Counterfactual Evaluation of Ads Ranking Models through Domain Adaptation

Mohamed A. Radwan, Himaghna Bhattacharjee, Quinn Lanners, Jiasheng Zhang, Serkan Karakulak, Houssam Nassif, Murat Ali Bayir

Abstract

We propose a domain-adapted reward model that works alongside an Offline A/B testing system for evaluating ranking models. This approach effectively measures reward for ranking model changes in large-scale Ads recommender systems, where model-free methods like IPS are not feasible. Our experiments demonstrate that the proposed technique outperforms both the vanilla IPS method and approaches using non-generalized reward models.

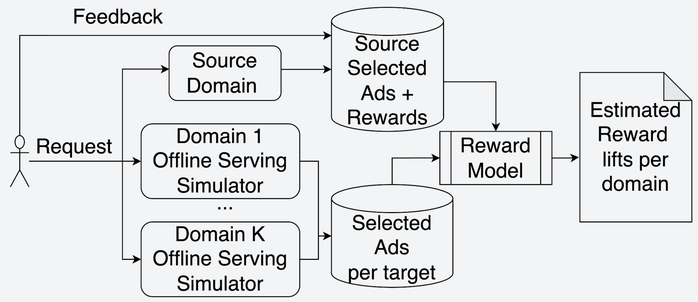
Problem

While offline evaluation of ranking models can be a trivial task in traditional machine learning setups, it is a challenge in the context of large-scale ad recommendation systems:

1. We want to avoid selection bias that would affect reward estimation when using model based approaches like the Direct Method.
2. Ads Recommendation systems involve auctions and rules for selecting organic feed and ads. This makes it too complex to get the correct Propensity Scoring for the selection policy. We thus want to avoid using model free methods like IPS.

Our goal is to estimate the expected reward lift $\widehat{\text{Lift}(T_k, S)}$ between a target domain T_k, and source domain S. Target domains will be one of the new ads recommendation models, and source domain S is the production model that generates evaluation data. We typically have multiple target domains we want to evaluate and one source domain.

Setup



Counterfactual evaluation setup

Proposed Reward Model

Metric: Our goal is to correctly rank target policies, and therefore we need the reward model to perform **fairly** across different domains. To achieve this, we choose the **coefficient of variance of recovery** (Rec_{cv}) as the main performance indicator for the reward model:

$$Rec_{cv} = \frac{Rec_{dev}}{Rec_{avg}}, \quad Rec(T_k, S) = \frac{\widehat{\text{Lift}(T_k, S)}}{\text{Lift}(T_k, S)}$$

where Rec_{dev} is the average absolute deviation of $Rec(.)$, and Rec_{avg} is the average of $Rec(.)$ across all target domains.

Estimator: We introduce a model based approach to **estimate the lift between two domains**. Our approach focuses on modeling the non-overlapping regions between source and target domains, and then using the trained reward model to calculate the expected lift of the target domain T_k over the source domain S (for each target domain k).

Training: we use the following loss function to train on source dataset D_S,

$$\sum_{(x_i, a_i, y_i) \in D_S} L[h(x_i, a_i, \theta), y_i] \times \left[\sum_{k=1}^{K} |w_{a_i}^k - 1| + \beta \sum_{\substack{k=1 \\ k'>k}}^{K} |w_{a_i}^k - w_{a_i}^{k'}| \right]$$

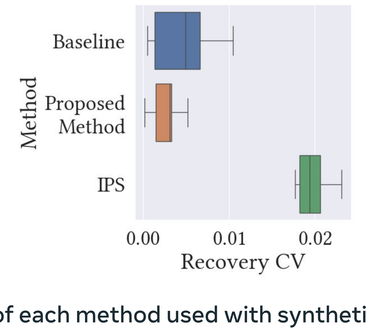
where β is a hyper-parameter and w_a^k is defined by $p_c(a|x)$ which is the probability of observing ad a under context x:

$$w_a^k = \frac{p_{T_k}(a \mid x)}{p_S(a \mid x)}$$

Experiments

We report our findings on both synthetic and online experiment results for a CTR prediction model.

For the **synthetic environment**, we generated domain policies, where each target domain or test variant represents an incremental improvement over the source domain or control variant. We compare our proposed domain adapted reward model to a Direct Model baseline and IPS.



Rec_{CV} of each method used with synthetic data

To evaluate reward models on **real experiments**, we explored the utilization of a completed A/B test for a CTR prediction model. The test included seven suggested variants that improved upon the control. We used the actual lifts reported per variant as ground truth for evaluating the reward models. Given the intractability of propensity score weight in a complex recommendation systems, we train an impression probability estimator per target domain to estimate weights used in the loss. Our proposed reward model showed a **17.6%** improvement on the Recovery CV metric

Take a photo to learn more: