.

LLM4MSR Details



- Multi-scenario Knowledge Reasoning
  - Scenario-level prompt construction
  - User-level prompt construction

LLM4MSR Details



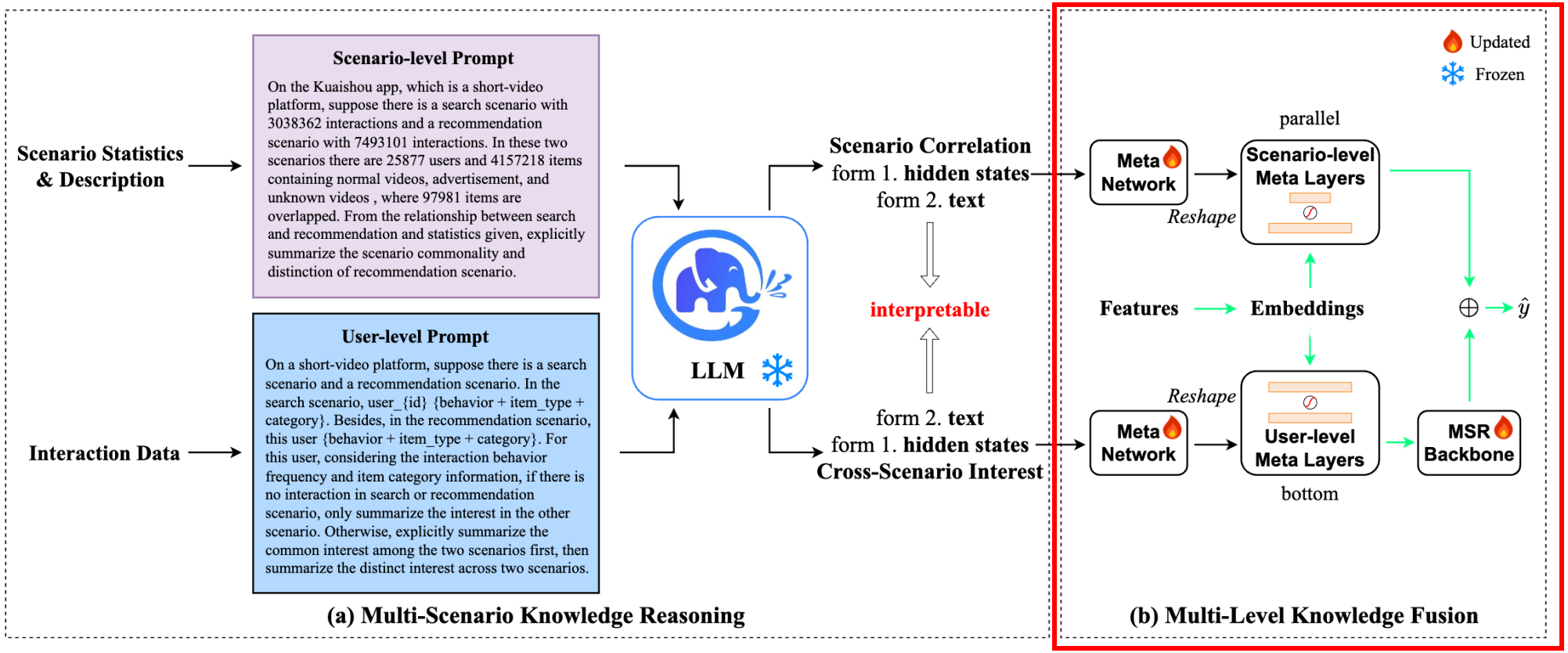
- Multi-Level Knowledge Fusion
  - Meta-net generates meta layers to fuse the scenario- and user-level knowledge

$$\boldsymbol{h}_{mw}, \boldsymbol{h}_{mb} = \text{Meta Network}(\boldsymbol{h}_{LLM}),$$

$$\boldsymbol{W}_l^{(i)} = Reshape(\boldsymbol{h}_{mw})$$

$$\boldsymbol{b}_l^{(i)} = Reshape(\boldsymbol{h}_{mb}), i \in \{1, 2, \ldots, K\}$$

LLM4MSR Details



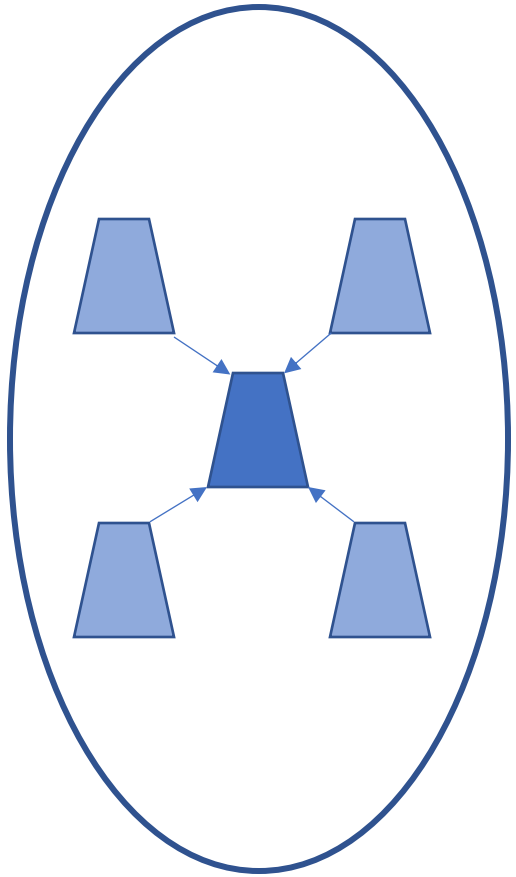
- Multi-Level Knowledge Fusion
  - Prediction

$$\boldsymbol{h}^{(i)} = \sigma(\boldsymbol{W}_l^{(i)}\boldsymbol{h}^{(i-1)} + \boldsymbol{b}_l^{(i)}), i \in \{1, 2, \ldots, K\}$$

$$\boldsymbol{h} = \mathrm{MSR}(\boldsymbol{h}_u^{(K)}),$$

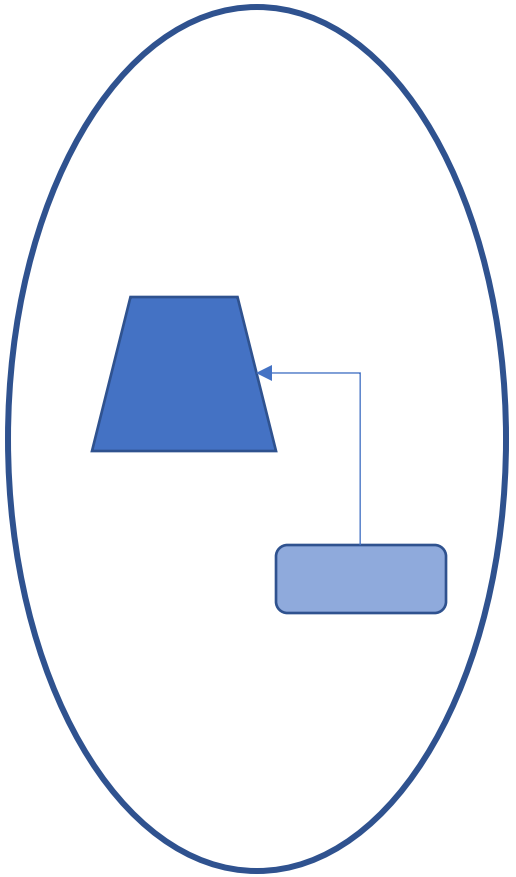
$$\hat{y} = \sigma'(\alpha \cdot \boldsymbol{h}_s^{(K)} + (1-\alpha) \cdot \boldsymbol{h}),$$

Table of Contents



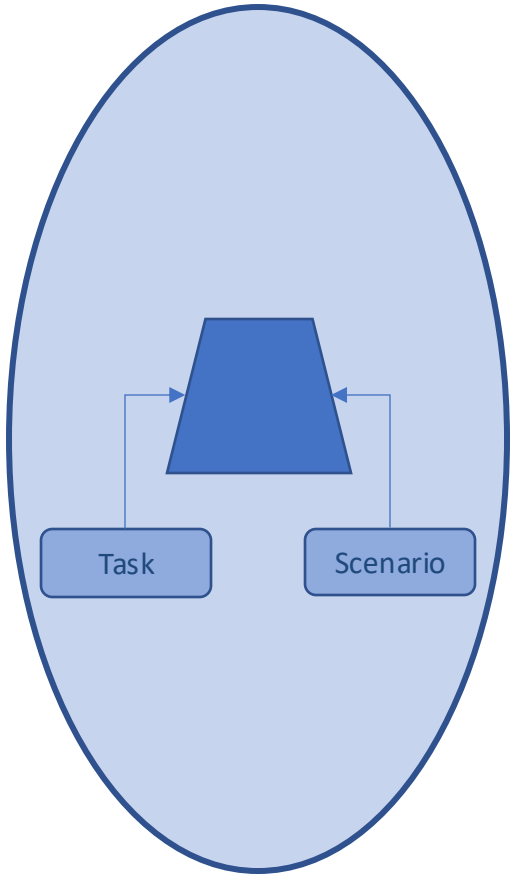
Shared-specific network paradigm

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



Dynamic weight

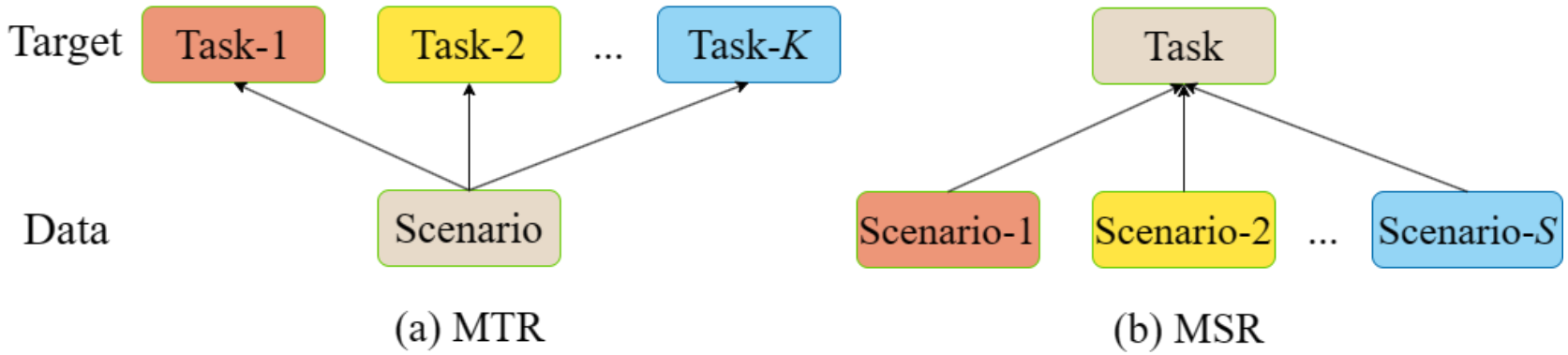
$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s)$



Multi-Scenario & Multi-Task

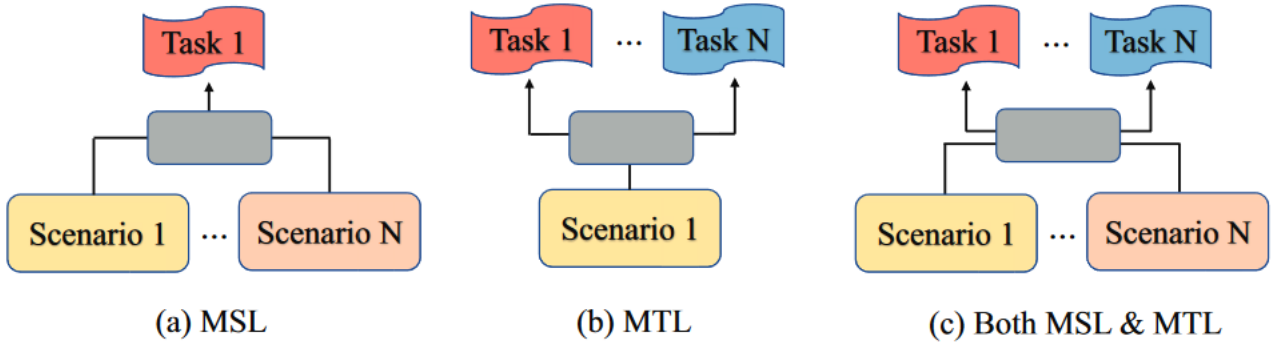
$wL(E^{Merge}, \Theta, \Theta^t, \Theta^s, \Theta^T)$

Multi-Scenario & Multi-Task Studies



AESM²

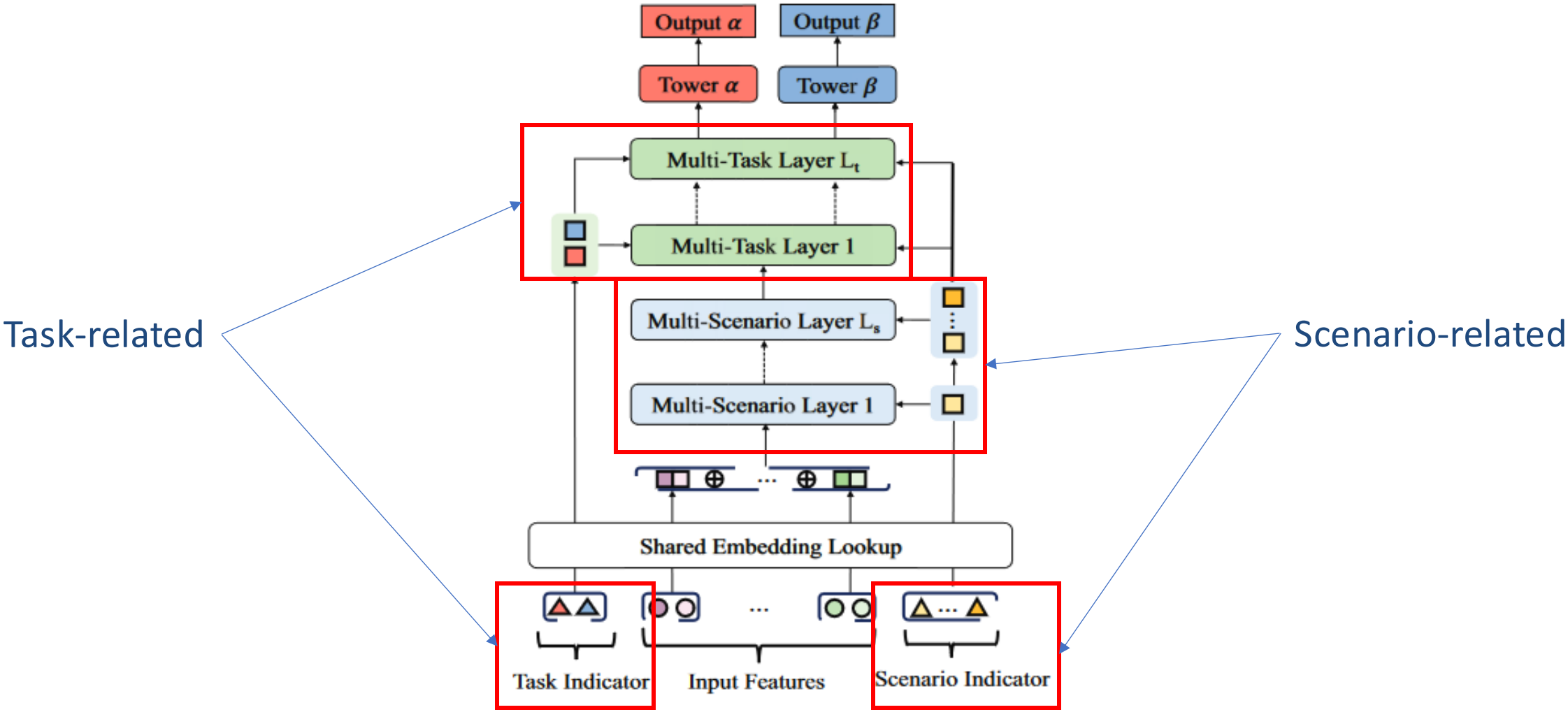
- Target
  - Develop a unified framework that could realize both MSL and MTL requirements
- Methods
  - Propose AESM², a flexible hierarchical structure where the multi-task layers are stacked over the multi-scenario layers
  - General expert selection algorithm



(a) MSL (b) MTL (c) Both MSL & MTL

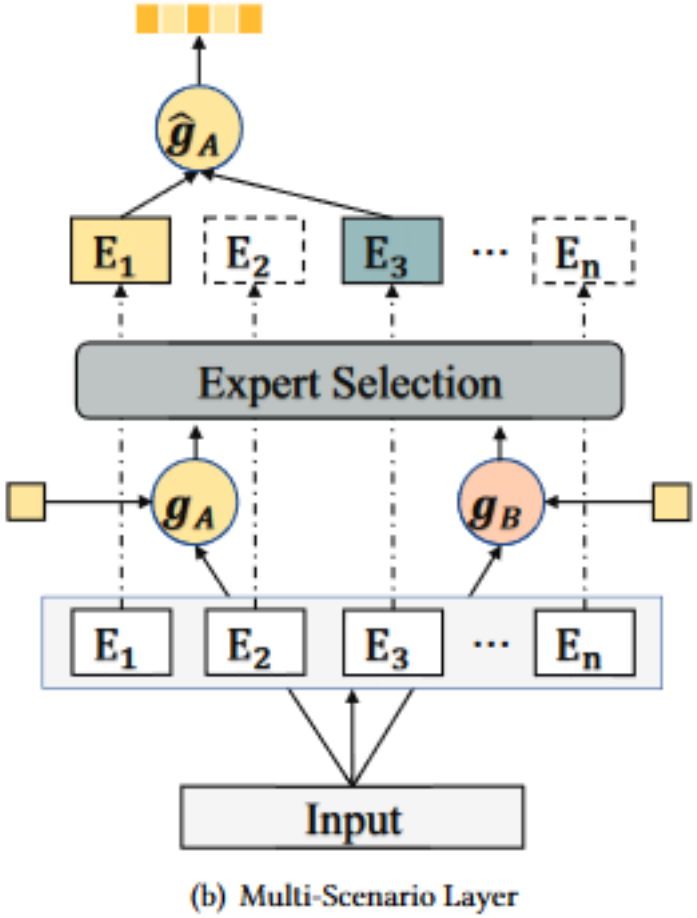
Automatic Expert Selection for Multi-Scenario and Multi-Task Search. SIGIR 2022.

AESM²


Output α
Output β
Tower α
Tower β
Multi-Task Layer L_t
Multi-Task Layer 1
Multi-Scenario Layer L_s
Multi-Scenario Layer 1
Shared Embedding Lookup
Task Indicator
Input Features
Scenario Indicator
Task-related
Scenario-related

AESM²

Multi-Scenario Layer



(b) Multi-Scenario Layer

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

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

- Expert selection

$$\mathcal{E}_{sp} = TopK(h_1^p, \cdots, h_n^p)$$
$$h_k^p = -KL(\mathbf{p}_j, \tilde{\mathbf{G}}[k,:])$$
$$\mathcal{E}_{sh} = TopK(h_1^q, \cdots, h_n^q)$$
$$h_k^q = -KL(\mathbf{q}_j, \tilde{\mathbf{G}}[k,:])$$
$$\mathbf{p}_j(e.g., [1, \cdots, 0])$$
$$\mathbf{q}_j = [1/m, \cdots, 1/m]$$

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

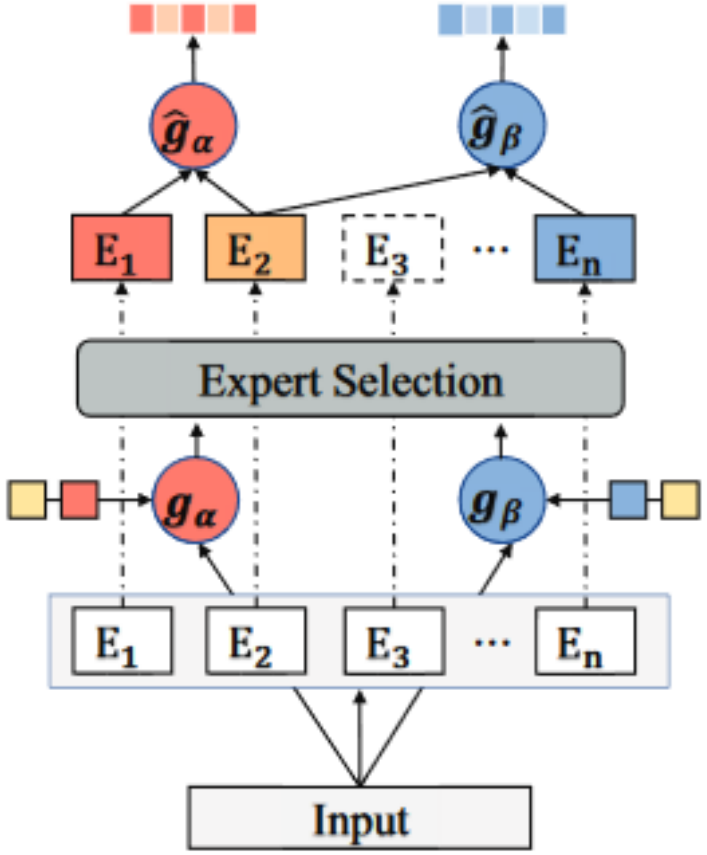
Specific / Shared

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

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

AESM²

Multi-Task Layer



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

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

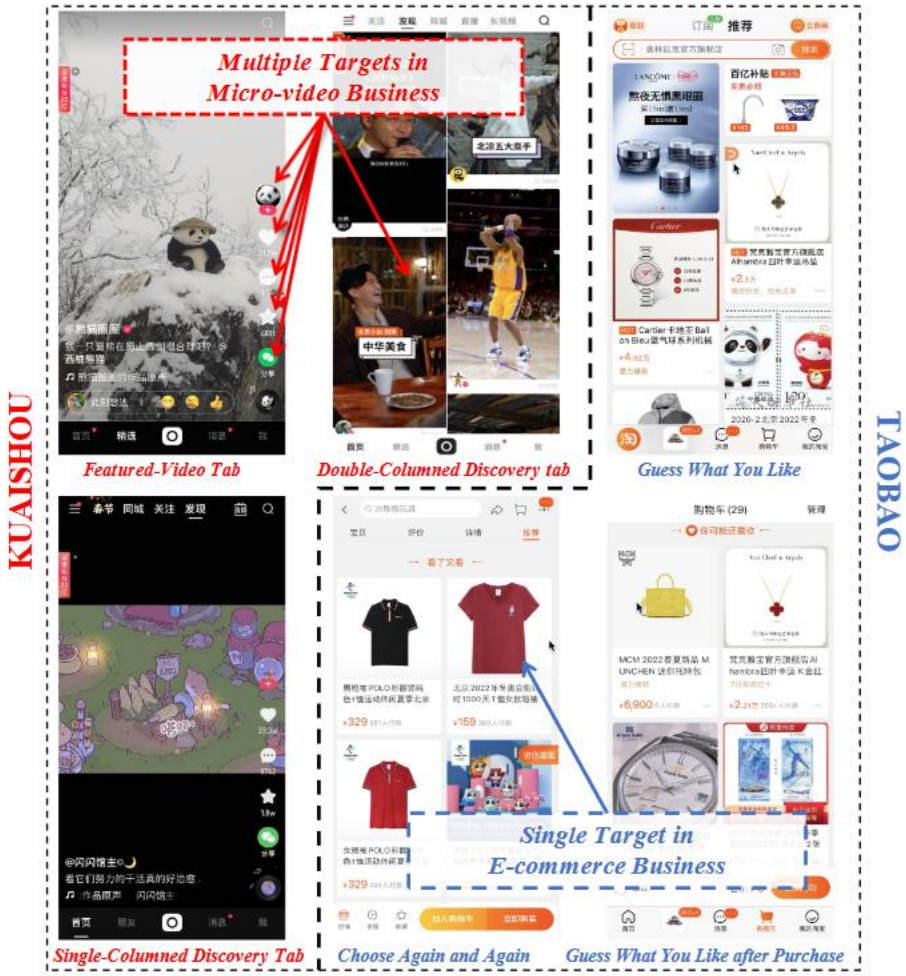
$$\mathbf{z}_k = TaskLayer(\mathbf{z}_j, \mathbf{t}_k) = MMoE(\mathbf{z_j}, \hat{\mathbf{g}}_k)$$

- Output layer

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

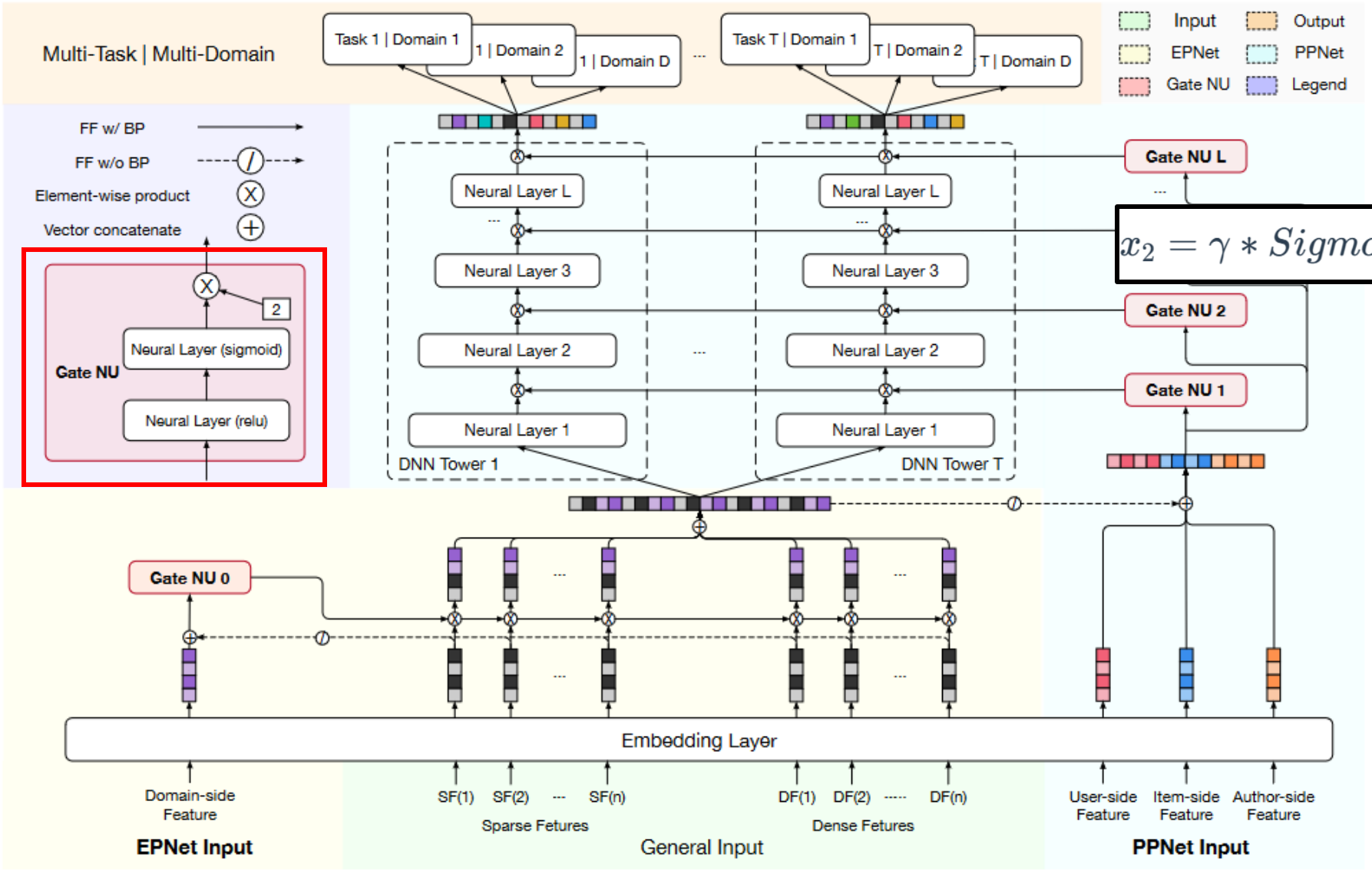
PEPNet

- Motivation
  - The imperfectly double seesaw phenomenon
  - More accurate personalization estimates can alleviate the imperfectly double seesaw problem
- Target
  - Jointly model multi-domain and multi-task
  - an efficient, low-cost deployment and plug-and-play method that can be injected in any network.



PEPNet: Parameter and Embedding Personalized Network for Infusing with Personalized Prior Information. KDD 2023.

