User Sensitivity Modeling in Instagram Ads

Taiyuan Zhang[1] , **Wooyoung Moon**[1] , **Xuhong Zhang**[1] , **William Chen**[1] and **Rui Li**[1]
[1]Meta
{taiyuanz, wooyoung, xuhongzhang, williamhc99, lir}@meta.com

Abstract

In the realm of social media advertising, a delicate balance must be struck between ads value, and user engagement. To achieve an optimal equilibrium, sophisticated ad supply strategies are required, which involve deciding the best time, place and density to show ads to individual users. Predicting users' sensitivity towards ads has been a critical component of the strategy.

This paper presents a case study on how Instagram Ads harnesses the power of causal inference machine learning, to learn each user's sensitivity towards ads, which is then leveraged to enable ads supply personalization. This is an extremely challenging task due to massive scales and subtlety of causal effects from supply changes. By leveraging data from RCTs and SOTA causal inference techniques, we demonstrated that our framework enables accurate estimation of user ads sensitivity over extended periods, through rigorous offline and online studies.

1 Introduction

The integration of sponsored content into social advertising platforms is a delicate balancing act, as it must harmonize user engagement with ads value generation. While increasing ad delivery can yield short-term gains, it may ultimately compromise long-term ads value prospects due to users being driven away. This inherent trade-off poses a significant challenge in the online advertising landscape.

Personalization has emerged as a promising strategy to enhance both ads value and engagement, typically through tailored content selection and ranking. However, this approach primarily solve the problem of "which content to show", but doesn't necessarily answer the question regarding placement: "where and how often should I should sponsored contents?", which is fundamentally an allocation problem instead of a ranking/recommendation problem.

Existing research on ads allocation are mainly focusing on techniques such as constrained optimization [[Yan *et al.*, 2020]] and Markov decision processes [[Liao *et al.*, 2022]] to customize the placement of ads and organic content. However there are notable gaps when applying these methods in industrial applications, which prompts more exploration in this area to achieve a better balance between theoretical soundness vs. practicability.

This paper presents a case study on how a personalized model which predicts user sensitivity towards ads can be leveraged to controlling ad-load levels. The main hypothesis behind this approach is that each user has an underlying preference/sensitivity towards sponsored contents, which should be taken into account to achieve a better equilibrium for both social platform users and the platform itself. We demonstrated we can leverage state-of-the-art causal inference modeling technology to predict such user intrinsic sensitivity, and optimize for value/engagement efficiency at scale.

2 Ads Sensitivity Modeling

Ads Sensitivity is a general term referring to trade-offs between ads value vs. user engagements. Mathematically, we can define it as

$Sensitivity = \frac{\Delta AdsValue}{\Delta Engagement}$

with

- $\Delta AdsValue = E(AdsValue|T = 1, X) - E(AdsValue|T = 0, X)$
- $\Delta Engagement = E(Engagement|T = 1, X) - E(Engagement|T = 0, X)$
- $T \in \{0, 1\}$ indicates the treatment assignment
- X is the user feature vector

The problem then turns into being able to predict both numerator and denominator, both of which are causal inference problems and therefore requiring their own models.

In our case, we adopt a similar methodology for predicting the 2 terms (numerator and denominator terms), however there's entirely flexibility in adopting different methods for predicting the 2 terms, given they can be viewed as independent sub-problems.

2.1 Data Collection

Technically, it's possible to train the 2 causal models from observational data, under the assumption that a good propensity model can be developed.

This assumption, however, is very unlikely to hold in an extremely complex application ecosystem like Instagram.

Therefore, we decided that it's important to collect high quality RCT (random controlled trials) data as foundations for building sensitivity models.

To achieve such a purpose, we leveraged internal systems to create A/B tests which randomly assign users to treatment and control groups. The user assignment is fixed and is expected to remain the same throughout the entire lifecycle of the experiment, to guarantee consistency.

Since the users assigned to the RCT experiments tend to be subjected to other A/B test treatments prior to being assigned to the current experiment, they might be under experiments that could largely alter their in-app experience, and therefore creating unnecessary noise in our RCT data. Therefore, after creating the RCT experiment, we usually enforce a period of no ads-supply treatment, so that these users' experience neutralizes back to normal production behaviors. Empirically, we observe that such a strategy tends to reduce the noise from the data. Here we refer to this as the "baking period".

After the so-called "baking period", we turn on a certain supply treatment for all of the users being assigned to the test group.

Finally, we collect the ads value and Engagement data for all users in both test and control groups, separately from the pre-treatment baking periods, as well as post-treatment periods. This becomes the labels that we train the models to learn.

For features, we take user features X prior to the treatment applications, to ensure no feature leakage.

