Fast Large Language Model Collaborative Decoding via Speculation

Presenter Jiale Fu
Affiliation Southeast University
Email jiale.fu@seu.edu.cn

June 11, 2025

Outline

1. Background

We all know that large language models generate text through the *"next token prediction"* mechanism.

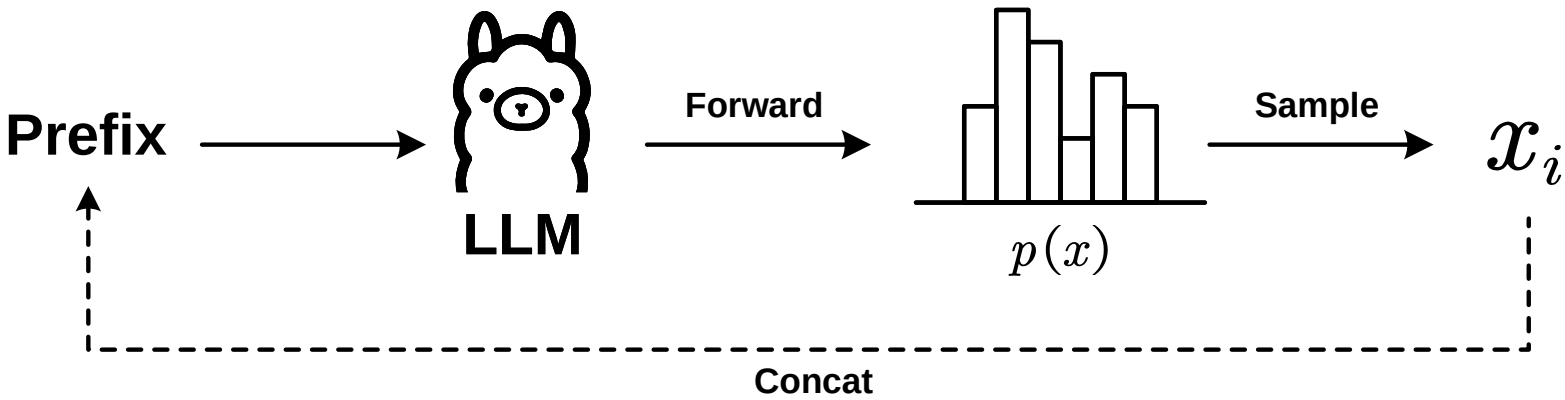


Figure: Large Language Model Collaborative Decoding.

A natural approach to improve generation quality is: *to use multiple LLMs*.

Specifically, at generation each step, each LLM separately performs a forward pass. Then, we combine their results to get a better prediction.

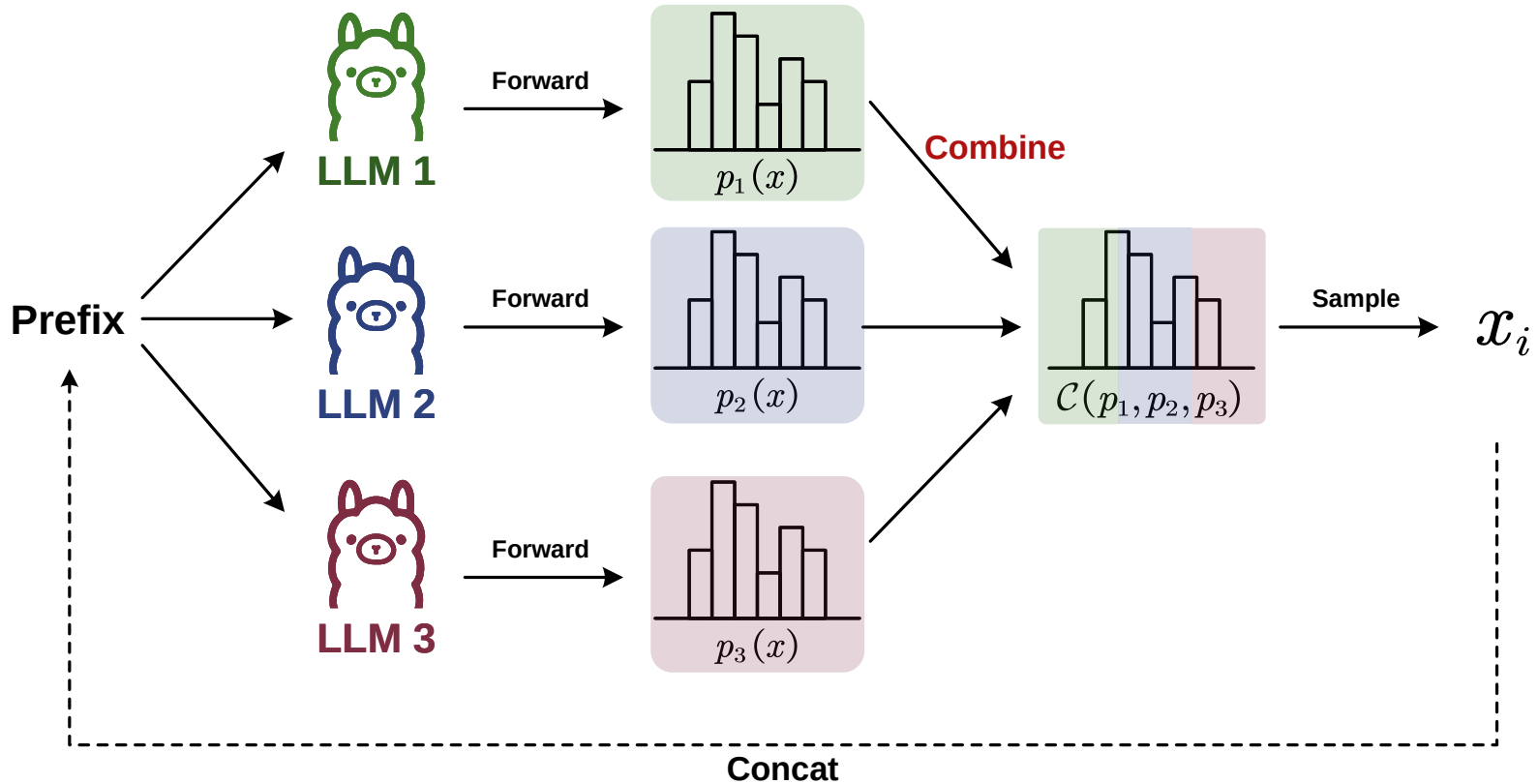


Figure: Large Language Model Collaborative Decoding.