PEPNet Details

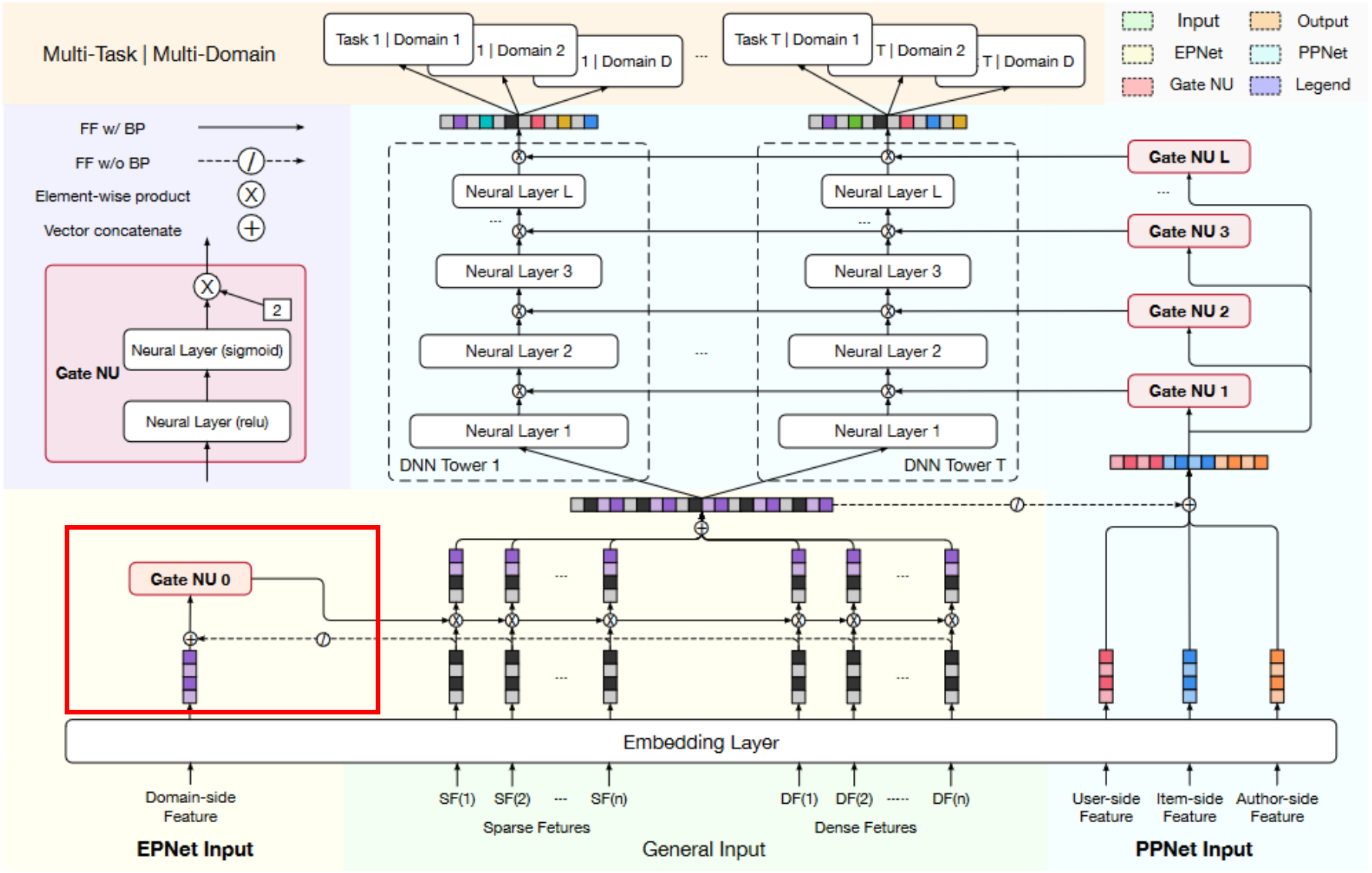


Gate Neural Unit (Gate NU)

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

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

PEPNet Details



EPNet

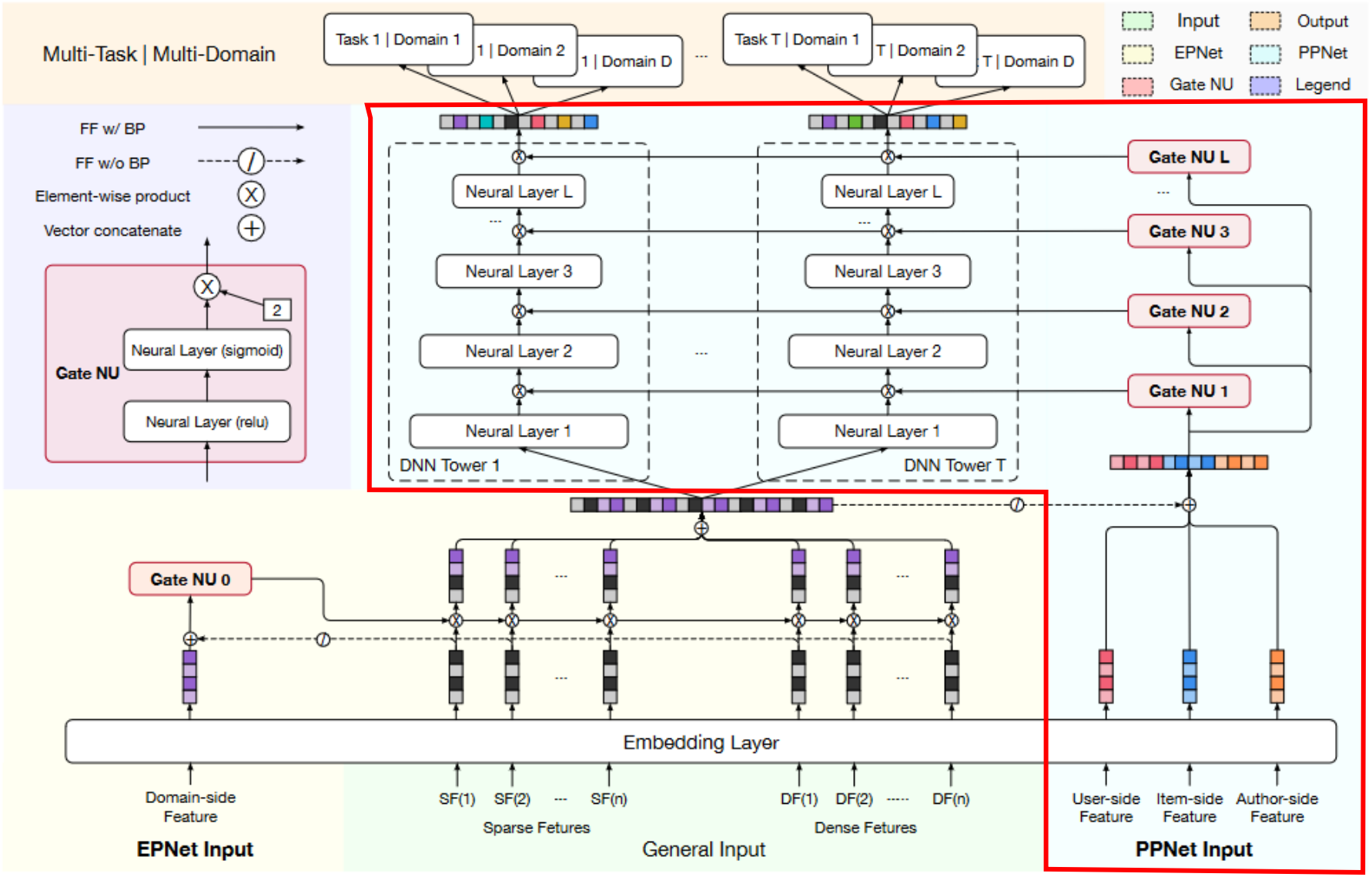
$$\mathbf{E} = E(\mathcal{F}_S) \oplus E(\mathcal{F}_D)$$

Embeddings of sparse features and dense features

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

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

PEPNet Details



PPNet

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

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

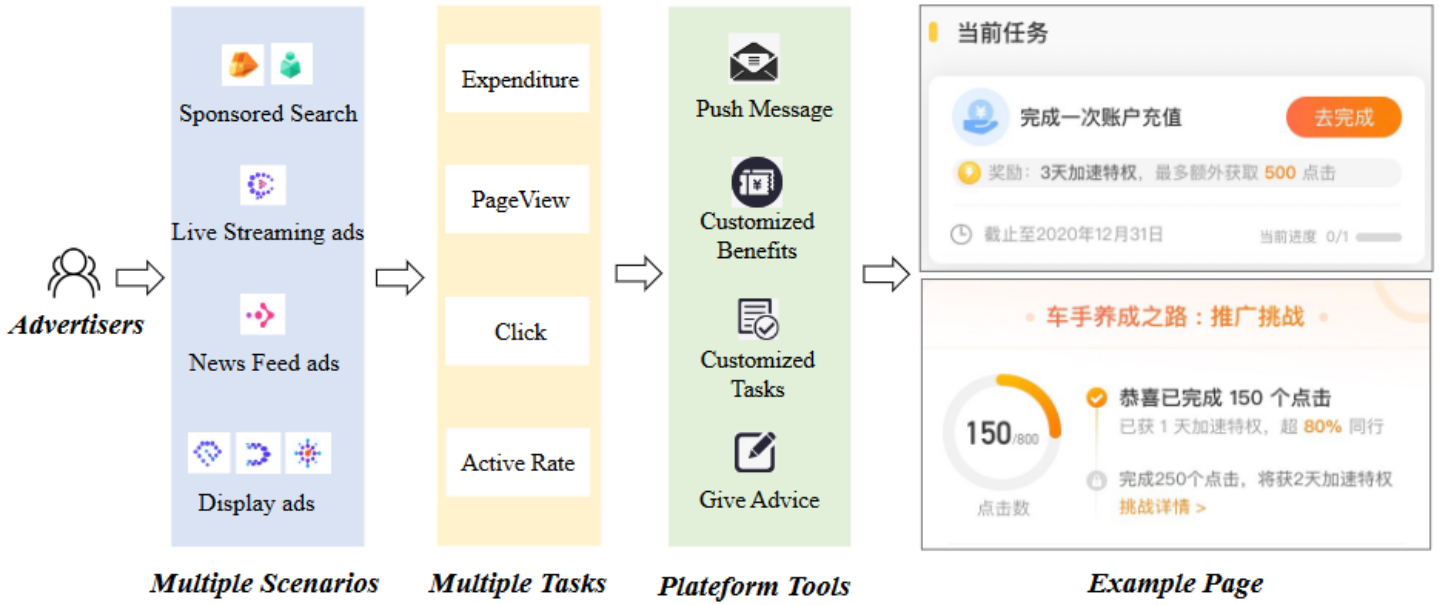
$$\mathbf{O}_{pp}^{(l)} = \boldsymbol{\delta}_{task}^{(l)} \otimes \mathbf{H}^{(l)},$$

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

M2M

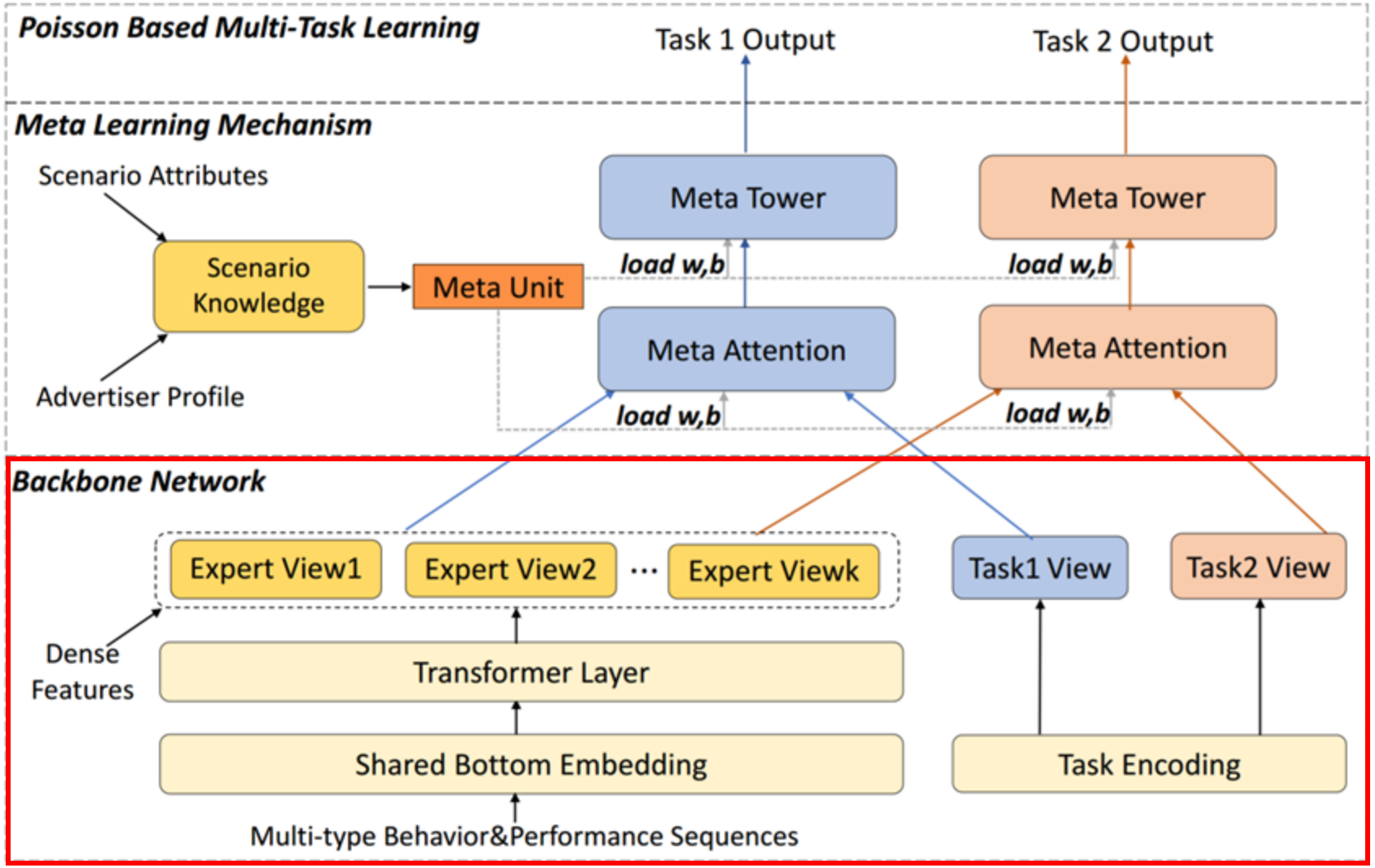
- Motivation
  - Less attention has been drawn to advertisers
  - Major e-commerce platforms provide multiple marketing scenarios.

- Methods
  - Meta unit
  - Meta attention module
  - Meta tower module

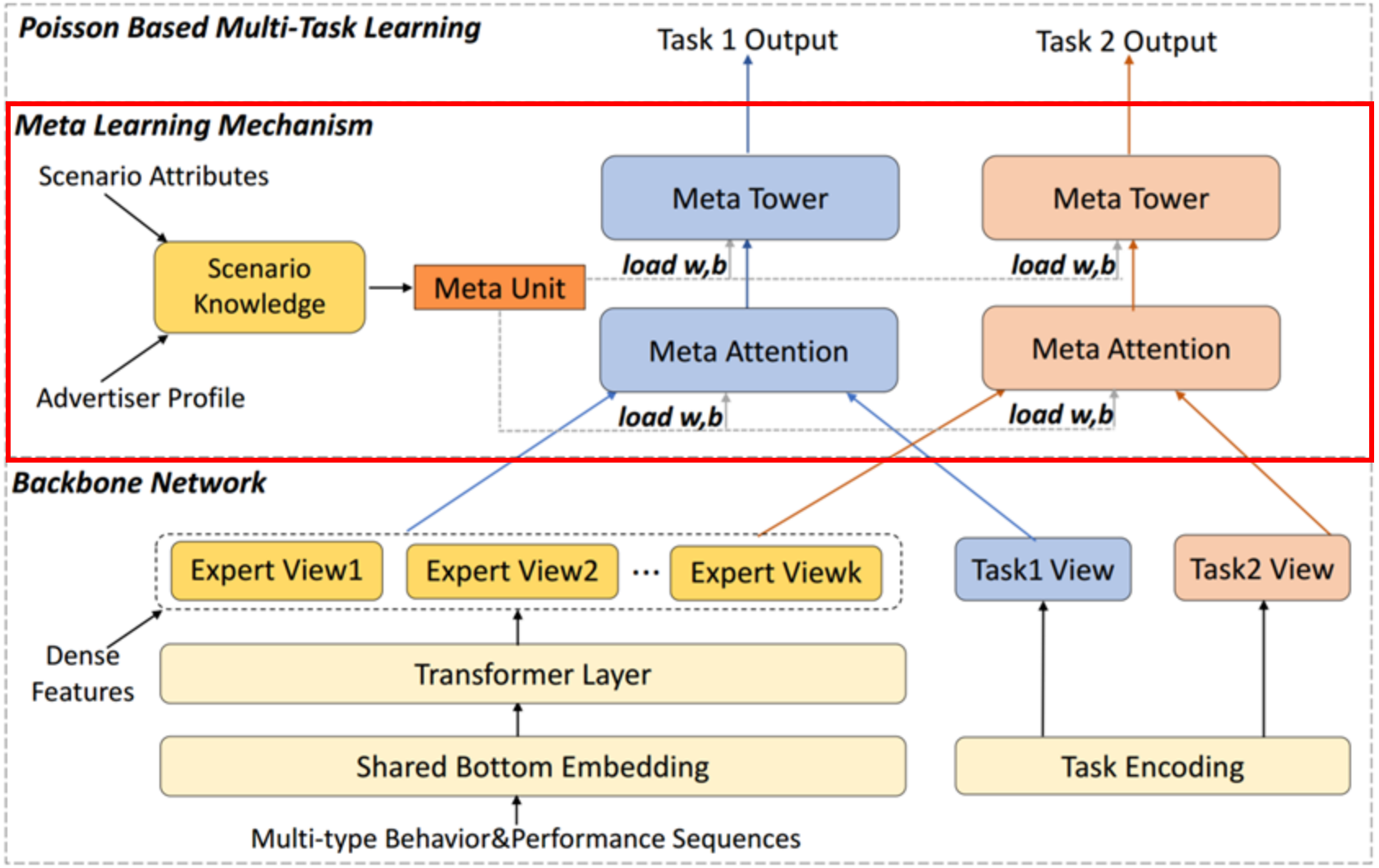


Leaving No One Behind: A Multi-Scenario Multi-Task Meta Learning Approach for Advertiser Modeling. WSDM 2022.

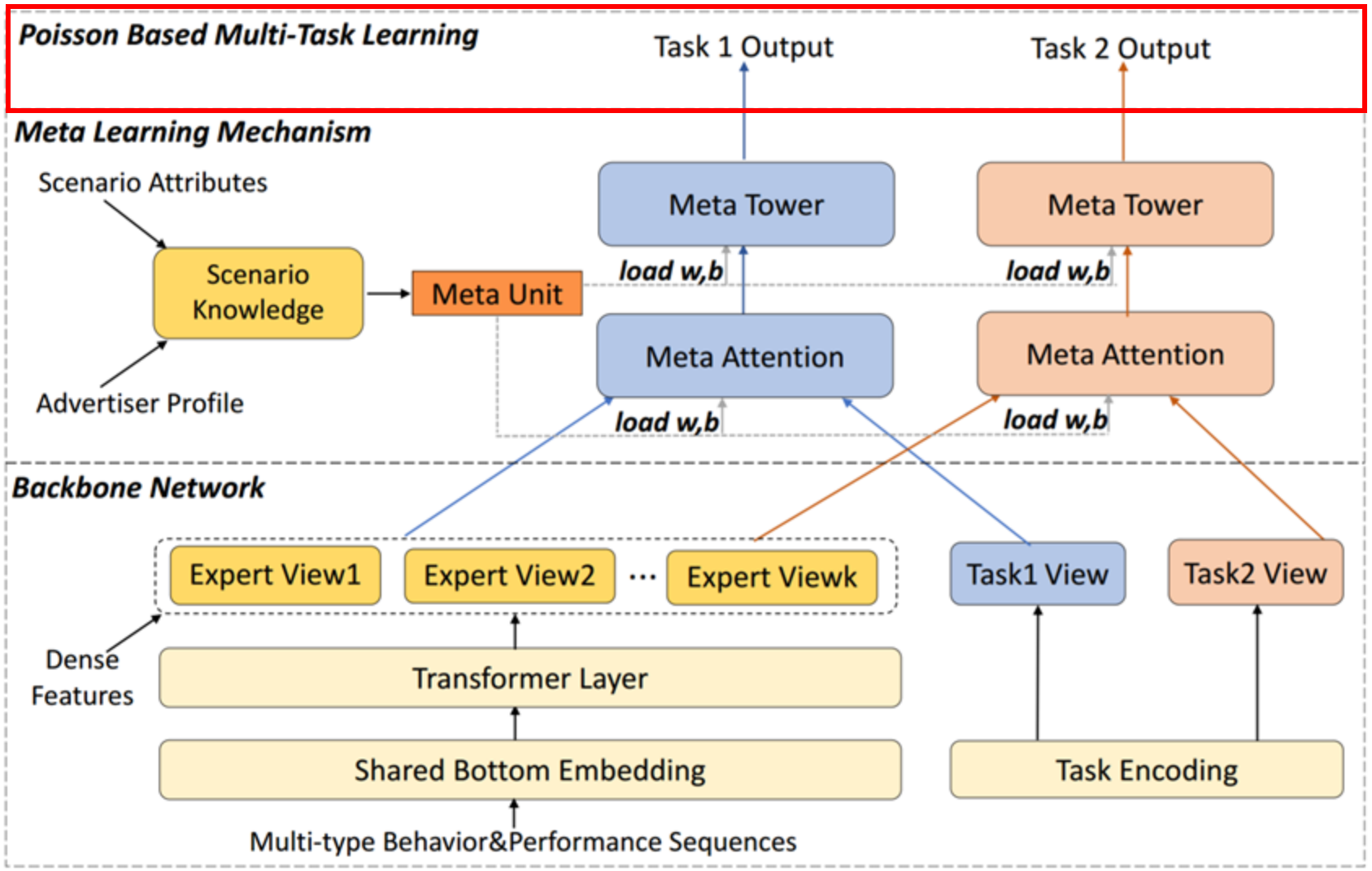
M2M Overview



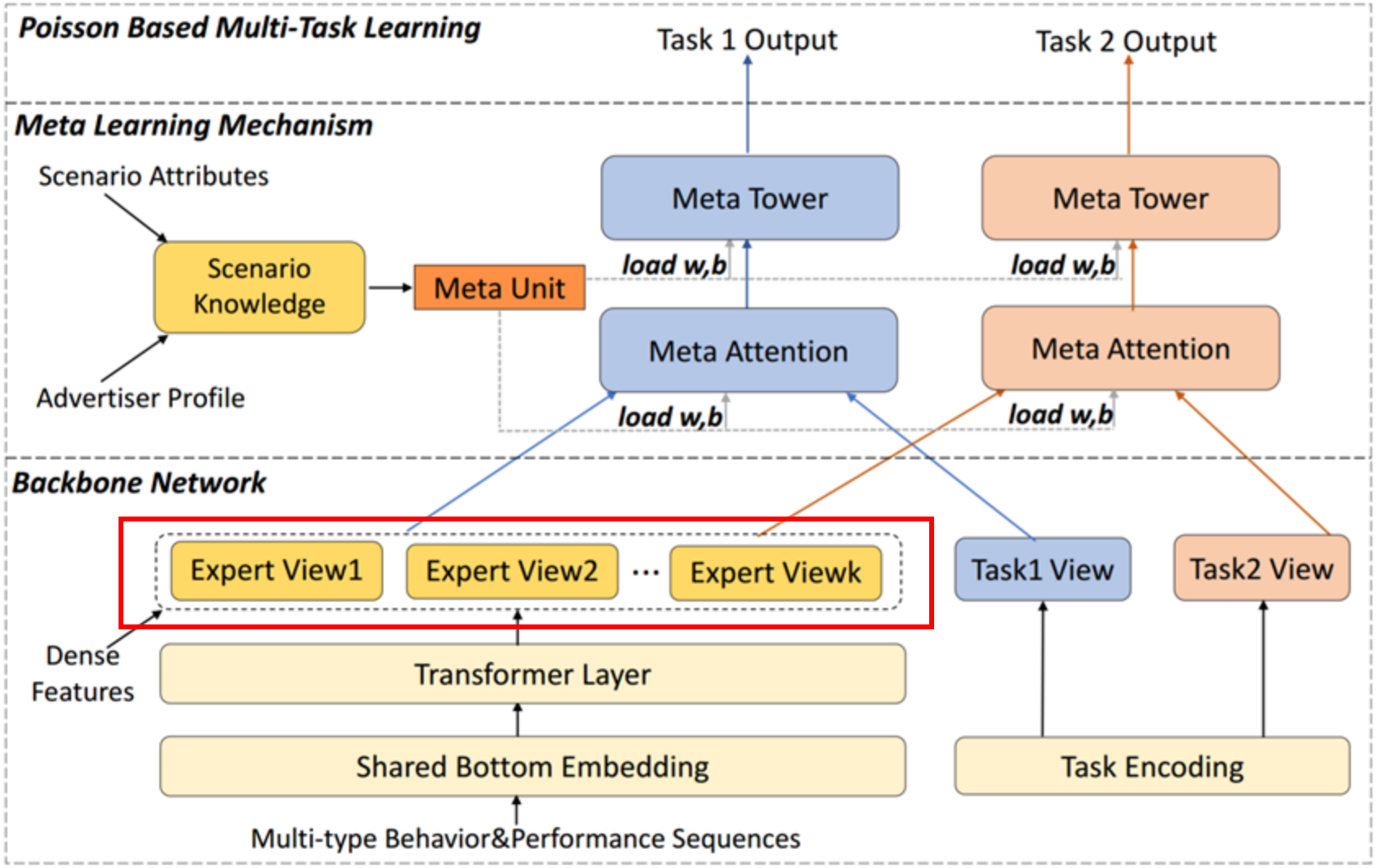
M2M Overview



M2M Overview



M2M Details



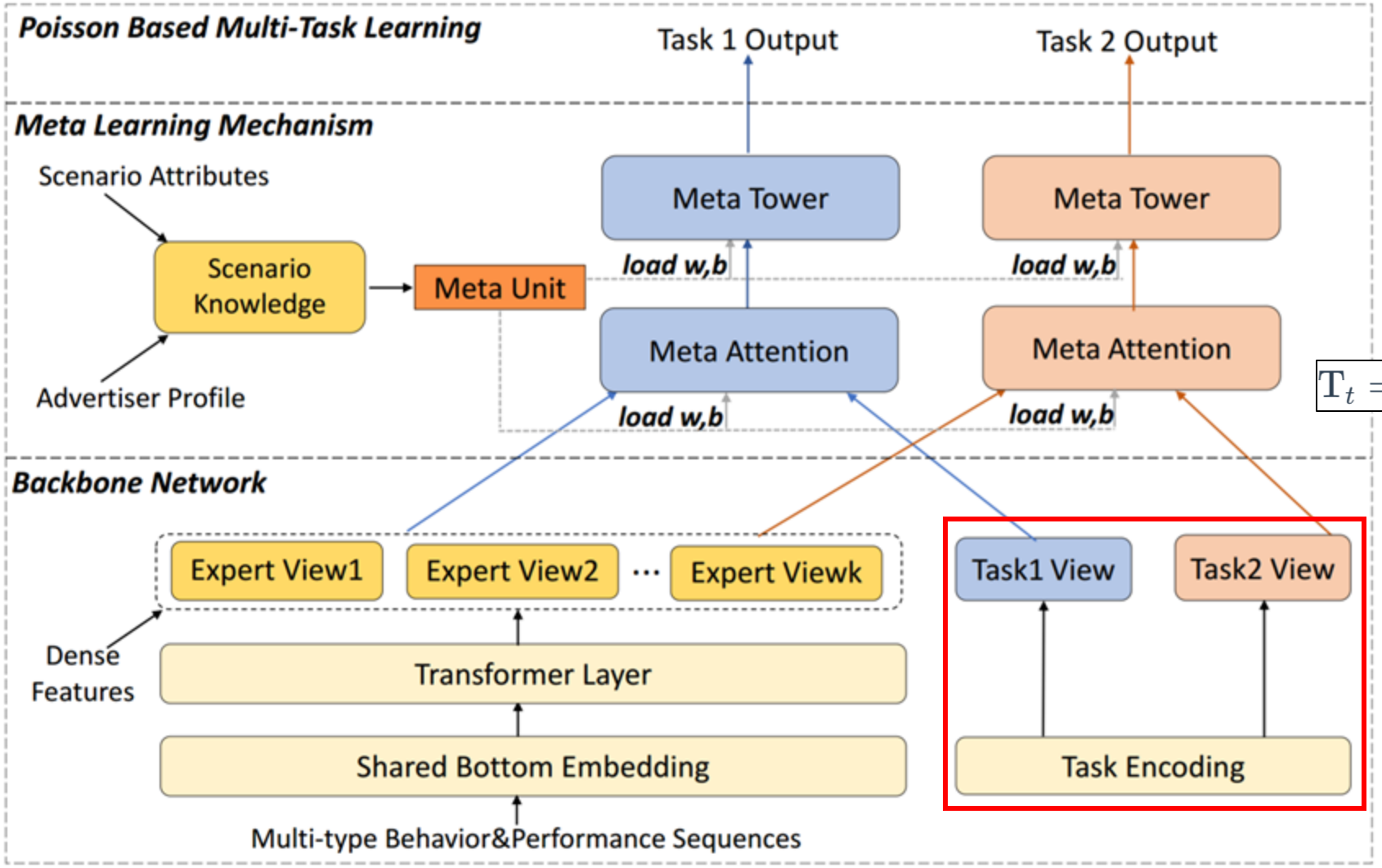
Backbone Network

Expert View Representation

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

F is the output of transformer layer

M2M Details



Backbone Network

Expert View Representation

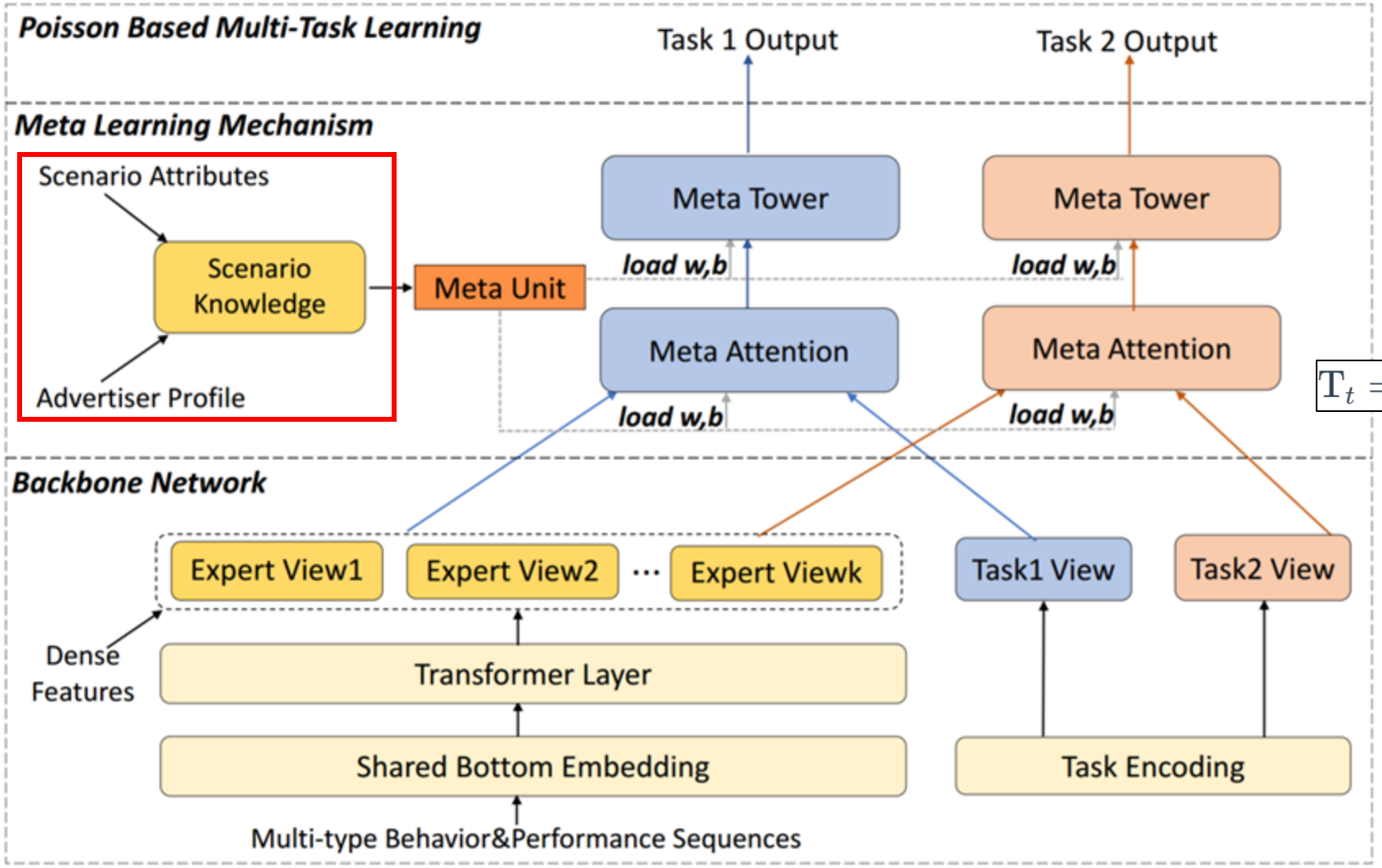
$$\mathrm{E_i} = f_{MLP}(\mathbf{F}), \forall i \in 1, 2, .., k$$

