

# Fast and Slow Generating: An Empirical Study on Large and Small Language Models Collaborative Decoding

Anonymous ACL submission

## Abstract

Collaborative decoding between large and small language models (LLMs/SLMs) is a key strategy to overcome LLM limitations in training and inference efficiency. While methods like speculative decoding and proxy tuning exist, a unifying understanding of these approaches is needed. Inspired by dual-process theory, we introduce FS-GEN, a framework defining LLMs as “System 2” (deliberate) and SLMs as “System 1” (intuitive). FS-GEN provides a unified lens to analyze collaborative decoding, revealing that minimal System 2 intervention ( $< 20\%$  of tokens in the generated completions) is often sufficient. We uncover a parameter-ratio scaling law governing this interaction and demonstrate that the effectiveness of collaboration hinges on the uncertainty of System 1’s next-token predictions. This uncertainty-centric view offers novel insights into optimizing collaborative decoding and developing more efficient and reliable language generation systems.

## 1 Introduction

In recent years, the landscape of artificial intelligence has been dramatically reshaped by the development of Large Language Models (LLMs) like GPT-4 (Achiam et al., 2023), Gemini (Team et al., 2023), Llama (Touvron et al., 2023a,b; AI@Meta, 2024), and Qwen (Bai et al., 2023). These models have emerged as versatile tools, enhancing software development (Qian et al., 2023; Yang et al., 2024), and accelerating scientific discovery (AI4Science and Quantum, 2023; Qi et al., 2023) in different domains (Boiko et al., 2023; Romera-Paredes et al., 2024). Despite their advancements, these models face significant challenges that affect their practical application. Key issues include managing inference time with constraints such as low-latency requirements (Miao et al., 2023; Zhou et al., 2024), and addressing concerns related to hallucinations

in generated content (Huang et al., 2023). Furthermore, the training of these models entails considerable resource consumption (Hu et al., 2021; Hoffmann et al., 2022; Ding et al., 2023), particularly during fine-tuning and in achieving domain specialization (Bommasani et al., 2021).

A novel trend in mitigating these challenges involves the collaborative decoding of large and small models. This approach leverages the strengths of both model sizes to improve efficiency and effectiveness across various downstream applications. In these methods, the next token during generation depends on both LLMs and SLMs, whose logits are combined to form the final predictions. For instance, speculative decoding (Leviathan et al., 2023; Kim et al., 2024; Sun et al., 2024; Xia et al., 2024) allows small models to swiftly generate initial drafts, which are then refined by large models, significantly cutting down on inference time. Similarly, in contrastive decoding (Li et al., 2022; O’Brien and Lewis, 2023; Sennrich et al., 2023), small models contribute amateur tokens as negative signals to enhance the decoding. Proxy tuning (Mitchell et al., 2023; Liu et al., 2024a) aims to reduce training costs by applying the delta logits from proxy small models to the base large models, rather than directly tuning the large models. This method significantly reduces training costs.

However, while the use of large and small models in tandem is increasingly common, the research community lacks a unified framework to analyze and understand the dynamics of their collaboration. This gap in knowledge motivates our current study, which seeks to formalize the collaboration between these models from the perspective of Systems 1 and System 2—a framework typically applied to modeling human cognitive processes (Kahneman, 2011). As shown in Figure 1, within the human brain, System 1 operates quickly and intuitively, automatically addressing everyday problems such as pattern recognition. Conversely, System 2 func-

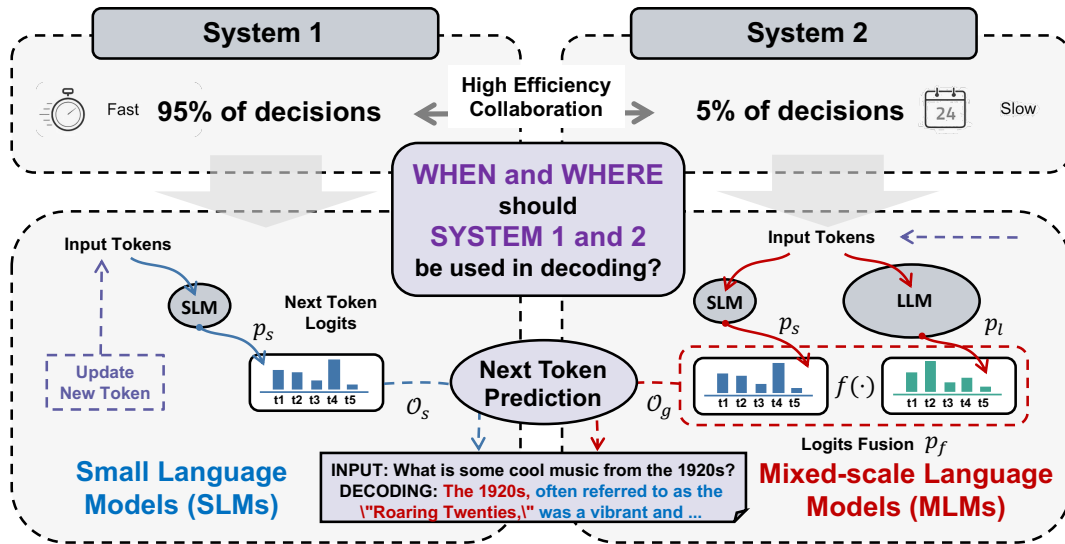


Figure 1: In the fast-and-slow generation framework, we conceptualize small language models as System 1 and a mixed-scale compound system of large and small language models as System 2. This compound system incorporates various collaborative decoding methods, such as speculative decoding, contrastive decoding, and proxy tuning. In this paper, we investigate the optimal conditions and contexts for applying System 1 and System 2 at the token-level decoding stage.

tions in a slower, more deliberate manner, tackling complex issues that require long-term planning. The integration of System 1 and System 2 in daily human activities achieves maximum efficiency and demonstrates robust intelligence. Notably, System 1 handles approximately 95% of everyday tasks, while System 2 is responsible for the remaining 5% (Daniel, 2017). The fast and slow thinking paradigm has been applied in various LLM applications (Hagendorff et al., 2023; Lin et al., 2024; Qi et al., 2024), which focus on improving the abilities of System 2 and the inference efficiency of LLMs. This paradigm provides a human-like mechanism for model optimization toward advanced artificial intelligence. Similarly, large and small models exhibit the natural characteristics of slow and fast processing. Previous works (Kim et al., 2024) on speculative decoding indicate that only a small percentage of generations require intervention by LLMs, while SLMs can independently complete most of the remainder.

However, many unknowns persist in these collaborations within this broader paradigm, including questions about when and where System 2 should be utilized. To the best of our knowledge, no previous work has attempted to unify the collaborative decoding methods of SLMs and LLMs from this perspectives. In this paper, we explore the fundamental questions of model collaboration under the fast and slow generation framework: 1) *What is the*

*essential difference between Systems 2 and 1 from the perspective of decoding? 2) Why does System 1 under-perform compared to System 2? 3) How can we improve efficiency of the large and small compound system?* In this paper, we focus exclusively on the high-efficiency features of System 1 and System 2 based on language models, deferring an exploration of their in-depth reasoning capabilities to future work.

To investigate these problems, we focus on analyzing the frequency and position characteristics of collaboration between large and small models, employing both qualitative and quantitative methods. Our research examines theoretical limits to collaboration frequency and employs scaling laws to predict how often different model combinations might collaborate based on the disparity in their parameters. Additionally, we explore the position tendencies of these collaborations, particularly in terms of where in the generation process they are most effective and how they correlate with the inherent uncertainty of small models. Our preliminary findings are summarized as follows:

(1) A comparison between the collaboration dynamics in large and small models and human dual-system processes reveals a shift from the expected 95:5 fast and slow ratio to an 80-20 distribution (§ 5.1.1). This finding also suggests a strong correlation between the frequency of interventions and the size differences in parameters between models

(§ 5.1.2), in accordance with scaling laws.

(2) Collaborative interactions among models are most critical at the beginning of the generation process, underscoring the adage that “Well begun is half done.” (§ 5.2.1) The uncertainty levels in next token prediction of smaller models play a pivotal role, serving as a reliable indicator of collaboration necessity (§ 5.2.2).

## 2 Related Work

**Speculative decoding** (Leviathan et al., 2023; Kim et al., 2024; Sun et al., 2024) speeds up LLM inference by coupling a cheap *drafter* (SLM) with a costly *verifier* (LLM). Current efforts focus on (i) more capable drafters to boost the fraction of tokens that pass verification (Xia et al., 2023; Liu et al., 2024b), and (ii) lighter (lossless or approximate) verification for both greedy and sampling regimes (Stern et al., 2018; Xia et al., 2023; Fu et al., 2023b). While most approaches fix the draft length, recent work adapts it dynamically (Mamou et al., 2024; Shen et al., 2024). **Contrastive decoding** (Li et al., 2022) blends logits from an *expert* and an *amateur* to reduce hallucinations and improve reasoning (O’Brien and Lewis, 2023; Senrich et al., 2023). Contrasting models with/without context further lowers factual errors (Jin et al., 2024). Research mainly refines fusion rules; how much collaboration is truly required remains open. **Emulator** (Mitchell et al., 2023) and **proxy tuning** (Liu et al., 2024a) add a third model: an SLM fine-tuned to imitate a large chat model via  $\Delta$  logits. Other work orchestrates multiple LLMs for domain generalisation (Shen et al., 2024) or cloud-device cooperation with privacy guarantees (Zhang et al., 2024). **Scaling laws** for parameters, data and rewards (Kaplan et al., 2020; Hoffmann et al., 2022; Gao et al., 2022) explain why certain abilities emerge only in large models (Wei et al., 2022). Recent analyses attribute the seeming discontinuity to discrete evaluation metrics; continuous token-level scores reveal smoother trends (Schaeffer et al., 2024; Hu et al., 2024). We revisit these size-related gaps through the lens of collaborative decoding and token uncertainty.

Despite rapid progress, a systematic empirical account of *why* and *when* collaboration helps is missing. Our study fills this gap, quantifying ability differences between LLMs and SLMs and translating the findings into higher decoding efficiency and quality.

## 3 Preliminary

### 3.1 Text Generation

Text generation, also referred to as decoding (Welleck et al., 2024), is the process whereby a language model  $\mathcal{M}$  selects the next token in a sequence based on the context. Mathematically, this involves estimating the conditional probability of a token given its preceding sequence. For a given source context with  $N$  tokens and a target response with  $T$  tokens, noted as  $X = (x_1, x_2, \dots, x_N)$  and  $Y = (y_1, y_2, \dots, y_T)$ , the probability of the response sequence is expressed as:

$$P(Y|X) = \prod_{i=1}^T p(y_i|X, y_1, y_2, \dots, y_{i-1})$$

During the decoding phase, at each step  $i$ , the model  $\mathcal{M}$  computes the probability distribution  $p(y_i|X, y_1, y_2, \dots, y_{i-1})$  over the entire vocabulary, based on the input sequence provided so far. The selection of the next token  $y_i$  is then based on this distribution, typically employing methods such as greedy search (Parsing, 2009), beam search (Vijayakumar et al., 2016; Freitag and Al-Onaizan, 2017), or various sampling strategies (Fan et al., 2018).

### 3.2 Definition of FS-GEN Methods

We generalize text decoding to *collaborative decoding* between a large model  $\mathcal{M}_l$  and a small model  $\mathcal{M}_s$ . Both receive the same prompt  $X_l = X_s$ . As shown in Fig. 1, they output logits  $p_l$  and  $p_s$ . We denote reference logits by  $p_f$  and categorize token sequences into three sets: fused (or “golden”) outputs  $\mathcal{O}_g$ , LLM outputs  $\mathcal{O}_l$ , and SLM outputs  $\mathcal{O}_s$ . Unless stated otherwise,  $p_f$  and  $\mathcal{O}_g$  arise from collaboration rather than ground-truth labels.

**Speculative decoding.** SLM proposals are verified by the LLM, giving  $p_f = p_l$  and  $\mathcal{O}_g = \mathcal{O}_l$ . This is the canonical, trust-based form of collaboration.

**Contrastive decoding** (O’Brien and Lewis, 2023). Discrepancies are corrected with  $p_f = p_l + \beta(p_l - p_s)$ ; the fused result  $\mathcal{O}_g$  typically surpasses both  $\mathcal{O}_l$  and  $\mathcal{O}_s$ .

**Proxy tuning.** A tuned SLM ( $p'_s$ ) imitates the LLM while saving compute. Let  $p_f = p_l + (p'_s - p_s)$ ; the fused output  $\mathcal{O}_g$  approaches LLM quality by adding the delta between chat-tuned and base SLM logits.

As shown in Table 1, in speculative decoding, SLMs serve as the base models for fast generation,

Table 1: Motivation, roles of SLMs and LLMs in different collaborative decoding methods.

Method	Speculative Decoding	Contrastive Decoding	Proxy Tuning
Motivation	Inference efficiency	Factuality and Reasoning	Training efficiency
Role of SLMs	Drafter (generates candidate tokens)	Amateur (generates suboptimal tokens)	Domain expert (provides delta logits)
Role of LLMs	Verifier of draft tokens	Expert generator	Base logits provider
System 1	SLMs	SLMs	SLMs
System 2	LLMs	LLMs + SLMs	LLMs + SLMs (FT/Base)
Collaborations	Generation-verification	Logits fusion	Logits fusion

while LLMs manage the more deliberate and slow aspects, providing interventions to the fast process. In contrast, there is difference in contrastive decoding and proxy tuning, with SLMs introducing interventions into the LLM generation process. In addition to above methods, further exploration of other FS-GEN methods is displayed in supplementary materials.

### 3.3 Definition of Collaboration Frequency

As shown in Figure 1 and as discussed previously, not all steps in the fast system’s generation require collaborations or interventions from the slow system. We denote the number of collaborations in FS-GEN methods as  $N_l$  and total length of generated content as  $N_{total}$ . We defined collaboration frequency as  $CoF = \frac{N_l}{N_{total}}$ , which is typically set to 1 in contrastive decoding and proxy tuning. We can establish a lower bound of collaboration frequency, denoted as  $CoF_{lower}$ , which has been preliminarily proven in prior research (Kim et al., 2024) and is further discussed in § 5.1.1. The term “lower bound” denotes the minimal collaboration rate required for the text produced by SLMs (System 1) to exactly match the output generated solely by SLMs and LLMs (System 2). We provide a running example for computing  $CoF_{lower}$  in supplementary materials for better understanding.

## 4 Empirical Study

### 4.1 Experiment Setup

**Models.** Our selection of language models primarily includes the Qwen (Bai et al., 2023) series and Pythia series (Biderman et al., 2023), and OpenELM series (Mehta et al., 2024) which feature the broadest range of model parameters. The Qwen series spans from 0.5B to 72B, while the Pythia series covers from 70M to 12B. For the FS-GEN methods, model combinations are strate-

gically grouped based on size to facilitate collaborative decoding, such as pairing 0.5B with 14B, and 7B with 72B. To validate and observe the original behavior of models, we utilize the base models in addition to implementing proxy tuning, which involves combining base and chat models. For the Pythia series, we fine-tuned the base models using the `deita-10k-v0` dataset (Liu et al., 2023).

**Datasets.** To explore the common features of collaborative decoding across various tasks, our experiments utilize datasets tailored to distinct domains: MMLU-STEM (Hendrycks et al., 2021) for general knowledge, GSM8k (Cobbe et al., 2021) for math, and MBPP (Austin et al., 2021) for coding. Analysis is primarily conducted on the test sets to mitigate the risk of training data leakage; however, we also perform statistical analyses on the training data for threshold setting.

Our experiments are conducted exclusively using greedy decoding, which effectively captures the fundamental uncertainty associated with tokens in both large and small models (Wang and Zhou, 2024). Drawing on previous studies involving speculative and contrastive decoding, greedy search could potentially be adapted to sampling decoding through temperature rescaling. (Leviathan et al., 2023; Chen et al., 2023, 2024). Additional implementation details are provided in supplementary materials.

### 4.2 Experiment Design

Our empirical study investigates two key aspects of collaboration in FS-GEN: frequency and position. Collaboration frequency evaluates the minimum necessary interactions between large and small language models, enabling predictions of interaction frequency across different model scales. Collaboration position identifies specific interaction points, predicting when collaboration is needed. For collaboration frequency, we first analyze lower bound

of frequency across various model scales and tasks to identify trends (§ 5.1.1). Next, we examine a scaling law in FS-GEN that links models scale ratios with optimal interaction frequency for prediction (§ 5.1.2). For collaboration position, we initially study mismatch points relative to generated content length (§ 5.2.1), followed by an analysis based on token uncertainty (§ 5.2.2).

## 5 Experimental Results

### 5.1 Exploring Collaboration Frequency

#### 5.1.1 What is the Lower Bound of the Number of Collaborations in FS-GEN Methods?

##### *Finding 1: 20% Collaborations (2:8 Law)*

The lower bound of collaboration frequency between models of different scales averages less than 20% on various tasks and methods, reflecting the 2:8 law (the Pareto Principle).

As illustrated in Figure 2 and 3, the collaboration frequency varies with the difference between the task and model combination. Notably, for the specific SLMs, the frequency of collaboration increases as the size of LLMs increases, while models of similar sizes exhibit lower frequencies. When the parameters of the models are more closely aligned, the  $CoF_{\text{lower}}$  is reduced, generally not exceeding 20% across various tasks. This observation suggests that the parameter gap between models does not conform to traditional scaling laws, where task performance is typically emergent (Wei et al., 2022). It also indicates that scaling laws may be influenced by the continuity of metrics (Schaeffer et al., 2024), which are affected by token uncertainty. The primary distinction between large and small models lies in the selection of essential tokens. Small models can be significantly enhanced by a few key tokens from larger models, particularly those that are difficult for them to select. This phenomenon is evident not only in oracle decoding, a form of speculative reasoning, but also in other collaborative decoding methods such as contrastive and proxy decoding.

#### 5.1.2 What is the Relationship Between Models' Ratios and Lower Bounds?

##### *Finding 2: Parameters Scale Ratio Law*

The frequency of collaboration between a large and a small model during token generation is inversely proportional to the ratio of their parameters raised to a specific power.

Inspired by previous works on the “Scaling Law” (Kaplan et al., 2020; Ruan et al., 2024; Besiroglu et al., 2024; Hu et al., 2024), we aim to develop a scaling law relating the models' parameter ratios to the lower bound of collaboration frequency ( $CoF_{\text{lower}}$ ). Let number of parameters of  $\mathcal{M}_l$  and  $\mathcal{M}_s$  are denoted as  $\mathcal{N}_l$  and  $\mathcal{N}_s$  respectively. Intuitively, the  $CoF_{\text{lower}}$  is inversely proportional to the scale ratio  $R = \frac{\mathcal{N}_l}{\mathcal{N}_s}$ . The relationship between the variables can be expressed by the equation:

$$CoF_{\text{lower}} = \gamma \cdot R^{-\alpha} + \beta \quad (1)$$

where  $\gamma$  is a negative constant that determines the sensitivity of the  $CoF_{\text{lower}}$  to the scale ratio,  $\alpha$  is the scaling exponent,  $\beta$  is a fitting parameter. As the scale ratio  $R$  increases, the collaboration frequency also increases. This is because the distance between the output distributions of the larger model and the smaller model increases (as “Scaling Law” works), necessitating a higher proportion of collaboration  $CoF_{\text{lower}}$  from the smaller model. Figure 4 shows the fitting curves for different  $R$  across various FS-GEN methods; the corresponding results for the different evaluation tasks are available in the appendix. The results indicate that the lower bound of collaboration frequency is related to the scale ratio between LLMs and SLMs, as well as the specific tasks and methods involved.

We find that the fitting line is significantly influenced by the models' performance, with better-fitting accuracy observed in models with higher performance. Additionally, we provide the fitting line for the Pythia series in the appendix, which is challenging to fit due to its weaker performance.

### 5.2 Exploring Collaboration Position

#### 5.2.1 Which Part of the Generation Most Requires Collaboration or Intervention?

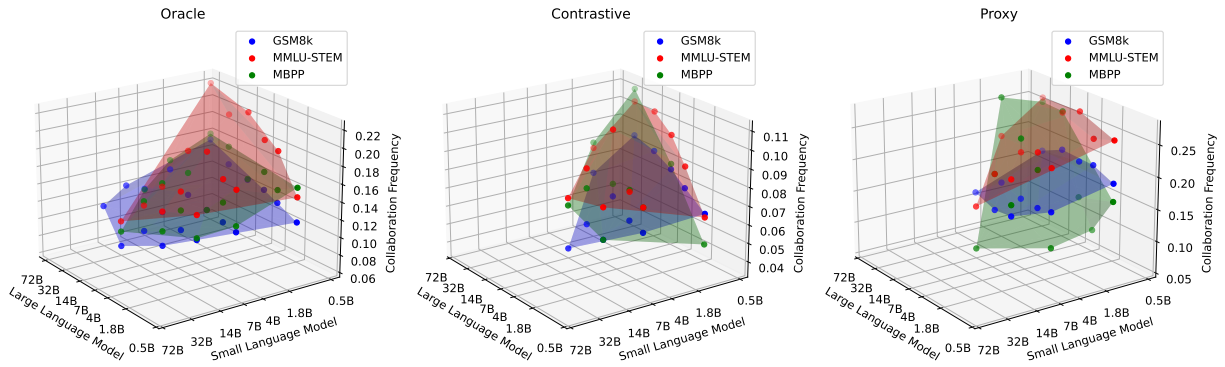


Figure 2: Collaboration frequency between models across various tasks (Qwen Series).

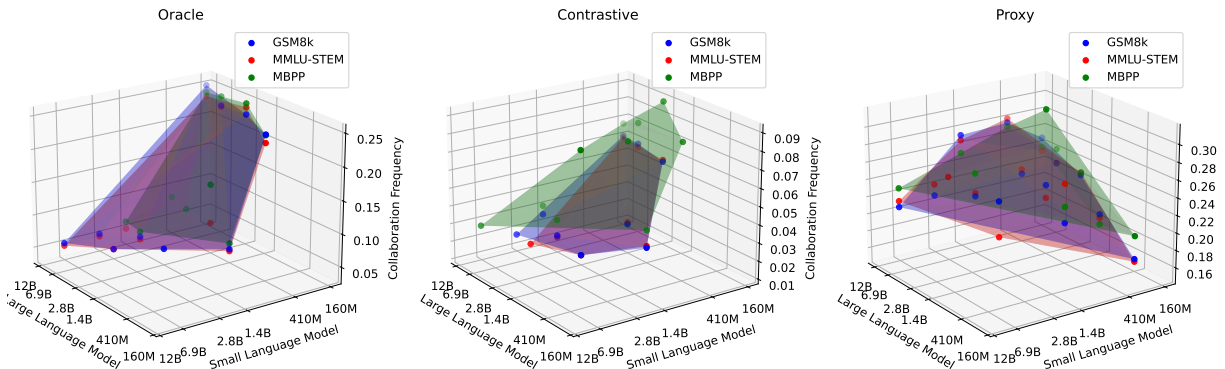


Figure 3: Collaboration frequency between models across various tasks (Pythia Series).

**Finding 3: “Well begun is half done”**

As the adage suggests, collaborations are most necessary at the beginning of the generation process, especially in reasoning tasks such as mathematics and coding.

As discussed, interventions from LLMs are necessary only at mismatch positions in generations between SLMs and LLMs. As depicted in Figure 5 and figures in appendix, we quantify these positions using a mismatch rate for all positions, analyzed with a 10% split where deeper colors indicate higher percentages and lighter colors represent lower percentages. Our initial analysis of mismatch positions across the entire generated content corroborates the importance of the generation’s start. Prior research on reasoning tasks demonstrates that the pass rate for smaller models can be significantly improved by substituting the initial steps with those generated by larger models (Jain et al., 2024; Wang and Zhou, 2024). This phenomenon is similarly observed during the collaborative decoding between large and small models. The findings reinforce the adage that a good start is crucial, largely due to the presence of more reliably conditioned tokens from

larger language models. This effect is particularly pronounced in GSM8k and MBPP tasks, illustrating that steps are more critical in reasoning tasks than in common knowledge tasks. Additionally, this finding could also support privacy protection in cloud-device collaborations, as highlighted in recent work (Zhang et al., 2024).

**5.2.2 Which Position in Fast Generation Most Requires Collaboration/Intervention?**

**Finding 4: Lost in Uncertain Tokens**

Tokens exhibiting high uncertainty within the fast system are likely to require collaboration or intervention from the slow system.

Beyond a macro analysis of whole-length position discrepancies, we also delve into the mismatch token logits and internal uncertainty within the vocabulary. Next token prediction (Radford et al., 2018, 2019; Brown et al., 2020; Achiam et al., 2023), a cornerstone of the “Scaling Law”, further highlights differences between large and small models. As depicted in Figure 6, this illustration shows that only a few tokens during the generation of a problem in the GSM8k task by small models

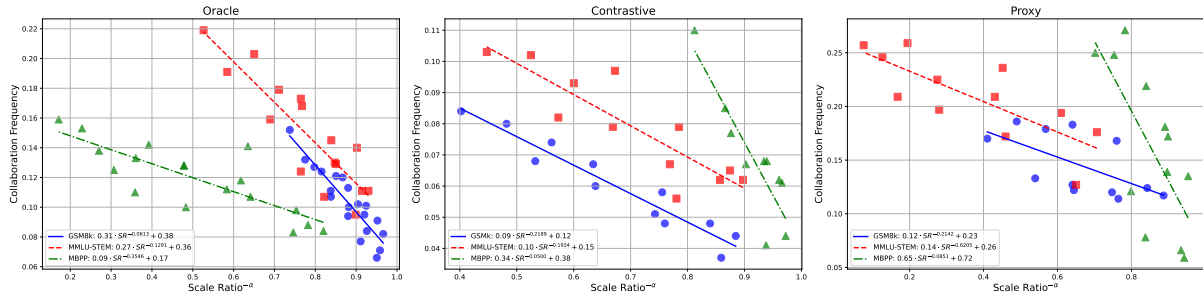


Figure 4: The Fitting Line of Parameters Scale Ratio Law for Qwen Series.

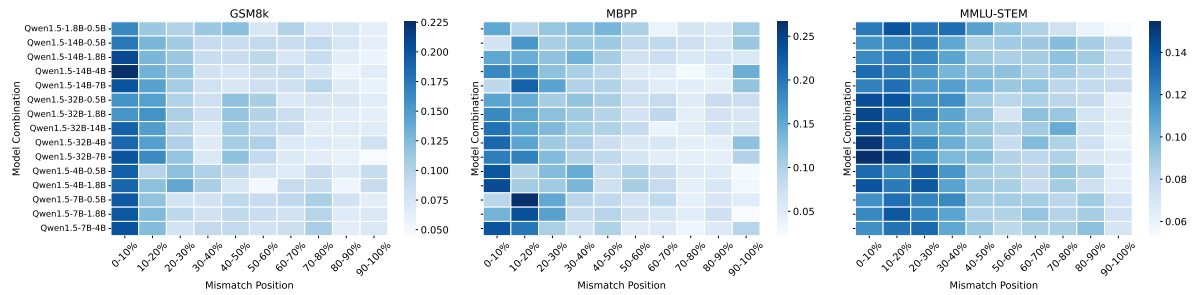


Figure 5: Percentage of mismatch positions relative to length for model combinations across various datasets (Qwen Series, Oracle).

require guidance from larger models, which are identified as red tokens.

Compared to large models, small models often struggle with uncertain tokens, which leads to incorrect responses. These tokens exhibit a similar distribution, characterized by high uncertainty at each step, with particularly low confidence in the top token, which helps identify crucial collaboration points. These findings have become fundamental principles for routing draft models in previous works (Kim et al., 2024; Du et al., 2024) on speculative decoding. We also observe their occurrence in various FS-GEN methods (i.e., contrastive decoding and proxy tuning), as detailed in supplementary materials. To further investigate the uncertainty of token logits, we conduct a visual analysis of the top-k token logits at each generative step, noting whether they match. The resulting correlation plots are displayed in supplementary materials. There is a high correlation between the mismatch positions and token uncertainty, particularly when more context tokens are considered. Additional cases and implementation details are provided in supplementary materials.

## 6 Discussion

**Difference in Various Tasks and Models.** While subtle differences in results may arise from vari-

ations in tasks and model capabilities, these differences do not impact our primary findings. The results are primarily influenced by two factors: data size and model performance: 1) Data Size. Due to computational constraints, we sampled approximately 500 data points for each task. This limited sampling may introduce fluctuations in the observed curves. 2) Model Performance. Parameter ratio scaling laws are strongly influenced by model performance. While Qwen series models exhibit consistent performance, Pythia models underperform due to insufficient pretraining, which affects the collaboration dynamics between large and small models.

**Cost-aware Collaboration Optimization.** In our experiments, we illustrate the lower bound of collaboration frequency; however, achieving this in practice is not fully feasible. We can optimize the collaboration frequency by examining token uncertainty, employing heuristic rules (Kim et al., 2024) that prove challenging to scale automatically across various scenarios. As shown in the right of Figure 7, we perform token-level routing based on threshold of the top-1 token logits from SLMs for proxy tuning, demonstrating promising efficiency-quality trade-offs for practical optimization. We use logits as features for routing, while other related metrics, such as entropy and perplexity-based

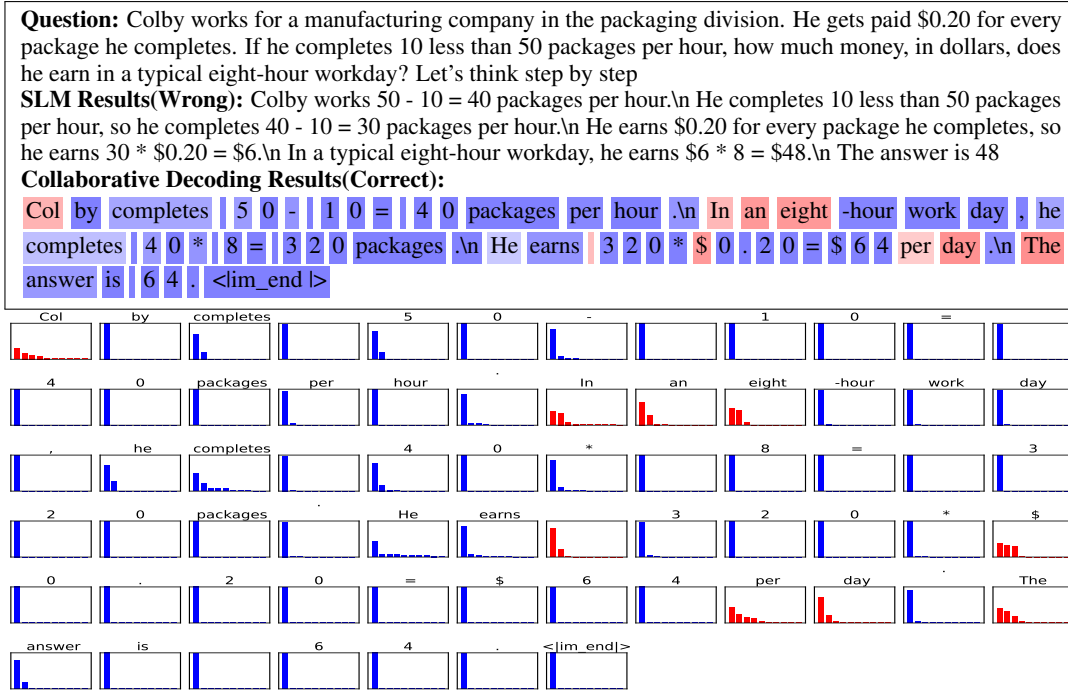


Figure 6: Visualization of token uncertainty in SLMs and LLMs. Red signifies mismatch tokens, while Blue indicates match tokens; darker shades represent higher confidence, where only with interventions from LLMs on red tokens can SLMs provide the correct answers.

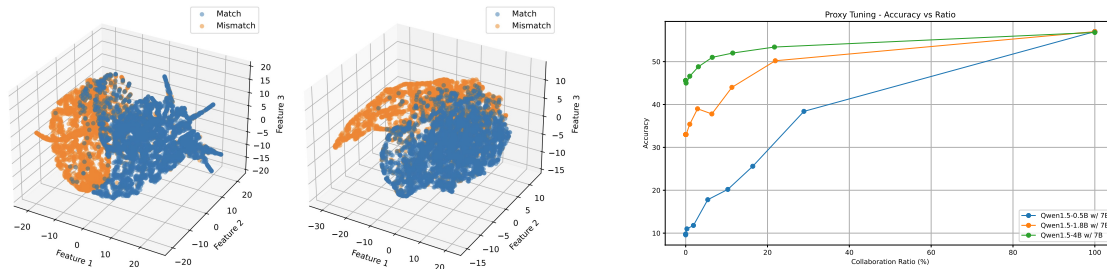


Figure 7: Top: 3D visualization of feature distribution for mismatch/uncertain-token logits (1 & 5 tokens). Bottom: Routing with Top-1 token logits of SLM for Proxy Tuning (Qwen Series).

logits, can be also explored. More analyses are provided in the Appendix. In conclusion, few studies consider these optimization of inference costs. Future research could explore strategies to balance inference costs and performance within a defined budget (Shen et al., 2024).

**More Model Series and Architecture.** In this paper, we exclusively conduct experiments on Qwen-series and Pythia-series models, which utilize transformer-based architecture. While further analysis could extend to different model series such as Llama and Yi (AI et al., 2024) models, the parameter scales of these models are not as extensive as those of the Qwen models. Additionally, recent state space models (SSM) (Gu et al., 2022) like Mamba (Gu and Dao, 2024; Dao and Gu, 2024) and hybrid architectures combining SSM and at-

tention mechanisms, such as Jamba (Lieber et al., 2024), are garnering researcher interest. Speculative decoding, when applied to multimodal models (Gagrani et al., 2024), allows for the expansion of FS-GEN methods across various multimodal applications (Zhu et al., 2023; Team, 2024) Our findings can be further verified on these models.

## 7 Conclusion

In conclusion, we analyze collaborative decoding through a fast-slow generation paradigm. Across speculative decoding, contrastive decoding, and emulator fine-tuning, we show that effective collaboration typically requires limited slow-model intervention, and is largely determined by the fast model's uncertainty in next-token prediction.

## 520 Limitations

521 Our experiments are limited to Transformer-series  
522 models, which use a transformer-based architecture.  
523 Extending the analysis to other model series like  
524 Llama, or recent state space models (SSM) such as  
525 Mamba and hybrid architectures like Jamba could  
526 provide further insights. However, they are still  
527 limited in terms of the wide range of parameters.  
528 Additionally, while we present the lower bound of  
529 collaboration frequency, achieving this in practice  
530 is challenging due to the difficulty in automatically  
531 scaling heuristic rules for different scenarios. Our  
532 tasks were focused on chat, math, coding, and gen-  
533 eral QA; more complex applications such as agent-  
534 based tasks and domains like finance, medical, and  
535 law should be explored. Currently, we propose  
536 a preliminary fitting law that still requires further  
537 empirical validation. We also hope this law can  
538 help us achieve a better balance between speed and  
539 performance in collaborative decoding. Finally, we  
540 did not fully address the balance between inference  
541 costs and performance, an area that requires further  
542 investigation to optimize within defined budgets.  
543 Future research should also include a broader range  
544 of tasks and more comprehensive data analysis to  
545 validate and extend our findings.

## 546 Ethics

547 This paper focuses on foundational findings in  
548 open-sourced models and publicly available bench-  
549 marks. Our empirical analysis enhances under-  
550 standing of black-box LLMs, which helps miti-  
551 gate potential risks associated with their deploy-  
552 ment. Improved transparency and predictability in  
553 LLM behavior contribute to safer and more reliable  
554 AI systems. Additionally, collaborative decoding  
555 methods, which aim to protect user data privacy,  
556 can benefit from our findings by further enhanc-  
557 ing and improving privacy measures. This ensures  
558 that user data remains secure during the decoding  
559 process, aligning with ethical standards for data  
560 protection and privacy.

## 561 References

562 Josh Achiam, Steven Adler, Sandhini Agarwal, Lama  
563 Ahmad, Ilge Akkaya, Florencia Leoni Aleman,  
564 Diogo Almeida, Janko Altenschmidt, Sam Altman,  
565 Shyamal Anadkat, and 1 others. 2023. Gpt-4 techni-  
566 cal report. *arXiv preprint arXiv:2303.08774*.

567 01. AI, :, Alex Young, Bei Chen, Chao Li, Chen-  
568 gen Huang, Ge Zhang, Guanwei Zhang, Heng Li,

Jiangcheng Zhu, Jianqun Chen, Jing Chang, Kaidong  
Yu, Peng Liu, Qiang Liu, Shawn Yue, Senbin  
Yang, Shiming Yang, Tao Yu, and 13 others. 2024.  
[Yi: Open foundation models by 01.ai](#). *Preprint*,  
arXiv:2403.04652. 569  
570  
571  
572  
573

Microsoft Research AI4Science and Microsoft Azure  
Quantum. 2023. [The impact of large language mod-  
els on scientific discovery: a preliminary study using  
gpt-4](#). *Preprint*, arXiv:2311.07361. 574  
575  
576  
577

AI@Meta. 2024. [Llama 3 model card](#). *GitHub*. 578

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten  
Bosma, Henryk Michalewski, David Dohan, Ellen  
Jiang, Carrie Cai, Michael Terry, Quoc Le, and 1  
others. 2021. Program synthesis with large language  
models. *arXiv preprint arXiv:2108.07732*. 579  
580  
581  
582  
583

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang,  
Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei  
Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin,  
Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu,  
Keming Lu, and 29 others. 2023. Qwen technical  
report. *arXiv preprint arXiv:2309.16609*. 584  
585  
586  
587  
588  
589

Tamay Besiroglu, Ege Erdil, Matthew Barnett, and Josh  
You. 2024. [Chinchilla scaling: A replication attempt](#).  
*Preprint*, arXiv:2404.10102. 590  
591  
592

Stella Biderman, Hailey Schoelkopf, Quentin Gregory  
Anthony, Herbie Bradley, Kyle O'Brien, Eric Hal-  
lahan, Mohammad Aflah Khan, Shivanshu Purohit,  
USVSN Sai Prashanth, Edward Raff, and 1 others.  
2023. Pythia: A suite for analyzing large language  
models across training and scaling. In *International  
Conference on Machine Learning*, pages 2397–2430.  
PMLR. 593  
594  
595  
596  
597  
598  
599  
600

Daniil A Boiko, Robert MacKnight, Ben Kline, and  
Gabe Gomes. 2023. Autonomous chemical research  
with large language models. *Nature*, 624(7992):570–  
578. 601  
602  
603  
604

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ  
Altman, Simran Arora, Sydney von Arx, Michael S  
Bernstein, Jeannette Bohg, Antoine Bosselut, Emma  
Brunskill, and 1 others. 2021. On the opportuni-  
ties and risks of foundation models. *arXiv preprint  
arXiv:2108.07258*. 605  
606  
607  
608  
609  
610

Tom Brown, Benjamin Mann, Nick Ryder, Melanie  
Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind  
Neelakantan, Pranav Shyam, Girish Sastry, Amanda  
Askell, and 1 others. 2020. Language models are  
few-shot learners. *Advances in neural information  
processing systems*, 33:1877–1901. 611  
612  
613  
614  
615  
616

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving,  
Jean-Baptiste Lespiau, Laurent Sifre, and John  
Jumper. 2023. [Accelerating large language model  
decoding with speculative sampling](#). *Preprint*,  
arXiv:2302.01318. 617  
618  
619  
620  
621

622	Ziyi Chen, Xiaocong Yang, Jiacheng Lin, Chenkai Sun,	Albert Gu, Karan Goel, and Christopher Re. 2022. <a href="#">Efficiently modeling long sequences with structured state spaces</a> . <i>Preprint</i> , arXiv:2111.00396.	674
623	Kevin Chen-Chuan Chang, and Jie Huang. 2024.		675
624	<a href="#">Cascade speculative drafting for even faster llm inference</a> . <i>Preprint</i> , arXiv:2312.11462.		676
625			
626	Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian,	Thilo Hagendorff, Sarah Fabi, and Michal Kosinski.	677
627	Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias	2023. <a href="#">Human-like intuitive behavior and reasoning</a>	678
628	Plappert, Jerry Tworek, Jacob Hilton, Reiichiro	<a href="#">biases emerged in large language models but disappeared in chatgpt</a> . <i>Nature Computational Science</i> ,	679
629	Nakano, Christopher Hesse, and John Schulman.	3(10).	680
630	2021. <a href="#">Training verifiers to solve math word problems</a> .		681
631	<i>Preprint</i> , arXiv:2110.14168.		
632	Kahneman Daniel. 2017. <i>Thinking, fast and slow</i> .	Dan Hendrycks, Collin Burns, Steven Basart, Andy	682
633		Zou, Mantas Mazeika, Dawn Song, and Jacob Stein-	683
634	Tri Dao and Albert Gu. 2024. <a href="#">Transformers are</a>	hardt. 2021. <a href="#">Measuring massive multitask language</a>	684
635	<a href="#">ssms: Generalized models and efficient algorithms</a>	<a href="#">understanding</a> . <i>Proceedings of the International Con-</i>	685
636	<a href="#">through structured state space duality</a> . <i>Preprint</i> ,	<a href="#">ference on Learning Representations (ICLR)</a> .	686
	arXiv:2405.21060.		
637	Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zong-	Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch,	687
638	han Yang, Yusheng Su, Shengding Hu, Yulin Chen,	Elena Buchatskaya, Trevor Cai, Eliza Rutherford,	688
639	Chi-Min Chan, Weize Chen, and 1 others. 2023.	Diego de Las Casas, Lisa Anne Hendricks, Johannes	689
640	<a href="#">Parameter-efficient fine-tuning of large-scale pre-</a>	Welbl, Aidan Clark, Tom Hennigan, Eric Noland,	690
641	<a href="#">trained language models</a> . <i>Nature Machine Intelli-</i>	Katie Millican, George van den Driessche, Bogdan	691
642	<a href="#">gence</a> , 5(3):220–235.	Damoc, Aurelia Guy, Simon Osindero, Karen Si-	692
643		mony, Erich Elsen, and 3 others. 2022. <a href="#">Training</a>	693
644	Cunxiao Du, Jing Jiang, Xu Yuanchen, Jiawei Wu,	<a href="#">compute-optimal large language models</a> . <i>Preprint</i> ,	694
645	Sicheng Yu, Yongqi Li, Shenggui Li, Kai Xu, Liqiang	arXiv:2203.15556.	695
646	Nie, Zhaopeng Tu, and Yang You. 2024. <a href="#">Glide with a</a>		
647	<a href="#">cape: A low-hassle method to accelerate speculative</a>	Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan	696
	<a href="#">decoding</a> . <i>Preprint</i> , arXiv:2402.02082.	Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and	697
648		Weizhu Chen. 2021. <a href="#">Lora: Low-rank adaptation of</a>	698
649	Angela Fan, Mike Lewis, and Yann Dauphin. 2018.	<a href="#">large language models</a> . <i>Preprint</i> , arXiv:2106.09685.	699
650	<a href="#">Hierarchical neural story generation</a> . <i>arXiv preprint</i>		
	<i>arXiv:1805.04833</i> .	Shengding Hu, Xin Liu, Xu Han, Xinrong Zhang, Chao-	700
651	Markus Freitag and Yaser Al-Onaizan. 2017. <a href="#">Beam</a>	qun He, Weilin Zhao, Yankai Lin, Ning Ding, Zebin	701
652	<a href="#">search strategies for neural machine translation</a> . In	Ou, Guoyang Zeng, Zhiyuan Liu, and Maosong Sun.	702
653	<i>Proceedings of the First Workshop on Neural Ma-</i>	2024. <a href="#">Predicting emergent abilities with infinite res-</a>	703
654	<i>chine Translation</i> . Association for Computational	<a href="#">olution evaluation</a> . <i>Preprint</i> , arXiv:2310.03262.	704
655	Linguistics.		
656	Yao Fu, Litu Ou, Mingyu Chen, Yuhao Wan, Hao	Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong,	705
657	Peng, and Tushar Khot. 2023a. <a href="#">Chain-of-thought</a>	Zhangyin Feng, Haotian Wang, Qianglong Chen,	706
658	<a href="#">hub: A continuous effort to measure large language</a>	Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting	707
659	<a href="#">models’ reasoning performance</a> . <i>arXiv preprint</i>	Liu. 2023. <a href="#">A survey on hallucination in large</a>	708
660	<i>arXiv:2305.17306</i> .	<a href="#">language models: Principles, taxonomy, challenges, and</a>	709
661	Yichao Fu, Peter Bailis, Ion Stoica, and Hao Zhang.	<a href="#">open questions</a> . <i>Preprint</i> , arXiv:2311.05232.	710
662	2023b. <a href="#">Breaking the sequential dependency of llm</a>		
663	<a href="#">inference using lookahead decoding</a> .	Kushal Jain, Niket Tandon, and Kumar Shridhar. 2024.	711
664		<a href="#">Well begun is half done: Importance of start-</a>	712
665	Mukul Gagrani, Raghav Goel, Wonseok Jeon, Jun-	<a href="#">ing right in multi-step math reasoning</a> . <i>Preprint</i> ,	713
666	young Park, Mingyu Lee, and Christopher Lott. 2024.	arXiv:2311.07945.	714
667	<a href="#">On speculative decoding for multimodal large lan-</a>		
	<a href="#">guage models</a> . <i>Preprint</i> , arXiv:2404.08856.	Lifeng Jin, Baolin Peng, Linfeng Song, Haitao Mi,	715
668		Ye Tian, and Dong Yu. 2024. <a href="#">Collaborative decod-</a>	716
669	Leo Gao, John Schulman, and Jacob Hilton. 2022.	<a href="#">ing of critical tokens for boosting factuality of large</a>	717
670	<a href="#">Scaling laws for reward model overoptimization</a> .	<a href="#">language models</a> . <i>Preprint</i> , arXiv:2402.17982.	718
	<i>Preprint</i> , arXiv:2210.10760.		
671	Albert Gu and Tri Dao. 2024. <a href="#">Mamba: Linear-</a>	Daniel Kahneman. 2011. <i>Thinking, fast and slow</i> .	719
672	<a href="#">time sequence modeling with selective state spaces</a> .	macmillan.	720
673	<i>Preprint</i> , arXiv:2312.00752.		
		Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B	721
		Brown, Benjamin Chess, Rewon Child, Scott Gray,	722
		Alec Radford, Jeffrey Wu, and Dario Amodei. 2020.	723
		<a href="#">Scaling laws for neural language models</a> . <i>arXiv</i>	724
		<i>preprint arXiv:2001.08361</i> .	725
		Sehoon Kim, Kartikeya Mangalam, Suhong Moon, Ji-	726
		tendra Malik, Michael W Mahoney, Amir Gholami,	727
		and Kurt Keutzer. 2024. <a href="#">Speculative decoding with</a>	728



838	Mitchell Stern, Noam Shazeer, and Jakob Uszkoreit.	Heming Xia, Zhe Yang, Qingxiu Dong, Peiyi Wang,	894
839	2018. <a href="#">Blockwise parallel decoding for deep autoregressive models</a> . In <i>Advances in Neural Information Processing Systems</i> , volume 31. Curran Associates, Inc.	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. <i>arXiv preprint arXiv:2401.07851</i> .	895
840			896
841			897
842			898
843	Ziteng Sun, Ananda Theertha Suresh, Jae Hun Ro, Ahmad Beirami, Himanshu Jain, and Felix Yu. 2024. Spectr: Fast speculative decoding via optimal transport. <i>Advances in Neural Information Processing Systems</i> , 36.	John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. 2024. <a href="#">Swe-agent: Agent-computer interfaces enable automated software engineering</a> . <i>Preprint</i> , arXiv:2405.15793.	899
844			900
845			901
846			902
847			903
848	Gemini Team. 2024. <a href="#">Gemini 1.5: Unlocking multi-modal understanding across millions of tokens of context</a> . <i>Preprint</i> , arXiv:2403.05530.		904
849		Kaiyan Zhang, Jianyu Wang, Ermo Hua, Biqing Qi, Ning Ding, and Bowen Zhou. 2024. Cogenesis: A framework collaborating large and small language models for secure context-aware instruction following. <i>The 62nd Annual Meeting of the Association for Computational Linguistics</i> .	905
850			906
851	Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, and 1 others. 2023. Gemini: a family of highly capable multimodal models. <i>arXiv preprint arXiv:2312.11805</i> .		907
852			908
853			909
854			910
855		Zixuan Zhou, Xuefei Ning, Ke Hong, Tianyu Fu, Jiaming Xu, Shiyao Li, Yuming Lou, Luning Wang, Zhihang Yuan, Xiuhong Li, Shengen Yan, Guohao Dai, Xiao-Ping Zhang, Yuhan Dong, and Yu Wang. 2024. <a href="#">A survey on efficient inference for large language models</a> . <i>Preprint</i> , arXiv:2404.14294.	911
856			912
857	Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023a. Llama: Open and efficient foundation language models. <i>arXiv preprint arXiv:2302.13971</i> .		913
858			914
859			915
860			916
861		Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. <a href="#">Minigt-4: Enhancing vision-language understanding with advanced large language models</a> . <i>Preprint</i> , arXiv:2304.10592.	917
862			918
863	Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajwal Bhargava, Shruti Bhosale, and 1 others. 2023b. Llama 2: Open foundation and fine-tuned chat models. <i>arXiv preprint arXiv:2307.09288</i> .		919
864			920
865			
866			
867			
868			
869	Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models. <i>arXiv preprint arXiv:1610.02424</i> .		
870			
871			
872			
873			
874	Xuezhi Wang and Denny Zhou. 2024. <a href="#">Chain-of-thought reasoning without prompting</a> . <i>Preprint</i> , arXiv:2402.10200.		
875			
876			
877	Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. <a href="#">Emergent abilities of large language models</a> . <i>Preprint</i> , arXiv:2206.07682.		
878			
879			
880			
881			
882			
883			
884	Sean Welleck, Amanda Bertsch, Matthew Finlayson, Hailey Schoelkopf, Alex Xie, Graham Neubig, Ilya Kulikov, and Zaid Harchaoui. 2024. From decoding to meta-generation: Inference-time algorithms for large language models. <i>arXiv preprint arXiv:2406.16838</i> .		
885			
886			
887			
888			
889			
890	Heming Xia, Tao Ge, Peiyi Wang, Si-Qing Chen, Furu Wei, and Zhifang Sui. 2023. <a href="#">Speculative decoding: Exploiting speculative execution for accelerating seq2seq generation</a> . <i>Preprint</i> , arXiv:2203.16487.		
891			
892			
893			