We refer to such approaches as collaborative decoding.

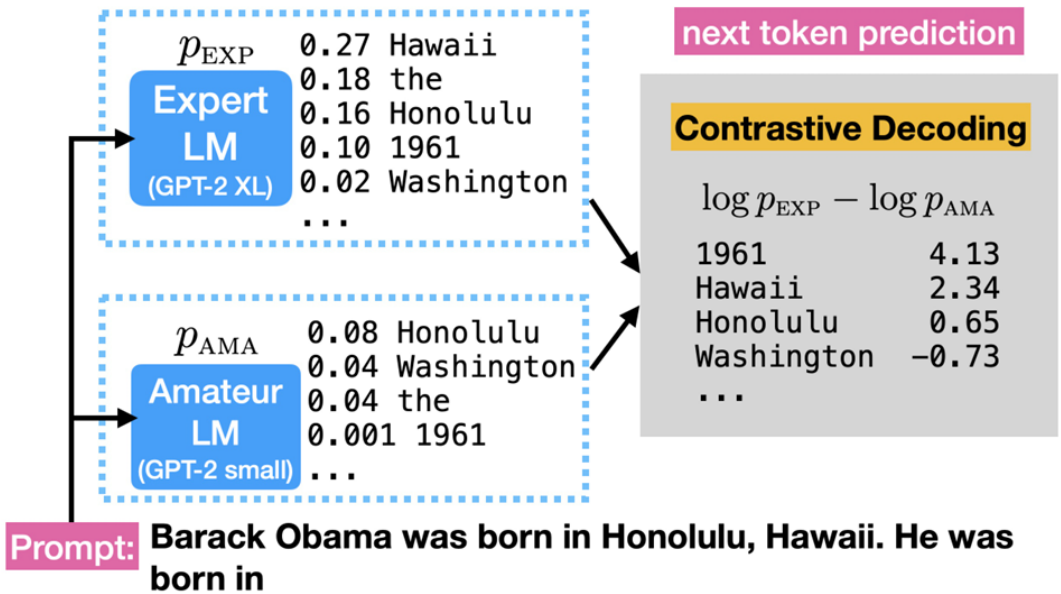
Three examples of combination:

1. Weighted ensemble:

$$\mathcal{C}(p_1(x), p_2(x), p_3(x)) = \frac{1}{3}\left(p_1(x) + p_2(x) + p_3(x)\right). \quad (1)$$

2. Contrastive Decoding[1]:

$$\mathcal{C}\left(p(x), q(x)\right) = \mathrm{Softmax}(\log p(x) - \log q(x)). \quad (2)$$



[1] Xiang Lisa Li et al. "Contrastive Decoding: Open-ended Text Generation as Optimization". In: *The 61st Annual Meeting Of The Association For Computational Linguistics*. 2023.

Three examples of combination:

3. Decoding-Time Realignment[2]:

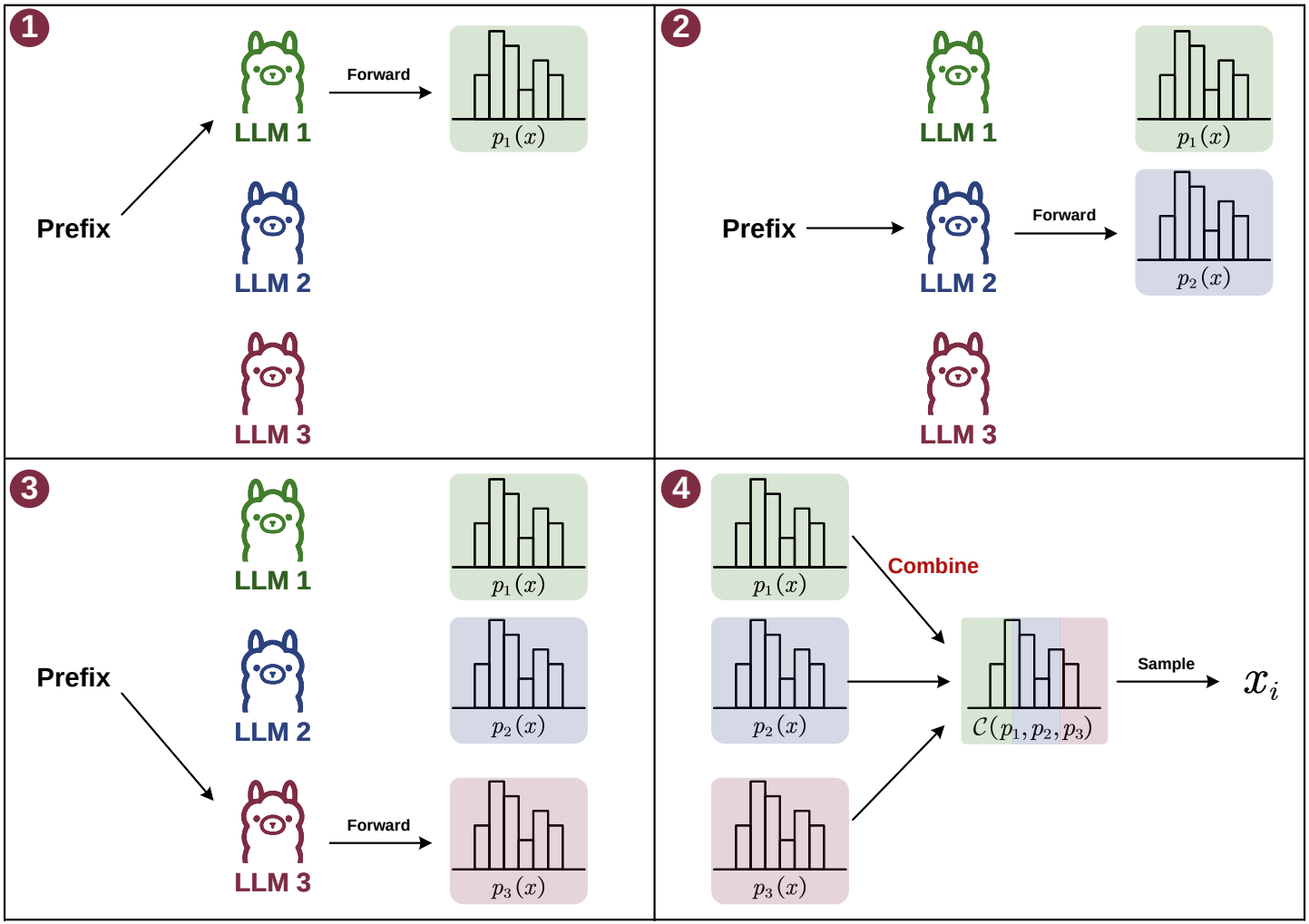
$$\mathcal{C}\left(p(x), q(x)\right) = \mathrm{Softmax}\left(\lambda l_p + (1-\lambda) l_q\right). \tag{3}$$

where l_p corresponds to the logits from an aligned model(with RLHF), and l_q from an unaligned model (without RLHF).

