Draft Model Knows When to Stop: Self-Verification Speculative Decoding for Long-Form Generation

Ziyin Zhang[1,2,*] **Jiahao Xu**[2] **Tian Liang**[2] **Xingyu Chen**[1,2*] **Zhiwei He**[1,2*]
Rui Wang[1†] **Zhaopeng Tu**[2†]
[1]Shanghai Jiao Tong University [2]Tencent
[1]{daenerystargaryen,galaxychen,zwhe.cs,wangrui12}@sjtu.edu.cn
[2]{jettexu,ttianliang,zptu}@tencent.com

Abstract

Conventional speculative decoding (SD) methods utilize a predefined length policy for proposing drafts, which implies the premise that the target model smoothly accepts the proposed draft tokens. However, reality deviates from this assumption: the oracle draft length varies significantly, and the fixed-length policy hardly satisfies such a requirement. Moreover, such discrepancy is further exacerbated in scenarios involving complex reasoning and long-form generation, particularly under test-time scaling for reasoning-specialized models. Through both theoretical and empirical estimation, we establish that the discrepancy between the draft and target models can be approximated by the draft model's prediction entropy: a high entropy indicates a low acceptance rate of draft tokens, and vice versa. Based on this insight, we propose **SVIP**: **S**elf-**V**er**i**fication Length **P**olicy for Long-Context Speculative Decoding, which is a training-free dynamic length policy for speculative decoding systems that adaptively determines the lengths of draft sequences by referring to the draft entropy. Experimental results on mainstream SD benchmarks as well as reasoning-heavy benchmarks demonstrate the superior performance of SVIP, achieving up to 17% speedup on MT-Bench at 8K context compared with fixed draft lengths, and 22% speedup for QwQ in long-form reasoning.

1 Introduction

Speculative decoding (Leviathan et al., 2023; Chen et al., 2023) is a novel technique that markedly enhances the generation wall-time of large language models (LLMs). This approach employs a small and efficient draft model to draft sequences, while concurrently utilizing a larger and more powerful expert model to verify the drafts. By avoiding the autoregressive generation of each token through the target LLM, speculative decoding achieves improved efficiency while preserving the quality of the output. This technique is particularly beneficial in the context of **inference-time scaling**, where LLMs generally generate long-form text.

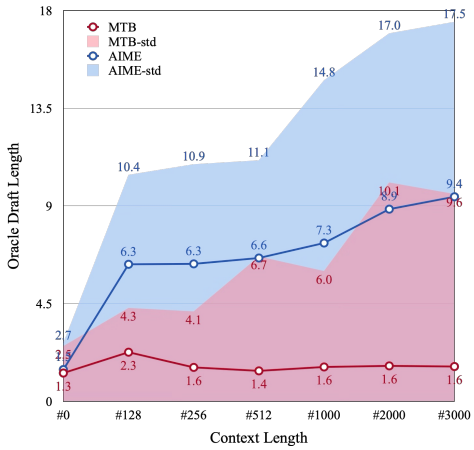


Figure 1: The variance of oracle draft length drastically increases with context length. MTB (MT-Bench): a conventional benchmark for SD systems. AIME: an extremely difficult mathematical testing set for advanced reasoning models.

The majority of research on speculative decoding focuses on improving the **acceptance rate of the draft sequences** (Sun et al., 2023; Li et al., 2024b; Elhoushi et al., 2024; Du et al., 2024; Li et al., 2024a; Lu et al., 2024) or introducing **novel draft model architectures** (Zhang et al., 2024a). However, they limit their settings to a **fixed draft length** (e.g. less than 5 tokens), which we find is sub-optimal in the scenario of long-form text generation.

Specifically, as discussed in Section 2.1, we investigate the token rejection phenomenon in speculative decoding (SD) systems. Our findings reveal that the oracle draft length varies considerably for long-form generation, as shown in fig. 1. Further-

[*]Work done during their internship at Tencent.
[†]Corresponding authors.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 16697–16709
November 4-9, 2025

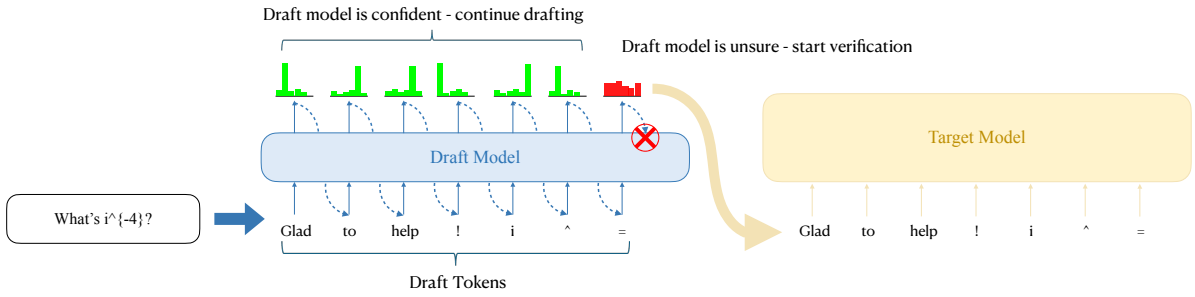


Figure 2: Overview of **SVIP**: the draft model proceeds the generation process (marked by green) until it encounters a token for which it has low confidence (marked by red), signaled by high entropy, at which point the draft model would cease generation and send the draft spans to the target model for verification.

more, heuristic methods, such as predicting the draft length, are impractically challenging because our further investigation reveals that token rejection occurs unexpectedly. Moreover, our closer investigation indicates a strong correlation between token rejection and the draft model's prediction entropy at that moment.

Inspired by such a correlation, we analyze the acceptance rate and propose our **SVIP**: **S**elf-**V**erification Length **P**olicy in Section 2.2. Specifically, we derive a lower bound for the acceptance rate based on the entropy information from the draft model. Notably, SVIP not only approximates this lower bound but also dynamically adjusts the length of draft sequences by determining whether to continue drafting or initiate verification after each token generation, as shown in Figure 2. By optimizing draft sequence lengths, SVIP enhances SD systems' efficiency. Importantly, our method is entirely training-free and thus can be seamlessly integrated with any SD decoding algorithm, making it broadly applicable and efficient.

With extensive experiments across multiple model sizes and evaluation benchmarks, we demonstrate the superior performance of SVIP in long-context generation. Compared with fixed-length draft policies, it yields up to 17% speedup on MT-Bench (Zheng et al., 2023) and 22% on AIME. As a training-free length policy, SVIP is also extremely flexible and compatible with state-of-the-art speculative decoding systems such as EAGLE-2 (Li et al., 2024a), achieving an additional 13% speed improvement.

In summary, our contributions are threefold:

1. We provide an in-depth analysis of the disagreement between draft model and target model in speculative decoding systems, highlighting the underperformance of fixed-length draft length policies.

2. Based on this analysis, we derive a low bound of speculative decoding systems, where the acceptance rate of the draft model could be modeled by its entropy only. We further develop SVIP, an entropy-based dynamic draft length policy for speculative decoding systems, which is extremely flexible and can be adapted to any auto-regressive draft model.

3. Experimental results demonstrate the superior performance of SVIP over baseline draft length policies on both conventional long-form generation and reasoning-heavy benchmarks.

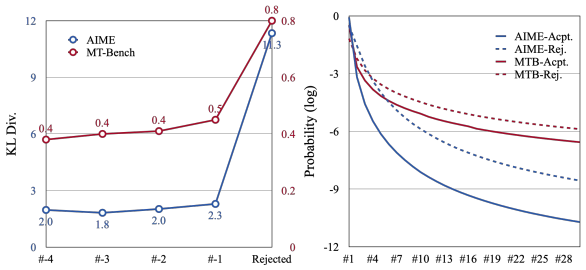
2 Draft Model Knows When to Stop

In this section, we first examine the behavior of draft models at the rejection phenomenon, and analyze the oracle lengths for SD systems. Then, we theoretically derive SVIP, which approximates the draft token acceptance rate using the draft model's own prediction entropy.

2.1 Investigation of Rejection

Speculative decoding enhances the efficiency of large language model (LLM) inference by assuming that draft tokens are *accepted* by the LLM, thus avoiding autoregressive generation. Should the target model exhibit a tendency to reject tokens, the overall performance of the system may experience considerable degradation. Consequently, empirically investigating the rejection phenomenon is our primary interest.

Specifically, we analyze the distribution characteristics of rejected tokens across two scenarios:



(a) KL Div. (↓) (b) Sorted vocab. distribution

Figure 3: Agreement scores and sorted vocabulary log probability at the rejection phenomenon. The x-axis of the Figure 3(a) represents rejected tokens and the four tokens before them.

- **AIME (2022-2024)**: a challenging math reasoning dataset, using greedy decoding with QwQ-32B-Preview (Team, 2024) and a 1.5B draft model;
- **MT-Bench**: a conversational and instruction-following benchmark, using sampling decoding with the Qwen2.5 family (Yang et al., 2024).

Rejection occurs out of the blue A natural question is: *How does the token rejection occur? Is there any symptom before it happens?* We investigate the rejected tokens with the following metrics:

- KL divergence for vocabulary distribution difference.
- Vocabulary distribution of accepted and rejected tokens.

We quantify the occurrence of the token rejection phenomenon along with its corresponding prefix tokens, as illustrated in Figures 3(a) and 3(b). It shows that KL metrics experienced a sudden and substantial surge in the position of the rejected token, and the output vocabulary distribution at rejected tokens differs significantly from previously correctly drafted tokens. Such a phenomenon is expected: KL divergence indicates a system discrepancy with rejection, where the ground-truth tokens are inherently difficult for the draft model to predict. We also quantify the entropy of the draft model at the acceptance and rejection positions in Table 1. It shows that the draft models suffer from a severely high entropy at rejection, indicating extreme difficulty in modeling the corresponding tokens.

Dataset	Vocab Entropy	
	Accepted.	*Rejected*
AIME	0.25	1.30
MT-Bench	2.18	3.99

Table 1: Vocabulary entropy of the draft model at accepted and rejected draft tokens.

2.2 Our Method: SVIP

Since the rejection phenomenon occurs suddenly and the draft model suffers from high entropy at the rejection position, can we detect the rejection with KL divergence or draft model's entropy? To achieve this, we seek a theoretical understanding of the rejection phenomenon.

Lower Bound of Acceptance Since rejection denotes the sudden decrease of acceptance rate, we investigate the theoretic acceptance rate of the SD systems. Specifically, given a target model p, a draft model q, an input sequence $x_{<t}$, and a draft token x_t, it's easy to derive that x_t's acceptance probability is $\min\left(1, \frac{p(x_t)}{q(x_t)}\right)$ (see Appendix A). Let β denote the expected acceptance probability over the distribution of x_t, and it follows that:

$$\begin{aligned}\beta &= \sum_x q(x) \cdot \min\left(1, \frac{p(x)}{q(x)}\right) \\ &= \sum_x \min\left(p(x), q(x)\right), \end{aligned} \tag{1}$$

where p and q denote the target and draft model respectively. Chen et al. (2023) has proven that β is related to the total variational distance (TVD) between p and q. Start from this, we utilize Pinsker's inequality in Equation (3) and yield the following bound in Equation (4):

$$\beta = 1 - \mathrm{TVD}(p, q) \tag{2}$$

$$\geqslant 1 - \sqrt{\frac{1}{2}\mathbb{KL}(q||p)} \tag{3}$$

$$= 1 - \sqrt{\frac{1}{2}H_{q,p} - \frac{1}{2}H_q} \tag{4}$$

where $H_{q,p}$ is the cross entropy between q and p, and H_q is the entropy of q. We denote the above bound as the *oracle bound*. Utilizing this bound for acceptance prediction is infeasible since it requires instantaneous access to the target model for cross entropy $H_{q,p}$, which is infeasible during the drafting phase.

Approximating the Oracle Bound with Draft Distribution Can we approximate the cross-entropy $H_{q,p}$ between the draft model's distribution q and the target distribution p using only q? We propose using the draft model's entropy H_q as a proxy, approximating $H_{q,p}$ as γH_q, where $\gamma = H_{q,p}/H_q$ is a random variable capturing the ratio between $H_{q,p}$ and H_q. This leads to a bound on the acceptance rate β:

$$\beta \geq 1 - \sqrt{\frac{1}{2}(\gamma - 1)H_q}. \tag{5}$$

To make this bound practical, we approximate γ with a constant c, yielding the *approximation bound*:

$$\beta \geq 1 - \sqrt{\frac{1}{2}(\gamma - 1)H_q} \approx 1 - \sqrt{cH_q}. \tag{6}$$

This bound holds when $1 - \sqrt{\frac{1}{2}(\gamma - 1)H_q} \geq 1 - \sqrt{cH_q}$, ensuring the approximation is conservative. Thus, the approximation bound $1 - \sqrt{cH_q}$ lower-bounds the true acceptance rate β.

Detecting Rejection with Draft Entropy With Equation (6) providing a way to estimate the acceptance probability using only the draft model's entropy, we introduce SVIP, which dynamically adapts the draft length. After generating each draft token, we compute the approximation bound and halt drafting if it falls below a threshold $\hat{h}$, i.e., if $1 - \sqrt{cH_q} < \hat{h}$. Since c and $\hat{h}$ are constant hyperparameters, we simplify the criterion by defining a new threshold $h = (1 - \hat{h})/\sqrt{c}$, absorbing $\sqrt{c}$ into h. This reduces the stopping condition to $\sqrt{H_q} > h$. Formally, given a prefix $x_{<t}$ of $t-1$ tokens, the stopping criterion is:

$$\sqrt{H_q(x_{<t})} > h, \tag{7}$$

where $H_q(x_{<t})$ is the entropy of the draft distribution conditioned on $x_{<t}$. This ensures drafting stops when the estimated acceptance probability is too low, optimizing efficiency while maintaining reliability. We formalize SVIP in Algorithm 1. The details of the methods `Verify` and `Correct` are given in Appendix A, for which different versions are available for sampling (Algorithm 2, 4) and greedy decoding (Algorithm 3, 5).

Justifying the Approximation Bound The tightness of the approximation depends on how well c captures the behavior of the random variable γ.

Algorithm 1 SVIP

Input: target model p, draft model q, input sequence $x_{\leqslant t}$, maximum length T, threshold h

Initialize $n \leftarrow t$
while $n < T$ **do**
  $j = 0$
  while True **do**
    Sample $x_{n+j} \sim q(x|x_{<n+j})$
    $j \leftarrow j + 1$
    if $\sqrt{H(q_{x|x_{<n+j}})} > h$ **then**
      Exit while loop
    end if
  end while
  $\gamma \leftarrow j$
  Compute $p(x|x_{<n+j}),\ j = 1, \cdots, \gamma + 1$ in parallel
  $\tilde{n} \leftarrow n$
  for $j = 1$ to γ **do**
    if Verify $\left(p_{x|x_{<n+j}},\ q_{x|x_{<n+j}}, x_{n+j}\right)$ **then**
      $\tilde{n} \leftarrow \tilde{n} + 1$
    else
      $x_{n+j} \leftarrow$ Correct $\left(p_{x|x_{<n+j}},\ q_{x|x_{<n+j}}\right)$
      Exit for loop
    end if
  end for
  if $\tilde{n} == n + \gamma$ **then**
    Sample $x_{n+\gamma+1}$ from $p(x|x_{\leqslant n+\gamma})$
  end if
  $n \leftarrow \tilde{n} + 1$
end while

Output: $x_{\leqslant n}$

For the approximation bound to be conservative, it requires:

$$\gamma \leq 2c + 1. \tag{8}$$

Given the right-skewed nature of $\gamma \geq 1$, we model $\gamma = 1 + X$, where $X \sim \text{Gamma}(\alpha, \beta)$. The probability that the approximation is valid is:

$$P(\gamma \leq 2c + 1) = P(X \leq 2c) = \frac{\gamma(\alpha, \beta \cdot 2c)}{\Gamma(\alpha)}, \tag{9}$$

where $\gamma(\alpha, z) = \int_0^z t^{\alpha-1} e^{-t}\, dt$ is the lower incomplete gamma function, and $\Gamma(\alpha)$ is the gamma

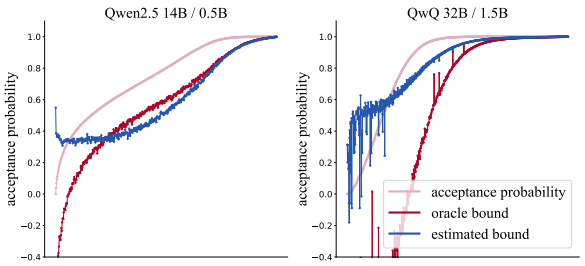


Figure 4: Comparison between the actual acceptance probability in Equation (1), the acceptance probability lower bound in Equation (4), and the estimated lower bound in Equation (6). Each position on the x-axis corresponds to a token, which has been sorted according to the actual acceptance probability.

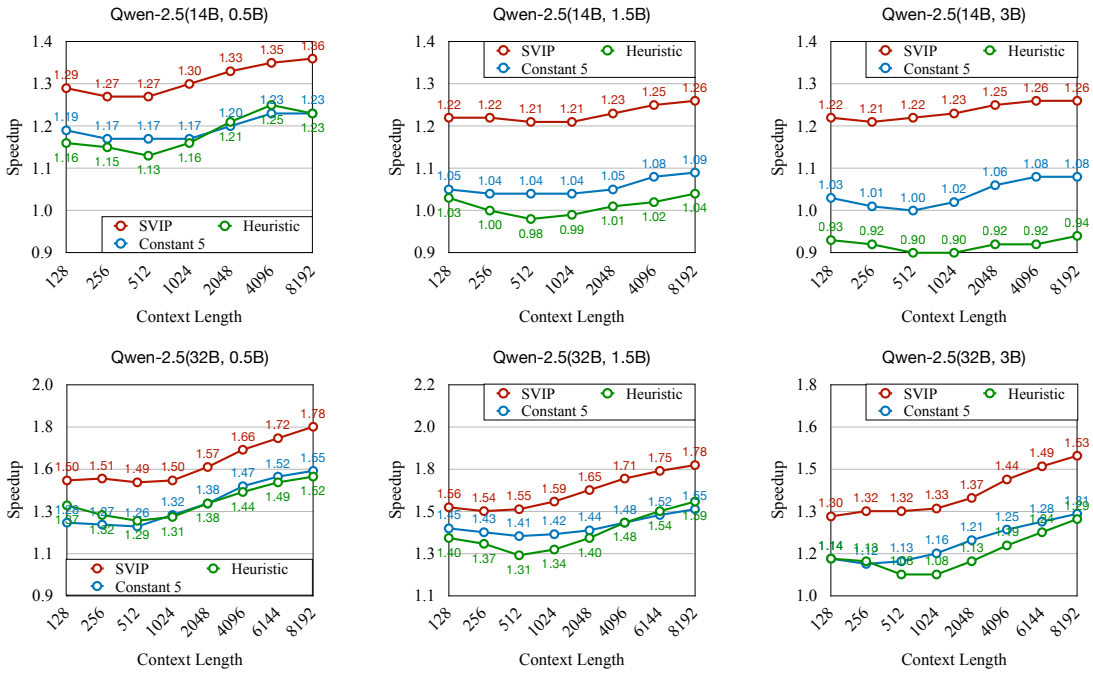


Figure 5: The SD system speedup on MT-Bench using Qwen2.5-14B (top) and Qwen2.5-32B (bottom) as targets and three different smaller models as drafts.

function. The choice of c trades off reliability and tightness:

- If c is small (e.g., $2c < \mathbb{E}[X] = \alpha/\beta$), the probability $P(X \leq 2c)$ is low, risking an invalid bound.
- If c is large, $P(X \leq 2c) \to 1$, but the bound $\beta \geq 1 - \sqrt{cH_q}$ becomes looser.

Optimal performance requires balancing the reliability of the approximation (high $P(X \leq 2c)$) with the tightness of the bound (small c).

We analyze such a trade-off on Qwen2.5 on MT-Bench and QwQ-32B on AIME in Figure 4. It shows that for most cases our estimated approximation bound works well (has a higher acceptance probability than the oracle bound when hyperparameter c (i.e. h) is properly selected, set to 0.18 in the figure).

3 Experiments

Next, to verify the effectiveness of SVIP, we conduct experiments on both conventional long-form generation (Section 3.1) and reasoning with test-time scaling (Section 3.2). Since SVIP is completely training-free, we also apply it to other speculative decoding methods and demonstrate its flexibility (Section 3.1).

As baselines, we consider two widely adopted policies for draft length:

1. **Constant**: a constant draft length (set to 5 unless otherwise stated), which is commonly used in the literature
2. **Heuristic**: the heuristics implemented in Hugging Face Transformers library (Wolf et al., 2019), where the draft length for the next draft iteration is increased by 2 if all draft tokens in the current iteration are accepted, and otherwise decreased by 1.

3.1 Results on Long-form Generation

Settings We first validate the effectiveness of SVIP on the widely used MT-Bench (Zheng et al., 2023) using different sizes of Qwen2.5 (Yang et al., 2024) as target and draft models. Unlike many existing works on speculative decoding (Chen et al., 2023; Du et al., 2024) that limit their experiments to generating short sequences of 128 tokens, we conduct experiments on long-form generation with up to 8K context to investigate the applicability of speculative decoding in a broader scope. We set the sampling temperature to 1, as we found that when using greedy decoding in long-form generation, both the draft and the target models are prone to repeat themselves, resulting in very low

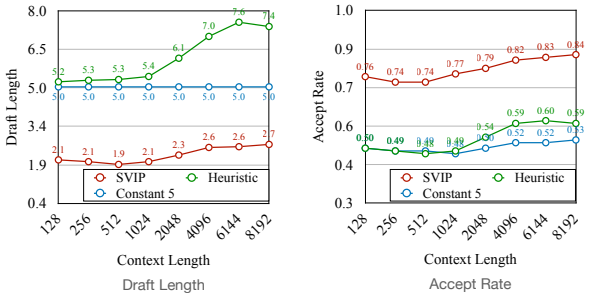
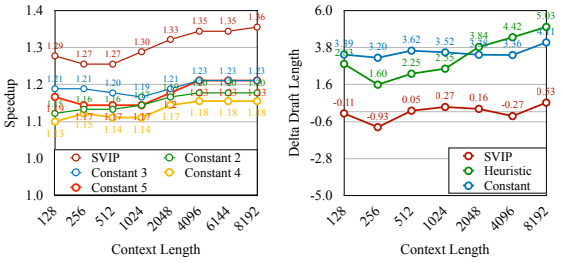


Figure 6: Analysis of Qwen2.5 14B/0.5B's behaviours on MT-Bench. Compared with constant and heuristics length policies, SVIP generates shorter drafts with a significantly higher accept rate.



(a) (b)

Figure 7: Further analysis of Qwen2.5 14B/0.5B on MT-Bench: (a) SVIP outperforms all constant length policies ranging from 2-5; (b) delta draft length (defined as proposed draft length minus oracle draft length) show that constant and heuristics length policies tend to over-generate drafts, while SVIP models the oracle draft length almost perfectly.

information entropy and exaggerated speedup ratios (Ouyang et al., 2024) (see Appendix D for details). The entropy threshold h in SVIP is chosen from $\{0.2, 0.3, 0.4, 0.5\}$ based on performance on 8 held out samples using the 14B model as target and the 0.5B model as draft, which is set to 0.3 and reused in all following experiments.

As evaluation metrics, we mainly report the average speedup over target-model-only autoregressive decoding, but also consider other auxiliary information including accepted draft lengths and draft token accept rate. Also, since the memory consumption of verifying n draft tokens is quadratic in n, we limit the maximum draft length to 40 in both heuristics and SVIP scenarios, beyond which we start to encounter out-of-memory issues.

SVIP outperforms all baselines We validate our proposed method, SVIP, with two target models: Qwen2.5-14B and Qwen2.5-32B, utilizing draft models that vary in size from 0.5B to 3B. We show the performance of SVIP and baselines in Figure 5. It shows that SVIP consistently outperforms constant and heuristics draft length by a large margin.

To further investigate the origin of SVIP's speedup, we analyze the proposed draft lengths and accepte rate of the different length policies in Figure 6. We observe that by terminating the draft process early when the draft model entropy is high, SVIP leads to shorter draft lengths and a much higher acceptance rate. However, in Figure 7(a) we also compare the performance of SVIP with shorter constant draft length policies, which suggest that simply using shorter constant draft length does not suffice, highlighting the importance of dynamically determining draft lengths. Figure 6 also suggests that while the heuristics draft length policy tends to produce very long drafts at long context, its acceptance rate remains low, resulting in no effective speedup compared with fixed draft length.

Table 2: Speedup on MT-Bench on top of EAGLE-2, using Vicuna as base models.

Model	Method	Context Length					
		128	256	512	1K	2K	4K
V-7B	E2	2.70	2.62	2.52	2.41	2.43	1.24
	+ SVIP	**2.80**	**2.76**	**2.71**	**2.69**	**2.75**	**1.41**
V-13B	E2	**2.95**	2.90	2.83	2.74	2.71	1.53
	+ SVIP	2.94	**2.99**	**2.93**	**2.86**	**2.79**	**1.64**

SVIP drafts fit oracle length In Figure 7(b), we revisit the concept of "oracle draft length" introduced in Section 2.1, and plot the differences between actual draft lengths and oracle draft lengths. The results suggest that both constant and heuristics draft length policies tend to generate drafts that are too long, while SVIP models the oracle draft length almost perfectly, with an average delta below 0.5 tokens.

SVIP further boosts strong baseline In the previous experiments, we evaluated SVIP on vanilla speculative decoding, where a standard pretrained Transformer decoder model with the same vocabulary as the target is used as the draft model. However, in the past years many works on speculative decoding have proposed other stronger or more efficient draft models (Cai et al., 2024; Du et al., 2024; Li et al., 2024b). Since most of these works assume a constant draft length, SVIP is orthogonal to them and can be applied on top of them without any additional training.

Specifically, we also apply SVIP to EAGLE-

2 (Li et al., 2024a), the current state-of-the-art (SOTA) speculative decoding system which utilizes the target model's language modeling head on top of the draft model's features to predict the next draft token, and dynamically constructs a draft tree at each draft position. Following Li et al. (2024a), we use Vicuna 7B, 13B (Chiang et al., 2023) as the base models, and set the sampling temperature to 1. To the best of our knowledge, we are also the first to investigate EAGLE-2's effectiveness in long-form generation. The results of EAGLE-2 on MT-Bench are given in Table 2. Even on top of this SOTA speculative decoding system, SVIP yields consistent improvement, which is especially notable at longer context length, surpassing the vanilla EAGLE-2 by 14% speedup for Vicuna 7B and 7% for Vicuna 13B.

3.2 Long-form Reasoning

Settings Recently, o1-style reasoning models have come into the spotlight of LLM research. Thus, we are especially interested in seeing the effectiveness of SVIP and other speculative decoding strategies on such models, which often have very long outputs. Consequently, we utilize QwQ-32B-Preview (Team, 2024), the only applicable open-source reasoning model at the time of writing, which does not have off-the-shelf smaller variants, so we train our own draft model based on Qwen2.5 1.5B by distilling QwQ 32B on 1M mathematical Persona data (Chan et al., 2024)[1]. Using this draft model, we conduct experiments on MATH (Hendrycks et al., 2021), AIME[2], and GPQA (Rein et al., 2023). We sample 200 questions from MATH ranging from level 1 to level 5, and use 73 questions released from 2022 to 2024 for AIME. For GPQA, we use the diamond test set.

SVIP achieves strong speedup in long-form reasoning The overall results are given in Table 3. SVIP outperforms the two baselines by a large margin across different benchmarks and context lengths. Detailed analysis of the different length policies' behaviours suggests similar results to the previous experiment: SVIP has an average proposal length similar to the constant draft length policy, but with a much higher draft token accept rate, leading to more effective speedup.

In Table 4, we present some token cases with very low or very high acceptance rate. We find that completing subword units or equations is quite easy for the small draft model, while several keywords in QwQ's reasoning patterns are much harder. The reverse correlation of draft entropy and acceptance rate on these tokens further validates the motivation of SVIP. In Figure 8, we plot the behaviors of different draft length policies in an example generation case. The drastic draft length oscillations in SVIP also highlight the importance of dynamic draft length policy.

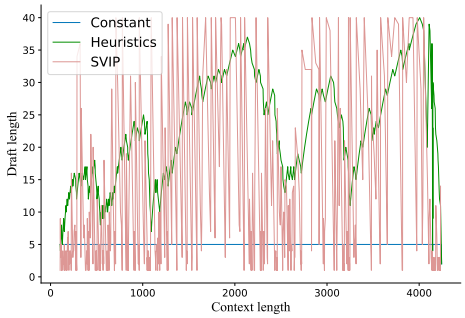


Figure 8: A case comparison of different length policies during one generation example. Drafts proposed by SVIP vary drastically in length, while constant and heuristics policies are insufficient to model such changes.

4 Related Work

Since Leviathan et al. (2023) and Chen et al. (2023) introduced speculative decoding into large language models, numerous works have followed their tracks in pursuit of more efficient LLM inference. We broadly categorize these works into three types: better draft models, draft tree expansion, and draft length control, which are orthogonal to each other. A more comprehensive review of speculative decoding is provided by Xia et al. (2024).

Better draft models. As Xia et al. (2024) suggest, draft models in speculative decoding can be either based on self-drafting or based on an independent draft model. For the first type, one may use a quantized (Zhao et al., 2024), early-exiting (Elhoushi et al., 2024), or forward-padded (Monea et al., 2023) version of the target model to produce draft tokens, while the second type is represented by the vanilla speculative decoding (Leviathan et al., 2023). Some works also take the best of both worlds and introduce extra layers on top of the target model's hidden representations to construct draft models, represented by EAGLE (Li et al., 2024b), GliDe (Du et al., 2024), and Medusa (Cai

[1]We have made this model publicly available and will provide links in camera-ready version.

[2]https://maa.org/maa-invitational-competitions/

Table 3: Speedup of QwQ on MATH, AIME, and GPQA, along with their average generation length.

	MATH500					GPQA	AIME	Avg
	Level1	Level2	Level3	Level4	Level5			
Avg. Length	1.3K	1.3K	1.6K	2.5K	3.6K	3.9K	6.2K	
Const.	1.45	1.50	1.52	1.56	1.56	1.25	1.58	1.49
Heuristics	1.29	1.26	1.27	1.30	1.33	1.18	1.34	1.28
SVIP	**1.65**	**1.68**	**1.75**	**1.78**	**1.82**	**1.52**	**1.77**	**1.71**

Table 4: The average draft model entropy and draft token acceptance rate of some representative tokens between QwQ-32B and 1.5B.

Token	Avg. Entropy	Accept Rate
All tokens	0.38	0.68
"Wait"	1.17	0.53
"Alright"	1.38	0.22
"Actually"	1.52	0.33
")]"	0.12	1.00
"}}"	0.02	1.00
"ynomials"	0.01	1.00
"ponents"	0.01	1.00

et al., 2024).

Unlike previous methods, SVIP has no requirement for draft models except for autoregression, and is a totally training-free adaptive-length policy, which could boost any draft model's performance.

Draft tree expansion. Given a draft model, one may verify multiple draft tokens for the same position in parallel to increase the probability of finding an accepted draft token, and we use "draft tree expansion" as an umbrella term for such techniques. Li et al. (2024a) introduce EAGLE-2, which reranks draft tokens in EAGLE's draft tree to select tokens with the highest confidence for verification. Similarly, CaPE (Du et al., 2024) improves GliDe by expanding the token set chosen for verification at each position based on top-1 confidence. Other works have also addressed the problem of multi-draft verification from a theoretic perspective (Sun et al., 2023; Yin et al., 2024).

In contrast to previous methods which introduce tree expansion for proposing fixed fine-grained n-gram draft, our method SVIP involves no fixed tree expansion rules, and is mainly about a more dynamic and flexible draft length policy.

Draft length control. Works in this category are few, but most relevant to ours. Liu et al. (2024) introduce PEARL, which lets the target model perform verification in parallel to draft generation, stopping the draft process when a mismatch is found. Huang et al. (2024) propose SpecDec++, which trains an acceptance prediction head on top of the draft model to predict the acceptance probability of the current draft token, stopping the draft round when the predicted acceptance probability falls below a constant threshold. Brown et al. (2024) propose Dynamic Depth Decoding (DDD) on top of EAGLE-2, which uses the sum of all tokens' confidences in one level of its draft tree as an indicator to predict whether or not to continue draft generation. Concurrent with our work, Zhang et al. (2024b) propose AdaEAGLE, which utilizes an MLP on top of EAGLE to predict the next round's draft length.

In contrast to prior length-control strategies that necessitate the training of a length-prediction module, SVIP stands out with its training-free nature. This unique characteristic endows it with remarkable flexibility, allowing it to be seamlessly integrated and applied to any autoregressive draft model.

5 Conclusion

We propose SVIP, a flexible, training-free, and plug-and-play dynamic draft length policy for speculative decoding systems. Based on a theoretical lower bound of acceptance probability and its empirical approximation, SVIP determines whether to continue draft generation or to quit drafting based on the draft model's entropy after the generation of each draft token. With extensive experiments spanning various base models, draft methods, test domains, and generation length, we validated the effectiveness of SVIP, sparking new insights on speculative decoding and more efficient large language models. For future work, we aim to investigate tighter bounds on the acceptance rate to improve the accuracy of acceptance probability estimates,

thereby enabling more efficient draft length adaptation. Additionally, our current analysis may not fully capture context-dependent patterns. A more nuanced investigation into these patterns could further enhance performance.

Limitations

While SVIP advances speculative decoding through adaptive draft length control, it has several limitations that offer avenues for future work. The acceptance rate bound in Equation (6) could be overly conservative. Developing a tighter bound would enhance the accuracy of acceptance probability estimates, enabling more effective draft length adaptation. Further, our analysis assumes they follow a simplified distribution discrepancy of SD systems. This may not fully capture the nuanced factors contributing to their occurrence, such as context-dependent patterns or model-specific biases. Context-dependent length proxy for SD systems could be the potential research direction.

Acknowledgments

This paper is supported by the General Program of National Natural Science Foundation of China (62176153).

References

Jean Bretagnolle and Catherine Huber. 1978. Estimation des densités : risque minimax. *Séminaire de probabilités de Strasbourg*, 12:342–363.

Oscar Brown, Zhengjie Wang, Andrea Do, Nikhil Mathew, and Cheng Yu. 2024. Dynamic depth decoding: Faster speculative decoding for llms. *CoRR*, abs/2409.00142.

Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D. Lee, Deming Chen, and Tri Dao. 2024. Medusa: Simple LLM inference acceleration framework with multiple decoding heads. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. 2024. Scaling synthetic data creation with 1,000,000,000 personas. *CoRR*, abs/2406.20094.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. 2023. Accelerating large language model decoding with speculative sampling. *CoRR*, abs/2302.01318.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Cunxiao Du, Jing Jiang, Yuanchen Xu, Jiawei Wu, Sicheng Yu, Yongqi Li, Shenggui Li, Kai Xu, Liqiang Nie, Zhaopeng Tu, and Yang You. 2024. Glide with a cape: A low-hassle method to accelerate speculative decoding. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Mostafa Elhoushi, Akshat Shrivastava, Diana Liskovich, Basil Hosmer, Bram Wasti, Liangzhen Lai, Anas Mahmoud, Bilge Acun, Saurabh Agarwal, Ahmed Roman, Ahmed A Aly, Beidi Chen, and Carole-Jean Wu. 2024. Layerskip: Enabling early exit inference and self-speculative decoding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 12622–12642. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Kaixuan Huang, Xudong Guo, and Mengdi Wang. 2024. Specdec++: Boosting speculative decoding via adaptive candidate lengths. *CoRR*, abs/2405.19715.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2023. Fast inference from transformers via speculative decoding. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 19274–19286. PMLR.

Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. 2024a. EAGLE-2: faster inference of language models with dynamic draft trees. *CoRR*, abs/2406.16858.

Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. 2024b. EAGLE: speculative sampling requires rethinking feature uncertainty. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Tianyu Liu, Yun Li, Qitan Lv, Kai Liu, Jianchen Zhu, and Winston Hu. 2024. Parallel speculative decoding with adaptive draft length. *CoRR*, abs/2408.11850.

Xiaofan Lu, Yixiao Zeng, Feiyang Ma, Zixu Yu, and Marco Levorato. 2024. Improving multi-candidate speculative decoding. *CoRR*, abs/2409.10644.

Giovanni Monea, Armand Joulin, and Edouard Grave. 2023. Pass: Parallel speculative sampling. *CoRR*, abs/2311.13581.

Siru Ouyang, Shuohang Wang, Minhao Jiang, Ming Zhong, Donghan Yu, Jiawei Han, and Yelong Shen. 2024. Temperature-centric investigation of speculative decoding with knowledge distillation. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 13125–13137. Association for Computational Linguistics.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2023. GPQA: A graduate-level google-proof q&a benchmark. *CoRR*, abs/2311.12022.

Ziteng Sun, Ananda Theertha Suresh, Jae Hun Ro, Ahmad Beirami, Himanshu Jain, and Felix X. Yu. 2023. Spectr: Fast speculative decoding via optimal transport. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Qwen Team. 2024. Qwq: Reflect deeply on the boundaries of the unknown.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Heming Xia, Zhe Yang, Qingxiu Dong, Peiyi Wang, Yongqi Li, Tao Ge, Tianyu Liu, Wenjie Li, and Zhifang Sui. 2024. Unlocking efficiency in large language model inference: A comprehensive survey of speculative decoding. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 7655–7671. Association for Computational Linguistics.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. Qwen2 technical report. *CoRR*, abs/2407.10671.

Ming Yin, Minshuo Chen, Kaixuan Huang, and Mengdi Wang. 2024. A theoretical perspective for speculative decoding algorithm. *CoRR*, arXiv:2411.00841.

Jun Zhang, Jue Wang, Huan Li, Lidan Shou, Ke Chen, Gang Chen, and Sharad Mehrotra. 2024a. Draft& verify: Lossless large language model acceleration via self-speculative decoding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 11263–11282. Association for Computational Linguistics.

Situo Zhang, Hankun Wang, Da Ma, Zichen Zhu, Lu Chen, Kunyao Lan, and Kai Yu. 2024b. Adaeagle: Optimizing speculative decoding via explicit modeling of adaptive draft structures. *CoRR*, abs/2412.18910.

Juntao Zhao, Wenhao Lu, Sheng Wang, Lingpeng Kong, and Chuan Wu. 2024. Qspec: Speculative decoding with complementary quantization schemes. *CoRR*, abs/2410.11305.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

A The Complete Speculative Decoding Algorithms

In Algorithm 2 to 6, we present the complete algorithms of the vanilla speculative decoding in both the greedy decoding and the sampling scenarios. For the sampling scenario, the `Verify` and `Correct` methods in Algorithm 6 resolve to Algorithm 2 and 4. For greedy decoding, they resolve to Algorithm 3 and 5.

Algorithm 2 Verify (Sampling)

Input: target distribution $p(x)$, draft distribution $q(x)$, draft token x_t

$accept \leftarrow$ False
$r \sim U[0,1]$
if $r < \frac{p(x_t)}{q(x_t)}$ **then**
    $accept \leftarrow$ True
end if

Output: $accept$

Algorithm 3 Verify (Greedy)

Input: target distribution $p(x)$, draft distribution $q(x)$, draft token x_t

$accept \leftarrow$ False
if $\arg\max p(x) == x_t$ **then**
    $accept \leftarrow$ True
end if

Output: $accept$

Algorithm 4 Correct (Sampling)

Input: target distribution $p(x)$, draft distribution $q(x)$

Sample $\hat{x} \sim \frac{\max(q(x)-p(x),0)}{\sum_i \max(q(x^i)-p(x^i),0)}$

Output: $\hat{x}$

Algorithm 5 Correct (Greedy)

Input: target distribution $p(x)$, draft distribution $q(x)$

Output: $\arg\max p(x)$

Algorithm 6 Speculative Decoding

Input: target model p, draft model q, input sequence $x_{\leqslant t}$, maximum length T, draft length γ

Initialize $n \leftarrow t$
while $n < T$ **do**
    for $j = 1$ to γ **do**
        Sample $x_{n+j} \sim q(x|x_{<n+j})$
    end for
    Compute $p(x|x_{<n+j}),\ j = 1, \cdots, \gamma+1$ in parallel
    $\tilde{n} \leftarrow n$
    for $j = 1$ to γ **do**
        if Verify$(p(x|x_{<n+j}),\ q(x|x_{<n+j}), x_{n+j})$ **then**
            $\tilde{n} \leftarrow \tilde{n} + 1$
        else
            $x_{n+j} \leftarrow$ Correct $(p(x|x_{<n+j}),\ q(x|x_{<n+j}))$
            Exit for loop
        end if
    end for
    if $\tilde{n} == n + \gamma$ **then**
        $x_{n+\gamma+1} \sim p(x|x_{\leqslant n+\gamma})$
    end if
    $n \leftarrow \tilde{n} + 1$
end while

Output: $x_{\leqslant n}$

We note that from Algorithm 2, it's straightforward that the acceptance rate of a draft token x_t in the sampling scenario is by definition $\min(1, \frac{p(x_t)}{q(x_t)})$.

B Alternatives for Acceptance Rate Lower Bound Computation

In Section 2, we used Pinsker's inequality to compute a lower bound for the expected acceptance probability:

$$\beta = \sum_x \min\left(p(x), q(x)\right) \tag{10}$$

$$\geqslant 1 - \sqrt{\frac{1}{2}\mathbb{KL}(q||p)}. \tag{11}$$

Another way to compute the lower bound of acceptance probability can be derived from Bretagnolle-Huber inequality (Bretagnolle and Huber, 1978):

$$\beta \geqslant 1 - \sqrt{1 - e^{-\mathbb{KL}(q||p)}}. \tag{12}$$

Compared with the Pinsker's bound, it's trivial to see that this bound is guaranteed to be always larger than 0. However, in practice we find that the Pinsker's bound is about 11% tighter.

C γ Approximation

C.1 Approximation Bound

Following Eq. (4), the acceptance rate β satisfies:

$$\beta \geq 1 - \sqrt{\frac{1}{2}\mathbb{KL}(q||p)} = 1 - \sqrt{\frac{1}{2}H_{q,p} - \frac{1}{2}H_q}$$

We denote the above bound as the *actual bound*. While this bound is theoretically sound, it relies on the exact access to the $H_{q,p}$ the cross entropy between target and the draft models, which is inaccessible during the SD drafting phase. To address this, approximate $H_{q,p}$ with γH_q, where γ is a random variable to describe the ratio between $H_{q,p}$ and H_q, i.e. $\gamma = H_{q,p}/H_q$, we could rewrite the bound:

$$\mathbb{KL}(q||p) = H_{q,p} - H_q = (\gamma - 1)H_q$$
$$\beta \geq 1 - \sqrt{\frac{1}{2}(\gamma - 1)H_q}$$

To make this bound more practical, we approximate it using a constant c, obtaining

$$\beta \geq 1 - \sqrt{\frac{1}{2}(\gamma - 1)H_q} \approx 1 - \sqrt{cH_q}$$

We denote the above bound as the *approximation bound*. Since γ is a random variable, the tightness and reliability of this approximation depend on how well c aligns with γ's behavior. Specifically, we need our approximation bound is smaller than the actual bound:

$$\beta \geq 1 - \sqrt{\frac{1}{2}(\gamma - 1)H_q} \geq 1 - \sqrt{cH_q}$$

Simplify the right side inequality:

$$\gamma \leq 2c + 1 \tag{13}$$

C.2 Theoretical Analysis

Now, from the γ's distribution in Figure 5, let's analyze the probability that the lower bounds hold by modeling γ's distribution.

Gaussian Distribution Let's assume $\gamma \sim N(\mu, \sigma^2)$, and the probability that the bound holds is:

$$P(\gamma \leq 2c + 1) = \Phi(\frac{2c + 1 - \mu}{\sigma}) \tag{14}$$

where Φ is the standard normal CDF. It demonstrates that:

- If c is small (e.g. $2c + 1 \leq \mu$), the probability that the bound holds is low.
- If c is large (e.g. $2c + 1 \geq \mu$), the bound holds with high probability; however the bound itself becomes loose.

Gamma Distribution Given the right-skewed nature of $\gamma \geq 1$, we model as a shifted Gamma distribution: $\gamma = 1 + X$, where $X \sim \text{Gamma}(\alpha, \beta)$. The conditions for the bound to hold is :

$$P(\gamma \leq 2c + 1) = P(X \leq 2c) = \frac{\gamma(\alpha, \beta \cdot 2c)}{\Gamma(\alpha)} \tag{15}$$

where $\gamma(\alpha, z) = \int_0^z t^{\alpha-1}e^{-t}dt$ is the lower incomplete gamma function, and $\Gamma(\alpha)$ is the gamma function. This probability depends on c, α, and β.

- If c is small (e.g. $2c < \mathbb{E}[x] = \alpha/\beta$), so the probability that our approximation further lowers the actual bound is low.
- It c is large, the probability approaches 1, but the bound $\beta \geq 1 - \sqrt{cH_q}$ is looser.

D Additional Results on Long-form Generation

In Table 5, we present the results of greedy decoding using Qwen2.5 14B as target and 0.5B as draft, and find that the speedup ratio of greedy decoding is much higher compared with the sampling experiments in Section 3.1. Further investigation suggests that this is a result of repetition hallucination in both target and draft models during long-form greedy generation.

Table 5: Results of greedy decoding on MT-Bench. Greedy decoding leads to repetition hallucinations in both target and draft models in long-form generation, resulting in exaggerated speedup ratio.

Model	**Methods**	**Context Length**							
		128	256	512	1K	2K	4K	6K	8K
Qwen2.5 (14B, 0.5B)	Const.	1.05	1.08	1.15	1.29	1.44	1.54	1.60	1.67
	Heuristics	1.04	1.06	1.13	1.32	1.54	1.72	1.85	1.97
	SVIP	**1.30**	**1.34**	**1.42**	**1.57**	**1.74**	**1.87**	**1.98**	**2.10**