## A Related Work

**Speculative decoding** (Leviathan et al., 2023; Kim et al., 2024; Sun et al., 2024) speeds up LLM inference by coupling a cheap *drafter* (SLM) with a costly *verifier* (LLM). Current efforts focus on (i) more capable drafters to boost the fraction of tokens that pass verification (Xia et al., 2023; Liu et al., 2024b), and (ii) lighter (lossless or approximate) verification for both greedy and sampling regimes (Stern et al., 2018; Xia et al., 2023; Fu et al., 2023b). While most approaches fix the draft length, recent work adapts it dynamically (Mamou et al., 2024; Shen et al., 2024). **Contrastive decoding** (Li et al., 2022) blends logits from an *expert* and an *amateur* to reduce hallucinations and improve reasoning (O’Brien and Lewis, 2023; Senrich et al., 2023). Contrasting models with/without context further lowers factual errors (Jin et al., 2024). Research mainly refines fusion rules; how much collaboration is truly required remains open. **Emulator** (Mitchell et al., 2023) and **proxy tuning** (Liu et al., 2024a) add a third model: an SLM fine-tuned to imitate a large chat model via  $\Delta$  logits. Other work orchestrates multiple LLMs for domain generalisation (Shen et al., 2024) or cloud-device cooperation with privacy guarantees (Zhang et al., 2024). **Scaling laws** for parameters, data and rewards (Kaplan et al., 2020; Hoffmann et al., 2022; Gao et al., 2022) explain why certain abilities emerge only in large models (Wei et al., 2022). Recent analyses attribute the seeming discontinuity to discrete evaluation metrics; continuous token-level scores reveal smoother trends (Schaeffer et al., 2024; Hu et al., 2024). We revisit these size-related gaps through the lens of collaborative decoding and token uncertainty.

Despite rapid progress, a systematic empirical account of *why* and *when* collaboration helps is missing. Our study fills this gap, quantifying ability differences between LLMs and SLMs and translating the findings into higher decoding efficiency and quality.

## B Definition of More FS-GEN methods

In accordance with specific definitions for various FS-GEN methods in § 3.2, we introduce additional variants as follows:

Mitchell et al. (2023) propose emulator tuning, which, like proxy tuning, is motivated similarly but differs in implementation. In this configuration, the SLMs are also divided into a base version and

a chat version, with their logits represented as  $p_s$  for the base version and  $p'_s$  for the chat version. The fused logits are represented by  $p_f = p_l \cdot \left(\frac{p'_s}{p_s}\right)$ , indicating that  $O_g$  combines the insights of the incremental changes between the tuned and untuned states of the SLMs, approaching the performance of a fully tuned LLMs.

Zhang et al. (2024) propose CoGenesis for secure, context-aware instruction following. In the logits-level variant of CoGenesis, the inputs for LLMs and SLMs differ, denoted as  $X_l \neq X_s$ . This approach utilizes predictions from both SLMs and LLMs, where SLMs generate personalized content and LLMs produce outlines. In this configuration, the reference logits are calculated as  $p_f = f(p_l - p_s)$ , where  $f$  represents a trained model that combines features from both, and  $O_g$  outperforms both  $O_l$  and  $O_s$ . This indicates that  $O_g$  effectively leverages the strengths of both models while preserving the privacy of user data.

## C Running Example for Collaborative Decoding

To enhance understanding of our empirical study settings for collaborative decoding, we provide a running example with step-by-step verification. This example demonstrates the process of judging matches or mismatches in the generation of SLMs using generated context from mix-scaled models, as observed in the results of collaborative decoding.

Our primary objective is to analyze the frequency of collaboration in various decoding settings. In our research, we explore collaborative decoding (CoDec) at all steps ( $CoF = 1$ ), for the given example in Table 2. For a lower collaboration frequency ( $CoF_{lower}$ ), we input the outputs of CoDec into smaller models token by token to assess the consistency of top tokens. (CoDec represents speculative decoding, contrastive decoding or proxy tuning). Assuming there are three mismatched tokens (e.g., "former", "78"), the calculated  $CoF_{lower} = \frac{2}{18}$ . However, unnecessary collaborations may occur even when matches are identified, leading to an variable where  $CoF_{lower} \leq CoF \leq 1$ . This motivates our investigation into the lower bounds of collaboration frequency, aiming to achieve similar outputs as full collaborative decoding with minimal collaborative steps. Our findings demonstrate this is a universal phenomenon across different collaborative decoding methods.

Speculative decoding currently selects a fixed

number of tokens (K-tokens) for generation-verification, which does not effectively reach. In contrast, methods such as contrastive decoding and proxy tuning entail collaborations at each step ( $CoF = 1$ ), which may not always be necessary.

## D Implementation Details

In speculative decoding, the number of tokens generated by draft small models is set as a hyperparameter, denoted as  $K$ . In our empirical study, we define  $K = 1$  as representing oracle decoding rather than speculative decoding. Although the experimental results are equivalent, implementing and analyzing  $K = 1$  is simpler.

For each task across all methods, we adapt in-context few-shot to enhance generation performance under base models, using 5-shots for MMLU, 10-shots for MBPP, and 8-shots for GSM8k. The few-shot examples are unified into a chain-of-thought format (Fu et al., 2023a).

Note that we regard the outputs from collaborative decoding as references; therefore, we do not compute accuracy for each task. By default, results from collaborative decoding are generally better than those from either smaller or larger models alone. Our primary focus is on exploring the differences between larger and smaller models, rather than comparing them to the ground truth.

Additionally, our intent was to compare the top-1 token selected by the SLM and the collaborative models under greedy decoding. While the final token is obtained using  $\arg \max$  logits, this approach is effectively equivalent to using probabilities after applying softmax. Therefore, we use logits throughout the main content, although probabilities could also be used for better understanding. For contrastive decoding, we primarily follow the implementation in (O’Brien and Lewis, 2023) rather than previous works (Li et al., 2022). This approach directly utilizes unnormalized scores (logits) assigned by the amateur and expert models.

In current work, we use greedy decoding to compute the matching rate of tokens between small and large language models. This choice aligns with our initial motivation of achieving collaborative decoding with minimal intervention in small models, treating the collaborative decoding results as golden tokens. For scenarios where exact matching is less critical and the focus shifts to performance-speed optimization, other sampling techniques can be explored. These techniques might yield better

performance with reduced collaboration frequency, leading to more efficient collaborations. However, quantifying results becomes more challenging due to the increased uncertainty introduced by sampling. We believe this is an exciting direction for future research, as it opens up possibilities for balancing efficiency and performance through alternative decoding strategies.

## E More Empirical Results

### E.1 Results on Collaboration Frequency

To validate the robustness of our findings, we conducted additional experiments on GPQA, MedQA, and IFEval, which include biology, medical, and physics question-answering tasks, as well as instruction-following tasks in open-domain settings. As shown in Table 3, the results indicate that the collaboration frequency consistently remains below 20% across various methods, tasks, and model combinations. Furthermore, we observe a decreasing trend in collaboration frequency as the ratio of model parameters decreases.

We also found that the collaboration rate of general models on domain-specific tasks is slightly higher than on general tasks. When extending model collaborations from generalist to specialist tasks, we anticipate that the collaboration frequency will decrease due to the narrower distribution of domain-specific terminology. However, the limited availability of comprehensive specialized model series constrains further analysis at this stage, leaving this exploration for future work.

We present the fitting line for the parameter scale ratio law of the Pythia series in Figure 8. In comparison to the Qwen series and OpenELM series shown in Figures 4 and 9, the Pythia series exhibit weaker performance, making it more challenging to fit the lines accurately.

### E.2 Results on Collaboration Position

We provide heatmaps of mismatch positions relative to Pythia model combinations for contrastive decoding and emulator fine-tuning in Figures 13 and 14 respectively. These heatmaps reveal findings similar to those obtained from oracle decoding and support the adage that “well begun is half done”.

The difference between Figures 15 and 7 lies in the selection of tokens within the generated content. Specifically, we select only one token for the former and five continuous tokens for the latter. This

Table 2: Running example for the computation of lower collaboration frequency.

**Input:**

User: "Who is Donald Trump?"

Assistant: "Donald Trump is the former President of the United States, who is 78 years old now."

**Computation:**

- First token verification, match:
  - CoDec: [Assistant]: "Donald"
  - Small: [Assistant]: "Donald"
- Second token verification, match:
  - CoDec: [Assistant]: "Donald Trump"
  - Small: [Assistant]: "Donald Trump"
- ...
- 5th token verification, mismatch:
  - CoDec: [Assistant]: "Donald Trump is the former President"
  - Small: [Assistant]: "Donald Trump is President"
- ... (match)
- 14th token verification, mismatch:
  - CoDec: [Assistant]: "Donald Trump is the former President of the United States, who is 78"
  - Small: [Assistant]: "Donald Trump is the former President of the United States, who is 80"
- ... (match)

Table 3: Collaboration frequency for Qwen models across GPQA, IFEval, and MedQA datasets.

$CoF_{lower}$	GPQA			IFEval			MedQA		
	SD	CD	PT	SD	CD	PT	SD	CD	PT
Qwen1.5-0.5B w 7B	0.162	0.211	0.157	0.208	0.298	0.200	0.230	0.296	0.225
Qwen1.5-1.8B w 7B	0.130	0.198	0.133	0.174	0.238	0.164	0.194	0.314	0.190
Qwen1.5-4B w 7B	0.099	0.155	0.098	0.149	0.221	0.145	0.169	0.308	0.165

relationship between the top-k logits for each position and mismatches with large models is clearly evident in the visualization. The results further indicate that identifying mismatch positions benefits from additional context at each step.

We provide additional model combinations for visualization of token-level uncertainty, as shown in Figure 28, 29, 30, 31, 32, 33. This figure also demonstrates that the uncertainty in tokens of SLMs results in incorrect answers. Examples illustrating uncertainty in SLMs using contrastive decoding and proxy tuning are presented in Figures 34 and 35, respectively.

## F Practical Application

### F.1 Quantitive Results of Uncertainty Analysis

To strengthen the evidence supporting our uncertainty analysis, we provide the corresponding quantitative results from the clustering in Figures 15 and 16, generalizing across all model combinations and methods. We utilize the following metrics to evaluate the correlation between matched and mismatched token logits:

- Silhouette Coefficient (SC). This metric (range: -1 to 1) assesses clustering quality by comparing intra-cluster cohesion and inter-cluster separation. Values >0.5 indicate strong clustering performance. A high SC value derived from Pearson or Spearman correlation demonstrates that the metric aligns well with the data.
- Davies-Bouldin Index (DBI). The DBI (range:  $[0, \infty)$ ) measures clustering compactness and separation, where lower values (<1) suggest better clustering quality. A low DBI derived from correlation methods indicates effective uncertainty estimation.
- Mean Cluster Center Distance (MCCD). MCCD measures the separation between cluster centers, with larger values indicating better distinction. Correlation methods that amplify these distances demonstrate their alignment with the data.