[2] Tianlin Liu et al. "Decoding-time Realignment of Language Models". In: *International Conference on Machine Learning*. 2024, pp. 31015–31031.

However, collaborative decoding faces an *efficiency challenge*.

At each step, the LLMs are usually invoked sequentially, which means ***3** forward passes are needed to generate one token*, which takes ***3x** longer inference time*.



You might ask:

Why don't we use a parallel implementation?

We found that due to communication delays between GPUs, *parallel decoding can actually be slower*. So, existing collaborative decoding methods typically use a sequential design.

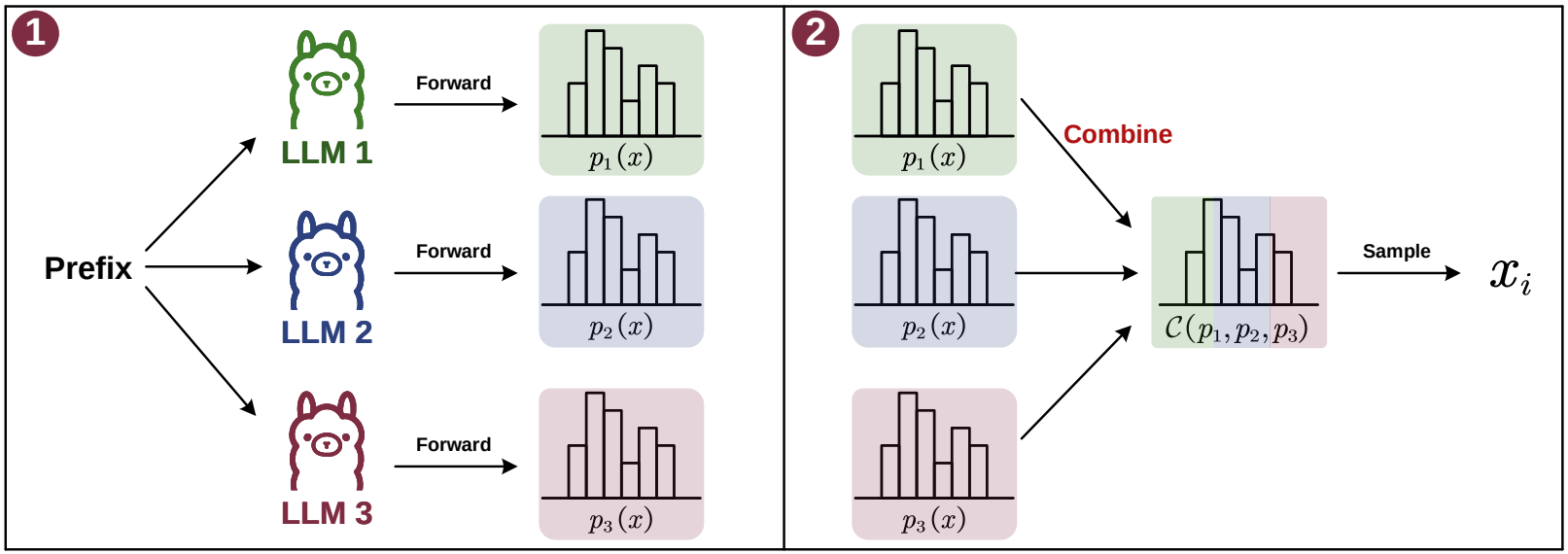


Figure: The parallel implemtation of collaborative decoding.

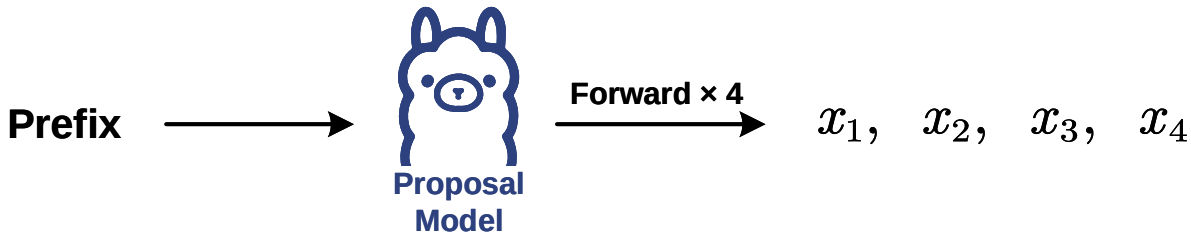
This naturally raises a question:

Can we accelerate LLM collaborative decoding without sacrificing its quality?

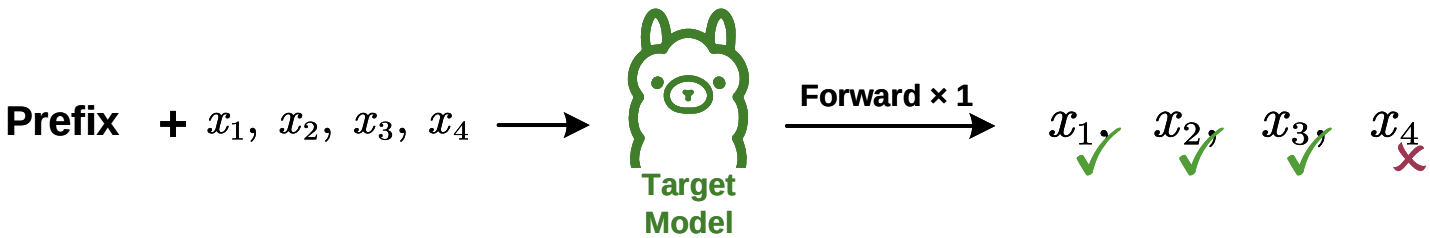
2. Collaborative Decoding via Speculation

Speculative Decoding (SD)[3] uses a proposal and verify process to accelerate LLM inference while *keeping the generation quality*.

