Regularized Adversarial Sampling and Deep Time-aware Attention for Click-Through Rate Prediction

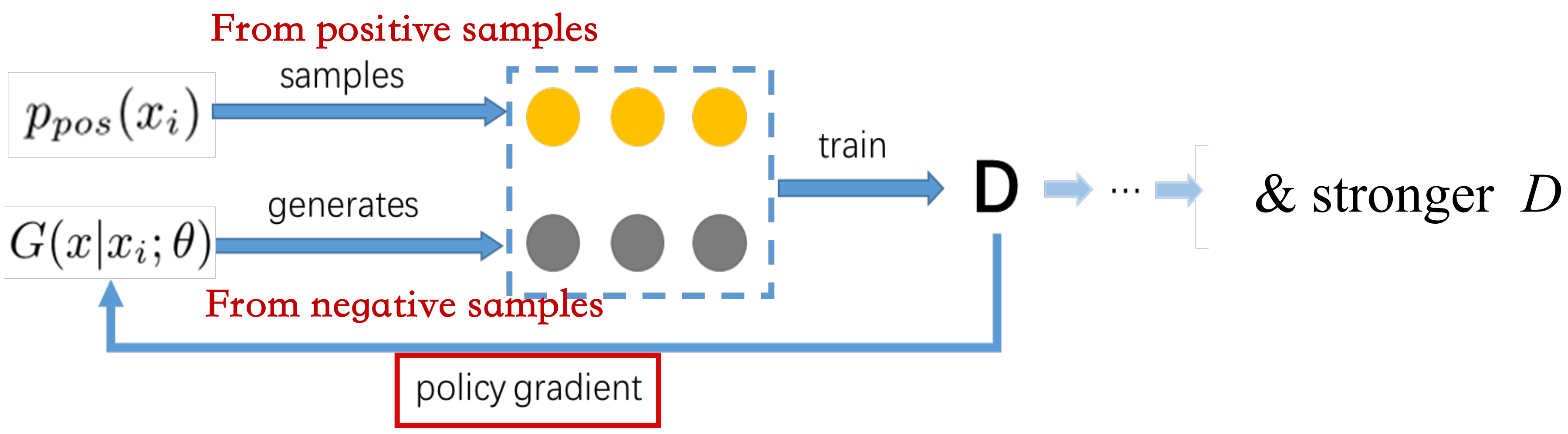
Yikai Wang*, Liang Zhang*, Quanyu Dai, Fuchun Sun,
Yang He, Bo Zhang, Weipeng Yan, and Yongjun Bao

Tsinghua University; JD.COM; The Hong Kong Polytechnic University

Regularized Adversarial Sampling

Background: Adversarial sampling via Policy Gradient

- Sampling with Generator： provides probabilities to candidate samples.
- As sampling probability cannot be updated via gradient BP, we use policy gradient (REINFORCE).



Regularized Adversarial Sampling

Existing adversarial sampling in recommender systems:

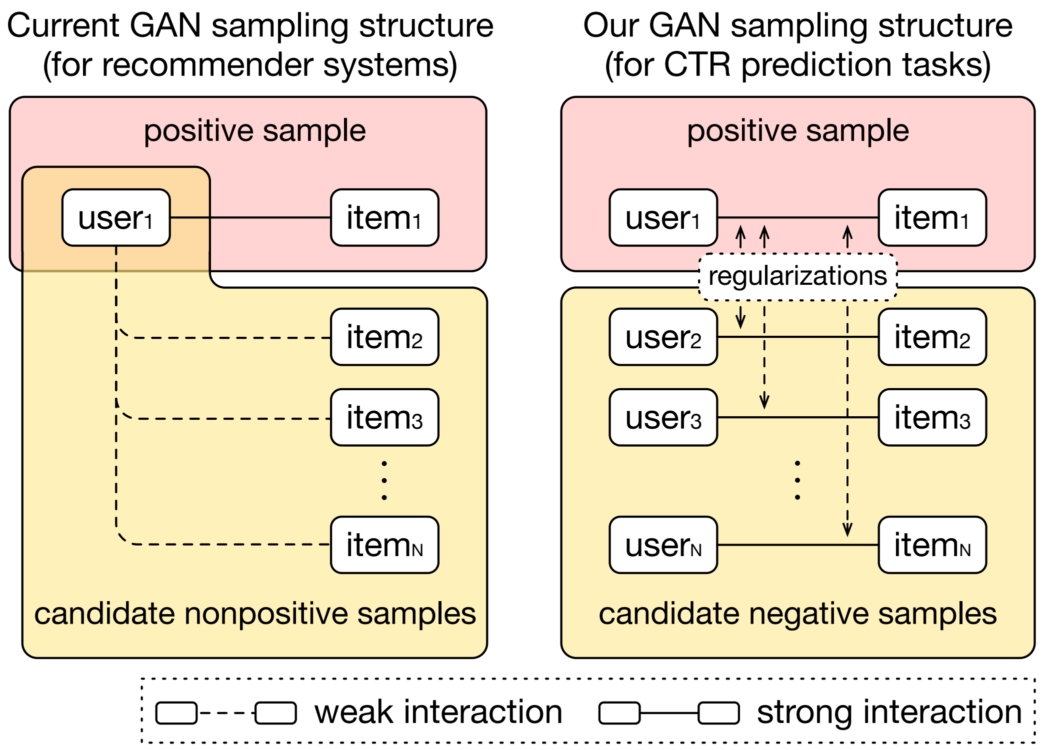
- Improves data efficiency by seeking competitive negative samples for pairwise training.
- Positive sample: a user-item pair, the user gives a like to the item.
- 'Negative' sample: a user-item pair, the user has not interacted with the item (random combination).

Cannot fit the CTR task:

- Because this sampling method cannot deal with real negative samples. Why?

Regularized Adversarial Sampling (rGAN):

- Common: Improves data efficiency by seeking competitive samples for pointwise or pairwise training.
- Key difference: Can use the strong information of the practical negative samples.



Regularized Adversarial Sampling (rGAN):

- Common: Improves data efficiency by seeking competitive samples for pointwise or pairwise training.
- Key difference: Can use the strong information of the practical negative samples.

rGAN indicates that the selected negative sample needs to be:

- Competitive among all the negatives.
- Correlative to the given positive sample.

Regularized Adversarial Sampling

loss function in the discriminator:

$$\mathcal{L}_D = \sum_{s \in \mathcal{T}} \left[- f_D\left(\boldsymbol{e}_D^s(s)\right) + f_D\left(\boldsymbol{e}_D^s(s')\right) + \gamma \right]_+, \; s' \sim p_G(s'|s)$$

Regularized Adversarial Sampling

loss function in the discriminator:

$$\mathcal{L}_D = \sum_{s \in \mathcal{T}} \left[- f_D\left(\boldsymbol{e}_D^s(s)\right) + f_D\left(\boldsymbol{e}_D^s(s')\right) + \gamma \right]_+, \ s' \sim p_G(s'|s)$$

The selected negative sample needs to be:

- Competitive among all the negatives, s' should have a high score, so as to be a strong impetus when training the discriminator.

Regularized Adversarial Sampling

loss function in the discriminator:

$$\mathcal{L}_D = \sum_{s \in \mathcal{T}} \left[- f_D\left(\boldsymbol{e}_D^s(s)\right) + f_D\left(\boldsymbol{e}_D^s(s')\right) + \gamma \right]_+, \ s' \sim p_G(s'|s)$$

The selected negative sample needs to be:

- Competitive among all the negatives, s' should have a high score, so as to be a strong impetus when training the discriminator.
- Correlative to the given positive sample: There should exist enough similarity between the embeddings of s and s'. We use Euclidean distances to restrict the embeddings calculated in the discriminator embedding space works better.

