Learning Neural Set Functions Under the Optimal Subset Oracle

(Selected as Oral)

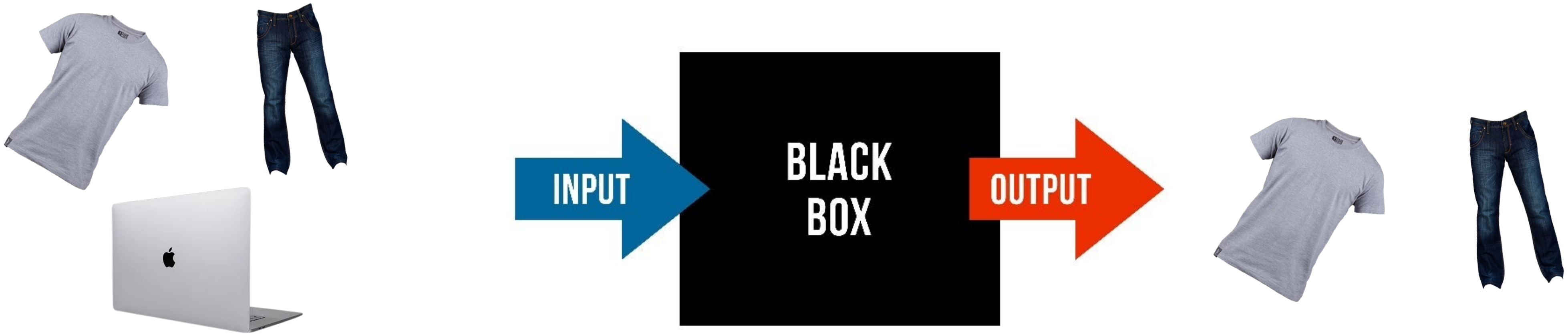
Zijing Ou[1], Tingyang Xu[2], Qinliang Su[3], Yingzhen Li[1], Peilin Zhao[2], Yatao Bian[2]
[1]Imperial College London
[2]Tencent AI Lab
[3]Sun Yat—sen University