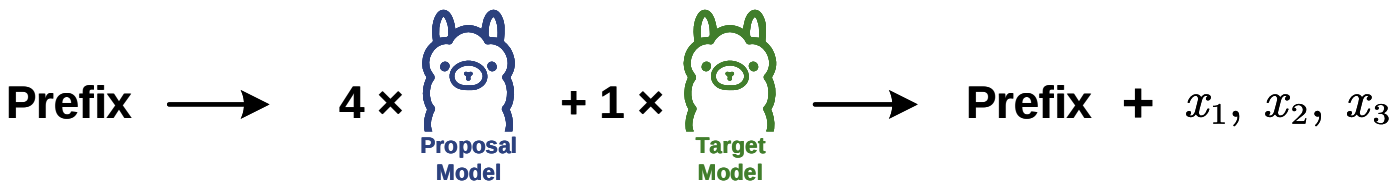
Proposal Phase:



Verification Phase:



Overall Effect:



[3] Heming Xia et al. "Speculative Decoding: Exploiting Speculative Execution for Accelerating Seq2seq Generation". In: *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Accacceptance-rejection Criteria

Let $q(x_i)$ and $p(x_i)$ be the distributions from the proposal and target models, respectively. SD's acceptance-rejection criteria is:

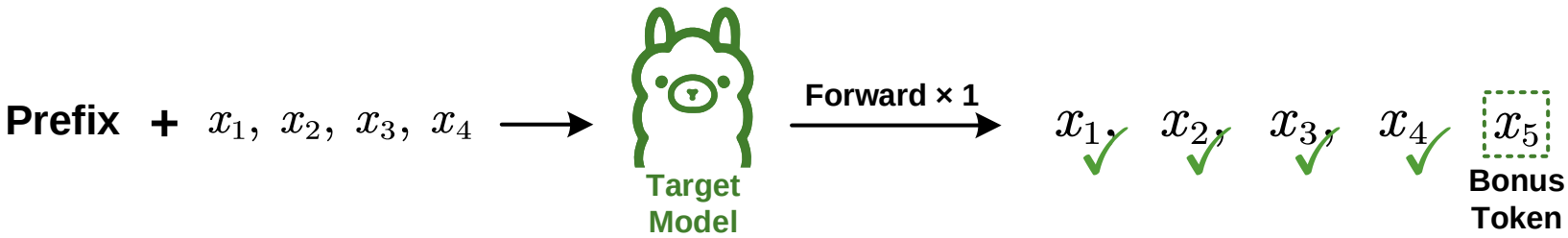
1. Sample $u \sim \mathcal{U}(0, 1)$;
2. If $u \leq \min\left(1, \frac{p(x_i)}{q(x_i)}\right)$, accept x_i;
3. Otherwise, sample a x_i from $\mathrm{norm}\left(p(x_i) - q(x_i)\right)_+$ and reject all remaining tokens.

This acceptance-rejection criteria ensures that *the generated tokens strictly follow the target model's distribution*[4] (i.e. *lossless acceleration*).

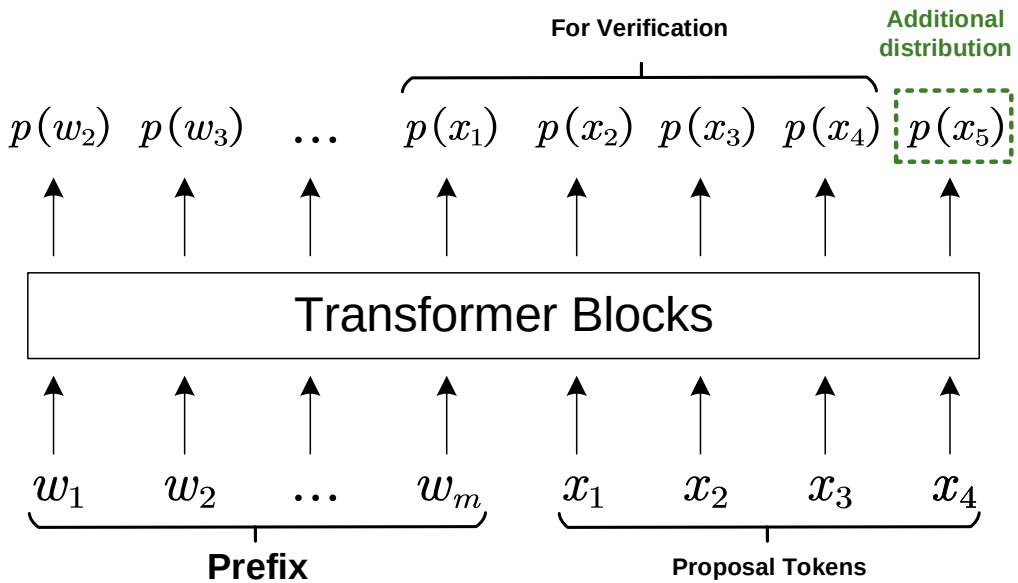
[4] Yaniv Leviathan, Matan Kalman, and Yossi Matias. "Fast inference from transformers via speculative decoding". In: *International Conference on Machine Learning*. PMLR. 2023, pp. 19274–19286.

Bonus Token

In SD, if all proposal tokens are accepted, the target model will naturally generate an additional token — we call this the *bonus token*.



Why does the bonus token appear?



