Beyond Cascaded Architectures: An End-to-end Generative Framework for Industrial Advertising

Zuowu Zheng*
Meituan
Shanghai, China
zhengzuowu@meituan.com

Ze Wang*
Meituan
Shanghai, China
wangze18@meituan.com

Fan Yang†
Meituan
Shanghai, China
yangfan129@meituan.com

Jiangke Fan
Meituan
Shanghai, China
jiangke.fan@meituan.com

Teng Zhang
Meituan
Shanghai, China
zhangteng09@meituan.com

Xingxing Wang
Meituan
Beijing, China
wangxingxing04@meituan.com

Abstract

Traditional online industrial advertising systems suffer from the limitations of multi-stage cascaded architectures, which often discard high-potential candidates prematurely and distribute decision logic across disconnected modules. While recent generative recommendation approaches provide end-to-end solutions, they fail to address critical advertising requirements of key components for real-world deployment, such as explicit bidding, creative selection, ad allocation, and payment computation.

To bridge this gap, we introduce End-to-End Generative Advertising (EGA), the first unified framework that holistically models user interests, point-of-interest (POI) and creative generation, ad allocation, and payment optimization within a single generative model. Our approach employs hierarchical tokenization and multi-token prediction to jointly generate POI recommendations and ad creatives, while a permutation-aware reward model and token-level bidding strategy ensure alignment with both user experiences and advertiser objectives. Additionally, we decouple allocation from payment using a differentiable ex-post regret minimization mechanism, guaranteeing approximate incentive compatibility at the POI level. Through extensive offline evaluations and large-scale online experiments on real-world advertising platforms, we demonstrate that EGA significantly outperforms traditional cascaded systems in both performance and practicality. Our results highlight its potential as a pioneering fully generative advertising solution, paving the way for next-generation industrial ad systems.

CCS Concepts

• **Information systems** → **Display advertising**.

Keywords

Generative Advertising, Recommendation System, Preference Alignment

ACM Reference Format:
Zuowu Zheng, Ze Wang, Fan Yang, Jiangke Fan, Teng Zhang, and Xingxing Wang. 2018. Beyond Cascaded Architectures: An End-to-end Generative Framework for Industrial Advertising. In *Proceedings of Woodstock '18: ACM Symposium on Neural Gaze Detection (Woodstock '18).* ACM, New York, NY, USA, 10 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

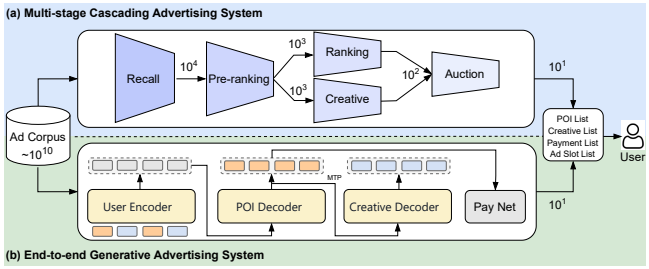

Figure 1: (a) A typical cascade advertising system. (b) Our proposed unified architecture for end-to-end generation.

Online advertising has become a critical revenue engine for major internet platforms, directly supporting the sustainability of a wide range of digital services. Each user request initiates a complex decision process, where the platform must select, rank, and display a mixture of ads and organic content. Traditionally, industrial advertising systems have relied on multi-stage cascading architecture, which is typically structured as: recall, pre-ranking, ranking, creative selection, and auction respectively. As shown in Figure 1 (a), each stage selects the top-k candidates from its input and forwards them to the subsequent stage. This paradigm effectively balances performance and efficiency by progressively narrowing down the optimal set of candidate items under strict latency constraints.

However, it still suffers from intrinsic limitations. Earlier stages in the cascade inherently constrain the performance upper bound of downstream modules [4]. For example, ads filtered out in upstream recall or ranking stages cannot be recovered, even if they

*Equal contribution.
†Corresponding author.

Woodstock '18, Woodstock, NY

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

would be highly valuable in the final allocation, resulting in reduced platform revenue and failure to achieve global optimality. Despite various efforts to enhance overall recommendation performance by promoting interaction between ranking modules [11, 36, 39], these approaches continue to operate within the traditional cascaded ranking framework.

The emergence of generative recommendation frameworks offers a new perspective on these longstanding challenges. Recent advances have demonstrated that transformer-based sequence models can unify retrieval, ranking, and generation in an end-to-end manner, delivering personalized content and capturing deep user-item dependencies [4, 20, 31]. However, while such approaches have shown remarkable promise in organic recommendation scenarios, the complexity of industrial advertising presents a unique set of obstacles. Advertising systems must satisfy strict business constraints, including bidding, creative selection, ad slot allocation, and payment rules, while optimizing both user and platform objectives. These requirements introduce complex dependencies and practical challenges that cannot be fully addressed by directly applying existing generative recommendation models.

To address above challenges, we propose End-to-end Generative Advertising (EGA), a novel generative framework that unifies all decision-making stages into a single model. EGA bridge the gap between generative modeling and the practical requirements of industrial advertising, which directly outputs the final ad sequence, as well as corresponding creatives, positions, and payment end-to-end. *Firstly*, inspired by generative recommendation techniques, we leverage Residual Quantized Variational AutoEncoder (RQ-VAE) to encode user behavior and item features into hierarchical semantic tokens, and employ an encoder-decoder architecture with multi-token prediction to jointly generate candidate ad sequences and creative content. *Secondly*, we propose a token-level bidding and generative allocation mechanism, enhanced by permutation-aware reward modeling. This strategy decouples allocation from payment, allowing the model to effectively reflect business objectives during generation while approximately preserving incentive compatibility (IC) with a differentiable payment network. *Last but not least*, we introduce a multi-phase training paradigm. A pre-training phase learns user interests from actual exposure sequences that contain ad and organic items. Then auction-based post-training is applied to fine-tune the model with ad-specific constraints, dynamically optimizing ad allocation based on auction signals and platform objectives.

Our contributions are as follows:

- We introduce EGA, which surpasses traditional multi-stage cascading architectures by an unified single generative model. To the best of our knowledge, this is one of the first end-to-end generative advertising framework in industry.
- We propose a novel multi-phase training strategy consisting of interest-based pre-training and auction-based post-training. This design effectively balances user interests modeling with advertising-specific constraints, leverages diverse data sources during training, and aligns model outputs with final platform objectives.
- Extensive offline experiments and online A/B tests in large-scale industrial datasets demonstrate the effectiveness and efficiency of our approach.

2 Related Works

To achieve end-to-end optimization in industrial advertising, it is crucial to unify user behavior modeling, auction mechanisms, and ad allocation strategies. Motivated by this integration, we review recent advances in each area in this section.

2.1 Generative Recommendation

In recent years, there has been growing interest in applying generative paradigms to recommendation systems. A particularly promising direction is to formulate recommendation as a sequence generation task, where user-item interactions are modeled using transformer-based autoregressive architectures. These methods aim to deeply capture user behavioral context and generate personalized item sequences in an end-to-end fashion [3, 4, 20, 23, 24, 31, 34].

Tiger [20] is a pioneering approach that introduces RQ-VAE to encode item content into hierarchical semantic IDs, allowing knowledge sharing across semantically similar items. Building upon this, COBRA [31] proposes a two-stage generation framework that first produces sparse IDs and then refines them into dense vectors, enabling a coarse-to-fine retrieval process. Another stream of research explores general-purpose generative recommenders. HSTU [34] reformulates recommendation as a sequential transduction task and designs a Transformer architecture tailored for high-cardinality, non-stationary streaming data. OneRec [4] further advances this by unifying retrieval, ranking, and generation within a single encoder-decoder framework, while incorporating session-level generation and preference alignment strategies to enhance output quality. Besides, several methods focus on enhancing semantic tokenization and representation. LC-Rec [40] aligns semantic IDs with collaborative filtering signals via auxiliary objectives. IDGenRec [22] leverages large language models to generate dense textual identifiers, showing strong generalization in zero-shot scenarios. SEATER [21] introduces tree-structured token spaces trained with contrastive and multi-task objectives to ensure consistency, while ColaRec [26] bridges content and interaction spaces for better alignment.

While these works demonstrate strong general recommendation performance, they are insufficient for online advertising systems, which require additional modeling of bidding, payment, and allocation constraints not captured in pure user interests modeling.

2.2 Auction Mechanism

Traditional auction mechanisms such as GSP [8] and its variants like uGSP [1] are widely deployed in online advertising due to their simplicity, interpretability, and strong revenue guarantees. However, they operate under the assumption of independence among ads, failing to account for externalities, that is, the influence of other ads [9, 12].

To address this, recent advances in computation have motivated the development of learning-based auction frameworks [35]. For example, DeepGSP [37] and DNA [19] extend classical auctions by incorporating online feedback into end-to-end learning pipelines. However, DNA suffers from the evaluation-before-ranking dilemma,

where the rank score must be predicted before knowing the final sequence, limiting its capacity to model set-level externalities. Other approaches attempt to integrate optimality into auction design. NMA [14] tackles this by exhaustively enumerating all possible allocations to ensure global optimality, but its computational cost renders it impractical for real-time applications. CGA [41] addresses the limitations of traditional and learning-based ad auctions by explicitly modeling permutation-level externalities through an autoregressive allocation model and a gradient-friendly reformulation of incentive compatibility, enabling end-to-end optimization of both allocation and payment.

2.3 Ad Allocation

Platforms initially assigned fixed positions to ads and organic items, but dynamic ad allocation strategies are now gaining attention for their potential to optimize overall page-level performance [15, 27–30, 38]. CrossDQN [16] introduces a DQN architecture to incorporate the arrangement signal into the allocation model without modifying the relative ranking of ads. HCA2E [2] proposes hierarchically constrained adaptive ad exposure that possesses the desirable game-theoretical properties and computational efficiency. MIAA [13] presents a deep automated mechanism that integrates ad auction and allocation, which simultaneously decides the ranking, payment, and display position of the ad.

3 Preliminary

This section provides the necessary preliminaries for our approach. We define the core task in online advertising, and present the auction mechanism design that connects interests modeling with business objectives. The main notations are summarized in Table 1.

3.1 Task Formulation

We formalize a typical task of joint ad generation and allocation in online advertising systems. Given a page view (PV) request from user u, there are N candidate ads and M candidate organic items (non-sponsored). The organic sequence O is assumed to be pre-ranked by an upstream module based on estimated GMV, and its internal order will remain fixed[1]. The system selects a final ranked list of K items ($K \ll (N+M)$) to display:

$$\mathcal{Y} = \{y_1, y_2, \ldots, y_K\}, \quad y_i \in X \cup O, \tag{1}$$

where the output list $\mathcal{Y}$ is generated by making the following decisions jointly under both user engagement and business constraints:

- **Ad ranking:** decide the permutation of selected ads from X;
- **Creative selection:** for each chosen ad, generate the most appropriate creative image;
- **Payment:** compute the payment p_i for each exposed ad based on its bid b_i and allocated position;
- **Ad slot:** determine the optimal display position for each ad when completing with organic contents.

Each advertiser i submits a bid b_i corresponding to its private click value v_i. Our objective is maximize the expected platform revenue

[1]This reflects common platform design constraints where organic content is ranked independently and must preserve user experience.

Table 1: Summary of Notation

Symbol	Description
u	User u
$X = \{x_1, \ldots, x_N\}$	Candidate set of N ads
$O = \{o_1, \ldots, o_M\}$	Candidate set of M organic contents
$\mathcal{Y} = (y_1, \ldots, y_K)$	Final ranked list of K selected items
$\mathcal{S}^u = \{y_1, ..., y_B\}$	User historical behaviors of length B
$\mathcal{V}$	Codebooks
C	The number of codebook layers
W	Codebook size of each layer
pCTR_i	Predicted click-through rate of i-th item
b_i	Bid value submitted by i-th advertiser
v_i	Private value of i-th advertiser
p_i	Payment charged to i-th advertiser
u_i	Utility of i-th advertiser
$\boldsymbol{b}_{-i}$	Bids profile of ads except i-th ad
$\mathcal{M}\langle\mathcal{A}, \mathcal{P}\rangle$	Auction mechanism
Rev	Platform expected revenue
$\boldsymbol{e}_i^{poi}, \boldsymbol{e}_i^{img}$	POI and creative image embedding of item i
$\boldsymbol{a}_i^{poi}, \boldsymbol{a}_i^{img}$	POI and creative image token of item i
$\mathcal{F}, \mathcal{R}$	Pre-training model $\mathcal{F}$ and reward model $\mathcal{R}$
$P(\boldsymbol{a}_i)$	Probability of generating token $\boldsymbol{a}_i$
$\hat{r}$	Estimated reward by reward model $\mathcal{R}$

with sequence $\mathcal{Y}$:

$$\max_{\theta} \mathbb{E}_{\mathcal{Y}}\left[\text{Rev}\right] = \max_{\theta} \mathbb{E}_{\mathcal{Y}}\left(\sum_{i=1}^{K} p_i \cdot \text{pCTR}_i\right). \tag{2}$$

3.2 Auction Mechanism Design

3.2.1 Auction Constraints. Unlike traditional recommender systems, advertising platforms must not only optimize platform revenue but also ensure advertiser utility. Given an auction mechanism $\mathcal{M}\langle\mathcal{A}, \mathcal{P}\rangle$ with allocation rule $\mathcal{A}$ and payment rule $\mathcal{P}$, the expected utility u_i for an advertiser i is defined as:

$$u_i(v_i; \boldsymbol{b}) = (v_i - p_i) \cdot \text{pCTR}_i, \tag{3}$$

Two key economic constraints in auction mechanism must be satisfied in Equation (2): incentive compatibility (IC) and individual rationality (IR). IC requires that truthful bidding maximizes the advertiser's utility. For ad x_i, it holds that

$$u_i(v_i; v_i, \boldsymbol{b}_{-i}) \geq u_i(v_i; b_i, \boldsymbol{b}_{-i}), \quad \forall v_i, b_i \in \mathbb{R}^+. \tag{4}$$

IR requires that no advertiser pays more than their bid, that is,

$$p_i \leq b_i, \quad \forall i \in [N]. \tag{5}$$

3.2.2 Learning-based Auction. To ensure incentive compatibility (IC) in our model, we adopt the concept of *ex-post regret* [7, 41] to quantify the potential gain an advertiser could obtain by untruthfully reporting their bid. This formulation enables us to enforce IC constraints in a differentiable manner, suitable for end-to-end optimization.

Formally, given the generated sequence $\mathcal{Y}$, ad $x_i \in \mathcal{Y}$ with true valuation v_i and contextual user input u, the ex-post regret is

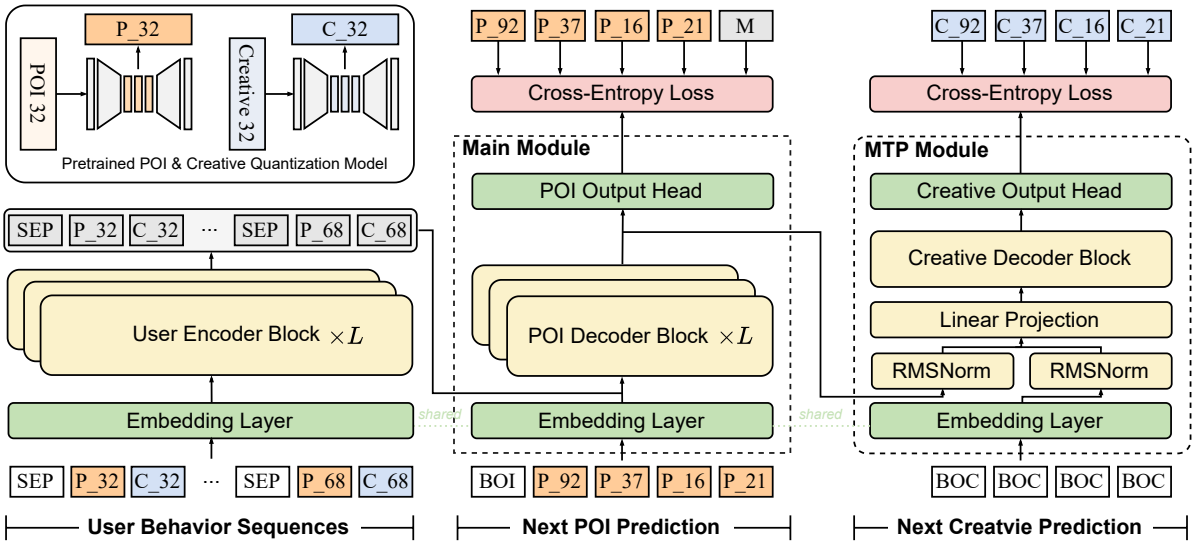

Figure 2: Overview of the interest-based pre-training architecture. The pre-training model consists of encoder for modeling historical behavior sequences, followed by a two-stage decoder: a POI decoder for next POI token prediction and a creative decoder for multi-modal next creative prediction. Both decoders are trained jointly using cross-entropy loss in MTP framework.

defined as:

$$\mathrm{rgt}_i(v_i, \mathcal{Y}, u) = \max_{b_i'} \left\{ u_i(v_i; b_i', \boldsymbol{b}_{-i}, \mathcal{Y}, u) - u_i(v_i; b_i, \boldsymbol{b}_{-i}, \mathcal{Y}, u) \right\}, \tag{6}$$

where b_i is the truthful bid, b_i' is a potential misreport, and $\boldsymbol{b}_{-i}$ represents bids excluding the item x_i. The IC constraint is satisfied if and only if $\mathrm{rgt}_i = 0$ for all advertisers. In practice, we approximate this using N_v sampled valuations from distribution $\mathbb{F}$, the empirical ex-post regret for ad x_i is

$$\widehat{\mathrm{rgt}}_i = \frac{1}{N_v} \sum_{j=1}^{N_v} \mathrm{rgt}_i(v_i^j, \mathcal{Y}, u). \tag{7}$$

We then formulate the auction design problem as minimizing the expected negative revenue under the constraint that the empirical ex-post regret remains zero for each ad x_i:

$$\min_{\boldsymbol{w}} \; -\mathbb{E}_{\boldsymbol{v}\sim\mathbb{F}} \left[\mathrm{Rev}^{\mathcal{M}} \right], \quad \text{s.t.} \quad \widehat{\mathrm{rgt}}_i = 0, \; \forall i \in [N], \tag{8}$$

4 Methodology

We introduce EGA, an end-to-end generative advertising framework. This section describes how we learn user interests via vector tokenization and encoder-decoder design, leverage permutation-aware reward modeling for business objectives, incorporate auction-based preference alignment, and integrate all components through a unified multi-phase optimization strategy.

4.1 Interest-based Pre-training

The goal of the pre-training stage is to capture user interests based on their full historical behaviors, including both ads and organic contents. This stage serves as the foundation for learning a unified representation that reflects the interests of the users. Mathematically, the objective of the pre-training model $\mathcal{F}$ is to generate a interest-aware output sequence $\mathcal{Y}$ conditioned on the input user behavior sequence $\mathcal{S}^u$:

$$\mathcal{Y} := \mathcal{F}(\mathcal{S}^u). \tag{9}$$

4.1.1 Feature Representations. We represent the user-side context using $\mathcal{S}^u = \{y_1, y_2, \ldots, y_B\}$, where each y_i is a previously interacted item (e.g., click, purchase, like), and B is the sequence length. The target label $\mathcal{Y} = \{y_1, y_2, \ldots, y_K\}$ corresponds to high-value items actually exposed in the current PV session. Each candidate item y_i is described by a multi-modal representation that includes Point of Interest (POI) feature embeddings $\boldsymbol{e}_i^{poi}$ (e.g., sparse ID features and dense features), and creative image embeddings $\boldsymbol{e}_i^{img}$ extracted from visual content. The final input can be represented as $\mathcal{S}^u = \{(\boldsymbol{e}_1^{poi}, \boldsymbol{e}_1^{img}), (\boldsymbol{e}_2^{poi}, \boldsymbol{e}_2^{img}), ..., (\boldsymbol{e}_B^{poi}, \boldsymbol{e}_B^{img})\}$.

4.1.2 Vector Tokenization. Inspired by existing generative recommendation models [4, 18, 20, 31], we employ Residual Quantized Variational Autoencoder (RQ-VAE) [33] to encode dense embeddings into semantic tokens. Each user behavior in the historical sequence is represented as a POI-creative pair:

$$\mathcal{S}^u = \{(\boldsymbol{a}_1^{poi}, \boldsymbol{a}_1^{img}), (\boldsymbol{a}_2^{poi}, \boldsymbol{a}_2^{img}), ..., (\boldsymbol{a}_B^{poi}, \boldsymbol{a}_B^{img})\}. \tag{10}$$

and the prediction target is a similarly structured sequence:

$$\mathcal{Y} = \{(\boldsymbol{a}_1^{poi}, \boldsymbol{a}_1^{img}), (\boldsymbol{a}_2^{poi}, \boldsymbol{a}_2^{img}), ..., (\boldsymbol{a}_K^{poi}, \boldsymbol{a}_K^{img})\}. \tag{11}$$

Each pair of tokens $(\boldsymbol{a}_i^{poi}, \boldsymbol{a}_i^{img})$ combines high-level categorical intent and fine-grained visual semantics. For simplicity, we assume that both POI and creative image are each represented by a single-level token $\boldsymbol{a}$, though the framework can be extended to hierarchical token representations if needed. For example, given the codebooks $\mathcal{V}$ that consists of C layers, each of size K, the token $\boldsymbol{a}_i$ is denoted

as

$$\boldsymbol{a}_i = (a_i^1, a_i^2, ..., a_i^C), \quad a_i^j \in \{\mathcal{V}_{j,1}, \mathcal{V}_{j,2}, ..., \mathcal{V}_{j,K}\}. \tag{12}$$

4.1.3 Probabilistic Decomposition. The modeling of the target item's probability distribution is decomposed into two stages, leveraging the complementary advantages of POI-level and creative-level representations. Rather than directly predicting the next item $\boldsymbol{a}_{t+1}$ from the historical interaction sequence $\mathcal{S}^u_{1:t}$, EGA first predicts the POI $\boldsymbol{a}^{poi}_{t+1}$, then determines the creative image $\boldsymbol{a}^{img}_{t+1}$.

$$P(\boldsymbol{a}^{poi}_{t+1}, \boldsymbol{a}^{img}_{t+1} \mid \mathcal{S}^u_{1:t}) = P(\boldsymbol{a}^{poi}_{t+1} \mid \mathcal{S}^u_{1:t}) \cdot P(\boldsymbol{a}^{img}_{t+1} \mid \boldsymbol{a}^{poi}_{t+1}, \mathcal{S}^u_{1:t}) \tag{13}$$

where $P(\boldsymbol{a}^{poi}_{t+1} \mid S^u_{1:t})$ denotes the probability of generating the next POI $\boldsymbol{a}^{poi}_{t+1}$ based on the historical sequence $S^u_{1:t}$, capturing the categorical identity of the next item. Meanwhile, $P(\boldsymbol{a}^{img}_{t+1} \mid \boldsymbol{a}^{poi}_{t+1}, \mathcal{S}^u_{1:t})$ models the probability of generating the creative image $\boldsymbol{a}^{img}_{t+1}$ conditioned on the POI and the history, capturing fine-grained multimodal details of the item.

