NLMR: NEURON-LEVEL LOCATE-THEN-MERGE FOR MULTILINGUAL REASONING IN ENGLISH-CENTRIC LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) exhibit strong reasoning capabilities in high-resource languages but struggle to generalize such capabilities to low-resource ones, making effective cross-lingual adaptation essential for multilingual reasoning. Common adaptation approaches, such as Supervised Fine-Tuning (SFT) on task data and Continual Pretraining (CPT) on language-specific text, face a fundamental challenge: SFT strengthens reasoning ability but provides limited target-language adaptation, whereas CPT enhances language-specific adaptation but can catastrophically forget previously acquired reasoning abilities. To address this challenge, we propose **N**euron-**L**evel Locate-then-Merge for **M**ultilingual **R**easoning **(NLMR)**, the first neuron-level locate-then-merge framework for multilingual reasoning that integrates the complementary strengths of task-specific SFT and language-specific CPT models. Specifically, we identify neurons critical to reasoning versus linguistic adaptation and selectively merge them, thus mitigating catastrophic forgetting while enhancing reasoning transfer across languages. To evaluate whether NLMR can jointly improve low-resource language reasoning and preserve language adaptation capability, we conduct extensive experiments across 10 languages, 3 reasoning benchmarks, and 2 backbone models, where NLMR consistently outperforms both single-trained models and strong parameter-level merging baselines. Furthermore, neuron-level analysis reveals distinct yet complementary subspaces for reasoning and language adaptation. By selectively preserving and integrating both subspaces, NLMR improves multilingual reasoning performance while also providing an interpretable view of how language and reasoning capabilities are transferred and combined.[1]

1 INTRODUCTION

Large Language Models (LLMs) have demonstrated remarkable abilities in reasoning, knowledge comprehension, and instruction following across a wide range of tasks in high-resource languages (HRLs) such as English (Brown et al., 2020; Kojima et al., 2022; Lou et al., 2024). However, due to their English-centric pretraining and limited exposure to low-resource languages (LRLs) (AI@Meta, 2024; Team, 2024), existing LLMs often lack sufficient adaptation to LRLs, resulting in degraded reasoning performance and limiting their ability to reliably transfer knowledge and solve complex tasks across diverse linguistic communities.

To improve low-resource language reasoning, existing adaptation approaches mainly rely on Supervised Fine-Tuning (SFT) and Continual Pretraining (CPT) (Tao et al., 2024). Specifically, SFT enhances task-specific reasoning through labeled demonstrations, but its transfer to LRLs is limited by the scarcity of high-quality reasoning supervision in these languages (Li et al., 2025). In contrast, CPT improves target-language competence by leveraging large-scale monolingual corpora (Nguyen et al., 2024), but may overwrite previously acquired reasoning abilities and cause catastrophic forgetting (Tao et al., 2024). Therefore, adapting English-centric LLMs to LRLs requires simultaneously preserving reasoning capability and enhancing language capability, which remains challenging for existing approaches.

[1] Code and data will be released upon publication.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To address this challenge, model merging has emerged as a promising alternative that integrates complementary capabilities from separately trained models without retraining from scratch (Tao et al., 2024). However, existing merging approaches mainly operate at the parameter level and uniformly combine updates, ignoring that different training objectives may modify different functional components of the model. Such coarse-grained merging makes it difficult to preserve both language adaptation and reasoning capability (Wortsman et al., 2022; Ilharco et al., 2022; Yadav et al., 2023; Yu et al., 2024; Deep et al., 2024). Motivated by the observation that reasoning-oriented SFT and language-oriented CPT induce distinct yet complementary neuron subspaces, we propose **N**euron-**L**evel **Locate-then-Merge** for **M**ultilingual **R**easoning **(NLMR)**, the first neuron-level locate-then-merge framework that identifies reasoning-related neurons from SFT and language-specific neurons from CPT, and selectively fuses them into a unified model. As illustrated in Figure 1, NLMR effectively resolves the conflict between language adaptation and reasoning preservation: while SFT, CPT, and parameter-level merging baselines fail to jointly maintain both capabilities, NLMR successfully integrates them. To validate whether NLMR can improve low-resource language reasoning while preserving language adaptation capability, we conduct extensive experiments on 10 languages and 3 reasoning benchmarks. NLMR consistently outperforms single-trained models and existing merging baselines, while neuron-level analysis further provides insights into how reasoning and language adaptation interact within English-centric LLMs.

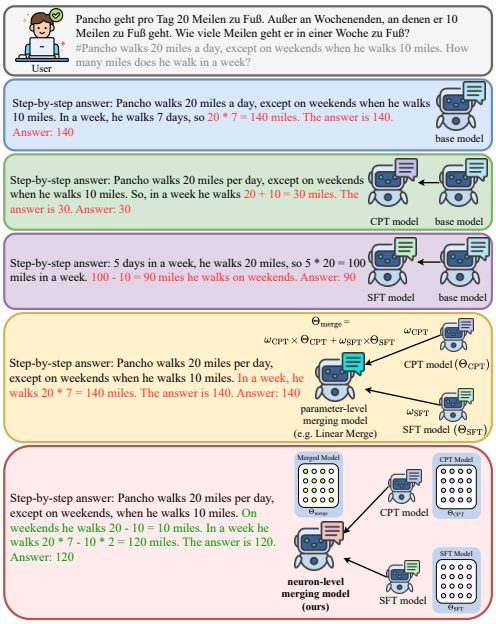


Figure 1: **An example of multilingual math reasoning in German.** The base model, CPT model, and parameter-level merging model fail to correctly answer the math problem due to either language barriers or catastrophic forgetting of reasoning logic. The SFT model violates the problem constraints. In contrast, our **NLMR** successfully preserves the reasoning capability from SFT and language proficiency from CPT, deriving the correct answer.

In summary, this work makes three key contributions: (1) To address the challenge of improving low-resource language reasoning while preserving language adaptation capability, we introduce **NLMR**, the first neuron-level locate-then-merge framework that integrates task-specific reasoning and language-specific adaptation to improve multilingual reasoning in English-centric LLMs. (2) To uncover the mechanism behind capability integration, we perform neuron-level analyses revealing that reasoning and language adaptation are associated with distinct yet complementary neuron subspaces across languages and model components. These findings provide insights into capability interaction and motivate more interpretable and fine-grained capability integration. (3) To validate whether NLMR can effectively balance reasoning capability and language capability, we conduct comprehensive multilingual experiments across 10 languages and 3 reasoning benchmarks, showing that NLMR consistently outperforms single-trained models and parameter-level merging baselines.

2 Related Work

2.1 Model Merging

Model merging has emerged as an efficient approach to combine the strengths of separately trained models into a single unified model without requiring access to their original training data or joint fine-tuning (Yang et al., 2024). Early merging strategies explored simple parameter averaging and its extensions, demonstrating that well-aligned models with shared initialization can be effectively combined to transfer capabilities across tasks or domains (Wortsman et al., 2022; Matena & Raffel, 2022). Building on this foundation, various merging methods have been proposed to improve

robustness and mitigate interference, including task arithmetic and linear interpolation techniques that represent individual task directions in parameter space (Ilharco et al., 2023), methods that explicitly resolve interference during merging (Yadav et al., 2023), and sparse mask–based merging approaches that scale multi-task model fusion (Davari & Belilovsky, 2024). Recent advances also explore magnitude-based sampling, parameter rescaling, and optimization of merging recipes to reduce destructive interactions among merged models (Deep et al., 2024; Yu et al., 2024; Akiba et al., 2025). Model merging has been shown to mitigate catastrophic forgetting by balancing learning from multiple models trained on different objectives or data distributions (Alexandrov et al., 2024; Feng et al., 2024). From this perspective, merging can be viewed as a post-hoc adaptation mechanism that preserves core capabilities while incorporating new skills, contrasting with standard sequential fine-tuning that often leads to performance degradation on previously learned tasks (Luo et al., 2025; Jiang et al., 2024).

While most work has focused on merging for multi-task performance and general transfer, only a few recent studies have investigated merging in multilingual or low-resource language contexts, indicating its potential for cross-lingual transfer (Tao et al., 2024; Bandarkar & Peng, 2025; Zhao et al., 2025; Bandarkar et al., 2025). Despite these promising directions, fine-grained merging that explicitly balances reasoning and language adaptation within English-centric LLMs remains underexplored, which NLMR addresses.

2.2 Neuron-Level Interpretability

Recent research has sought to understand how transformer-based LLMs encode knowledge and reasoning at the neuron level. Studies reveal that feed-forward layers function as structured key-value memories where individual neurons represent interpretable patterns or concepts in the vocabulary space (Geva et al., 2021; 2022). Further analyses identified knowledge neurons that store factual associations, enabling direct attribution and controllable editing of model behavior (Dai et al., 2022; Yu & Ananiadou, 2024). Building on this, multilingual studies have discovered language-specific neurons that encode linguistic competence across languages (Kojima et al., 2024; Chen et al., 2024; Tang et al., 2024; Zhang et al., 2024). Beyond interpretability, neuron-level fusion methods extend this perspective by identifying capability-critical neurons for selective parameter merging, effectively mitigating catastrophic forgetting and preserving specialized skills (Yu & Ananiadou, 2025; Fang et al., 2025). These findings collectively suggest that neurons form functional units that can be localized, analyzed, and merged, motivating our neuron-level merging framework for multilingual reasoning.

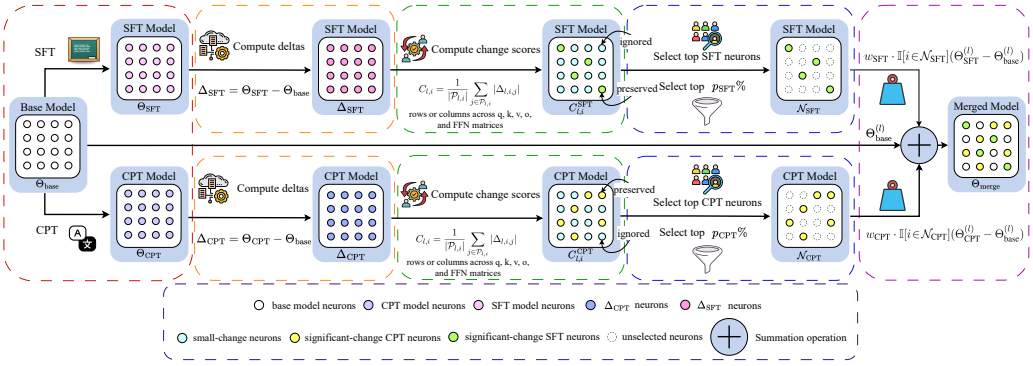


Figure 2: **Overview of the proposed NLMR framework.** Starting from a shared base model Θ_{base}, we independently obtain a continual-pretrained model (Θ_{CPT}) and a supervised fine-tuned model (Θ_{SFT}). Parameter deltas with respect to the base model are computed and aggregated into neuron-level change scores across attention and FFN modules. NLMR adopts a *double-locate* strategy to identify the top $p_{\text{CPT}}\%$ and $p_{\text{SFT}}\%$ neurons that are most affected by CPT and SFT, respectively. Selected neuron slices are then selectively merged into the base model with weighting coefficients, producing the final merged model Θ_{merge} without additional training.

3 METHODOLOGY

3.1 OVERVIEW

Motivated by the complementary capabilities learned through SFT and CPT, we consider integrating them through model merging. However, parameter-level merging treats all updates uniformly and may introduce interference between capability-specific changes. This motivates our neuron-level fusion strategy, which selectively identifies and integrates reasoning- and language-related updates. Specifically, we propose **N**euron-**L**evel Locate-then-**M**erge for **M**ultilingual **R**easoning (**NLMR**), which is training-free at the merge stage. NLMR integrates the complementary strengths of task-specific supervised fine-tuning (SFT) and language-specific continual pretraining (CPT) within English-centric LLMs. The goal of NLMR is to enhance reasoning in low-resource languages (LRLs) by merging two capability-specialized models—the task reasoning model and the language adaptation model—while avoiding catastrophic forgetting. Unlike prior multimodal merging frameworks (Yu & Ananiadou, 2025), NLMR adopts a *double-locate* strategy that independently identifies capability-critical neurons from both SFT and CPT models and selectively merges them. Figure 2 illustrates the pipeline: the base model Θ_{base} is first adapted through CPT and SFT to obtain Θ_{CPT} and Θ_{SFT}, which are then fused through NLMR to produce Θ_{merge}. The detailed procedure is formally described in Algorithm 1.

Algorithm 1 Neuron-Level Locate-then-Merge

Require: Base model Θ_{base}, CPT model Θ_{CPT}, SFT model Θ_{SFT}
1: Compute deltas: $\Delta_{\text{CPT}} = \Theta_{\text{CPT}} - \Theta_{\text{base}}$, $\Delta_{\text{SFT}} = \Theta_{\text{SFT}} - \Theta_{\text{base}}$
2: **for** each layer l and neuron i **do**
3: Compute change scores $C_{l,i}^{\text{CPT}}$ and $C_{l,i}^{\text{SFT}}$ using Eq. (4)
4: **end for**
5: Select top $p_{\text{CPT}}\%$ and $p_{\text{SFT}}\%$ neurons: $\mathcal{N}_{\text{CPT}}, \mathcal{N}_{\text{SFT}}$
6: **for** each selected neuron i **do**
7: Merge using Eq. (6) with weights $w_{\text{CPT}}, w_{\text{SFT}}$
8: **end for**
9: **return** merged model Θ_{merge}

3.2 PROBLEM FORMULATION

Let Θ_{base}, Θ_{CPT}, and Θ_{SFT} denote the parameter sets of the base, continual-pretrained, and fine-tuned models, respectively. All models share the same initialization and architecture. We define the *parameter deltas* as:

$$\Delta_{\text{CPT}} = \Theta_{\text{CPT}} - \Theta_{\text{base}}, \quad \Delta_{\text{SFT}} = \Theta_{\text{SFT}} - \Theta_{\text{base}}. \tag{1}$$

The objective is to produce a merged model

$$\Theta_{\text{merge}} = \Theta_{\text{base}} + F\big(\text{Sub}_{\text{CPT}}(\Delta_{\text{CPT}}), \text{Sub}_{\text{SFT}}(\Delta_{\text{SFT}})\big), \tag{2}$$

where $\text{Sub}(\cdot)$ denotes the function that locates capability-critical neurons, and $F(\cdot)$ performs selective parameter fusion. The merged model Θ_{merge} preserves reasoning and linguistic capabilities while requiring no further fine-tuning.

