On Predicting Post-Click Conversion Rate via Counterfactual Inference

Junhyung Ahn[1], **Sanghack Lee**[2]

[1]Naver Corporation, [2]Seoul National University

Executive Summary

Problem

- Conversion labels exist only for clicked items → **Data Sparsity & Bias**

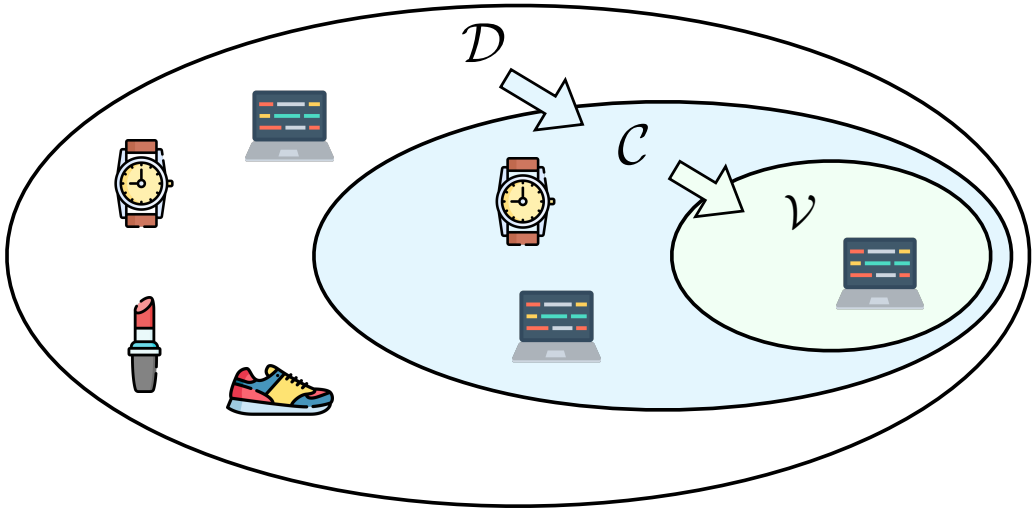
Solution: ESCIM (Entire Space Counterfactual Inference Multi-task Model)

- Generates counterfactual conversion labels for **non-clicked** samples
- Offline: +1% CVR AUC / Online: +17% CVR

Takeaway

- Turning unclicked data into useful training signals

Background



$\mathcal{D}$: Exposure (inference) space

$\mathcal{C}$: Click (training) space

$\mathcal{V}$: Conversion space

: Click-through rate (CTR)

: Conversion rate (CVR)

- Conversion are only recorded after a click
- Click ~ 4%, conversion ~ 0.5% → Very sparse
- Models trained only on clicked data → Selection bias

Related Work

Model	Key Idea	Limitation
ESMM (SIGIR'18)	Joint CTR & CVR prediction	Estimates CVR higher
ESCM^2 (SIGIR'22)	Adds IPS/DR CVR Loss	Relies only on clicked data
DCMT (ICDE'23)	Exploits counterfactual space	Labels all non-clicks = 1

Related Work

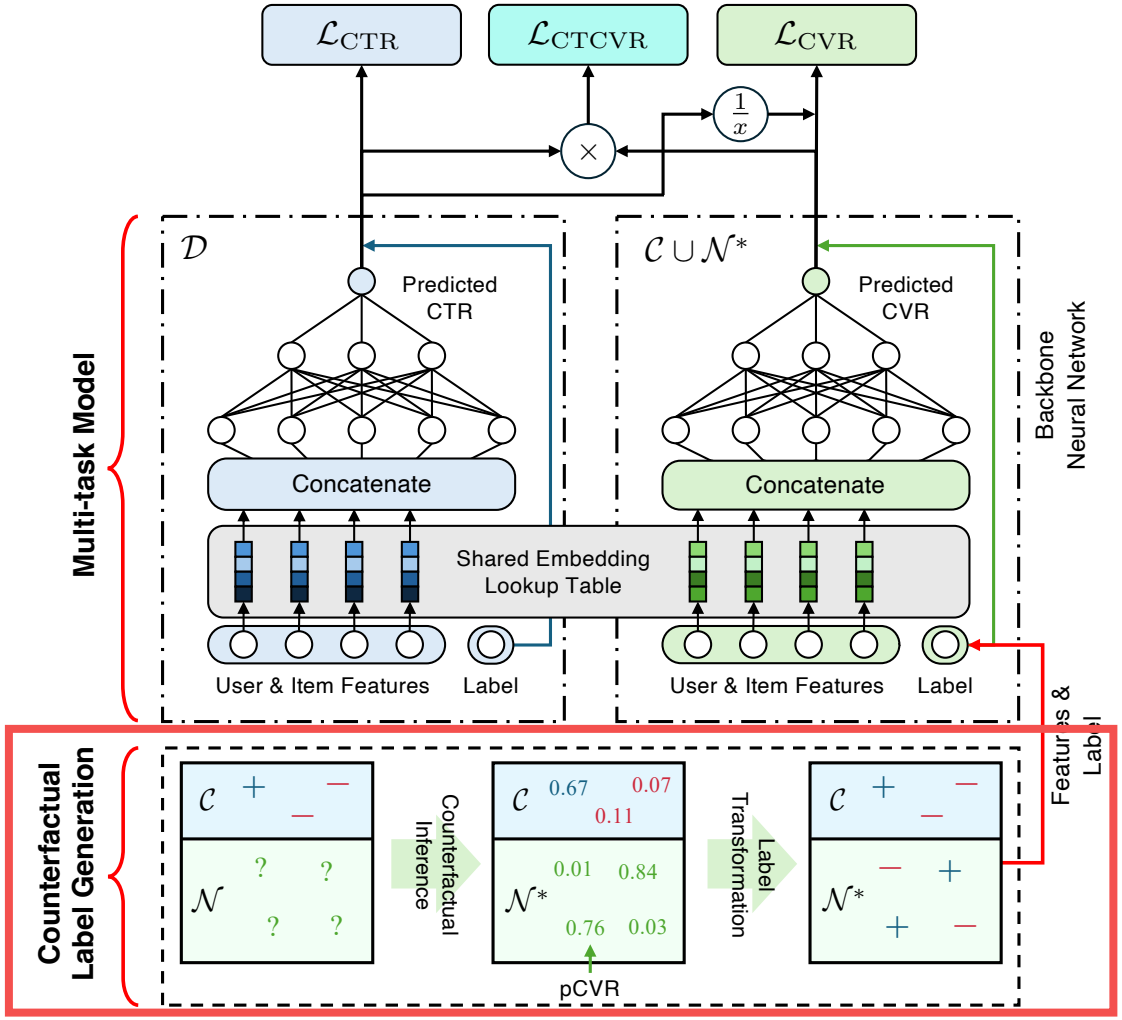
Model	Key Idea	Limitation
ESMM (SIGIR'18)	Joint CTR & CVR prediction	Estimates CVR higher
ESCM^2 (SIGIR'22)	Adds IPS/DR CVR Loss	Relies only on clicked data
DCMT (ICDE'23)	Exploits counterfactual space	Labels all non-clicks = 1

How to generate counterfactual labels precisely for non-clicks?

ESCIM (Entire Space Counterfactual Inference Multi-task Model)

Consists of Two Steps:

- 1. Counterfactual Inference
- 2. Label Transformation



Counterfactual Inference

- Primary Objective: Estimate a counterfactual probability

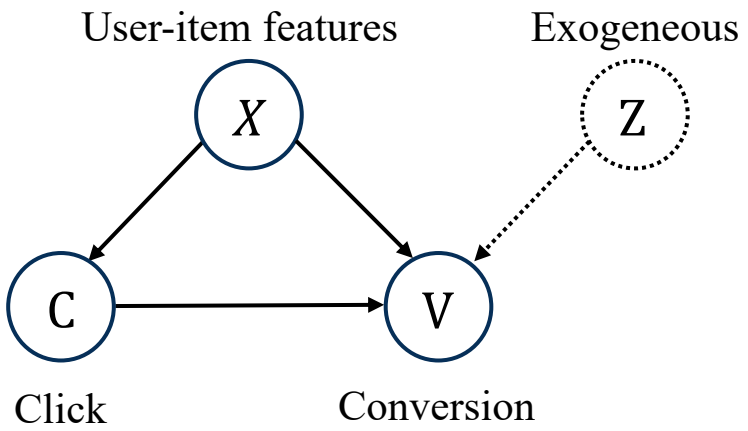
$$\mathbb{P}\left(V_{u,i}(C_{u,i}=1)=1 \mid X_{u,i}, C_{u,i}=0, V_{u,i}=0\right)$$