2.2 Model Architecture

We leverage X-learner, which is one of the state-of-the-art causal inference models, to help us solve the sub-problems of predicting $\Delta AdsValue$ and $\Delta Engagement$. The methodology for the numerator and denominator are similar, so for the rest of this section we'll use $\Delta AdsValue$ as the illustration example.

Here's how the base X-learner model works for predicting $\Delta AdsValue$. It's slightly different from the initial X-learner method, as we combine the treatment and control populations when training the second-stage model.

- $M_1(X)$: This is an Outcome model. It predicts how much ads value will change after applying treatment. This is a regression model, with
  - **Sample Population** = Users for which we apply a certain supply treatment, aka users in the test group
  - **Features** = User level historical features **before** the treatment is applied
  - **Label** = post-treatment ads value - pre-treatment ads value
- $M_0(X)$: Also an Outcome model. It predicts how much ads value will change **without** applying any treatment. This is also a regression model. We can think of this as learning any ecosystem shift that's happening during the training data period.
  - **Sample Population** = Users for which we didn't apply any supply treatment, aka control group
  - **Features** = User level historical features **before** the treatment is applied
  - **Label** = post-treatment ads value - pre-treatment ads value
- $M_\tau(X)$: This is for actually predicting the ΔAdsValue = Y(1) - Y(0) for a single user, with vs. without the supply treatment. This is also a regression model. This model requires us to have observations of the same user's data, with vs. without the treatment. However, the same user can only get 1 treatment, so we only have 1 actual observation, and missing the other observation This where $M_1(X)$ and $M_0(X)$ come in handy - they are used for estimating each user's missing observation in a counterfactual way. We can use them for generating the training data for $M_\tau(X)$:
  - **Sample Population** = All users
  - **Features** = User level historical features **before** the treatment is applied
  - **Label** (if the user is in control) = $M_1(X) - Y(0)$
  - **Label** (if the user is in treatment) = $Y(1) - M_0(X)$

Doubly Robust Estimator

In practice, due to the complexity of the problems and high noise level, this version of X-learner does not perform well on our task. Therefore, we adopt a modified version of X-learner, by incorporating the Doubly Robust Estimator [Shi *et al.*, 2024] [Foster and Syrgkanis, 2023].

Here's how Doubly Robust Outcome model is defined: $\hat{Y}_i^{DR}(X_i, t) = M_t(X_i) + (Y_i - M_t(X_i))1\{T_i = t\}p(T_i = t|X_i)^{-1}$

Here $p(T_i = t|X_i)$ is the propensity score, meaning the likelihood of individual X receiving treatment t. In our case, because we learn from an RCT, we have a fairly high confidence that propensity = constant 0.5, which simplifies our problem.

The key point is that $\hat{Y}_i^{DR}(X_i, 1) - \hat{Y}_i^{DR}(X_i, 0)$ is an unbiased estimator for $ITE(X_i)$

Therefore, by using the DR-enhanced pseudo label, we can achieve better model performance.

2.3 Feature Selection

Ads Supply is an extremely complex system in Instagram. Supply decisions would influence ad density and alter user behaviors to a certain degree, naturally.

This means there's inherent high risks of strong feedback loops when it comes to supply modeling, if any of the features used by the model can be heavily shifted due to supply treatments.

Given the model is learned from pre-treatment features, it does not know the feature distribution after the treatment is applied. If there's a significant feature shift from the treatment itself, then the domain shift will likely render the model less efficient or even completely irrelevant.

Empirically, we do observe when significant feature shifts happen due to the supply treatment itself, the model will perform ok initially but then gradually degrade noticeably.

Therefore, we adopted a set of feature principles to mitigate such risks. There are 2 main principles we adopt

- **Avoid strong feedback loop features**. We quantify how strong the feedback loop is for each feature by comparing their distribution post/pre treatments. As imagined, counter features like ads quantity will be heavily affected by ads supply and therefore ruled out by this principle
- **Stability**. There are feature that will go through large changes in an uncontrollable way. The most obvious examples are those tied to advertiser spends. Advertiser budgets are constantly changing due to the advertiser industry landscape and ecosystem; and big advertisers have the habits of resetting budgets/spends periodically. Therefore, any feature that could be influenced heavily by advertiser spends should be avoided

This helps us narrow down from 1000 features into a lean set of less than 100 features, which also saves training/inference costs at Instagram scale.

3 Offline Evaluation

We evaluate our models under 2 different measures -

- Uplift AUUC (which is a classic, conventional wisdom type of option)
- Bucketed Calibration, 1 new approach we proposed due to the limitation of only looking at AUUC

3.1 Uplift Curve AU(U)C

In ads sensitivity modeling, ads value vs. engagement trade-offs can be visualized using an Uplift Curve. The key idea is, if we sort all of the users by their sensitivity, and apply cutoffs to only apply the treatment to the most efficient users, then each cutoff point gives us an XFor instance, in the hypothetical example below, we see Model A can harvest approximately 50

In general, the better the model, the better trade-offs we can get, and the curve should be higher than the baseline. How do we condense such information into a single number for better quantitative comparison? The typical approach is to use Area Under Uplift Curve, abbreviated as AUUC.