3.3 NLMR FRAMEWORK

NLMR builds upon the Locate-then-Merge paradigm (Yu & Ananiadou, 2025), where model fusion is achieved through identifying and combining a subset of critical parameter changes:

$$\Theta_{\text{merge}} = \Theta_{\text{base}} + w_{\text{CPT}} \cdot G_{\text{CPT}}(\text{Sub}_{\text{CPT}}(\Delta_{\text{CPT}})) + w_{\text{SFT}} \cdot G_{\text{SFT}}(\text{Sub}_{\text{SFT}}(\Delta_{\text{SFT}}))\,. \tag{3}$$

Here, w_{CPT} and w_{SFT} are weighting coefficients, and $G(\cdot)$ transforms the selected neurons for fusion. Unlike prior single-source merging, NLMR employs a dual-locate mechanism that captures the most influential neurons for language adaptation and reasoning separately.

Neuron-Level Change Computation Each neuron i in layer l is represented as a structured unit over its incoming and outgoing parameters. We compute its mean absolute change between two models as:

$$C_{l,i} = \frac{1}{|\mathcal{P}_{l,i}|} \sum_{j \in \mathcal{P}_{l,i}} |\Delta_{l,i,j}|\,, \tag{4}$$

where $\mathcal{P}_{l,i}$ denotes all parameters associated with neuron i (e.g., rows or columns across q, k, v, o, and FFN matrices). We obtain change scores $C^{\text{CPT}}_{l,i}$ and $C^{\text{SFT}}_{l,i}$ and select the top $p_{\text{CPT}}\%$ and $p_{\text{SFT}}\%$ neurons as:

$$\mathcal{N}_{\text{CPT}} = \text{TopK}(C^{\text{CPT}}), \quad \mathcal{N}_{\text{SFT}} = \text{TopK}(C^{\text{SFT}}). \tag{5}$$

Selective Neuron-Level Merging For each neuron subset, NLMR fuses corresponding slices of the CPT and SFT models while retaining the base parameters elsewhere:

$$\Theta^{(l)}_{\text{merge}} = \Theta^{(l)}_{\text{base}} + w_{\text{CPT}} \cdot \mathbb{I}[i \in \mathcal{N}_{\text{CPT}}](\Theta^{(l)}_{\text{CPT}} - \Theta^{(l)}_{\text{base}}) + w_{\text{SFT}} \cdot \mathbb{I}[i \in \mathcal{N}_{\text{SFT}}](\Theta^{(l)}_{\text{SFT}} - \Theta^{(l)}_{\text{base}}), \tag{6}$$

where $\mathbb{I}[\cdot]$ is an indicator function specifying neuron selection. This structured fusion ensures that language-sensitive neurons from CPT and reasoning-sensitive neurons from SFT are preserved.

Compared with existing merging methods such as TIES (Yadav et al., 2023), DELLA (Deep et al., 2024), or DARE (Yu et al., 2024), which locate parameters by magnitude globally, NLMR operates at the neuron level and distinguishes between distinct capability subspaces (language vs. reasoning). This fine-grained fusion preserves both functional domains, mitigates catastrophic forgetting, and remains efficient, requiring no additional training.

4 Neuron-Level Analysis

To understand the mechanism underlying the effectiveness of NLMR, we conduct a neuron-level analysis of how CPT and SFT reshape different parts of the model. We investigate whether the two training stages induce *distinct yet complementary neuron subspaces*, how these subspaces behave across languages, and how they are distributed across model components. All neuron-level analyses in this section are conducted using Gemma-2 2B (Team, 2024) as the base model, with CPT and SFT variants trained under the experimental settings described in Section 5.1, where the SFT model is fine-tuned on the GSM8K reasoning dataset (Cobbe et al., 2021).

Specifically, we first introduce an overlap-based analysis protocol for defining and comparing training-induced neuron subspaces. Based on this protocol, we structure our empirical analysis around the following research questions (**RQ**): (1) Are CPT- and SFT-induced neuron subspaces consistently complementary across languages? (2) How are CPT- and SFT-induced neuron subspaces distributed across different model modules? (3) Do the distributions and interactions of neuron importance scores provide additional evidence for the complementary relationship between CPT and SFT? We answer **RQ1–RQ3** in the following analysis.

4.1 Analysis Protocol: Defining Training-Induced Neuron Subspaces

Before addressing the research questions, we define the neuron subsets and overlap metric used throughout the analysis. Let $\mathcal{N}$ denote the set of all neurons in the model. For each training stage $T \in \{\text{CPT}, \text{SFT}\}$, as described in Section 3.3, we compute a neuron importance score $s_T(n)$ for every neuron $n \in \mathcal{N}$, based on the magnitude of parameter changes induced by training.

Sorting neurons by $s_T(n)$ in descending order, we define the top-$p\%$ neuron subset as:

$$\mathcal{N}^{(p)}_T = \{n \in \mathcal{N} : \text{rank}_{s_T}(n) \leq p|\mathcal{N}|/100\}.$$

To quantify the alignment between CPT-induced and SFT-induced neuron subspaces, we define the *neuron overlap ratio*:

$$\text{Overlap}(p) = \frac{\left|\mathcal{N}^{(p)}_{\text{CPT}} \cap \mathcal{N}^{(p)}_{\text{SFT}}\right|}{\left|\mathcal{N}^{(p)}_{\text{CPT}}\right|}.$$

This metric measures the fraction of language-sensitive neurons identified by CPT that are also selected as important during reasoning-oriented SFT. A low overlap at small p indicates that CPT and SFT emphasize largely different neurons at the top ranks, suggesting the existence of distinct training-induced subspaces.

4.2 RQ1: Cross-Lingual Consistency of Complementary Subspaces

We examine whether the relationship between CPT and SFT neuron subspaces is consistent across languages. We compute overlap curves for 10 typologically diverse languages, considering all neurons jointly (ALL). Figure 3 shows that across all languages, the overlap remains relatively low at small selection ratios (e.g., 1–10%), indicating that the most language-sensitive neurons induced by CPT are largely distinct from the most reasoning-sensitive neurons emphasized by SFT.

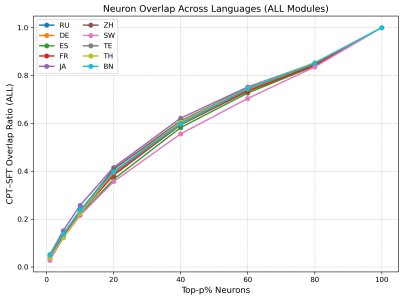


Figure 3: Cross-lingual overlap of CPT- and SFT-induced neuron subspaces. Overlap curves computed on GSM8K across ten languages.

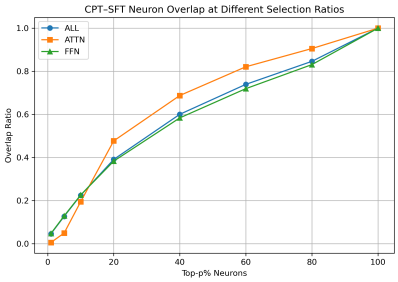


Figure 4: Module-wise CPT and SFT neuron overlap for **German** on MGSM across ATTN, FFN, and ALL.

