Ad Asset Portfolio Optimization via Policy Learning

Koichi Tanaka
Keio University
Tokyo, Japan
kouichi_1207@keio.jp

Zhi Wang
HAKUHODO Technologies Inc.
Tokyo, Japan
zhi.wang@hakuhodo-technologies.co.jp

Masahiro Asami
HAKUHODO Technologies Inc.
Tokyo, Japan
masahiro.asami@hakuhodo-technologies.co.jp

Kota Ishizuka
HAKUHODO Technologies Inc.
Tokyo, Japan
kota.ishizuka@hakuhodo-technologies.co.jp

Kosuke Kawakami
HAKUHODO Technologies Inc.
Institute of Science Tokyo
Tokyo, Japan
kosuke.kawakami@hakuhodo-technologies.co.jp

Yuta Saito
Hanjuku-kaso, Co., Ltd.
Tokyo, Japan
saito@hanjuku-kaso.com

Abstract

In online advertising, it is crucial to identify optimal ad assets for diverse users to maximize platform revenue. Many algorithms have been developed for selecting personalized ad assets, and have been applied in industrial systems. However, to fully leverage the performance of these algorithms, advertisers are required to submit an appropriate set of ad assets, which we call an *ad asset portfolio*, to the platforms. This problem, which we call *ad asset portfolio optimization (Ad-POP)*, is a crucial issue for online advertising, but it remains underexplored in the existing literature. To tackle this problem, we first formulate Ad-POP in the contextual combinatorial bandits, and reduce it to an off-policy learning problem (OPL), which aims to optimize a new policy solely using historical logged data collected by a different policy. Typical OPL methods can be applied to the Ad-POP problem and categorized into two types of approaches: the independent approach and the exact approach. The independent approach constructs an ad asset portfolio by adding ads with the highest value, while the exact approach treats an ad set as a single action and directly optimizes ad asset portfolios. However, these approaches suffer from high bias and high variance, respectively. To address these challenges, we propose a novel algorithm, named ***Ad Asset Portfolio Optimization via Meta Information (Ad-POM)***. Our algorithm decomposes an ad selection policy into a first-stage policy for selecting meta information, such as the size of the portfolio and the degree of diversity, and a second-stage policy for selecting the portfolio given the meta information. Specifically, we propose a new policy gradient estimator to learn the first-stage policy. This method can achieve stable optimization since it applies importance weighting only to meta information. Our comprehensive experiments on real-world data demonstrate that the proposed method can provide substantial improvements in Ad-POP, where existing methods fail due to the large action space and the interactions among ad assets.

ACM Reference Format:
Koichi Tanaka, Zhi Wang, Masahiro Asami, Kota Ishizuka, Kosuke Kawakami, and Yuta Saito. 2026. Ad Asset Portfolio Optimization via Policy Learning. In *Proceedings of AdKDD (AdKDD'26)*. ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

In online advertising, many studies have proposed ad display algorithms for selecting personalized ad assets given a set (portfolio) of ad assets [2, 5, 6, 15]. However, even if ad display algorithms are optimal, their performance degrades depending on the given ad asset portfolio. For instance, if an advertiser submits few ad assets or low-quality ad assets, the algorithm no longer has any room to optimize ad selection. In contrast, submitting too many ad assets, which may be generated by LLMs, to cover diverse user preferences leads to substantial exploration costs required for the display algorithm to identify the optimal ad asset. This problem, which we call *ad asset portfolio optimization (Ad-POP)*, is a crucial issue for online advertising, but it remains underexplored in the existing literature. To differentiate Ad-POP from the typical ad selection problem, Figure 1 depicts the flow of displaying ads for users. The existing studies mainly focus on constructing algorithms to select optimal ad assets, given a certain ad asset portfolio. In contrast, our focus is on how to choose an optimal ad asset portfolio given a candidate of ad assets. Recent advances in LLMs have made it possible to generate ad asset candidates at almost zero cost, so solving Ad-POP is increasingly important for online advertising.

To tackle the Ad-POP problem, we reduce it to the off-policy learning (OPL) problem by formulating Ad-POP as a contextual combinatorial bandit problem. Based on this formulation, we can apply typical OPL methods to the Ad-POP problem. The main approaches to Ad-POP include exact and independent approaches. The exact approach treats a portfolio as an action and learns a policy selecting an optimal portfolio. This approach can be based on unbiased policy gradients and learns effective policies with sufficient logged data. However, in combinatorial settings, in particular, where the action space corresponds to the combination of unique