F is the output of transformer layer

Task View Representation

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

M2M Details



Backbone Network

Expert View Representation

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

F is the output of transformer layer

Task View Representation

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

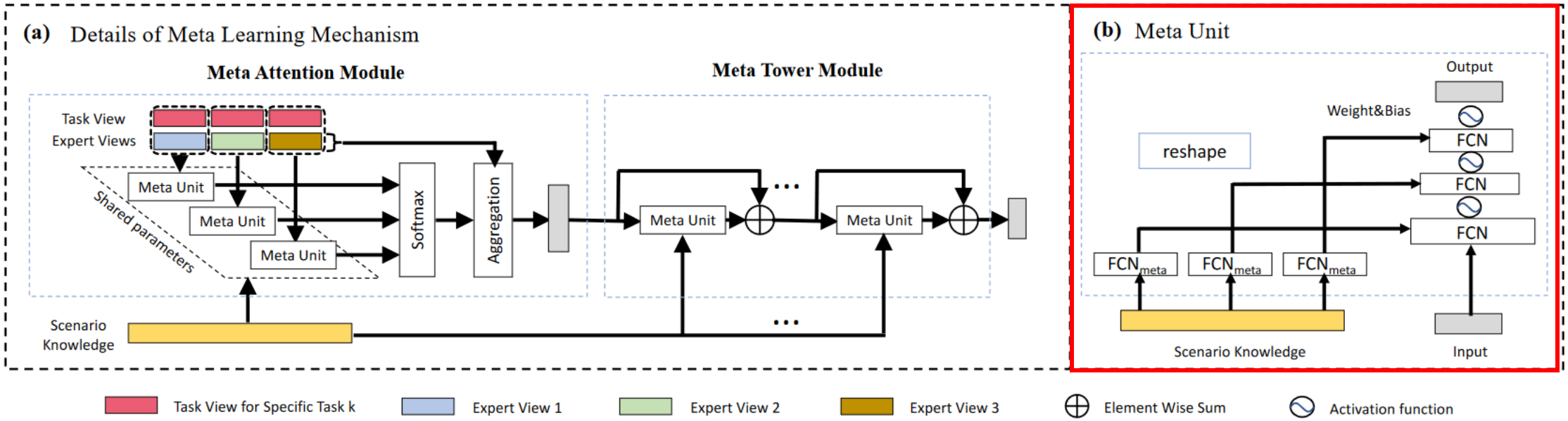
Scenario Knowledge Representation

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

M2M Details

Meta Unit

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

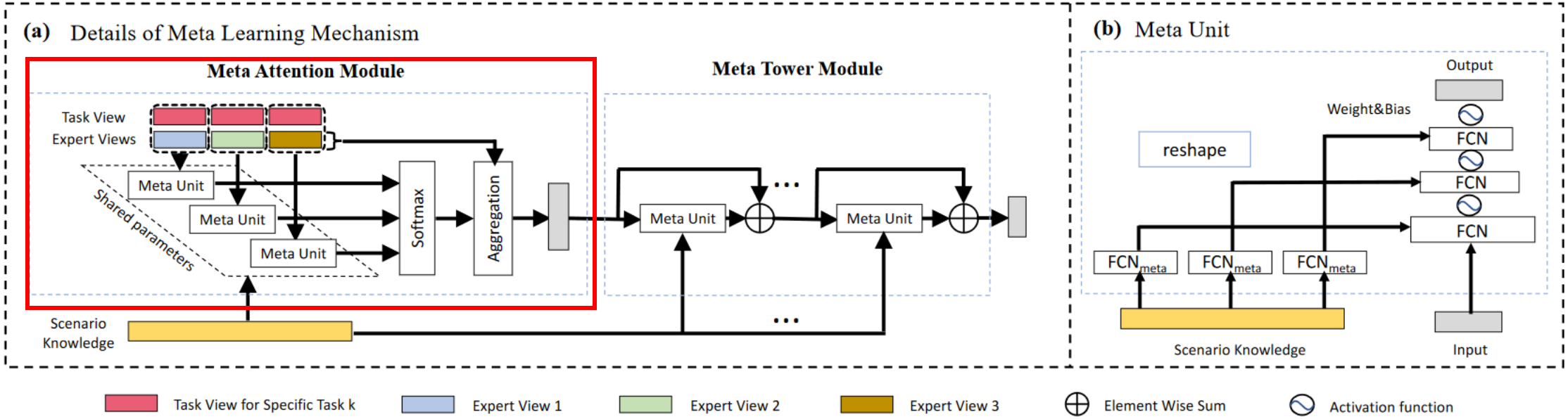


M2M Details

Meta Attention Module

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

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



M2M Details

Meta Attention Module

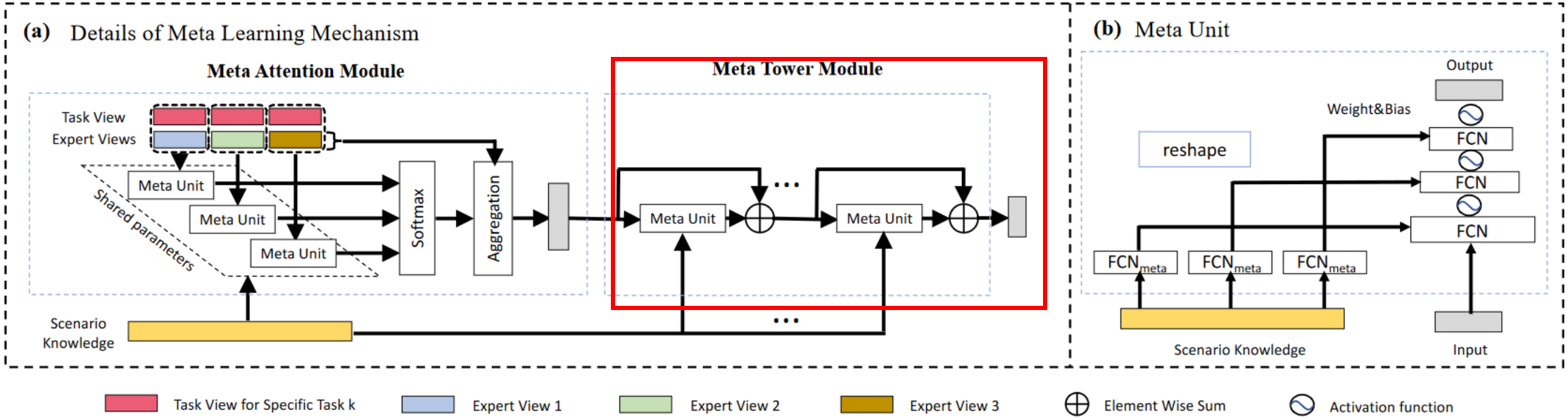
$$a_{t_i} = \mathrm{v}^T Meta_t([\mathrm{E_i} \parallel \mathrm{T}_t])$$

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

Meta Tower Module

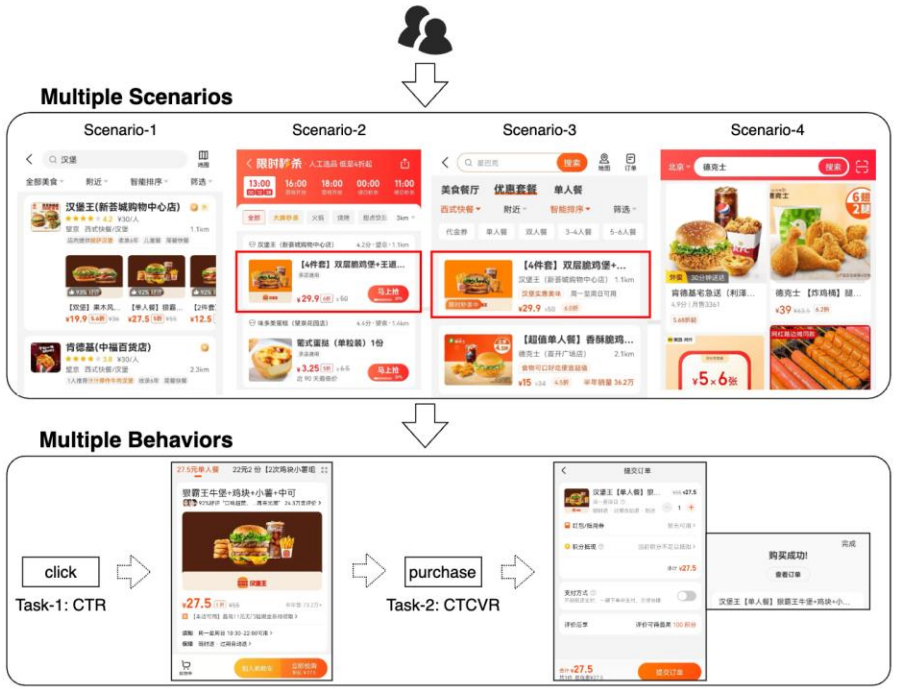
$$\mathrm{L}_t^{(0)} = \mathrm{R}_t$$

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



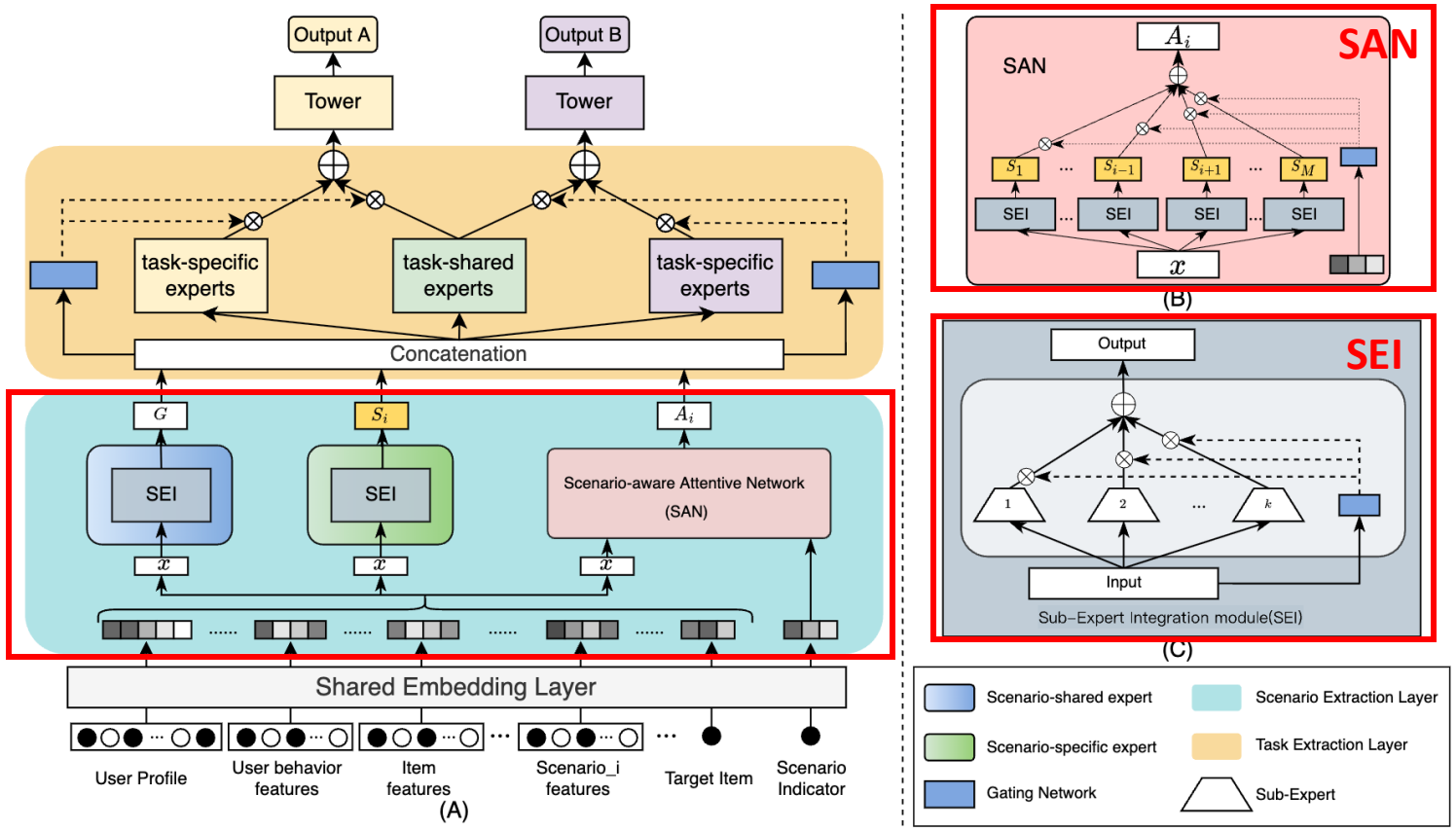
HiNet

- 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



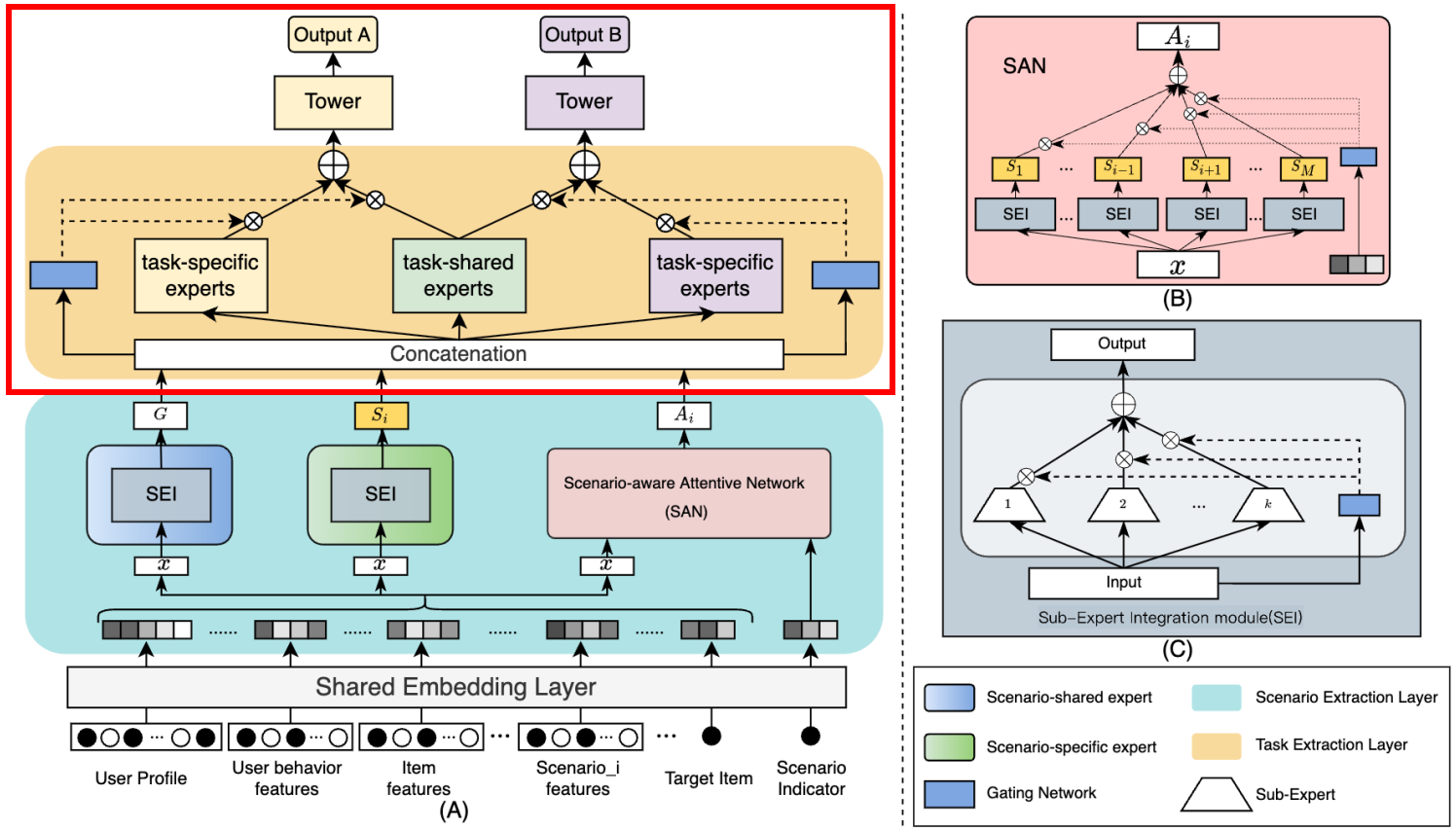
HiNet: Novel Multi-Scenario & Multi-Task Learning with Hierarchical Information Extraction. ICDE 2023

HiNet



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

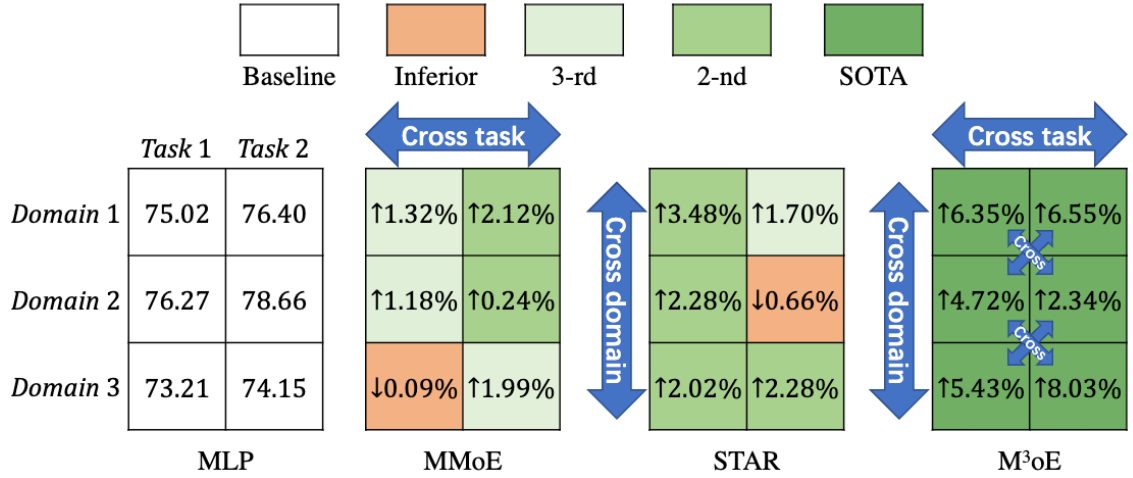
HiNet



- 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

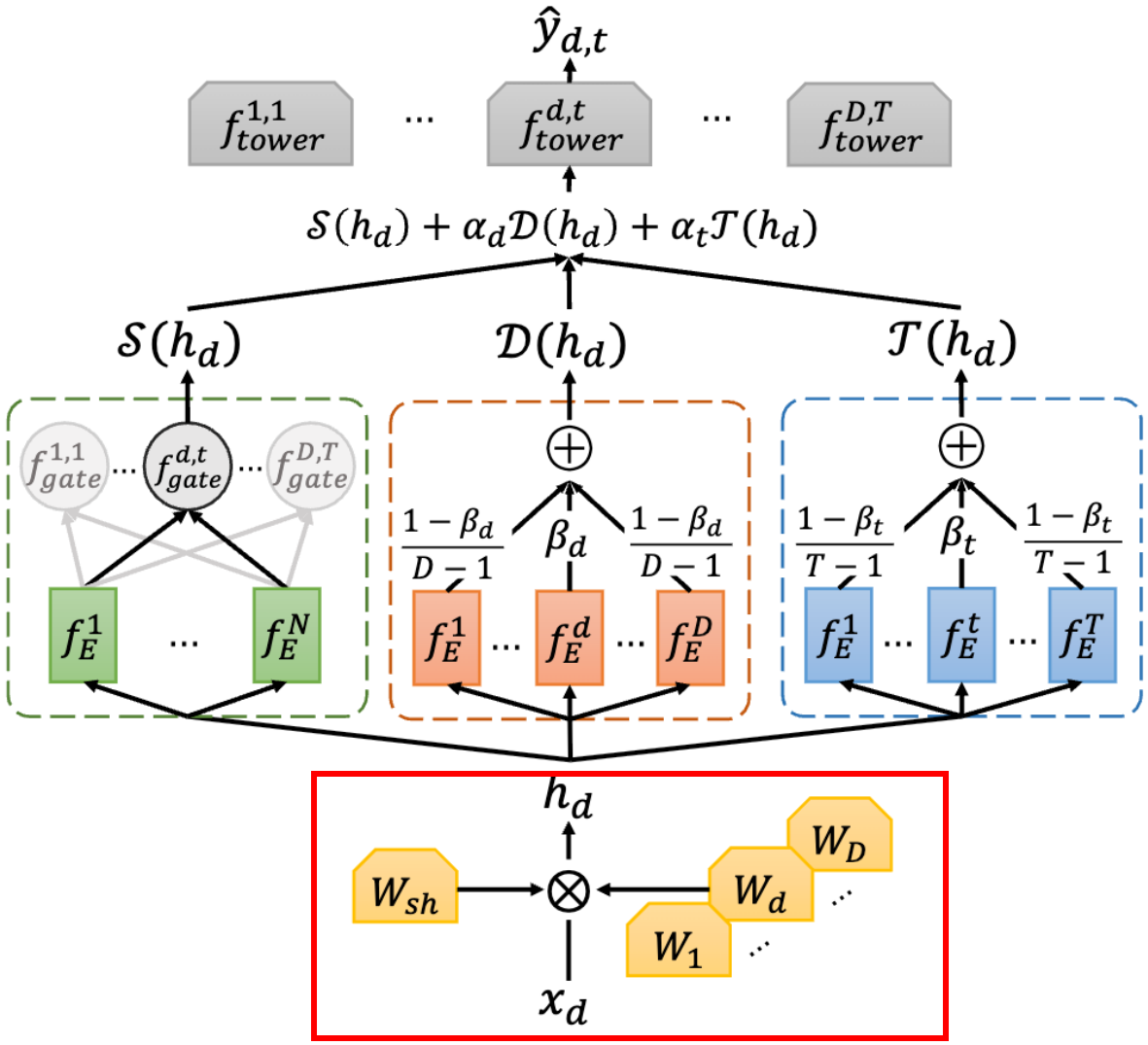
M3oE

- 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



M3oE: Multi-Domain Multi-Task Mixture-of Experts Recommendation Framework. SIGIR 2024

M3oE



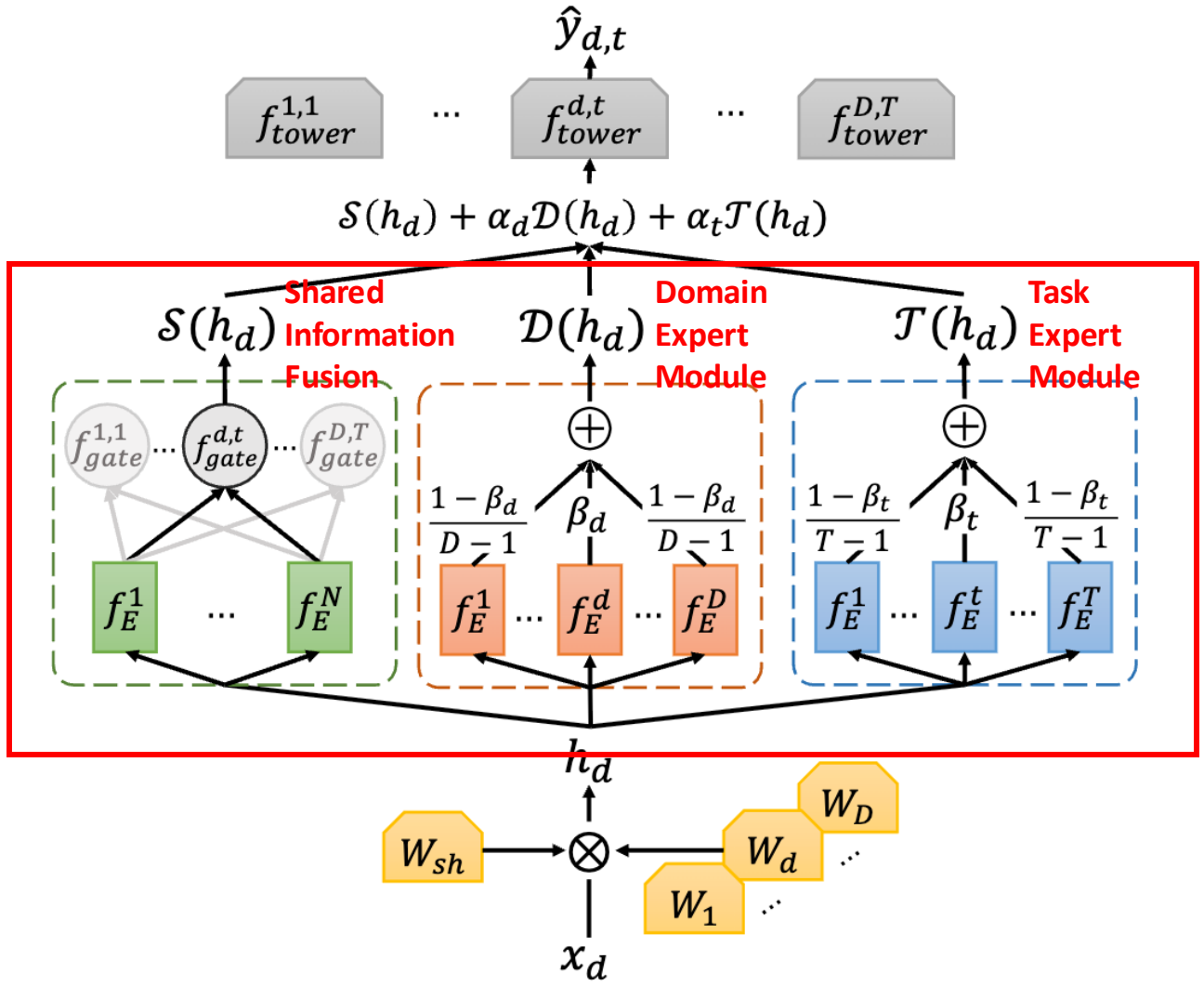
- Domain Representation Extraction Layer

$$\widehat{\boldsymbol{W}}_d = \boldsymbol{W}_d \otimes \boldsymbol{W}_{\text{sh}}$$

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

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

M3oE



- Domain Representation Extraction Layer

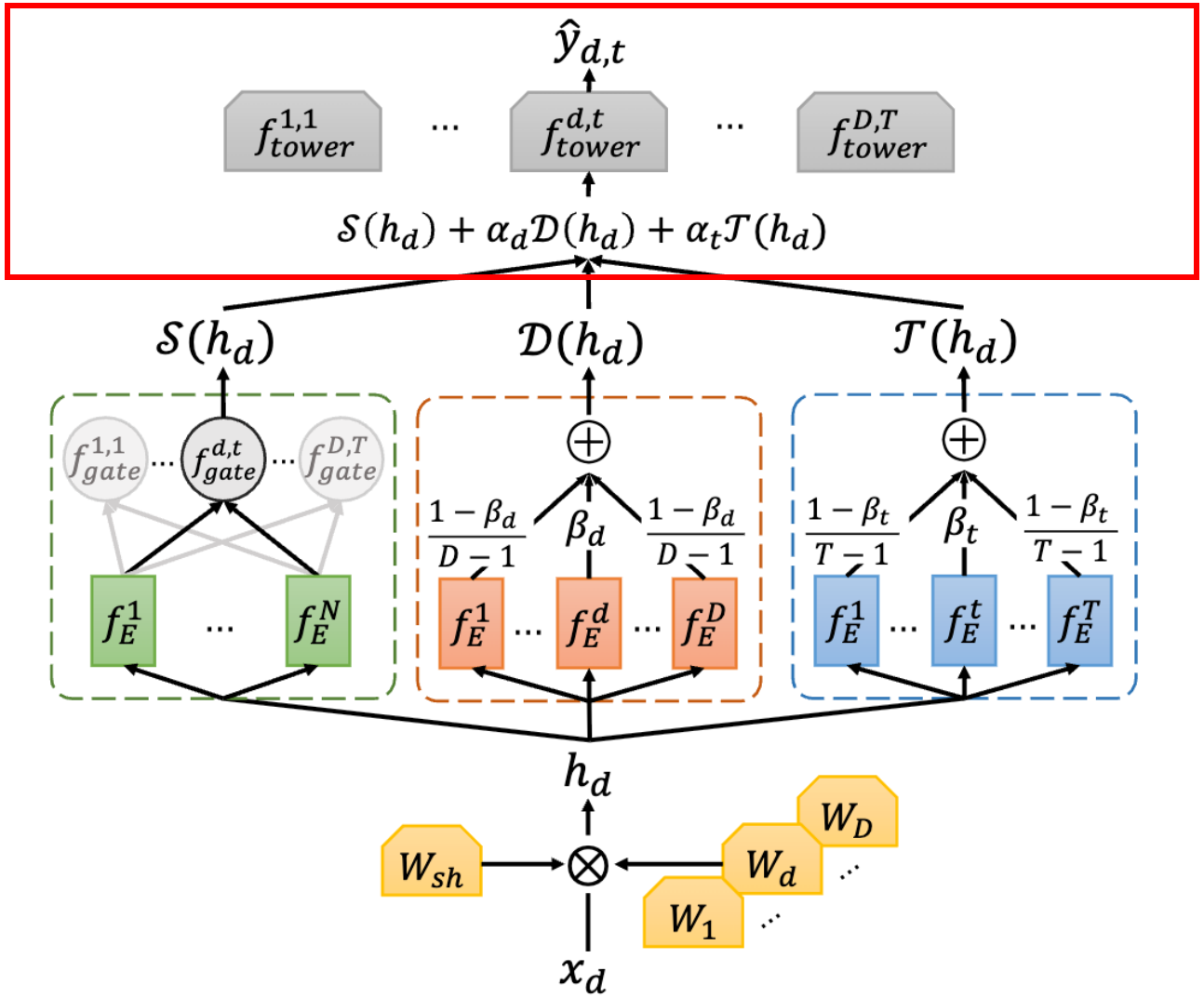
$$\widehat{\boldsymbol{W}}_d = \boldsymbol{W}_d \otimes \boldsymbol{W}_{\text{sh}}$$

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

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

- Multi-View Expert Learning Layer
  - Shared Information Fusion
  - Domain Expert Module
  - Task Expert Module

M3oE



- Domain Representation Extraction Layer

$$\widehat{\boldsymbol{W}}_d = \boldsymbol{W}_d \otimes \boldsymbol{W}_{\text{sh}}$$

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

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

- Multi-View Expert Learning Layer
  - Shared Information Fusion
  - Domain Expert Module
  - Task Expert Module
- Multi-View Representation Balancing

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

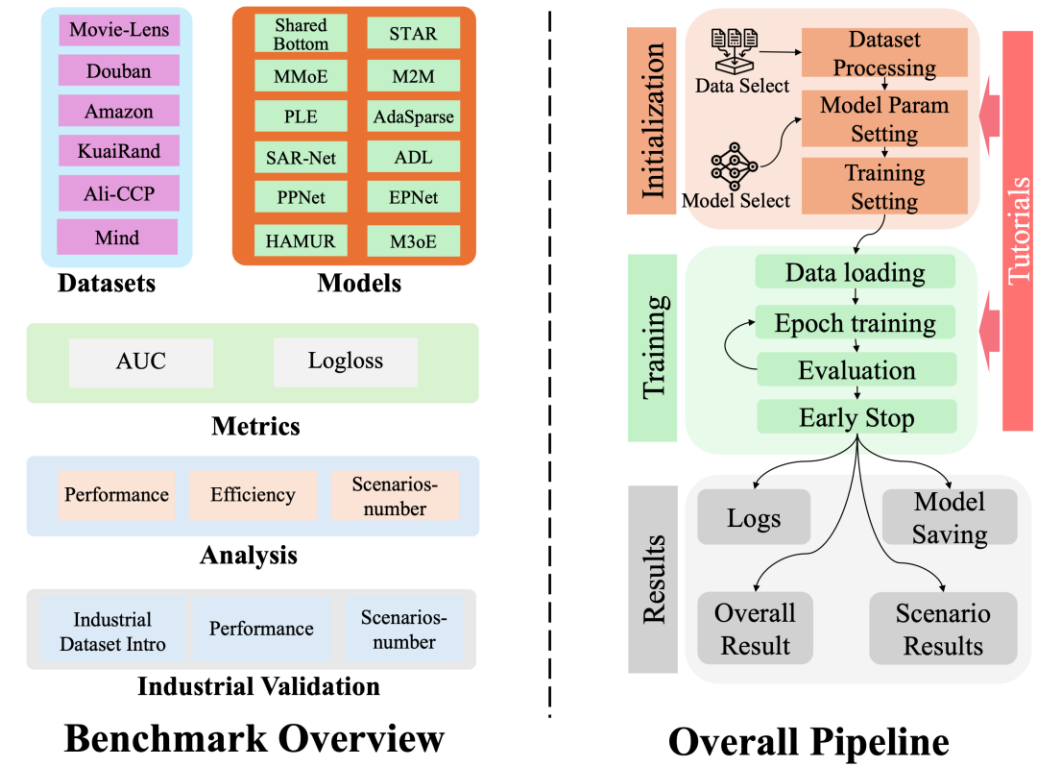
$$f_{tower}^{d,t}(\bar{\boldsymbol{h}}_d) = \boldsymbol{W}_{d,t}^2 \text{ReLU}(\boldsymbol{W}_{d,t}^1 \bar{\boldsymbol{h}}_d + \boldsymbol{b}_{d,t}^1) + \boldsymbol{b}_{d,t}^2$$

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

Benchmark: Scenario-Wise Rec

Scenario - Wise Rec

Benchmark for Multi-Scenario Recommendation



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
  - Compresentative analysis and step-by-step tutorial

Summary

➢ Multi-Scenario Recommendation

Model	Setting	Methods
STAR	Multi-Scenario	Shared-Specific
SAR-Net	Multi-Scenario	Shared-Specific; Experts
ADI	Multi-Scenario	Shared-Specific
Uni-CTR	Multi-Scenario	Shared-Specific; LLMs
M-LoRA	Multi-Scenario	Shared-Specific; LoRAs
HAMUR	Multi-Scenario	Dynamic Weight
HierRec	Multi-Scenario	Dynamic Weight

Model	Setting	Methods
LLM4MSR	Multi-Scenario	Dynamic Weight; LLMs
AESM2	Multi-Scenario & Multi-Task	Experts
PEPNet	Multi-Scenario & Multi-Task	Dynamic Weight
M2M	Multi-Scenario & Multi-Task	Dynamic Weight; Experts
HiNet	Multi-Scenario & Multi-Task	Experts
M3oE	Multi-Scenario & Multi-Task	Experts
Scenario-Wise Rec	Multi-Scenario	Benchmark

Future Directions

➢ Multi-Scenario Recommendation

Topic	Challenge & future direction
LLM-based multi-scenario & multi-task modeling	• Explore quantification or compression techniques for handling large-scale scenarios. • More fine-grained modeling to bridge semantic gaps between LLM and MSR models.
Robustness	• Scenarios with different available information (multimodal …)
Privacy	• Data need to be shared between different scenarios to build a unified model. Methods to protect user privacy should be proposed.
Fairness and Bias	• The issue of fairness in recommendation scenarios.