As the selection ratio p increases, the overlap gradually rises, suggesting that CPT and SFT share a broader pool of moderately important neurons, despite maintaining separation among their most critical neurons. Importantly, this trend is consistent across languages, indicating that language and reasoning neuron subspaces exhibit stable complementary patterns independent of linguistic variation. These results provide affirmative evidence for **RQ1**, showing that the complementary structure between CPT- and SFT-induced subspaces generalizes robustly across languages.

4.3 RQ2: Module-Wise Distribution of Complementary Subspaces

We analyze how CPT- and SFT-induced neuron subspaces are distributed across different model modules. We decompose neurons into attention (ATTN) and feed-forward (FFN) components and analyze their overlap separately. Figure 4 presents module-wise overlap curves for German as a representative example; corresponding results for the other 9 languages are provided in Appendix A.3.

We observe a clear structural divergence between modules. FFN layers exhibit substantially higher overlap across small and medium selection ratios, suggesting that FFN neurons serve as a shared representational substrate supporting both multilingual language modeling and downstream reasoning. In contrast, attention layers show markedly lower overlap at small and medium p, indicating stronger specialization and task-specific reconfiguration. This module-wise distinction suggests that CPT and SFT reuse FFN neurons more heavily, while attention neurons are selectively reallocated to support different functional demands.

4.4 RQ3: Supporting Evidence from Neuron Importance Scores

In this subsection, we provide additional evidence for the complementary relationship between CPT- and SFT-induced neuron subspaces by analyzing the distribution and interaction of neuron importance scores. All results in this subsection are reported for **German (de)** as a representative language; corresponding results for the remaining 9 languages are provided in Appendix A.4.

Figure 5 compares importance score distributions under CPT and SFT across ALL, ATTN, and FFN neurons for German. CPT produces broader and higher-magnitude distributions, reflecting widespread structural adaptation during multilingual language learning. In contrast, SFT yields more concentrated and lower-magnitude scores, indicating selective emphasis on a smaller subset of neurons. Figure 6 further illustrates neuron-level interactions between CPT and SFT for German. While CPT and SFT scores show positive correlation—particularly in FFN layers—the scatter deviates substantially from the diagonal. Many neurons are highly important for only one training stage,

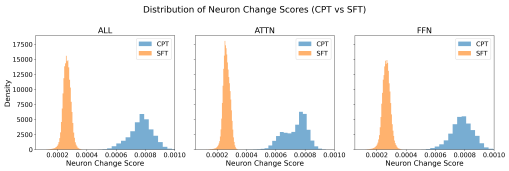


Figure 5: Distribution of neuron importance scores under CPT and SFT for **German**.

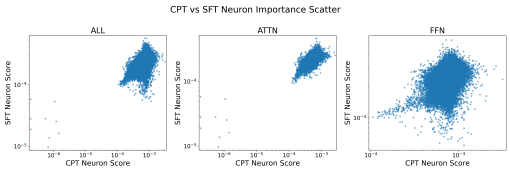


Figure 6: Neuron-level interaction between CPT- and SFT-induced subspaces for **German**.

reinforcing the conclusion that CPT and SFT induce distinct yet partially interacting subspaces rather than a single shared space.

Summary. In summary, our neuron-level analysis answers **RQ1–RQ3** by demonstrating that CPT and SFT induce distinct but complementary neuron subspaces. CPT constructs a broad, language-oriented representational foundation, while SFT selectively emphasizes a reasoning-oriented subset that partially overlaps with, yet remains functionally distinct from, the CPT-induced subspace. This complementary structure provides a mechanistic explanation for why selective neuron fusion can preserve multilingual knowledge while enhancing reasoning performance.

5 Experiments

5.1 Setup

Evaluation Languages and Datasets. Following Xue et al. (2021), we evaluate our method on 10 target languages: German (de), French (fr), Japanese (ja), Russian (ru), Chinese (zh), Bengali (bn), Spanish (es), Swahili (sw), Telugu (te), and Thai (th), covering diverse language families and writing systems. Following Tao et al. (2024), we adopt the MGSM dataset (Xue et al., 2021) for mathematical reasoning and evaluate on the full test split of 250 instances per language using the official multilingual test sets. Following Yu & Ananiadou (2025), we use ARC-Easy (Clark et al., 2018) for knowledge-based reasoning; since ARC-Easy is originally available only in English, we randomly sample 250 test instances and translate them into each target language using the `googletrans` library. Following Yu & Ananiadou (2025), we further evaluate commonsense reasoning on mCSQA (Sakai et al., 2024). For de, fr, ja, ru, and zh, we directly use the native test splits with 250 instances per language, while for bn, es, sw, te, and th, we randomly sample 250 examples from the English test set and translate them into the corresponding target languages using the same procedure as in ARC-Easy.

Evaluation Metrics. Following Yu & Ananiadou (2025), for all evaluated tasks, including mathematical, knowledge-based, and commonsense reasoning, we adopt answer accuracy as the evaluation metric. A prediction is considered correct if the generated answer matches the ground-truth label.

Baselines. We compare our approach against several representative model merging strategies implemented in MergeKit (Goddard et al., 2024), a framework widely adopted for combining task-specific or language-specific model parameters. These baselines serve as strong references for evaluating the effectiveness of different parameter merging mechanisms. Specifically, we include the following methods: Linear merging (Wortsman et al., 2022), Task Arithmetic (Ilharco et al., 2022), TIES-Merging (Yadav et al., 2023), DARE (Yu et al., 2024), and DELLA (Deep et al., 2024).

Base Model and Training Data. In our experiments, we use LLaMA-3 8B (AI@Meta, 2024) and Gemma-2 2B (Team, 2024) as the main backbone models. This choice allows us to focus on an *English-centric adaptation setting*, where task-specific reasoning supervision is provided in English and target-language capability is enhanced through language-specific CPT. For the CPT models, training data is drawn from the CulturaX (Nguyen et al., 2024) corpus across 10 target languages. For the SFT models, we train separate models for each task using their respective training sets: GSM8K (Cobbe et al., 2021), ARC-Easy, and mCSQA.

More experimental details, including training data, training configurations, prompt templates, and merging settings, are presented in Appendix B.

Table 1: Multilingual reasoning accuracy of **LLaMA-3 8B** and **Gemma-2 2B** across 10 languages. **Best** and second-best results within each dataset are highlighted. Task Ari: Task Arithmetic.