AdKDD'26, Jeju, Korea

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

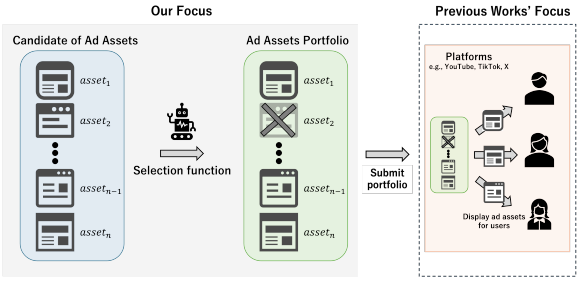


Figure 1: The problem setting of ad asset portfolio optimization. The previous works focus on developing a novel algorithm displaying ad assets within an ad asset portfolio. In contrast, our focus is on how to construct an optimal ad asset portfolio which maximizes the performance of the ad display algorithm.

ad assets, the exact approach collapses due to extremely high variance [1, 9, 10]. On the other hand, the independent approach, which learns the value of each ad asset and selects a portfolio with the highest predicted reward, can avoid the variance issue since it deals with each ad asset rather than combinations of ad assets [4, 13]. However, a portfolio consisting of high-value ad assets is not necessarily optimal since this strategy completely ignores the interaction or diversity of ad assets within a portfolio.

To mitigate the limitations of the exact and independent approaches, we develop a novel ad asset portfolio optimization algorithm called ***Ad Asset Portfolio Optimization via Meta Information (Ad-POM)***. The key idea of Ad-POM is to decompose a policy into a first-stage policy for selecting meta information, such as the size of the ad asset portfolio and the degree of diversity, and a second-stage policy for selecting ad asset portfolios given the meta information. Based on this decomposition, we learn the first-stage policy through the gradient ascent method with a novel policy gradient estimator, called the Ad-POM gradient estimator. The Ad-POM gradient estimator includes importance weighting in the meta information space and a reward regression model to consider the effect of each ad asset. Our analysis shows that the Ad-POM gradient estimator is unbiased under reasonable conditions.

Compared to the exact approaches, Ad-POM can mitigate the variance issue caused by the large action space because the Ad-POM gradient estimator applies importance weighting only to the meta information space. Moreover, compared to the independent approaches, Ad-POM can avoid introducing large bias since Ad-POM explicitly takes into account the interactions between ad assets. Comprehensive experiments on real-world data demonstrate that the proposed method can provide substantial improvements in the Ad-POP problem, where existing methods fail due to the large action space and the interaction effects among ad assets.

2 Problem Formulation

In this section, we first formulate the Ad-POP problem in contextual combinatorial bandits. Let $x \in \mathcal{X} \subseteq \mathbb{R}^{d_x}$ be a context vector such as platforms and campaigns, sampled from an unknown distribution $p(x)$. The finite set of (unique) ad assets is denoted as $\mathcal{B}$, with $b \in \mathcal{B}$ corresponding to an ad asset. Let $a = (b_1, b_2, \cdots, b_l, \cdots, b_{|\mathcal{B}|})$ be a binary ad asset portfolio vector, where b_l is 1 if the advertisement b_l is in the ad asset portfolio. Given a context x, a (possibly) stochastic policy $\pi(a|x)$ chooses an ad asset portfolio a from the finite portfolio space $\mathcal{A}$. Let r be a reward, drawn from an unknown conditional distribution $p(r|x,a)$. We typically measure the performance of a policy π by the following *policy value*.

$$V(\pi) \coloneqq \mathbb{E}_{p(x)\pi(a|x)}[q(x,a)], \tag{1}$$

where $q(x,a) = \mathbb{E}_{p(r|x,a)}[r]$ is the expected reward function given x and a.

Let $\mathcal{D} \coloneqq \{(x_i, a_i, r_i)\}_{i=1}^n$ be the logged data, which contains n independent observations sampled from the logging policy π_0 as $(x,a,r) \sim p(x)\pi_0(a|x)p(r|x,a)$. In Ad-POP, the ultimate goal is to optimize a policy π_θ, parameterized by θ, to maximize the policy value as $\theta^* = \arg\max_{\theta\in\Theta} V(\pi_\theta)$. To solve this policy learning task, there are two typical approaches, namely the exact and independent approaches, as described in detail below.

2.1 Limitations of existing approaches

Although there is no existing literature that formally studies ad asset portfolio optimization, we discuss how to apply the conventional approaches to this problem and their limitations.

First, we review two typical methods within the exact approaches, which are categorized into policy-based and regression-based approaches in terms of their learning methodologies.

