ABSTRACT

In the realm of online advertising, automated bidding has become a pivotal tool, enabling advertisers to efficiently capture impression opportunities in real-time. Recently, generative auto-bidding has shown significant promise, offering innovative solutions for effective ad optimization. However, existing offline-trained generative policies lack the near-term foresight required for dynamic markets and usually depend on simulators or external experts for post-training improvement. To overcome these critical limitations, we propose **S**elf-**E**volved **G**enerative **B**idding (SEGB) [1], a framework that plans proactively and refines itself entirely offline. SEGB first synthesizes plausible short-horizon future states to guide each bid, providing the agent with crucial, dynamic foresight. Crucially, it then performs value-guided policy refinement to iteratively discover superior strategies without any external intervention. This self-contained approach uniquely enables robust policy improvement from static data alone. Experiments on the AuctionNet benchmark and a large-scale A/B test validate our approach, demonstrating that SEGB significantly outperforms state-of-the-art baselines. In a large-scale online deployment, it delivered substantial business value, achieving a +10.19% increase in target cost, proving the effectiveness of our advanced planning and evolution paradigm.

SEGB: SELF-EVOLVED GENERATIVE BIDDING WITH LOCAL AUTOREGRESSIVE DIFFUSION

Anonymous authors
Paper under double-blind review

1 INTRODUCTION

With the continuous advancement of digital commerce, online advertising platforms have grown significantly Evans (2009); Huh et al. (2024). Major platforms like Google Google (2021), Alibaba Alibaba (2021), and Facebook Facebook (2021) have developed automated bidding solutions. Advertisers specify marketing objectives and KPIs, while platforms leverage historical and real-time data to estimate CTR and CVR, automatically generating optimal bids. Given the fluid auction environments, auto-bidding is regarded as a long-horizon sequential decision-making process.

Reinforcement learning has emerged for bidding optimization Cai et al. (2017); He et al. (2021); Sutton and Barto (2018). While traditional RL builds on Markov Decision Processes Liu et al. (2023); Ye et al. (2019), recent research challenges this assumption Guo et al. (2024), showing future states can depend on extended historical sequences. Various generative techniques have emerged in offline RL Janner et al. (2021): diffusion-based approaches Ho et al. (2020); Song et al. (2020) model long-range dependencies but can disrupt constraints when applied globally Guo et al. (2024), while return-to-go methods like Decision Transformer Chen et al. (2021) and variants Janner et al. (2021) lack mechanisms for planning with future context. Moreover, offline RL Levine et al. (2020); Figueiredo Prudencio et al. (2024) suffers from limited state-action coverage and restricted exploration beyond static datasets.

To address these challenges, we propose **S**elf-**E**volved **G**enerative **B**idding (SEGB), a synergistic offline framework. The "Self-Evolved" terminology emphasizes that the policy evolves entirely during offline training, reaching an improved state before deployment, rather than requiring online adaptation or continual learning. **Building upon the foundational Autoregressive Diffusion (LAD) paradigm Li et al. (2024),** SEGB employs LAD for high-fidelity state planning, providing

[1] Our source code is publicly available at `https://anonymous.4open.science/r/abde-2D64/README.md`

future-aware context while respecting causal constraints. We integrate this foresight into a Decision Transformer, transforming it from a reactive imitator into a proactive planner. Finally, through Group Relative Policy Optimization (GRPO) Shao et al. (2024), the policy evolves entirely offline, discovering superior strategies beyond the dataset's limitations without requiring simulators or online interaction.

Our main contributions are: (1) An end-to-end SEGB framework that synergistically combines Local Autoregressive Diffusion with future-state-aware RL, enabling both high-fidelity causal planning and proactive decision-making. (2) A GRPO post-training strategy that evolves the policy entirely offline without simulators, discovering superior strategies beyond dataset limitations. (3) Extensive validation through offline experiments and online A/B tests on a real advertising platform, demonstrating strong practicality and scalability.

2 PRELIMINARY

2.1 PROBLEM STATEMENT

In online advertising, advertisers compete for impressions by submitting bids. During a period, N opportunities arrive sequentially. The advertiser who submits the highest bid b_i wins, obtaining value v_i and incurring cost c_i. The objective is to maximize total value $\sum_i x_i v_i$ (where $x_i \in [0,1]$ indicates win probability) subject to budget constraint $\sum_i c_i x_i \le B$ and KPI constraints $\frac{\sum_i c_{ij} x_i}{\sum_i p_{ij} x_i} \le k_j, \forall j$. This yields the constrained optimization:

$$\begin{aligned} \text{maximize} \quad & \sum_i v_i x_i \\ \text{s.t.} \quad & \sum_i c_i x_i \le B \\ & \frac{\sum_i c_{ij} x_i}{\sum_i p_{ij} x_i} \le k_j \quad , \forall j \\ & 0 \le x_i \le 1 \qquad , \forall i \end{aligned} \tag{1}$$

The optimal bid is He et al. (2021): $bid_i^* = \lambda_0 v_i + k_i \sum_{j=1}^{J} \lambda_j p_{ij}$, where λ_j are Lagrange multipliers for constraints.

2.2 AUTO-BIDDING AS SEQUENTIAL DECISION-MAKING

Due to dynamic auction environments, bidding parameters must be adjusted in real time, formulating auto-bidding as a sequential decision problem. At each time t, an agent observes state s_t (budget, delivery time, impressions, costs, CPC/CPA), selects action $a_t = (a_t^{\lambda_0}, \ldots, a_t^{\lambda_J})$ via policy π, and receives reward r_t. A trajectory τ is a sequence of states, actions, and rewards over the campaign duration.