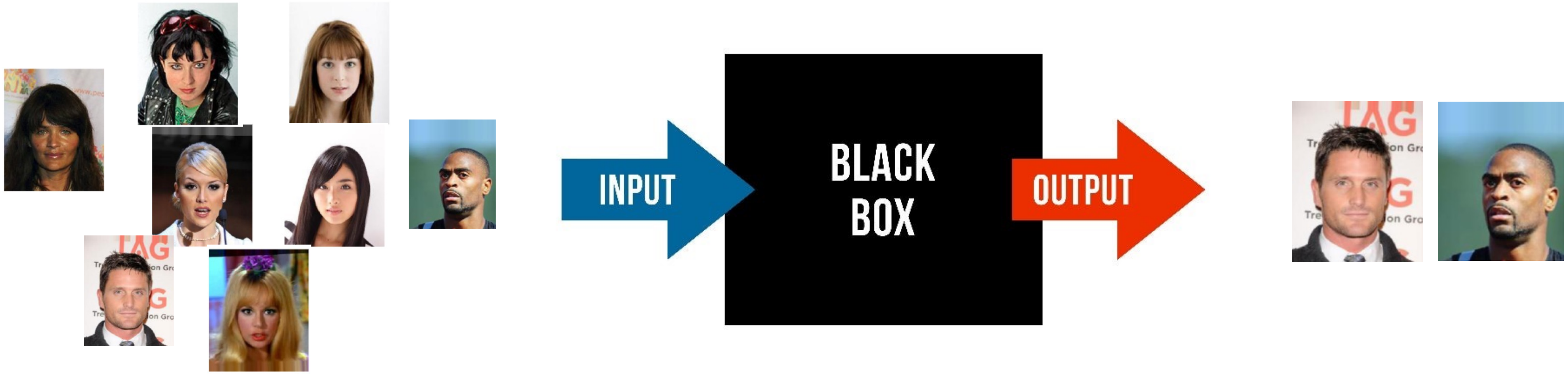
Background of Set Function Learning

Product Recommendation

Anomaly Detection

Background of Set Function Learning





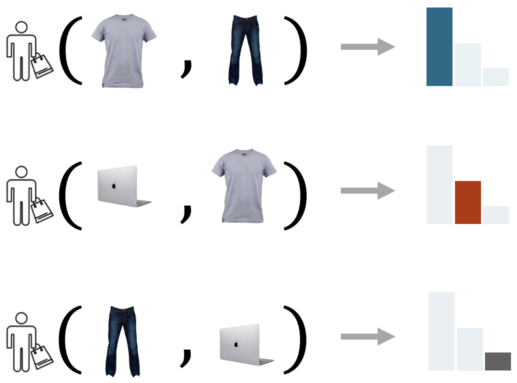
Background of Set Function Learning

NEURAL INFORMATION PROCESSING SYSTEMS

Database

Customer

Shopping cart



Ground set V

Utility function
$F_{\theta^*}(\cdot; V): 2^V \to \mathbb{R}$

Optimal subset S^*

Data Generation Process:

$$S^* = \underset{S \in 2^V}{\operatorname{argmax}} F_{\theta^*}(S; V)$$

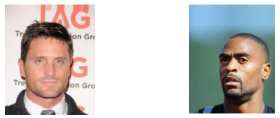
$\sim \mathbb{p}(S, V) =: \delta_{S=S^*|V}$

Background of Set Function Learning

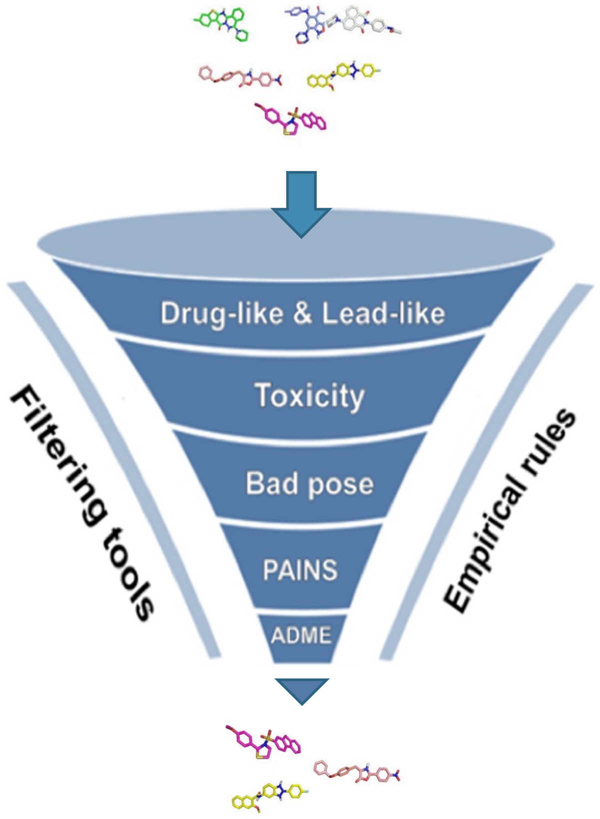
$$S^* = \underset{S \in 2^V}{\operatorname{argmax}} \, F_{\theta^*}(S; V)$$

Set anomaly detection

$$F_{\theta^*}(S; V) = \sum_{s_i \in S} \mathbb{I}(s_i \ is \ male) - \mathbb{I}(s_i \ is \ female)$$