Coffee Break

Huawei Noah's Ark Lab

WWW25 Huawei Noah's Ark Lab Chat Group

AML Lab
CityU

Agenda

Introduction

Yejing Wang

Joint Modeling in RS

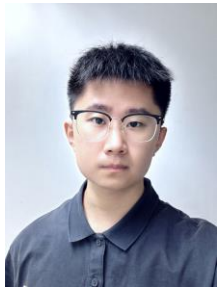
Pengyue Jia

Multi-task Recommendation

Yuhao Wang

Multi-scenario Recommendation

Pengyue Jia

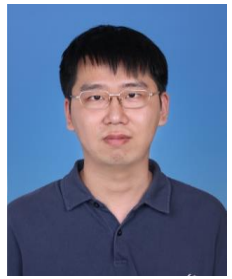


Xiaopeng Li

Multi-behavior Recommendation

Jingtong Gao

Multi-modal Recommendation



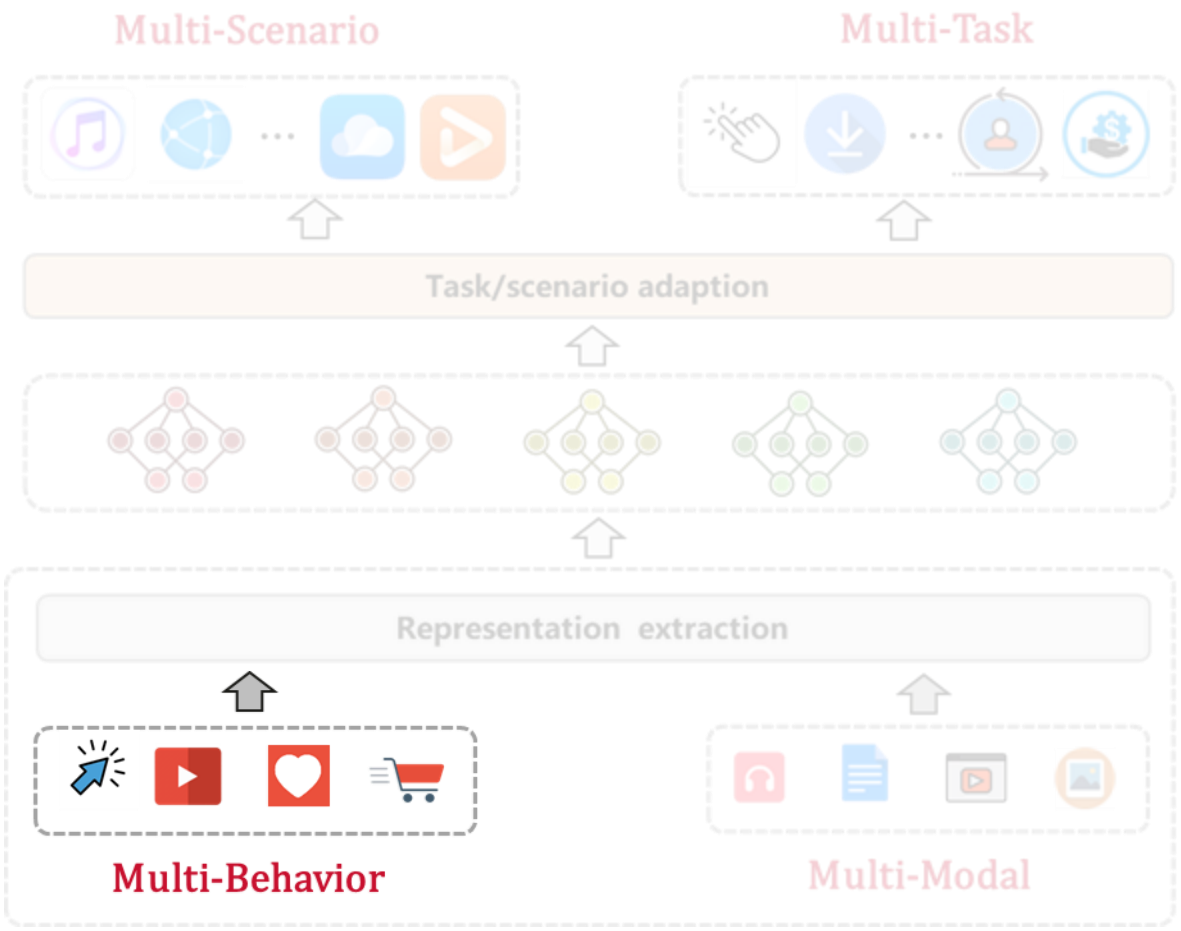
Qidong Liu

Conclusion

Future Work

Yichao Wang

Multi-Behavior Modeling



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

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

$E^{B} = G(H_1, H_2, \ldots, H_N)$ — Multi-behavior

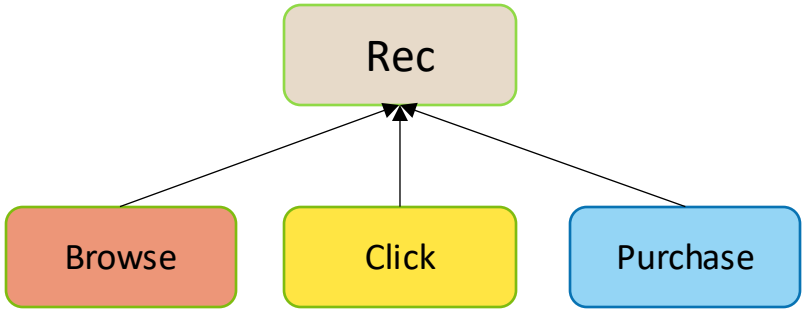
$E^{M} = M(E^{txt}, E^{v}, \ldots, E^{p})$ — Multi-modal

$wL(E^{Merge}, \theta^{sh}, \theta^{t}, \theta^{s})$ — Multi-scenario

$wL(E^{Merge}, \theta^{sh}, \theta^{t}, \theta^{s})$ — Multi-task

Multi-Behavior Modeling

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



Multi-behavior recommendation

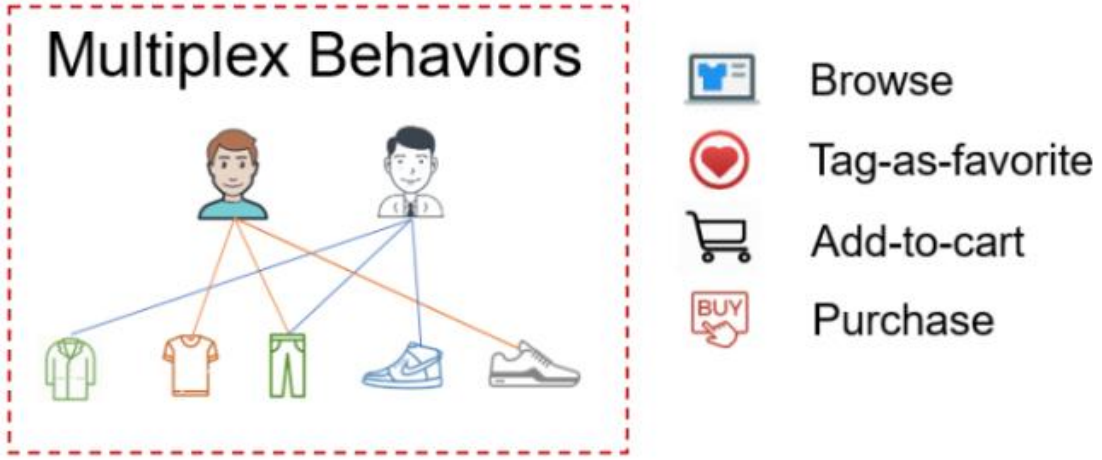
Behavior Type Definition

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

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

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

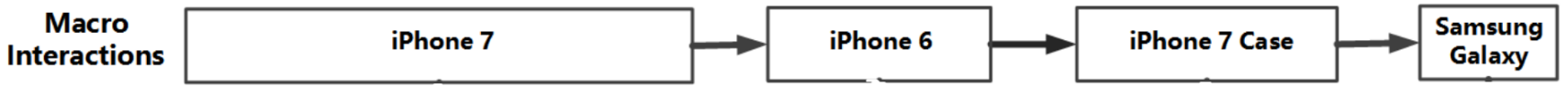
    E.g. Same behavior in two domains => different behaviors (highlight the distinctions)



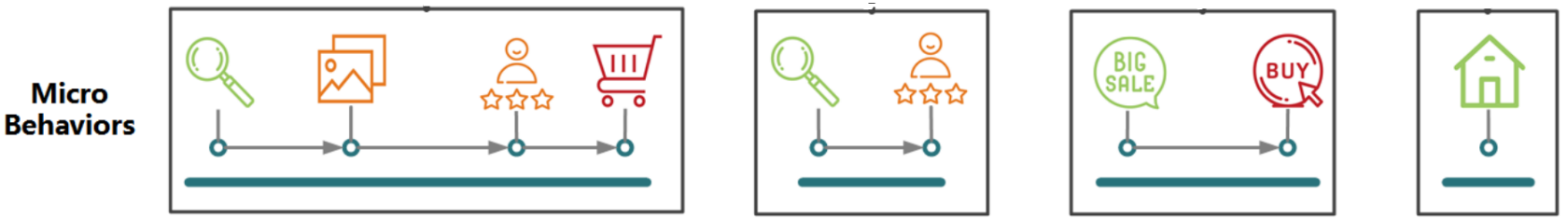
A Survey on User Behavior Modeling in Recommender Systems. arxiv preprint 2023.
Graph Meta Network for Multi-Behavior Recommendation. SIGIR 2021.

Behavior Type Definition

➢Macro behaviors:

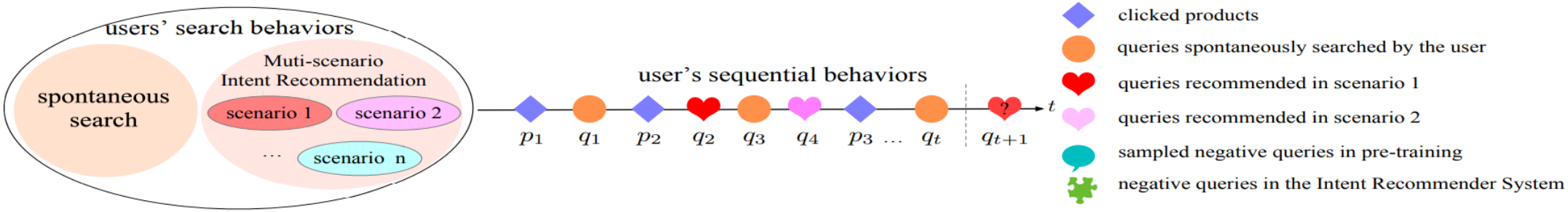


➢Micro behaviors:



➢Behaviors from different domains or scenarios

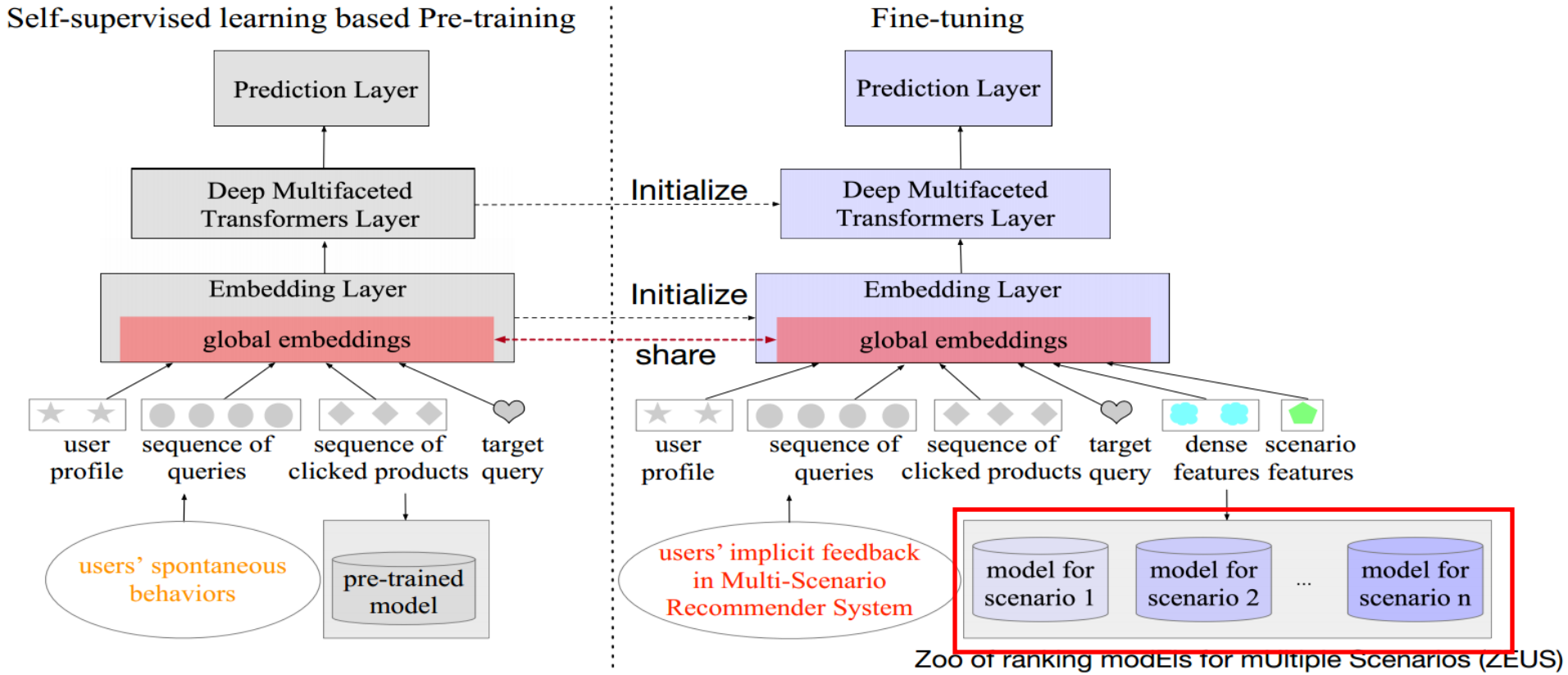
E.g. Same behavior in two domains => different behaviors (highlight the distinctions)



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

Multi-Behavior Fusion

➢Modeling the complicated cross-scenario behavior dependencies

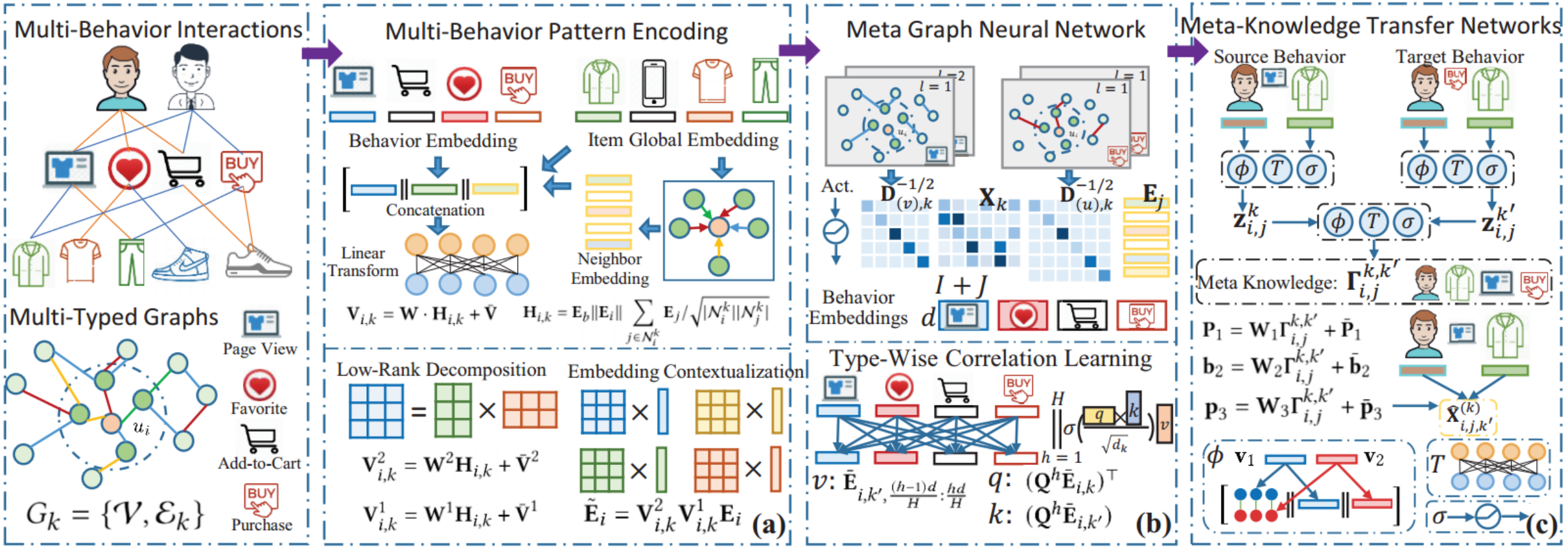


Example: pre-training and fine-tuning of ZEUS

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

Multi-Behavior Fusion

➢Modeling the complicated cross-type behavior dependencies



Example: MB-GMN

Graph Meta Network for Multi-Behavior Recommendation. SIGIR 2021.

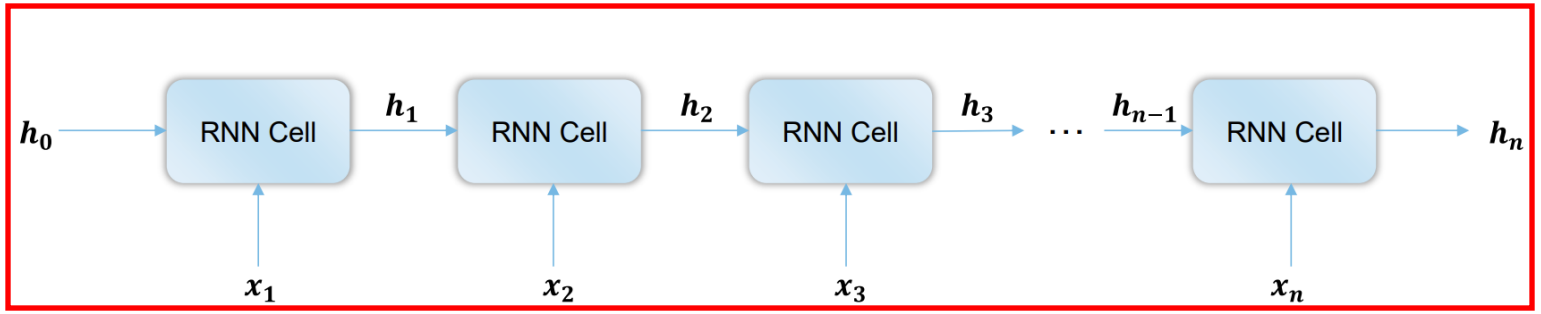
Challenges

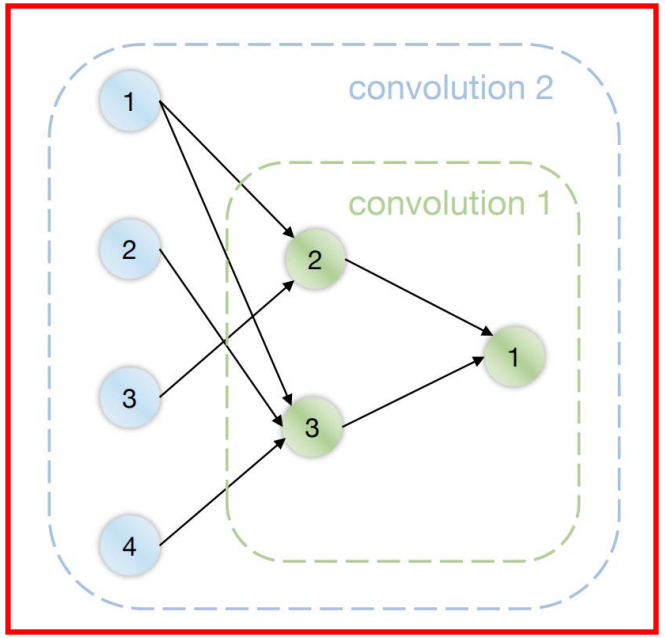
- Sequence modeling of heterogeneous behavioral feedback
  - How to model different behaviors and their feedbacks
- Modeling behavior relations
  - How to capture complicated behavior relations
- Joint long-term and short-term preference modeling with heterogeneous behaviors of users
  - How to combining 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

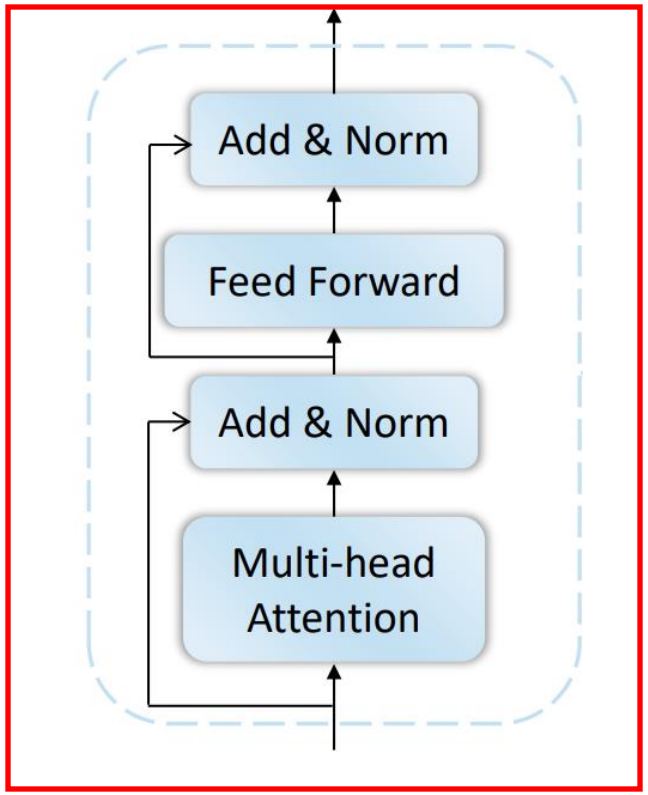
A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

Taxonomy

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







A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

Taxonomy

- 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.

A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

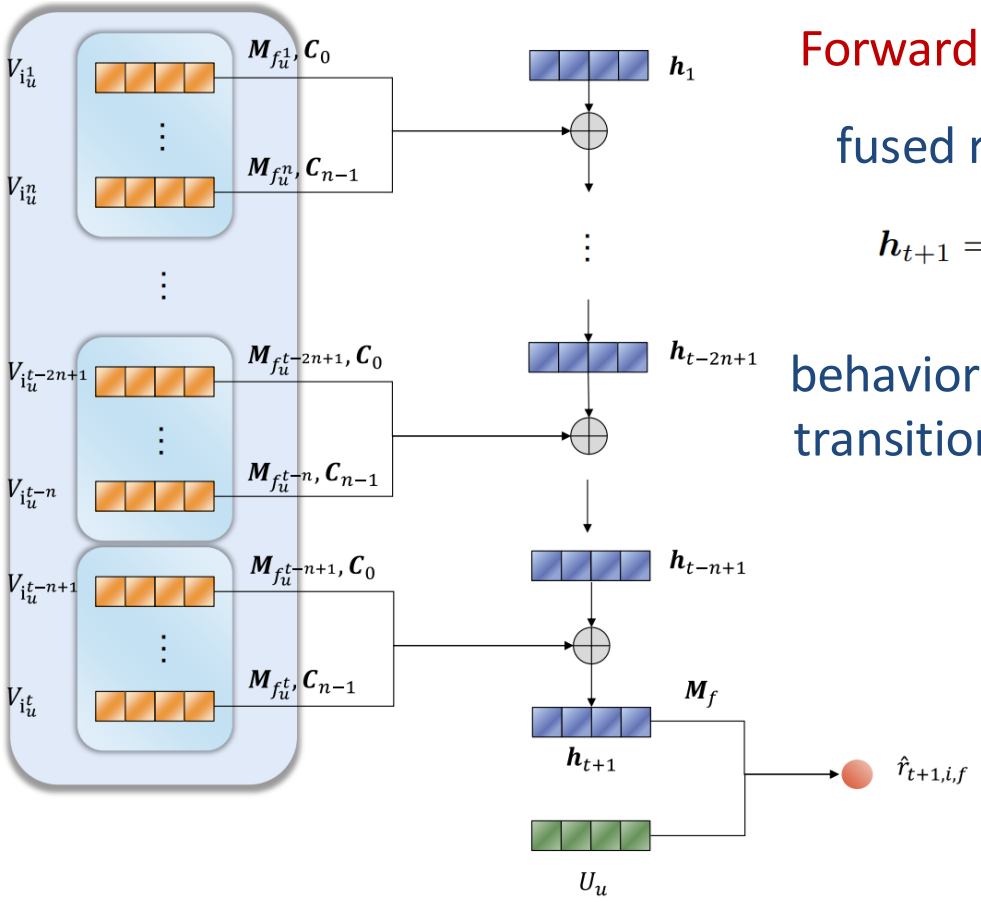
RNN-based Methods—RLBL

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

windowed representations

Item-behavior pairs

$$\{(i_u^{t-n+1}, f_u^{t-n+1}), \ ..., (i_u^t, f_u^t)\}$$



RLBL

Forwarding steps:

fused representation Item embedding

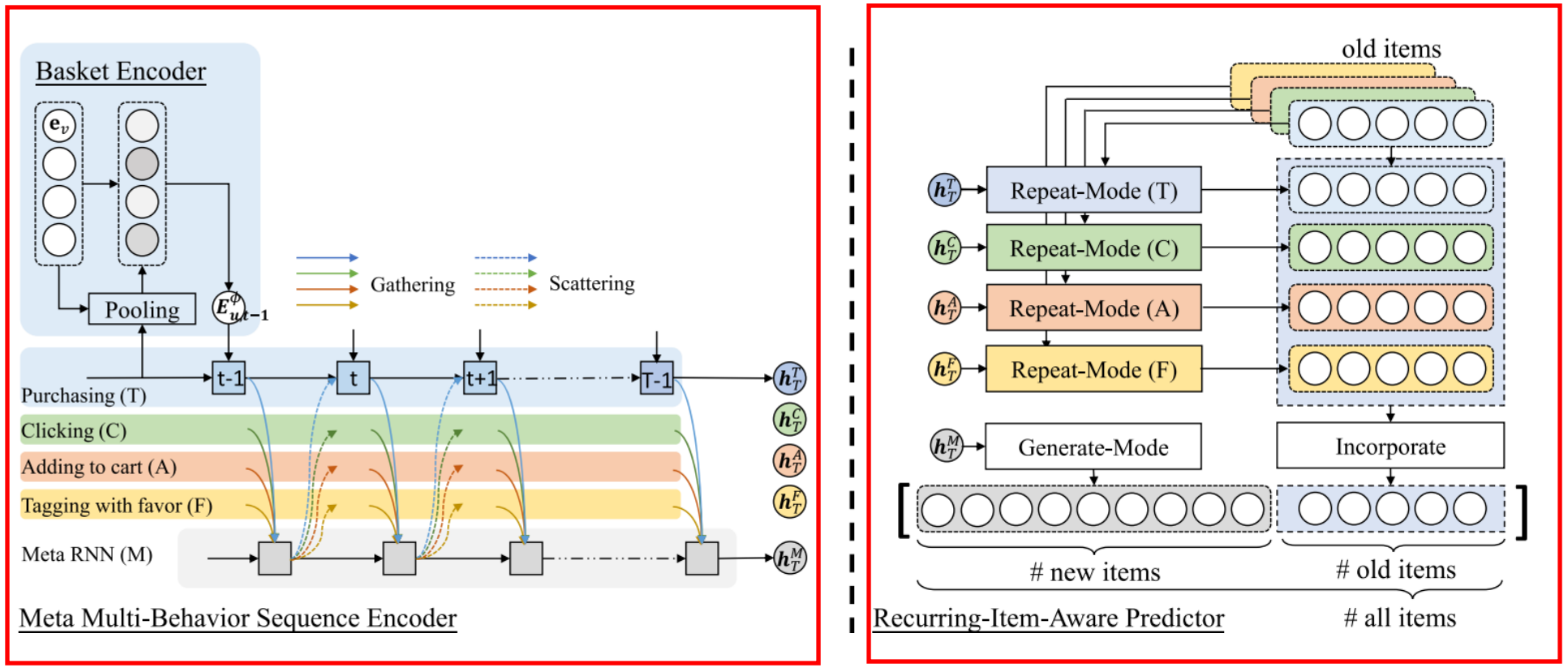
$$h_{t+1} = W_{\text{RLBL}} h_{t-n+1} + \sum_{i=0}^{n-1} C_i M_{f_u^{t-i}} V_{i_u^{t-i}}$$

behavior-specific transition matrix position transition embeddin

Multi-Behavioral Sequential Prediction with Recurrent Log-Bilinear Model. TKDE 2017.

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



MBN

MBN: Towards multi-behavior sequence modeling for next basket recommendation. TKDD 2022.

Taxonomy

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

Methods	Prons	Cons
RNN-based	• Suitable for sequence problems and can store short-term memories	• Gradient disappearance & explosion problems • Inefficient in predicting future sequences • Rarely used currently
Graph-based		
Transformer-based		
Others		

A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

Taxonomy

- 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.