Potential outcome of conv. if clicked

Non-clicked

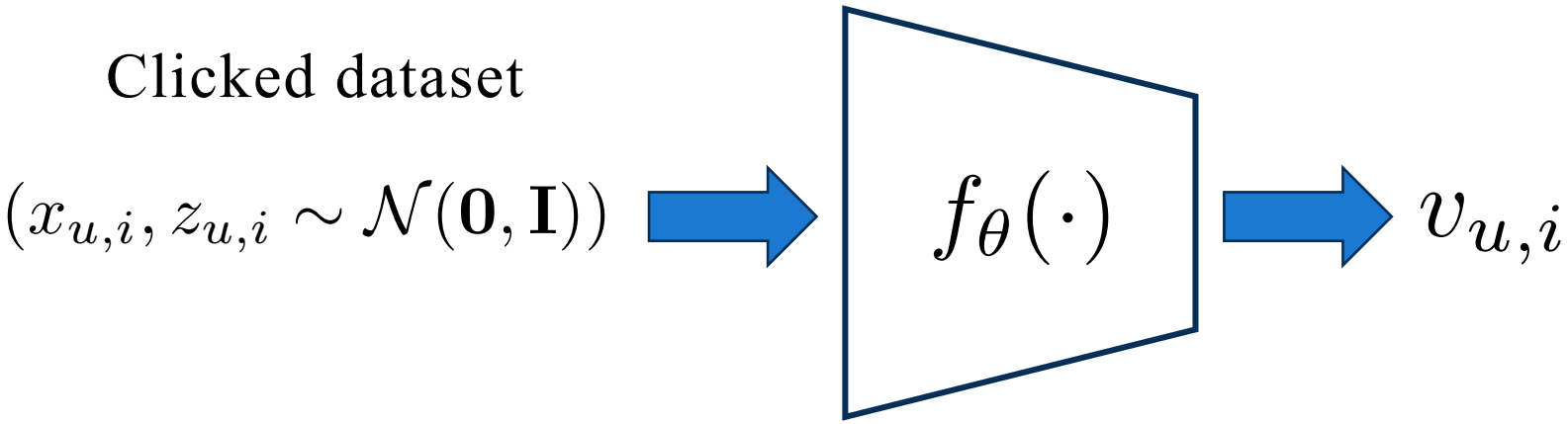
- Pearl's counterfactual framework: Abduction → Action → Prediction

- First, construct structural causal model (SCM)



Pre-training

- Prior to C.I., train conversion model f_θ w/ observational (clicked) data
- Will be used in the "Prediction" step later

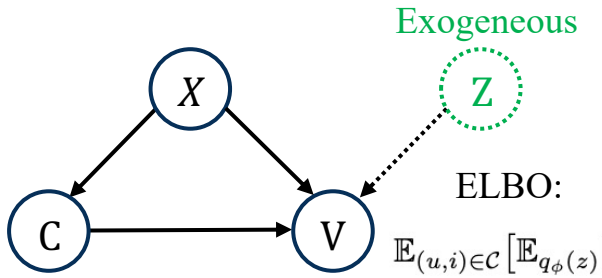


$$\mathcal{L}_{\text{pre-train}} = \mathbb{E}_{(u,i)\in\mathcal{C}} \left[\ell(f_\theta(x_{u,i}, z_{u,i}), v_{u,i})\right]$$

Abduction → Action → Prediction

1. Abduction

- Infer posterior exogeneous Z via VAE g_ϕ

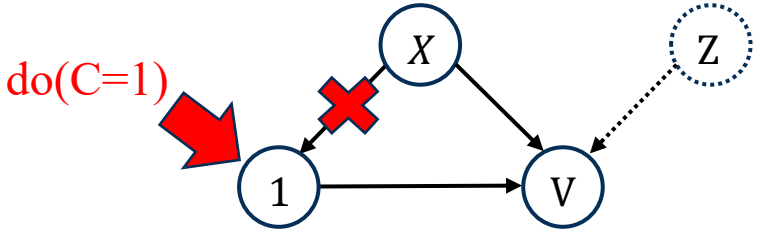


ELBO:

$$\mathbb{E}_{(u,i)\in\mathcal{C}}\left[\mathbb{E}_{q_\phi(z)}[p(z \mid x_{u,i}, v_{u,i})] - \beta \mathrm{KL}(p(z)||q_\phi(z))\right]$$