Dataset	Method	Language										Avg
		ru	de	es	fr	ja	zh	sw	te	th	bn	
LLaMA-3 8B												
MGSM	BASE	37.2	34.4	37.6	40.0	26.0	26.0	21.6	21.6	34.0	27.2	30.56
	CPT	23.6	25.2	25.6	26.4	18.0	24.0	21.2	13.2	28.4	7.2	21.28
	SFT	25.2	30.4	32.4	28.0	17.2	23.6	6.0	5.2	19.6	8.8	19.64
	Linear	36.0	35.6	40.4	41.6	32.8	33.6	28.0	24.8	36.8	28.8	33.84
	TIES	35.2	30.8	31.2	30.4	21.2	33.6	19.2	17.2	29.6	25.2	27.36
	DARE	26.0	32.0	28.8	27.6	19.2	25.2	16.4	18.8	27.6	20.0	24.16
	DELLA	37.2	39.6	38.4	39.6	30.4	**36.4**	31.2	**27.6**	**45.2**	28.8	35.44
	Task Ari	34.4	40.4	40.8	35.6	23.2	27.6	21.2	15.2	31.2	24.0	29.36
	NLMR	**42.0**	**43.2**	**49.6**	**48.4**	**33.2**	34.4	**31.6**	26.4	40.0	**34.0**	**38.28**
ARC-Easy	BASE	50.8	55.2	54.4	56.2	55.6	58.4	32.4	42.8	58.8	50.4	51.50
	CPT	68.0	65.2	76.8	68.7	65.2	54.0	51.2	42.4	69.2	27.2	58.79
	SFT	34.8	42.4	38.8	42.6	37.2	40.8	19.6	20.8	30.4	24.4	33.18
	Linear	61.2	65.2	66.0	80.3	60.4	78.4	38.8	49.6	54.8	49.6	60.43
	TIES	61.6	62.4	63.2	62.2	52.4	50.8	25.6	4.8	50.0	61.6	49.46
	DARE	61.6	72.4	72.8	69.5	60.8	60.0	33.6	54.0	56.0	53.6	59.43
	DELLA	**77.6**	76.4	**81.6**	77.5	**74.8**	80.0	41.6	16.0	**74.0**	62.8	66.23
	Task Ari	54.4	61.6	60.0	57.0	58.0	60.4	32.0	45.2	48.0	49.2	52.58
	NLMR	**77.6**	**77.6**	**81.6**	**81.9**	**74.8**	**82.4**	**58.4**	**57.6**	73.6	**67.6**	**73.31**
mCSQA	BASE	38.4	71.6	63.2	63.2	70.8	57.6	44.0	49.2	56.8	52.4	56.72
	CPT	39.6	64.8	63.2	64.0	73.2	28.4	48.0	50.8	60.4	32.8	52.52
	SFT	41.6	65.2	64.8	60.4	47.6	50.4	21.6	26.4	44.4	21.2	44.36
	Linear	49.6	73.6	55.2	51.6	74.0	60.8	58.0	46.8	51.2	41.6	56.24
	TIES	13.2	78.8	50.8	53.6	55.6	52.4	32.8	50.4	33.6	52.4	47.36
	DARE	22.4	84.0	68.8	68.8	67.2	59.2	35.6	**62.0**	51.2	60.0	57.92
	DELLA	48.8	**86.8**	**78.0**	**76.0**	74.0	**70.0**	34.4	58.0	**66.8**	**66.8**	66.04
	Task Ari	49.6	74.0	55.2	51.6	74.0	60.8	57.6	46.8	51.2	41.6	56.24
	NLMR	**50.0**	83.6	76.8	74.8	**78.4**	66.0	**60.0**	58.4	61.2	60.0	**66.92**
Gemma-2 2B												
MGSM	BASE	14.8	10.0	14.8	13.6	9.2	12.0	8.4	6.8	8.8	8.0	10.64
	CPT	8.4	9.2	7.6	9.2	8.0	7.2	7.2	6.8	4.4	3.2	7.12
	SFT	27.2	30.0	27.6	26.0	17.6	21.6	15.2	7.6	18.4	14.8	20.60
	Linear	28.0	29.6	29.2	25.6	22.4	26.8	21.2	20.0	24.4	23.6	25.08
	TIES	18.4	14.8	19.6	20.4	14.4	17.6	22.8	13.2	16.8	8.8	16.68
	DARE	22.0	23.6	20.0	20.0	16.4	14.0	21.6	12.0	18.4	7.2	17.52
	DELLA	24.4	22.8	28.8	20.8	17.6	21.2	18.4	18.0	14.8	15.6	20.24
	Task Ari	29.2	30.8	31.2	27.2	22.4	26.0	22.4	21.6	24.8	**24.4**	26.00
	NLMR	**35.2**	**34.8**	**36.8**	**34.8**	**27.2**	**28.0**	**26.8**	**22.0**	**26.0**	**24.4**	**29.60**
ARC-Easy	BASE	60.8	62.4	64.8	66.7	61.6	64.0	44.8	38.0	54.0	44.4	56.15
	CPT	60.8	58.0	62.4	62.7	55.2	52.0	51.2	47.2	49.6	40.4	53.95
	SFT	48.4	52.8	54.4	53.8	46.4	49.2	34.8	30.4	41.6	31.2	44.30
	Linear	64.4	66.4	70.0	67.9	62.0	68.4	49.6	48.4	57.6	53.2	60.79
	TIES	63.2	64.4	**72.8**	70.7	64.8	51.6	48.0	29.2	50.4	22.0	53.71
	DARE	65.6	66.0	71.2	69.9	64.0	64.8	42.8	44.0	56.4	48.0	59.27
	DELLA	67.2	67.6	70.4	**73.5**	65.6	71.6	53.2	48.8	63.2	55.2	63.63
	Task Ari	64.0	66.4	70.0	68.3	61.6	68.4	49.6	48.8	57.6	53.2	60.79
	NLMR	**67.6**	**68.4**	**72.8**	71.5	**67.6**	**73.2**	**60.8**	**53.6**	**65.2**	**58.8**	**65.95**
mCSQA	BASE	36.8	64.0	60.4	56.4	69.2	56.4	38.8	38.0	44.0	40.4	50.44
	CPT	34.8	64.8	50.0	54.8	62.8	36.8	41.6	39.2	44.4	27.6	45.68
	SFT	42.4	67.6	68.8	61.6	47.2	53.6	32.0	30.0	54.0	32.4	48.96
	Linear	48.8	81.6	75.6	71.6	71.2	64.4	**57.2**	61.6	64.8	61.2	65.80
	TIES	**50.4**	80.8	76.0	70.8	**74.4**	65.6	49.2	59.2	63.2	61.2	65.08
	DARE	49.2	79.2	75.2	68.0	71.6	64.4	49.6	**67.2**	**66.0**	61.6	65.20
	DELLA	48.4	**82.8**	74.8	70.8	73.2	65.2	54.4	59.2	62.0	60.0	65.08
	Task Ari	48.4	81.2	75.2	71.6	71.6	64.4	56.8	61.6	64.8	61.2	65.68
	NLMR	49.6	82.0	**76.4**	**72.4**	74.0	**66.8**	56.0	64.8	**66.0**	**63.2**	**67.12**

5.2 Main Results

NLMR consistently outperforms single-source adaptation baselines across both backbones. Table 1 reports multilingual reasoning results on MGSM, ARC-Easy, and mCSQA across 10 languages using **LLaMA-3 8B** and **Gemma-2 2B**. On LLaMA-3 8B, NLMR achieves average accuracies of 38.28, 73.31, and 66.92 on MGSM, ARC-Easy, and mCSQA, respectively. On Gemma-2 2B, NLMR obtains 29.60, 65.95, and 67.12, respectively. Across both backbones, NLMR consistently outperforms BASE, CPT, and English-only SFT, indicating that neither language-only adaptation nor reasoning-only fine-tuning is sufficient for multilingual reasoning, while NLMR effectively integrates both capabilities. To further verify that these gains do not come at the expense of English-side capabilities, Appendix A.2 reports retention results on English reasoning and understanding tasks.

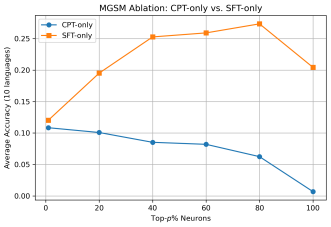


Figure 7: Ablation of **CPT-only** and **SFT-only** neuron selection on MGSM.