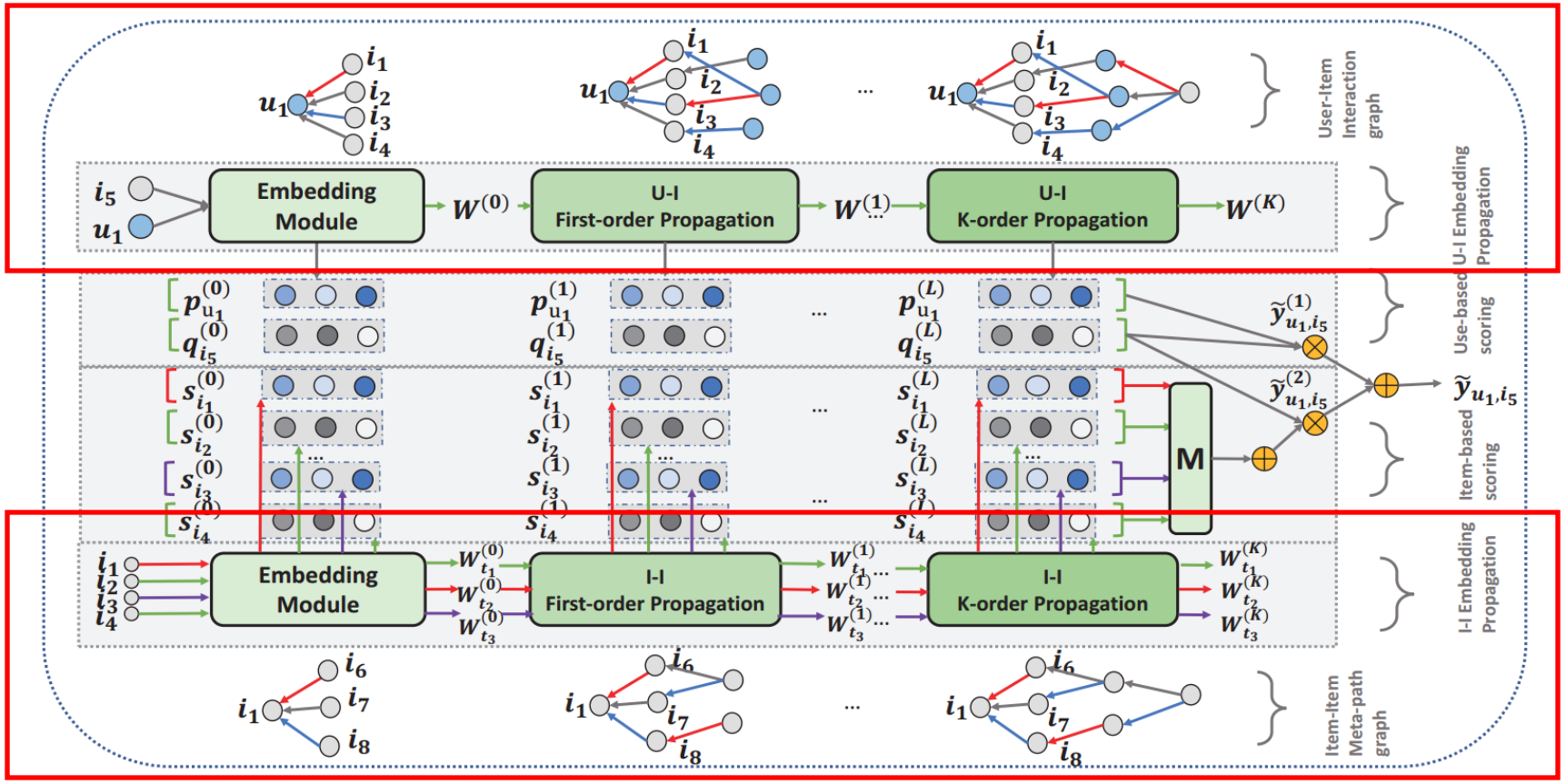
A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

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 Propogation

I-I Embedding Propogation



MB-GCN

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

Graph-based Methods—MB-GCN

- 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

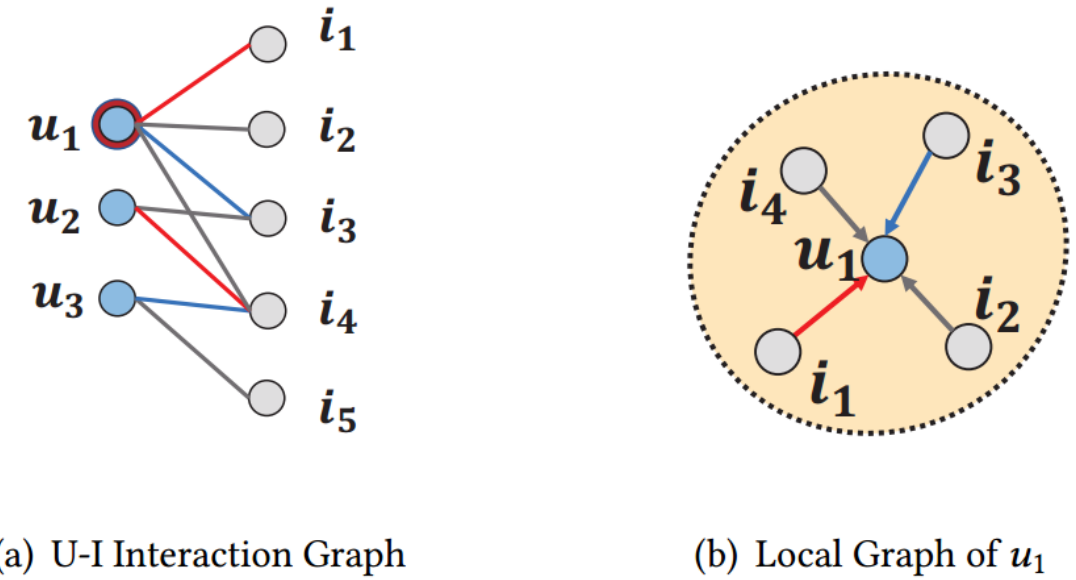
- Why?
  - The limitations of existing methods lie in the fact that they cannot thoroughly address the above two challenges: modeling user-to-item based strength and item-to-item based semantics of multiple types of behaviors.

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

Graph-based Methods—MB-GCN

➢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

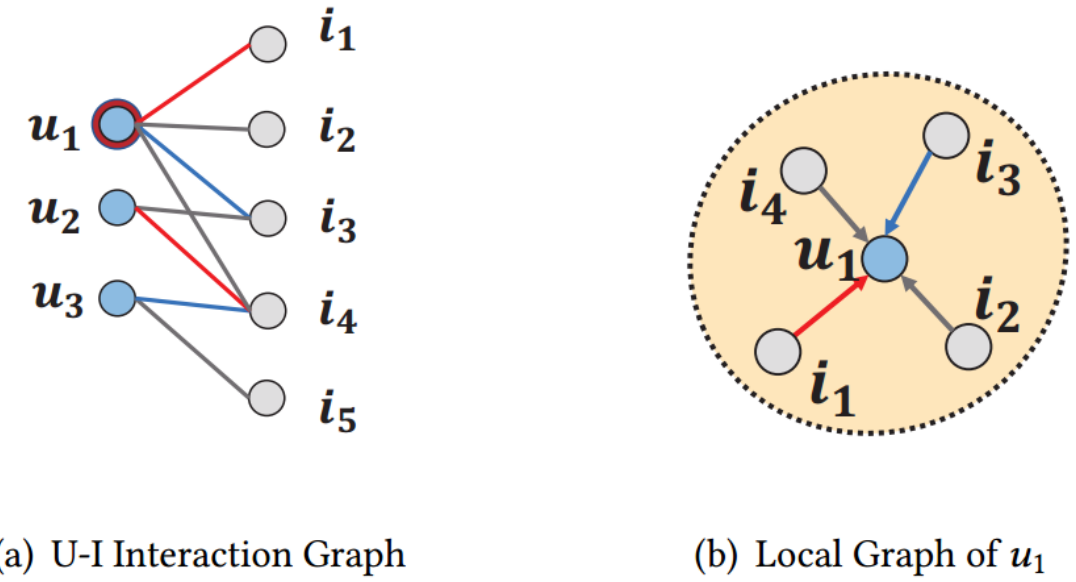
An illustration of the user-item multi-behavior graph

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

Graph-based Methods—MB-GCN

➢Framework design

- Input: The user-item interaction data of T types of behaviors, $\{Y^1, Y^2, \ldots, Y^T\}$
- 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.



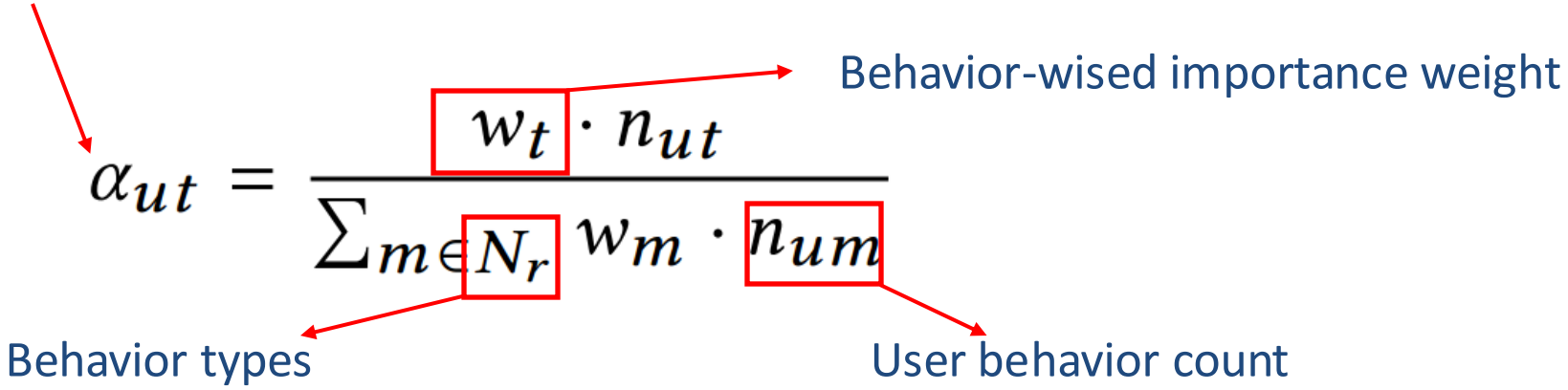
(a) U-I Interaction Graph (b) Local Graph of u_1

An illustration of the user-item multi-behavior graph

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

User Embedding Propagation

- 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



Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

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

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

User Embedding Propagation

- 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

Propagation weight

User embedding at layer l

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

Item Embedding Propagation

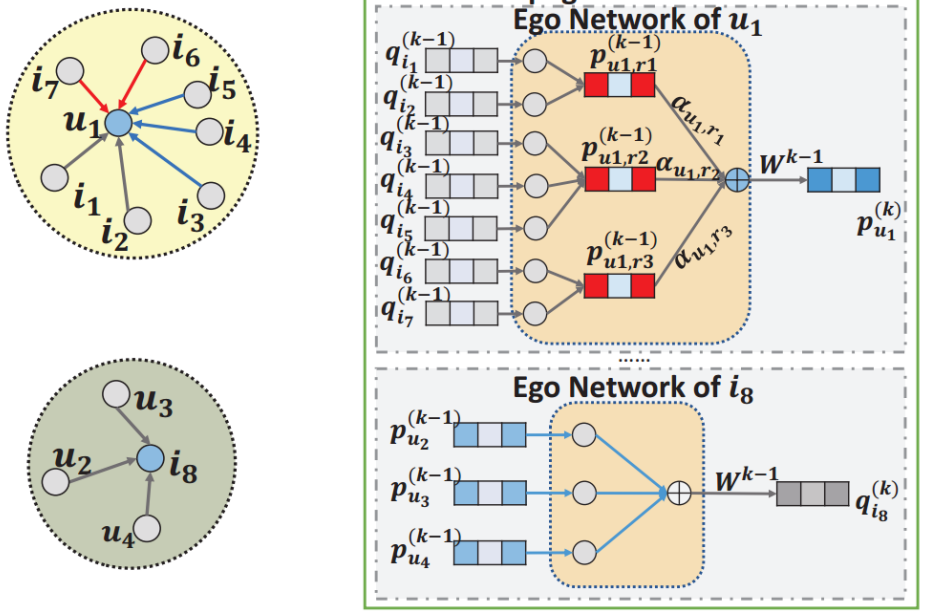
➢The feature of the item is static

- 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



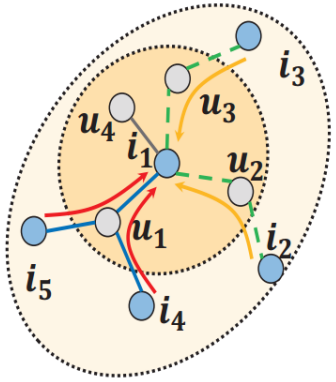
(a) Local Graph (b) k-order Propagation Process

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

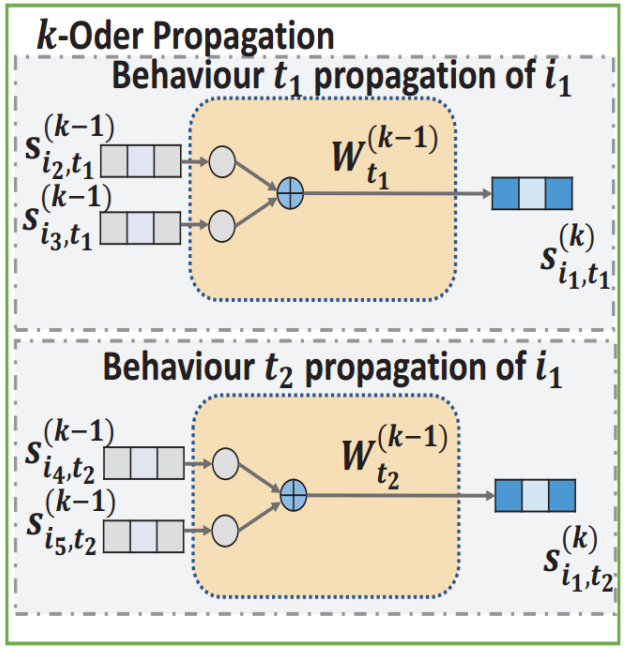
Item-Relevance Aware I-I Propagation

➢Item information extracting

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



(a) Local Graph



(b) k-order Propagation Process

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

Joint Prediction

- Embedding aggregation

$$p_u^* = p_u^{(0)}||...||p_u^{(L)},$$
$$q_i^* = q_i^{(0)}||...||q_i^{(L)},$$
$$s_{it}^* = s_{it}^{(0)}||...||s_{it}^{(L)}, t \in N_r$$

- 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

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

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

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

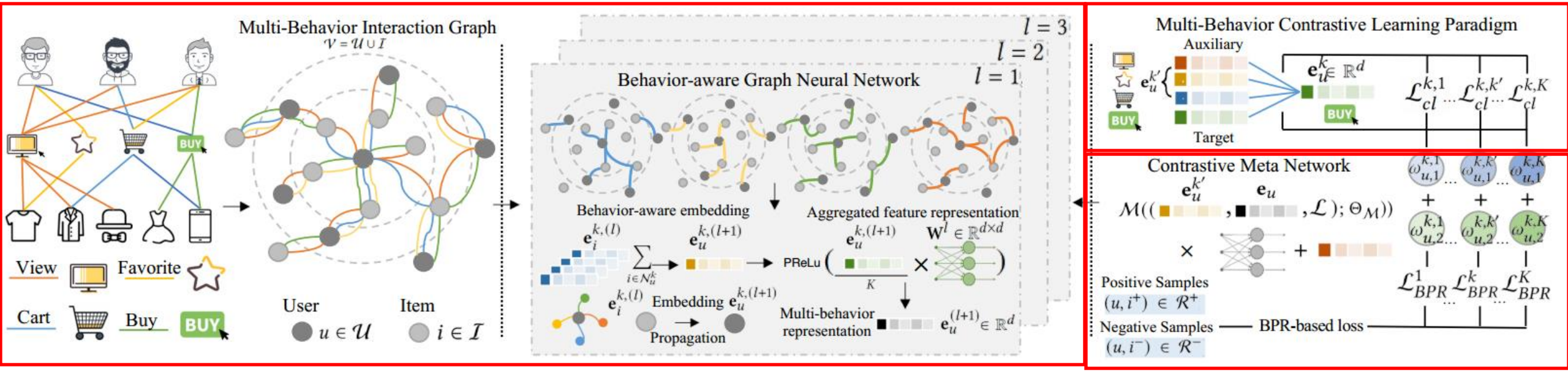
	Method	Recall@10	NDCG@10	Recall@20	NDCG@20	Recall@40	NDCG@40	Recall@80	NDCG@80
One-behavior	MF-BPR	0.03873	0.02286	0.05517	0.02676	0.08984	0.03388	0.14137	0.04258
	NCF	0.04209	0.02394	0.05609	0.02579	0.09118	0.03410	0.15426	0.04022
	GraphSAGE-OB	0.034536	0.01728	***0.06907***	0.02594	***0.11567***	0.03547	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

Multi-behavior recommendation with graph convolutional networks. SIGIR 2020.

Graph-based Methods—CML

➢Conducting constrastive learning among behaviors



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

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

Contrastive Meta Learning with Behavior Multiplicity for Recommendation. WSDM 2022.

Taxonomy

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

Methods	Prons	Cons
RNN-based	• Suitable for sequence problems and can store short-term memories	• Gradient disappearance & explosion problems • Inefficient in predicting future sequences • Rarely used currently
Graph-based	• Detailed modeling for behavior relations • Improved performance	• Suffering from low efficiency
Transformer-based		
Others		

A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

Taxonomy

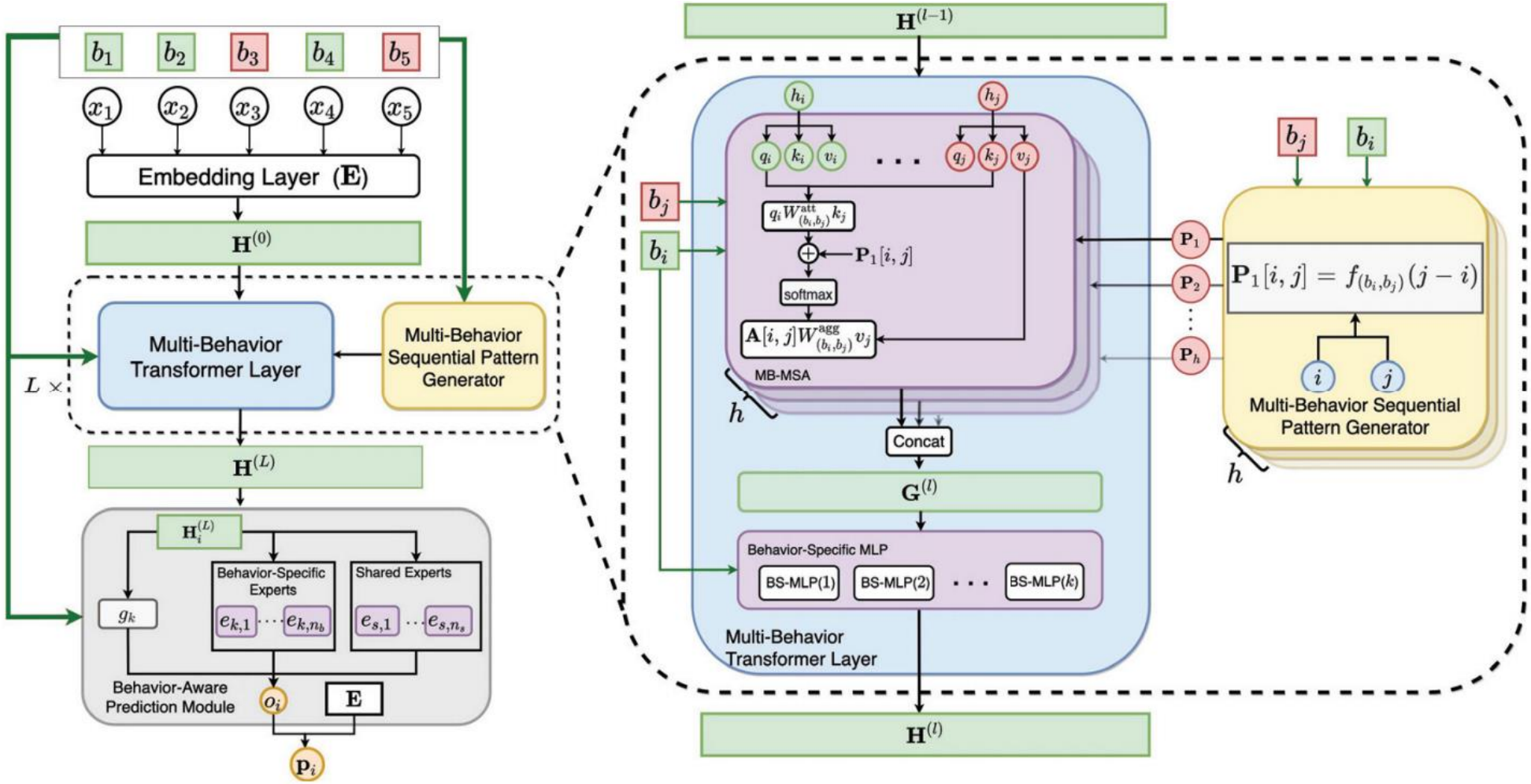
- RNN-based methods
- Graph-based methods
- Transformer-based methods
- Other methods

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.

A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

Transformer-based Methods—MB-STR

➢Conducting sequential modeling for multiple behaviors



Multi-behavior sequential transformer recommender. SIGIR 2022.

Transformer-based Methods—MB-STR

➢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 sequential transformer recommender. SIGIR 2022.

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

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

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

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

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

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

/* Step 2. Cross behavior similarity. */

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

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

end

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

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

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

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

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

end

end

$\mathbf{G}^{(l)} \leftarrow \mathbf{Concat}(\mathbf{G}^{(l)}_1, ..., \mathbf{G}^{(l)}_h)$

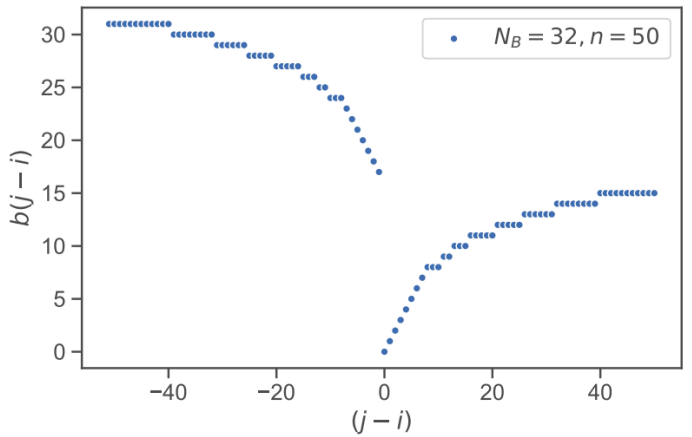
Multi-behavior sequential transformer recommender. SIGIR 2022.

Multi-Behavior Sequential Pattern Modeling

- A designed position encoding function for balanced position pairs
  - k: heads
  - i,j: behavior position in a sequence

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

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



Multi-behavior sequential transformer recommender. SIGIR 2022.

Behavior-Aware Masked Item Prediction

➢Gating & expert

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

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

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

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

Multi-behavior sequential transformer recommender. SIGIR 2022.

Overall performance

O/S: One/multiple behavior

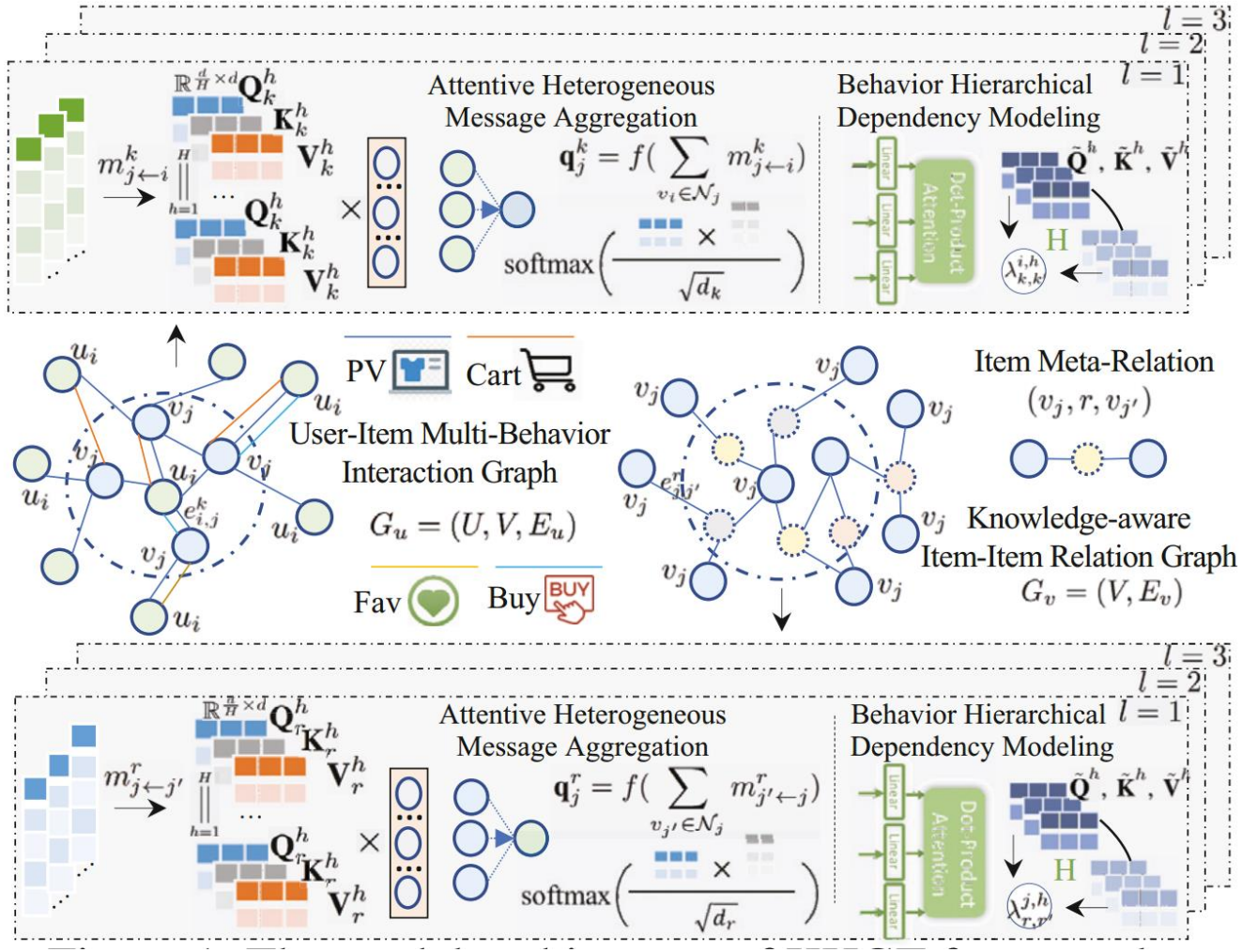
NS/S: Non-sequential/sequential

		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%

Multi-behavior sequential transformer recommender. SIGIR 2022.

Transformer-based Methods—KHGT

➢Conducting graph-structured transformer



Knowledge-enhanced hierarchical graph transformer network for multi-behavior recommendation. AAAI 2021.

Taxonomy

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

Methods	Prons	Cons
RNN-based	• Suitable for sequence problems and can store short-term memories	• Gradient disappearance & explosion problems • Inefficient in predicting future sequences • Rarely used currently
Graph-based	• Detailed modeling for behavior relations • Improved performance	• Suffering from low efficiency
Transformer-based	• Exceptional performance from attention mechanism • Superior parallel computing capabilities • Enhanced ability to capture long-term dependencies • Stronger explanability	\
Others		

A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

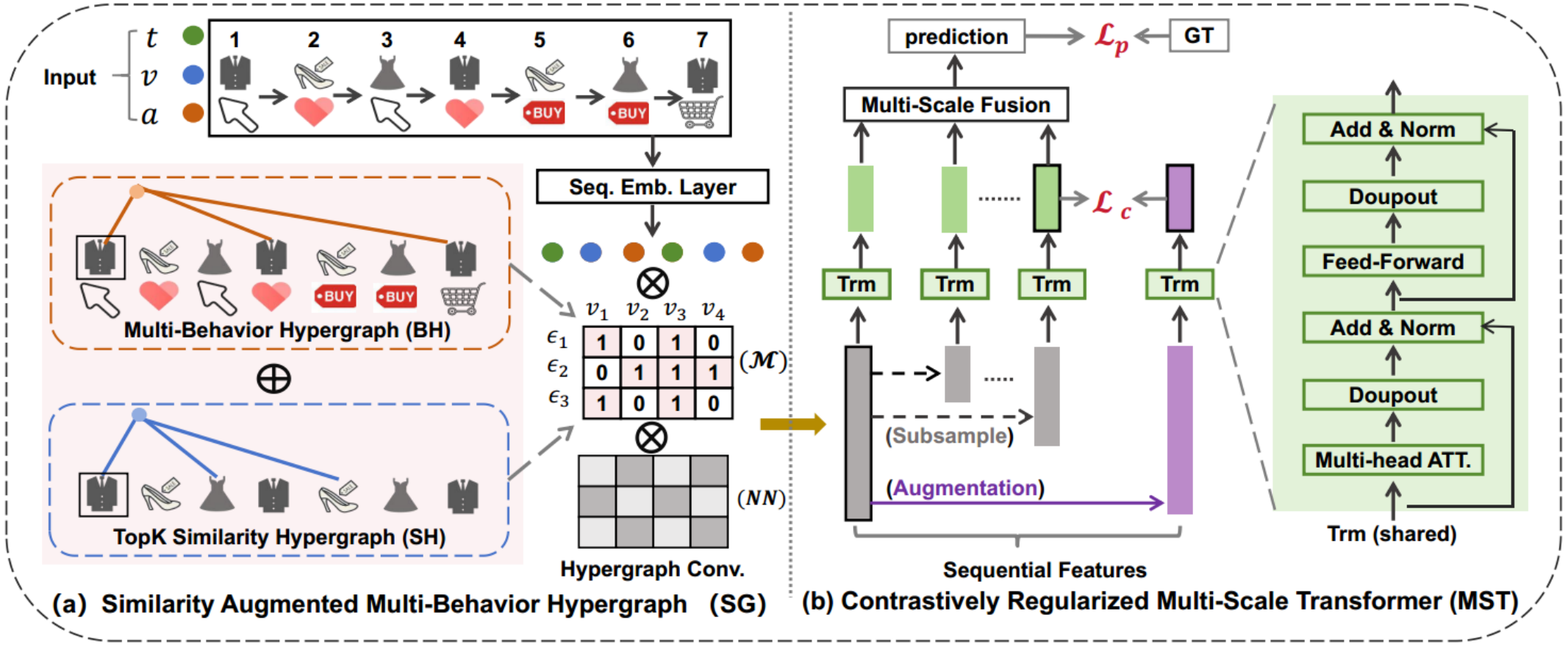
Taxonomy

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

A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

Other Methods—SG-MST

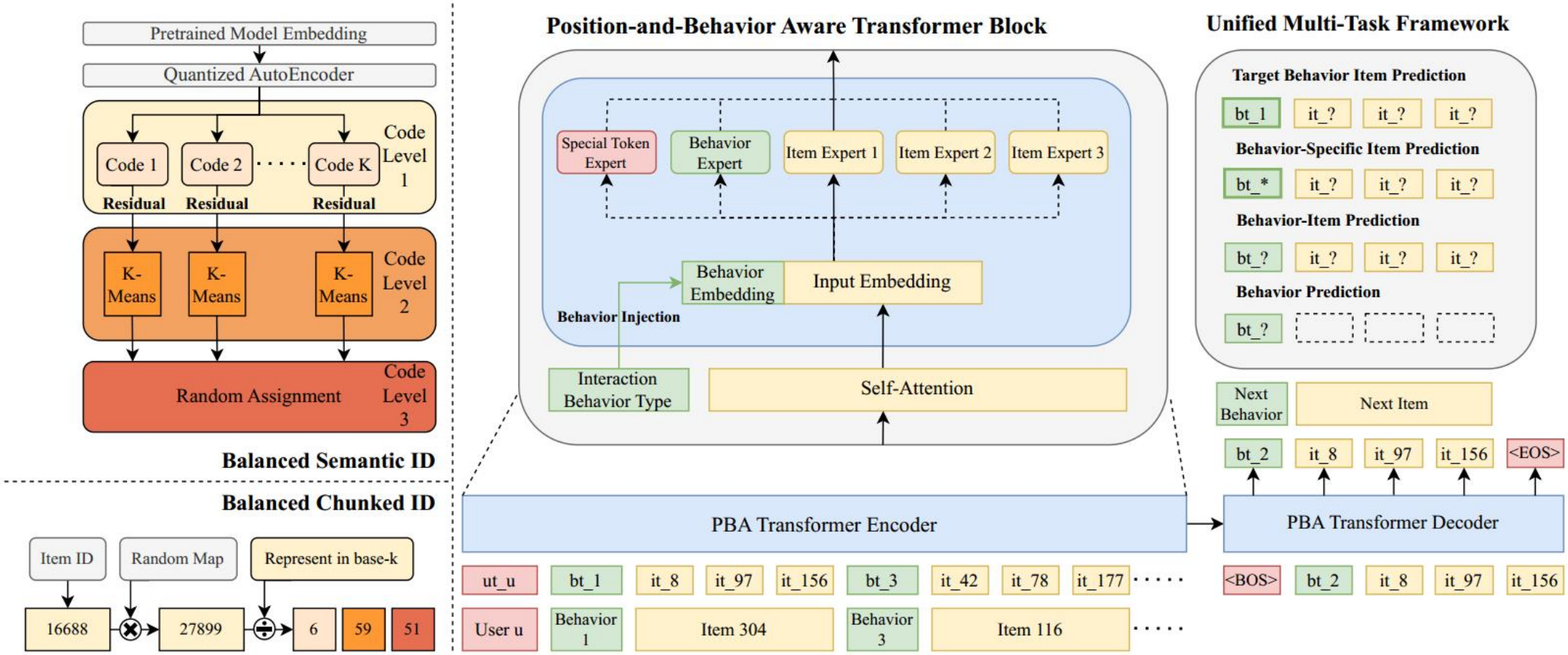
- Integrating a Similarity Augmented Multi-Behavior Hypergraph that captures complex behavior-aware dependencies among items and strengthens connections through item context similarities, producing more informative latent representations



Hypergraph-Enhanced Contrastively Regularized Transformer for Multi-Behavior E-commerce Product Recommendation. ICDM 2024.