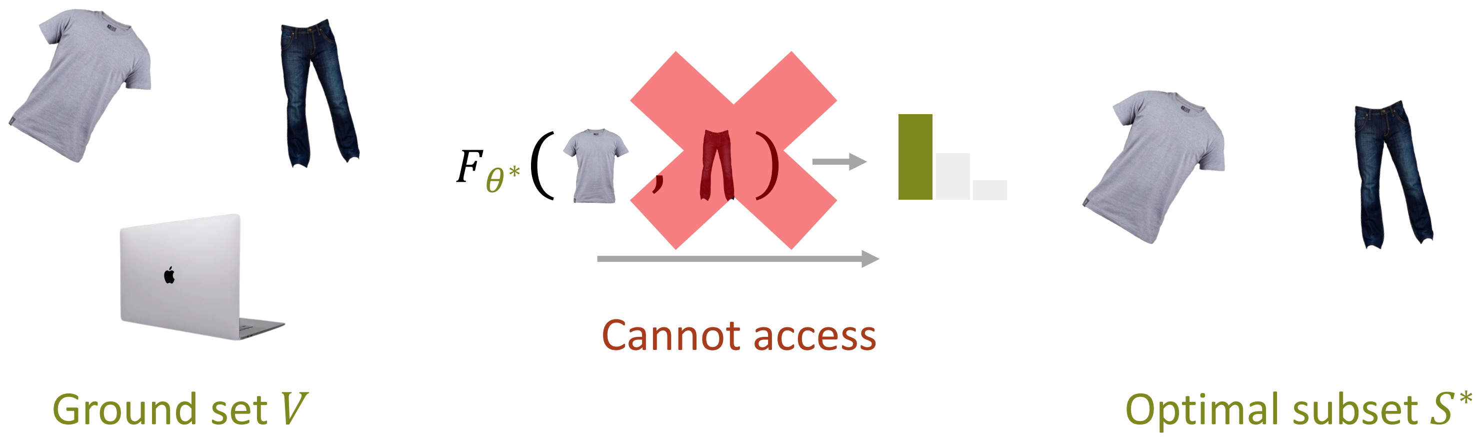


Compound selection



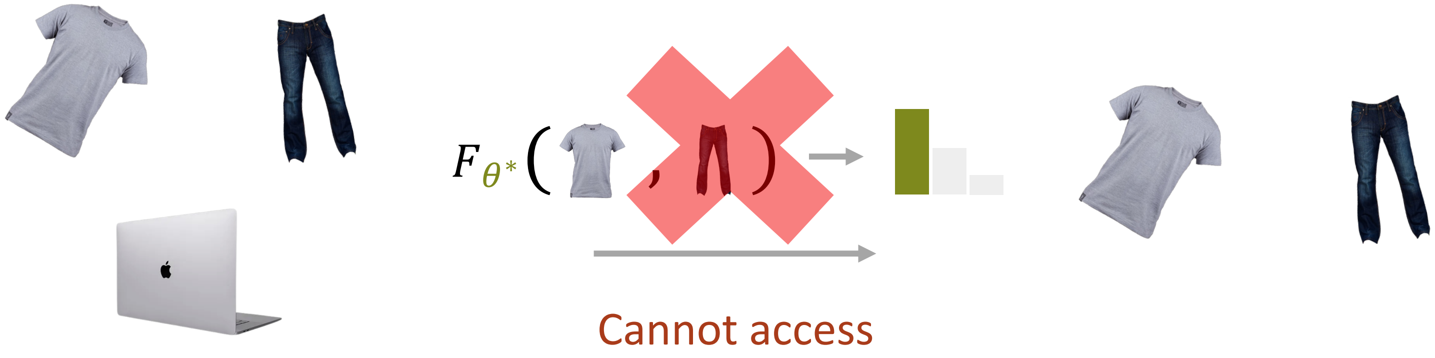
Background of Set Function Learning

Setting: Training data is given in form of $\{(V_i, S_i^*)\}_{i=1}^{N}$



Background of Set Function Learning

Setting: Training data is given in form of $\{(V_i, S_i^*)\}_{i=1}^N$



Ground set V

Optimal subset S^*

Goal: Learn a surrogate F_θ to approximate the oracle utility function F_{θ^*}.

$$S^* = \underset{S \in 2^V}{\operatorname{argmax}} F_\theta(S; V), \forall (V, S^*) \in \{(V_i, S_i^*)\}_{i=1}^N$$

Learning Set Function Under the OS Oracle

Maximum Likelihood:

Empirical distribution

$$\underset{\theta}{\operatorname{argmax}}\, \mathbb{E}_{\mathbb{P}(S^*,V)}[\log p_\theta(S^*|V)]$$

$$s.t.\ p_\theta(S|V) \propto F_\theta(S;V), \forall S \in 2^V$$

Monotonically grows with the utility function

Learning Set Function Under the OS Oracle MLE

Maximum Likelihood:

Empirical distribution

$$\operatorname*{argmax}_{\theta} \mathbb{E}_{\mathbb{P}(S^*,V)}[\log p_\theta(S^*|V)]$$

$$s.t.\ p_\theta(S|V) \propto F_\theta(S;V), \forall S \in 2^V$$

Monotonically grows with the utility function

How to construct a proper set mass function $p_\theta(S|V)$?

Learning Set Function Under the OS Oracle MLE

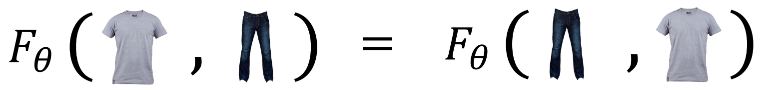
Maximum Likelihood:

$$\underset{\theta}{\operatorname{argmax}}\ \mathbb{E}_{\mathbb{P}(S^*,V)}[\log p_\theta(S^*|V)]$$

$$s.t.\ p_\theta(S|V) \propto F_\theta(S;V), \forall S \in 2^V$$

Desiderata:

Permutation invariance



Varying ground set

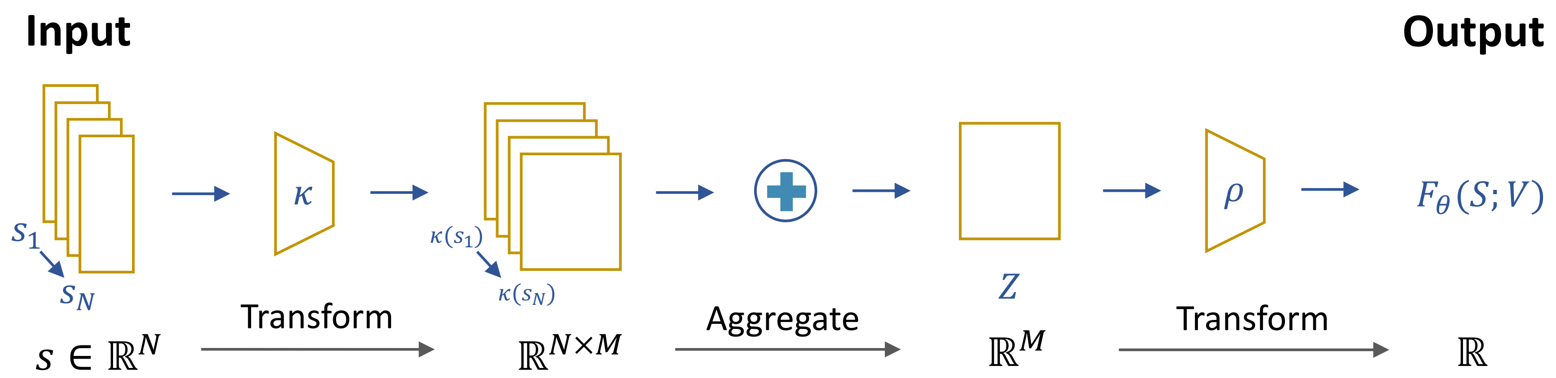
……

Energy-based Modeling for Set

$$p_\theta(S|V) = \frac{\exp(F_\theta(S;V))}{Z}$$

Z ← Partition function $Z := \sum_{S \subseteq V} \exp(F_\theta(S;V))$

DeepSet for Permutation Invariance & Varying Ground Set:



Zaheer Kottur, Ravanbakhsh, Poczos, Salakhutdinov and Smola. Deep sets. NIPS 2017.

Training Discrete EBMs

NEURAL INFORMATION PROCESSING SYSTEMS

Marginal-based Loss:

ψ^* is differentiable w.r.t. θ

$q(S;\psi) = \prod_{i \in S} \psi_i \prod_{j \notin S} (1 - \psi_j), \psi \in [0,1]^{|V|}$

$$\psi^{*|\theta} = \underset{\psi}{\operatorname{argmin}}\, \mathbb{KL}(q_\phi(S;\psi) \| p_\theta(S))$$

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{N} [-\sum_{j \in S_i^*} \log \psi_j^{*|\theta} - \sum_{j \in V_i \setminus S_i^*} \log(1 - \psi_j^{*|\theta})]$$

Training θ by differentiating through ψ^* using cross entropy loss

Justin Domke. Learning graphical model parameters with approximate marginal inference. PAMI 2013.

Training Discrete EBMs

Marginal-based Loss:

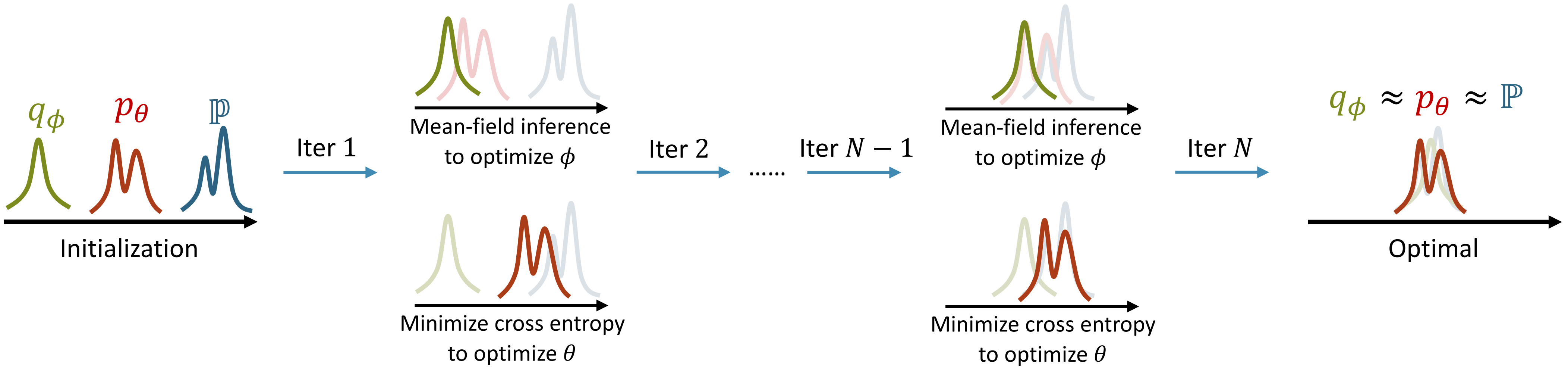
ψ^* is differentiable w.r.t. θ

$q(S;\psi) = \prod_{i \in S} \psi_i \prod_{j \notin S} (1 - \psi_j), \psi \in [0,1]^{|V|}$

$$\psi^{*|\theta} = \underset{\psi}{\operatorname{argmin}} \, \mathbb{KL}(q_\phi(S;\psi) \| p_\theta(S))$$

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{N} \Big[-\sum_{j \in S_i^*} \log \psi_j^{*|\theta} - \sum_{j \in V_i \setminus S_i^*} \log(1 - \psi_j^{*|\theta}) \Big]$$

Training θ by differentiating through ψ^* using cross entropy loss

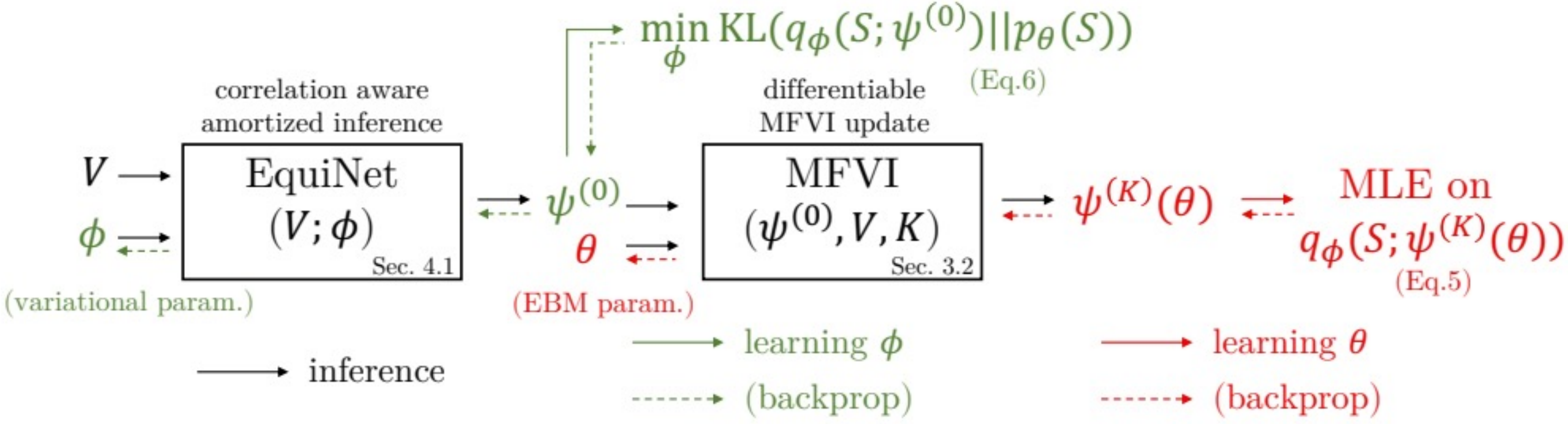


EquiVSet Overview

$$\psi^{*|\theta} = \underset{\psi}{\operatorname{argmin}}\, \mathbb{KL}(q(S;\psi)\|p_\theta(S))$$