3 THE SEGB METHODOLOGY

To bridge the critical offline-to-online gap in auto-bidding, we propose SEGB as a synergistic, multi-stage framework addressing the core challenges of planning, decision-making, and exploration within a fully offline setting. SEGB consists of three stages: (1) **High-Fidelity State Planning** via Local Autoregressive Diffusion (LAD) that generates causally consistent future state predictions; (2) **Foresight-driven Action Generation** using a Next-State-Aware Decision Transformer that conditions on both long-term goals and immediate future states; and (3) **Offline Policy Evolution** via GRPO fine-tuning guided by an IQL critic, enabling the policy to discover superior strategies beyond the dataset. We now elaborate on each stage.

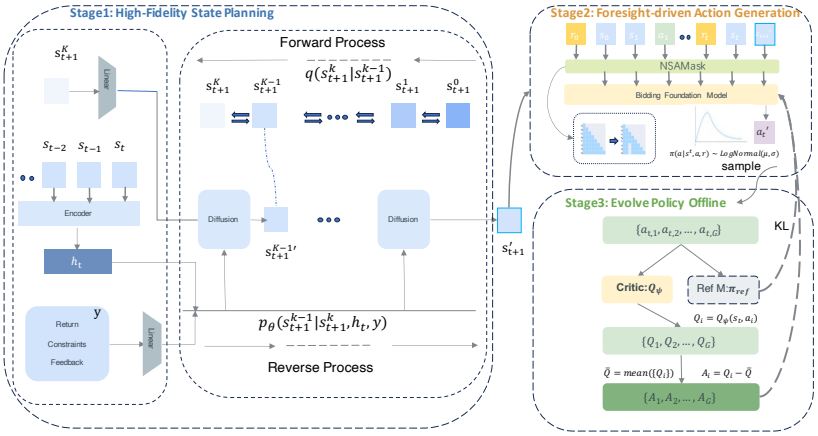


Figure 1: Overview of the SEGB Framework. SEGB consists of three stages. (1) Planning: A LAD model generates a high-fidelity future state prediction (s'_{t+1}). (2) Action Generation: A Next-State-Aware DT conditions on this prediction to generate an action a'_t. (3) Offline Evolution: The policy is then evolved via GRPO, guided by a frozen Critic and Reference Model to update the DT. Note that Stage 3 is only performed during offline training; online inference relies solely on the efficient Stage 1 and Stage 2 pipeline.

3.1 High-Fidelity State Planning via Local Autoregressive Diffusion

We adapt the Local Autoregressive Diffusion (LAD) framework Li et al. (2024) to the auto-bidding domain. Unlike global diffusion models (e.g., DDPM Ho et al. (2020)) which struggle with time-dependent constraints like monotonic budget consumption, LAD generates future states locally and autoregressively, conditioned on historical context. This design enforces temporal causality and domain-specific constraints while providing the high-fidelity foresight required for dynamic bidding environments.

3.1.1 LAD: Model Formulation and Framework

The core idea behind LAD is to model the state generation process autoregressively, thereby preserving temporal dependencies. Formally, we aim to maximize the likelihood of observing the state trajectories in our dataset D. For a given trajectory $\tau = (s_1, s_2, \ldots, s_T)$, this is expressed as:

$$\max_\theta E_{\tau\sim D}\left[\prod_{t=1}^{T} p_\theta(s_t|s_{<t}, y(\tau))\right] \tag{2}$$

where $y(\tau)$ represents campaign-level conditional attributes. This formulation naturally captures the requirement that each state s_t depends on its history $s_{<t}$.

To implement this, LAD applies the diffusion and denoising processes locally to each state s_t while conditioning on the historical context. The reverse denoising process, which generates the state, is conditioned on an embedding of the history, $z_t = f(s_1, \ldots, s_{t-1})$:

$$p_\theta(s_t^{k-1}|s_1, \ldots, s_{t-1}, s_t^k, y(\tau)) = p_\theta(s_t^{k-1}|z_t, s_t^k, y(\tau)) \tag{3}$$

This ensures that the generation of each state explicitly accounts for the preceding states, enforcing causal adherence throughout the trajectory.

3.1.2 Diffusion Training and Inference

LAD applies standard diffusion to each local state s_t. Following DDPM Ho et al. (2020), the forward process adds Gaussian noise: $q(s_t^k \mid s_t^{k-1}) = \mathcal{N}(s_t^k; \sqrt{1-\beta_k}s_t^{k-1}, \beta_k I)$. The reverse process learns to denoise using a neural network ϵ_θ conditioned on history z_t and campaign attributes $y(\tau)$:

$$\mathcal{L}_{\text{LAD}} = E_{(\tau,s_t,\epsilon,k)}\left[\left|\epsilon - \epsilon_\theta(s_t^k, k, z_t, y(\tau))\right|^2\right] \tag{4}$$

For inference, we use classifier-free guidance with strength ω: $\hat{\epsilon}_k = \epsilon_\theta(s_t^k, z_t, k) + \omega(\epsilon_\theta(s_t^k, z_t, y(\tau), k) - \epsilon_\theta(s_t^k, z_t, k))$. The denoised state is computed via $\mu_\theta(s_t^k, k) = \frac{1}{\sqrt{\alpha_k}}(s_t^k - \frac{\beta_k}{\sqrt{1-\bar{\alpha}_k}}\hat{\epsilon}_k)$, iteratively applied to generate the next state prediction $\hat{s}_{t+1}$.

3.2 Foresight-driven Action Generation with a Next-State-Aware DT

Standard Decision Transformers Chen et al. (2021) are reactive, conditioning only on past states and sparse Return-to-Go (RTG) signals. To enable proactive planning, we evolve the DT into a **Next-State-Aware** agent by incorporating the predicted next state $\hat{s}_{t+1}$ from LAD:

$$a_t \sim \pi_\theta(a|s_{\leq t}, a_{<t}, R_{\leq t}, \hat{s}_{t+1}) \tag{5}$$

This creates dual-signal guidance: the long-term RTG provides strategic direction, while $\hat{s}_{t+1}$ offers a dense, immediate target. This is crucial for auto-bidding where rewards are sparse—the predicted next state (e.g., remaining budget) provides a concrete target at every step, enabling proactive constraint management that is difficult to learn from sparse final rewards alone.

3.2.1 Architecture and Supervised Pre-training

The backbone is a GPT-like Transformer with causal self-attention. States, actions, returns, and predicted states are projected into a shared embedding space and fed into the Transformer with positional encodings. Training minimizes the behavioral cloning loss:

$$\mathcal{L}_{\text{DT}}(\theta) = E_{(\tau,\hat{s})\sim D}\left[-\log \pi_\theta(a_t \mid s_{\leq t}, a_{<t}, R_{\leq t}, \hat{s}_{t+1})\right] \tag{6}$$

This yields a strong initial policy that serves as the starting point for offline evolution.

3.3 Offline Policy Evolution

A supervised policy cannot discover novel strategies beyond the training data. We address this **exploration dilemma** through two-step offline optimization: training a reliable IQL critic to guide evolution, then using GRPO to fine-tune the policy.

3.3.1 IQL-based Critic Training

We train a Transformer-based critic using IQL Kostrikov et al. (2021), which avoids OOD action evaluation via expectile regression. The critic conditions on trajectory history: $Q^\pi_\phi(s_t, a_t) = \text{QT}\phi(s_t, a_t; s_{<t}, a_{<t})$. Training minimizes:

$$\mathcal{L}_{\text{IQL}}(\phi, \psi) = E[(Q_\phi(s,a) - (r + \gamma V_\psi(s')))^2] + E[L_2^\tau(Q_\phi(s,a) - V_\psi(s))] \tag{7}$$

where L_2^τ is the expectile loss.

3.3.2 Offline Policy Evolution with GRPO

With the IQL critic in place, we refine π_θ using GRPO. Our hybrid is necessary: IQL alone achieves only 325.89 (vs. DT 335.34) due to unstable policy extraction; GRPO alone requires simulators; alternatives (PPO, DPO) need live environments or preference data. Table 2 validates this synergy: full model achieves 356.0 vs. 346.4 without GRPO (+2.77%).

Stochastic Policy. We model the policy output as a **LogNormal distribution** $\pi_\theta(\cdot) \sim \text{LogNormal}(\mu_\theta, \sigma_\theta)$. To natively support GRPO sampling, Stage 1 pre-training (Eq. 6) directly minimizes the Negative Log-Likelihood (NLL) instead of MSE, jointly learning both the location (μ_θ) and scale (σ_θ) of expert actions.

Motivation: Why IQL-GRPO Hybrid? Our hybrid approach addresses limitations of either method alone. IQL learns robust Q-functions via expectile regression Kostrikov et al. (2021), but direct policy extraction is unstable in continuous action spaces—empirically, IQL alone achieves only 325.89 on AuctionNet, underperforming DT (335.34). Conversely, GRPO Shao et al. (2024) requires reliable value signals typically from simulators, which suffer compounding errors over our 48-step horizon. Alternative approaches are also unsuitable: supervised fine-tuning cannot synthesize novel actions; online methods like PPO Schulman et al. (2017) require live environments; preference-based methods like DPO Rafailov et al. (2023) need explicit preference data unavailable in offline logs. Our synergy uses IQL for stable offline value estimation and GRPO for guided policy optimization. Empirically, Table 2 validates this: the full model achieves 356.0 vs. 346.4 without GRPO (+2.77%) and 325.9 for IQL alone (+9.2%).

Algorithm 1 Training of SEGB

Input: Randomly initialized planner θ_{LAD}, policy θ_π, critic ϕ, ψ; bidding trajectory dataset $\mathcal{D}$.
Output: Optimized policy $\theta_\pi^{\text{final}}$.

while not converged on supervised objectives **do**
  Sample a batch of trajectories $\mathcal{B}$ from $\mathcal{D}$;
  for all $\tau \in \mathcal{B}$ **do**
    Sample $k \sim \text{Uniform}(1, K)$, $\epsilon \sim \mathcal{N}(0, I)$;
    Compute $x_k(\tau)$ via the forward process $q(x_k(\tau)|x_0(\tau))$;
    Compute $\mathcal{L}(\theta_{LAD}, \theta_\pi)$ by Eq (10);
    Perform gradient descent to optimize θ_{LAD} and θ_π;
  end for
end while
Let $\theta_\pi^{\text{pre-trained}} \leftarrow \theta_\pi$;

while not converged on IQL objective **do**
  Sample a batch of trajectories $\mathcal{B}$ from $\mathcal{D}$;
  for all $\tau \in \mathcal{B}$ **do**
    Compute $\mathcal{L}(\theta_{LAD}, \theta_\pi)$ by Eq (7);
    Perform gradient descent to optimize ϕ and ψ;
  end for
end while
Freeze critic ϕ, ψ;
while not converged on GRPO objective **do**
  Sample a batch of trajectories $\mathcal{B}$ from $\mathcal{D}$;
  for all $\tau \in \mathcal{B}$ **do**
    Compute policy evolution objective J_{GRPO} by Eq (9) using the frozen critic Q_ϕ;
    Update policy parameters θ_π (initialized from $\theta_\pi^{pre-trained}$);
  end for
end while
Let $\theta_\pi^{\text{final}} \leftarrow \theta_\pi$;
return $\theta_\pi^{\text{final}}$.

GRPO Post-training. GRPO maximizes a clipped surrogate objective using importance sampling ratio $r_i(\theta) = \frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}$ and advantage estimates $\hat{A}_i$ from our critic:

$$\mathcal{L}_i^{\text{CLIP}}(\theta) = \min\Big(r_i(\theta)\hat{A}_i,\ \text{clip}\big(r_i(\theta), 1-\varepsilon, 1+\varepsilon\big)\hat{A}_i\Big) \tag{8}$$

Finally, the full per-sample objective, $\mathcal{L}^{\text{GRPO}}$, incorporates a KL-divergence penalty to regularize the policy and prevent it from deviating too far from a trusted reference policy π_{ref}:

$$\mathcal{L}^{\text{GRPO}}(\dots) = \frac{1}{G}\sum_{i=1}^{G}\left(\mathcal{L}_i^{\text{CLIP}}(\theta) - \beta \cdot D_{KL}\big[\pi_\theta(\cdot \mid q) \,\|\, \pi_{\text{ref}}(\cdot \mid q)\big]\right) \tag{9}$$

3.4 SEGB TRAINING

The training of SEGB follows a powerful two-stage paradigm: supervised pre-training to build a strong policy foundation, followed by offline reinforcement learning for policy evolution. In the first stage, we focus entirely on supervised learning to create a foresight-aware initial policy. This is guided by a unified objective, $\mathcal{L}_{\text{supervised}}$, which combines the diffusion loss for the planner and the behavioral cloning loss for the policy:

$$\mathcal{L}_{\text{supervised}} = \mathcal{L}_{\text{LAD}} + \mathcal{L}_{\text{DT}} \tag{10}$$

Optimizing this objective endows the agent with a strong ability to imitate expert behavior based on future plans, all without any value estimation. The second stage enables the agent to evolve beyond simple imitation. This reinforcement learning phase begins by preparing a reliable value guide: we first train a robust Q-function critic by minimizing the IQL expectile loss, $\mathcal{L}_{\text{IQL}}$. Then, with this pre-trained critic frozen to provide stable value estimates, the policy from Stage 1 is fine-tuned by maximizing the GRPO policy improvement objective, $J_{\text{GRPO-bid}}$. This two-step "prepare guide, then evolve" process allows SEGB to safely discover superior strategies in a fully offline manner. The entire structured procedure is summarized in Algorithm 1.

4 EXPERIMENTS

We conduct extensive experiments to validate SEGB, addressing: **RQ1** (Overall Performance)—does SEGB outperform baselines across datasets and budgets? **RQ2** (Component Contribution)—how do LAD and GRPO each contribute? **RQ3** (Real-World Efficacy)—do offline gains translate to live production improvements?

4.1 EXPERIMENTAL SETUP

4.1.1 DATASET

We adopt AuctionNet Su et al. (2024), a large-scale simulated bidding benchmark by Alibaba. This dataset includes: (1) AuctionNet, with complete bidding trajectories, and (2) AuctionNet-Sparse, a sparser version with fewer conversions. Both contain 500K trajectories from 10K episodes, each spanning 48 time steps. Detailed statistics are in Appendix Table 4.

4.1.2 EVALUATION METRICS

Following AuctionNet, we use the score metric: $score = \sum_i (o_i v_i) \cdot \min\{(C/CPA)^\beta, 1\}, \beta = 2$, balancing value maximization and KPI constraints. We employ rotation-based testing: each model replaces the 48 agents sequentially. For each rotation, we run 30 initializations and average the top-5 scores.

4.1.3 BASELINES

We compare against representative baselines: **IQL** Kostrikov et al. (2021) achieves policy iteration via expectile regression Kostrikov et al. (2021) without evaluating out-of-distribution actions; **BCQ** Fujimoto et al. (2019) constrains policies to behavior-close actions using VAE Kingma and Welling (2022); **CQL** Kumar et al. (2020) learns conservative Q-functions by penalizing OOD actions to prevent overestimation; **DiffBid** Guo et al. (2024) uses conditional diffusion models Ho et al. (2020) for trajectory generation with inverse dynamics Agrawal et al. (2016); Pathak et al. (2018); **DT** Chen et al. (2021) is a Transformer-based Vaswani et al. (2017) sequence model; **GAS** Li et al. (2025) employs Monte Carlo Tree Search Kocsis and Szepesvári (2006) at inference.

4.1.4 IMPLEMENTATION DETAILS

Experiments use PyTorch on NVIDIA A100 GPUs. Hyperparameters are in Appendix Table 5. Key configurations: LAD has 8 layers, 16 heads, 512-d embedding, $R = 38$ diffusion steps, $\omega = 0.2$ guidance, trained with AdamW (lr=1×10^{-5}). Next-State-Aware DT has 6 layers, 8 heads, context=28, with LayerNorm and GELU. The IQL critic uses $\tau = 0.8$ expectile, and GRPO uses $\beta = 0.1$ KL penalty, both trained with lr=3×10^{-5}.

Computational Efficiency. Offline training (LAD, NSA-DT, IQL, GRPO) takes 4 hours on two A100 GPUs (one-time cost). Online inference achieves P99 latency <0.0375s, meeting the <100ms real-time constraint. GRPO adds zero online cost.

4.2 OVERALL PERFORMANCE COMPARISON (RQ1)

Table 1 shows SEGB consistently outperforms all baselines across both datasets and all budgets. Key findings:

(1) **SEGB achieves state-of-the-art performance** in all settings, with improvements ranging from 1.65% to 12.25% over the best baseline, demonstrating the overall superiority of our synergistic framework.

(2) **The performance gap widens on AuctionNet-Sparse**, validating that LAD's dense next-state prediction provides crucial guidance when long-term rewards are scarce. In sparse reward scenarios, the explicit short-term target from $\hat{s}_{t+1}$ becomes even more valuable.

(3) **DiffBid performs poorly**, generating entire trajectories globally. We attribute this to the difficulty of maintaining causal consistency (e.g., budget monotonicity) over long horizons. Its failure

Table 1: Performance comparison on AuctionNet and AuctionNet-Sparse. SEGB consistently and significantly outperforms all baselines. Results are reported as mean score over 5 runs. The * indicates statistical significance ($p < 0.05$) over the best baseline.

Dataset	Budget	BCQ	CQL	DiffBid	IQL	DT	GAS	SEGB	*Improve*
AuctionNet	50%	100.22 ± 2.96	186.92 ± 2.26	33.25 ± 1.52	191.56 ± 2.54	191.36 ± 3.73	$\underline{200.41 \pm 2.96}$	203.71 ± 1.35*	1.65%
	75%	164.45 ± 4.23	256.35 ± 5.10	55.97 ± 2.44	260.68 ± 4.67	271.46 ± 2.42	$\underline{279.31 \pm 5.32}$	285.01 ± 2.39*	2.04%
	100%	204.52 ± 9.11	318.11 ± 2.70	70.44 ± 2.20	325.89 ± 4.16	335.34 ± 3.34	$\underline{347.07 \pm 3.34}$	355.99 ± 2.01*	2.57%
	125%	296.79 ± 6.32	374.10 ± 3.00	100.04 ± 4.09	377.37 ± 3.92	389.44 ± 1.65	$\underline{398.12 \pm 5.88}$	417.88 ± 4.86*	4.96%
	150%	356.19 ± 5.75	420.47 ± 5.40	129.84 ± 5.45	421.46 ± 4.63	436.98 ± 4.74	$\underline{449.78 \pm 5.50}$	462.77 ± 3.72*	2.89%
AuctionNet-Sparse	50%	15.50 ± 0.88	17.90 ± 1.14	5.67 ± 0.39	15.13 ± 0.94	18.94 ± 0.99	$\underline{19.58 \pm 0.77}$	20.82 ± 0.79*	6.33%
	75%	24.24 ± 0.57	25.36 ± 0.77	18.26 ± 1.03	7.15 ± 0.64	25.89 ± 0.93	$\underline{26.48 \pm 1.02}$	28.58 ± 1.00*	7.93%
	100%	30.88 ± 0.69	31.74 ± 1.35	9.34 ± 0.57	19.42 ± 0.88	32.57 ± 1.05	$\underline{33.05 \pm 0.77}$	37.10 ± 0.42*	12.25%
	125%	37.07 ± 0.52	37.69 ± 0.83	10.73 ± 0.79	20.68 ± 1.18	37.13 ± 1.16	$\underline{37.69 \pm 0.82}$	41.53 ± 0.71*	10.19%
	150%	44.29 ± 1.12	44.42 ± 0.58	27.76 ± 0.59	21.62 ± 1.85	42.08 ± 0.33	$\underline{43.02 \pm 1.91}$	47.12 ± 1.51*	9.53%

Table 2: Ablation study on AuctionNet (100% budget).

Model Variant	**Description**	**Score**
SEGB (Full)	Our framework	356.0
w/o LAD	Diffusion + DT + GRPO	341.5
w/o s'	DT + GRPO	345.5
w/o GRPO	LAD + DT	346.4

serves as strong evidence for our LAD design choice, which locally and autoregressively enforces these constraints.

4.3 Ablation Study: Deconstructing SEGB's Success (RQ2)

Table 2 shows ablations on AuctionNet (100% budget). Results reveal each component's critical contribution:

(1) **Removing GRPO** (-9.6 pts) confirms offline evolution's value. Without GRPO, the policy can only imitate the dataset; with it, the policy discovers superior strategies beyond what the offline data explicitly demonstrates.

(2) **Removing foresight** (-10.5 pts) shows explicit future state conditioning is critical. The predicted $\hat{s}_{t+1}$ provides a concrete, immediate target that RTG alone cannot offer.

(3) **Replacing LAD** (-14.5 pts) demonstrates LAD's causal planning is most crucial. Standard diffusion violates temporal constraints, while LAD's local autoregressive design ensures causally consistent trajectories.

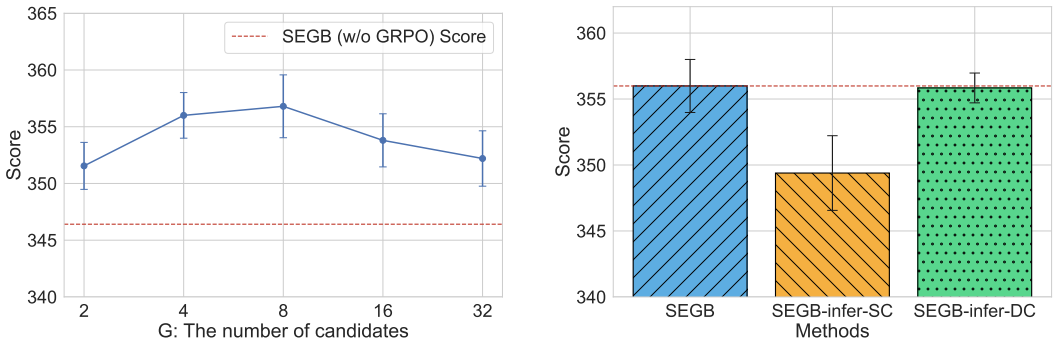
All three components contribute synergistically to achieve state-of-the-art performance.

4.4 Further Analysis: Efficacy of Offline Evolution

To further understand SEGB's offline evolution, we analyze the impact of GRPO group size G in Figure 2a. The results show that $G = 4$ achieves optimal performance. When G is too small (e.g., 2), the policy suffers from insufficient exploration, limiting its ability to discover diverse strategies. When G is too large (e.g., 8, 16), the advantage signal becomes diluted across too many samples, weakening the learning signal. $G = 4$ strikes the best balance between exploration diversity and signal quality, enabling effective policy improvement.

We further analyze the GRPO evolution stage by studying its sensitivity to the group size, G, and justifying our purely offline approach.

Impact of Group Size. The number of candidate actions, G, sampled in GRPO could influence the balance between performance and computational cost. To investigate this, we conduct a sensitivity analysis, with the results shown in Figure 2a. We find that the performance is efficiently enhanced by increasing the number of candidates, with all settings significantly outperforming the baseline without GRPO. Specifically, the score rises sharply to 355.8 at $G = 4$, capturing the vast majority of potential gains. While performance peaks at $G = 8$, the minor improvement does not justify doubling the computational cost. Thus, we chose $G = 4$ as the optimal trade-off between performance and efficiency.



(a) Impact of GRPO Group Size (G)

(b) Impact of Online Voting (K)

Figure 2: Further analysis on key hyperparameters.

Sufficiency of Offline Evolution. Counter-intuitively, augmenting SEGB with a common online exploration technique—critic-based voting—proves detrimental. As visualized in Figure 2b, our final, purely offline model significantly outperforms both online voting variants. The performance collapses when using the same critic as GRPO ("Consistent Critic"), suggesting that online exploration merely disrupts an already-converged optimal policy. These findings provide strong evidence that SEGB's offline evolution is a sufficient and complete optimization process, validating our design of a fully self-contained framework.

4.5 Online A/B Test (RQ3)

To provide the ultimate validation and answer RQ3, we deployed SEGB in a large-scale online A/B test on our advertising platform.

4.5.1 Experimental Setup

Baseline and Deployment. Our baseline is the incumbent production model, a highly-optimized system rooted in Behavior Cloning (BC). This baseline was chosen for its exceptional stability and predictability—critical requirements for a platform handling hundreds of billions of daily requests. It represents a strong industrial benchmark that has been refined over years of production use.

SEGB is deployed in a near-line serving system with GPU acceleration. The system operates in a streamlined two-step process: first, the LAD planner predicts the next state $\hat{s}_{t+1}$ based on history; subsequently, the GRPO-refined policy generates the optimal bid a_t conditioned on this foresight. The model is served via a distributed inference framework that ensures fault tolerance and load balancing across multiple GPU instances.

Multi-Stage Experiment Design. To ensure safe and robust validation, we conducted a multi-stage experiment:

- **Phase 1 (Observation)**: May 28 - June 19, 2025. SEGB was deployed on 20% of budget and traffic to validate stability and initial performance in a controlled setting.
- **Phase 2 (Scale-up)**: June 20 - June 30, 2025. Following successful Phase 1 results, we scaled to 50% of budget and traffic for broader validation.

Consistent performance gains were observed across both phases, confirming SEGB's robustness to varying traffic conditions and budget allocations.

Latency and Performance. The deployed SEGB model achieves a P99 latency of under 0.0375s per request, compared to the baseline's 0.015s. While SEGB introduces additional computational overhead due to the LAD planning step, this latency comfortably meets the platform's stringent <100ms constraint. Notably, GRPO incurs zero online computational cost, as it only refines the policy offline during training.

Table 3: Online A/B test.

Cost	Conversion	ROI	Target Cost
+15.32%	+8.13%	+3.26%	+10.19%

4.5.2 Results and Business Impact

As shown in Table 3, SEGB delivered substantial business impact, achieving a **+10.19%** increase in target cost. This demonstrates that our synergistic, multi-stage architecture successfully bridges the critical offline-to-online gap. It proves that the intelligence learned and refined by SEGB's offline pipeline can generalize and excel in the dynamic, real-world online environment, delivering tangible value.

4.5.3 Robustness to Distribution Shift

We validated OOD robustness through two tests: (1) consistent performance across multi-stage experiments (20% to 50% traffic) confirming robustness to traffic distribution changes, and (2) cold-start campaigns where SEGB achieved +18.03% target cost improvement (vs. +10.19% average), demonstrating strong generalization without campaign-specific fine-tuning.

5 Related Work

Auto-Bidding as Sequential Decision-Making. Auto-bidding optimizes advertisers' strategies by balancing budget constraints and ROI Zhang et al. (2014; 2016). Early approaches used PID controllers Chen et al. (2011) or online linear programming Agarwal et al. (2014), but lacked adaptability. The community increasingly formulated it as sequential decision-making Jin et al. (2018), paving the way for RL-based solutions.

Offline RL for Bidding. RL formulates bidding as learning optimal decisions for long-term rewards Zhao et al. (2018). Given online exploration risks, modern systems use offline RL Levine et al. (2020). To address distributional shift, methods include policy constraints (BCQ Fujimoto et al. (2019)), value regularization (CQL Kumar et al. (2020), EDAC An et al. (2021)), and implicit approaches like IQL Kostrikov et al. (2021) via expectile regression Koenker and Hallock (2001). However, these methods rely on Markov assumptions, while bidding is often non-Markovian Guo et al. (2024).

Generative Approaches. Generative models like VAEs Kingma et al. (2013), GANs Goodfellow et al. (2020), and DDPMs Ho et al. (2020); Kong et al. (2020) have inspired sequence modeling in offline RL. Decision Transformer Chen et al. (2021) pioneered trajectory modeling via Transformers Vaswani et al. (2017). However, DiffBid Guo et al. (2024) violates causal constraints with global diffusion, and DT variants like GAS Li et al. (2025) are reactive. SEGB uniquely combines causal planning via LAD Li et al. (2024), foresight-driven decisions, and offline evolution via GRPO Shao et al. (2024).

6 Conclusion

We proposed SEGB, a synergistic framework addressing the offline-to-online gap in auto-bidding. It integrates Local Autoregressive Diffusion (LAD) for causal state planning, a Next-State-Aware Decision Transformer for foresight-driven actions, and GRPO-based offline policy evolution. Experiments on AuctionNet and a large-scale A/B test (+10.19% target cost) validate SEGB's state-of-the-art performance and real-world business value. Future work includes robustness to non-stationary dynamics and more efficient planning models.

A Dataset Statistics

Table 4 summarizes the key statistics of the AuctionNet and AuctionNet-Sparse datasets used in our experiments.

Table 4: Data statistics for AuctionNet and AuctionNet-Sparse.

Params	AuctionNet	AuctionNet-Sparse
Trajectories	479,376	479,376
Delivery Periods	9,987	9,987
Time steps per trajectory	48	48
State dimension	16	16
Action dimension	1	1
Return-To-Go dimension	1	1
Action range	[0, 493]	[0, 589]
Impression value range	[0, 1]	[0, 1]
CPA range	[6, 12]	[60, 130]
Total conversion range	[0, 1512]	[0, 57]

B Hyperparameter Details

Table 5 provides the complete hyperparameter configuration for all SEGB components.

Table 5: Detailed hyperparameters for SEGB components.

Component	Hyperparameter	Value
LAD Planner	Layers	8
	Attention Heads	16
	Embedding Dimension	512
	Context Length	48
	Diffusion Steps (R)	38
	Guidance Strength (ω)	0.2
	Conditional Dropout Rate	0.2
	Noise Schedule (β_k)	Linear, 10^{-4} to 0.02
Next-State-Aware DT	Layers	6
	Attention Heads	8
	Embedding Dimension	512
	Context Length	28
	Dropout (Attn, Embed)	0.1
	Activation Function	GELU
	Pre-training Batch Size	256
IQL Critic & GRPO Post-training	Critic Architecture	Same as DT
	IQL Expectile (τ)	0.8
	Discount Factor (γ)	0.99
	GRPO KL Penalty (β)	0.1
	GRPO Clipping (ϵ)	0.1
	GRPO Group Size (G)	4
General Optimization	Optimizer	AdamW
	Learning Rate (LAD)	1×10^{-5}
	Learning Rate (DT, IQL)	3×10^{-5}
	Weight Decay	0.01
	Gradient Clipping Norm	1.0

C Implementation Details

In practice, optimizing the training objectives is best achieved via a structured two-step process for stability: we first train the LAD planner to convergence by minimizing $\mathcal{L}_{\text{LAD}}$, and then, with the planner frozen, train the policy by minimizing $\mathcal{L}_{\text{DT}}$. This ensures the policy is trained with a high-quality, stable foresight signal.

References

Deepak Agarwal, Souvik Ghosh, Kai Wei, and Siyu You. 2014. Budget pacing for targeted online advertisements at linkedin. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1613–1619.

Pulkit Agrawal, Ashvin V Nair, Pieter Abbeel, Jitendra Malik, and Sergey Levine. 2016. Learning to poke by poking: Experiential learning of intuitive physics. *Advances in neural information processing systems* 29 (2016).

Alibaba. 2021. Alimama Super Diamond. https://zuanshi.taobao.com/.

Gaon An, Seungyong Moon, Jang-Hyun Kim, and Hyun Oh Song. 2021. Uncertainty-based offline reinforcement learning with diversified q-ensemble. *Advances in neural information processing systems* 34 (2021), 7436–7447.

Han Cai, Kan Ren, Weinan Zhang, Kleanthis Malialis, Jun Wang, Yong Yu, and Defeng Guo. 2017. Real-time bidding by reinforcement learning in display advertising. In *Proceedings of the tenth ACM international conference on web search and data mining*. 661–670.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. 2021. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems* 34 (2021), 15084–15097.

Ye Chen, Pavel Berkhin, Bo Anderson, and Nikhil R Devanur. 2011. Real-time bidding algorithms for performance-based display ad allocation. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1307–1315.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in neural information processing systems*, Vol. 30.

David S Evans. 2009. The online advertising industry: Economics, evolution, and privacy. *Journal of Economic Perspectives* 23, 3 (2009), 37–60.

Facebook. 2021. Advertising on Facebook. https://www.facebook.com/business/ads.

Scott Fujimoto, David Meger, and Doina Precup. 2019. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*. PMLR, 2052–2062.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

Google. 2021. Google Ads. https://ads.google.com/.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300* (2024).

Jiayan Guo, Yusen Huo, Zhilin Zhang, Tianyu Wang, Chuan Yu, Jian Xu, Bo Zheng, and Yan Zhang. 2024. Generative auto-bidding via conditional diffusion modeling. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5038–5049.

Yue He, Xiujun Chen, Di Wu, Junwei Pan, Qing Tan, Chuan Yu, Jian Xu, and Xiaoqiang Zhu. 2021. A unified solution to constrained bidding in online display advertising. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2993–3001.

Diederik P Kingma and Max Welling. 2022. Auto-Encoding Variational Bayes. *arXiv preprint arXiv:1312.6114* (2022).

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

Jisu Huh, Michelle R Nelson, and Cristel Antonia Russell. 2024. Introduction to Computational Advertising Research Methodology Themed Issue. 639–643 pages.

Michael Janner, Qiyang Li, and Sergey Levine. 2021. Offline reinforcement learning as one big sequence modeling problem. *Advances in neural information processing systems* 34 (2021), 1273–1286.

Xingyu Jiang, Ning Gao, Xiuhui Zhang, Hongkun Dou, and Yue Deng. 2025. Value-aligned Behavior Cloning for Offline Reinforcement Learning via Bi-level Optimization. In *International Conference on Learning Representations*.

Junqi Jin, Chengru Song, Han Li, Kun Gai, Jun Wang, and Weinan Zhang. 2018. Real-time bidding with multi-agent reinforcement learning in display advertising. In *Proceedings of the 27th ACM international conference on information and knowledge management*. 2193–2201.

Diederik P Kingma, Max Welling, et al. 2013. Auto-encoding variational bayes.

Levente Kocsis and Csaba Szepesvári. 2006. Bandit based monte-carlo planning. In *European conference on machine learning*. Springer, 282–293.

Roger Koenker and Kevin F Hallock. 2001. Quantile regression. *Journal of economic perspectives* 15, 4 (2001), 143–156.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2020. Score-Based Generative Modeling through Stochastic Differential Equations. *CoRR* abs/2011.13456 (2020).

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. 2020. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761* (2020).

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. 2021. Offline Reinforcement Learning with Implicit Q-Learning. In *International Conference on Learning Representations*.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems* 33 (2020), 1179–1191.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. 2020. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643* (2020).

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. 2024. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems* 37 (2024), 56424–56445.

Yewen Li, Shuai Mao, Jingtong Gao, Nan Jiang, Yunjian Xu, Qingpeng Cai, Fei Pan, Peng Jiang, and Bo An. 2025. GAS: Generative Auto-bidding with Post-training Search. In *Companion Proceedings of the ACM on Web Conference 2025*. 315–324.

Ziru Liu, Jiejie Tian, Qingpeng Cai, Xiangyu Zhao, Jingtong Gao, Shuchang Liu, Dayou Chen, Tonghao He, Dong Zheng, Peng Jiang, et al. 2023. Multi-task recommendations with reinforcement learning. In *Proceedings of the ACM web conference 2023*. 1273–1282.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Deepak Pathak, Parsa Mahmoudieh, Guanghao Luo, Pulkit Agrawal, Dian Chen, Yide Shentu, Evan Shelhamer, Jitendra Malik, Alexei A Efros, and Trevor Darrell. 2018. Zero-shot visual imitation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2050–2053.

Yunpeng Qing, Shunyu Liu, Jingyuan Cong, Kaixuan Chen, Yihe Zhou, and Mingli Song. 2024. A2PO: Towards Effective Offline Reinforcement Learning from an Advantage-aware Perspective. In *Advances in Neural Information Processing Systems*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *Thirty-seventh Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=HPuSIXJaa9

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *CoRR* abs/1707.06347 (2017).

Kefan Su, Yusen Huo, Zhilin Zhang, Shuai Dou, Chuan Yu, Jian Xu, Zongqing Lu, and Bo Zheng. 2024. Auctionnet: A novel benchmark for decision-making in large-scale games. *Advances in Neural Information Processing Systems* 37 (2024), 94428–94452.

Richard S Sutton and Andrew G Barto. 2018. Reinforcement learning: An introduction. *A Bradford Book* (2018).

Rafael Figueiredo Prudencio, Marcos R. O. A. Maximo, and Esther Luna Colombini. 2024. A Survey on Offline Reinforcement Learning: Taxonomy, Review, and Open Problems. *IEEE Transactions on Neural Networks and Learning Systems* 35, 8 (2024), 10237–10257.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

Yujian Ye, Dawei Qiu, Mingyang Sun, Dimitrios Papadaskalopoulos, and Goran Strbac. 2019. Deep reinforcement learning for strategic bidding in electricity markets. *IEEE Transactions on Smart Grid* 11, 2 (2019), 1343–1355.

Weinan Zhang, Yifei Rong, Jun Wang, Tianchi Zhu, and Xiaofan Wang. 2016. Feedback control of real-time display advertising. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. 407–416.

Weinan Zhang, Shuai Yuan, and Jun Wang. 2014. Optimal real-time bidding for display advertising. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1077–1086.

Jun Zhao, Guang Qiu, Ziyu Guan, Wei Zhao, and Xiaofei He. 2018. Deep reinforcement learning for sponsored search real-time bidding. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1021–1030.