Regularized Adversarial Sampling

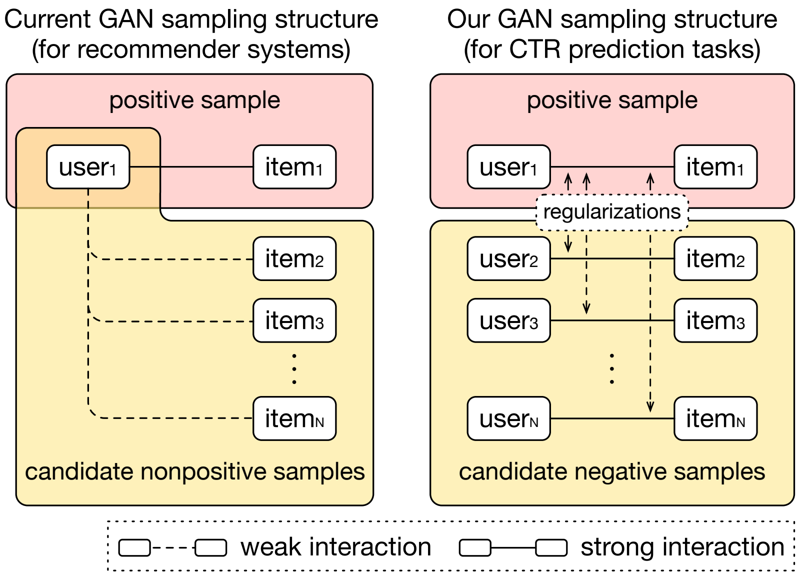
We use Euclidean distances as a penalty p(s,s'):

$$p(s,s') = \lambda_i \|\boldsymbol{e}_D^i(s) - \boldsymbol{e}_D^i(s')\|_2 + \lambda_h \|\boldsymbol{e}_D^h(s) - \boldsymbol{e}_D^h(s')\|_2$$

We design our generator based on the two properties: to maximize the expectation of scores of the selected negative samples with the penalties for correlative restriction.

$$\mathcal{L}_G = \sum_{s \in \mathcal{T}} \underbrace{\mathbb{E}_{s' \sim p_G(s'|s)} \left[f_D\left(\boldsymbol{e}_D^s(s')\right) - p(s,s') \right]}_{\text{denote as } \mathcal{J}_G(s)}$$

Update the gradient via policy gradient



Regularized Adversarial Sampling

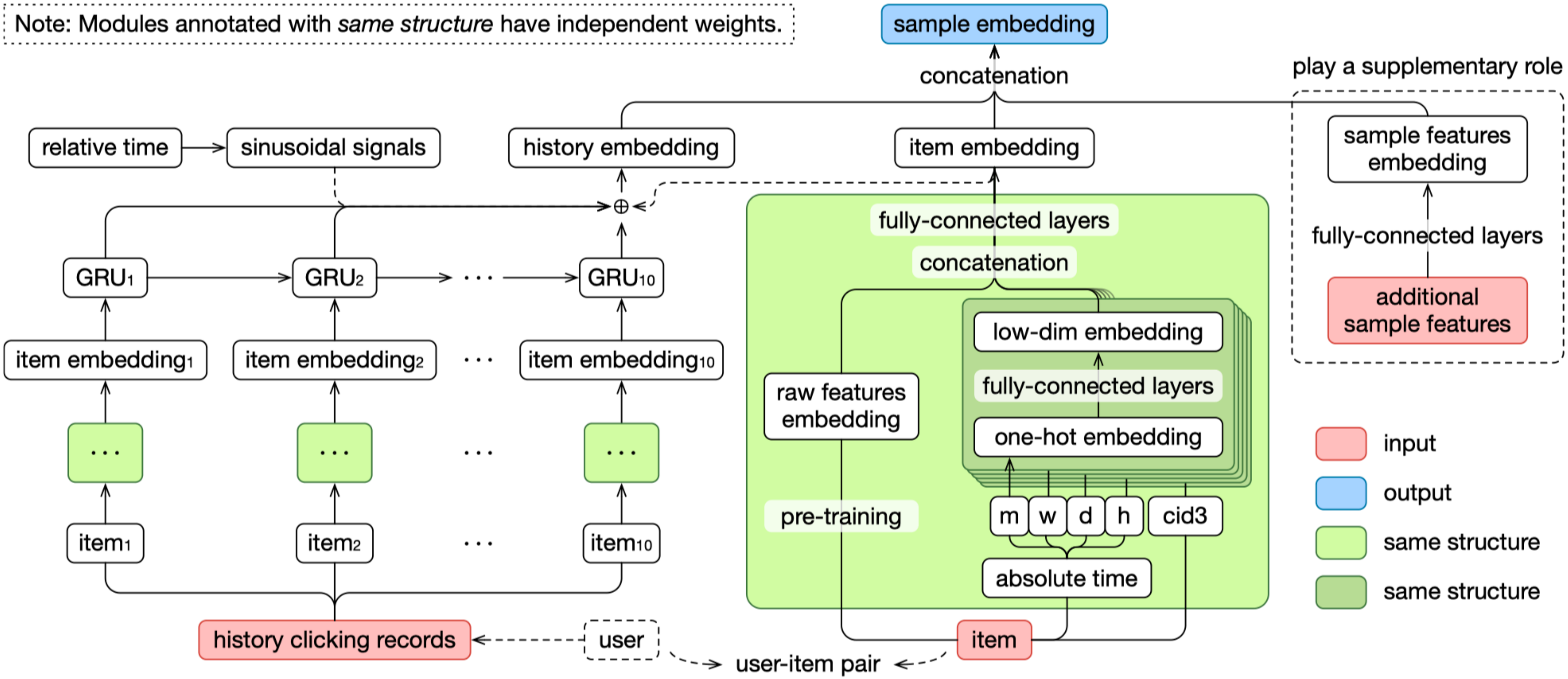
The sampling policy for the negative sample s' regarding to a positive sample s, is modeled as a union function of the generator sample embeddings:

$$p_G(s'|s) = softmax_{s' \in Neg(s)} \frac{\boldsymbol{e}_G^s(s')^\top \boldsymbol{e}_G^s(s)}{T\|\boldsymbol{e}_G^s(s')\|_2}$$

Time-aware Attention:

- Explicitly use the clicking temporal signals of users' historical data.
- Absolute time for periodicity representation.
- Relative time for temporal relation representation.