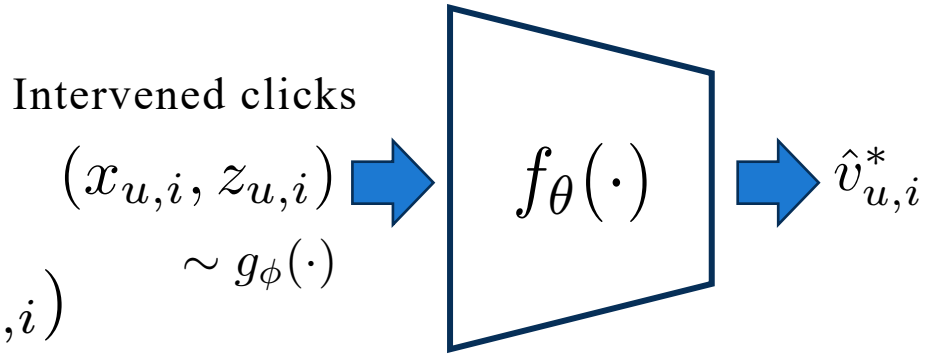
2. Action

- Intervention on click for non-clicks

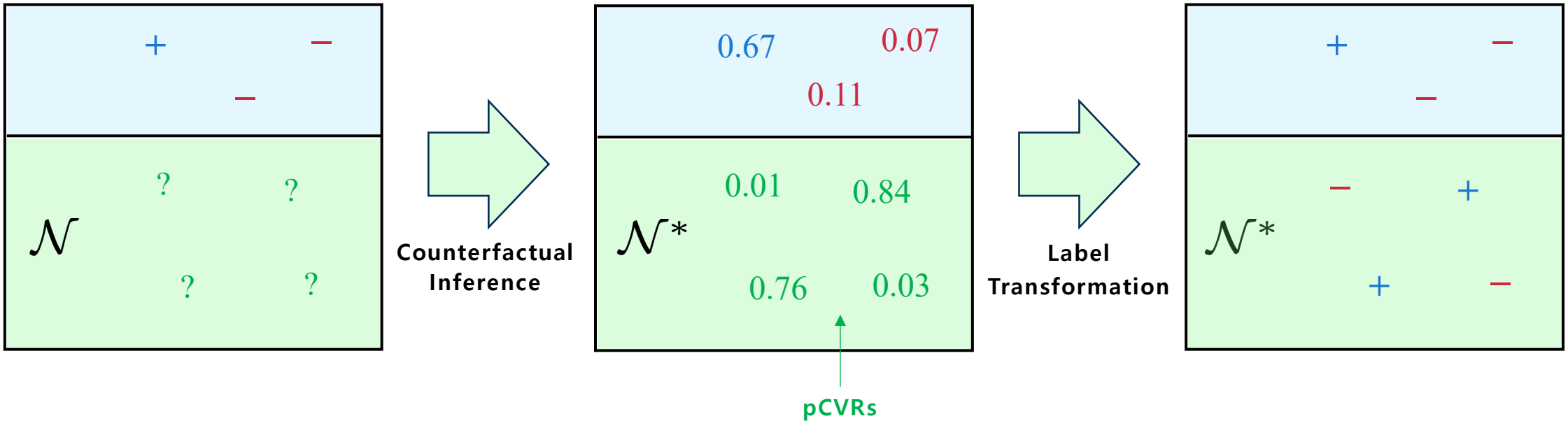


3. Prediction

- Sample z from encoder of g_ϕ
- For non-clicked, infer pCVR $\hat{v}^*_{u,i} = f_\theta(x_{u,i}, z_{u,i})$



Next Step: Label Transformation



Q: How to use these predicted values in training?