As shown in Table 4, our results demonstrate the effectiveness of uncertainty estimation: SC values are consistently close to 0.5; DBI values are below

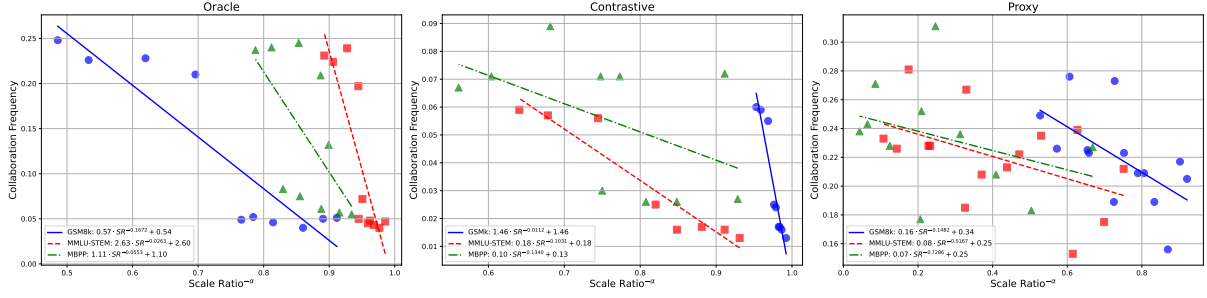


Figure 8: The relationship between models’ parameters ratios and the lower bounds of collaboration frequency (Pythia Series).

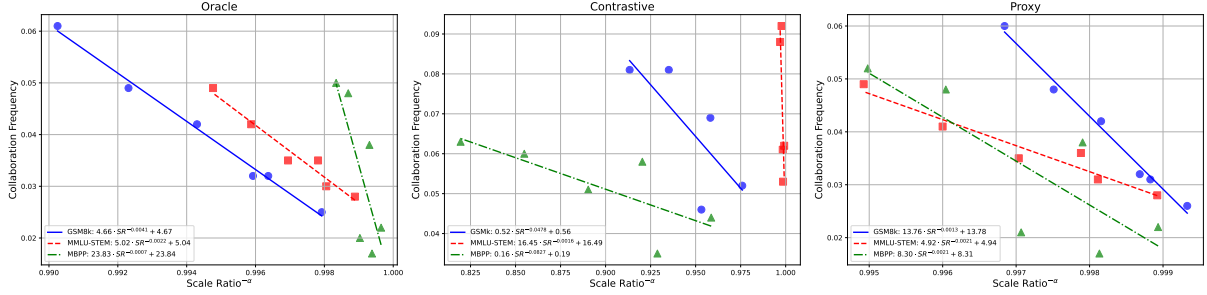


Figure 9: The Fitting Line of Parameters Scale Ratio Law for OpenELM Series.

1166 1, indicating compact and well-separated clusters;  
 1167 MCCD values range between 10 and 20, reflecting  
 1168 robust inter-cluster distinction. An exception is ob-  
 1169 served with Pythia series models, likely due to their  
 1170 insufficient pretraining. In conclusion, these results  
 1171 validate the effectiveness of Finding 4 discussed in  
 1172 Section 5.2.2.

## 1173 F.2 Token-level Routing for Collaborative 1174 Decoding

1175 Based on the quantitative results presented in Sec-  
 1176 tion F.1, we propose practical applications for  
 1177 proxy tuning, illustrated in Figure 7 and Figure 17.  
 1178 By utilizing only a few key positions for collabo-  
 1179 rative decoding, we can achieve higher accuracy  
 1180 outputs based on SLMs. These results highlight  
 1181 the effectiveness of token-level routing from SLMs  
 1182 to mix-scaled models using only the logits of the  
 1183 top-1 token from SLMs. Additionally, we provide  
 1184 a qualitative analysis of other related metrics, such  
 1185 as entropy and perplexity of the next token from  
 1186 SLMs, in Figure 27. These metrics exhibit simi-  
 1187 larly distinguishable distributions and offer prelim-  
 1188 inary potential for routing.

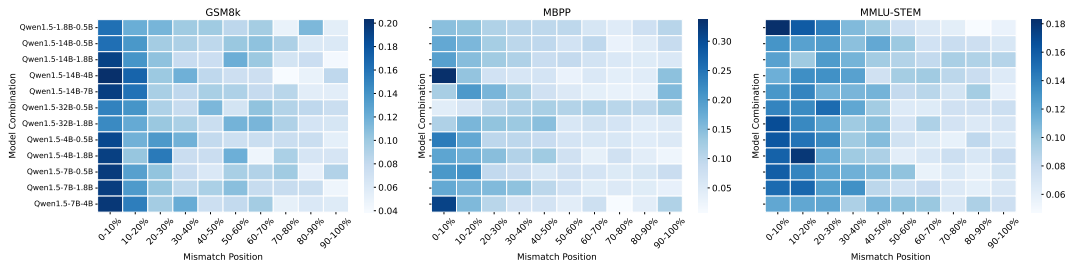


Figure 10: Percentage of mismatch positions relative to length for model combinations across various datasets (Qwen Series, Contrastive Decoding).

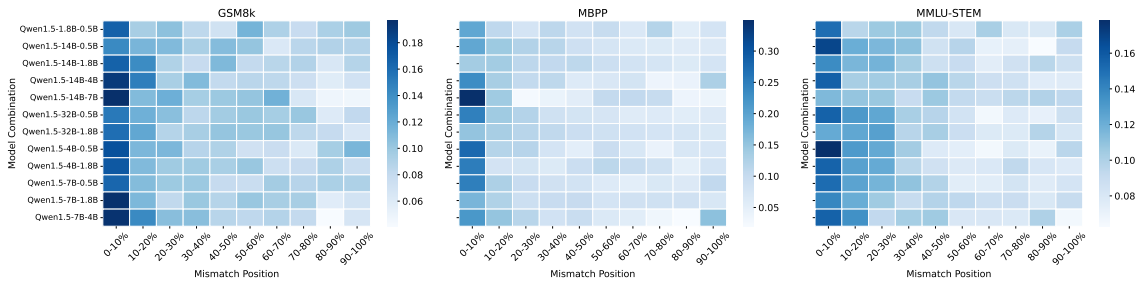


Figure 11: Percentage of mismatch positions relative to length for model combinations across various datasets (Qwen Series, Proxy Tuning).

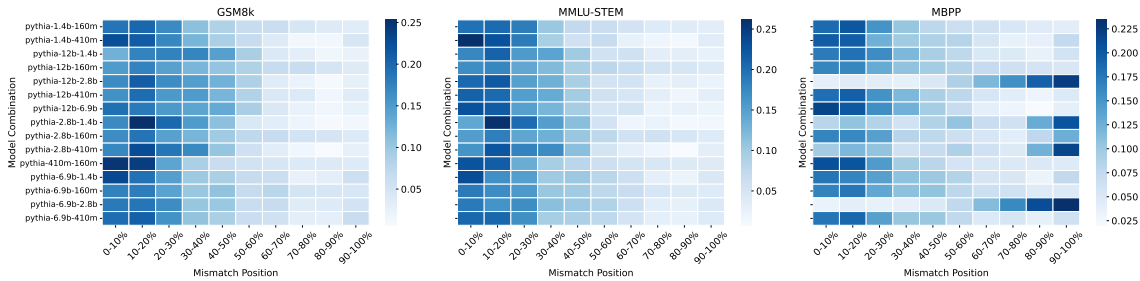


Figure 12: Percentage of mismatch positions relative to length for model combinations across various datasets (Pythia Series, Oracle Decoding).

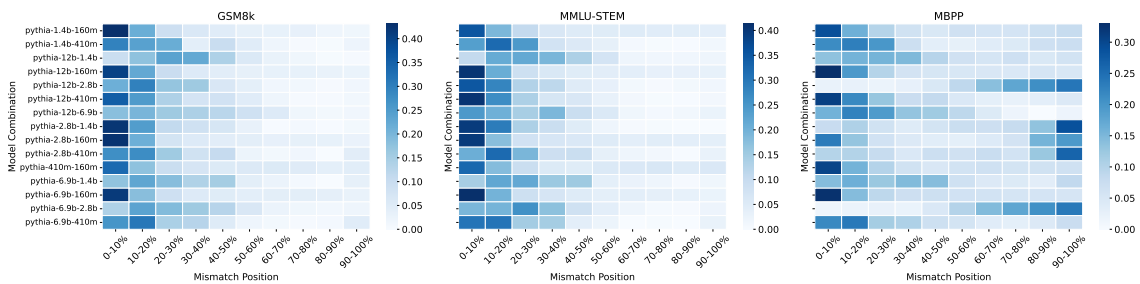


Figure 13: Percentage of mismatch positions relative to length for model combinations across various datasets (Pythia Series, Contrastive Decoding).

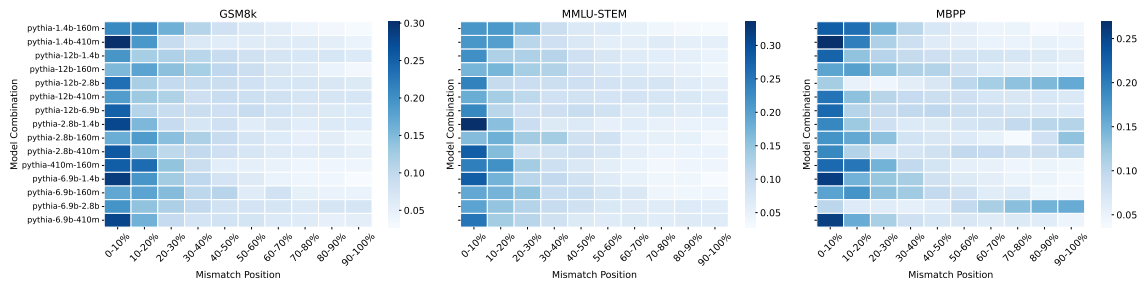


Figure 14: Percentage of mismatch positions relative to length for model combinations across various datasets (Pythia Series, Proxy Fine Tuning).

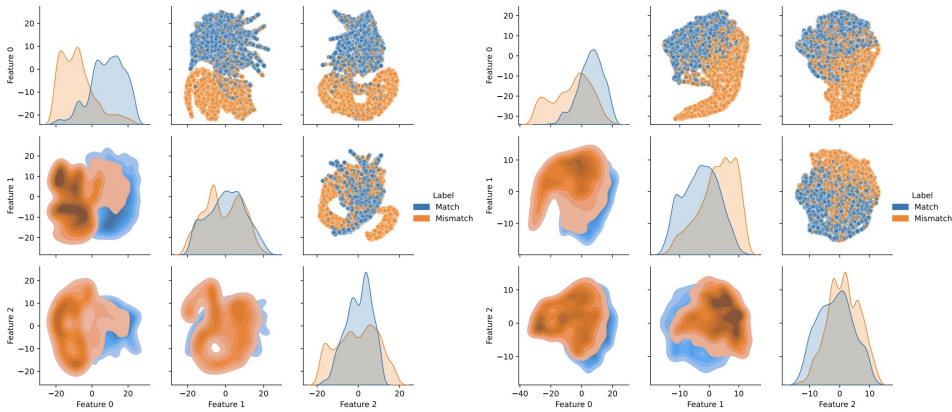


Figure 15: KDE and scatter visualization of logits distribution for mismatch/uncertain tokens from SLMs (Qwen Series). We provide the corresponding quantitative results in the Appendix F.1.

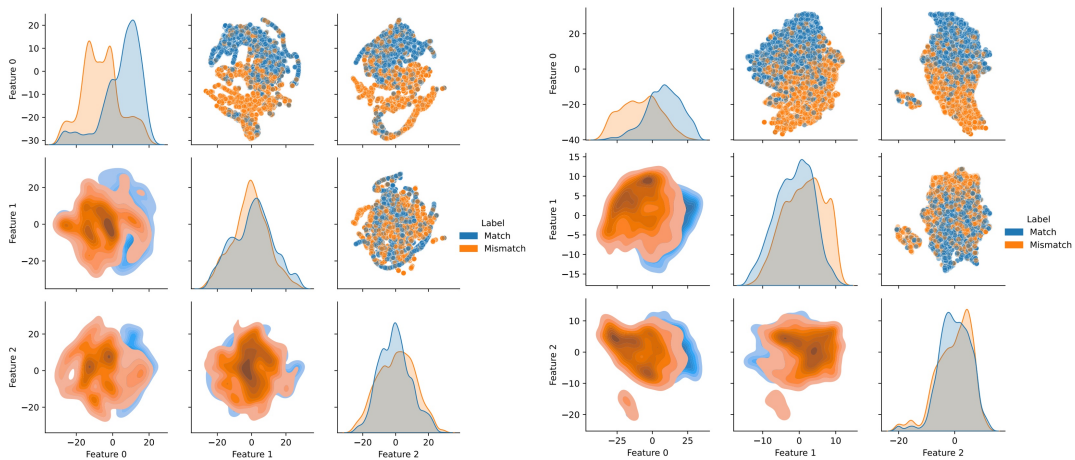


Figure 16: KDE and scatter visualization of feature distribution for mismatch/uncertain tokens (1 & 5 tokens) (Pythia Series).