The policy-based approach learns the policy parameters via iterative gradient ascent as $\theta_{t+1} \leftarrow \theta_t + \nabla_\theta V(\pi_\theta)$, where

$$\nabla_\theta V(\pi_\theta) = \mathbb{E}_{p(x)\pi_\theta(a|x)}[q(x,a)\nabla_\theta \log \pi_\theta(a|x)] \tag{2}$$

is called the policy gradient. The true policy gradient is unknown, so we need to estimate it from the logged data. A common way to implement the policy-based exact approach is to apply *inverse propensity scoring (IPS)* [3] as

$$\nabla_\theta \widehat{V}_{\mathrm{IPS}}(\pi_\theta; \mathcal{D}) = \frac{1}{n}\sum_{i=1}^n w(x_i,a_i) r_i s_\theta(x_i,a_i), \tag{3}$$

where $w(x,a) = \pi_\theta(a\,|\,x)/\pi_0(a\,|\,x)$ is called the vanilla importance weight, which is defined as the ratio of probabilities that an ad asset portfolio is chosen under two different policies. We also use $s_\theta(x,a) = \nabla_\theta \log \pi_\theta(a|x)$ to denote the policy score function. The IPS policy gradient estimator (IPS-PG) is unbiased under the full support condition.

Condition 2.1. *(Full Support) The logging policy π_0 is said to have full support if $\pi_0(a|x) > 0$ for all $a \in \mathcal{A}$ and $x \in \mathcal{X}$.*

However, in order to satisfy the full support condition, the logging policy assigns extremely small values to some portfolios, leading to severe variance [11, 12].

The regression-based method of the exact approach (Reg-based) estimates the reward function as $\hat{q}(x,a) \approx q(x,a)$ using conventional machine learning methods. It then constructs a policy, for example, by applying the argmax operator as shown below.

$$\pi(a|x) \coloneqq \begin{cases} 1 & (a = \arg\max_{a' \in \mathcal{A}} \hat{q}(x,a')) \\ 0 & (\text{otherwise}) \end{cases} \tag{4}$$

Reg-based can avoid the variance issue because of not using importance weighting. However, Reg-based can introduce severe bias

caused by the difficulty of accurately regressing the expected rewards for every ad asset portfolio. Thus, the exact approaches posit no assumption for the policy optimization, but they suffer from severe bias and variance due to the large action space.

Secondly, the independent approaches posit an independent assumption on the reward function in order to mitigate the aforementioned issue of the exact approaches.

Assumption 2.1. *(Independent Assumption) The independent approaches assume the following structure of the reward function.*

$$q(x,a) = \sum_{l=1}^{|\mathcal{B}|} \eta(x, b_l)\mathbb{I}\{b_l = 1\},$$

where $\eta(x,b)$ represents the independent value of each ad asset.

This assumption implies that ad assets have no interaction effects, and the value of a portfolio can be represented by the sum of independent rewards.

Under the assumption, the independent approach estimates the policy gradient in Eq. (2) by applying the *pseudo inverse* estimator (PI) [13] as follows.

$$\nabla_\theta \widehat{V}_{\mathrm{PI}}(\pi_\theta; \mathcal{D}) \coloneqq \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{l=1}^{|\mathcal{B}|}\frac{\pi_\theta(b_i^{(l)} \mid x_i)}{\pi_0(b_i^{(l)} \mid x_i)}\nabla_\theta \log \pi_\theta(b_i^{(l)}|x_i)\right) r_i, \tag{5}$$

where $w(x,b) = \pi_\theta(b^{(l)}|x)/\pi_0(b^{(l)}|x)$ is the *asset-wise* importance weight and $\pi(b^{(l)} = 1|x) = \sum_{a\in\mathcal{A}} \pi(a|x)\mathbb{I}\{b^{(l)} = 1\}$. The PI gradient estimator (PI-PG) is unbiased under Assumption 2.1 and can reduce variance compared to IPS-PG due to its importance weight regarding the ad asset space $\mathcal{B}$ rather than the ad asset portfolio space $\mathcal{A}$ ($|\mathcal{B}| \ll |\mathcal{A}|$). However, PI-PG suffers from severe bias when violating the independent assumption [4].

We can also use the regression-based approach under the independent assumption. The independent regression-based (Independent Reg-based) estimates the independent value using conventional supervised machine learning methods as follows.

$$\theta = \arg\min_\theta \left(r_i - \sum_{l=1}^{|\mathcal{B}|}\hat{\eta}(x, b_i^{(l)})\mathbb{I}\{b_i^{(l)} = 1\}\right) \tag{6}$$

After obtaining $\hat{\eta}(x,b)$, we construct an ad asset portfolio, for example, by adding high-value ad assets. This strategy is not optimal because Independent Reg-based ignores the interaction effects among ad assets.

As discussed above, existing approaches to Ad-POP suffer from bias and variance problems. To achieve a more efficient Ad-POP, the following develops a novel algorithm for Ad-POP that circumvents the bias and variance issues simultaneously.