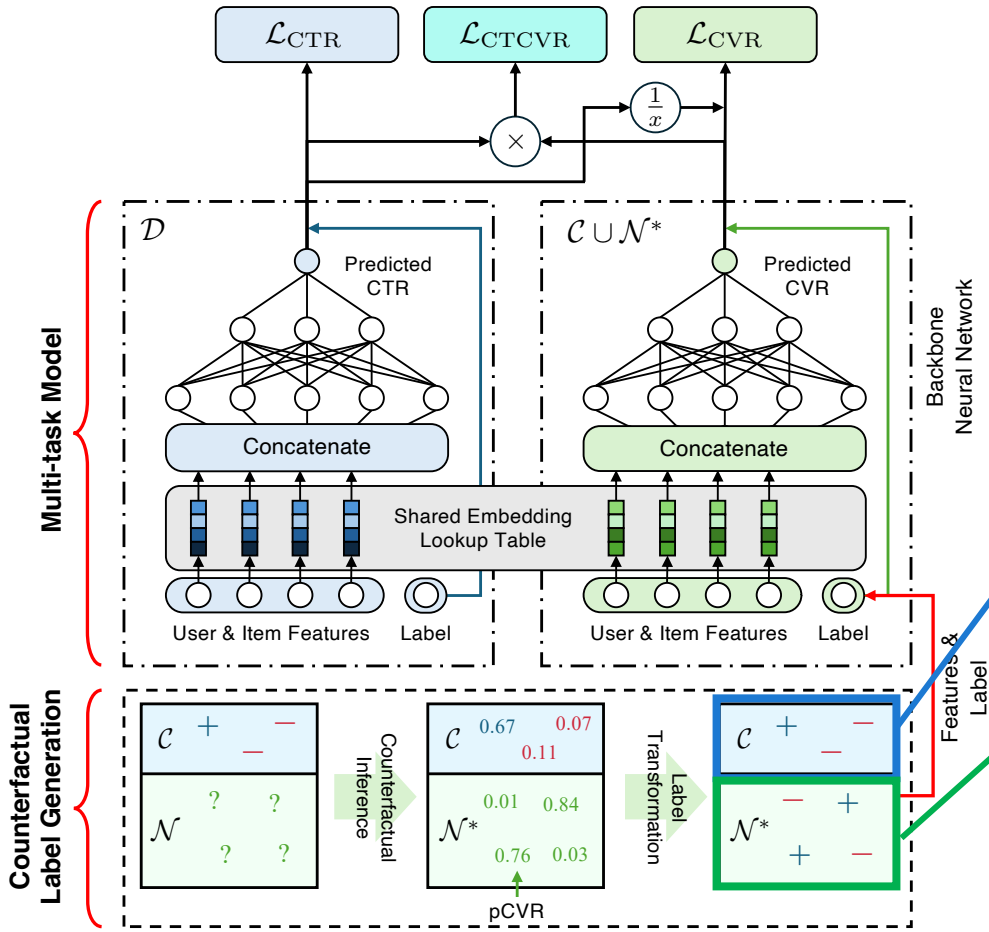
➔ Transform these values into binary labels

Label Transformation

Goal: convert pCVR to binary label

Type	Rule	Intuition
Max	Label 1 if $pCVR \geq max(pCVR_clicked)$	Strict threshold
Ratio	Label 1 to top-k that matches conv. ratio of clicked data	Balanced label distribution

Objective



$$\mathcal{L} = \mathcal{L}_{CTR} + \mathcal{L}_{CTCVR} + \alpha_F \mathcal{L}_{CVR-F} + \alpha_{CF} \mathcal{L}_{CVR-CF}$$

$$\mathcal{L}_{CVR-F} = \mathbb{E}_{(u,i)\in\mathcal{C}} \left[\frac{\ell(\hat{v}_{u,i}, v_{u,i})}{\hat{c}_{u,i}} \right]$$

$$\mathcal{L}_{CVR-CF} = \mathbb{E}_{(u,i)\in\mathcal{N}^*} \left[\frac{\ell(\hat{v}_{u,i}, v^*_{u,i})}{\hat{c}_{u,i}} \right]$$

Experiments

Offline Test

(CVR AUC) / (CTCVR AUC)