Other Methods—MBGen

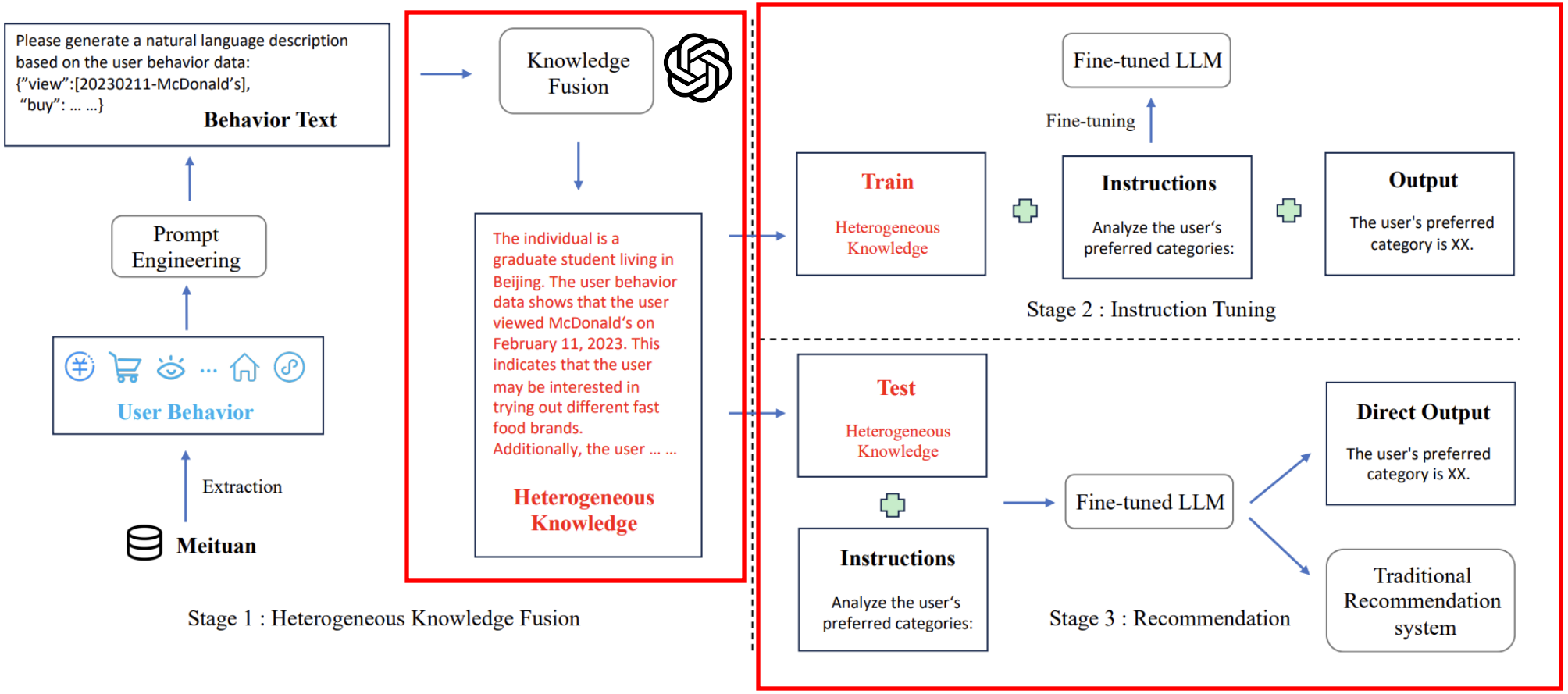
➢ Generative modeling for multi-behavior sequential recommendations



Multi-Behavior Generative Recommendation. CIKM 2024.

Other Methods—HKFR

➢ Representation modeling for behaviors via LLM



Heterogeneous knowledge fusion: A novel approach for personalized recommendation via llm. RecSys 2023.

Other Methods—HKFR

- 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
$HKFR_{no-IT}$	0.1241	0.1175	0.2267	0.1415	0.1014	0.0952	0.2050	0.1165
$HKFR_{no-HKF}$	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**

Heterogeneous knowledge fusion: A novel approach for personalized recommendation via llm. RecSys 2023.

Taxonomy

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

Methods	Prons	Cons
RNN-based	• Suitable for sequence problems and can store short-term memories	• Gradient disappearance & explosion problems • Inefficient in predicting future sequences • Rarely used currently
Graph-based	• Detailed modeling for behavior relations • Improved performance	• Suffering from low efficiency
Transformer-based	• Exceptional performance from attention mechanism • Superior parallel computing capabilities • Enhanced ability to capture long-term dependencies • Stronger explanability	\
Others	• Better modeling structure • Generative modeling • Modeling with LLM	

A survey on multi-behavior sequential recommendation. arxiv preprint 2023.

Conclusion

Methods	Prons	Cons
RNN-based	• Suitable for sequence problems and can store short-term memories	• Gradient disappearance & explosion problems • Inefficient in predicting future sequences • Rarely used currently
Graph-based	• Detailed modeling for behavior relations • Improved performance	• Suffering from low efficiency
Transformer-based	• Exceptional performance from attention mechanism • Superior parallel computing capabilities • Enhanced ability to capture long-term dependencies • Stronger explanability	\
Others	• 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

Future Directions

- **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

Agenda

Introduction

Yejing Wang

Joint Modeling in RS

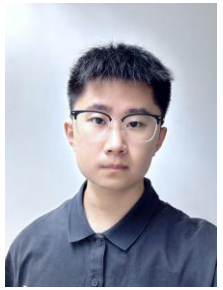
Pengyue Jia

Multi-task Recommendation

Yuhao Wang

Multi-scenario Recommendation

Pengyue Jia

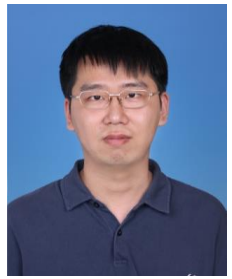


Xiaopeng Li

Multi-behavior Recommendation

Jingtong Gao

Multi-modal Recommendation



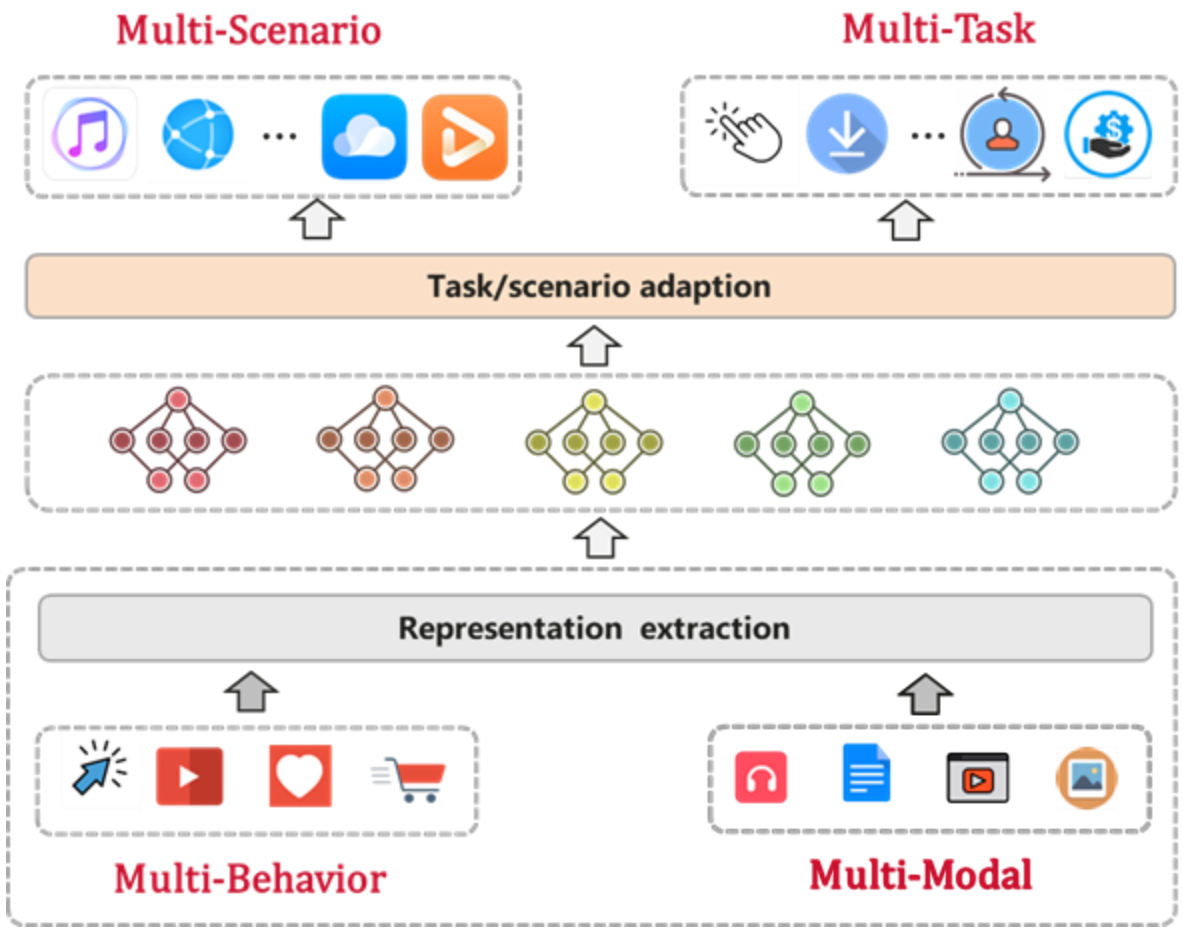
Qidong Liu

Conclusion

Future Work

Yichao Wang

Multi-Modal Modeling



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

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

$$\boldsymbol{E}^{B} = G(\boldsymbol{H}_1, \boldsymbol{H}_2, \ldots, \boldsymbol{H}_N)$$ — Multi-behavior

$$\boldsymbol{E}^{M} = M(\boldsymbol{E}^{txt}, \boldsymbol{E}^{v}, \ldots, \boldsymbol{E}^{p})$$ — Multi-modal

$$wL(\boldsymbol{E}^{Merge}, \boldsymbol{\theta}^{sh}, \theta^{t}, \boldsymbol{\theta}^{s})$$ — Multi-scenario

$$wL(\boldsymbol{E}^{Merge}, \boldsymbol{\theta}^{sh}, \boldsymbol{\theta}^{t}, \theta^{s})$$ — Multi-task

Formulation

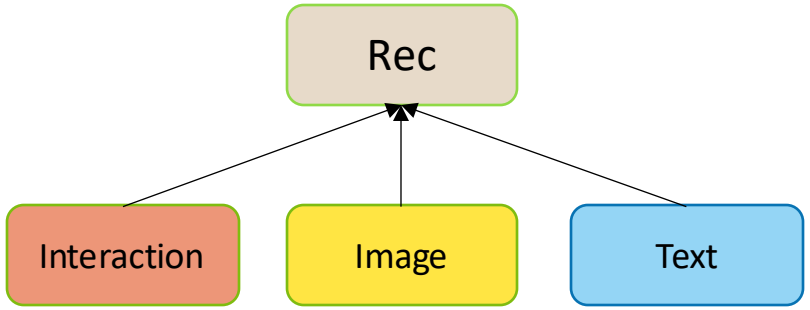
- **Problem:**
  - Learning a unified multimodal representations for users and items by various raw multimodal features $(x^{txt}, x^{v}, \ldots, x^{p})$
- **Optimization problem:**

$$E^{M} = M\left(E^{txt}, E^{v}, \ldots, E^{p}\right) = M\left(\mathcal{E}_{txt}(x^{txt}), \mathcal{E}_{v}(x^{v}), \ldots, \mathcal{E}_{p}(x^{p})\right)$$

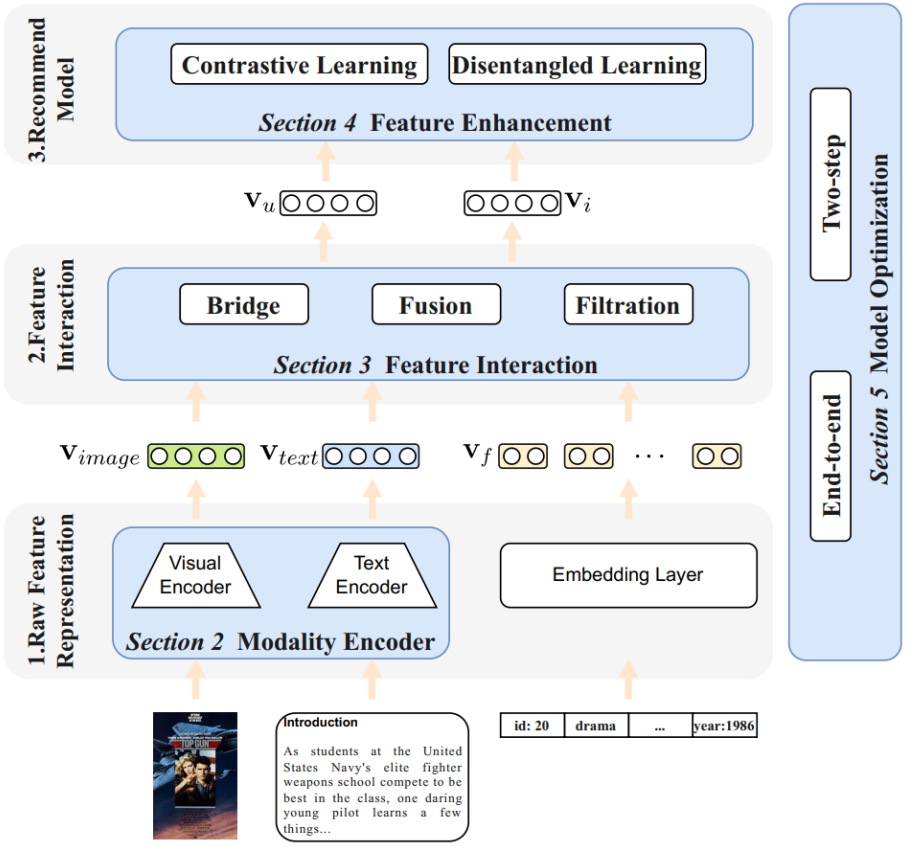
- $\mathcal{E}_{*}(\cdot)$: the corresponding modality encoder
- $M(\cdot)$: feature interaction function, enabling combination of various modalities

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



Multi-modal recommendation

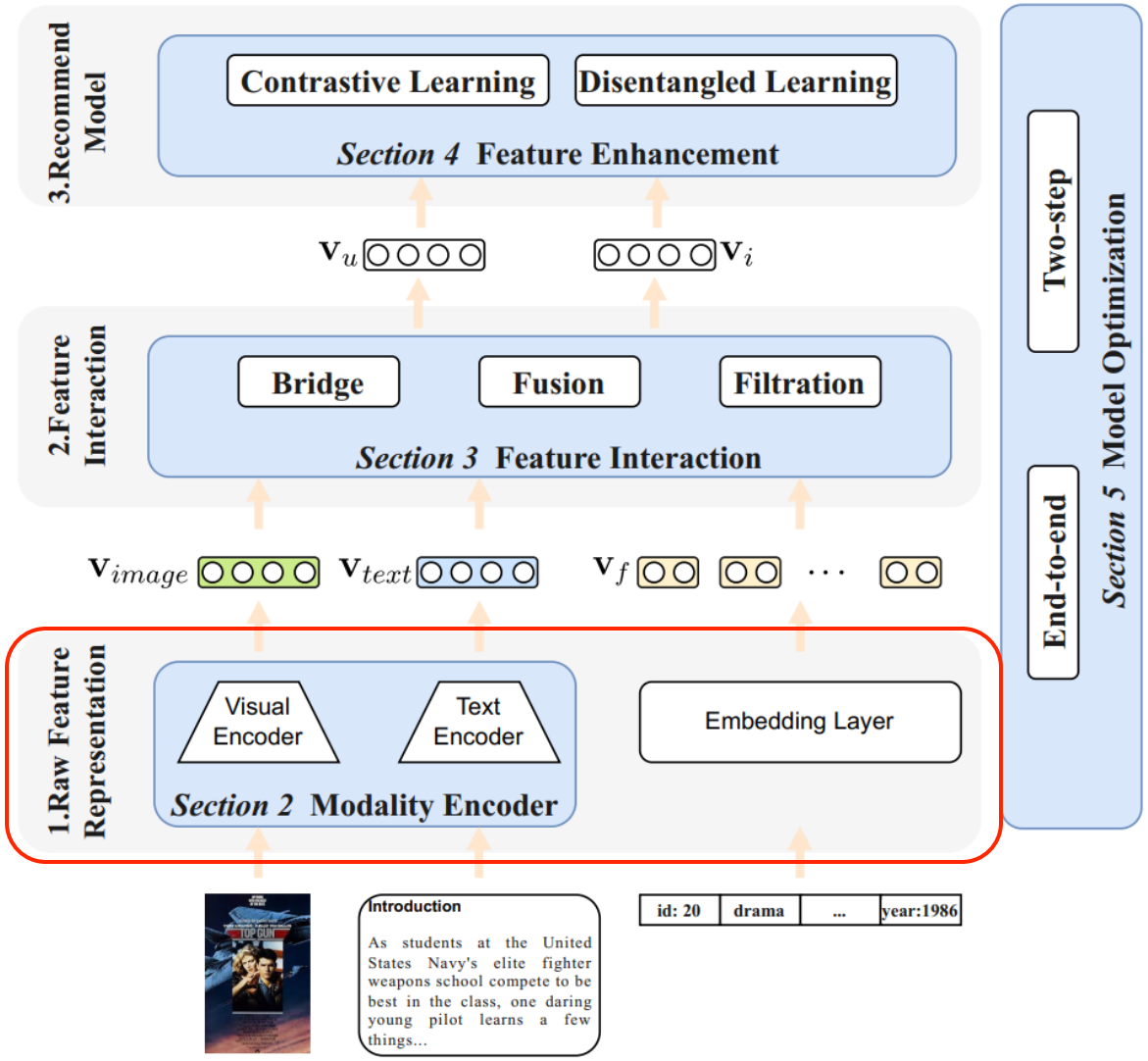


Procedures of MRS

Multimodal Recommender Systems: A Survey. ACM Computing Surveys 2024.

Challenges

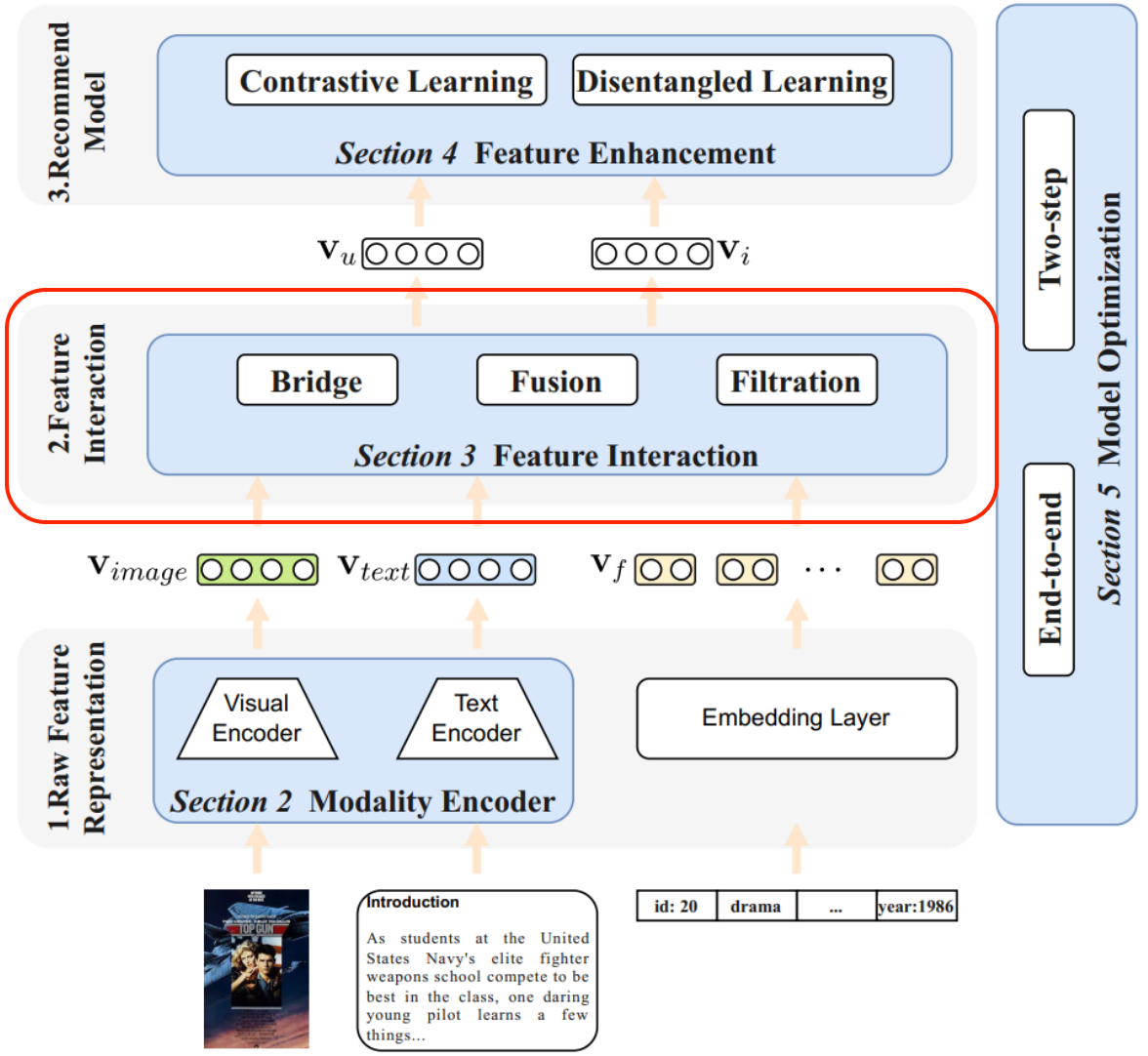
- Modality Encoder
  - Challenge: how to extract representations from complex raw features
  - Specialty: various encoders



Procedures of MRS

Challenges

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

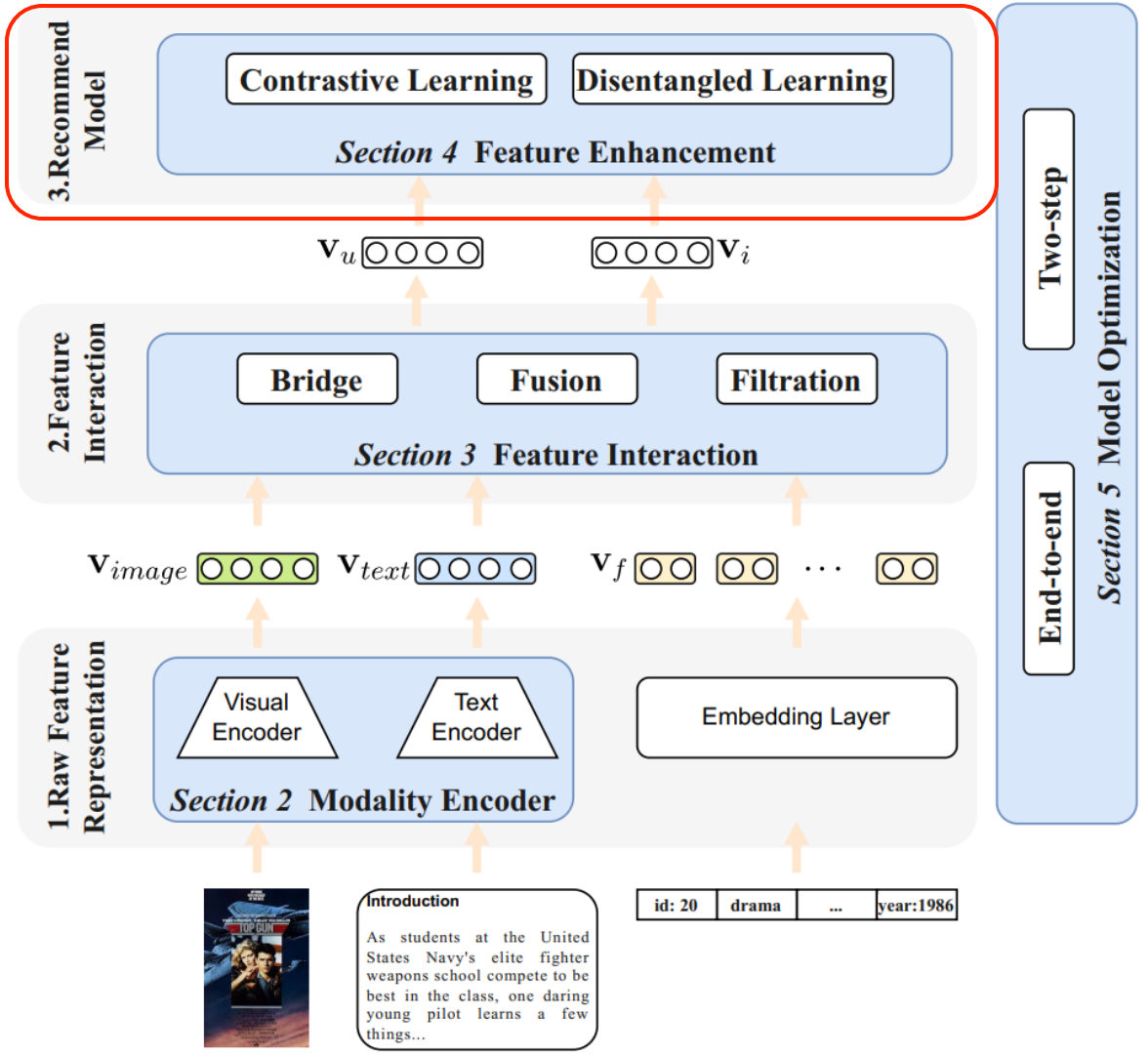


Procedures of MRS

Multimodal Recommender Systems: A Survey. ACM Computing Surveys 2024.

Challenges

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

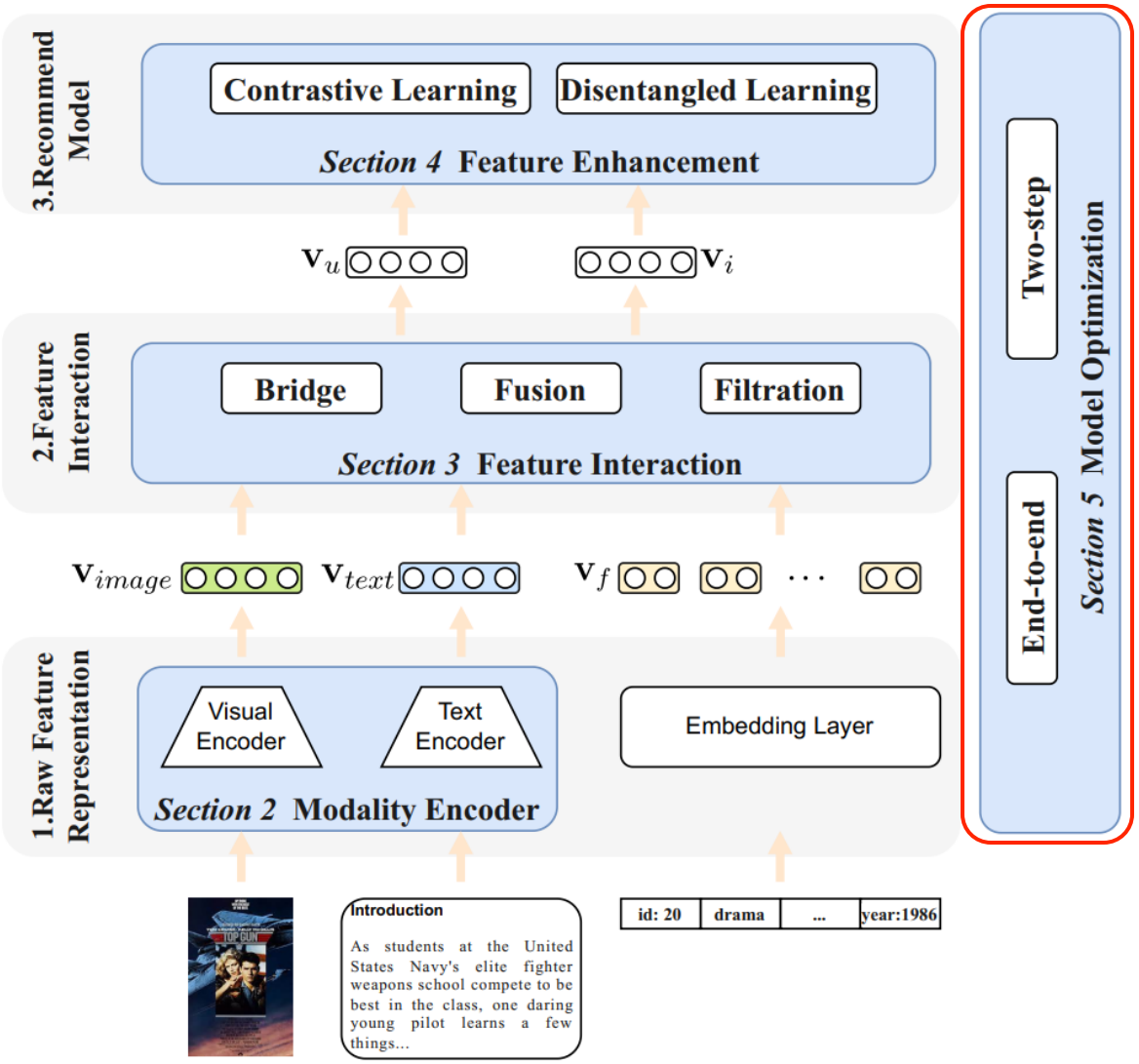


Procedures of MRS

Multimodal Recommender Systems: A Survey. ACM Computing Surveys 2024.

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



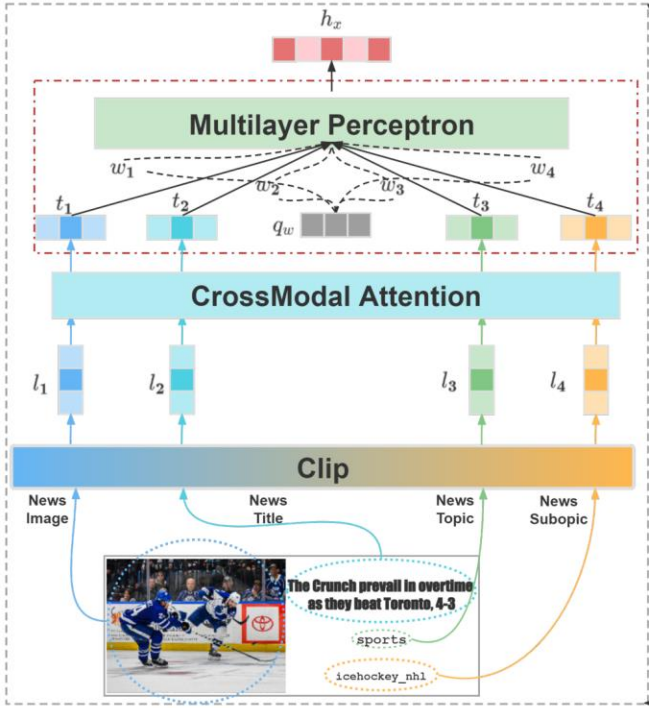
Procedures of MRS

Multimodal Recommender Systems: A Survey. ACM Computing Surveys 2024.

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



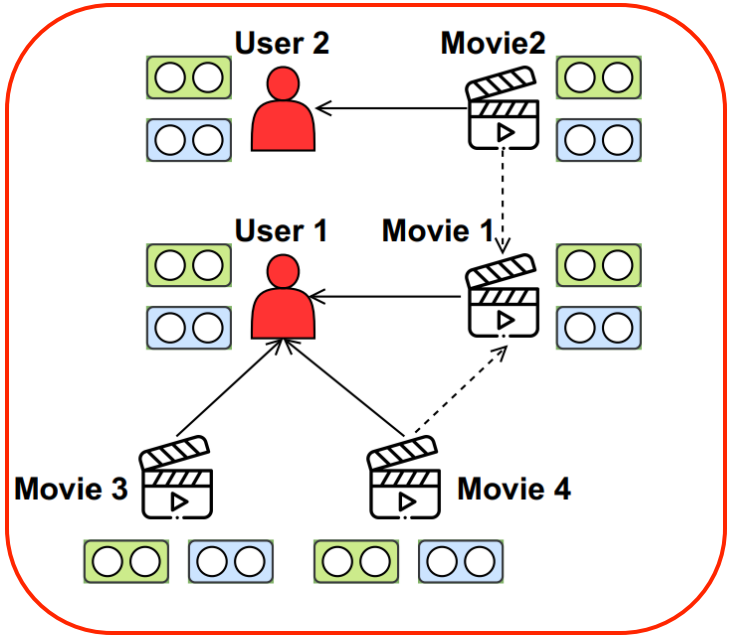
MultiModal Encoder

Example: Multimodal encoder in VLSNR: Clip+ViT

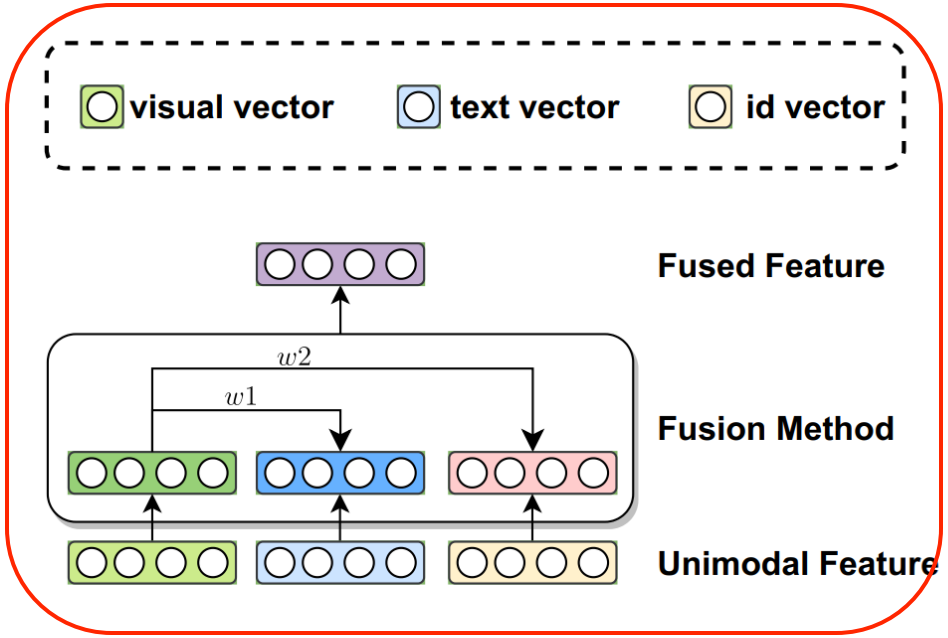
VLSNR: Vision-Linguistics Coordination Time Sequence-aware News Recommendation. arXiv preprint 2022.

Feature Interaction

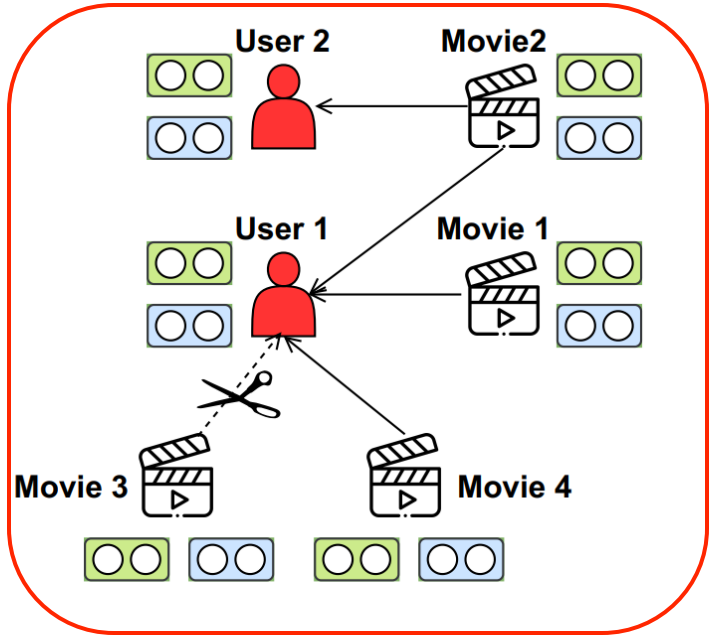
- Target: connecting various modalities
- Taxonomy:
  - **Bridge**: capturing inter-relationship between users and items considering modalities
  - **Fusion**: capturing multimodal intra-relationships of items
  - **Filtration**: filtering out noisy data in interaction graph or multimodal features



(a) Bridge



(b) Fusion



(c) Filtration

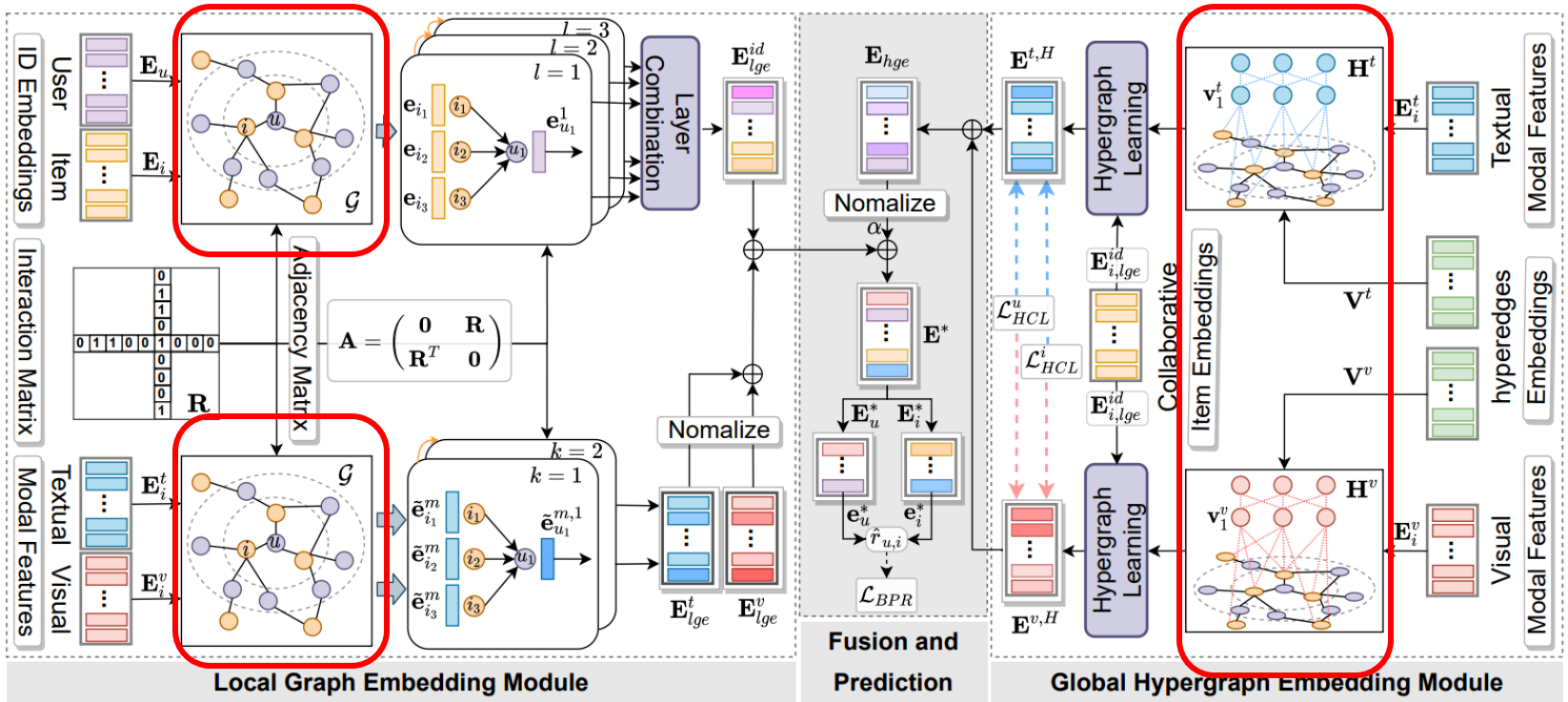
Feature Interaction: Bridge

➢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



LGMRec: Local and Global Graph Learning for Multimodal Recommendation. TAAAI 2024.

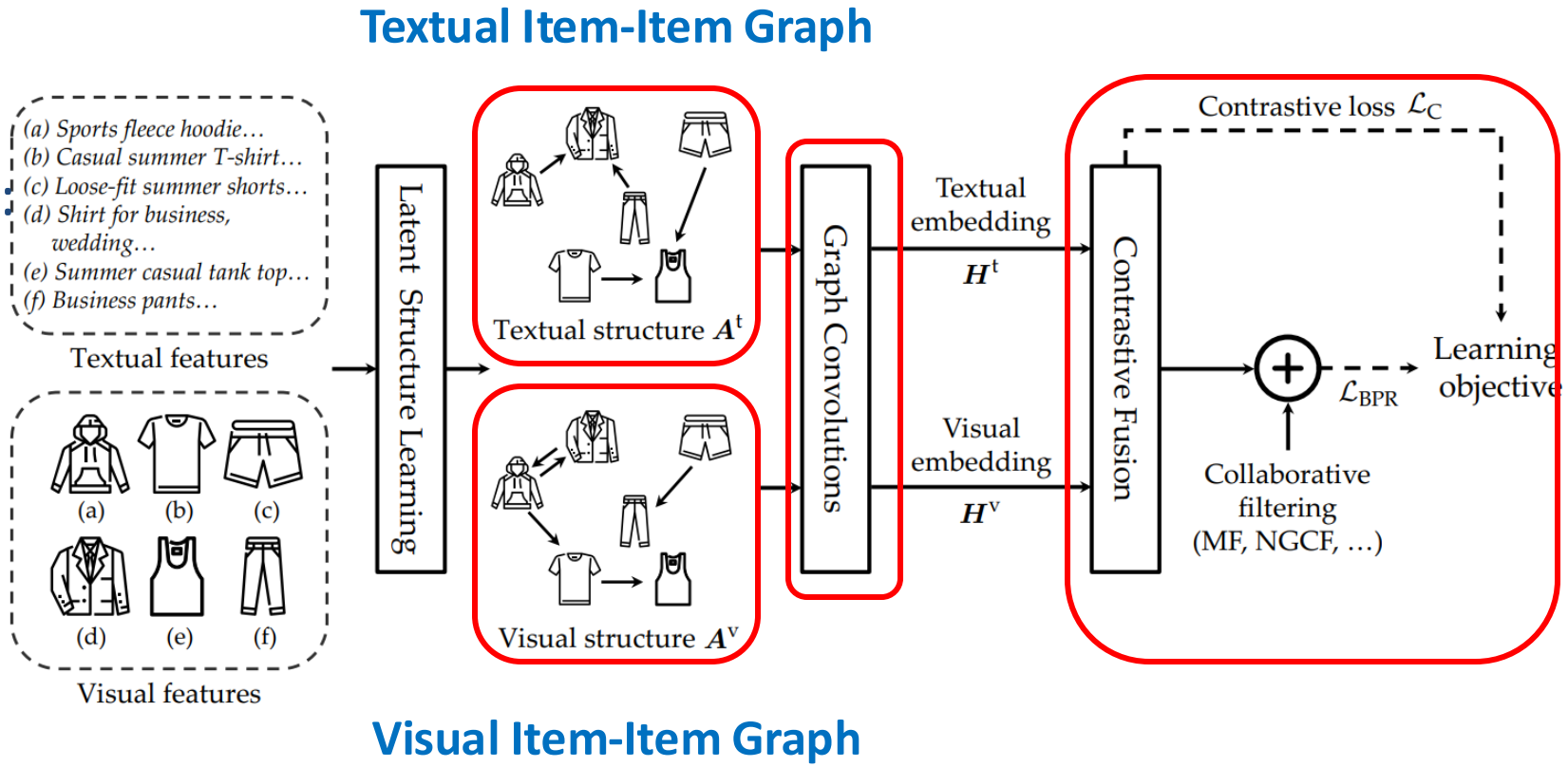
Feature Interaction: Bridge

➢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



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

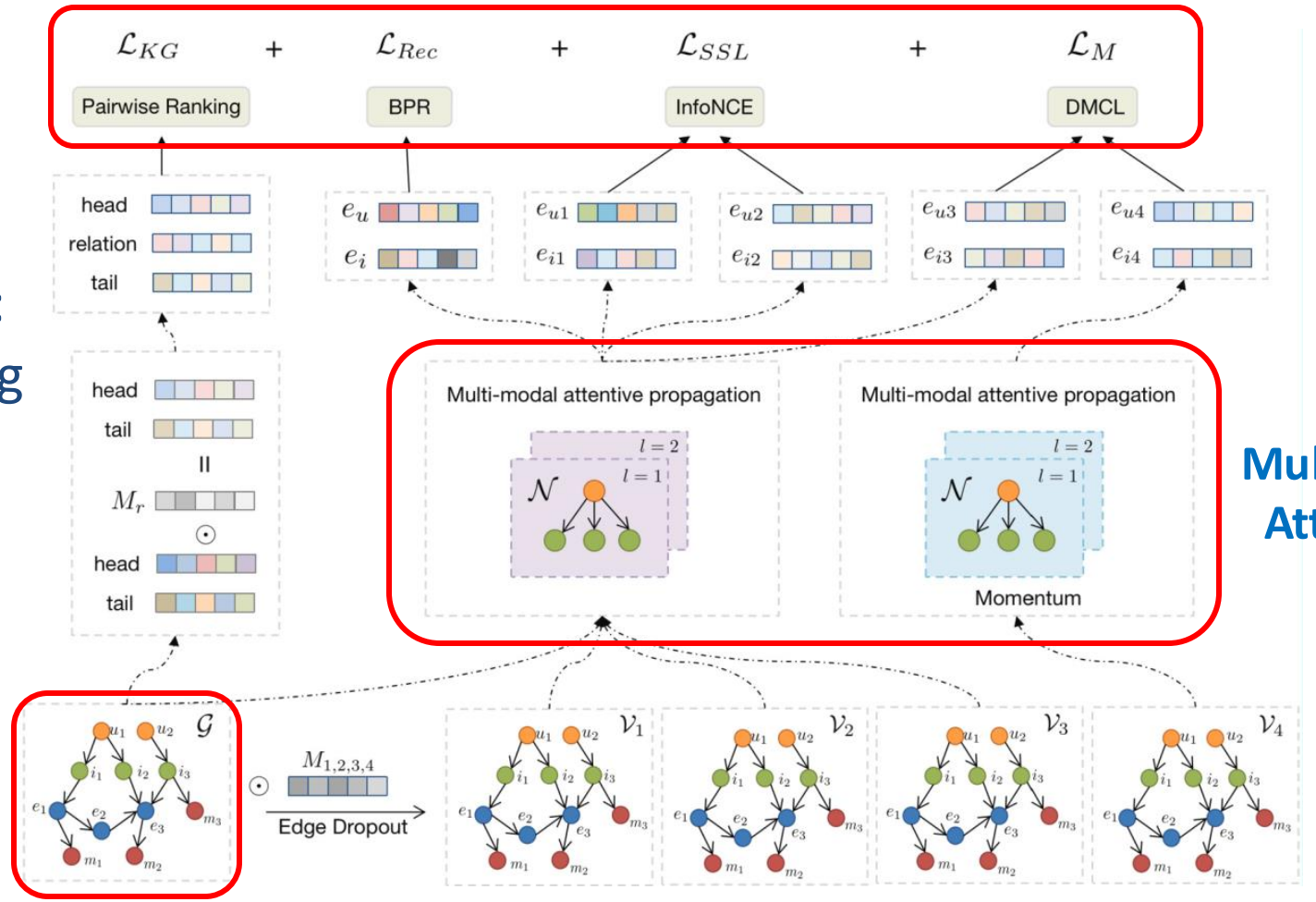
Feature Interaction: Bridge

➢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



M3KGR: A momentum contrastive multi-modal knowledge graph learning framework for recommendation. Information Science 2024.

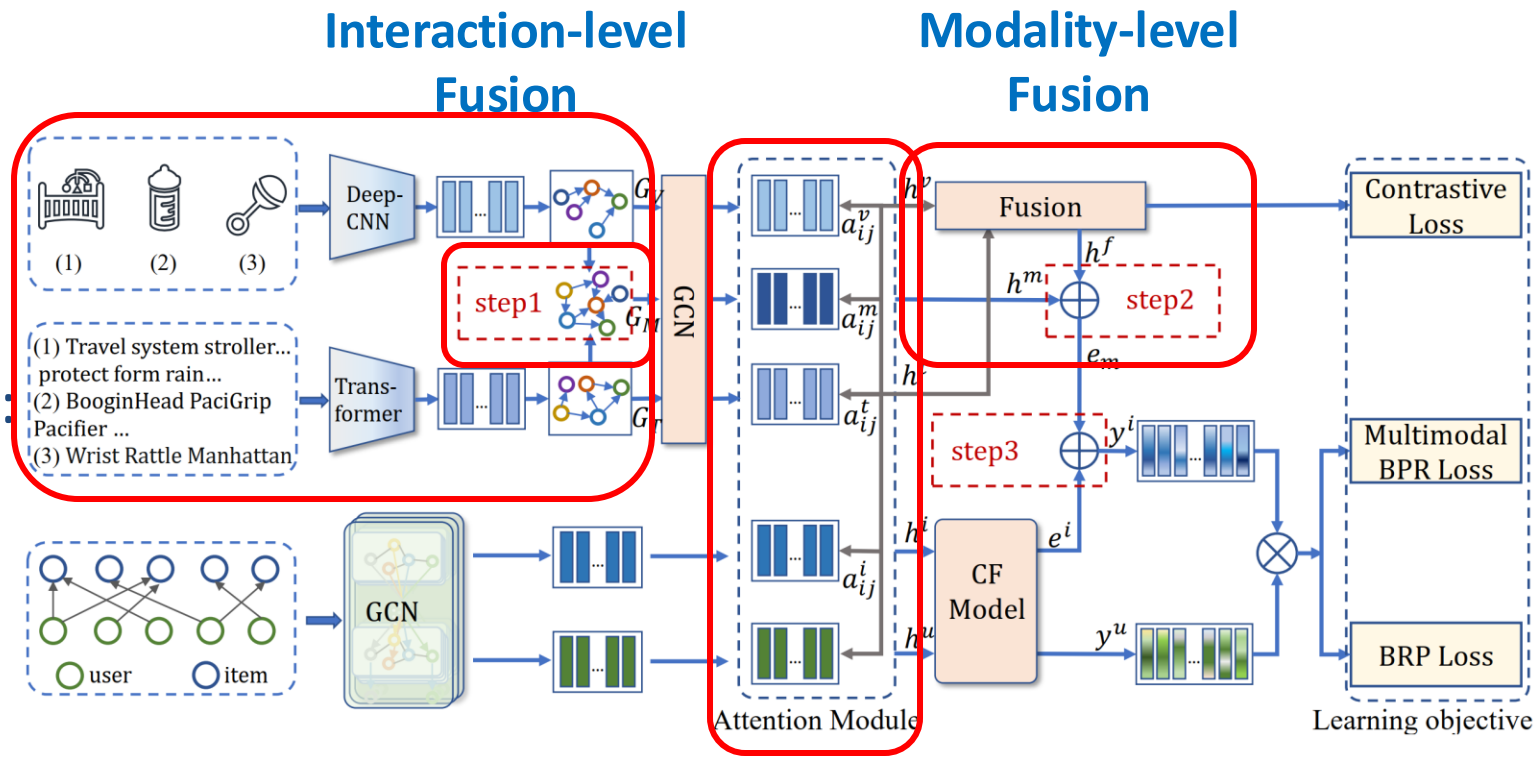
Feature Interaction: Fusion

➢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



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

Attention-guided multi-step fusion: a hierarchical fusion network for multimodal recommendation. SIGIR 2023.

Feature Interaction: Fusion

➢Fine-grained Attention

- Capturing the multimodal relationships between modalities

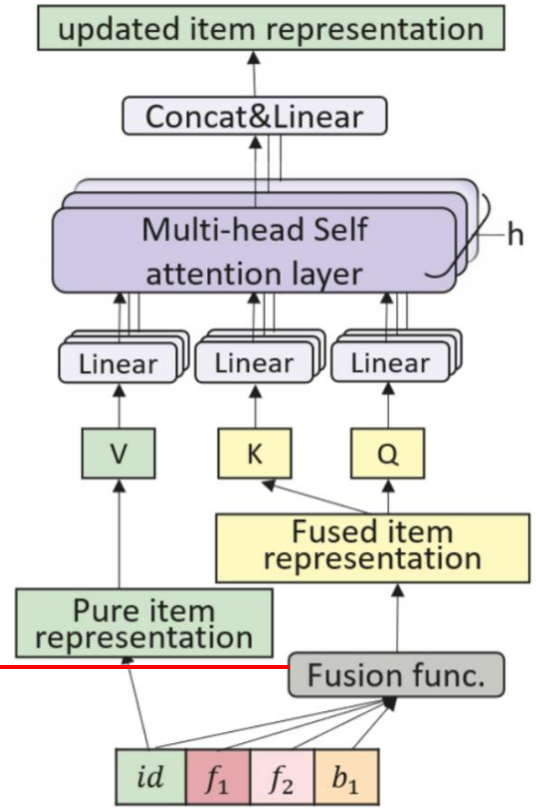
➢NOVA (AAAI'21)

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

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

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

$$G = \sigma(FW^F)$$



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

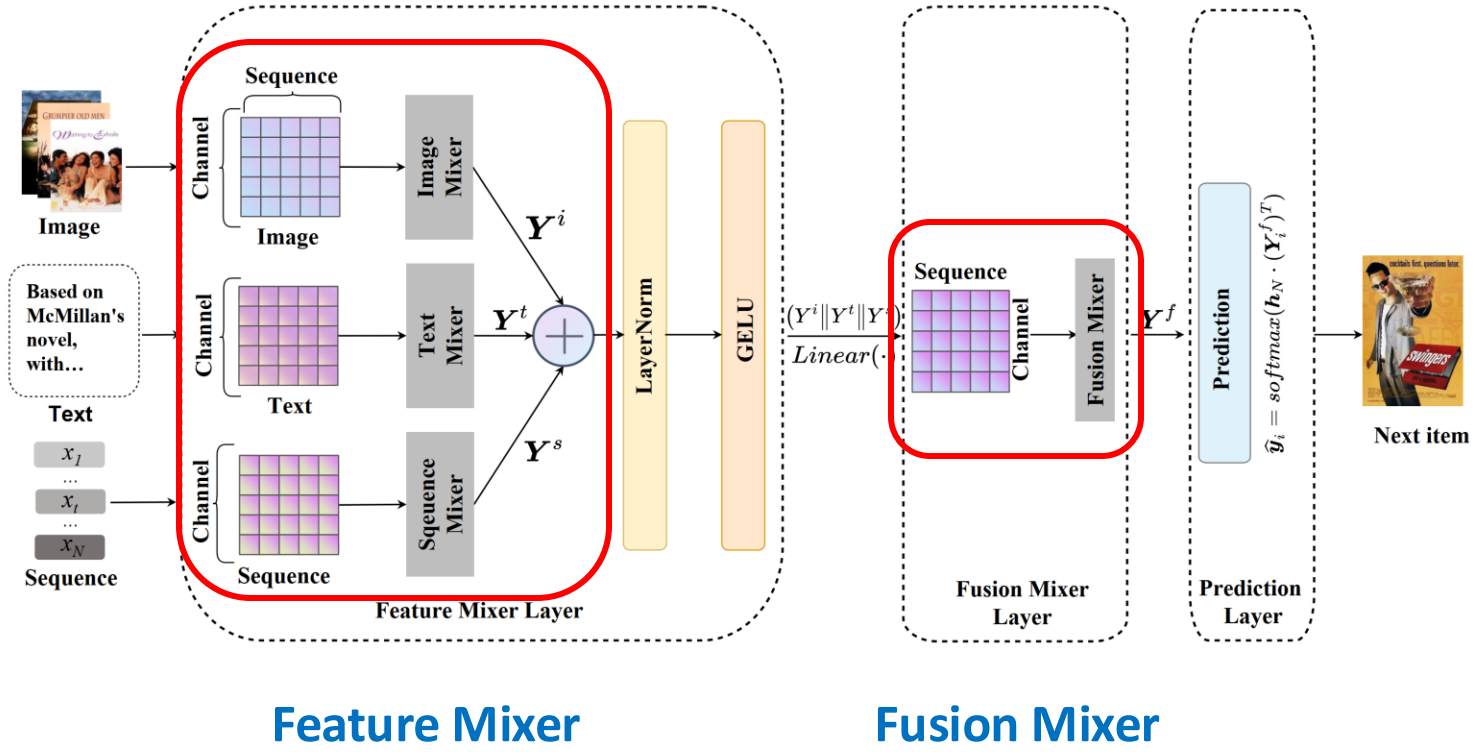
Feature Interaction: Fusion

➢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

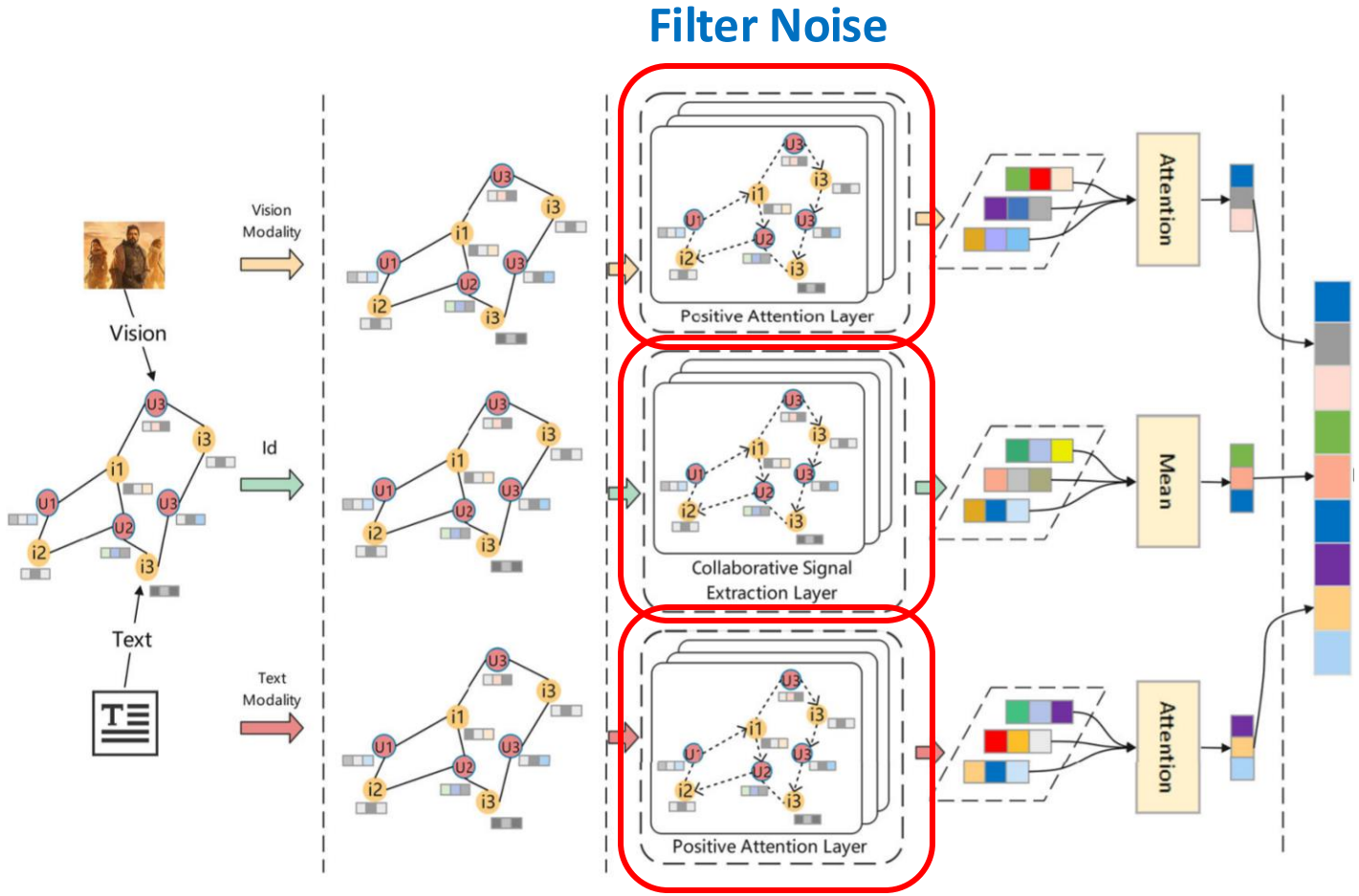


MMMLP: multi-modal multilayer perceptron for sequential recommendations. WWW 2023.

Feature Interaction: Filtration

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

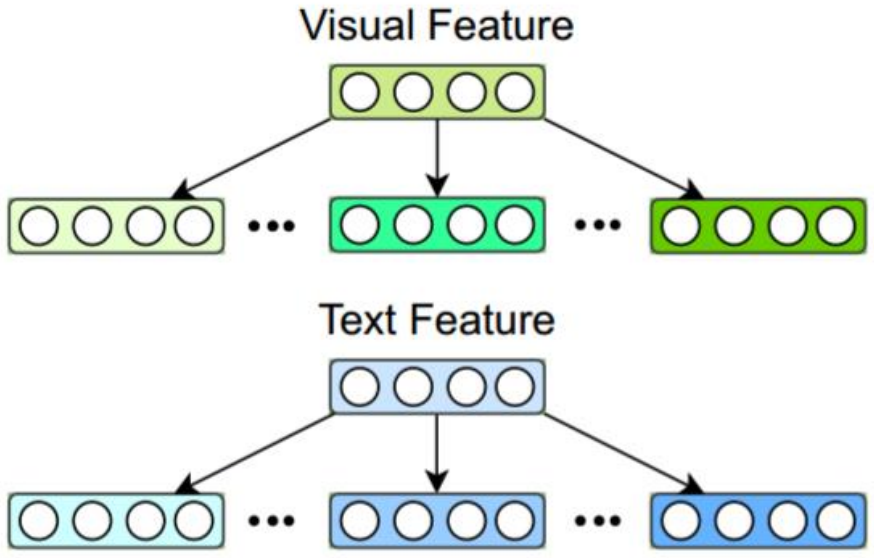
- PMGCRN (APPL INTELL'23)
  - **Positive Attention Layer**: remove implicit noisy edges by node similarity
  - **Collaborative Signal Extraction Layer**: integrate original user-item interaction information



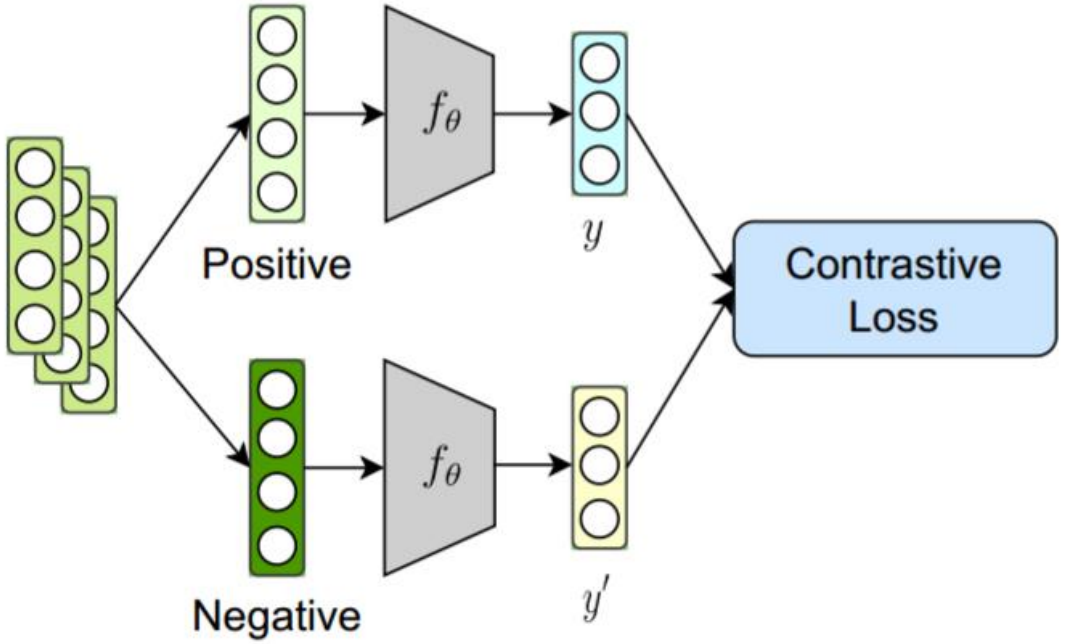
Preference-corrected multimodal graph convolutional recommendation network. Applied Intelligence 2023.