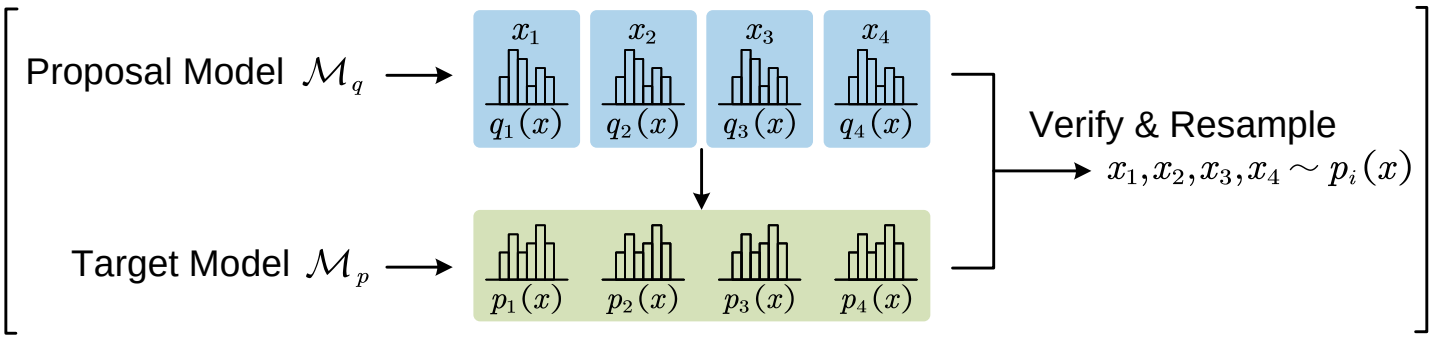
Collaborative Decoding via Speculation Naive-CoS

We made two improvements to the original SD to better fit the collaborative decoding setting:

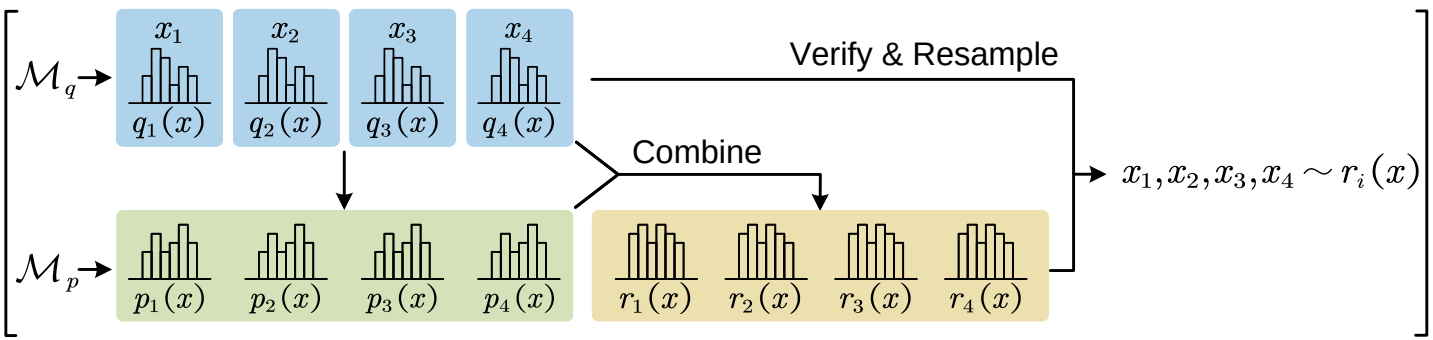
Firstly, we find that we can use the combined distribution of proposal model and target model for verification instead of the target distribution. This makes the generated tokens follow the combined distribution $\mathcal{C}(p, q)$.

In other words, if we want to ensemble two LLMs $\mathcal{M}_q$ and $\mathcal{M}_q$, we can treat one as proposal model, the other as target model, and use their combined distribution for verification. Then, we get a accelerated verision of collaborative decoding. We call this *Naive-CoS*.

(b) Speculative Decoding:



(c) Collaborative Decoding via Speculation:



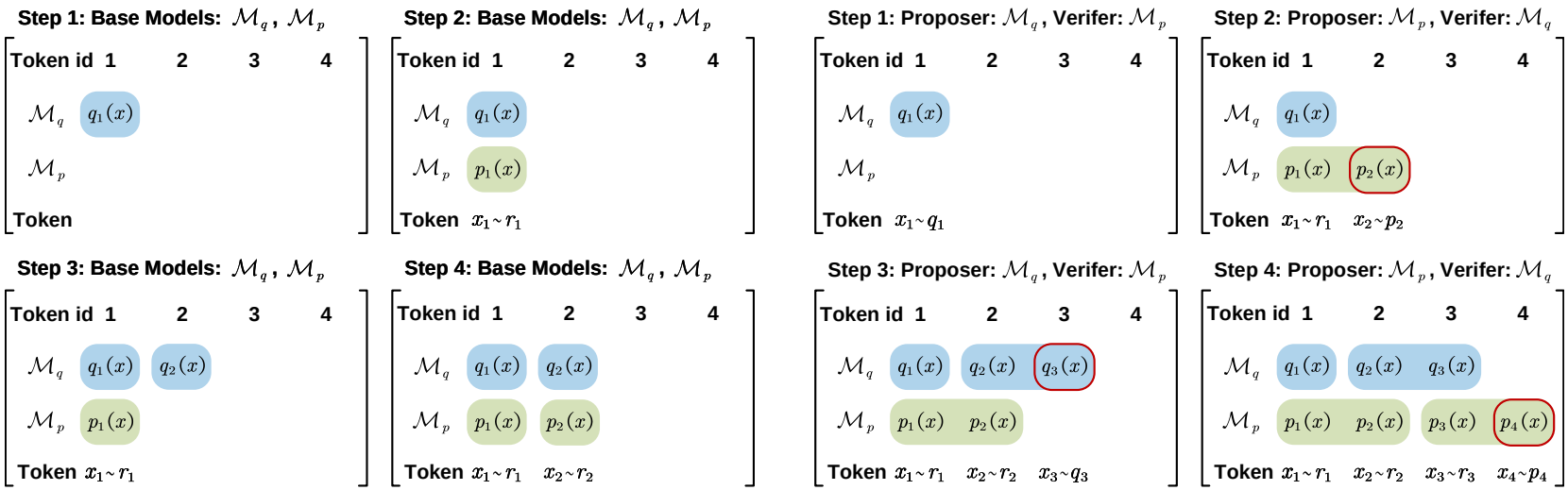
Modified Accacceptance-rejection Criteria

Let $q(x_i)$ and $p(x_i)$ be the distributions from the proposal and target models, respectively. SD's acceptance-rejection criteria is:

1. Sample $u \sim \mathcal{U}(0, 1)$;
2. If $u \leq \min\left(1, \frac{\cancel{p(x_i)}\, \mathcal{C}(p,q)}{q(x_i)}\right)$, accept x_i;
3. Otherwise, sample a x_i from $\text{norm}\left(\cancel{p(x_i)}\, \mathcal{C}(p, q) - q(x_i)\right)_+$ and reject all remaining tokens.

This acceptance-rejection criteria ensures that the generated tokens strictly follow the ~~target model's distribution~~ *combined distribution*.

Secondly, we propose an *alternative proposal framework*, which further accelerates collaborative decoding based on Naive-CoS.



(a) Standard Collaborative Decoding
1 distribution per invokation

(b) Alternative Proposal Framework
2 distribution per invokation

Figure: The sketch of Alternate Proposal Framework. For clarity, we assume that the proposal length for each model is 1 and that all proposed tokens are accepted.

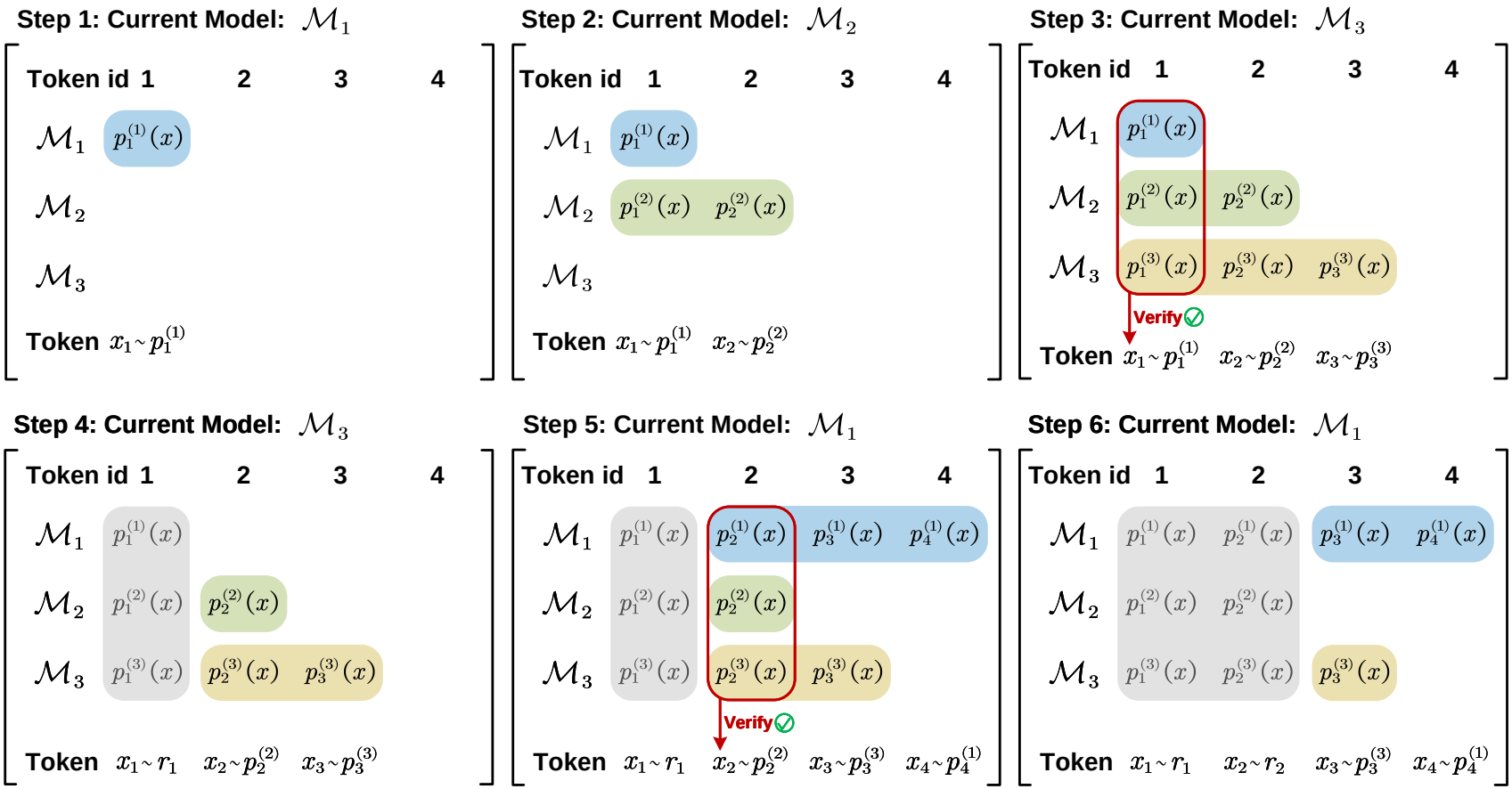


Figure: An example in 3-model scenario.