Time-aware Attention



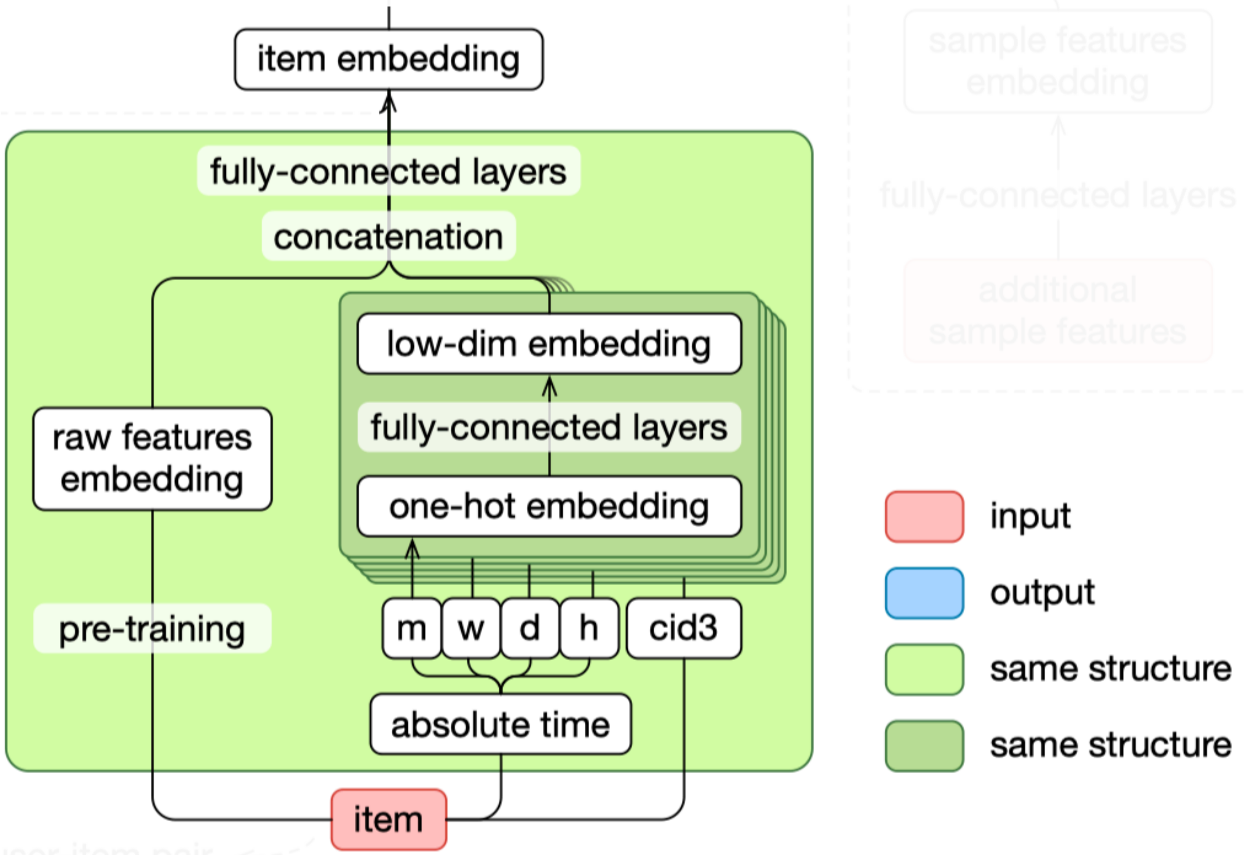
Regularized Adversarial Sampling and Deep Time-aware Attention for Click-Through Rate Prediction

