Text-to-Image Diffusion Model Selection

Fulvio Nardi Dei da Filicaia Dotti

Delft University of Technology

TU Delft

Text-to-Image Diffusion Model Selection

by

Fulvio Nardi Dei da Filicaia Dotti

Fulvio

Nardi Dei da Filicaia Dotti

Supervisor:	Y. Chen
External Supervisor:	B. Lewandowski
Project Duration:	March, 2025 - March, 2026
Faculty:	Data-Intensive Systems, Delft

Cover:	Digitally generated with DALL·E 3
Style:	TU Delft Report Style, with modifications by Daan Zwaneveld

TU Delft

Preface

I would like to express my sincere gratitude to all those who supported me throughout this journey. Firstly, I would like to show appreciation to my supervisors Dr. Lydia Chen and Basile Lewandowski. They followed me throughout this project despite numerous hurdles and setbacks with remarkable patience and support. I would also like to thank VP for his kind and considerate nature as a teacher who closely followed my progress during the last stages of this project. Finally, I owe my deepest thanks to family and friends. All of which were available to support me unconditionally whenever and if ever needed. Namely, my Father Vincenzo, my Mother Yoko, my Brother Andrea and Shawie, who were pillars of support to whom I am forever grateful for.

Fulvio Nardi Dei da Filicaia Dotti
Delft, March 2026

Introduction

Text-to-image diffusion models have advanced significantly in recent years. Different models show strong performance across various generation tasks. Choosing the right model is becoming increasingly important since no single model consistently outperforms others in all cases. However, existing model selection approaches are typically evaluated only at the dataset level. Such evaluation overlooks prompt-level variation, where different models may excel on different prompts. In this thesis, we investigate diffusion model selection during inference. The goal is to pick the best model for each individual prompt. We first examine the online setting, where model selection occurs adaptively during deployment. In this context, we create a framework for online diffusion model selection and test it against recent methods from the literature. Our findings show that this approach outperforms existing online selection strategies, highlighting the benefits of prompt-aware model selection. In addition to the online setting, we present an offline approach to diffusion model selection, where decisions are made without online interaction. Overall, this thesis claims that diffusion model selection should be viewed as a prompt-level decision rather than a dataset-level comparison. By exploring both online and offline settings and providing empirical results alongside detailed ablations, we aim to promote a more practical and adaptable approach to diffusion model selection.

Contents

Nomenclature

Abbreviations

Abbreviation	Definition
UCB	Upper Confidence Bound
MAB	Multi-Armed Bandit
CMAB	Contextual Multi-Armed Bandit
RHFL	Rich Human Feedback for Text-to-Image Generation
MPS	Multi-dimensional Preference Score
LLM	Large Language Models

Symbols

Notation	Definition
$\mathcal{G}$	Set of models (or possible CMAB options)
G	Number of models (or arms)
t	Timestep incrementing on every new prompt p
$r_{g,t}$	Image reward observed from diffusion model g at timestep t
x	Prompt text features
$X_{g,t}$	Sample from model g at timestep t
R_T	Total cumulative regret over timesteps T
$\mu_g(x)$	Expected reward from model g given prompt features x
K	Number of image quality prediction branches
$\hat{y}_k$	Image quality prediction from branch predictor k
$q_\beta(\cdot)$	$Beta$-th quantile used as UCB for image quality prediction
θ	Hyperparameter Correcting the UCB index
t_g	Total timesteps per models g
$\mathcal{K}_g$	Set of image quality prediction branches of model g
$\mathcal{Y}_{g,t}$	Set of predictions from branches g at timestep t

1

Research Paper

Diffusion Model Selection at Inference through an Ensemble of Predictors: An Online Method

Fulvio Nardi Dei[1], **Basile Lewandowski**[2], **Lydia Chen**[1]

[1]Delft Universityh of Technology, The Netherlands
[2]University of Neuchâtel, Switzerland

fulvio.nardi1@gmail.com, basile.lewandowski@unine.ch

Figure 2. Example of 16 images generated by 4 different diffusion models. For each prompt, a different diffusion model performs best according to the Reinforcement Learning with Human Feedback (RLHF) Scores [1]. Displayed Prompts are from Pic-a-Pick dataset [2].

Abstract -

No two text-to-image diffusion models perform equally well across all prompts. Although existing model selection frameworks benchmark models by their average performance across datasets, this overlooks the prompt-level complementarity between models, where one model may excel for a given prompt while underperforming on another. In this paper, we formulate prompt-level diffusion model selection as a Contextual Multi-Armed Bandit (CMAB) problem and propose an online framework that dynamically selects the best diffusion model for each incoming prompt. More specifically, we design a novel ensemble of predictors built on top of a CLIP encoder, capable of both: predicting image quality for a given model and estimating the uncertainty of that prediction to guide Upper Confidence Bound (UCB) exploration. We validate our approach by demonstrating how the performance of our selecting agent improves as more prompts are observed, eventually outperforming a single strong diffusion model after relatively few prompts from the Pick-a-Pic dataset. We further benchmark and outperform other recent diffusion model selection frameworks using 5 different diffusion models across 3 different image quality metrics. Finally, we verify the effectiveness of each component through various ablation studies

1 Introduction

In recent years, text-to-image diffusion models have garnered a lot of popularity and have seen a lot of advancements. For example diffusion models have seen use in super resolution [3, 4, 5], inpainting [6, 7, 8], restoration [9, 10], domain translation [11] and editing [12]. Further applications of text-to-image Diffusion Models involve data generation [13, 14] or as an artistic output [15, 16].

The common diffusion model selection framework considers selecting the model that outperforms all others given specific benchmark datasets. Notably, CLIPScore [17] is a well known metric that evaluates a text-to-image diffusion models' ability to produce images alignment with its respective input text. With CLIPScore, Diffusion models are benchmarked against each other with the better model considered to be the one that has the higher average CLIPScore throughout all text-image pairs on a specific dataset [18]. However, this approach does not take into account that Diffusion Models can perform better than others for

specific prompts while underperforming on others on some specific image evaluation metric. Consider the examples of prompts shown in figure 2, for each prompt, a different Diffusion Model displays the highest RLHF score while underperforming on others [2]. RLHF is an image evaluation metrics that, similarly to CLIPScore, measures semantic alignment as well as aesthetic human preference score. In that example, PixArt produces the highest RLHF score for the prompt "*a cat in a forest*" while underperforming for the prompt "*Antarctica as drawn by Van Gogh.*"

Therefore, we consider the problem of prompt-level diffusion model selection: given a pool of pretrained diffusion models and a stream of input prompts, select the model that maximizes image quality for each prompt while minimizing cumulative regret over time. We formulate this setting as an online learning problem, where at each round the system observes a prompt, selects a diffusion model from the available pool, and receives feedback based on the generated image quality. The objective is to learn and predict the image quality score of each diffusion model at inference, balancing exploration of different models (for training the predictor) with exploitation of those that perform well (to increase the overall image metric average).

More specifically, we break down the contributions of this paper in the following aspects:

- We introduce a novel ensemble-based exploration–exploitation strategy for online prompt-level diffusion model selection.
- We propose a neural prediction model trained on online streaming data to estimate diffusion model sample quality scores together with uncertainty. The uncertainty of prediction aids to the decision making of exploring or exploiting diffusion model samples.
- We introduce an Ordinary Least Squares (OLS) regression method trained on warm start samples to initialise the weights of our neural prediction models. This method reduces the number of initial sampling rounds that would be required for the neural network to converge to accurate predictions.
- We provide empirical results showing that our strategy outperforms both the best single diffusion model from the selection pool and recent online diffusion model selection strategies.

2 Background and Related Work

Text-to-Image Diffusion Model. Text-to-Image Diffusion models have been established to be the standard approach to synthesising images from text compared to GANs [19], due to their superior image quality and diversity measured through standard image quality metrics such as FID and IS [20]. On a high level, diffusion models synthetize images through a series of denoising steps, starting from a random Gaussian image x_T and ending with a clear image x_0. Each denoising step slightly denoises each image x_t into image x_{t-1} by predicting and removing the noise ϵ through a function parametised as $\epsilon_\theta(x_t, t)$ [20, 21].

In text-to-image generation, the denoising process is conditioned on a textual prompt, typically encoded using a pretrained text encoder and incorporated into the denoising network through cross-attention mechanisms. This conditioning guides the generation process so that the resulting image aligns with the semantic content of the input prompt [22].

Multi Armed Bandit Problem.

The multi-armed bandit problem is a decision making framework in which, given n potential profitable actions (or arms), an agent must balance exploration and exploitation to maximise cumulative reward while minimising the regret incurred from exploring suboptimal actions.

One of the earliest formulations of this problem considers $\mathcal{G}$ gambling machines (or armed bandits), each of which yields a reward $X_{i,t}$ drawn from an unknown distribution with an unknown expectation μ_i, where i is the index of the the machine and t the trial number [23] [24]. Consecutive pulls from these armed bandits would yield rewards $X_{i,1}, X_{i,2}, ...$, which are assumed to be independent and identically distributed (i.i.d.).

An exploratory strategy would consider pulling arms from multiple bandits to estimate the global distribution of rewards μ_i at the cost of exploitation. On the other hand an exploitative strategy would consider frequently pulling the arm of the known most profitable bandit at the risk of missing out on other unknown more profitable bandits.

A key framework commonly employed in MAB to balance between exploration and exploitation is the Upper Confidence Bound (UCB) [25]. The key principle of UCB is to act optimistically in the face of uncertainty by selecting the arm that has the combined highest estimated reward (exploitation) and uncertainty in that estimate (exploration). The UCB framework ensures exploration of uncertain arms showing high estimated upper bound while exploiting those that show less uncertainty but high overall expected reward.

A more sophisticated version of MAB is the CMAB (Contextual Multi Armed Bandit) where, agents use context or feature vectors along with past observations to choose which arm to pull in the current iterations [26]. Applications of CMAB are commonly seen in personalized recommendation systems [27, 28, 29] or reinforcement learning settings [30], where contextual information about users or environments is used to sequentially select actions that maximize cumulative reward while balancing exploration and exploitation.

Image Quality Metrics for Text-to-Image Mod-

els. Diffusion Model performance is often benchmarked through metrics such as CLIPScore [17], IS [31] and FID [32]. CLIPScore evaluates text–image alignment by measuring the cosine similarity between CLIP text and image embeddings, while FID assesses image realism by comparing the feature distribution of generated images to real images using Inception-v3 features. More recent metrics such as ImageReward [33], PickScore [2], and HPDv2 [34] take into consideration aspects such as the aesthetic appeal of an image by using human preference evaluations in their training dataset. Moreover, RLHF [1] and MPS [35] further account for additional factors such as unnatural artefacts and fine-grained alignment between the generated image and the prompt text.

Generative Model Selection. Diffusion model selection is an emerging and relatively unexplored domain. Recent works by Lewandowski et al. [36] propose a novel framework for selecting the most appropriate diffusion model to fine-tune by leveraging image datasets and model features. On the other hand, Zhang et al. [37] propose AdaDiff, a denoising step prediction model that reduces the number of denoising steps for image sampling depending on the prompt difficulty, achieving overall faster sampling with minimal quality drop. Finally, Luo et al. [38] propose Stylus, a retrieval system that finds diffusion model adapters from a database by matching user prompt encodings with adapter metadata encodings.