4.1.4 Encoder-decoder. The overall generative architecture follows an encoder-decoder design aligned with the modular blocks shown in Figure 2. The encoder module first encodes the user interaction sequence $\mathcal{S}^u$ using stacked self-attention and feed-forward layers:

$$\mathcal{S}^e = \text{Encoder}(\mathcal{S}^u), \tag{14}$$

where $\mathcal{S}^e$ represents the latent contextualized embedding of user interests.

The decoder generates the target sequence $\mathcal{Y}$ in an autoregressive manner, conditioned on the encoded user context $\mathcal{S}^e$. Each decoding step t consists of two sub-stages: POI token generation followed by creative token generation. To enable joint learning, inspired by Multi-Token Prediction (MTP) [10, 17], we integrate a MTP module that supervises both heads simultaneously using cross-entropy loss on POI and creative predictions. All modules share the same token embedding space and quantization backbone, facilitating efficient training and consistent generation quality.

At step t+1, the POI decoder predicts the next POI token based on previously generated tokens and the encoded historical sequences:

$$\boldsymbol{a}^{poi}_{t+1} = \text{POI-Decoder}(\mathcal{Y}_{1:t}, \mathcal{S}^e). \tag{15}$$

Then, the creative decoder predicts the corresponding creative token by attending to both the POI prediction and the encoded context:

$$\boldsymbol{a}^{img}_{t+1} = \text{Creative-Decoder}(\mathcal{Y}_{1:t}, \boldsymbol{a}^{poi}_{t+1}, \mathcal{S}^e). \tag{16}$$

The generation of the entire target sequence $\mathcal{Y}$ is modeled as a product of conditional probabilities:

$$P(\mathcal{Y} \mid \mathcal{S}^u) = \prod_{t=1}^{T} P(\boldsymbol{a}^{poi}_t \mid \mathcal{Y}_{1:t-1}, \mathcal{S}^e) \cdot P(\boldsymbol{a}^{img}_t \mid \mathcal{Y}_{1:t-1}, \boldsymbol{a}^{poi}_t, \mathcal{S}^e). \tag{17}$$

4.2 Permutation-aware Reward Model

To ensure that the generated ad sequences align with real user interests, we introduce a permutation-aware reward model (RM) to guide iterative optimization. Unlike NLP tasks where interests signals are typically annotated by humans, the advertising domain benefits from more accurate, point-wise feedback derived from user interactions such as clicks and conversions.

Let $R(\mathcal{Y})$ denote the reward model that estimates reward signals for a candidate target items $\mathcal{Y} = \{\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_K\}$, where each $\boldsymbol{a}_i$ is a generated token. Besides, each generated token does not necessarily correspond to a unique item, which complicates the assignment of supervision signals in reward model. To address this, we enrich each token representation $\boldsymbol{a}_i$ by concatenating raw item embeddings $\boldsymbol{e}^{poi}_i$, which is represented as:

$$\boldsymbol{h}_i = [\boldsymbol{a}_i; \boldsymbol{e}^{poi}_i]. \tag{18}$$

where $[\cdot;\cdot]$ denotes concatenation. The target items becomes $\boldsymbol{h} = \{\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_K\}$.

The target items $\boldsymbol{h}$ are then processed by self-attention layers, which enable interaction among them to capture contextual dependencies and aggregate relevant information across the sequence.

$$\boldsymbol{h}_f = \text{SelfAttention}(\boldsymbol{h}W^Q, \boldsymbol{h}W^K, \boldsymbol{h}W^V). \tag{19}$$

For making fine-grained predictions such as pCTR for POI and creative image respectively, and pCVR, we augment the reward model with multiple task-specific towers:

$$\hat{r}^{\text{pctr}} = \text{Tower}^{\text{pctr}}\left(\sum \boldsymbol{h}_f\right), \quad \hat{r}^{\text{pcvr}} = \text{Tower}^{\text{pcvr}}\left(\sum \boldsymbol{h}_f\right) \tag{20}$$

where $\text{Tower}(\cdot) = \text{Sigmoid}(\text{MLP}(\cdot))$. After obtaining the estimated rewards $\hat{r}^{\text{pcxr}}$ and the corresponding ground-truth labels y^{pcxr} for each item and reward type, we train the reward model by directly minimizing the binary cross-entropy loss. The detailed training procedure is described in the Section 4.4.

4.3 Auction-based Preference Alignment

In the industrial advertising scenario, generative recommendation models need to fulfill not only user interests but also critical business constraints. Specifically, two main business constraints must be simultaneously satisfied: i) advertiser demands for effective exposure, bidding compatibility, and payment consistency; and ii) platform constraints for balancing revenue with user experience, such as controlling ad exposure ratios. To address these challenges, we propose an integrated generative allocation and payment strategy to do preference alignment, consisting of a token-level bidding based allocation module and a decoupled POI-level payment network. The overall structure is illustrated in Figure 3.

4.3.1 Token-level Bidding. After applying RQ-VAE for item tokenization, the original item-level bids no longer align directly with the token space due to a many-to-many mapping between items and latent tokens. Inspired by Google's token-level bidding theory [6], we introduce a token-level allocation mechanism that aggregates bids across items associated with each token. As mentioned in Equation (12), each token $\boldsymbol{a}_i = (a_i^1, a_i^2, ..., a_i^C)$. To maintain differentiation in bids across different tokens, the bid of token a_i^j is $b(a_i^j) = \max(b_1, b_2, ..., b_{N_i})$, where N_i is the number of items associated with token a_i^j. Formally, the aggregated bid weight for token a_i^j is defined as:

$$w(a_i^j) = \left[b(a_i^j)\right]^\alpha + \beta = \left[\max(b_1, b_2, ..., b_{N_i})\right]^\alpha + \beta \tag{21}$$

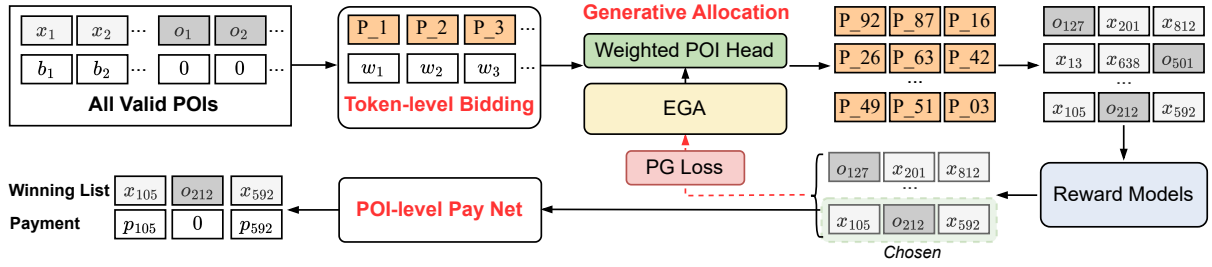

Figure 3: Illustration of the proposed generative ad allocation and payment architecture. Token-level bidding aggregates item bids for generative allocation, while a dedicated POI-level payment network ensures incentive-compatible (IC) constraint. The integrated framework supports dynamic trade-off between revenue and user experience.

where α and β are hyperparameters. By adjusting α, the influence of bids on the allocation process can be flexibly controlled in real-time; meanwhile, dynamically tuning β enables effective balancing of the proportion between ads and organic content in the generated results.

4.3.2 Generative Allocation. The probability of selecting token a_i^j in the generative allocation is given by a softmax normalization:

$$z(a_i^j) = \frac{w(a_i^j) \cdot e^{a_i^j}}{\sum_{k=1}^{W} \left[w(a^{j,k}) \cdot e^{a^{j,k}} \right]}, \tag{22}$$

where $a^{j,k}$ is the k-th code of j-th layer. Based on the generative probabilities $\boldsymbol{z} \in \mathbb{R}^{C \times W}$, where C is the codebook layers and W is the codebook size of each layer, we apply beam search to generate N_S candidate sequences of length K, ensuring both diversity and high-quality selection in the allocation:

$$\mathcal{S}^{(1)}, \mathcal{S}^{(2)}, \ldots, \mathcal{S}^{(N_S)} = \text{BeamSearch}(\boldsymbol{z}, N_S), \tag{23}$$

where $\mathcal{S}^{(j)}$ represents j-th generated candidate sequence of tokens, and N_S denotes both the beam width and the number of generated candidate sequences. We evaluate each sequence using the reward model $\mathcal{R}$ to measure its expected business value, which outputs the reward $\hat{r}_j$ for j-th sequence:

$$\hat{r}_j = \mathcal{R}(S^{(j)}). \tag{24}$$

The final output is the sequence $\mathcal{S}^*$ with the highest reward. It is worth noting that RM provides a flexible interface to accommodate diverse reward signal combinations as needed by the platform.

4.3.3 POI-level Payment Network. While allocation operates at the token level, payment is calculated at the item or POI level, which aligns better with traditional advertiser expectations and business logics. The decoupled payment network is specifically designed to satisfy IC and IR constraints, which computes payments based on item representations.

Formally, the payment network inputs include item representations $\mathcal{S}^* = \{y_1, y_2, ..., y_K\} \in \mathbb{R}^{K \times d}$, the self-exclusion bidding profile $\mathcal{B}^- = \{\boldsymbol{b}_{-1}, \boldsymbol{b}_{-2}, ..., \boldsymbol{b}_{-K}\} \in \mathbb{R}^{K \times (K-1)}$, and the expected value profile $\mathcal{Z} \cdot \Theta \in \mathbb{R}^{K \times 1}$, where $\mathcal{Z} = \{z_1, z_2, ..., z_K\}$ denotes the allocation probability defined in Equation (22) and $\Theta = \{\hat{r}_1^{pctr}, \hat{r}_2^{pctr}, ..., \hat{r}_K^{pctr}\}$ denotes the permutation-aware pCTR estimated by the reward model in Equation (20). The payment rate is defined as:

$$\hat{p} = \sigma(\text{MLP}(\mathcal{S}^*; \mathcal{B}^-; \mathcal{Z} \cdot \Theta)) \in [0, 1]^K, \tag{25}$$

where σ denotes the sigmoid activation to satisfy IR constraint, and the final POI-level payment p is calculated as:

$$p = \hat{p} \odot b. \tag{26}$$

It should be noted that payments are calculated exclusively for ads if $y_i \in X$, while the payment for organic content is zero if $y_i \in O$.

4.4 Optimization and Training

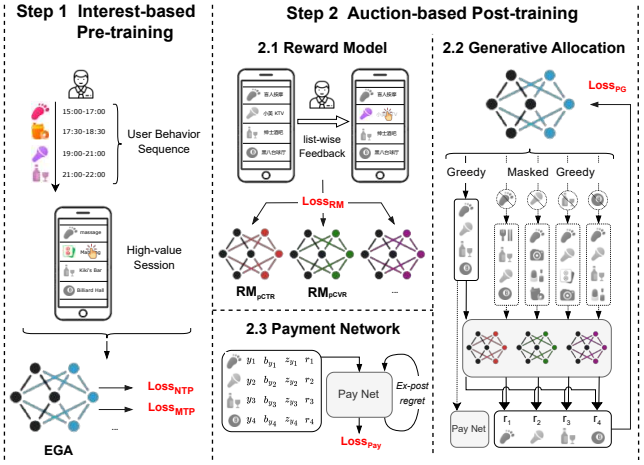

Figure 4: Overview of the proposed optimization and training pipeline. The training process consists of two steps: 1) Interest-based Pre-training, where behavior sequences are used to pre-train EGA that predicts the next POI and creative tokens; and 2) Auction-based Post-training, which includes: 2.1) a permutation-aware reward model trained with list-wise feedback to predict reward signals, such as pCTR and pCVR; 2.2) generative allocation, where different candidate sequences are scored by the reward model to optimize policy gradients; and 2.3) a payment network optimized with ex-post regret to ensure IC approximately.

The overall procedure of training is shown in Figure 4.

4.4.1 Interest-based Pre-training. In the pre-training phase, we first train the generative backbone to capture user interests from historical interaction behaviors. Specifically, we optimize two separate cross-entropy losses for predicted target sequence $\mathcal{Y}$: next-POI prediction loss $\mathcal{L}_{\text{NTP}}$ of main module, and next-creative prediction loss $\mathcal{L}_{\text{MTP}}$ of MTP module. Formally, according to Equation (17),

$$\mathcal{L}_{\text{NTP}} = -\frac{1}{K}\sum_{i=1}^{K} \log P(\boldsymbol{a}_i^{poi} \mid \mathcal{Y}_{1:i-1}, \mathcal{S}^e) \tag{27}$$

$$\mathcal{L}_{\text{MTP}} = -\frac{1}{K}\sum_{i=2}^{K+1} \log P(\boldsymbol{a}_{i-1}^{img} \mid \mathcal{Y}_{1:t-1}, \boldsymbol{a}_i^{poi}, \mathcal{S}^e) \tag{28}$$

Then total pre-training loss is defined as:

$$\mathcal{L}_{\text{pre-train}} = \mathcal{L}_{\text{NTP}} + \mathcal{L}_{\text{MTP}}. \tag{29}$$

4.4.2 Auction-based Post-training. Following the pre-training phase, we freeze its parameters and optimize the generative advertising model under auction constraints through an auction-based post-training stage. This phase aligns the generative outputs with platform revenue objectives and advertiser demands, consisting of two components: i) reward model training, ii) generative allocation training based on policy gradient, and iii) payment network optimization based on Lagrangian method.

Reward Model Training. We train a separate reward model (RM) using users' real feedback signals (e.g., clicks and conversions). The RM is optimized by minimizing the binary cross-entropy loss:

$$\mathcal{L}_{\text{RM}}^{\text{pcxr}} = -\frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}\sum_{i=1}^{K}\left(y^{\text{pcxr}}\log\hat{r}^{\text{pcxr}} + (1-y^{\text{pcxr}})\log(1-\hat{r}^{\text{pcxr}})\right), \tag{30}$$

where y^{pcxr} represents ground-truth labels derived from real user interactions, $\hat{r}^{\text{pcxr}}$ is the predicted probability from the reward model by Equation (20), and $\mathcal{D}$ is the training dataset.

Generative Allocation Training. After convergence of the reward model, we adopt a non-autoregressive policy gradient based method. Given a generated winning ad sequence $\mathcal{S}^* = \{y_1, y_2, ..., y_K\}$, we define the marginal contribution of each item y_i to platform revenue as:

$$r_{y_i} = \sum_{y_j\in\mathcal{S}^*} b_j\hat{r}_j^{\text{pctr}} - \sum_{y_j\in\mathcal{S}_{-i}^*} b_j\hat{r}_j^{\text{pctr}}, \tag{31}$$

where $\mathcal{S}_{-i}^*$ denotes the best alternative ad sequence excluding y_i. We then apply a policy gradient objective to maximize the expected rewards:

$$\mathcal{L}_{\text{PG}} = -\frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}\sum_{y_i\in\mathcal{S}^*} r_{y_i}\log z_{y_i}, \tag{32}$$

where z_{y_i} is the allocation probability for item y_i by Equation (22). This design encourages the generator to produce sequences that yield higher overall revenue, using fixed reward model parameters.

Payment Network Optimization. The payment network optimizes Equation (8) to balance revenue maximization and IC constraint via Lagrangian dual formulation. The loss function integrates both total platform payment and ex-post regret minimization. Given the selected sequence $\mathcal{Y}$, the payment loss $\mathcal{L}_{\text{Pay}}$ is defined as follows:

$$\mathcal{L}_{\text{Pay}} = -\frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}\left(\sum_{y_i\in\mathcal{S}^*} p_i\hat{r}_i^{\text{pctr}} - \sum_{y_i\in\mathcal{S}^*}\lambda_i\widehat{\text{rgt}}_i - \frac{\rho}{2}\sum_{y_i\in\mathcal{S}^*}(\widehat{\text{rgt}}_i)^2\right), \tag{33}$$

where $\widehat{\text{rgt}}_i$ is the ex-post regret of ad y_i, p_i is the predicted payment from the payment network, λ_i is a Lagrange multiplier, and $\rho > 0$ is the hyperparameter for the IC penalty term. To solve this constrained optimization, we adopt an iterative Lagrangian-based approach to jointly optimize the payment network. Specifically, we alternate between two steps:

- Payment Network Update: Optimize the parameters $\boldsymbol{\theta}_{\text{Pay}}$ of the payment network by minimizing the Lagrangian objective with fixed multipliers:

$$\boldsymbol{\theta}_{\text{Pay}}^{\text{new}} = \arg\min_{\boldsymbol{\theta}_{\text{Pay}}} \mathcal{L}_{\text{Pay}}(\boldsymbol{\theta}_{\text{Pay}}^{\text{old}}, \boldsymbol{\lambda}^{\text{old}}). \tag{34}$$

- Multiplier Update: Adjust the Lagrange multipliers based on the observed empirical ex-post regret:

$$\boldsymbol{\lambda}^{\text{new}} = \boldsymbol{\lambda}^{\text{old}} + \rho\cdot\widehat{\text{rgt}}(\boldsymbol{\theta}_{\text{Pay}}^{\text{new}}). \tag{35}$$

Note that the overall objective is non-convex, and convergence to the global optimum is not theoretically guaranteed. However, our empirical results show that this optimization strategy effectively minimizes regret while maintaining near-optimal revenue in real-world scenarios.

5 Experiments

In this section, we evaluate our proposed model on industrial dataset and aim to answer the following research questions:

- **RQ1**: How does our EGA model perform, compared to the state-of-the-art advertising models?
- **RQ2**: What is the impact of designs (e.g. MTP module, token-level bidding, payment network, and multi-phase training) on the performance of EGA?
- **RQ3**: How do hyperparameters affect model performance?
- **RQ4**: How does EGA perform in the real-world industrial advertising scenarios?

5.1 Experiment Setup

5.1.1 Dataset. The industrial dataset used in our experiments consists of real interaction logs collected from a large-scale location-based services (LBS) platform, spanning the period from September 2024 to April 2025. The dataset contains 200 million requests from over 2 million users and nearly 10 million unique ads. We use the first 200 days for pre-training and randomly sample 10% data for preference alignment. The last 14 days are used for testing.

5.1.2 Evaluation Metrics. In offline experiments, we employ the following metrics[2] to comprehensively evaluate platform revenue, user experience, and ex-post regret of advertisers, respectively.

- **Revenue Per Mille:** RPM = $\frac{\sum \text{click}\times\text{payment}}{\sum \text{impression}} \times 1000$.
- **Click-Through Rate:** CTR = $\frac{\sum \text{click}}{\sum \text{impression}}$.

[2]To protect business confidentiality, the reported results on Meituan have been transformed in a way that preserves their statistical properties while ensuring that sensitive business information cannot be inferred or reconstructed from the published data.

- **IC Metric:** $\Psi = \frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}\sum_{i\in k}\frac{\widehat{\text{rgt}}_i^d}{u_i(v_i^d,\boldsymbol{b}^d)}$, where $\widehat{\text{rgt}}_i^d$ denotes the empirical ex-post regret for advertiser i in session data d as defined in Equation (7), and u_i is the realized utility. This metric evaluates incentive compatibility (IC), representing the relative utility gain an advertiser could obtain by manipulating its bid [5, 14, 19]. Following [25, 41], IC is empirically tested via counterfactual perturbation: for each advertiser, the bid b_i is replaced with $\gamma\times b_i$, where $\gamma \in \{0.2\times j \mid j = 1, 2, \ldots, 10\}$.

For offline experiments, evaluation metrics are computed using the predicted values from the reward model.

5.1.3 Baselines. We evaluate EGA against the following two widely adopted industrial architectures.

- **MCA:** The multi-stage cascading architecture (MCA) is a standard paradigm in industrial online advertising. It consists of five key stages: recall, ranking, creative selection, auction, and ad allocation. For a strong baseline, we implement MCA using representative methods: Tiger [20] for recall, HSTU [34] for ranking, Peri-CR [32] for creative selection, CGA [41] for auction, and CrossDQN [16] for ad allocation.
- **GR:** Generative recommendation (GR) formulates recommendation as a sequence generation task, where user-item interactions are modeled using transformer based autoregressive architectures. To apply GR in online advertising scenario, we construct this baseline by integrating OneRec [4] with Peri-CR [32] for creative selection and GSP [8] for payment.

5.1.4 Implementation Details. We train EGA using the Adam optimizer with an initial learning rate of 0.0024. The batch size is set to 128. Model training and optimization are performed on NVIDIA A100 GPUs with 80G memory. For hyperparameters, we tried different hyperparameters using grid search. Due to space constraints, we report only the most optimal hyperparameter settings in this paper. For interest-based pre-training, the block number L of encoder and decoder is 3, the number of codebook layers $C = 3$, and codebook size W of each layer is 1024. We consider $K = 10$ target session items and use $B = 256$ historical behaviors as context. For auction-based preference alignment, the hyperparameters in token-level bidding $\alpha = 1.2$ and $\beta = 2$. We generate $N_S = 64$ different sequences for each request by beam search. For reward model and payment network, the hidden layers of the MLP are 128, 32, and 10.

5.2 Offline Performance (RQ1)

Table 2: The experimental results of out model EGA and competitors on industrial dataset. The bold value marks the best one in each column. Each result is presented in the form of mean (lift percentage). Lift percentage means the improvement of EGA over the best baselines.

Model	RPM	CTR-poi	CTR-img	Ψ
MCA	192.45 (-16.5%)	0.0558 (-8.8%)	0.0529 (-9.3%)	3.6%
GR	206.73 (-10.3%)	0.0582 (-4.9%)	0.0546 (-6.3%)	8.4%
EGA	**230.41**	**0.0612**	**0.0583**	**2.7%**

As shown in Table 2, our key observations are 1) the Multi-stage Cascading Architecture (MCA) suffers from the well-known early-stage filtering problem: promising ads are often eliminated in the initial recall or ranking stages, leading to suboptimal overall performance. This limitation is reflected in its relatively lower RPM and CTR metrics compared to more unified approaches. 2) In contrast, the generative recommendation baseline (GR) is designed to improve sequence modeling and personalization. However, GR still falls short in real advertising scenarios due to its limited ability to satisfy practical business constraints. Specifically, GR does not guarantee incentive compatibility with GSP (as evidenced by a much higher IC regret of 8.4%), cannot flexibly implement dynamic ad allocation or control ad exposure rates, and is unable to support parallel creative selection for each ad. These limitations reduce the effectiveness of GR when applied to industrial advertising.

Our proposed EGA overcomes these shortcomings by integrating end-to-end generation, permutation-aware reward modeling, token-level bidding, and a dedicated payment network. As a result, EGA achieves the best overall performance: it significantly improves revenue (RPM), enhances both POI and creative CTRs, and delivers superior economic robustness with the lowest IC regret among all baselines. This demonstrates that addressing both early-stage candidate loss and business constraints is crucial for practical deployment in industrial advertising systems.

5.3 Ablation Study (RQ2)

Table 3: Ablation Study of EGA.

Model	RPM	CTR-poi	CTR-img	Ψ
EGA	**230.41**	**0.0612**	**0.0583**	**2.7%**
EGA-mtp	225.45 (-2.1%)	0.0593 (-3.1%)	0.0562 (-3.6%)	2.7%
EGA-end	218.21 (-5.3%)	0.0600 (-2.0%)	0.0572 (-1.9%)	3.0%
EGA-bid	222.94 (-3.2%)	0.0603 (-1.5%)	0.0576 (-1.2%)	4.1%
EGA-gsp	226.17 (-1.8%)	0.0608 (-0.6%)	0.0580 (-0.5%)	8.2%

We conduct experiments to validate the effectiveness of different components. Correspondingly, we design a series of ablation studies, which considers four variants to simplify EGA in different ways:

- **EGA-mtp** removes the Multi-Token Prediction (MTP) module, replacing it with a standard next-token prediction (NTP) approach that independently predicts POI and creative in a sequential manner. This setting examines the importance of joint modeling for POI and creative generation.
- **EGA-end** disables the multi-phase training strategy and instead trains the entire model in a single end-to-end stage. The comparison assesses the benefit of our proposed interest-based pre-training and auction-based post-training pipeline.
- **EGA-bid** simplifies the token-level bidding mechanism by replacing the maximum aggregation with an average operation, i.e., each token's bid is computed as the average over all relevant items rather than the maximum. Formally, $b(a_i^j) = \text{avg}(b_1, b_2, ..., b_{N_i})$. This tests the effect of aggregation strategy in the bidding process.

- **EGA-gsp** removes the dedicated payment network and replaces it with standard GSP payment computation, keeping other modules unchanged. The comparison highlights the role of the learned payment network in optimizing revenue and enforcing incentive compatibility.

The performance results are shown in Table 3, from which we observe the full EGA model consistently outperforms all ablation variants across all major metrics, validating the effectiveness of our design. Specifically,

(1) EGA-mtp leads to a noticeable drop in RPM, CTR-poi, and CTR-img (-2.1%, -3.1%, and -3.6% respectively), indicating the importance of jointly modeling POI and creative selection.
(2) The EGA-end variant, which disables multi-phase training, also results in a performance decrease, especially on RPM (-5.3%), highlighting the benefit of our two-stage optimization strategy for aligning user interests modeling and business objectives.
(3) The EGA-bid variant, which replaces the max aggregation in token-level bidding with an average operation, causes a moderate decrease in all metrics, and notably increases the IC regret Ψ to 4.1%, showing the importance of our aggregation choice for incentive compatibility and allocation efficiency.
(4) EGA-gsp results in the highest IC regret (Ψ rises from 2.7% to 8.2%) due to the absence of payment network and degrading to GSP payment, although other metrics drop only slightly. This demonstrates that the payment network is crucial for achieving incentive compatibility approximately in practice.

In summary, each component of EGA, including multi-token prediction, multi-phase training, max aggregation in bidding, and a dedicated payment network, plays an important and complementary role in improving revenue and user experience, and ensuring economic robustness.

5.4 Hyperparameters (RQ3)

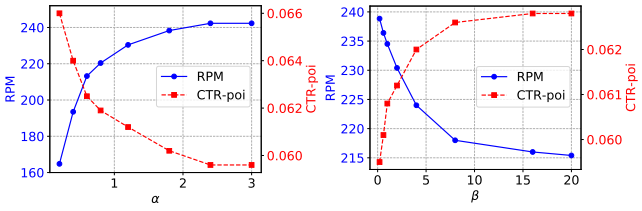

Figure 5: Effect of hyperparameters of EGA.

We conduct a comprehensive study to evaluate the sensitivity of EGA with respect to key hyperparameters. The results are summarized in Figure 5, where RPM and CTR-poi are used as the evaluation metrics. Figure 5 examines the impact of token-level bidding parameters α and β in Equation (21). α measures the importance of bids during the allocation process. Increasing α significantly boosts RPM at first, as higher-bid ads become more likely to win in the generative allocation. However, this trend gradually plateaus as further increasing α provides diminishing marginal returns. In contrast, CTR-poi shows a consistent decline as α rises, reflecting a shift in exposure from more organic contents to higher-bid ads. Besides, β acts as a global balancing factor between ads and organic content. Increasing β reduces the probability of ads being allocated. As a result, RPM decreases steadily with increasing β, and eventually flattens out as ads rarely win positions in the list. Conversely, CTR-poi increases with β, as the display list becomes dominated by organic content. In practice, the optimal values of α and β are selected to achieve a trade-off between platform revenue and user experience.

5.5 Online A/B Test (RQ4)

To validate the real-world effectiveness of our proposed EGA framework, we conducted online A/B tests on an industrial advertising platform, comparing EGA with MCA baseline. Table 4 summarizes the results of online A/B testing conducted from April 2 to April 8, 2025, utilizing 2% of total production traffic. EGA achieves a 15.2% increase in RPM, a 6.4% improvement in CTR, and a 3.1% lift in return on investment (ROI), compared to the MCA baseline. These consistent improvements across revenue, user experience, and advertiser utility highlight the practical value of EGA in a complex and production-scale environment. Besides, benefiting from algorithm engineering and system optimizations, the online response time (RT) of EGA increases by only 7 ms per request (2.5% relatively). This marginal overhead demonstrates that EGA maintains high computational efficiency.

Table 4: The experimental results from Online A/B tests.

Relative change in metrics	RPM	CTR	ROI	RT
EGA over baseline-MCA	+15.2%	+6.4%	+3.1%	+2.5%

6 Conclusion

In this work, we presented End-to-end Generative Advertising (EGA), a novel framework that unifies ranking, creative selection, ad allocation, and payment computation into a single generative model for industrial advertising systems. By leveraging hierarchical semantic tokenization, permutation-aware reward modeling, and token-level bidding and allocation, EGA bridges the gap between user interests modeling and business-critical auction constraints such as IC and IR. The proposed multi-phase training paradigm, including interest-based pre-training and auction-based post-training, ensures that EGA captures both user interests and advertiser utility under complex real-world conditions. Extensive offline experiments and large-scale online A/B tests on an industrial dataset demonstrate that EGA achieves substantial improvements over both multi-stage cascading architecture and recent generative recommendation baselines in platform revenue, user experience, and advertiser return on investment, all with minimal latency overhead. Our ablation studies validate the effectiveness of each design component and highlight the flexibility of the proposed framework.

Looking forward, we believe EGA opens new directions for the integration of generative modeling and economic mechanism design in online advertising. Future work will further explore scaling laws and enhance business interpretability.

References

[1] Yoram Bachrach, Sofia Ceppi, Ian A Kash, Peter Key, and David Kurokawa. 2014. Optimising trade-offs among stakeholders in ad auctions. In *Proceedings of the fifteenth ACM conference on Economics and computation*. 75–92.

[2] Dagui Chen, Qi Yan, Chunjie Chen, Zhenzhe Zheng, Yangsu Liu, Zhenjia Ma, Chuan Yu, Jian Xu, and Bo Zheng. 2022. Hierarchically constrained adaptive ad exposure in feeds. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3003–3012.

[3] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2020. Autoregressive entity retrieval. *arXiv preprint arXiv:2010.00904* (2020).

[4] Jiaxin Deng, Shiyao Wang, Kuo Cai, Lejian Ren, Qigen Hu, Weifeng Ding, Qiang Luo, and Guorui Zhou. 2025. OneRec: Unifying Retrieve and Rank with Generative Recommender and Iterative Preference Alignment. *arXiv preprint arXiv:2502.18965* (2025).

[5] Yuan Deng, Sébastien Lahaie, Vahab Mirrokni, and Song Zuo. 2020. A data-driven metric of incentive compatibility. In *Proceedings of The Web Conference 2020*. 1796–1806.

[6] Paul Duetting, Vahab Mirrokni, Renato Paes Leme, Haifeng Xu, and Song Zuo. 2024. Mechanism design for large language models. In *Proceedings of the ACM Web Conference 2024*. 144–155.

[7] Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David Parkes, and Sai Srivatsa Ravindranath. 2019. Optimal auctions through deep learning. In *International Conference on Machine Learning*. PMLR, 1706–1715.

[8] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. 2007. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American economic review* 97, 1 (2007), 242–259.

[9] Nicola Gatti, Alessandro Lazaric, and Francesco Trovo. 2012. A truthful learning mechanism for contextual multi-slot sponsored search auctions with externalities. In *Proceedings of the 13th ACM Conference on Electronic Commerce*. 605–622.

[10] Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. 2024. Better & faster large language models via multi-token prediction. *arXiv preprint arXiv:2404.19737* (2024).

[11] Siyu Gu and Xiangrong Sheng. 2022. On Ranking Consistency of Pre-ranking Stage. *arXiv preprint arXiv:2205.01289* (2022).

[12] Patrick Hummel and R Preston McAfee. 2014. Position auctions with externalities. In *Web and Internet Economics: 10th International Conference, WINE 2014, Beijing, China, December 14-17, 2014. Proceedings 10*. Springer, 417–422.

[13] Xuejian Li, Ze Wang, Bingqi Zhu, Fei He, Yongkang Wang, and Xingxing Wang. 2024. Deep automated mechanism design for integrating ad auction and allocation in feed. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1211–1220.

[14] Guogang Liao, Xuejian Li, Ze Wang, Fan Yang, Muzhi Guan, Bingqi Zhu, Yongkang Wang, Xingxing Wang, and Dong Wang. 2022. NMA: neural multi-slot auctions with externalities for online advertising. *arXiv preprint arXiv:2205.10018* (2022).

[15] Guogang Liao, Xiaowen Shi, Ze Wang, Xiaoxu Wu, Chuheng Zhang, Yongkang Wang, Xingxing Wang, and Dong Wang. 2022. 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*. 2292–2296.

[16] Guogang Liao, Ze Wang, Xiaoxu Wu, Xiaowen Shi, Chuheng Zhang, Yongkang Wang, Xingxing Wang, and Dong Wang. 2022. Cross DQN: Cross deep Q network for ads allocation in feed. In *Proceedings of the ACM Web Conference 2022*. 401–409.

[17] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437* (2024).

[18] Han Liu, Yinwei Wei, Xuemeng Song, Weili Guan, Yuan-Fang Li, and Liqiang Nie. 2024. Mmgrec: Multimodal generative recommendation with transformer model. *arXiv preprint arXiv:2404.16555* (2024).

[19] Xiangyu Liu, Chuan Yu, Zhilin Zhang, Zhenzhe Zheng, Yu Rong, Hongtao Lv, Da Huo, Yiqing Wang, Dagui Chen, Jian Xu, et al. 2021. Neural auction: End-to-end learning of auction mechanisms for e-commerce advertising. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 3354–3364.

[20] Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, et al. 2023. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems* 36 (2023), 10299–10315.

[21] Zihua Si, Zhongxiang Sun, Jiale Chen, Guozhang Chen, Xiaoxue Zang, Kai Zheng, Yang Song, Xiao Zhang, Jun Xu, and Kun Gai. 2024. Generative Retrieval with Semantic Tree-Structured Identifiers and Contrastive Learning. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*. 154–163.

[22] Juntao Tan, Shuyuan Xu, Wenyue Hua, Yingqiang Ge, Zelong Li, and Yongfeng Zhang. 2024. Idgenrec: Llm-recsys alignment with textual id learning. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 355–364.

[23] Yubao Tang, Ruqing Zhang, Jiafeng Guo, and Maarten de Rijke. 2023. Recent advances in generative information retrieval. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*. 294–297.

[24] Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022. Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems* 35 (2022), 21831–21843.

[25] Yiqing Wang, Xiangyu Liu, Zhenzhe Zheng, Zhilin Zhang, Miao Xu, Chuan Yu, and Fan Wu. 2022. On designing a two-stage auction for online advertising. In *Proceedings of the ACM Web Conference 2022*. 90–99.

[26] Yidan Wang, Zhaochun Ren, Weiwei Sun, Jiyuan Yang, Zhixiang Liang, Xin Chen, Ruobing Xie, Su Yan, Xu Zhang, Pengjie Ren, et al. 2024. Content-Based Collaborative Generation for Recommender Systems. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 2420–2430.

[27] Ze Wang, Guogang Liao, Xiaowen Shi, Xiaoxu Wu, Chuheng Zhang, Yongkang Wang, Xingxing Wang, and Dong Wang. 2022. Learning List-wise Representation in Reinforcement Learning for Ads Allocation with Multiple Auxiliary Tasks. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3555–3564.

[28] Ruobing Xie, Shaoliang Zhang, Rui Wang, Feng Xia, and Leyu Lin. 2021. Hierarchical reinforcement learning for integrated recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4521–4528.

[29] Yue Xu, Qijie Shen, Jianwen Yin, Zengde Deng, Dimin Wang, Hao Chen, Lixiang Lai, Tao Zhuang, and Junfeng Ge. 2023. Multi-channel Integrated Recommendation with Exposure Constraints. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5338–5349.

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

[31] Yuhao Yang, Zhi Ji, Zhaopeng Li, Yi Li, Zhonglin Mo, Yue Ding, Kai Chen, Zijian Zhang, Jie Li, Shuanglong Li, et al. 2025. Sparse Meets Dense: Unified Generative Recommendations with Cascaded Sparse-Dense Representations. *arXiv preprint arXiv:2503.02453* (2025).

[32] Zhiguang Yang, Liufang Sang, Haoran Wang, Wenlong Chen, Lu Wang, Jie He, Changping Peng, Zhangang Lin, Chun Gan, and Jingping Shao. 2024. Parallel ranking of ads and creatives in real-time advertising systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 9278–9286.

[33] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2021. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 30 (2021), 495–507.

[34] Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Michael He, et al. 2024. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. *arXiv preprint arXiv:2402.17152* (2024).

[35] Zhanhao Zhang. 2021. A survey of online auction mechanism design using deep learning approaches. *arXiv preprint arXiv:2110.06880* (2021).

[36] Zhixuan Zhang, Yuheng Huang, Dan Ou, Sen Li, Longbin Li, Qingwen Liu, and Xiaoyi Zeng. 2023. Rethinking the role of pre-ranking in large-scale e-commerce searching system. *arXiv preprint arXiv:2305.13647* (2023).

[37] Zhilin Zhang, Xiangyu Liu, Zhenzhe Zheng, Chenrui Zhang, Miao Xu, Junwei Pan, Chuan Yu, Fan Wu, Jian Xu, and Kun Gai. 2021. Optimizing multiple performance metrics with deep GSP auctions for e-commerce advertising. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 993–1001.

[38] Xiangyu Zhao, Changsheng Gu, Haoshenglun Zhang, Xiwang Yang, Xiaobing Liu, Jiliang Tang, and Hui Liu. 2021. Dear: Deep reinforcement learning for online advertising impression in recommender systems. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 750–758.

[39] Zhishan Zhao, Jingyue Gao, Yu Zhang, Shuguang Han, Siyuan Lou, Xiang-Rong Sheng, Zhe Wang, Han Zhu, Yuning Jiang, Jian Xu, et al. 2023. COPR: Consistency-Oriented Pre-Ranking for Online Advertising. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 4974–4980.

[40] Bowen Zheng, Yupeng Hou, Hongyu Lu, Yu Chen, Wayne Xin Zhao, Ming Chen, and Ji-Rong Wen. 2024. Adapting large language models by integrating collaborative semantics for recommendation. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 1435–1448.

[41] Ruitao Zhu, Yangsu Liu, Dagui Chen, Zhenjia Ma, Chufeng Shi, Zhenzhe Zheng, Jie Zhang, Jian Xu, Bo Zheng, and Fan Wu. 2024. Contextual Generative Auction with Permutation-level Externalities for Online Advertising. *arXiv preprint arXiv:2412.11544* (2024).