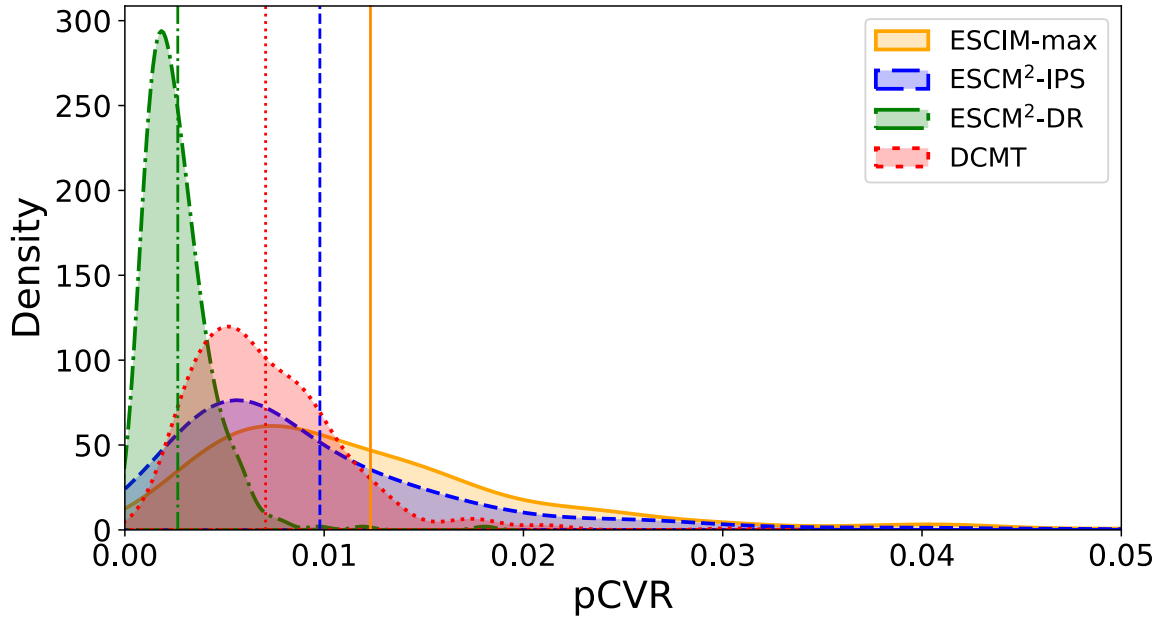
Method	Ali-CCP across backbones				Ali-Express w/ MLP across countries			
	MLP	DeepFM	AutoInt	DCN-V2	AE-ES	AE-FR	AE-NL	AE-US
ESMM	0.6474 / 0.6379	0.6398 / 0.6213	0.6543 / 0.6302	0.6495 / 0.6373	0.8099 / 0.8717	0.7949 / 0.8500	0.7795 / 0.8479	0.7952 / 0.8425
Multi-IPS	0.6523 / 0.6390	0.6431 / 0.6244	0.6587 / 0.6357	0.6535 / 0.6379	0.8205 / 0.8592	0.7947 / 0.8504	0.7829 / 0.8377	0.8068 / 0.8433
Multi-DR	0.6437 / 0.6305	0.6388 / 0.6199	0.6542 / 0.6304	0.6488 / 0.6353	0.8135 / 0.8600	0.7943 / 0.8538	0.7803 / 0.8292	0.7956 / 0.8396
ESCM^2-IPS	0.6691 / 0.6424	0.6487 / 0.6281	0.6681 / 0.6407	0.6609 / 0.6374	0.8199 / 0.8777	0.8078 / 0.8511	0.7867 / 0.8511	0.8018 / 0.8573
ESCM^2-DR	0.6609 / 0.6353	0.6502 / 0.6280	0.6485 / 0.6329	0.6489 / 0.6313	0.8112 / 0.8739	0.8073 / 0.8535	0.7772 / 0.8473	0.7917 / 0.8404
DCMT	0.6743 / 0.6451	0.6528 / 0.6338	0.6702 / 0.6389	0.6603 / 0.6350	0.8251 / 0.8838	0.8089 / 0.8628	0.7897 / 0.8525	0.8140 / 0.8620
ESCIM-max	**0.6792** / 0.6487	**0.6585** / **0.6401**	0.6737 / 0.6413	**0.6698** / 0.6467	**0.8314** / 0.8935	**0.8210** / 0.8776	**0.7951** / **0.8613**	**0.8332** / **0.8782**
ESCIM-ratio	0.6756 / **0.6566**	0.6523 / 0.6382	**0.6774** / **0.6475**	0.6683 / **0.6534**	0.8305 / **0.8951**	0.8192 / **0.8821**	0.7907 / 0.8583	0.8201 / 0.8643

Online A/B Test

Metric	Day 1	Day 2	Day 3	Day 4	Day 5
CVR (↑)	+21.55%	+16.94%	+12.59%	+17.42%	+18.24%
CTCVR (↑)	+7.41%	+4.76%	+3.98%	+5.56%	+6.31%
CPA (↓)	-27.16%	-26.67%	-16.05%	-22.53%	-24.38%

Performance on Latent Conversion Users

Def: users who never clicked during training but click in test



Model	CVR AUC
ESCM²-IPS	0.6724
ESCM²-DR	0.6627
DCMR	0.6783
ESCIM-max	**0.6905**

➔ **More robust to unseen user behaviors**

Conclusion

- **ESCIM = Counterfactual Inference + Multi-task Learning**
- **Utilizes non-click data → Better generalization**
- **Proven offline and online effectiveness**

Thank you!