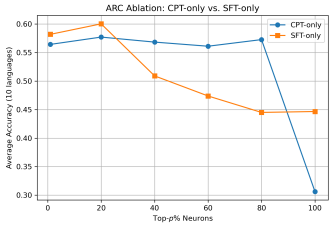


Figure 8: Ablation of **CPT-only** and **SFT-only** neuron selection on ARC-Easy.

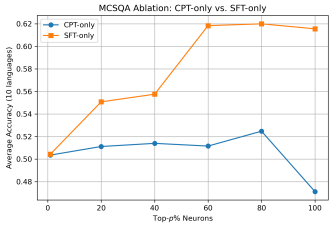


Figure 9: Ablation of **CPT-only** and **SFT-only** neuron selection on mCSQA.

NLMR also consistently outperforms parameter-level merging baselines. We further compare NLMR with parameter-level merging methods, including Linear, TIES, DARE, DELLA, and Task Arithmetic. On LLaMA-3 8B, NLMR surpasses the strongest parameter-level baseline on MGSM (DELLA, 35.44), ARC-Easy (DELLA, 66.23), and mCSQA (DELLA, 66.04). The same trend holds on Gemma-2 2B, where NLMR outperforms the strongest parameter-level baseline on MGSM (Task Arithmetic, 26.00), ARC-Easy (DELLA, 63.63), and mCSQA (Linear, 65.80). These consistent improvements across the two English-centric backbones demonstrate the robustness of neuron-level selective merging for integrating language adaptation and reasoning capabilities.

5.3 Ablation Study

To understand the role of training-induced neuron subspaces, we conduct an ablation study on **Gemma-2 2B** with two controlled variants: (1) **CPT-only**, which merges only CPT-induced neuron updates; and (2) **SFT-only**, which merges only SFT-induced neuron updates. For each variant, we vary the neuron selection ratio and evaluate average accuracy across 10 languages on MGSM, ARC-Easy, and mCSQA. Figures 7–9 report the results.

Effect of CPT-only neuron selection. Across all three datasets, increasing the proportion of CPT-selected neurons generally reduces reasoning performance. This suggests that CPT mainly strengthens language-specific adaptation while providing limited task-specific reasoning signals; excessive CPT-sensitive updates can therefore interfere with downstream reasoning.

Effect of SFT-only neuron selection. In contrast, SFT-only merging exhibits a non-monotonic trend. Selecting a small to moderate fraction of SFT-sensitive neurons improves performance by transferring reasoning ability from English supervision, whereas overly large ratios degrade performance by overriding language-sensitive representations.

Implications for neuron subspace complementarity. Together, these results show that CPT- and SFT-induced neurons form distinct yet complementary subspaces. Over-selecting either subspace harms performance, motivating NLMR to selectively balance language adaptation and reasoning updates during merging.

6 Conclusion

Improving reasoning capability in low-resource languages while preserving language adaptation remains a key challenge for English-centric LLMs. Although SFT and CPT provide complementary improvements, effectively combining these capabilities remains challenging due to coarse-grained parameter-level merging. To mitigation this challenge, we propose **Neuron-Level Locate-then-Merge for Multilingual Reasoning (NLMR)**, the first neuron-level locate-then-merge framework for multilingual reasoning that selectively integrates updates from different objectives. Specifically, NLMR locates reasoning-related neurons from SFT and language-adaptation neurons from CPT, selectively fusing these subsets into a unified model. To validate NLMR, we conduct experiments across 10 languages, 3 reasoning benchmarks, and 2 backbone models. NLMR consistently outperforms single-trained models and strong parameter-level merging baselines. Furthermore, neuron-level analyses reveal distinct yet complementary SFT- and CPT-induced subspaces across languages and model components, providing insights into capability fusion in multilingual LLM adaptation.

AI USE STATEMENT

In this work, we used generative AI tools only to assist with language polishing and improving the clarity and presentation of the manuscript. We did not use generative AI tools to formulate the research methodology, design or conduct experiments, generate or modify experimental results, or draw scientific conclusions. All AI-assisted text was reviewed and revised by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The methodology and experimental protocol required to reproduce our results are documented throughout the paper and appendix. The NLMR procedure, including neuron-level change computation, neuron selection, and selective merging, is formally described in the methodology and Algorithm 1. Section 5.1 specifies the evaluation languages, datasets, evaluation protocol, baselines, backbone models, and training data. Appendix B provides the complete implementation details, including data construction, CPT and SFT training configurations, optimization settings, random seeds, hardware setup, training steps, merging hyperparameters, and evaluation prompt templates.

REFERENCES

AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Takuya Akiba, Makoto Shing, Yujin Tang, Qi Sun, and David Ha. Evolutionary optimization of model merging recipes. *Nature Machine Intelligence*, 7(2):195–204, 2025.

Anton Alexandrov, Veselin Raychev, Mark Niklas Müller, Ce Zhang, Martin Vechev, and Kristina Toutanova. Mitigating catastrophic forgetting in language transfer via model merging. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 17167–17186, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.1000.

Lucas Bandarkar and Nanyun Peng. The unreasonable effectiveness of model merging for cross-lingual transfer in LLMs. In David Ifeoluwa Adelani, Catherine Arnett, Duygu Ataman, Tyler A. Chang, Hila Gonen, Rahul Raja, Fabian Schmidt, David Stap, and Jiayi Wang (eds.), *Proceedings of the 5th Workshop on Multilingual Representation Learning (MRL 2025)*, pp. 131–148, Suzhuo, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-345-6. doi: 10.18653/v1/2025.mrl-main.10.

Lucas Bandarkar, Benjamin Muller, Pritish Yuvraj, Rui Hou, Nayan Singhal, Hongjiang Lv, and Bing Liu. Layer swapping for zero-shot cross-lingual transfer in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Yuheng Chen, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Journey to the center of the knowledge neurons: Discoveries of language-independent knowledge neurons and degenerate knowledge neurons. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17817–17825, Mar. 2024. doi: 10.1609/aaai.v38i16.29735.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2475–2485, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1269. URL https://aclanthology.org/D18-1269/.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8493–8502, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.581.

MohammadReza Davari and Eugene Belilovsky. Model breadcrumbs: Scaling multi-task model merging with sparse masks. In *European Conference on Computer Vision*, pp. 270–287. Springer, 2024.

Pala Tej Deep, Rishabh Bhardwaj, and Soujanya Poria. Della-merging: Reducing interference in model merging through magnitude-based sampling. *arXiv preprint arXiv:2406.11617*, 2024.

Zitao Fang, Guodong Du, Shuyang Yu, Yifei Guo, Yiwei Zhang, Yiyao Cao, Jing Li, Ho-Kin Tang, and Sim Kuan Goh. To see a world in a spark of neuron: Disentangling multi-task interference for training-free model merging. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 15731–15751, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.793.

Yujie Feng, Xu Chu, Yongxin Xu, Guangyuan Shi, Bo Liu, and Xiao-Ming Wu. TaSL: Continual dialog state tracking via task skill localization and consolidation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1266–1279, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.69.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.446.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 30–45, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.3.

Charles Goddard, Shamane Siriwardhana, Malikeh Ehghaghi, Luke Meyers, Vladimir Karpukhin, Brian Benedict, Mark McQuade, and Jacob Solawetz. Arcee's MergeKit: A toolkit for merging large language models. In Franck Dernoncourt, Daniel Preoţiuc-Pietro, and Anastasia Shimorina (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 477–485, Miami, Florida, US, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-industry.36.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023.