3 The Proposed Method

This section introduces a novel Ad-POP algorithm called **Ad-POM**. The key idea of Ad-POM is to decompose a policy into first-stage and second-stage policies, using meta information m, as follows.

$$\pi_{\theta,\phi}^{overall}(a|x) = \sum_{m\in\mathcal{M}} \pi_\theta^{1st}(m|x)\cdot\pi_\phi^{2nd}(a|x,m), \tag{7}$$

where $m \in \mathcal{M}$ represents meta information of an ad asset portfolio. The first-stage policy is a part of the overall policy responsible for selecting the meta information (m). In contrast, the second-stage policy is a part to choose the ad asset portfolio (a) based on the meta information already sampled by the first-stage policy. This policy decomposition is defined via a pre-defined meta information space $\mathcal{M}$. This meta information can include any information assumed to influence the value of ad asset portfolios, such as the number of ad assets and the degree of diversity.

Based on this decomposition, Ad-POM optimizes the first-stage policy $\pi_\theta^{1st}(m|x)$ via the policy-based approach, and optimizes the second-stage policy $\pi_\phi^{2nd}(a|x,m)$ via the regression-based approach. Intuitively, we can avoid the variance issue of the exact approaches, such as IPS-PG (Eq. (3)), since we apply the importance sampling technique to the meta information space, which is much smaller than the original action space ($|\mathcal{M}| \ll |\mathcal{A}|$). Moreover, we also avoid the bias issue of the independent approaches, as the second-stage policy considers the meta information selected by the first-stage policy. In the following, we describe how to learn first- and second-stage policies from the logged data $\mathcal{D}$ in order to improve the value of the overall policy.

Optimizing the first-stage policy π_θ^{1st}. We first consider optimizing the first-stage policy π_θ^{1st} given a (pre-trained) second-stage policy. Our ultimate goal is to obtain a better overall policy, so we should train the first-stage policy to improve the overall policy. Therefore, we train the first-stage policy via the policy-based approach as follows.

$$\theta_{t+1} \leftarrow \theta_t + \nabla_\theta V\left(\pi_{\theta,\phi}^{overall}\right) \tag{8}$$

We derive the true policy gradient of the overall policy regarding the first-stage policy parameters below.

$$\nabla_\theta V(\pi_{\theta,\phi}^{overall}) = \mathbb{E}_{p(x)\pi_\theta^{1st}(m|x)}[q^{\pi_\phi^{2nd}}(x,m)s_\theta(x,m)], \tag{9}$$

where we use $q^{\pi_\phi^{2nd}}(x,m) = \mathbb{E}_{\pi_\phi^{2nd}(a|x,m)}[q(x,a)]$ to denote the value of meta information under π_ϕ^{2nd} and $s_\theta(x,m) = \nabla_\theta \log \pi_\theta^{1st}(m|x)$ to denote the policy score function of the first-stage policy.

We have no access to the true policy gradient in Eq. (9) due to the inability to know $q^{\pi_\phi^{2nd}}$, so we have to estimate it using the logged data. We achieve this via the following Ad-POM gradient estimator.

$$\nabla_\theta \widehat{V}_{Ad-POM}(\pi_{\theta,\phi}^{overall}; \mathcal{D}) \coloneqq \frac{1}{n}\sum_{i=1}^{n}\Big\{w(x_i, m_{a_i})\left(r_i - \hat{f}(x_i, a_i)\right)s_\theta(x_i, m_{a_i}) + \mathbb{E}_{\pi_\theta^{1st}(m|x_i)}\left[\hat{f}^{\pi_\phi^{2nd}}(x_i,m)s_\theta(x_i,m)\right]\Big\}, \tag{10}$$

where $w(x,m) = \pi_\theta^{1st}(m|x)/\pi_0^{1st}(m|x)$ is the *meta information importance weight* and m_a represents the meta information to which the ad asset portfolio a belongs. Specifically, the first term of Eq. (10) estimates the value of meta information m via the meta information importance weighting and the second term deals with the values of each ad asset via a regression model $\hat{f}$. Since our policy gradient estimator applies importance weighting with respect to only the meta information space, it is expected to have a much lower variance compared to IPS-PG in Eq. (3). Additionally, since it does

not introduce the independent assumption, it is expected not to produce large bias due to ignoring the interaction effects.

As a theoretical analysis, we characterize the bias of the Ad-POM gradient estimator under the following full meta information support condition, which is milder than the full support condition required for the unbiasedness of IPS-PG.