More well known offline approaches have been explored in other domains such as LLMs in Mixture-of-Experts (MoE) [39]. In MoE, LLM tokens are routed to separate submodels, enabling scalable models with billions of parameters while keeping computation efficient. Similarly, Ding et al. [40] propose a routing algorithm that redirects prompts to pre-trained LLM models to reduce inference costs with minimal loss in sampling quality.

Finally, a branch of Diffusion Model selection in Online Learning settings is emerging through the works of Hu et Al through PAK-UCB and Daux et Al throguh BALROG [41].

PAK-UCB [42] proposes a per-arm kernelized contextual bandit algorithm for prompt-aware online selection of generative models such as text-to-image, text-to-video, image-captioning models, and LLMs. PAK-UCB learns an individual kernel ridge regression estimator and uncertainty for each model and uses UCB framework to balance exploration-exploitation over time.

Similarly, BALROG [41] uses a UCB-based selection framework, where predictions and uncertainties are inferred using a k-NN approach. The expected reward is estimated by averaging the rewards observed for similar prompts in the embedding space. On the other hand, the confidence bonus is estimated from both, statistically unexplored arms and the distance between the current prompt and previously seen prompts.

3 Problem Definition

We formalize the process of online selection of diffusion models as a CMAB problem where an agent selects, evaluates and trains on samples $X_{g,t}$ from a set of diffusion models $\mathcal{G} = \{g_1, \ldots, g_n\}$, where each model corresponds to an arm in a CMAB setting. At each sampling round $t = 1, \ldots, T$ the learner observes context vector $x \in X$ representing the embedding of an input text prompt.

At each round t, the agent selects a diffusion model g observing $r_{g,t} \in \mathbb{R}$ representing the image-quality reward (e.g. CLIPScore, MPS or RLHF) from the generated image.

The expected reward for model g under context x is given by an unknown function:

$$\mu_g(x) = \mathbb{E}\left[r_{g,t} \mid x_t = x\right]$$

The objective is to maximize the cumulative reward over T rounds, or equivalently, to minimize the cumulative regret accumulated by skipping arms yielding the highest reward:

$$R_T = \sum_{t=1}^{T} \left(\max_{g \in \mathcal{G}} \mu_g(x_t) - \mu_{g_t}(x_t) \right)$$

This formulation captures the fundamental exploration–exploitation trade-off: the learner must explore uncertain models to accurately estimate their context-dependent reward functions, while exploiting models believed to yield high image quality for the current prompt.

3.1 Key Challenges

1) Predicting the image reward of Diffusion Models Samples: One key challenge in an online diffusion model selection setting is training a contextual reward prediction model $\mu_g(x_t)$ given model g and prompt input x. Predicting expected reward $\mu_g(x_t)$ is particularly challenging due to the stochastic nature of diffusion-based image generation, where repeated samples from the same model and prompt can yield images with varying quality scores. Moreover, diffusion models often exhibit closely matched performance, requiring highly accurate reward estimation to reliably distinguish the optimal model under a given context.

2) Estimating the UCB of prediction models: Another key challenge in our setting is constructing reliable UCB index for non-linear parametric reward predictors. This is particularly difficult because neural models do not naturally provide calibrated uncertainty estimates. As a result, predictive uncertainty must be carefully approximated to reflect out-of-distribution uncertainty arising from finite observations.

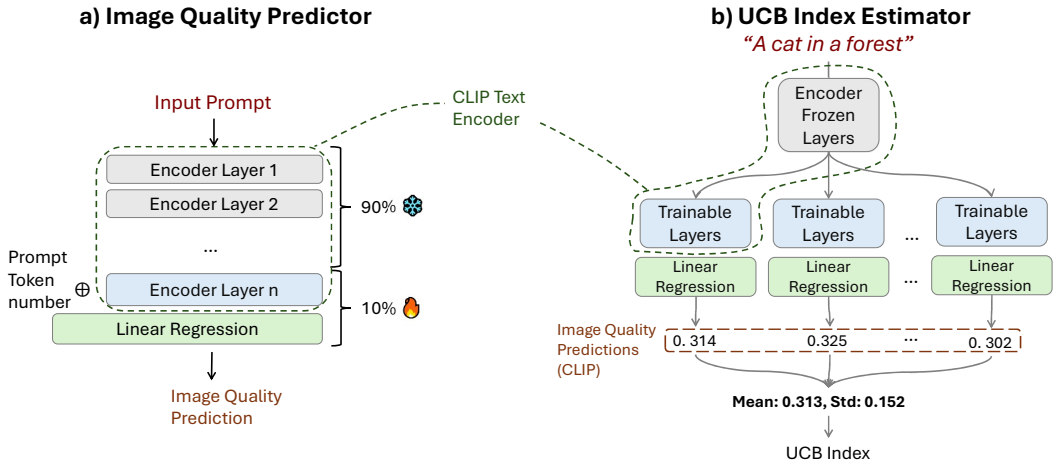


Figure 3. a) Image Quality Prediction model: Given an Input Prompt (e.g. "*A cat in a forest*") the predictor goes through a series of CLIP Text Encoder Layers where the encoded representation is concatenated with the normalised number of tokens in the prompt and passed to a neural network linear regressor. The first 90% of text encoder layers are frozen while the 10% are trained together with the linear regressor. b) UCB Index Estimator: High level visualization for extracting the mean reward and variance for a specific prompt and model regressor pair. The estimator uses K image quality predictor branches to estimate the mean and standard deviation of the input prompt.

4 The Proposed Method

The proposed solution considers a gating network that selects Diffusion Models based on their respective estimate UCB indexes for a text prompt at timestep t. More specifically, the gating network selects and trains a prediction model (arms) based on the sampled image quality reward. Model selection is based on the prompt features that expect the highest estimated UCB index as shown in figure 4. The UCB index itself reflects a balance of exploitation and exploration gain from that image: during early stages UCB index is higher for models that need exploring, on later stages the UCB is higher for models that need exploiting. We further explore in more detail the implementation details of the image reward predictor in section 4.1 addressing key challenge 1 and the UCB estimation in sections 4.2 4.3 addressing key challenge 2.

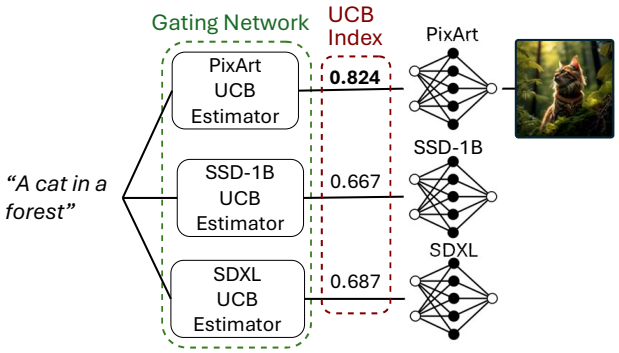


Figure 4. Model selection at inference for prompt "A cat in a forest". The gating network selects PixArt Diffusion Model due to its UCB Estimator measuring the highest UCB Index (0.824) out of the model pool.

4.1 Diffusion Model Image Reward Predictor

We build our diffusion model image reward predictor on top of CLIP-ViT/L14, partially freezing the text-encoder backbone during training. The final layer of the encoder is a linear layer that takes the encoded features concatenated with the normalised token number of the prompt to estimate the final reward value. Empirically, we find that both (i) fine-tuning the CLIP text encoder and (ii) augmenting the embedding with normalized prompt length independently yield a relative RMSE improvement of approximately 10% when predicting image score values on the validation dataset.

4.2 Ensemble of Branches for UCB Estimation

To balance exploration and exploitation when selecting which diffusion model to use for a given prompt, we employ the UCB framework from multi-armed bandit theory. Classical UCB approaches rely on concentration inequalities that require strong distributional assumptions and yield conservative, data-independent bounds. Instead, we leverage an ensemble of reward regressors to obtain data-dependent uncertainty estimates that adapt to the prompt difficulty and prevalence. When presented with a prompt at round t, we obtain K predictions $\{\hat{y}_1, \ldots, \hat{y}_K\}$ from the K branches of model $g \in \mathcal{G}$. The disagreement among these predictions serves as our uncertainty measure where high variances indicate high uncertainty about the

prompt's reward while low variances indicate higher confidence. Due to the computational cost of maintaining an ensemble of full models, we train and sample K independent branches that act as proxies for K distinct prediction models as shown in figure b. Each branch $k \in \mathcal{K}_\}$ shares a common frozen CLIP text encoder as its backbone, but maintains its own set of trainable weights in the upper layers. This design means that while all branches process the same input representation, they produce independently calibrated reward predictions $\hat{y_k}$ for a prompt p and model g. Notably, the variance between predictions of the K branches shrinks the more data is used for training. We use the notion that a high variance in branches $k_i \in \mathcal{K}_g$ correlates with uncertainty in the embedding space to construct our UCB.

Algorithm 1 Model Selection

Require: Model set G, Exploration parameter β, Model Timestep set T, prompt input x, Model branch sets $\mathcal{K}$

```
Best_m ← 0
Highest_UCB ← 0
for all g ∈ G do
    P_list ← {}
    x ← FrozenLayers(x)
    for all k ∈ K_m do
        p ← TrainableLayers(k, x)
        Append p to P_list
    end for
    UCB_m ← EstimateUCB(P_list, x, β, t_m)
    t_m ← t_m + 1
    if UCB_m > Highest_UCB then
        Highest_UCB ← UCB_m
        Best_m ← m
    end if
end for
return Best_m
```

4.3 UCB Index Construction

At round t, we construct the UCB index $I_g(X_t)$ for each model $g \in \mathcal{G}$ to approximate the expected reward. Following the bootstrapped UCB framework of Hao et al. [43], our index combines three components: a mean prediction, an uncertainty estimation, and a finite-sample correction term.

We first obtain the **mean prediction** $\bar{y}_{g,t}$ by averaging the predictions across branches $k_i \in \mathcal{K}_g$ of model g's reward predictors:

$$\bar{y}_{g,t} = \frac{1}{K} \sum_{i=1}^{K} \hat{y}_i \in \mathcal{Y}_{g,t}, \tag{1}$$

where $\{\hat{y}_i\}_{i=1}^{K}$ are the predictions from the K independent branches. This serves as our point estimate of the expected reward.

To quantify the **uncertainty estimation** in our prediction, we compute the bootstrap quantile of the centered branch predictions.

$$q_\beta(\mathcal{Y}_{g,t} - \bar{y}_{g,t}) \tag{2}$$

where $q_\beta(\cdot)$ computes the β-th quantile of the centered branch predictions $\mathcal{Y}_{g,t} - \bar{y}_{g,t}$, and $\beta \in (0, 1)$ is a fixed hyperparameter controlling the degree of optimism. Since the branches actively learn via online updates, the quantile naturally concentrates around the true reward function over time. As branches converge, the centered predictions tighten around zero, causing this term to shrink organically without requiring an explicit decay schedule. When branches disagree (high variance), this quantile is large, encouraging exploration while exploitation is enabled when branches agree (low variance).

Finally, we include a **correction term** to account for finite-sample uncertainty:

$$\theta \sqrt{\frac{\log(t+1)}{t_g}} \tag{3}$$

Where θ controls the factor of this term. This term is governed by two independent mechanisms: the $n_{g,t}$ denominator shrinks the bonus as model g is pulled more frequently, while the $\log(t+1)$ numerator grows slowly with global time, preserving a persistent exploration bonus for rarely pulled arms and ensuring no arm is permanently neglected.

Combining these three components, we obtain the **complete UCB index**:

$$\mathrm{UCB}_g(t) = \underbrace{\bar{y}_{n_{g,t}}}_{\substack{\text{mean} \\ \text{prediction}}} + \underbrace{q_\beta(\mathcal{Y}_{g,t} - \bar{y}_{g,t})}_{\substack{\text{uncertainty} \\ \text{quantile}}} + \underbrace{\theta \sqrt{\frac{\log(t+1)}{t_g}}}_{\substack{\text{correction} \\ \text{term}}}. \tag{4}$$

At each round, we select the model $g_t = \arg\max_{g \in \mathcal{G}} \mathrm{UCB}_g(t)$ with the highest index, balancing exploitation of models with high estimated rewards against exploration of uncertain models.

4.4 OLS-Based Warm Start

We initialize each branch by fitting Ordinary Least Squares (OLS) on disjoint subsets of warmup data to promote early ensemble diversity. Given N_{warmup} prompt–reward pairs $\{(x_j, r_j)\}_{j=1}^{N_{\text{warmup}}}$, we partition them into K subsets of equal size and fit a separate linear model on each subset to predict rewards from encoded representations. The resulting weights initialize the linear head of each branch as shown in algorithm 2. We find that

this approach facilitates training when reward signals span different scales. For example, $I_g^{\text{MPS}}(X_t) \in [0, 20]$ and $I_g^{\text{CLIP}}(X_t) \in [0, 1]$ lie on substantially different ranges, which would otherwise require significant optimization to adapt to during training.

Algorithm 2 Initialize Models with OLS

Require: Model set $\mathcal{G}$, prompt set X, branch sets $\{\mathcal{K}_g\}_{g\in\mathcal{G}}$
Ensure: Initialized linear regressors for all branches
  for all $g \in \mathcal{G}$ **do**
    $K \leftarrow |\mathcal{K}_g|$
    Partition X into K disjoint subsets $\{X_1, \ldots, X_K\}$ of equal size
    for $i = 1$ to K **do**
      Collect reward set R_i for prompts in X_i
      $\hat{\beta}_i \leftarrow (X_i^\top X_i)^{-1} X_i^\top R_i$
      Initialize branch linear regression weights of $k_i \in \mathcal{K}_m$ with weights $\hat{\beta}_i$
    end for
  end for

5 Evaluation

We evaluate our approach using metrics established in prior work [44, 41]:

OutScore (Gap vs Best Single Model). We measure the performance gain over the best static model baseline. Let $g^* = \arg\max_{g\in\mathcal{G}} \frac{1}{T}\sum_{t=1}^{T} r_g(p_t)$ denote the model with the highest average reward across all prompts. The OutScore is:

$$\text{OutScore} = \frac{1}{T}\sum_{t=1}^{T}\left(r_{g_t}(p_t) - r_{g^*}(p_t)\right) \tag{5}$$

where g_t is the model selected at round t. A positive OutScore indicates that adaptive selection outperforms the best fixed model, demonstrating the value of context-dependent model choice.

Regret. We report the average per-round regret, defined as the difference between the reward of the optimal model for each prompt and the selected model's reward:

$$\text{Regret} = \frac{1}{T}\sum_{t=1}^{T}\left(\max_{g\in\mathcal{G}} r_g(p_t) - r_{g_t}(p_t)\right) \tag{6}$$

Lower regret indicates more accurate model selection, where no regret would imply perfect model selection at each $t \in T$.

Optimal Pick Ratio. We track the percentage of rounds where the algorithm selects the optimal model for each prompt, providing a direct measure of selection accuracy.

Dynamic Model Pool. To evaluate adaptability, we expand the model pool during evaluation by introducing SD-Flash at round t = 10,000 and Image-Turbo at round t = 15,000. This tests whether the algorithm can quickly explore new options while maintaining performance on existing models.

5.1 Setup

Dataset. We sample images from the Pick-a-Pic dataset [2], which contains prompts submitted by real users, providing the semantic diversity necessary to evaluate generalization across a wide range of text-to-image generation scenarios. For each evaluation run, we randomly select disjoint subsets of prompts for OLS warm-up and online training from the total 35,000 unique prompts available in the dataset, ensuring no overlap between initialization and evaluation data.

Model Pool. We evaluate our method across six text-to-image diffusion models: SDXL Turbo [45], SSD-1B, Sana1.5 [46], Pixart-α-XL [47], Kolors [48], and Z-Image-Turbo [49]. All models are run using the `diffusers` library[50] with each model's recommended default parameters, ensuring a fair comparison without model-specific tuning.

Hyperparameters. Our reward predictor is initialized using Ordinary Least Squares (OLS) regression on 500 warm-up prompt-reward pairs per model, providing a data-driven starting point before online updates begin. We use the AdamW optimizer with a learning rate of 3×10^{-5} for all online updates. Each UCB estimator maintains $K = 10$ independent branches to quantify predictive uncertainty, with 90% of the CLIP encoder frozen during training to preserve the pretrained text representations while keeping computational cost tractable. The UCB confidence parameter is set to $\beta = 0.95$ across all evaluations.

5.2 Baselines

We evaluate our online learning framework against other online diffusion model selection strategies. Specifically, we measure performance against random selection a polynomial kernel regression of degree 3 adopted in PAK-UCB [44] and against a kNN strategy adopted in Balrog [41]. Both PAK-UCB and Balrog frame the online learning framework of model selection as a contextual multi-armed bandits problem where the agent selects the best diffusion model according to the contextual prompt.

5.3 Results Overview

Figure 5 shows performance across three key metrics over 20,000 rounds. Our bootstrap-UCB method (red) is the only approach to consistently outperform the best sin-

gle model baseline, achieving positive advantage after approximately 2,500 rounds of exploration with CLIPScore.

The instantaneous regret (center) shows our method's learning curve: high initial regret during exploration decreases steadily as the algorithm learns which models perform best for different prompt types. The optimal pick ratio (right) rises from the random baseline of 20% (5 models) to approximately 35%, indicating the algorithm correctly identifies the optimal model for half of all prompts by the end of training.

In contrast, BALROG (blue) and PAK-UCB (orange) fail to outperform the baseline throughout training, with PAK-UCB showing the largest negative gap of approximately −0.018. This suggests that bootstrap-based uncertainty estimation is better suited for this task than the kNN and kernel-based approaches used by BALROG and PAK-UCB, as it more effectively balances exploration of uncertain models with exploitation of known high performers.

Moreover, we plot OutScore to Best for MPS and RLHF respectively in figure 7. Notably we see that BALROG and our approach both achieve a positive OutScore to Best results, indicating the viability of MPS as a metric for model selection.

5.3.1 Dynamic Model Pool Addition

Figure 6 demonstrates the algorithm's adaptability when the model pool expands mid-evaluation. We introduce SD-Flash at round 10,000 and Image-Turbo at round 15,000. As expected, each addition causes a temporary performance drop as the UCB framework explores the new model with high uncertainty. However, the algorithm recovers within approximately 2,500 rounds after each addition, returning to positive advantage.

The regret plot (right) confirms this pattern: sharp spikes at rounds 10,000 and 15,000 correspond to exploration of new models, followed by steady decline as the predictor learns when each new model is appropriate. This recovery demonstrates that the online learning framework can quickly adapt to expanded model pools without catastrophic performance degradation, a crucial property for real-world deployment where new models are regularly released.

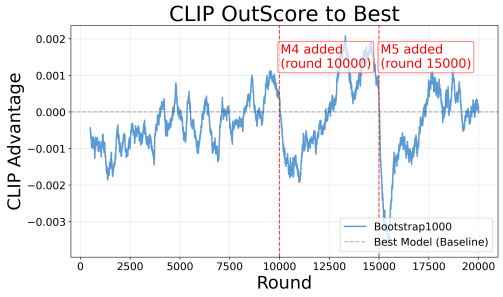
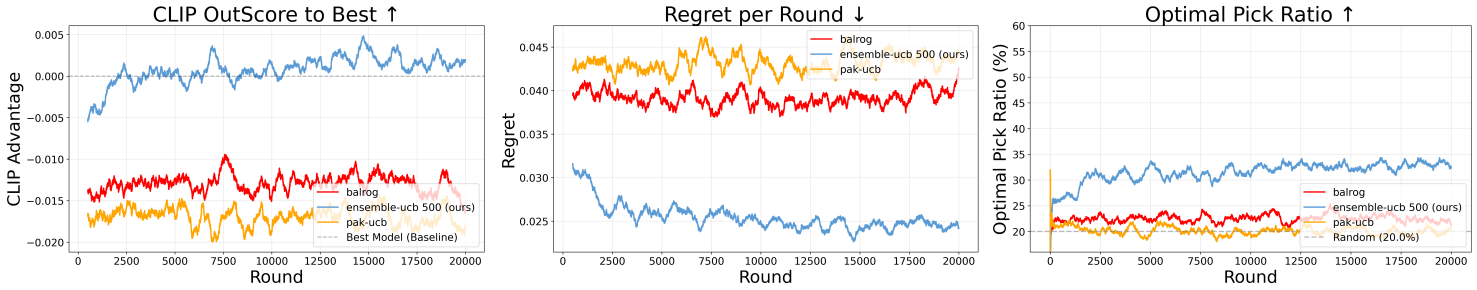


Figure 6. Dynamic model pool expansion. Models 4 and 5 are added at rounds 10,000 and 15,000 respectively, demonstrating exploration through large dips in average scores quickly followed up by positive OutScore to Best.



(a) Advantage over best single model shows positive improvement after exploration phase.

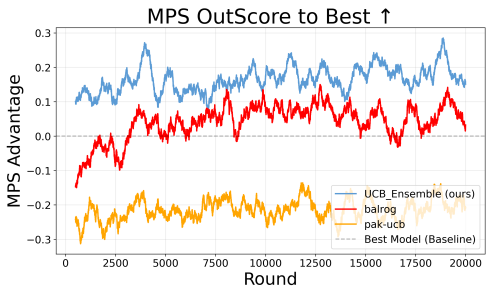
(b) Average regret decreases as the algorithm learns optimal selections.

(c) Optimal pick ratio rises from random baseline (20%) to 35%, demonstrating learning progress.

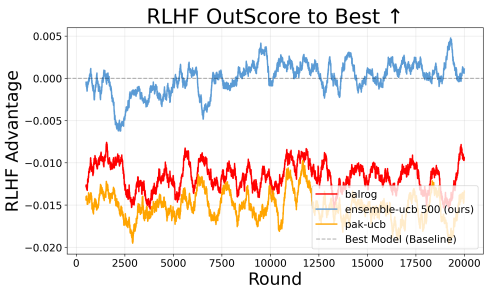
Figure 5. Prompt based selection among 5 diffusion models reporting OutScore to Best, Regret and Optimal Pick Ratio over 20000 prompts for CLIPScore Image Quality Metrics.

Metric	Balrog	Pak-UCB	Ensemble (ours)	Best Model	Random
Avg Score ↑	0.3590 ± 0.0001	0.3551 ± 0.0001	**0.3727 ± 0.0002**	0.3720 ± 0.0003	0.3548 ± 0.0002
Avg Regret ↓	0.0392 ± 0.0001	0.0432 ± 0.0002	**0.0255 ± 0.0002**	0.0000	-
Cumulative Regret ↓	784.04 ± 2.89	863.33 ± 4.78	**510.21 ± 4.06**	0.00	-
Selection Accuracy (%) ↑	22.51 ± 0.21	20.13 ± 0.37	**31.62 ± 0.28**	100.00	-

Table 1. Numerical results from figure 5 showing average score and average regret on the last 500 averaged datapoints, and cumulative regret on the total 20,000 datapoints.



(a) Advantage over best single model shows positive improvement after exploration phase.



(b) Average regret decreases as the algorithm learns optimal selections.

Figure 7. Prompt based selection among 5 diffusion models reporting OutScore to Best over 20000 prompts for MPS and RLHF.

5.4 Ablation Studies

In **Appendix** we explore other warmup and branch number parameters. More specifically, we show how increasing the number of branches or OLS warmup datapoints can increase overall performance. Furthermore, we show the effectiveness of using the normalised token number as one of our features as well as showing the decreasing uncertainty during training.

6 Conclusion

We presented an online framework for prompt-level diffusion model selection that formulates the problem as a Contextual Multi-Armed Bandit and addresses it through an ensemble of CLIP-based reward predictors with bootstrapped uncertainty estimation. Our approach achieves positive advantage over the best single static model after approximately 2,500 rounds, demonstrating that adaptive model selection can effectively exploit prompt-level complementarity among diffusion models.

Our key contributions are threefold. First, we introduced a novel UCB index construction that combines ensemble disagreement with bootstrap quantiles to obtain data-dependent uncertainty estimates without requiring strong distributional assumptions. Second, we introduced fine-tuning CLIP text encoders with OLS warm-start initialization enabling accurate reward prediction across diverse quality metrics (CLIPScore, MPS, RLHF) and diffusion models. Third, we showed that the framework gracefully adapts to dynamic model pools, recovering performance within 2,500 rounds after new model introductions, a critical capability for real-world deployment where models are regularly updated.

Our experimental results across 20,000 prompts from the Pick-a-Pic dataset validate that bootstrap-based uncertainty estimation outperforms kNN and kernel-based approaches (BALROG and PAK-UCB) for this task, achieving final OutScore improvements of $+0.0015 \pm 0.0014$ and optimal pick ratios near 35%. The method's ability to balance exploration and exploitation makes it practical for scenarios where different diffusion models excel at different prompt types, offering a path toward more efficient and higher-quality text-to-image generation.

6.1 Limitations

While our approach demonstrates consistent improvements over static model selection, several limitations warrant discussion. First, the method requires an initial warmup period per model before achieving competitive performance, which may be impractical when computational budgets are severely constrained or when deploying many models simultaneously. Second, our evaluation focuses exclusively on text-to-image diffusion models; the framework's effectiveness for other generative tasks (e.g., text-to-video, text-to-3D) remains unexplored. Third, the ensemble-based uncertainty estimation incurs computational overhead proportional to the number of branches ($K = 10$ in our experiments), though this cost is offset across inference when models are reused for multiple prompts. Finally, our evaluation uses a fixed set of quality metrics (CLIPScore, MPS, RLHF); the framework's sensitivity to metric choice and its performance under multi-objective reward scenarios (e.g., jointly optimizing quality and diversity) remain open questions.

6.2 Future Work

Several promising directions extend this work. First, investigating more efficient warm-start strategies could reduce the initial data requirements, potentially through transfer learning from related model selection tasks or meta-learning across prompt distributions. Second, adapting this method to other domains such as LLM or text-to-video selection. Third, extending the framework to balance hardware budget constraints, combining our approach with model cascades or early-exit strategies could

enable adaptive compute allocation, selecting not only which model to use but also how many diffusion steps to run per prompt. Finally, exploring multi-armed bandit algorithms with lower regret guarantees (e.g., Thompson Sampling variants or information-directed sampling) may further improve sample efficiency.

References

[1] Youwei Liang, Junfeng He, Gang Li, Peizhao Li, Arseniy Klimovskiy, Nicholas Carolan, Jiao Sun, Jordi Pont-Tuset, Sarah Young, Feng Yang, et al. Rich human feedback for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19401–19411, 2024.

[2] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.

[3] Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. *arXiv preprint arXiv:2111.08005*, 2021.

[4] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4713–4726, 2022.

[5] Tao Yang, Rongyuan Wu, Peiran Ren, Xuansong Xie, and Lei Zhang. Pixel-aware stable diffusion for realistic image super-resolution and personalized stylization. In *European conference on computer vision*, pages 74–91. Springer, 2024.

[6] Ling Yang, Jingwei Liu, Shenda Hong, Zhilong Zhang, Zhilin Huang, Zheming Cai, Wentao Zhang, and Bin Cui. Improving diffusion-based image synthesis with context prediction. *Advances in Neural Information Processing Systems*, 36:37636–37656, 2023.

[7] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022.

[8] Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. Smartbrush: Text and shape guided object inpainting with diffusion model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22428–22437, 2023.

[9] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *Advances in neural information processing systems*, 35:23593–23606, 2022.

[10] Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. Diffir: Efficient diffusion model for image restoration. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13095–13105, 2023.

[11] Asha Anoosheh, Eirikur Agustsson, Radu Timofte, and Luc Van Gool. Combogan: Unrestrained scalability for image domain translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 783–790, 2018.

[12] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

[13] Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX security symposium (USENIX Security 23)*, pages 5253–5270, 2023.

[14] Jingyuan Zhu, Shiyu Li, Yuxuan Andy Liu, Jian Yuan, Ping Huang, Jiulong Shan, and Huimin Ma. Odgen: Domain-specific object detection data generation with diffusion models. *Advances in Neural Information Processing Systems*, 37:63599–63633, 2024.

[15] Ling Yang, Zhaochen Yu, Chenlin Meng, Minkai Xu, Stefano Ermon, and Bin Cui. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal llms. In *Forty-first International Conference on Machine Learning*, 2024.

[16] Bohan Zeng, Ling Yang, Jiaming Liu, Minghao Xu, Yuanxing Zhang, Pengfei Wan, Wentao Zhang, and Shuicheng Yan. Editworld: Simulating world dynamics for instruction-following image editing. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 12674–12681, 2025.

[17] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free

evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 7514–7528, 2021.

[18] Xiaoyan Hu, Ho-fung Leung, and Farzan Farnia. A multi-armed bandit approach to online selection and evaluation of generative models. *arXiv preprint arXiv:2406.07451*, 2024.

[19] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1): 53–65, 2018.

[20] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

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

[22] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[23] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[24] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[25] Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *International conference on algorithmic learning theory*, pages 174–188. Springer, 2011.

[26] Tyler Lu, Dávid Pál, and Martin Pál. Contextual multi-armed bandits. In *Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*, pages 485–492. JMLR Workshop and Conference Proceedings, 2010.

[27] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

[28] Djallel Bouneffouf and Irina Rish. A survey on practical applications of multi-armed and contextual bandits. *arXiv preprint arXiv:1904.10040*, 2019.

[29] Jérémie Mary, Romaric Gaudel, and Philippe Preux. Bandits and recommender systems. In *International Workshop on Machine Learning, Optimization and Big Data*, pages 325–336. Springer, 2015.

[30] Tor Lattimore, Csaba Szepesvari, and Gellert Weisz. Learning with good feature representations in bandits and in rl with a generative model. In *International conference on machine learning*, pages 5662–5670. PMLR, 2020.

[31] Shane Barratt and Rishi Sharma. A note on the inception score. *arXiv preprint arXiv:1801.01973*, 2018.

[32] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[33] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

[34] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

[35] Sixian Zhang, Bohan Wang, Junqiang Wu, Yan Li, Tingting Gao, Di Zhang, and Zhongyuan Wang. Learning multi-dimensional human preference for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8018–8027, 2024.

[36] Basile Lewandowski, Robert Birke, and Lydia Y Chen. Match & choose: Model selection framework for fine-tuning text-to-image diffusion models. *arXiv preprint arXiv:2508.10993*, 2025.

[37] Hui Zhang, Zuxuan Wu, Zhen Xing, Jie Shao, and Yu-Gang Jiang. Adadiff: Adaptive step selection for fast diffusion. *arXiv preprint arXiv:2311.14768*, 2023.

[38] Michael Luo, Justin Wong, Brandon Trabucco, Yanping Huang, Joseph E Gonzalez, Zhifeng Chen, Ruslan Salakhutdinov, and Ion Stoica. Stylus: Automatic

adapter selection for diffusion models. *Advances in Neural Information Processing Systems*, 37:32888–32915, 2024.

[39] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[40] Dujian Ding, Ankur Mallick, Shaokun Zhang, Chi Wang, Daniel Madrigal, Mirian Del Carmen Hipolito Garcia, Menglin Xia, Laks VS Lakshmanan, Qingyun Wu, and Victor Rühle. Best-route: Adaptive llm routing with test-time optimal compute. *arXiv preprint arXiv:2506.22716*, 2025.

[41] Jules Damidaux, Basile Lewandowski, Farzan Farnia, and Lydia Y. Chen. BALROG: Contextual bandits meets active learning for online generative model selection, 2026. URL https://openreview.net/forum?id=6a2CJrizrh.

[42] Xiaoyan Hu, Ho-Fung Leung, and Farzan Farnia. Pak-ucb contextual bandit: An online learning approach to prompt-aware selection of generative models and llms. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pages 24447–24481, 2025.

[43] Botao Hao, Yasin Abbasi Yadkori, Zheng Wen, and Guang Cheng. Bootstrapping upper confidence bound. *Advances in neural information processing systems*, 32, 2019.

[44] Xiaoyan Hu, Ho-fung Leung, and Farzan Farnia. An online learning approach to prompt-based selection of generative models and llms. In *Forty-second International Conference on Machine Learning*, 2024.

[45] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[46] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformer, 2024. URL https://arxiv.org/abs/2410.10629.

[47] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023.

[48] Kolors Team. Kolors: Effective training of diffusion model for photorealistic text-to-image synthesis. *arXiv preprint*, 2024.

[49] Tongyi-MAI. Z-image: An efficient image generation foundation model with single-stream diffusion transformer. https://huggingface.co, 2025.

[50] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022.

2

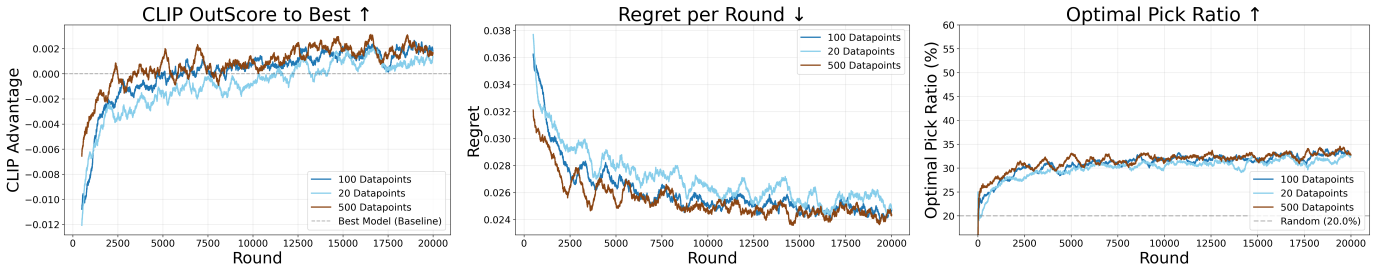
Appendix

2.1. Ablation Study on Ensemble-UCB Components

In this section, we explore a set of ablation studies exploring each component of our proposed method. Namely, we evaluate Ensemble-UCB on other image quality metrics, the OLS-based warmup method, the regression model, and finally we test out different amounts of branch numbers.

2.1.1. Multiple OLS Warmup Set Sizes

We investigate the effect of warmup data size on learning dynamics by comparing initialization with 20, 100, and 500 datapoints. Figure 2.1 shows that larger warmup sets enable faster convergence, the 500-datapoint configuration achieves positive OutScore-to-Best within 2,500 rounds, while the 20-datapoint variant requires approximately 5,000 rounds to reach baseline performance on CLIPScore. However, all three configurations converge to similar long-term performance (gap ≈ 0.002), suggesting that warmup size primarily affects exploration efficiency rather than asymptotic reward. This pattern is confirmed by regret plot per round, decreasing more rapidly for larger warmup sets but stabilizing at comparable levels after 10,000 rounds.



(a) CLIPScore OutScore to Best **(b)** CLIPScore Average Regret per Prompt **(c)** CLIPScore Optimal Pick Ratio

Figure 2.1: Multiple Online Learning metrics to evaluate OLS Warmup Sizes

2.1.2. Online Training Without OLS Warmup

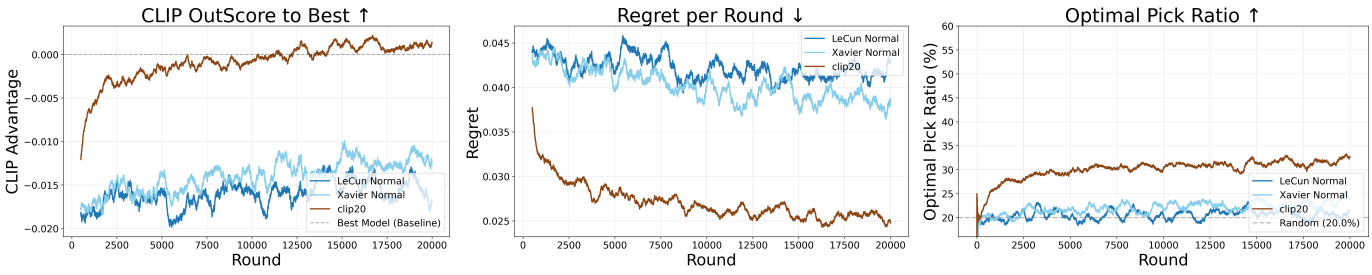
We evaluate the effectiveness of our OLS weight initialisation method by benchmarking online training against other weight initialization methods. Following standard practice in neural network weight initialization, we consider LeCun [7] and Xavier [3] methods as baseline weight initialization schemes. Both methods initialize neural network weights following a normal distribution around 0. More specifically LeCun variance are sampled from:

$$\mathrm{Var}(W) = \frac{1}{n_{\text{in}}}, W_{ij} \sim \mathcal{N}\left(0, \frac{1}{n_{\text{in}}}\right) \tag{2.1}$$

Whereas Xavier layer weights are sampled from:

$$\mathrm{Var}(W) = \frac{2}{n_{\text{in}} + n_{\text{out}}}, W_{ij} \sim \mathcal{N}\left(0, \frac{2}{n_{\text{in}} + n_{\text{out}}}\right) \tag{2.2}$$

Where n_{in} are the number of input features, n_{out} are the number of output features and $W_{i,j}$ are the weights of all neurons i connected to j. To further demonstrate the effectiveness of OLS weight initialisation we show that even 20 warmup samples (2 datapoints per branch) are enough to beat the best model in the pool when training on 20,000 total learning prompts as shown in figure 2.2. Notably, we see that the optimal pick ratio from LeCun [7] and Xavier grow marginally above the random chance (20%) highlighting the importance of non-naive linear layer weight initialization.

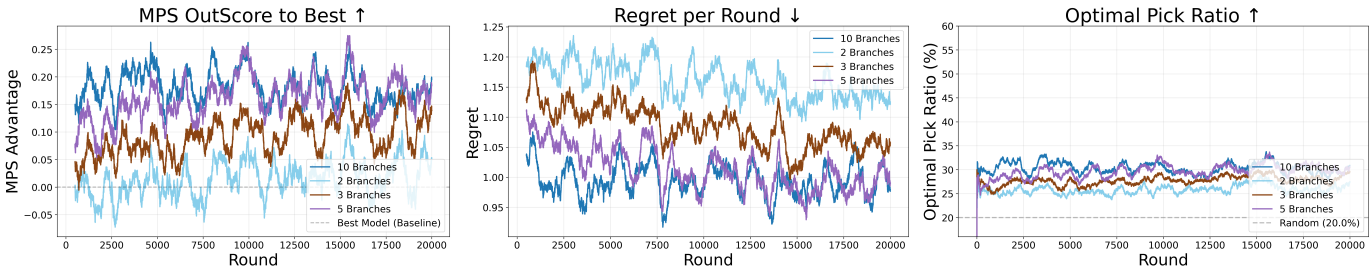


(a) CLIPScore Optimal Pick Ratio **(b)** CLIPScore Average Regret per Prompt **(c)** CLIPScore Optimal Pick Ratio

Figure 2.2: Multiple Online Learning metrics to evaluate OLS Warmup Strategy

2.1.3. Online Model Selection with Different Amounts of Branches

We investigate the effect of ensemble size on exploration quality by varying the number of branches from 2 to 10. Figure 2.3 reveals a clear positive relationship between branch size and performance: configurations with more branches consistently achieve higher OutScore-to-Best, with 10 branches outperforming 2 branches. We reason that these findings are results of (i) more accurate mean reward estimates through variance reduction, and (ii) more reliable UCB quantification through better-sampled bootstrap distributions. The regret curves confirm that 10 branches maintains substantially lower instantaneous regret throughout training compared to 2 or 3 branches but gain diminishing returns compared to 5 branches.

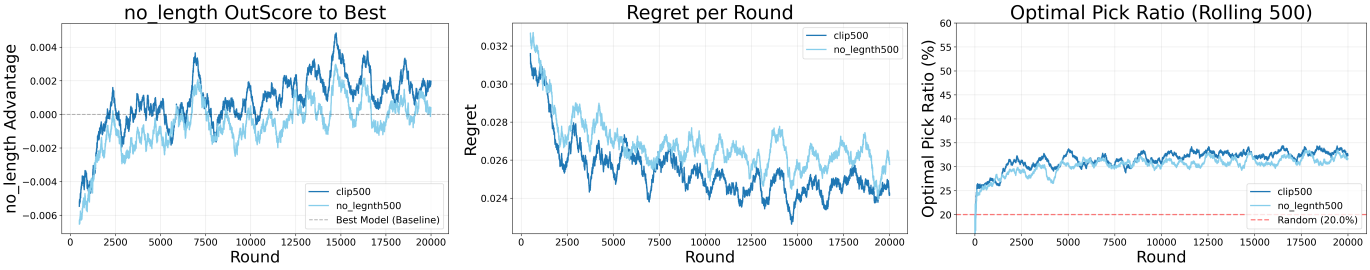


(a) MPS Optimal Pick Ratio **(b)** MPS Average Regret per Prompt **(c)** MPS Optimal Pick Ratio

Figure 2.3: Performance metrics for bootstrapped UCB algorithm

2.1.4. Online Training Without Normalised Token Number

We evaluate the effectiveness of concatenating the normalised token number to prompt encoding by benchmarking online training with and without this feature. We run this experiment on CLIPScore online as shown in figure 2.4, showing that normalised token length is a generally beneficial feature for image-quality prediction, improving overall outscore to best, regret and opr. We reason that token count influences CLIPScore since image generation models struggle to preserve all semantic content from longer, more compositionally complex prompts. Therefore, the image quality predictor assimilates that the longer the text prompt, the worse CLIPScore performs.



(a) CLIPScore OutScore to Best **(b)** CLIPScore Average Regret **(c)** Optimal Pick Ratio

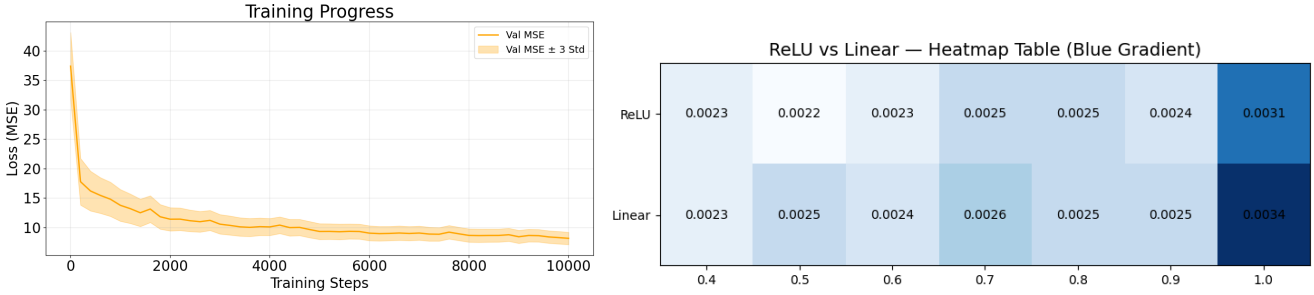
Figure 2.4: CLIPScore Optimal Pick Ratio

2.1.5. Shrinking Uncertainty and Frozen Layers Choice.

In this section, we justify the frozen layer parameter choice, as well as showcase the shrinking nature of uncertainty on image predictions over 10,000 prompts with MPS image quality score using 500 OLS Warmup Datapoints. Specifically, in Figure 2.5a, we demonstrate two key properties of our reward predictor. First, the mean squared error decreases rapidly in the early training steps, dropping from an initial MSE of 42 to below 12 within the first 2,000 prompts, indicating that the OLS warm-start provides a strong initialization that the online updates quickly refine. Second, the $\pm 3\sigma$ uncertainty band, which reflects disagreement across the $K = 10$ branches, narrows consistently as more datapoints are observed. This is a desirable property for our UCB formulation: early in training, wide uncertainty bands encourage exploration across models, while the narrowing uncertainty in later rounds drives the bandit towards exploitation of the best-performing model.

To determine the optimal frozen layer ratio, we conduct a grid search over freeze ratios $\beta \in \{0.4, 0.5, \ldots, 1.0\}$ and compare two activation functions for the reward head: ReLU and linear (no activation), as shown in Figure 2.5b. The heatmap reports validation MSE on CLIPScore predictions, where darker cells indicate higher error. Both activation functions produce comparable MSE across most freeze ratios, with values remaining stable between 0.0022 and 0.0026 for $\beta \in [0.4, 0.9]$. Performance degrades noticeably at $\beta = 1.0$ for both ReLU (0.0031) and linear (0.0034), suggesting that completely freezing the encoder and relying solely on the linear head is insufficient to capture the reward signal.

Based on these results, we select $\beta = 0.9$ with a linear activation for all subsequent evaluations. While $\beta = 1.0$ yields the lowest computational cost, its higher MSE makes it unsuitable. At $\beta = 0.9$, only the final encoder layer remains trainable, keeping the additional parameter count minimal while still achieving competitive prediction accuracy.



(a) Shrinking uncertainty and MSE during training **(b)** Parameter Search for Different Values of Frozen Encoder Portion

Figure 2.5: Shrinking Uncertainty and Frozen Layers Choice.

3

Offline Diffusion Model Selection

In contrast to the online setting, offline model selection operates under a full-scope regime in which all candidate models are trained using the complete set of available prompts. This allows for a comprehensive evaluation of each model's capacity to predict image quality. Inspired by similar works in LLM Model Selections and in MoE (Mixture of Expert) systems, we adopt a simple routing logic to select the best predicted model at inference. More specifically, we follow algorithm 1 and the high level representation as shown in figure 1. The selection agent operates as follows: given an incoming text prompt, it queries each model's reward predictor and selects the diffusion model expected to produce the highest image-quality reward, from which the final image is then sampled.

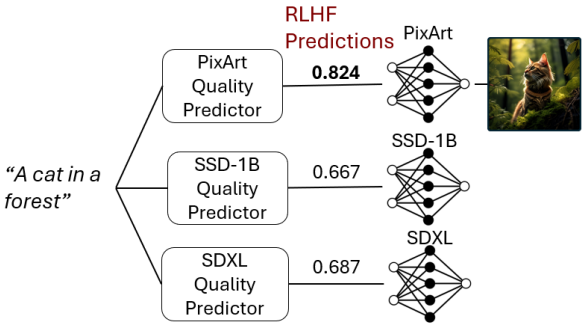


Figure 3.1: Offline Diffusion Model Selection

Algorithm 1 Diffusion Model Selection

Require: Model set $\mathcal{G}$, reward function Reward$(\cdot,\cdot)$, prompt p
Ensure: Selected model $g^\star$

▷ Step 1: Find highest-reward model

$best\_reward \leftarrow -\infty$
$best\_model \leftarrow$ None
for all $g \in \mathcal{G}$ **do**
  if Reward$(g, p) > best_reward$ **then**
    $best\_reward \leftarrow$ Reward(m, p)
    $best\_model \leftarrow g$
  end if
end for
return $selected\_model$

3.1. Image Quality Predictor

The key component of offline diffusion model selection is the image-quality predictor. We argue that the performance of the selection agent is fundamentally bounded by the accuracy of this reward predictor. In other words, more accurate image-quality estimates lead directly to better model selection decisions. Accordingly, we build our diffusion model image reward predictor on top of CLIP-ViT/L14 [11], partially freezing the text-encoder backbone during training. The final layer of the encoder is a linear layer that takes the encoded features concatenated with the normalised token number of the prompt to estimate the final reward value.

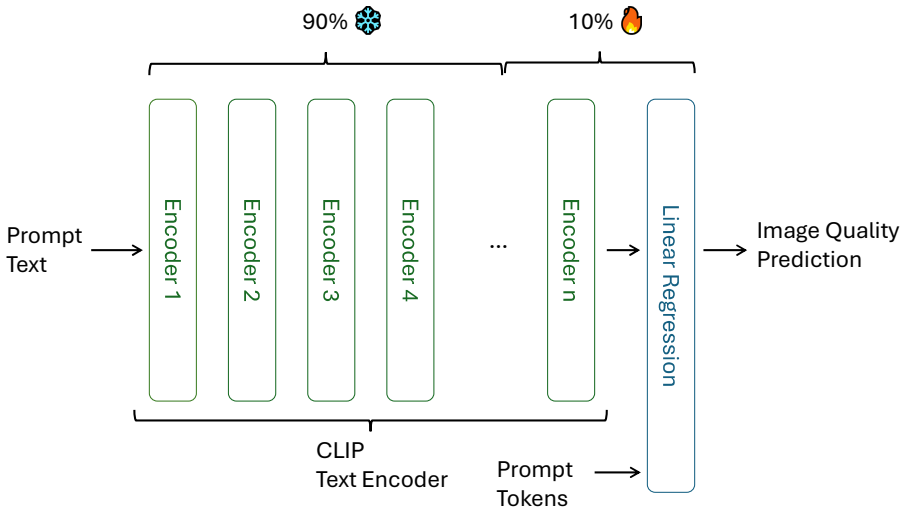


Figure 3.2: MPS Prompt Augmentation for training [18]

3.2. Evaluation

Training Parameters

To train the image quality predictor we finetune a text encoder model with a linear regressor head. We adopt AdamW optimizer with a decay of 0.01. We set the learning rate to $1e-5$, batch size 8 and number of epochs to 30. We use the full Pic-a-Pick dataset for the training data due to its diverse and large set of user generated prompts.

Prompt Dataset

We train the image-quality predictor using images generated and evaluated from 35,000 unique Pick-a-Pic prompts [6]. Pick-a-Pic contains prompts submitted by real users, providing the semantic diversity needed to assess generalization across a broad range of text-to-image generation scenarios. For evaluation, we use the Pick-a-Pic benchmark split containing 500 prompts. In addition to Pick-a-Pic we evaluate our methods on PartiPrompts dataset [16]. PartiPrompts is a diverse benchmark for text-to-image generation that containing over 1600 prompts spanning different categories, levels of complexity, and compositional structures. We use it to evaluate whether our method generalizes beyond the prompt distribution of Pick-a-Pic.

Diffusion Model Pool

We evaluate our method across six text-to-image diffusion models: SDXL Turbo[10], SSD-1B, Sana1.5[14], Pixart-α-XL[1], Kolors[12], and Z-Image-Turbo [13]. All models are run using the `diffusers` library[9] with each model's recommended default parameters, ensuring a fair comparison without model-specific tuning.

3.2.1. Evaluation Metrics

To asses the performance of offline training we evaluate 2 main components: (i) **the image-quality prediction model** trained on the full prompt dataset and (ii) the **overall meta model** composed of a pool of diffusion models and image-quality predictions. We use specifically, MPS, CLIPScore and RLHF due to their popularity among image quality metrics and due to their ability to measure human preference score.

To assess prediction performance, we employ two complementary metrics: Mean Squared Error (MSE), which quantifies the magnitude of prediction errors, and Pearson Correlation Coefficient, which measures the linear relationship between predicted and ground-truth quality scores. To demonstrate the effectiveness of our selection method, we report two aggregate performance indicators: the Optimal Pick Ratio, which reflects how frequently our method identifies the best-performing model across evaluation instances, and the Average Metric Score, which captures the expected quality of the selected model over the full evaluation set. Together, these measures provide a rigorous assessment of our method's ability to reliably identify the most suitable model in an offline regime.

3.2.2. Baselines

We evaluate our offline image quality predictor against other predictors adopted in online diffusion model selection strategies. Specifically, we measure performance against a Cubic Kernel regression adopted in PAK-UCB [5] and against a kNN strategy adopted in BALROG [2]. Additionally, we find that a simpler Linear Kernel can outperform a Polynomial Kernel in this setting. Motivated by this observation, we also include a Linear Kernel predictor as an additional baseline in our evaluation.

3.2.3. Results

CLIPScore Image-Quality Predictor Evaluation

		PixArt		SSD-1B		Sana		SD-Flash		Image-Turbo	
Model	Dataset	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑
Ours	Parti-Prompts	**0.0017**	**0.7625**	**0.0012**	**0.7694**	**0.0013**	**0.8077**	**0.0013**	**0.7448**	**0.0017**	**0.7459**
	Pick-a-Pic	**0.0014**	**0.8183**	**0.0011**	**0.7744**	**0.0012**	**0.8430**	**0.0012**	**0.7343**	**0.0022**	**0.7472**
Linear Kernel	Parti-Prompts	0.0017	0.7076	0.0014	0.7118	0.0017	0.7472	0.0015	0.6931	0.0019	0.7051
	Pick-a-Pic	0.0019	0.7416	0.0014	0.7117	0.0018	0.7608	0.0015	0.6692	0.0025	0.6915
Cubic Kernel	Parti-Prompts	0.0018	0.7013	0.0015	0.7123	0.0017	0.7284	0.0016	0.6710	0.0024	0.6246
	Pick-a-Pic	0.0018	0.7757	0.0013	0.7382	0.0013	0.8110	0.0014	0.6815	0.0024	0.7099
kNN (k=8)	Parti-Prompts	0.0021	0.6663	0.0018	0.6732	0.0021	0.7186	0.0020	0.6689	0.0030	0.6722
	Pick-a-Pic	0.0023	0.6919	0.0018	0.6710	0.0022	0.7077	0.0019	0.6506	0.0032	0.6162

Table 3.1: MSE and Spearman's Correlation on CLIPScore Image Quality Predictor

MPS Image-Quality Predictor Evaluation

		PixArt		SSD-1B		Sana		SD-Flash		Image-Turbo	
Model	Dataset	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑
Ours	Parti-Prompts	**8.6972**	**0.7030**	**8.4021**	**0.7805**	**8.8385**	**0.6918**	**8.3805**	**0.6922**	**8.5947**	**0.6816**
	Pick-a-Pic	**8.5254**	0.7897	**8.2898**	**0.8084**	**8.0794**	**0.7787**	**8.1508**	**0.7623**	**8.2256**	**0.7672**
Linear Kernel	Parti-Prompts	8.8867	0.7021	9.3675	0.7119	9.2506	0.6782	9.1983	0.6702	9.4189	0.6647
	Pick-a-Pic	8.6124	**0.7958**	9.0280	0.7833	9.3670	0.7452	8.9875	0.7501	9.0560	0.7475
Cubic Kernel	Parti-Prompts	9.6802	0.6810	10.0431	0.7013	9.8238	0.6692	9.3972	0.6748	9.8145	0.6578
	Pick-a-Pic	10.1052	0.7440	10.1238	0.7439	10.1556	0.7128	9.6815	0.7130	10.1731	0.7037
kNN (k=8)	Parti-Prompts	12.5444	0.5839	12.5494	0.6230	12.3454	0.5699	12.2777	0.5756	12.3947	0.5675
	Pick-a-Pic	12.7028	0.6872	13.3153	0.6707	12.6473	0.6402	12.6144	0.6362	12.9459	0.6144

Table 3.2: MSE and Spearman's Correlation on MPS Image Quality Predictor

RLHF Image-Quality Predictor Evaluation

		PixArt		SSD-1B		Sana		SD-Flash		Image-Turbo	
Model	Dataset	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑	MSE ↓	Spearman's Correlation ↑
Ours	Parti-Prompts	**0.0033**	**0.8001**	**0.0036**	**0.8515**	**0.0034**	**0.8248**	**0.0038**	**0.8163**	**0.0041**	**0.8168**
	Pick-a-Pic	**0.0025**	**0.8327**	**0.0030**	**0.8539**	**0.0025**	**0.8350**	**0.0033**	**0.8178**	**0.0031**	**0.8155**
Linear Kernel	Parti-Prompts	0.0045	0.7267	0.0056	0.7734	0.0048	0.7562	0.0053	0.7471	0.0060	0.7237
	Pick-a-Pic	0.0036	0.7320	0.0044	0.7841	0.0038	0.7332	0.0043	0.7683	0.0043	0.7201
Cubic Kernel	Parti-Prompts	0.0043	0.7533	0.0053	0.7720	0.0045	0.7790	0.0054	0.7475	0.0057	0.7457
	Pick-a-Pic	0.0038	0.7459	0.0042	0.8016	0.0037	0.7661	0.0042	0.7754	0.0043	0.7669
kNN (k=8)	Parti-Prompts	0.0057	0.6404	0.0077	0.6661	0.0066	0.5699	0.0067	0.6396	0.0074	0.6508
	Pick-a-Pic	0.0052	0.6547	0.0070	0.6826	0.0061	0.6402	0.0062	0.6624	0.0058	0.6454

Table 3.3: MSE and Spearman's Correlation on RLHF Image Quality Predictor

Model Selection Evaluation

		Random	Oracle	One Model Fits All	Ours		Linear Kernel		Polynomial Kernel		kNN	
Metric	Dataset	Average Score ↑	Average Score ↑	Average Score ↑	Average Score ↑	OPR ↑	Average Score ↑	OPR ↑	Average Score ↑	OPR ↑	Average Score ↑	OPR ↑
MPS	Parti-Prompts	12.0910	13.2465	12.2478	**12.6226**	**0.3940**	12.5778	0.3700	12.5372	0.3280	12.4512	0.3260
	Pick-a-Pic	11.6501	13.1532	12.0124	**12.1803**	**0.3407**	12.1665	0.3321	12.1543	0.3358	12.1051	0.3021
RHFL	Parti-Prompts	0.6041	0.6566	0.6161	**0.62451**	**0.3680**	0.62322	0.3600	0.6185	0.3120	0.6216	0.3080
	Pick-a-Pic	0.6201	0.6751	0.6201	**0.64990**	**0.3180**	0.64711	0.2990	0.6458	0.2739	0.6449	0.2782
CLIP	Parti-Prompts	0.3493	0.3837	0.3642	0.36748	0.356	0.36750	0.3760	**0.3683**	**0.3940**	0.3652	0.3240
	Pick-a-Pic	0.3321	0.3842	0.3531	**0.36396**	**0.296**	0.36254	0.2917	0.3629	0.2874	0.3608	0.2819

Table 3.4: Selection Average Score and Optimal Pick Ratio (OPR) against baselines

Result Overview

Our strategy consistently performs better than the competing image-quality predictors across all evaluation settings. Compared with the Linear Kernel, Cubic Kernel, and kNN baselines, it repeatedly achieves lower MSE and higher Spearman's correlation across diffusion models and datasets. These results indicate that our predictor not only improves accuracy, but also generalizes more reliably across different models and prompt distributions.

The improvement is even clearer when considering OPR, where our method also performs best in five of the six settings. Compared with the strongest competing baseline, it improves OPR by +6.49% for MPS on Parti-Prompts, +1.46% for MPS on Pick-a-Pic, +2.22% for RLHF on Parti-Prompts, +6.35% for RLHF on Pick-a-Pic, and +1.47% for CLIP on Pick-a-Pic. These gains indicate that our approach is not only competitive in average reward, but also more effective at selecting the optimal model for each prompt.

The only exception is CLIP on Parti-Prompts, where the Polynomial Kernel slightly exceeds our method, by 0.22% in average score and 9.64% in OPR. Nevertheless, the overall trend across metrics and datasets consistently favors our approach, showing that it provides the most reliable selection strategy among the evaluated baselines.

4

Evaluation of Image Quality Metrics

4.0.1. CLIPScore

CLIPScore measures the semantic alignment between a text prompt and a generated image by leveraging the pretrained CLIP model [11] [4], which learns a shared embedding space where semantically similar text-image pairs are mapped to nearby points through contrastive learning on large-scale web data. Formally, given a prompt pp p and a generated image II I, CLIPScore is computed as the cosine similarity between their normalised CLIP embeddings:

$$\mathbf{v}p = \frac{\text{CLIPtext}(p)}{|\text{CLIP}_{\text{text}}(p)|2}, \quad \mathbf{v}I = \frac{\text{CLIPimage}(I)}{|\text{CLIPimage}(I)|_2} \tag{4.1}$$

$$\text{CLIPScore}(p, I) = \max(0, 100 \cdot \mathbf{v}_p^\top \mathbf{v}_I) \tag{4.2}$$

where higher values indicate stronger semantic alignment between the prompt and the generated image.

4.0.2. RLHF

RLHF is image quality metric scoring image-text pairs based on plausibility, image-text alignment, aesthetics and overall rating [8]. RLHF is trained from a multimodal transformer (built on ViT + T5X) to automatically predict all of the previously mentioned scores from an image-text pair. This image quality metric is trained from 18k rich human annotated Pick-a-Pic generated images. Annotations include marking of implausible or misaligned regions, and labeled keywords from the text prompt that are missing or misrepresented in the image. Each image is annotated by three raters, with scores averaged and spatial annotations merged into heatmaps.

4.0.3. MPS

Similarly to RLHF, MPS introduces a text-image pair quality score evaluating its aesthetic appeal, text-image alignment, detail quality and overall score [18]. MPS model is trained on 607,541 prompts collected from DiffusionDB, PromptHero, Kolors and augmented with GPT4. Notably, GPT4 agumentated prompt add up to 66,389 covering underepresented prompt categories as shown in figure. Images are then generated through 9 distinct text-to-image models and annotated with a score ranging from 1 to 5 for each criteria. MPS introduces a preference condition setting, allowing a single unified model to evaluate images across multiple dimensions by conditioning on dimension-specific descriptor words that selectively focus the model's attention on the most relevant parts of the prompt (e.g. Aesthetics: *light, color, clarity, tone, style, ambiance, artistry*).

4.0.4. Evaluating Image Quality Metrics on Prompt Category

In this section, we investigate the sensitivity of image quality metrics to prompt category and model choice. Specifically, we examine whether image quality scores vary primarily as a function of the generative model used, or whether they are significantly influenced by the semantic category of the input prompt. Understanding this distinction is important for model selection: if scores are largely

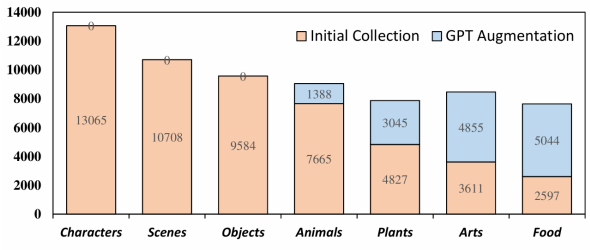


Figure 4.1: MPS Prompt Augmentation for training

model-dependent, a single best model can be reliably identified regardless of prompt type. However, if scores vary substantially across prompt categories, this would support the need for prompt-level model selection, where different models are preferred for different types of prompts. To this end, we evaluate a set of diffusion models across prompts spanning multiple semantic categories and analyse the variance in image quality scores both within and across categories.

4.1. RHFL and MPS have Better Distinguibility than CLIPScore

As highlighted by the authors in [15], RHFL evaluation metric shows a much higher granulanity than CLIPScore, showing an overall much higher variance across different models. We observe similar results when comparing RHFL and MPS to CLIPScore using different prompt types (anime, cat, elephant, bicycle and train), suggesting that CLIPScore yields relatively uniform performance across prompt domains. For instance, the average CLIPScore remains largely invariant across prompt types, whereas RHFL and MPS display varying sensitivities and results across these categories as shown in figure 4.2. This finding suggests that RHFL and MPS benefit the most from a model selection framework: as multiple models have different strong prompt types, a selection framework can exploit these complementary strengths to consistently route each prompt to its most suitable model.

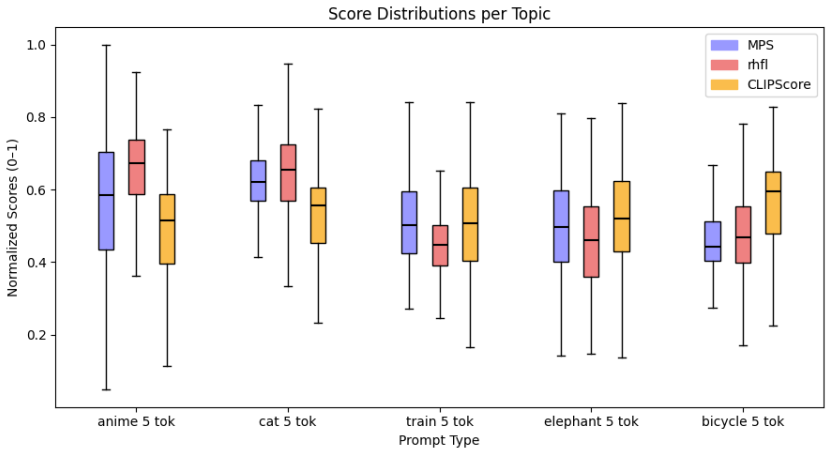


Figure 4.2: Boxplot of MPS, RHFL and CLIPScore across

We extend our analysis by examining multiple diffusion model performance across different prompt types. Specifically, we observe that certain diffusion models excel at generating images for specific prompt types while performing poorly on others. At the same time, we find that RHFL and MPS are more effective than CLIPScore in distinguishing each model's strong and weak prompt domains. To quantify this, we rank diffusion models using RHFL, MPS, and CLIPScore for each prompt type and observe that RHFL and MPS produce more diverse rankings across prompts, reflecting greater sensitivity to domain-specific differences as shown in the example below 4.1.1.

4.1.1. Example of Model Ranking across different prompt types

In this particular diffusion model rankings, we rank SSD-1B, Kandinsky 3, PixArt α and Kolors-Diffusers on 5 different prompt types using CLIPScore, MPS and RHFL. We quantify the variability in rankings by applying Shannon entropy of the distribution of ranks. Specifically, we apply equation 4.3 to get entropy H, with a higher value indicating higher unpredictability in the rankings.

$$H = \frac{1}{R}\sum_{r=1}^{R} H_r, \qquad H_r = -\sum_{m=1}^{M} p_r(m)\log p_r(m) \tag{4.3}$$

Where H_r is the rank specific entropy, R is the number of rank positions (e.g., 1–4), M is number of models, and $p_r(m)$ is the measured probability that model m appears at rank r across prompt types. Finally the entropy is normalised so that it lies within $[0, 1]$ following equation 4.4.

$$H_{\text{norm}} = \frac{H}{\log M} \tag{4.4}$$

For example table 4.1 shows that CLIPScore rankings stays almost invariant across all prompt types, and the invariability is also shown by the low entropy of 0.1805. MPS and RHFL, on the other hand, have a much more unpredictable rankings showing an entropy of 0.7232 and 0.7421 each in tables 4.2 and 4.3.

Model Ranking	Anime	Cat	Train	Bicycle	Elephant
1	SSD-1B	SSD-1B	SSD-1B	SSD-1B	SSD-1B
2	Kandinsky 3	Kandinsky 3	Kandinsky 3	Kandinsky 3	PixArt α
3	PixArt α	PixArt α	PixArt α	PixArt α	Kandinsky 3
4	Kolors-Diffusers	Kolors-Diffusers	Kolors-Diffusers	Kolors-Diffusers	Kolors-Diffusers

Table 4.1: CLIPScore Model Ranking for prompt types. Entropy of this rankings is **0.1805**, indicating low variability in model performance across different prompt types.

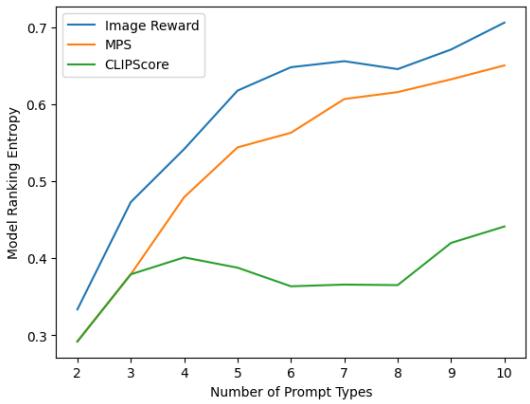
Model Ranking	Anime	Cat	Train	Bicycle	Elephant
1	SSD-1B	Kandinsky 3	PixArt α	Kolors-Diffusers	Kolors-Diffusers
2	Kandinsky 3	SSD-1B	Kolors-Diffusers	Kandinsky 3	PixArt α
3	PixArt α	PixArt α	Kandinsky 3	PixArt α	Kandinsky 3
4	Kolors-Diffusers	Kolors-Diffusers	SSD-1B	SSD-1B	SSD-1B

Table 4.2: MPS Model Ranking for prompt types. Entropy of this ranking is **0.7232**, indicating high variability in model performance across different prompt types.