Models	Metric	GSM8k		MMLU		MBPP	
		5 tokens	1 token	5 tokens	1 token	5 tokens	1 token
Qwen1.5	SC	0.465	0.503	0.445	0.457	0.470	0.469
	DBI	0.806	0.805	0.838	0.917	0.772	0.909
	MCCD	7.533	18.176	11.036	15.64	13.431	16.156
Pythia	SC	0.465	0.358	0.485	0.286	0.464	0.315
	DBI	0.790	1.180	0.755	1.416	0.779	1.300
	MCCD	21.584	14.125	22.584	16.289	21.325	16.843

Table 4: Correlation Between Match/Mismatch Tokens and Top-K Token Logits of SLMs.

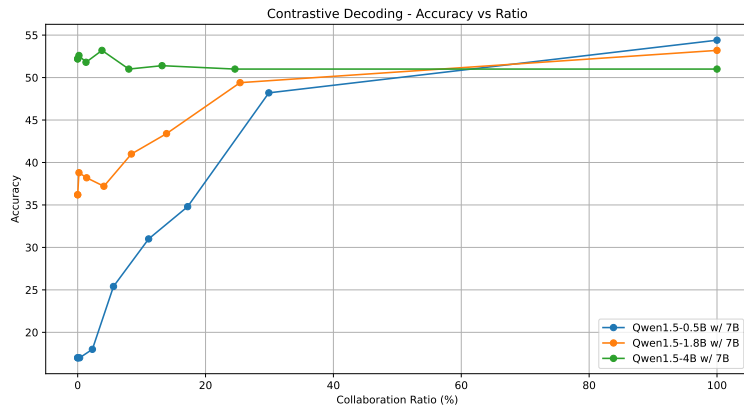


Figure 17: Routing with Top-1 Token Logits of SLM for Contrastive Decoding (Qwen Series).

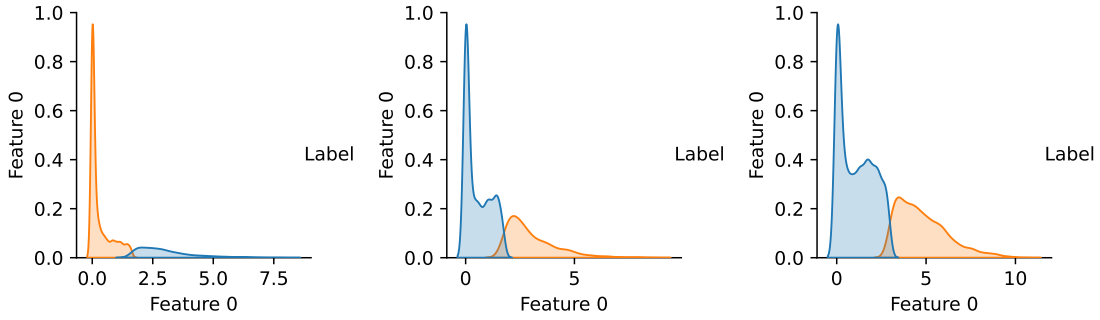


Figure 18: Entropy GPQA

Figure 19: Entropy MEDQA

Figure 20: Entropy IFEVAL

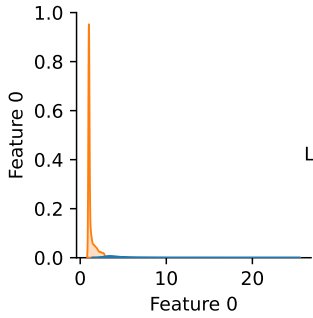


Figure 21: PPL GPQA

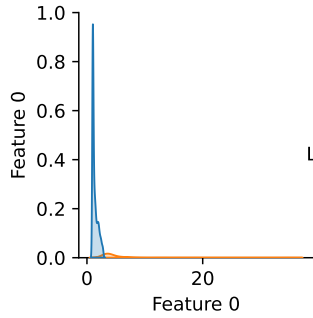


Figure 22: PPL MEDQA

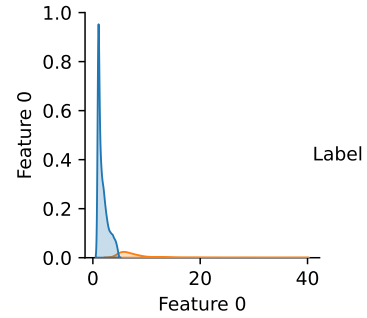


Figure 23: PPL IFEVAL

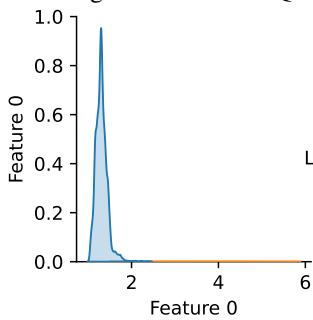


Figure 24: PPL\_SEQ GPQA

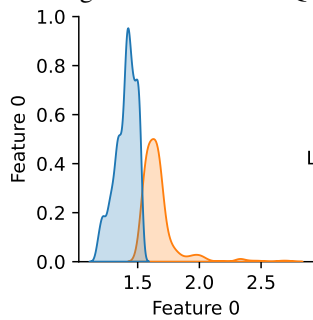


Figure 25: PPL\_SEQ MEDQA

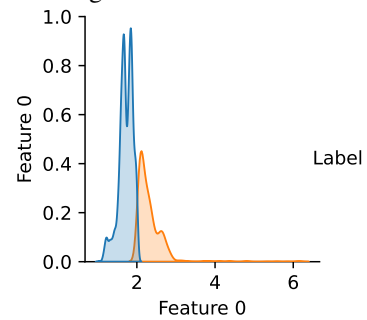


Figure 26: PPL\_SEQ IFEVAL

Figure 27: Comparison of Routing Metrics across Datasets

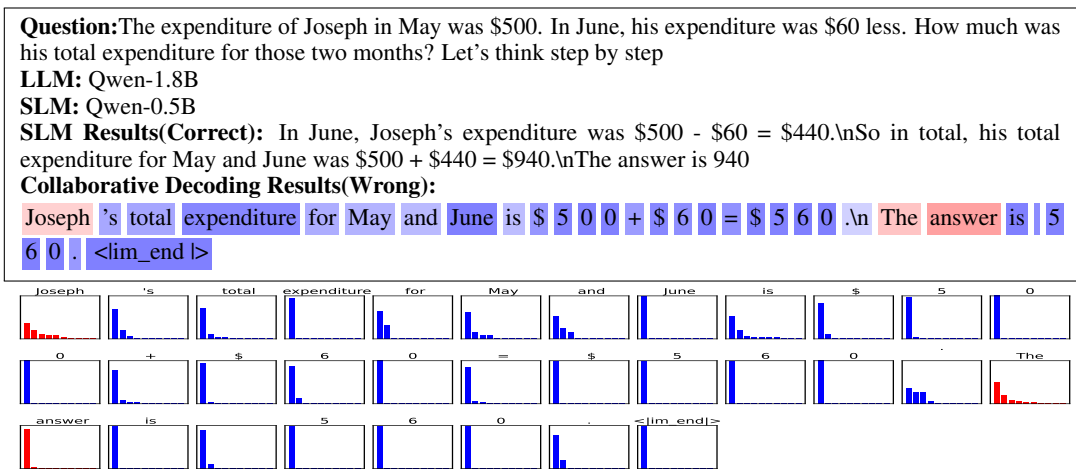


Figure 28: Visualization of Token Contributions. Red signifies mismatch tokens, Blue indicates match tokens, with darker shades representing higher weights.

**Question:** Grace weighs 125 pounds. Alex weighs 2 pounds less than 4 times what Grace weighs. What are their combined weights in pounds? Let's think step by step  
**LLM:** Qwen-1.8B  
**SLM:** Qwen-0.5B  
**SLM Results(Wrong):** Alex weighs  $4 * 125 - 2 = 502$  pounds. Their combined weight is  $125 + 502 = 627$  pounds. The answer is 627. <lim\_end>

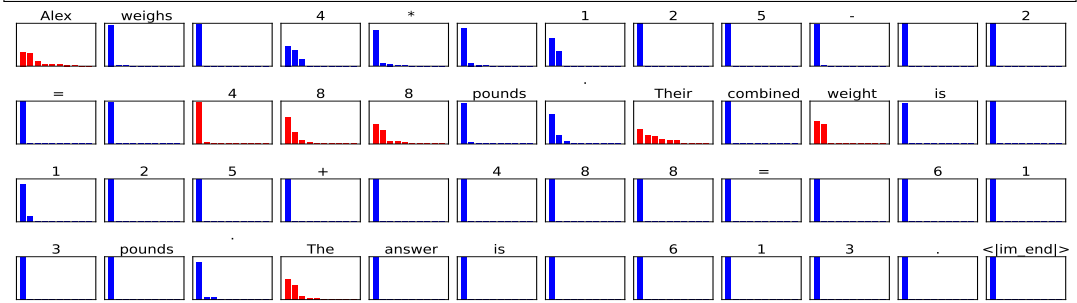


Figure 29: Visualization of Token Contributions. Red signifies mismatch tokens, Blue indicates match tokens, with darker shades representing higher weights.

**Question:** A bakery has 40 less than seven times as many loaves of bread as Sam had last Friday. If Sam had seventy loaves of bread last Friday, how many loaves of bread does the bakery have? Let's think step by step  
**LLM:** Qwen-1.8B  
**SLM:** Qwen-0.5B  
**SLM Results(Wrong):** The bakery has  $7 * 40 = 280$  loaves of bread. If Sam had 70 loaves of bread last Friday, then the bakery has  $280 - 70 = 210$  loaves of bread. The answer is 210  
**Collaborative Decoding Results(Wrong):** If Sam had 70 loaves of bread last Friday, then the bakery has  $70 - 40 = 30$  loaves of bread. ##### 30 The answer is : 30 <lim\_end>

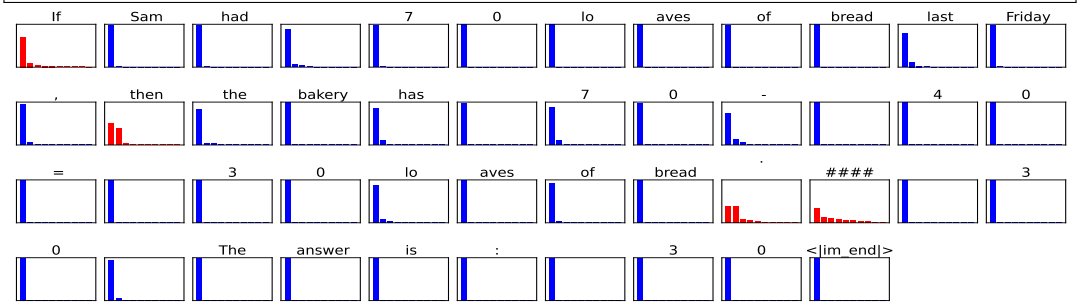


Figure 30: Visualization of Token Contributions. Red signifies mismatch tokens, Blue indicates match tokens, with darker shades representing higher weights.