Condition 3.1. *(Full Meta Information Support) The logging policy π_0 is said to have full meta information support if $\pi_0(m|x) > 0$ for all $m \in \mathcal{M}$ and $x \in \mathcal{X}$.*

Theorem 3.1. *(Bias of the Ad-POM gradient estimator) Under Condition 3.1, the Ad-POM gradient estimator has the following bias for a given regression model $\hat{f}(x,a)$.*

$$\begin{aligned} &Bias(\nabla_\theta \widehat{V}_{Ad-POM}(\pi^{overall}_{\theta,\phi}; \mathcal{D})) \\ &= \mathbb{E}_{p(x)\pi_0^{1st}(m|x)}\Bigg[\sum_{a<b:m_a=m_b=m} \pi_0^{2nd}(a|x,m)\pi_0^{2nd}(b|x,m) \\ &\quad \times (\Delta_q(x,a,b) - \Delta_{\hat{f}}(x,a,b))(w(x,b)-w(x,a))s_\theta(x,m)\Bigg], \quad (11) \end{aligned}$$

where $a, b \in \mathcal{A}$. $\Delta_q(x,a,b) = q(x,a) - q(x,b)$ represents the difference in the expected rewards between a pair of ad asset portfolios a and b. $\Delta_{\hat{f}}(x,a,b) = \hat{f}(x,a) - \hat{f}(x,b)$ is the difference in the expected rewards estimated by a regression model $\hat{f}$.

The bias of the Ad-POM gradient estimator in Eq. (11) is characterized by $\Delta_q(x,a,b) - \Delta_{\hat{f}}(x,a,b)$. When a regression model $\hat{f}(x,a)$ preserves the relative value differences of ad asset portfolios represented by the same meta information, its bias becomes small. This analysis suggests that the Ad-POM gradient estimator is unbiased under the following Conditional Pairwise Correctness (CPC) condition.

Condition 3.2. *(Conditional Pairwise Correctness; CPC) A regression model $\hat{f}(x,a)$ satisfies conditional pairwise correctness if $\Delta_q(x,a,b) = \Delta_{\hat{f}}(x,a,b)$ for all $x \in \mathcal{X}$ and $a, b \in \mathcal{A}$ s.t. $m_a = m_b$.*

Thus, a regression model used in the Ad-POM gradient estimator is only required to satisfy conditional pairwise correctness, which is less restrictive than correctly estimating the expected reward $q(x,a)$, needed for Reg-based in Eq. (4).

Next, we characterize the variance of the Ad-POM gradient estimator to demonstrate its variance reduction.

Theorem 3.2. *(Variance of the Ad-POM gradient estimator) Under Condition 3.1 and 3.2, the Ad-POM gradient estimator has the following variance.*

$$\begin{aligned} &n\mathbb{V}_{\mathcal{D}}(\nabla_\theta \widehat{V}^{(j)}_{\mathrm{Ad-POM}}(\pi^{overall}_{\theta,\phi}; \mathcal{D})) \\ &= \mathbb{E}_{p(x)\pi_0(a|x)}\left[\left(w(x,m)s^{(j)}_\theta(x,m)\right)^2 \sigma^2(x,a)\right] \\ &\quad + \mathbb{E}_{p(x)}\left[\mathbb{V}_{\pi_0(a|x)}\left[w(x,m)\Delta_{q,\hat{f}}(x,a)s^{(j)}_\theta(x,m)\right]\right] \\ &\quad + \mathbb{V}_{p(x)}\left[\mathbb{E}_{\pi^{1st}_\theta(m|x)}\left[q^{\pi^{2nd}_\phi}(x,m)s^{(j)}_\theta(x,m)\right]\right], \quad (12) \end{aligned}$$

where $\Delta_{q,\hat{f}}(x,a) \coloneqq q(x,a) - \hat{f}(x,a)$ is the estimation error of $\hat{f}(x,a)$ against the expected reward function $q(x,a)$.

Algorithm 1 The second-stage policy

Require: the number of creatives d, degree of diversity λ, a regression model $\hat{\eta}(x,b)$, and diversity function $\mathrm{Div}(a)$.
Ensure: advertisement group a

```
Set a = (0, ..., 0) and B_1 = B
for t = 1, ..., d do
    b^(l_t) = arg max_{b ∈ B_t} η̂(x, b) + λ · Div(a ∪ b)
    a^(l_t) = 1 and B_{t+1} = B_t \ b^(l_t)
```

Theorem 3.2 shows that the variance of the Ad-POM gradient estimator depends only on $w(x,m)$ rather than $w(x,a)$, resulting in a significant variance reduction compared to IPS-PG.