Gangwei Jiang, Zhaoyi Li, Defu Lian, and Ying Wei. Refine large language model fine-tuning via instruction vector. *arXiv preprint arXiv:2406.12227*, 2024.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

Takeshi Kojima, Itsuki Okimura, Yusuke Iwasawa, Hitomi Yanaka, and Yutaka Matsuo. On the multilingual ability of decoder-based pre-trained language models: Finding and controlling language-specific neurons. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6919–6971, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.384.

Zihao Li, Yucheng Shi, Zirui Liu, Fan Yang, Ali Payani, Ninghao Liu, and Mengnan Du. Language ranker: A metric for quantifying llm performance across high and low-resource languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 28186–28194, 2025.

Renze Lou, Kai Zhang, and Wenpeng Yin. Large language model instruction following: A survey of progresses and challenges. *Computational Linguistics*, 50(3):1053–1095, September 2024. doi: 10.1162/coli_a_00523.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *IEEE Transactions on Audio, Speech and Language Processing*, 2025.

Michael S Matena and Colin A Raffel. Merging models with fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35:17703–17716, 2022.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 4226–4237, Torino, Italia, May 2024. ELRA and ICCL.

Yusuke Sakai, Hidetaka Kamigaito, and Taro Watanabe. mCSQA: Multilingual commonsense reasoning dataset with unified creation strategy by language models and humans. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14182–14214, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.844.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. Language-specific neurons: The key to multilingual capabilities in large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5701–5715, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.309.

Mingxu Tao, Chen Zhang, Quzhe Huang, Tianyao Ma, Songfang Huang, Dongyan Zhao, and Yansong Feng. Unlocking the potential of model merging for low-resource languages. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 8705–8720, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.508.

Gemma Team. Gemma. 2024. doi: 10.34740/KAGGLE/M/3301. URL `https://www.kaggle.com/m/3301`.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pp. 23965–23998. PMLR, 2022.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 483–498, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.41.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36:7093–7115, 2023.

Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666*, 2024.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*, 2024.

Zeping Yu and Sophia Ananiadou. Neuron-level knowledge attribution in large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 3267–3280, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.191.

Zeping Yu and Sophia Ananiadou. Locate-then-merge: Neuron-level parameter fusion for mitigating catastrophic forgetting in multimodal LLMs. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 7065–7078, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.372.

Zhihao Zhang, Jun Zhao, Qi Zhang, Tao Gui, and Xuanjing Huang. Unveiling linguistic regions in large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6228–6247, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.338.

Yiran Zhao, Wenxuan Zhang, Huiming Wang, Kenji Kawaguchi, and Lidong Bing. AdaMergeX: Cross-lingual transfer with large language models via adaptive adapter merging. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 9785–9800, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.493.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics.

A Additional Results and Analysis

A.1 Comparison with Sequential CPT→SFT Adaptation

To further examine whether language adaptation and reasoning adaptation can be effectively combined through conventional sequential training, we compare NLMR with a sequential CPT→SFT baseline on **Gemma-2 2B**. For each target language, the CPT→SFT model is initialized from the

corresponding language-specific CPT checkpoint and subsequently undergoes full-parameter SFT on the corresponding English reasoning dataset. Specifically, GSM8K is used for MGSM, while ARC-Easy and mCSQA use their respective English training sets. The SFT stage follows the same training configuration as the task-specific SFT models described in Appendix B.2. Therefore, CPT→SFT and NLMR are constructed from the same language adaptation and task supervision, while differing in how the two sources of adaptation are integrated: CPT→SFT applies them sequentially, whereas NLMR independently locates and selectively merges the neuron updates induced by CPT and SFT.

Table 2 reports the complete per-language results. NLMR consistently outperforms sequential CPT→SFT across all three reasoning benchmarks. On average over the ten languages, NLMR improves upon CPT→SFT by **7.88** points on MGSM, **34.11** points on ARC-Easy, and **27.48** points on mCSQA. Across the 30 language–task combinations, NLMR achieves higher accuracy in **29 cases**, with Thai on MGSM being the only exception.

Table 2: Comparison between sequential CPT→SFT adaptation and NLMR on **Gemma-2 2B** across 10 languages. CPT→SFT sequentially performs language-specific continual pretraining followed by full-parameter supervised fine-tuning, whereas NLMR selectively integrates CPT- and SFT-induced neuron updates. The better result for each language and dataset is shown in **bold**.

Dataset	Method	Language										Avg.
		ru	de	es	fr	ja	zh	sw	te	th	bn	
MGSM	CPT→SFT	29.6	21.2	28.8	24.4	14.8	23.2	13.2	20.8	**27.2**	14.0	21.72
	NLMR	**35.2**	**34.8**	**36.8**	**34.8**	**27.2**	**28.0**	**26.8**	**22.0**	26.0	**24.4**	**29.60**
ARC-Easy	CPT→SFT	19.6	21.6	25.2	42.4	41.2	34.0	26.8	46.8	33.2	27.6	31.84
	NLMR	**67.6**	**68.4**	**72.8**	**71.5**	**67.6**	**73.2**	**60.8**	**53.6**	**65.2**	**58.8**	**65.95**
mCSQA	CPT→SFT	46.0	20.4	48.0	66.0	28.8	21.6	38.4	20.0	55.2	52.0	39.64
	NLMR	**49.6**	**82.0**	**76.4**	**72.4**	**74.0**	**66.8**	**56.0**	**64.8**	**66.0**	**63.2**	**67.12**

The substantial gap between the two approaches indicates that simply applying language adaptation and reasoning adaptation in sequence is insufficient to reliably preserve both capabilities. In CPT→SFT, the second-stage SFT updates parameters that have already been adapted during CPT, which can introduce interference with the previously acquired language-specific representations. This effect is particularly pronounced on ARC-Easy and mCSQA, where CPT→SFT achieves substantially lower average accuracy than NLMR. In contrast, NLMR separates the two adaptation processes, independently identifies the neuron updates most strongly associated with language adaptation and reasoning adaptation, and selectively integrates them into the shared base model. The resulting performance suggests that neuron-level selective integration provides a more effective mechanism for preserving complementary capabilities than conventional sequential adaptation.

A.2 English-Side Retention Analysis

To further examine whether NLMR mitigates catastrophic forgetting on English-side capabilities, we conduct an additional English-side retention evaluation on **Gemma-2 2B**. We evaluate English reasoning performance on the corresponding task benchmark. We additionally adopt XNLI (Conneau et al., 2018) as an English language-understanding benchmark and randomly sample 250 English test examples for evaluation.

Table 3 summarizes the comparison among CPT, SFT, and NLMR models. We analyze the results from two complementary perspectives.

Compared with CPT models, NLMR substantially improves English reasoning performance on the corresponding tasks. Specifically, NLMR improves English MGSM, ARC-Easy, and mCSQA performance by 28.76, 5.92, and 24.12 points over CPT models, respectively. This indicates that language-specific CPT alone mainly preserves or adapts linguistic competence, but is insufficient for recovering task-specific reasoning ability. NLMR effectively incorporates the reasoning capability from the SFT branch while retaining a comparable level of English language understanding. Although NLMR is slightly lower than CPT on XNLI for ARC-Easy and mCSQA, the decreases are modest compared with the large task-side gains.