A more general example

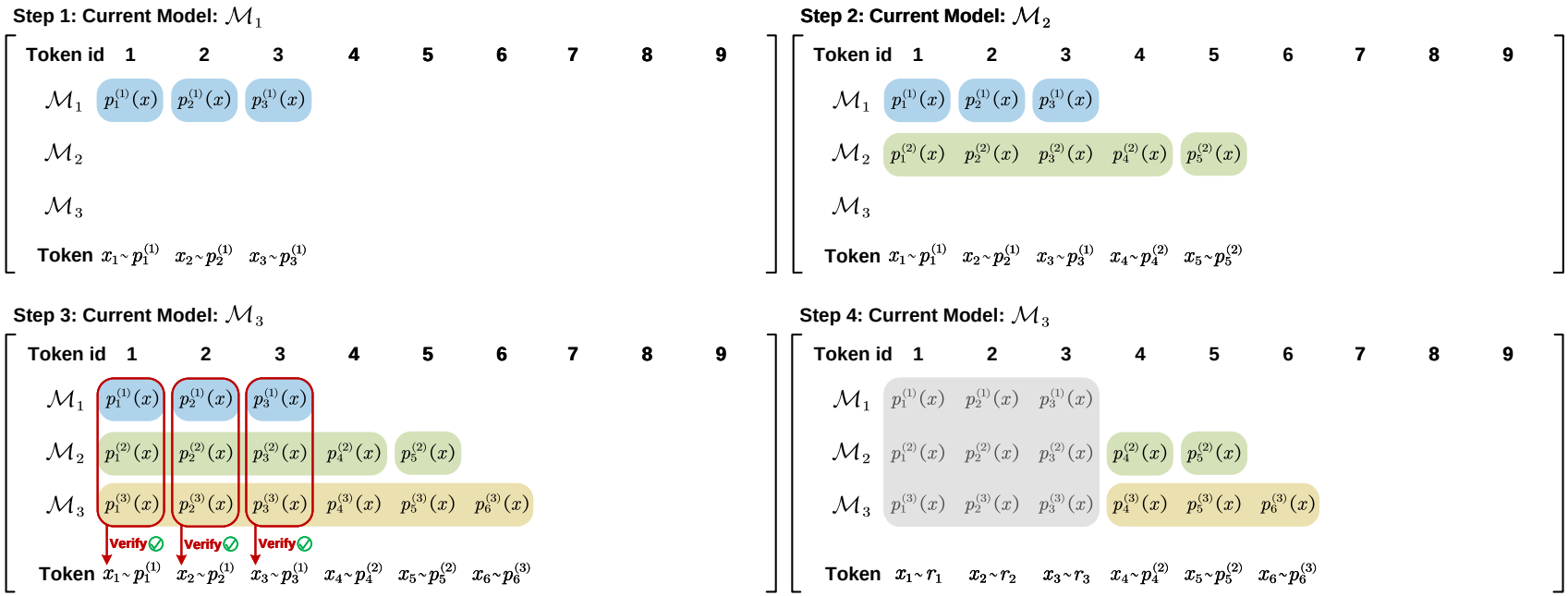


Figure: A more general example of CoS in 3-model ensemble scenario. The proposal length for model $\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3$ is 3, 2, 1, respectively.

A more general example

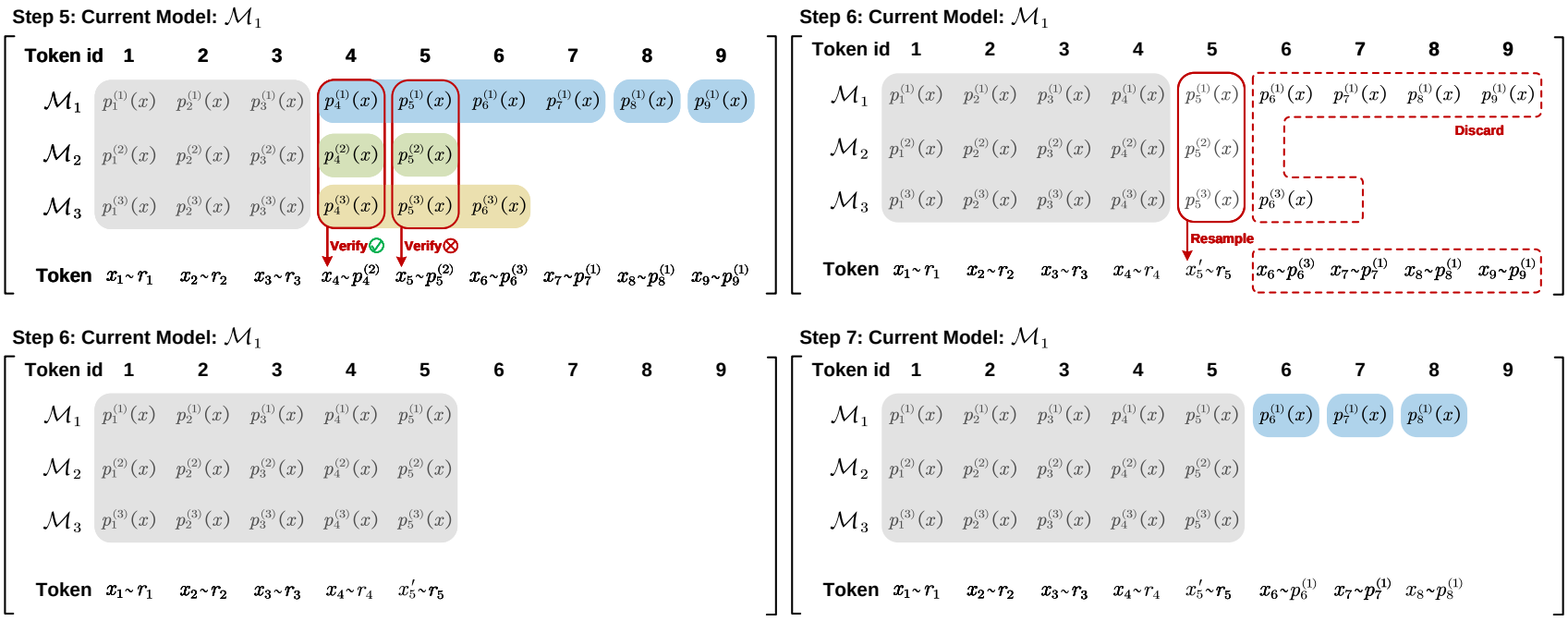
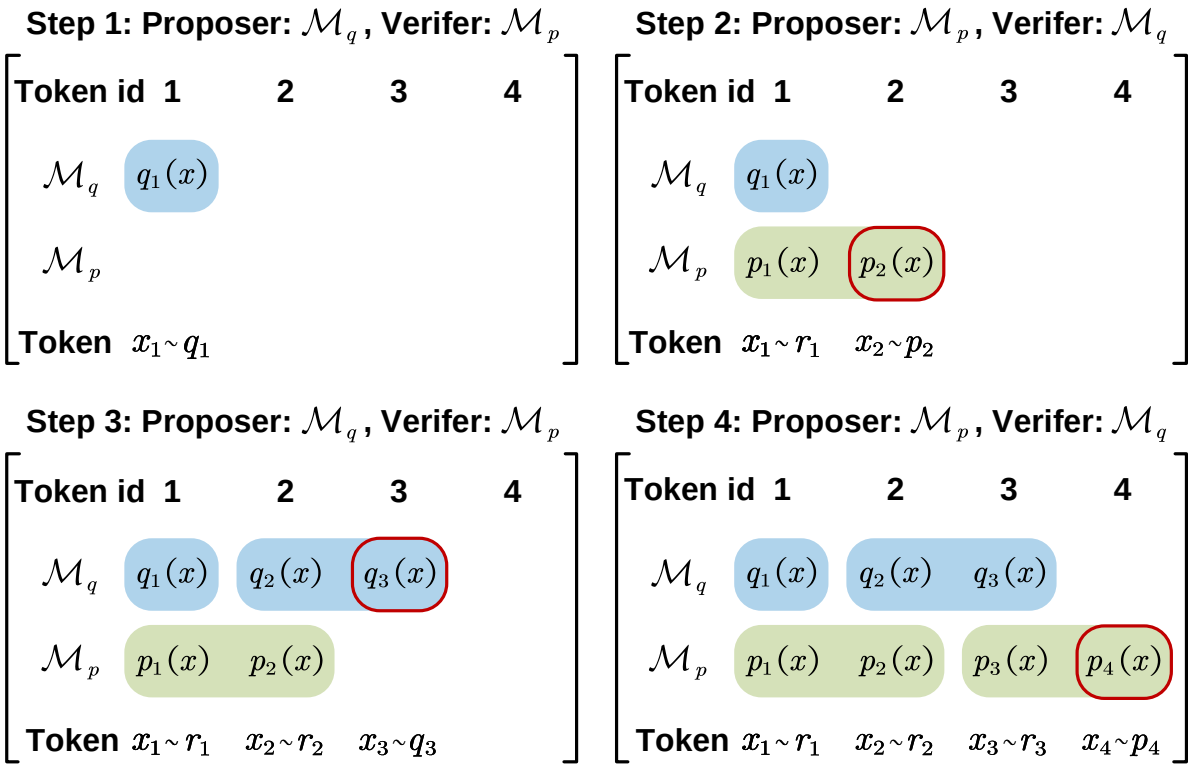