Optimizing the second-stage policy π^{2nd}_ϕ. We thus develop the policy-based approach to optimize the first-stage policy via the Ad-POM gradient estimator. The remaining objective is to identify the optimal ad asset portfolios, given meta information selected by the first-stage policy. For example, if the meta information includes the number of ad assets d and a degree of ad assets' diversity λ, the second-stage policy chooses an ad asset portfolio like Algorithm 1. First, we estimate the structure of the expected reward function. If we assume that an ad asset portfolio effect includes the independent value of ad assets and diversity, the structure of the expected reward function is given by

$$\tilde{q}(x,a) = \sum_{l=1}^{|\mathcal{B}|} \eta(x,b_l)\mathbb{I}\{b_l = 1\} + \lambda \cdot \mathrm{Div}(a) \quad (13)$$

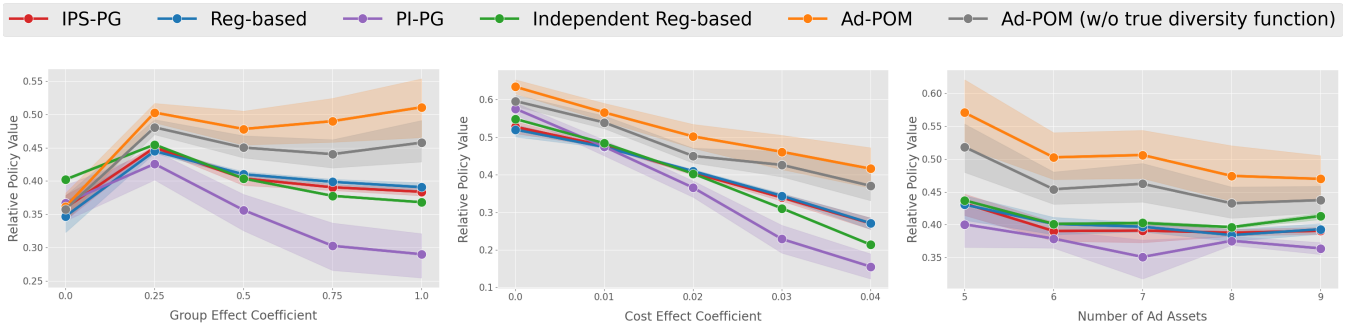
where $\mathrm{Div}(a)$ is an arbitrary diversity function. Then, the second-stage policy selects an ad asset with the highest value, given meta information. This algorithm considers the asset diversity as well as the independent value of assets, so Ad-POM has lower bias compared to the independent approaches, which rely on the independent assumption. It is worth noting that we need not estimate the exact structure of the expected reward function, as we demonstrate in the experiments.

4 Empirical Evaluation

To evaluate the performance of the Ad-POM algorithm, this section conducts Ad-POP experiments on the public dataset in display advertising, called *CreativeRanking* [14].[1] CreativeRanking contains a large and diverse set of ad assets from the Alibaba display advertising platform as well as their real impression and click data.

To conduct Ad-POP experiments, we use the data for products that have at least n_b types of ad assets in the dataset for our experiments. We then transform ad asset images into 10-dimensional ad asset context vectors e using CLIP [8] and PCA implemented in scikit-learn [7]. We define a product context vector x as the average of ad asset context vectors e within the product. For each product x and ad asset portfolio a pair, we synthesize the expected reward

[1] Note that our code to replicate the experiment results will be made available on a GitHub repository upon publication.



(a) Comparisons of relative policy value normalized by the optimal policy value with varying group effect coefficient (λ_g^*).

(b) Comparisons of relative policy value normalized by the optimal policy value with varying cost effect coefficient (λ_c^*).

(c) Comparisons of relative policy value normalized by the optimal policy value with varying the number of ad assets.

Figure 2: Comparisons of relative policy value normalized by the optimal policy value with varying (a) group effect coefficient (λ_g^*), (b) cost effect coefficient (λ_c^*), (c) the number of ad assets. Note that the shaded regions in the plots represent the 95% confidence intervals.

as follows.

$$q(x,a) = (1-\lambda_g^*)\cdot\sum_{l=1}^{|\mathcal{B}|}\eta(x,b_l)\mathbb{I}\{b_l=1\} + \lambda_g^*\cdot \mathrm{Div}(a) - \lambda_c^*\cdot|a|, \tag{14}$$

where λ_g^*, λ_c^* are the group and cost effect coefficients, and $\mathrm{Div}(a)$ is a diversity function. The cost term considers the exploration cost of an ad display algorithm, which penalizes a large size of ad asset portfolios. We define the independent reward of each ad asset $\eta(x,b)$ by calculating the click through rate (CTR). The diversity function is given by

$$\mathrm{Div}(a) = 1 - \frac{1}{|a|^2}\sum_{i<j}\frac{e_i\cdot e_j}{|e_i|\cdot|e_j|}. \tag{15}$$

The diversity function becomes small when the cosine similarity of ad assets within a portfolio is large. Based on the above synthetic reward function, we sample reward r from a normal distribution whose mean is $q(x,a)$ and standard deviation σ is 0.5.