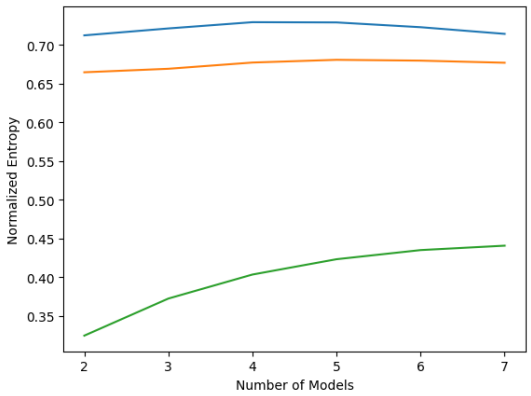
Model Ranking	Anime	Cat	Train	Bicycle	Elephant
1	SSD-1B	Kandinsky 3	Kolors-Diffusers	Kandinsky 3	PixArt α
2	Kandinsky 3	PixArt α	PixArt α	PixArt α	Kolors-Diffusers
3	PixArt α	SSD-1B	Kandinsky 3	Kolors-Diffusers	Kandinsky 3
4	Kolors-Diffusers	Kolors-Diffusers	SSD-1B	SSD-1B	SSD-1B

Table 4.3: RHFL Model Ranking for prompt types. Entropy of this ranking is **0.7421**, indicating high variability in model performance across different prompt types.

We further analyze the entropy of diffusion model selection across different prompt types by plotting MPS, ImageReward and CLIPScore across varying number of diffusion models and prompt types in figures 4.3a and 4.3b.



(a) Increasing Model Ranking Entropy with Increasing Number of Prompt Types



(b) Constant Model Ranking Entropy with Increasing Number of Models

Figure 4.3: Entropy of Increasing Models and Prompt Types

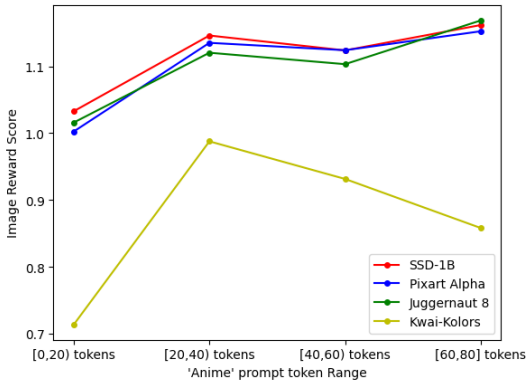
In figure 4.3a, as the the number of prompt types increases, the average normalised entropy increases as well. This effect can be attributed due to models showing diversified strong and weak metric scores in new incoming prompt types. Notably CLIPScore, has a much lower entropy level than MPS and ImageReward.

In figure 4.3b, on the other hand, the number of models does not affect entropy. The constant entropy could mean that the ranking diversity pattern doesn't fundamentally change when more models are added.

4.2. Prompt Length Effect on Image Score

Recent literature on diffusion model parameter selection [17, 19] suggests that prompt difficulty influences ImageReward scores. Specifically, [17] demonstrates that prompts containing a large number of words or objects benefit from more diffusion denoising steps, whereas simpler prompts perform best with fewer steps. Similarly, [19] shows that adjusting the classifier-free guidance (CFG) scale based on prompt difficulty (quantified by prompt length and entropy) can enhance image quality.

Our work further extends these findings by showing that both prompt length and prompt type affect image scores as shown in figures 4.4 and 4.5. We illustrate this by plotting ImageReward and MPS scores across four prompt-length ranges for two prompt types ("cat" and "anime"), using 400 images sampled from distinct DiffusionDB prompts per range. These results indicate that image score depends jointly on prompt type and prompt length.



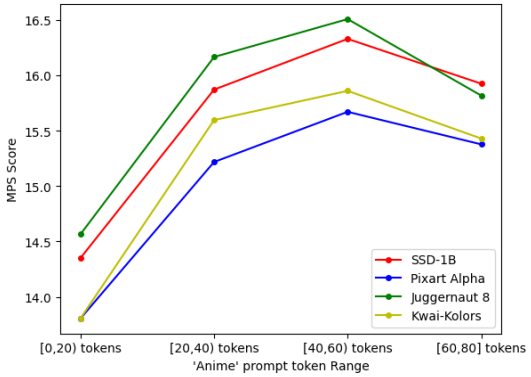


Figure 4.4: Changing RHFL and MPS scores with varying prompt lengths

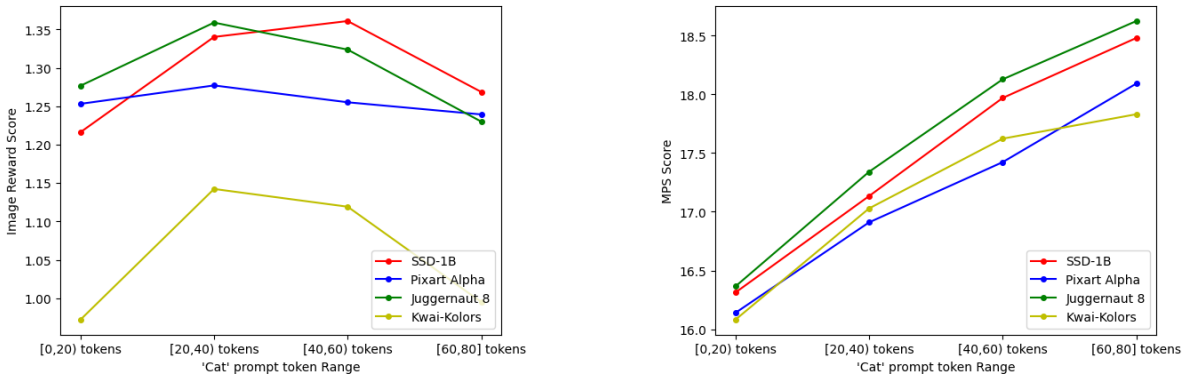


Figure 4.5: Changing RHFL and MPS scores with varying prompt lengths

5

Conclusion

This thesis framed diffusion model selection as a prompt-level decision problem, showing that adaptive routing consistently outperforms committing to a single best model. In the online setting, Ensemble-UCB achieves positive advantage over the best static model after roughly 2,500 rounds, outperforming both BALROG and PAK-UCB across all three metrics. The bootstrapped UCB index and OLS warm-start together enable efficient exploration without requiring explicit decay schedules, and the framework adapts gracefully when new models are introduced mid-deployment. In the offline setting, our CLIP-based reward predictor outperforms kernel and kNN baselines in five of six evaluation settings, translating to consistent gains in both average score and optimal pick ratio. A consistent finding across both settings is that MPS and RLHF are better suited for model selection than CLIPScore, as their higher sensitivity to prompt-level variation makes the selection problem more tractable. Overall, the results validate that prompt-aware model selection is a practical and effective alternative to static model choice, with clear benefits across both online and offline deployment regimes.

5.0.1. Explored Directions and Unsuccessful Attempts

Throughout the course of this thesis, several research directions were explored that ultimately did not yield the desired results. One such direction involved the development of a multi-dimensional diffusion model selection framework, in which candidate models would be selected not solely based on generative performance, but also according to hardware-aware criteria. Specifically, the framework aimed to jointly consider model size, sampling speed, and image quality as orthogonal dimensions of a composite selection criterion, thereby rendering model selection dependent not only on the model's generative capabilities but also on the constraints and characteristics of the target hardware environment.

We also explored a topic-driven model selection approach, where the goal is to identify the best diffusion model for a specific user-defined topic rather than across general prompts. Given a fixed evaluation budget, a multi-armed bandit agent using the Sequential Halving (SH) algorithm iteratively generates topic-relevant images and progressively eliminates underperforming models across rounds until a single winner is identified. Prompts are generated on-the-fly by a fine-tuned LLM, conditioned on the target topic and a desired prompt length, ensuring that all models are evaluated on semantically consistent inputs throughout the selection process.

References

[1] Junsong Chen et al. *PixArt-α: Fast Training of Diffusion Transformer for Photorealistic Text-to-Image Synthesis*. 2023. arXiv: 2310.00426 [cs.CV].

[2] Jules Damidaux et al. *BALROG: Contextual Bandits meets Active Learning for Online Generative Model Selection*. 2026. URL: https://openreview.net/forum?id=6a2CJrizrh.

[3] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2010, pp. 249–256.

[4] Jack Hessel et al. "Clipscore: A reference-free evaluation metric for image captioning". In: *Proceedings of the 2021 conference on empirical methods in natural language processing*. 2021, pp. 7514–7528.

[5] Xiaoyan Hu, Ho-fung Leung, and Farzan Farnia. "PAK-UCB contextual bandit: An online learning approach to prompt-aware selection of generative models and LLMs". In: *arXiv preprint arXiv:2410.13287* (2024).

[6] Yuval Kirstain et al. "Pick-a-pic: An open dataset of user preferences for text-to-image generation". In: *Advances in neural information processing systems* 36 (2023), pp. 36652–36663.

[7] Yann LeCun et al. "Efficient backprop". In: *Neural networks: Tricks of the trade*. Springer, 2002, pp. 9–50.

[8] Youwei Liang et al. "Rich human feedback for text-to-image generation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 19401–19411.

[9] Patrick von Platen et al. *Diffusers: State-of-the-art diffusion models*. https://github.com/huggingface/diffusers. 2022.

[10] Dustin Podell et al. "Sdxl: Improving latent diffusion models for high-resolution image synthesis". In: *arXiv preprint arXiv:2307.01952* (2023).

[11] Alec Radford et al. "Learning transferable visual models from natural language supervision". In: *International conference on machine learning*. PmLR. 2021, pp. 8748–8763.

[12] Chitwan Saharia et al. "Photorealistic text-to-image diffusion models with deep language understanding". In: *Advances in neural information processing systems* 35 (2022), pp. 36479–36494.

[13] Tongyi-MAI. *Z-Image: An Efficient Image Generation Foundation Model with Single-Stream Diffusion Transformer*. https://huggingface.co. 2025.

[14] Enze Xie et al. *Sana: Efficient High-Resolution Image Synthesis with Linear Diffusion Transformer*. 2024. arXiv: 2410.10629 [cs.CV]. URL: https://arxiv.org/abs/2410.10629.

[15] Jiazheng Xu et al. "Imagereward: Learning and evaluating human preferences for text-to-image generation". In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 15903–15935.

[16] Jiahui Yu et al. "Scaling autoregressive models for content-rich text-to-image generation". In: *arXiv preprint arXiv:2206.10789* 2.3 (2022), p. 5.

[17] Hui Zhang et al. "AdaDiff: adaptive step selection for fast diffusion models". In: 39.9 (2025), pp. 9914–9922.

[18] Sixian Zhang et al. "Learning multi-dimensional human preference for text-to-image generation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 8018–8027.

[19] Xuanhao Zhang and Chang Li. "Prompt-aware classifier free guidance for diffusion models". In: *arXiv preprint arXiv:2509.22728* (2025).