Time-aware Attention

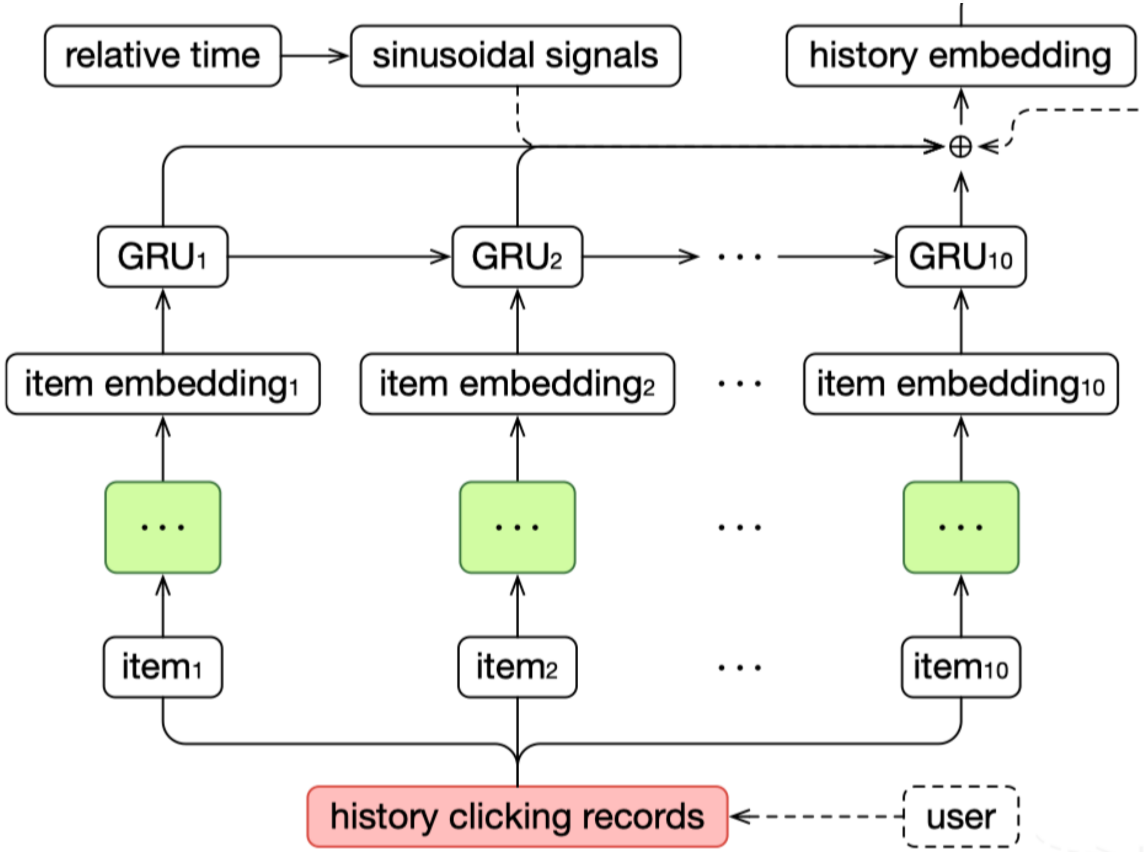
$$\boldsymbol{e}_0^c = \mathcal{R}(\boldsymbol{W}_c O(c_0))$$