We then synthetically define the first-stage and second-stage logging policies as follows.

$$\pi_0^{1st}(m|x) = \frac{\exp(\beta_1\cdot(w^T\cdot x_m))}{\sum_{m'\in\mathcal{M}}\exp(\beta_1\cdot(w^T\cdot x_m))}, \tag{16}$$

$$\pi_0^{2nd}(a|x,m) = \frac{\exp(\beta_2\cdot q(x,a))}{\sum_{a'\in\mathcal{A}_m}\exp(\beta_2\cdot q(x,a'))}, \tag{17}$$

where x_m represents a meta information context vector and w is sampled from a uniform distribution with range $[-1,1]$. β_1 and β_2 are the experiment parameters to control the stochasticity and optimality of these policies. When β_1 and β_2 approach zero, the first-stage and second-stage logging policies become close to a uniform random distribution. We use $\beta_1 = -2.0$ and $\beta_2 = 5.0$.

Compared methods. We compare Ad-POM with IPS-PG (Eq. (3)), Reg-based (Eq. (4)), PI-PG (Eq. (5)), and Independent Reg-based (Eq. (6)). Regarding Ad-POM, we compare its results with its variant, "Ad-POM (w/o true diversity function)". Ad-POM uses the estimated reward structure $\tilde{q}$ in Eq. (13) as $\tilde{q}(x,a) = q(x,a)$. However, this method is infeasible in practice because we do not have access to the true reward structure. Therefore, Ad-POM (w/o true diversity function) provides a practical performance of Ad-POM by using a different diversity function $\widehat{\mathrm{Div}}$.

$$\widehat{\mathrm{Div}}(a) = 1 - \frac{1}{|a|}\sum_{i<j}||e_i - e_j||^2. \tag{18}$$

By comparing Ad-POM with Ad-POM (w/o true diversity function), we can evaluate Ad-POM's performance under unknown reward functions. Moreover, we define the meta information space using the size of the ad asset portfolio, the magnitude of the group effect λ_g and the cost effect λ_c.

4.1 Results

From Figure 2a to 2c, we show the policy values normalized by those of the optimal policy in the test set. To produce synthetic data instances, we compute over 25 simulations with different random seeds. Unless otherwise specified, the logged data size is set to $n = 2000$, the number of ad assets is $n_b = 6$, the group effect coefficient is $\lambda_g^* = 0.5$, and the cost effect coefficient is $\lambda_c^* = 0.02$.

How does Ad-POM perform with varying the group effect coefficient? Figure 2a varies the group effect coefficient from 0 to 1.0. The larger value of λ_g^* indicates that the diversity among ad assets has a larger effect on the value of ad asset portfolios. The results show that Ad-POM outperforms the baselines across various group effect coefficients by effectively reducing bias and variance. Specifically, the exact approaches, such as IPS-PG and Reg-based, show little variation and yield lower performance than Ad-POM. This is because Ad-POM uses importance weighting regarding only the meta information while the exact approaches treat an ad asset portfolio as an action. Moreover, we see that PI-PG gradually decreases as the degree of diversity increases due to ignoring the interaction effect among ads. In contrast, Ad-POM provides effective Ad-POP with larger values of λ_g^* since Ad-POM posits no

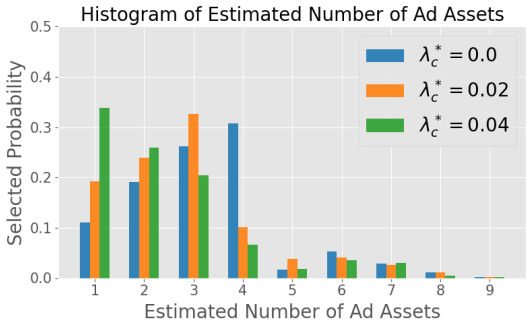


Figure 3: Histogram of the number of ad assets estimated by Ad-POM.

assumption of the reward function, like Assumption 2.1. It is worth noting that the performance of Ad-POM is slightly worse than that of the independent approaches, such as PI-PG and Independent Reg-based, only when $\lambda_g^* = 0.0$. This is because $\lambda_g^* = 0.0$ indicates that there is no interaction effect among the ad assets, which means the independent assumption is rigorously satisfied.

It would also be intriguing to see that Ad-POM (w/o true diversity function) is superior to the baselines even though Ad-POM (w/o true diversity function) conducts Ad-POP based on a non-true reward function. This suggests that Ad-POM is effective in a more realistic situation where the reward function is not perfectly known.

How does Ad-POM perform with varying the cost effect coefficient? Figure 2b compares the effectiveness of policy learning by the resulting policy values of each Ad-POP method with varying the cost effect coefficient. The cost effect considers the exploration cost of an ad display algorithm. In Figure 2b, we observe that Ad-POM provides the best Ad-POP performance compared to the baselines with every value of λ_c^*. Specifically, the performance of all methods gradually decreases as the cost effect increases. This is because a larger cost effect increases the number of ad assets with negative effects on the ad asset portfolio, which makes optimization more difficult. To evaluate how Ad-POM adapts to changes in cost effects, Figure 3 illustrates Ad-POM's probability of selecting an ad asset portfolio of a given size. We observe that the number of ad assets within the selected portfolio gradually decreases as the exploration cost λ_c^* increases. Specifically, the most selected number of ad assets is 4 when $\lambda_c^* = 0.0$, 3 when $\lambda_c^* = 0.02$, and 1 when $\lambda_c^* = 0.04$. This result suggests that the first-stage policy of Ad-POM can optimize meta information depending on the structure of the expected reward function.