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

Feature Enhancement

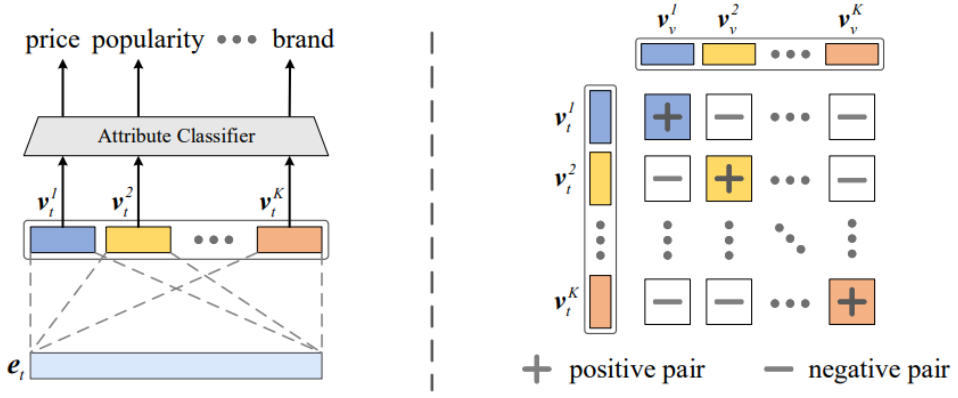
➢Disentangled Representation Learning (DRL)

- 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

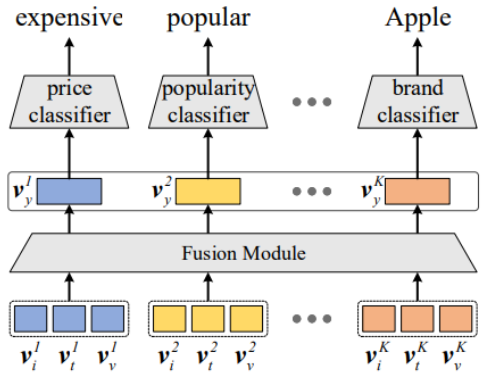
Low-level



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

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

High-level



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

Attribute-driven disentangled representation learning for multimodal recommendation. MM 2024.

Feature Enhancement

➢Contrastive Learning (CL)

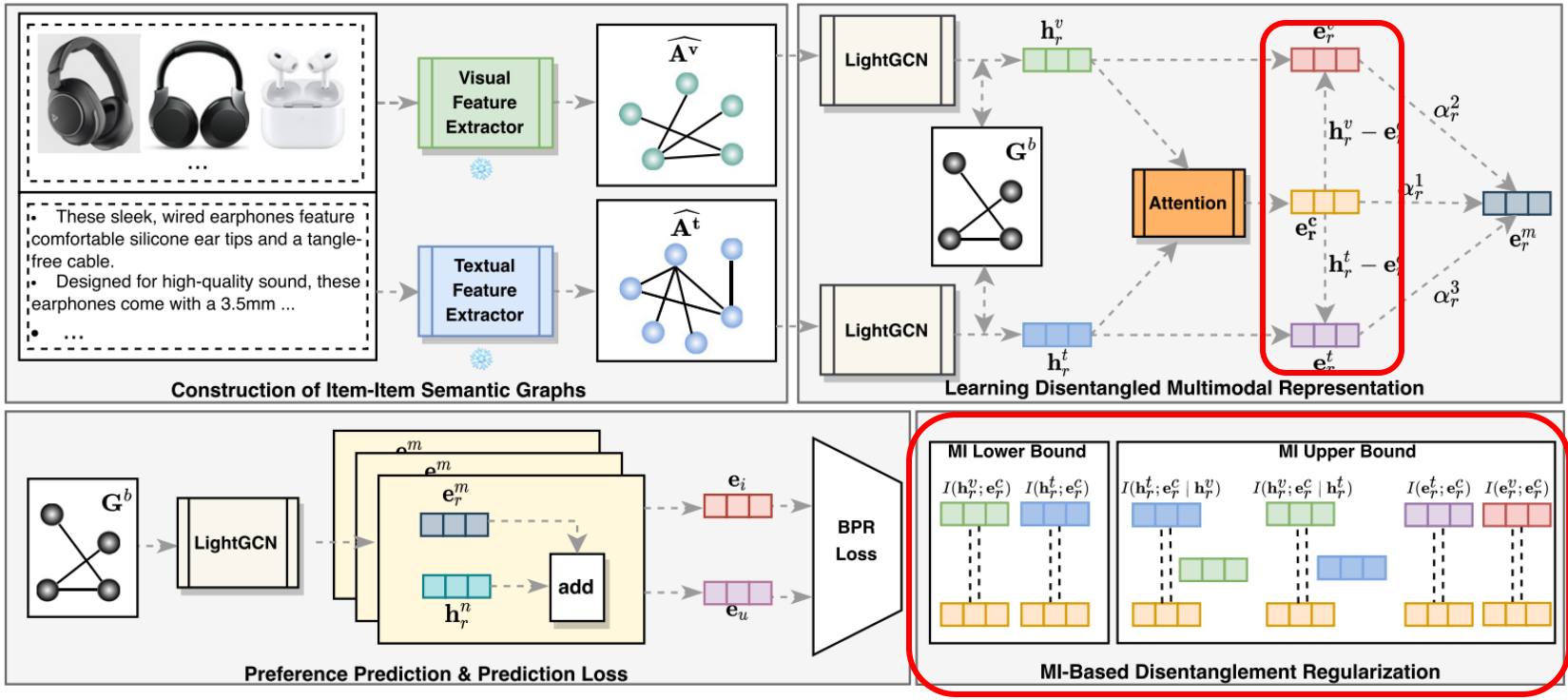
- minimizing the distance between modalities of one item and maximizing the distance between the modalities of different ones

Modality-shared

Modality-specific

➢CMDL (TOIS'25)

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

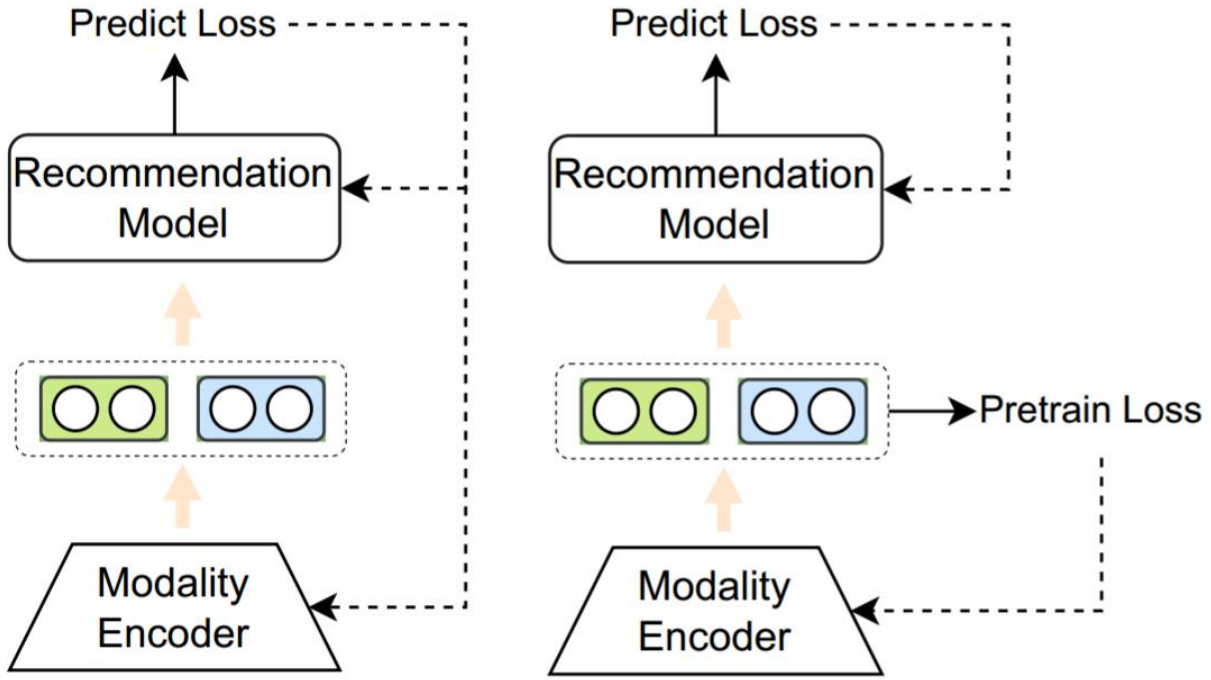


Extended CL Loss: $\mathcal{L}^{fub}(X,Y) = \mathbb{E}_{(x,y)\sim p(x,y)}[\hat{f}(x,y)] - \mathbb{E}_{x\sim p(x)}\mathbb{E}_{y\sim p(y)}[\hat{f}(x,y^-)],$

Contrastive Modality-Disentangled Learning for Multimodal Recommendation. TOIS 2025.

Model Optimization

- Target: meeting the computational requirements for training MRS, including the multimodal encoders and RS model
- Taxonomy:
  - **End-to-end Training** and **Two-step Training**

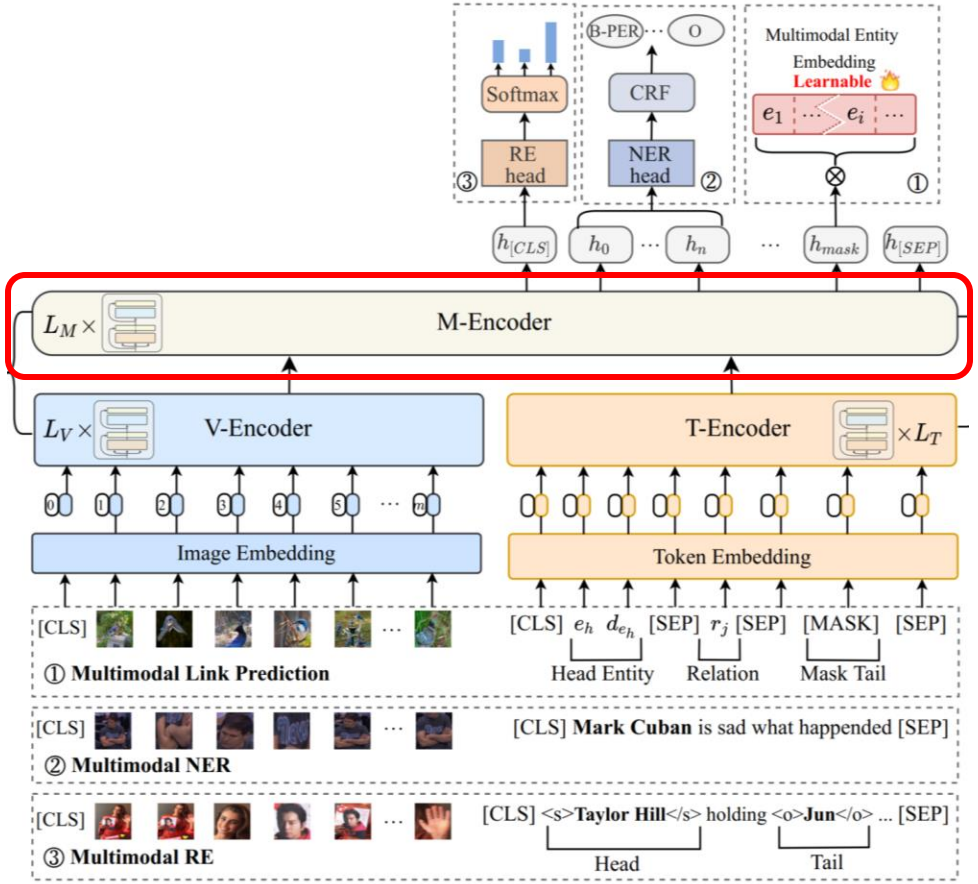


(a) End-to-end Training (b) Two-step Training

Model Optimization

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

- MKGformer (SIGIR'22)
  - **Modality-specifc Parameters**: each modality maintain their own layers, i.e., V-Encoder and T-Encoder
  - **Modality-shared Parameters**: both modalities share serveal layers, i.e., M-Encoder

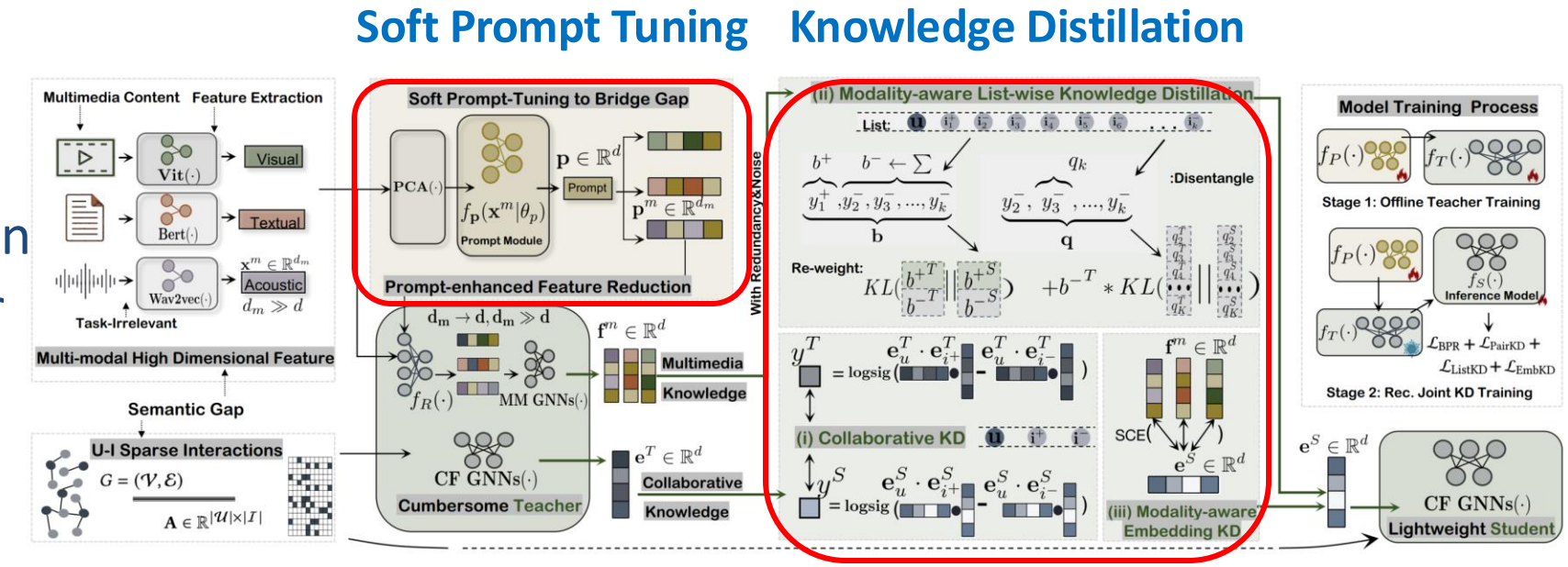


Hybrid transformer with multi-level fusion for multimodal knowledge graph completion. SIGIR 2022.

Model Optimization

- 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



PromptMM: Multi-Modal Knowledge Distillation for Recommendation with Prompt-Tuning. WWW 2024.

Large Language Model for MRS

➢ 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

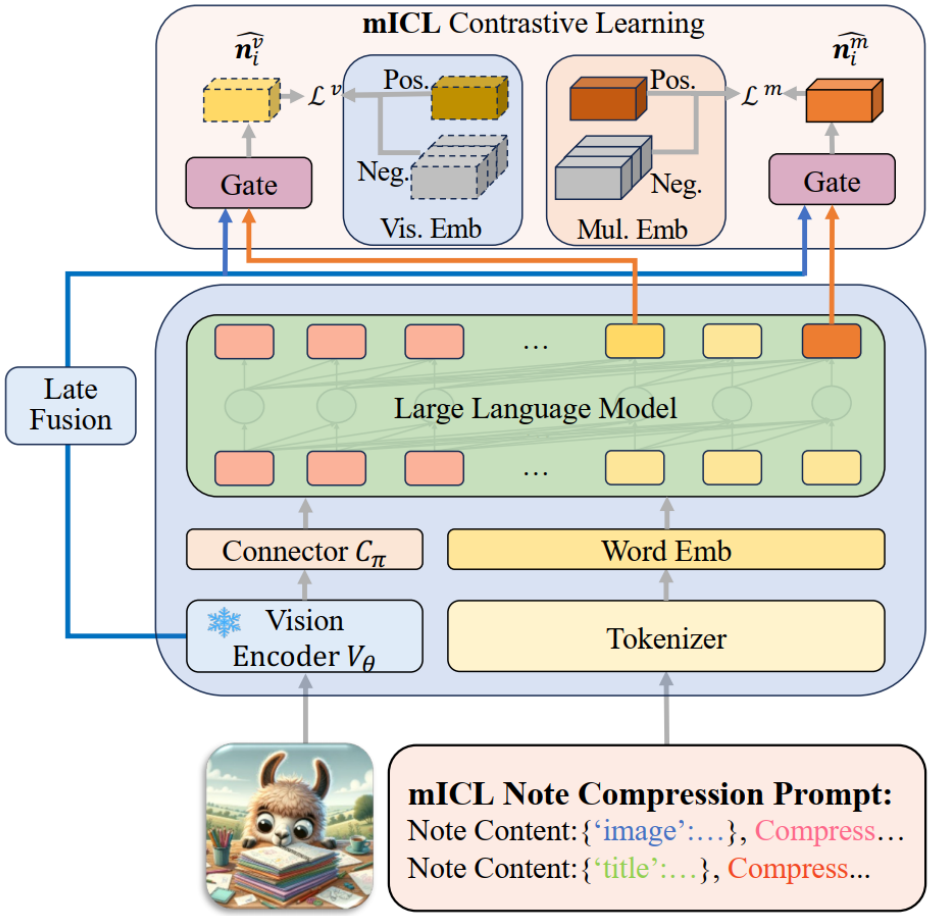
Large Language Model for MRS

➢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



NoteLLM-2: Multimodal Large Representation Models for Recommendation. KDD 2025.

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/1xFdx5xuNXHGsUVG2VIohFTXf9S7G5veq
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.

Future Directions

- **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

Conclusion

- **Challenges**:

  Raw feature representation, feature interaction, recommendation

- **Methodology**:

  Modality encoders, feature interaction, feature enhancement, model optimization

Survey

[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

Agenda

Introduction

Yejing Wang

Joint Modeling in RS

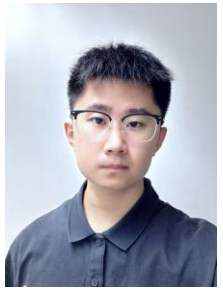
Pengyue Jia

Multi-task Recommendation

Yuhao Wang

Multi-scenario Recommendation

Pengyue Jia

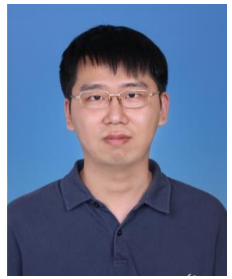


Xiaopeng Li

Multi-behavior Recommendation

Jingtong Gao

Multi-modal Recommendation



Qidong Liu

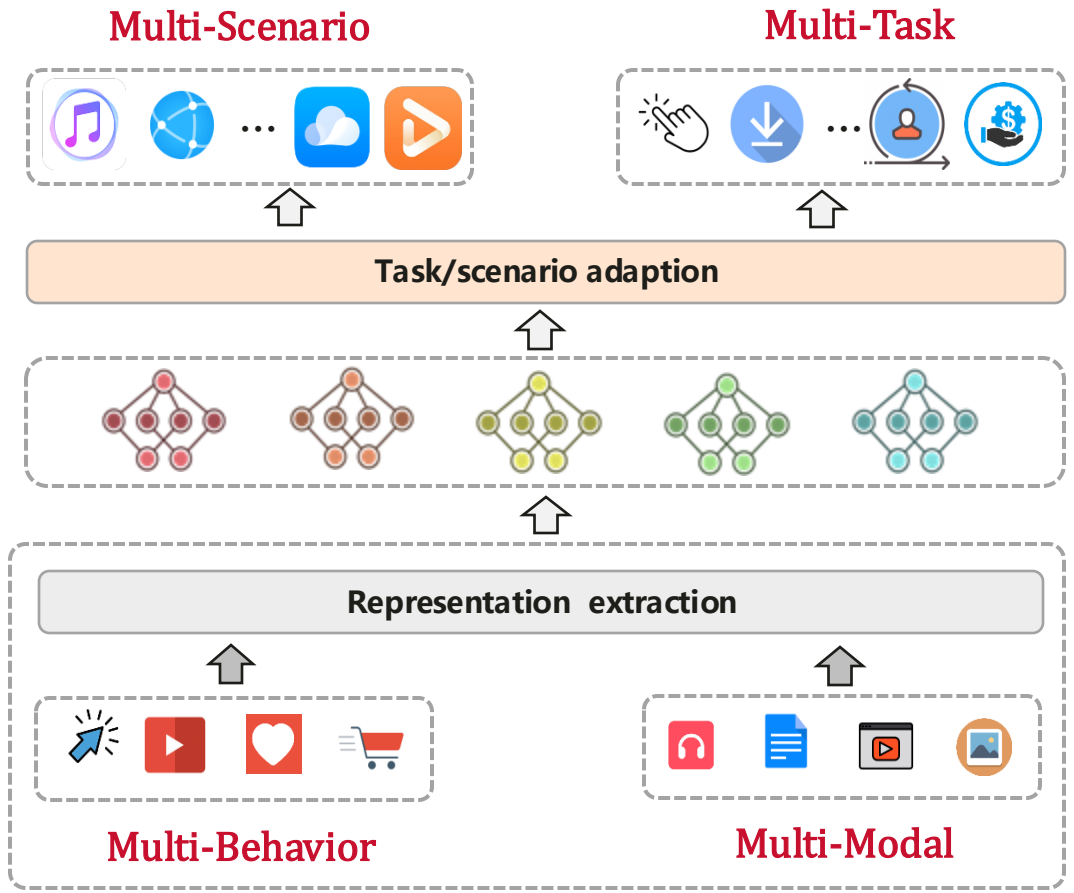
Conclusion

Future Work

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**



Summary

- **Multi-Task Recommendation**
  - Task relation:

    Parallel, Cascaded, Auxiliary with Main
  - Methodology:

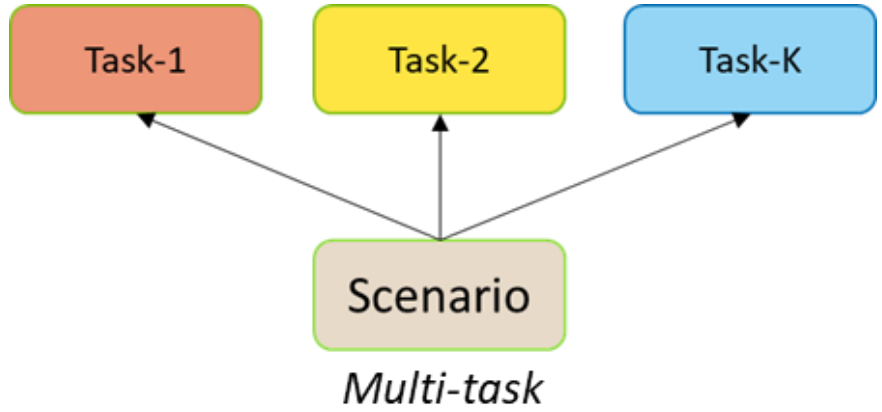
    Parameter Sharing, Optimization, Training Mechanism
  - Trends:

    Using LLM Modeling the similarities and differences across tasks with MoE-LoRA to enable task generalization.

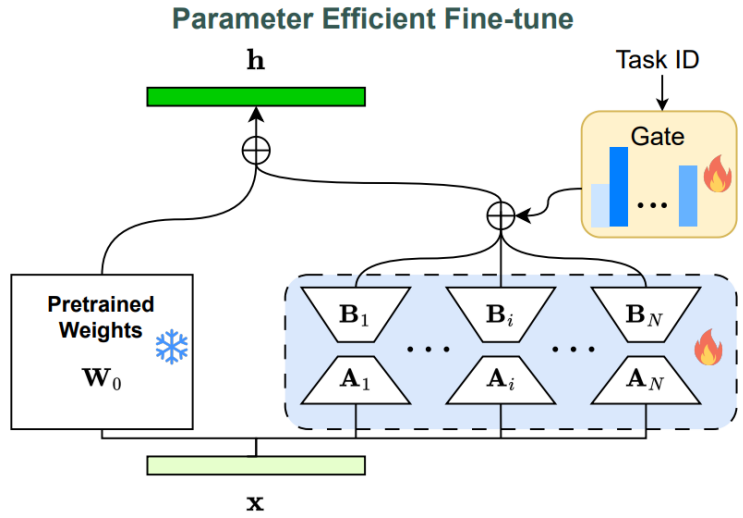
    e.g., MOELoRA
  - Future Direction:

    Mitigating negative transfer with LLM's world knowledge
  - Take Away:

    Designing prompt for different tasks and using LLM backbone as recommender systems



Multi-task



Summary

- **Multi-Scenario Recommendation**
  - Modeling methods:

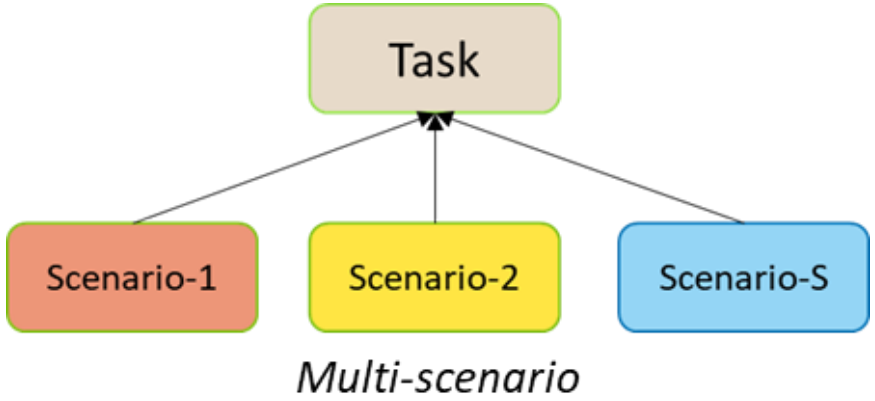
    shared-specific network paradigm, and dynamic weight paradigm.
  - Trends:

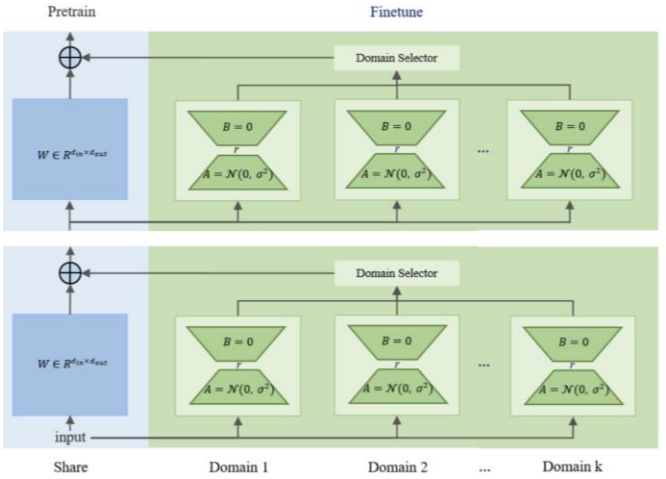
    Incorporating Low-Rank Adaptor (LoRA) for domain-specific fine-tuning (e.g., M-LoRA); Incorporating LLM's world knowledge in domain understanding (e.g., LLM4MSR)
  - Future Direction:

    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.





Summary

➢ **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)

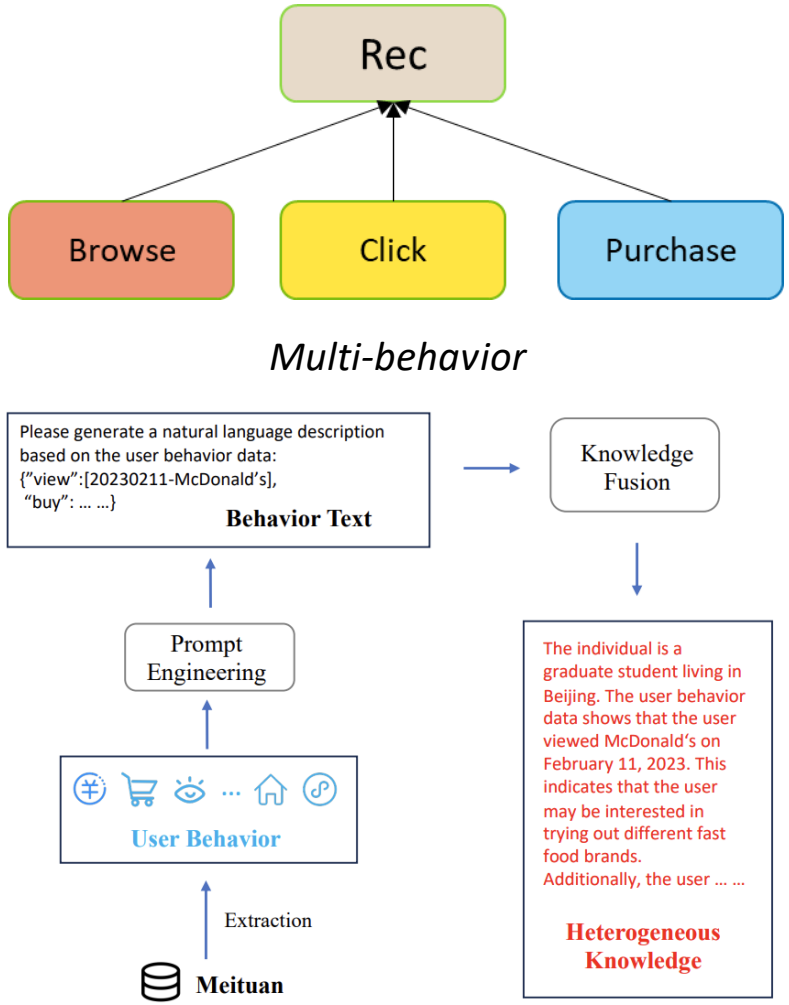
- Future Direction:

  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

Summary

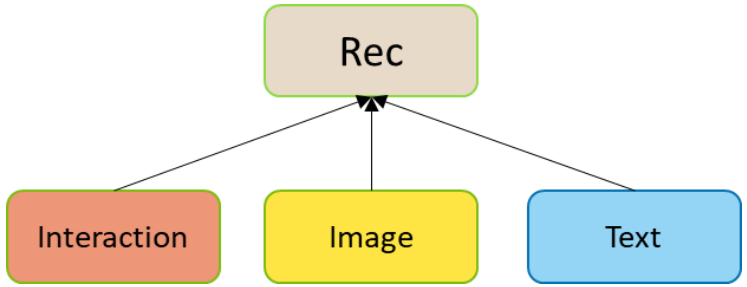
- **Multi-Modal Recommendation**
  - Methods:

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

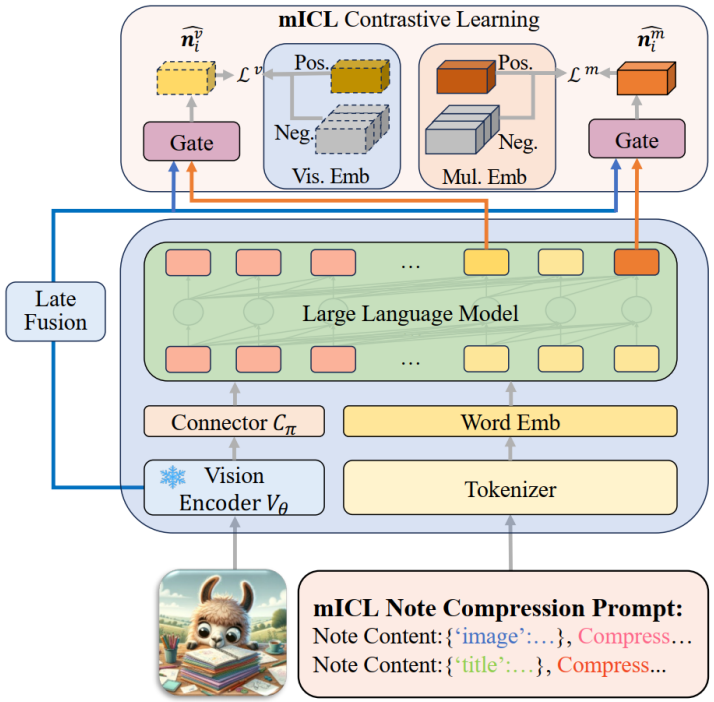
    Utilizing the powerful understanding abilities of MLLM to enhance Multi-Modal Rec (e.g., NoteLLM)
  - Future Direction:

    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.



Multi-modal



Q&A

We are hiring !

Huawei Noah's Ark Lab

WWW25 Huawei Noah's Ark Lab Chat Group

AML Lab
CityU

Tutorial Slides

https://zhaoxyai.github.io/paper/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