$$\boldsymbol{e}_0^t = \mathcal{R}(concat(\boldsymbol{W}_m O(t_0^m), \boldsymbol{W}_w O(t_0^w), \boldsymbol{W}_d O(t_0^d), \boldsymbol{W}_h O(t_0^h)))$$

$$\boldsymbol{e}_0^i = \mathcal{F}(concat(\boldsymbol{e}_0^r, \boldsymbol{e}_0^c, \boldsymbol{e}_0^t))$$



Time-aware Attention

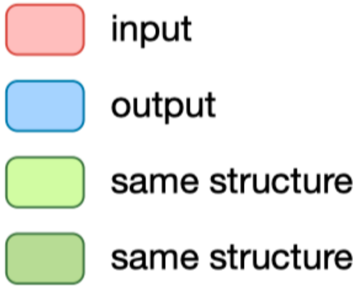


$$r_l = \sigma(W_{er} e_l^i + W_{hr} h_{l-1} + b_r)$$

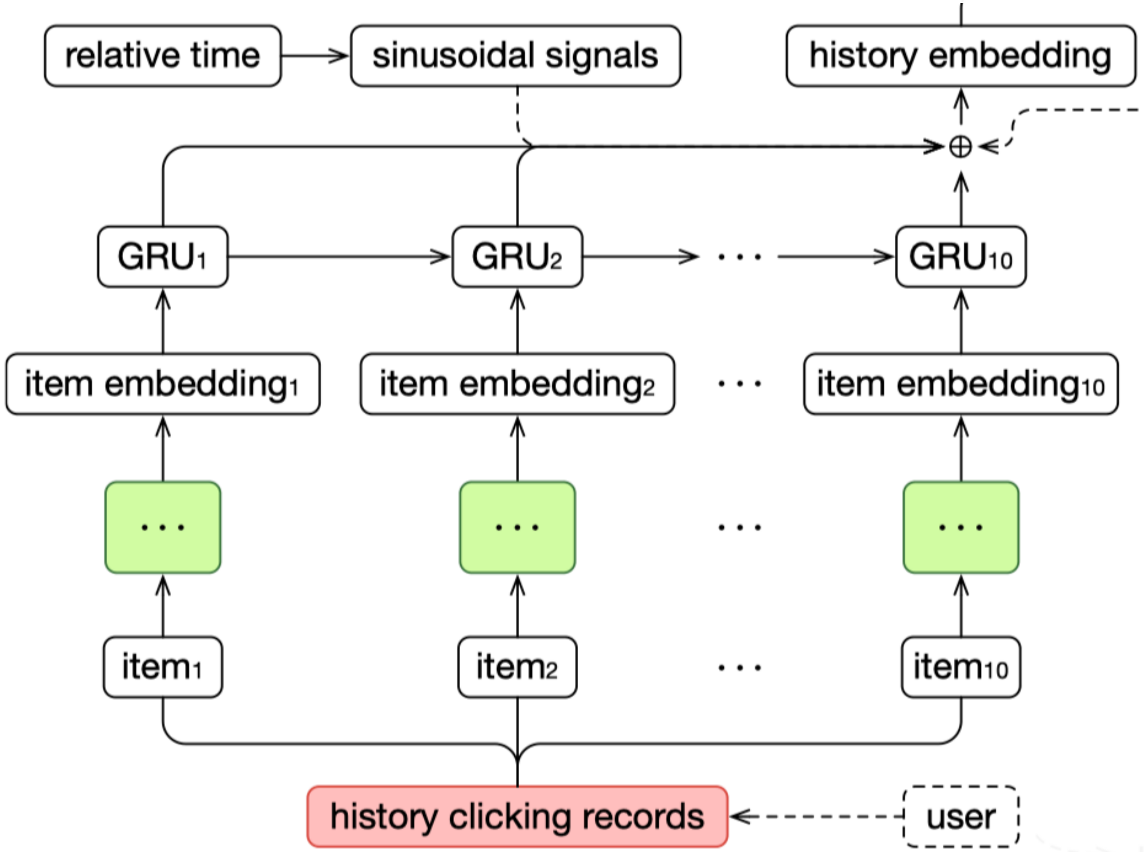
$$z_l = \sigma(W_{ez} e_l^i + W_{hz} h_{l-1} + b_z)$$

$$c_l = tanh\left(W_{ec} e_l^i + W_{hc}(r_l \odot h_{l-1}) + b_c\right)$$

$$h_l = (1 - z_l) \odot c_{l-1} + z_l \odot c_l$$



Time-aware Attention



$$e_l^t[2j] = sin\left((t_0^a - t_l^a)/10000^{2j/d}\right)$$

$$e_l^t[2j+1] = cos\left((t_0^a - t_l^a)/10000^{2j/d}\right)$$

$$u_l = \boldsymbol{v}^\top tanh(\boldsymbol{W}_h \boldsymbol{h}_l + \boldsymbol{W}_i \boldsymbol{e}_0^i + \boldsymbol{W}_t \boldsymbol{e}_l^t)$$

$$a_l = softmax(u_l)$$

$$\boldsymbol{e}^h = \sum_{l=1}^{L} a_l \boldsymbol{h}_l$$

input

output

same structure

same structure

Comparison Results

Performance of various sampling strategies

Model	in-station-Sep. AUC	in-station-Sep. RelaImpr	out-station-Jul. AUC	out-station-Jul. RelaImpr	out-station-May. AUC	out-station-May. RelaImpr
Logistic Regression	0.7643	0.15%	0.6790	-5.34%	0.6251	-7.81%
1:5 Under Sampling	0.7587	-1.97%	0.6818	-3.86%	0.6270	-6.41%
User-fixed Sampling	0.7589	-1.89%	0.6866	-1.32%	0.6379	1.62%
Uniform Sampling*	0.7639	0.00%	0.6891	0.00%	0.6357	0.00%
IRGAN Sampling	0.7366	-10.34%	0.6597	-15.55%	0.6165	-14.15%
IRGAN++ Sampling	0.7655	0.61%	0.6924	1.75%	0.6380	1.69%
rGAN Sampling	**0.7745**	**4.02%**	**0.7021**	**6.87%**	**0.6439**	**6.04%**