**Question:** The expenditure of Joseph in May was \$500. In June, his expenditure was \$60 less. How much was his total expenditure for those two months? Let's think step by step

**SLM Results(Correct):** In June, Joseph's expenditure was  $\$500 - \$60 = \$440$ . So in total, his total expenditure for May and June was  $\$500 + \$440 = \$940$ . The answer is 940

**LLM:** Qwen-4B

**SLM:** Qwen-1.8B

**Collaborative Decoding Results(Correct):**

In May , Joseph spent \$ 5 0 0 .\n In June , his expenditure was \$ 6 0 less , so he spent \$ 5 0 0 - \$ 6 0 = \$ 4 4 0 .\n To find the total expenditure for those two months , we add the expenditures : \$ 5 0 0 + \$ 4 4 0 = \$ 9 4 0 .\n The answer is 9 4 0 . <lim\_end >

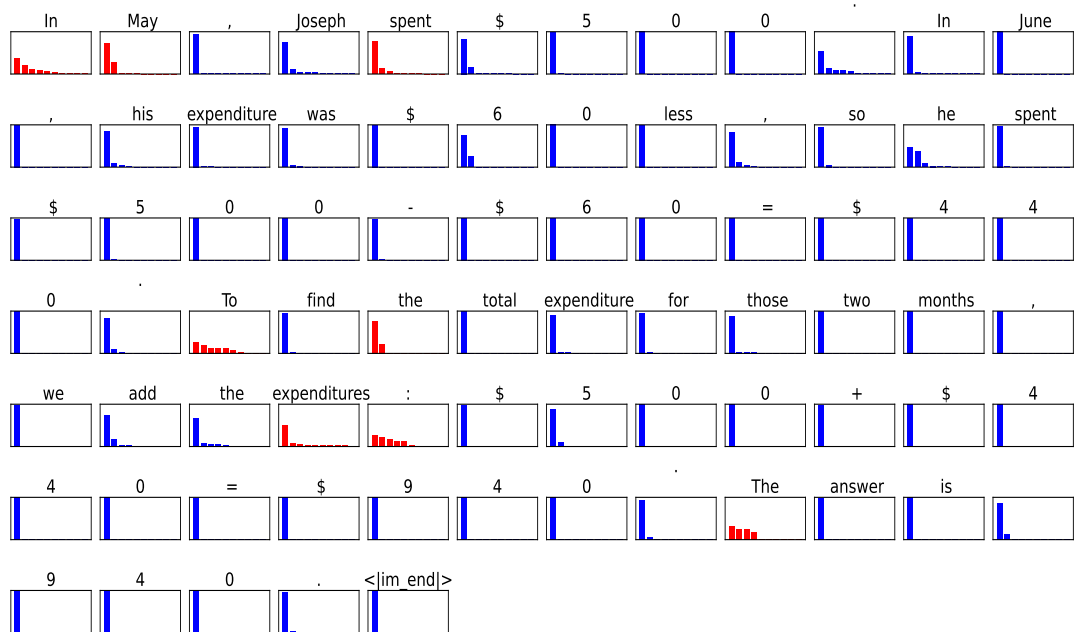


Figure 31: Visualization of Token Contributions. Red signifies mismatch tokens, Blue indicates match tokens, with darker shades representing higher weights.

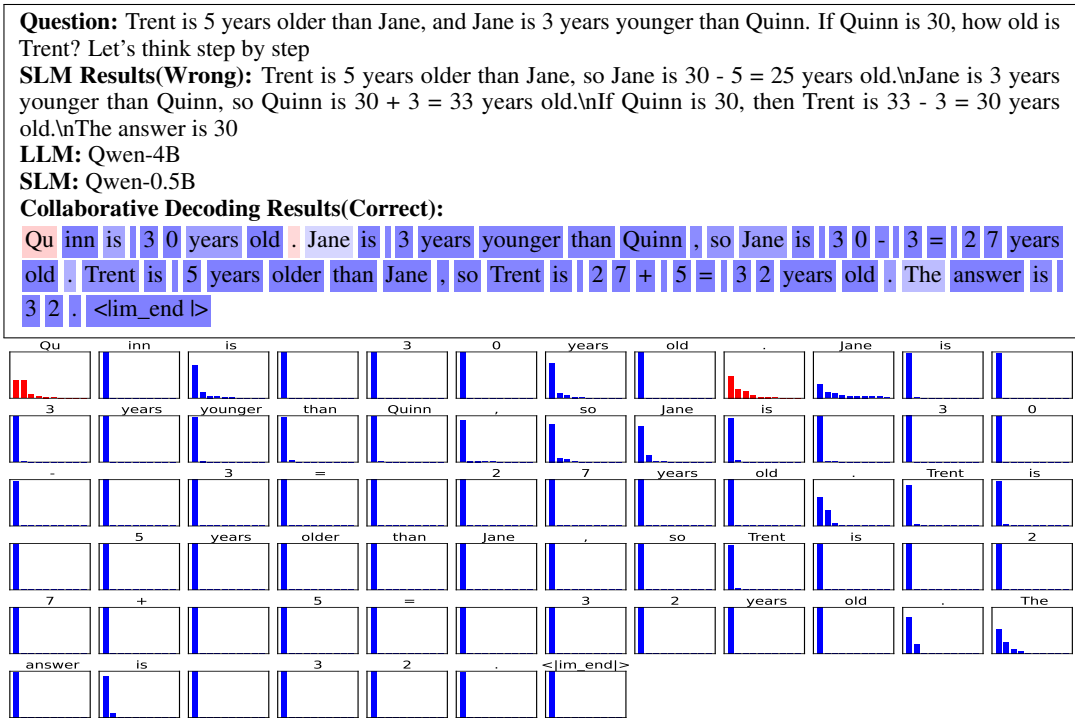


Figure 32: Visualization of Token Contributions. Red signifies mismatch tokens, Blue indicates match tokens, with darker shades representing higher weights.

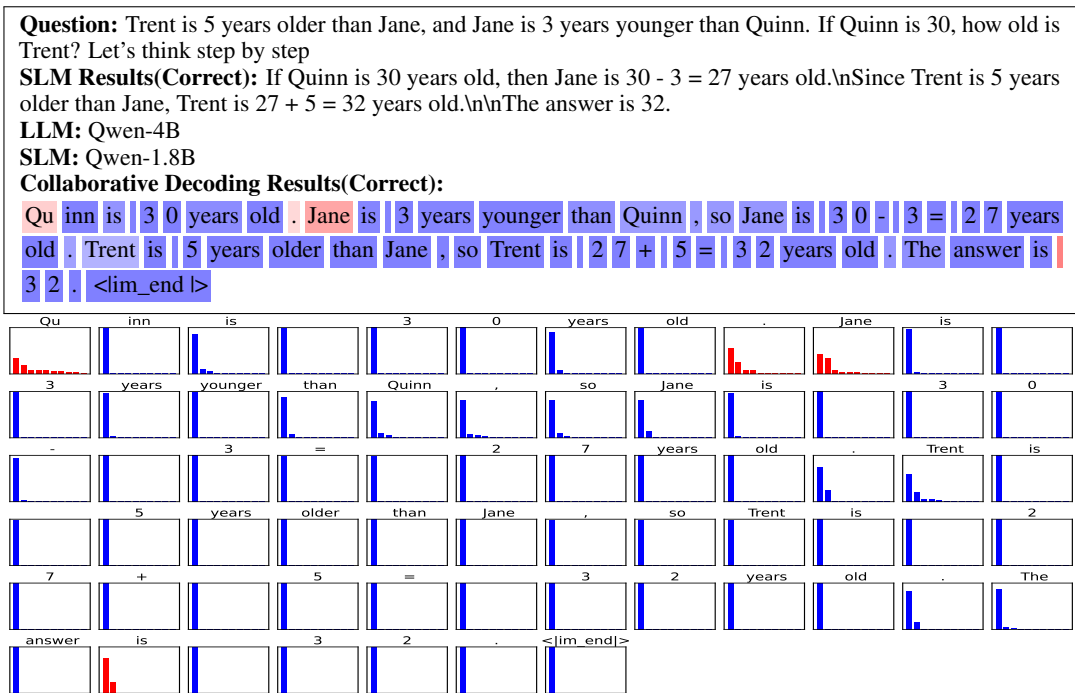


Figure 33: Visualization of Token Contributions. Red signifies mismatch tokens, Blue indicates match tokens, with darker shades representing higher weights.

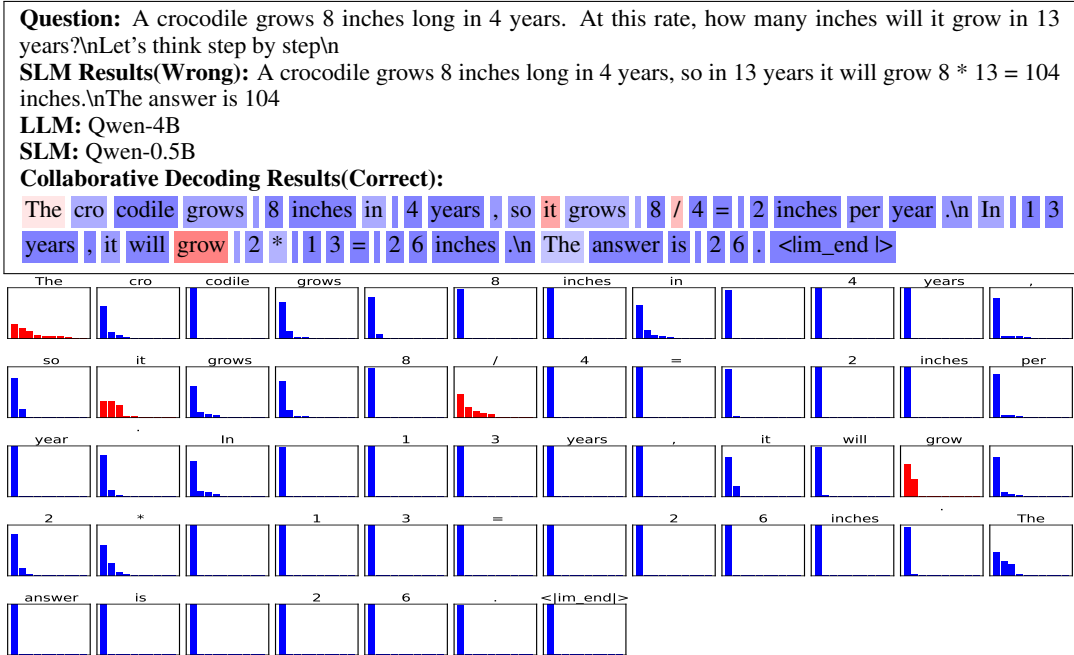


Figure 34: Visualization of Token Contributions in **Contrastive Decoding**. Red signifies mismatch tokens, Blue indicates match tokens, with darker shades representing higher weights.

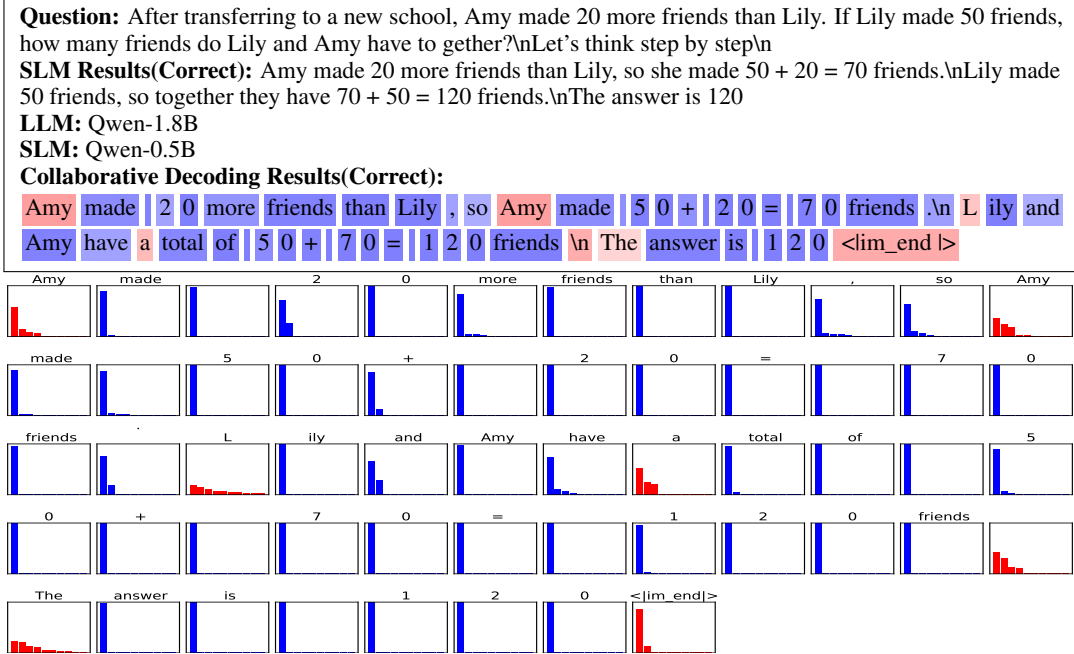


Figure 35: Visualization of Token Contributions in **Proxy Tuning**. Red signifies mismatch tokens, Blue indicates match tokens, with darker shades representing higher weights.