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \textstyle\sum_{i=1}^{N}[-\sum_{j\in S_i^*} \log \psi_j^{*|\theta} - \sum_{j\in V_i \backslash S_i^*} \log(1-\psi_j^{*|\theta})]$$

Equivariant **V**ariational inference for **Set** function learning (**EquiVSet**)

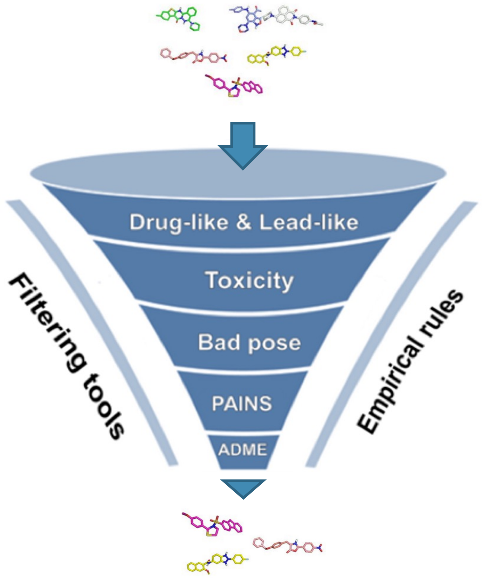


Applications

EquiVSet achieves the **BEST** results on 3 different tasks!

Product Recommendation

Set Anomaly Detection



Compound Selection

Applications: product recommendation

Baselines:

- Random: random pick
- PGM: probabilistic greedy models
- DeepSet (NoSetFn): only the q net

Metric:

$$\mathrm{MJC} := \frac{1}{|\mathcal{D}_t|} \sum_{(V,S^*) \in \mathcal{D}_t} \frac{|S^* \cap S|}{|S^* \cup S|}$$

Table 2: Product recommendation results on the Amazon dataset with different categories.

Categories	Random	PGM	DeepSet (NoSetFn)	DiffMF (ours)	EquiVSet$_{\text{ind}}$ (ours)	EquiVSet$_{\text{copula}}$ (ours)
Toys	0.083	0.441 ± 0.004	0.429 ± 0.005	0.610 ± 0.010	0.650 ± 0.015	**0.680 ± 0.020**
Furniture	0.065	0.175 ± 0.007	**0.176 ± 0.007**	0.170 ± 0.010	0.170 ± 0.011	0.172 ± 0.009
Gear	0.077	0.471 ± 0.004	0.381 ± 0.002	0.560 ± 0.020	0.610 ± 0.020	**0.700 ± 0.020**
Carseats	0.066	**0.230 ± 0.010**	0.210 ± 0.010	0.220 ± 0.010	0.214 ± 0.007	0.210 ± 0.010
Bath	0.076	0.564 ± 0.008	0.424 ± 0.006	0.690 ± 0.006	0.650 ± 0.020	**0.757 ± 0.009**
Health	0.076	0.449 ± 0.002	0.448 ± 0.004	0.565 ± 0.009	0.630 ± 0.020	**0.700 ± 0.020**
Diaper	0.084	0.580 ± 0.009	0.457 ± 0.005	0.700 ± 0.010	0.730 ± 0.020	**0.830 ± 0.010**
Bedding	0.079	0.480 ± 0.006	0.482 ± 0.008	0.641 ± 0.009	0.630 ± 0.020	**0.770 ± 0.010**
Safety	0.065	0.250 ± 0.006	0.221 ± 0.004	0.200 ± 0.050	0.230 ± 0.030	**0.250 ± 0.030**
Feeding	0.093	0.560 ± 0.008	0.430 ± 0.002	0.750 ± 0.010	0.696 ± 0.006	**0.810 ± 0.007**
Apparel	0.090	0.533 ± 0.005	0.507 ± 0.004	0.670 ± 0.020	0.650 ± 0.020	**0.750 ± 0.010**
Media	0.094	0.441 ± 0.009	0.420 ± 0.010	0.510 ± 0.010	0.551 ± 0.007	**0.570 ± 0.010**

EquiVSet achieves improvements up to 33% on average

Tencent AI Lab Imperial College London 中山大學 SUN YAT-SEN UNIVERSITY

Thank you!

Code is available at: https://github.com/SubsetSelection/EquiVSet

✓ Welcome to try it out in Colab: Open in Colab