Regularized Adversarial Sampling

strength and weakness:

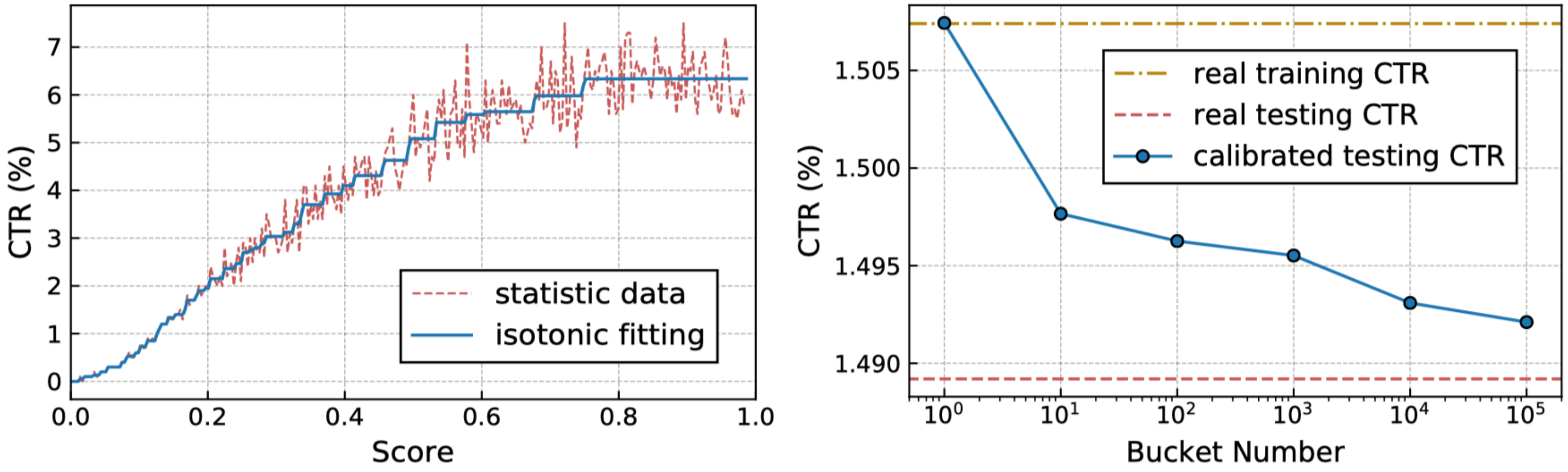
rGAN brings better relative CTR values —— is beneficial for ranking.

But the proportion of the positives and the negatives in the constructed training data will not match the real data proportion. Such mismatching will lead to an inaccurate absolute CTR estimates —— is bad for bidding.

Kuang-chih Lee, Burkay Orten, Ali Dasdan, and Wentong Li. 2012. Estimating conversion rate in display advertising from past erformance data.

Comparison Results

Calibration results



(a) $p(v_j)$ and its isotonic fitting (b) Calibrated results w.r.t. bucket numbers

Comparison Results

Performance of various embedding models

Model	in-station-Sep.		out-station-Jul.		out-station-May.	
	AUC	RelaImpr	AUC	RelaImpr	AUC	RelaImpr
Two-layer GRU	0.7350	-6.33%	0.6726	-6.70%	0.6271	-4.36%
DIN*	0.7509	0.00%	0.6850	0.00%	0.6329	0.00%
GRU Attention	0.7523	0.56%	0.6892	2.27%	0.6322	-0.53%
Time-aware Attention	**0.7745**	**9.41%**	**0.7021**	**9.24%**	**0.6439**	**8.28%**

Thanks for listening!

Yikai Wang*, Liang Zhang*, Quanyu Dai, Fuchun Sun,
Yang He, Bo Zhang, Weipeng Yan, and Yongjun Bao

Tsinghua University; JD.COM; The Hong Kong Polytechnic University