Figure: A more general example of CoS in 3-model ensemble scenario. The proposal length for model $\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3$ is 3, 2, 1, respectively.

SE has following properties:

- **Losslessness** (Appendix A.1);
- **Acceleration**: CoS is never slower than the standard collabrative docoding (Corollary 3.7);



- **Lower Bound**: In weighted ensemble (WE) setting, $(\lambda p(x) + (1-\lambda) q(x))$, the acceptance rate α has a lower bound of λ (Theorem 3.2).

Collaborative Decoding Setup:

1. Weighted Ensemble (WE)
2. Contrastive Decoding (CD)

Methods Compared:

1. Sequential Collaborative Decoding (WE, CD)
2. Parallel Collaborative Decoding (WE-P, CD-P)
3. Directly accelerated Collaborative Decoding using SD (WE-SD, CD-SD)
4. Collaborative Decoding via Speculation (CoS)

Model Configuration:

- 2 model setting:

	Name	$\mathcal{M}_q$	$\mathcal{M}_p$
WE	Llama-Vicuna	Llama-2-7B	Vicuna-7B-V1.5
	Qwen-3b	Qwen2.5-3B-Instruct	Qwen2.5-Coder-3B-Instruct
	Qwen-1.5b	Qwen2.5-1.5B-Instruct	Qwen2.5-Coder-1.5B-Instruct
CD	Llama-3	Llama-3.2-1B	Llama-3.1-8B-Instruct
	Llama-2	Llama-68M	Llama-2-7B
	OPT	OPT-125M	OPT-13B

- 3 model setting: Qwen2.5-1.5B-Instruct and its code and math versions

Table: The speedup ratio of each method in WE setting.

	Method	HumanEval	GSM8K	MMLU	CNNDM
Llama Vicuna	WE	1.00x	1.00x	1.00x	1.00x
	WE-P	0.69x	0.73x	0.70x	0.75x
	SD	1.27x	1.21x	1.19x	1.15x
	CoS	**1.58x**	**1.52x**	**1.41x**	**1.46x**
Qwen-3b	WE	1.00x	1.00x	1.00x	1.00x
	WE-P	0.74x	0.79x	0.79x	0.77
	SD	1.13x	1.06x	1.09x	1.08x
	CoS	**1.62x**	**1.52x**	**1.42x**	**1.38x**
Qwen-1.5b	WE	1.00x	1.00x	1.00x	1.00x
	WE-P	0.63x	0.62x	0.64x	0.63x
	SD	1.11x	1.13x	1.08x	1.10x
	CoS	**1.56x**	**1.46x**	**1.34x**	**1.35x**
Qwen-1.5b (3 Model)	WE	1.00x	1.00x	1.00x	1.00x
	WE-P	0.54x	0.73x	0.80x	0.82x
	SD	0.96x	0.92x	0.98x	0.95x
	CoS	**1.85x**	**1.53x**	**1.38x**	**1.27x**

Table: The speedup ratio of each method in CD setting.

	T	Method	HumanEval	GSM8K	MMLU	CNNDM
Llama-3	0	CD	1.00x	1.00x	1.00x	1.00x
		CD-P	0.41x	0.40x	0.41x	0.41x
		SD	2.04x	1.81x	1.52x	1.58x
		CoS	**2.23x**	**2.00x**	**1.77x**	**1.61x**
	1	CD	1.00x	1.00x	1.00x	1.00x
		CD-P	0.39x	0.41x	0.42x	0.41x
		SD	1.55x	1.21x	1.20x	1.07x
		CoS	**1.65x**	**1.44x**	**1.31x**	**1.18x**
Llama-2	0	CD	1.00x	1.00x	1.00x	1.00x
		CD-P	0.59x	0.50x	0.54x	0.48x
		SD	1.15x	1.62x	1.08x	0.93x
		CoS	**1.26x**	**1.65x**	**1.68x**	**1.30x**
	1	CD	1.00x	1.00x	1.00x	1.00x
		CD-P	0.56x	0.51x	0.53x	0.49x
		SD	0.94x	1.16x	1.23x	1.10x
		CoS	**1.15x**	**1.20x**	**1.37x**	**1.11x**

References

[1] Yaniv Leviathan, Matan Kalman, and Yossi Matias. “Fast inference from transformers via speculative decoding”. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 19274–19286.

[2] Xiang Lisa Li et al. “Contrastive Decoding: Open-ended Text Generation as Optimization”. In: *The 61st Annual Meeting Of The Association For Computational Linguistics*. 2023.

[3] Tianlin Liu et al. “Decoding-time Realignment of Language Models”. In: *International Conference on Machine Learning*. 2024, pp. 31015–31031.

[4] Heming Xia et al. “Speculative Decoding: Exploiting Speculative Execution for Accelerating Seq2seq Generation”. In: *The 2023 Conference on Empirical Methods in Natural Language Processing*.