As the name suggests, AUUC refers to the area between the curve and the x-axis. It needs to be greater than 0.5 (because baseline uniform treatment gives you 0.5), and is always ¡= 1 (so normalized and bounded, which is a nice property to have). The conventional wisdom is that the higher the AUUC, the better the model is.

We leverage AUUC as a preliminary metric for filtering out really bad models. However, from the empirical experience of using AUUC, we observe it has several limitations:

1. It only evaluates the final prediction, which is a ratio of 2 individual predictions (one model predicts $\Delta AdsValue$ and one model predicts $\Delta engagement$). If the results aren't good, it's hard to know which model (or if both) has/have problems. Therefore, we need a methodology to evaluate individual lift models.
2. The AUC is across the entire curve, but when applying the model to the system we usually care most about the left part of the curve, aka the high precision area, where we can get a very high % ads value with very little % engagement loss. See figure 1 for a hypothetical example

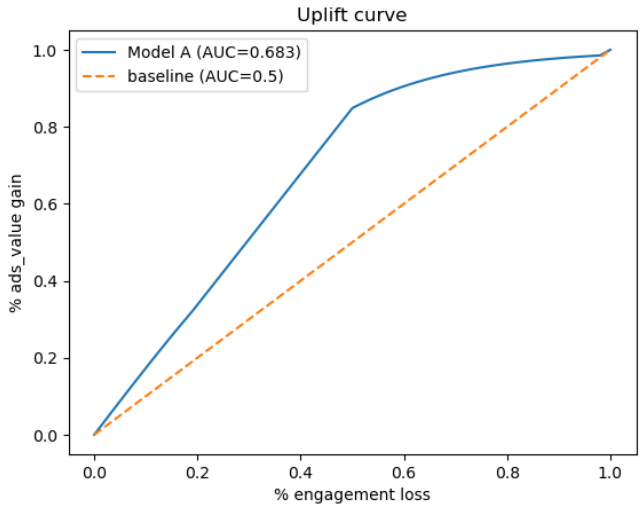

Figure 1: Uplift Curve Example

As example in figure 1 shows, A has higher AUUC than B, but B has obviously better trade-off at the high precision region, and therefore B would be a better candidate for online experimentation. If we were to only look at AUUC, we would have missed this.

3.2 Bucket Calibration

The first problem to crack is how can we look at the individual lift models separately. Inspired by existing work and external literature, we propose to look at Bucketed Calibration of the individual lift models.

1. We take uplift model's predictions $\{M_\tau(X_i)\}$, use it to sort all users
2. Based on the sorting, we can divide users into N buckets (in our settings, we commonly use N=10)
3. Within each bucket, we can compute both actual $CATE$ (Conditional Average Treatment Effect), and $\hat{CATE}$ which is the predicted CATE

where

- $CATE = \frac{\sum_i^N Y_i \cdot (2T_i - 1)}{N}$
- $\hat{CATE} = \frac{\sum_i^N (\{M_\tau(X_i))}{N}$

See figure [2][3] which plot the calibration per bucket of Δ AdsValue model and Δ engagement models respectively, from an actual model iteration.

There are 2 things to look for here:

- **Monotonicity**. The blue bars (which represent the TRUE lifts aggregated from those users) should be monotonically increasing.
- **Calibration**. If the model is perfect, the paired blue bar and the green bar should have the same number (Calibration 100%)