How does Ad-POM perform with varying the number of ad assets? Figure 2c reports the comparisons of Ad-POP methods with varying the number of ad assets n_b within each product. When the value of n_b increases from 5 to 9, the number of possible portfolios increases from $2^5 = 32$ to $2^9 = 512$. We observe that all methods gradually decrease as the number of ad assets increases. However, Ad-POM retains higher performance compared to the baselines with various values of n_b. This is because Ad-POM efficiently reduces its variance by using the importance weighting regarding only the meta information compared to the importance weighting in $\mathcal{A}$ for IPS-PG.

5 Conclusion

This paper studies the Ad-POP problem for the first time. We started by formulating Ad-POP in contextual combinatorial bandits. We then analyzed the bias and variance issues of the standard approaches, which arise due to the large action space and interactions among ad assets. To mitigate these issues, we proposed the Ad-POM algorithm based on policy decomposition via meta information. The Ad-POM algorithm optimizes the first-stage policy selecting meta information through a new policy gradient estimator, which is unbiased under a relaxed condition and has significantly lower variance than existing approaches. The second-stage policy, which selects an ad asset portfolio given meta information, is constructed via a regression model and an estimated reward structure. Empirical evaluations demonstrate that Ad-POM can provide substantial improvements in challenging situations where the action space and the interaction effects are large.

References

[1] Matej Cief, Jacek Golebiowski, Philipp Schmidt, Ziawasch Abedjan, and Artur Bekasov. 2024. Learning action embeddings for off-policy evaluation. In *European Conference on Information Retrieval*. Springer, 108–122.
[2] Zhe Feng, Christopher Liaw, and Zixin Zhou. 2023. Improved online learning algorithms for ctr prediction in ad auctions. In *International Conference on Machine Learning*. PMLR, 9921–9937.
[3] Daniel G Horvitz and Donovan J Thompson. 1952. A generalization of sampling without replacement from a finite universe. *Journal of the American statistical Association* 47, 260 (1952), 663–685.
[4] Haruka Kiyohara, Masahiro Nomura, and Yuta Saito. 2024. Off-policy evaluation of slate bandit policies via optimizing abstraction. In *Proceedings of the ACM on Web Conference 2024*. 3150–3161.
[5] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*. 661–670.
[6] Sandeep Pandey and Christopher Olston. 2006. Handling advertisements of unknown quality in search advertising. *Advances in neural information processing systems* 19 (2006).
[7] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
[8] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PmLR, 8748–8763.
[9] Noveen Sachdeva, Lequn Wang, Dawen Liang, Nathan Kallus, and Julian McAuley. 2023. Off-Policy Evaluation for Large Action Spaces via Policy Convolution. *arXiv preprint arXiv:2310.15433* (2023).
[10] Yuta Saito and Thorsten Joachims. 2022. Off-Policy Evaluation for Large Action Spaces via Embeddings. In *International Conference on Machine Learning*. PMLR, 19089–19122.
[11] Yuta Saito, Jihan Yao, and Thorsten Joachims. 2024. POTEC: Off-Policy Learning for Large Action Spaces via Two-Stage Policy Decomposition. *arXiv preprint arXiv:2402.06151* (2024).
[12] Tatsuhiro Shimizu, Koichi Tanaka, Ren Kishimoto, Haruka Kiyohara, Masahiro Nomura, and Yuta Saito. 2024. Effective Off-Policy Evaluation and Learning in Contextual Combinatorial Bandits. In *Proceedings of the 18th ACM Conference on Recommender Systems*. 733–741.
[13] Adith Swaminathan, Akshay Krishnamurthy, Alekh Agarwal, Miro Dudik, John Langford, Damien Jose, and Imed Zitouni. 2017. Off-Policy Evaluation for Slate Recommendation. In *Advances in Neural Information Processing Systems*, Vol. 30. 3632–3642.
[14] Shiyao Wang, Qi Liu, Tiezheng Ge, Defu Lian, and Zhiqiang Zhang. 2021. A hybrid bandit model with visual priors for creative ranking in display advertising. In *Proceedings of the web conference 2021*. 2324–2334.
[15] Min Xu, Tao Qin, and Tie-Yan Liu. 2013. Estimation bias in multi-armed bandit algorithms for search advertising. *Advances in Neural Information Processing Systems* 26 (2013).