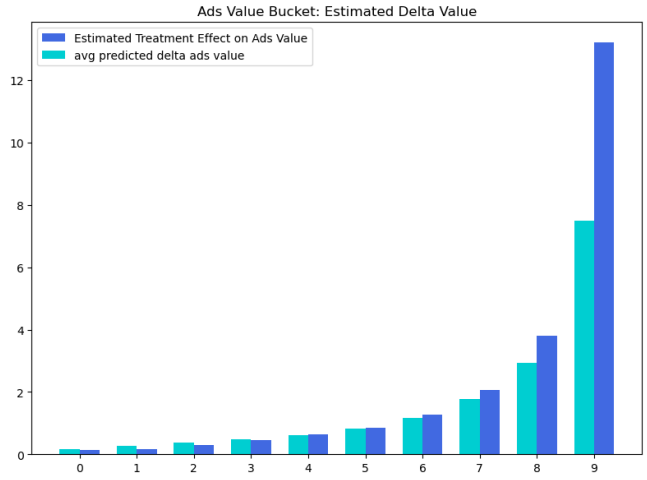

Figure 2: AdsValue Bucket Calibration

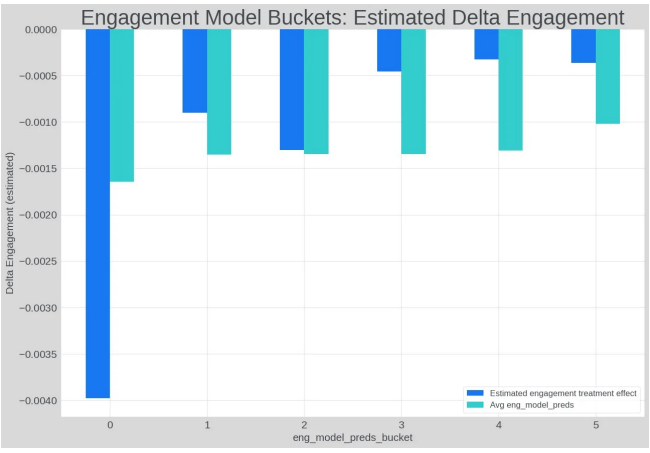

Figure 3: Engagement Bucket Calibration

Based on these 2 criteria, we can gain deeper insights:

- The Δ AdsValue model is decent.
  - Pretty monotonic increase from low to high buckets
  - Calibration is mostly OK (except a little under-cali at top bucket which is expected due to extreme outliers, and bottom bucket likely noise)
- The Δ engagement model is not that great –
  - The blue bars are not monotonic, for instance bucket 2 is actually lower than bucket 1 (but should be higher)
  - Calibration is off everywhere
  - Additionally, all users pretty much have the same predictions (i.e. the green bars are pretty much the same for all buckets). This is also why there are only 6 buckets, even though we requested 10 buckets

Based on the insights we were able to identify the engagement model to be the problem and shifted our focus on improving it.

Note: Why is **Calibration** also important besides **Monotonicity**? Because we compute the final sensitivity based on the ratio of 2 predictions. If one prediction's calibration is off, the final prediction could be directionally off.

Similarly, we can also apply the same BucketCalibration methodology on final Sensitivity predictions (which are ratios of the two separate models).

From there, besides the aforementioned Monotonicity / Calibration, we can also check the Variance between top and bottom regions. The larger the gap is between top and bottom, the more discriminative power the model has and it's usually better. Empirically, we find this approach to be more effective at identifying models that work well in regions we care about the most.

4 Online Results

We conduct an online A/B test before product launch (1 month) and a long-term hold-out test after launch (a couple of months). The results can be found in table [1]

The main efficiency metric can be viewed as the ratio of % ads value gain vs. % engagement loss. We see that after adopting the sensitivity model, we can see a significant lift in efficiency. The efficiency lift show case the power of the sensitivity model. For protection purpose we omit the exact number, and instead we can focus on the comparison between the proposed strategies vs. baseline.

Method	Efficiency Metric
Baseline	X
Strategy 1	2X
Strategy 2	10X

Table 1: Online experiment results

Here

- **Baseline**: Universally applying the treatment to all users

- **Strategy 1**: Only apply the treatment to users with highest predictions
- **Strategy 2**: Apply the treatment to users with highest predictions; and apply the **opposite** treatment to users with lowest predictions

5 Conclusion and Future Work

In this paper, we discussed the case study of how causal inference techniques can be applied to ads sensitivity modeling, and achieve a win-win situation for advertisers, social platform, and social platform users.

The optimization and innovation does not stop here. There are a lot of unique challenges in this domain that are yet to be solved

- A ratio definition of Sensitivity will tend to lead to larger variance due to the 2 variable's variance being compounded. We are actively looking for solutions to reduce the variance and make the final prediction more consistent and stable
- RCT data collection remains an operational challenge for user level modeling. We are actively exploring solutions to enable training from pure observational data. This will also help nautrally solve other problems, such as model performance monitoring, and unblocking recurring training

References

[Foster and Syrgkanis, 2023] Dylan J Foster and Vasilis Syrgkanis. Orthogonal statistical learning. *The Annals of Statistics*, 51(3):879–908, 2023.

[Liao *et al.*, 2022] Guogang Liao, Xiaowen Shi, Ze Wang, Xiaoxu Wu, Chuheng Zhang, Yongkang Wang, Xingxing Wang, and Dong Wang. Deep page-level interest network in reinforcement learning for ads allocation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2292–2296, 2022.

[Shi *et al.*, 2024] Wei Shi, Chen Fu, Qi Xu, Sanjian Chen, Jizhe Zhang, Qinqin Zhu, Zhigang Hua, and Shuang Yang. Ads supply personalization via doubly robust learning. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 4874–4881, New York, NY, USA, 2024. Association for Computing Machinery.

[Yan *et al.*, 2020] Jinyun Yan, Zhiyuan Xu, Birjodh Tiwana, and Shaunak Chatterjee. Ads allocation in feed via constrained optimization. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3386–3394, 2020.