Table 3: English-side retention summary on **Gemma-2 2B**. CPT and NLMR results are averaged over the ten target-language variants. Δ_{CPT} and Δ_{SFT} denote the improvement of NLMR over CPT and task-specific SFT models, respectively. For CPT, the XNLI result is shared across task comparisons because CPT models are language-specific but not task-specific.

Task	Metric	CPT Avg.	SFT	NLMR Avg.	Δ_{CPT}	Δ_{SFT}
MGSM	XNLI	35.24	32.40	**42.68**	+7.44	+10.28
	Corresponding Task	13.80	36.80	**42.56**	+28.76	+5.76
ARC-Easy	XNLI	**35.24**	0.00	34.36	-0.88	+34.36
	Corresponding Task	67.60	64.00	**73.52**	+5.92	+9.52
mCSQA	XNLI	**35.24**	0.00	32.08	-3.16	+32.08
	Corresponding Task	58.00	79.20	**82.12**	+24.12	+2.92

Table 4: English-side evaluation of task-specific SFT models on **Gemma-2 2B**.

SFT Model	XNLI (EN)	Corresponding Task (EN)
MGSM(GSM8K)-SFT	32.40	36.80
ARC-Easy-SFT	0.00	64.00
mCSQA-SFT	0.00	79.20

Compared with task-specific SFT models, NLMR better preserves English language-understanding ability while also improving the corresponding English task performance. For ARC-Easy and mCSQA, SFT models obtain 0.00 on English XNLI, suggesting severe degradation of general English-side language understanding. In contrast, NLMR restores English XNLI to 34.36 and 32.08, respectively. Meanwhile, NLMR also improves the corresponding English task accuracy by 9.52 and 2.92 points over SFT-only. For MGSM, NLMR improves both English XNLI and English task accuracy over SFT.

Overall, these results show that NLMR does not obtain multilingual reasoning gains by sacrificing English-side capabilities. Instead, it achieves a better balance between preserving English language understanding and maintaining or improving English reasoning performance.

The summary results in Table 3 are computed from the detailed evaluations reported below. Specifically, Table 4 reports the English-side performance of task-specific SFT models. Table 5 further reports the per-target-language English-side results of CPT and NLMR models, where each language column corresponds to the target language used to construct the CPT branch or the corresponding merged model.

A.3 Module-Wise Analysis Across Additional Languages

To complement the module-wise neuron overlap analysis presented in the main paper, this subsection reports results for the remaining 9 languages (*bn, es, fr, ja, ru, sw, te, th, zh*). Following the analysis in Section 4.3, we present module-wise overlap curves for each language in Figures 10 (Bengali, Spanish, French, Japanese, Russian, and Swahili) and 11 (Telugu, Thai, and Chinese), showing consistent patterns across attention and feed-forward modules.

A.4 Score Distributions and Rank Correlation for Additional Languages

This subsection presents neuron importance score distributions and CPT–SFT rank correlation results for the remaining 9 languages (*bn, es, fr, ja, ru, sw, te, th, zh*), complementing the German results reported in Section 4.4 in Figure 12 - 29. Following the same analysis protocol, we observe trends that are consistent across languages: CPT induces broader and higher-magnitude importance score distributions, while SFT concentrates importance on a smaller subset of neurons. Rank scatter plots similarly reveal positive but partial correlations between CPT and SFT, indicating that language- and reasoning-oriented training emphasize distinct yet complementary neuron sub-

Table 5: Detailed English-side evaluation on **Gemma-2 2B**. Each language column indicates the target-language CPT model or the corresponding target-language NLMR merged model, while all evaluations are performed on English examples.

Model	Eval (EN)	ES	FR	DE	RU	ZH	JA	TH	SW	BN	TE	Avg.
CPT target-language models												
CPT	XNLI	32.80	32.80	41.60	35.60	39.60	31.20	34.00	31.60	34.80	38.40	35.24
CPT	MGSM	9.20	15.20	20.80	12.00	20.00	18.00	12.00	8.00	8.40	14.40	13.80
CPT	ARC-Easy	68.00	67.20	69.60	67.20	61.60	66.80	65.20	76.80	68.00	65.60	67.60
CPT	mCSQA	56.80	59.60	65.60	60.40	47.60	58.80	60.00	64.40	52.80	54.00	58.00
NLMR merged models												
NLMR-MGSM (GSM8K)	MGSM	47.20	46.80	47.20	47.60	38.40	33.20	39.60	42.80	37.60	45.20	42.56
NLMR-MGSM (GSM8K)	XNLI	40.40	40.40	39.60	40.00	46.40	45.60	38.40	42.00	46.40	47.60	42.68
NLMR-ARC-Easy	ARC-Easy	77.60	80.80	75.60	70.00	77.20	71.20	71.60	78.80	64.40	68.00	73.52
NLMR-ARC-Easy	XNLI	31.20	43.20	37.60	31.60	34.00	32.80	30.80	32.80	31.60	38.00	34.36
NLMR-mCSQA	mCSQA	83.60	82.40	83.20	83.60	80.80	83.60	84.80	83.20	74.00	82.00	82.12
NLMR-mCSQA	XNLI	31.60	32.00	31.20	32.00	35.20	31.60	31.20	31.60	32.00	32.40	32.08

spaces. These results confirm that the neuron-level patterns observed for German generalize across languages.

B Experiment Reproducibility

B.1 Details of Training Data

For the Continual Pretraining (CPT) models, training data is drawn from the CulturaX (Nguyen et al., 2024) corpus across 10 target languages. For Swahili, we use all 66,506 available instances. For each of the other nine languages, we use the first one million instances from the corresponding language corpus. The selected CPT corpora are shuffled with a fixed random seed of 42 before training.

To mitigate catastrophic forgetting during CPT, we additionally sample English data from CulturaX using a target language : English ratio of 3:1, and merge it with the respective target-language corpus to construct the final training set.

For the Supervised Fine-Tuning (SFT) models, we use the complete official training splits of GSM8K (Cobbe et al., 2021), ARC-Easy, and mCSQA, and train a separate task-specific model for each dataset.

B.2 Training Details

B.2.1 Training Configuration

For CPT, we train with a batch size of 32 using the AdamW optimizer with hyperparameters β_1=0.9, β_2=0.95, and a learning rate of 2×10^{-5}. A cosine annealing schedule is applied to adjust the learning rate over training. For SFT, the training uses a batch size of 16, the Adam optimizer with a learning rate of 1.5×10^{-5}, and a cosine annealing schedule. Each SFT model is trained for 3 epochs on its corresponding task dataset. Both CPT and SFT training pipelines are implemented using the LLaMA-Factory framework (Zheng et al., 2024).

B.2.2 Supervised Fine-Tuning Details

We perform full-parameter supervised fine-tuning (SFT) on GSM8K, ARC-Easy, and mCSQA using the same base model and largely consistent training configurations.

Training Setup. For all 3 datasets, we adopt full fine-tuning with FlashAttention-2 enabled. The maximum input length is set to 1024 tokens. We train each model for 3 epochs using a cosine learning rate schedule with a warmup ratio of 0.03. The learning rate is fixed to 1.5×10^{-5}, and gradient clipping with a maximum norm of 1.0 is applied for stability.

We use a per-device batch size of 2 and accumulate gradients for 4 steps, resulting in an effective batch size of 8. All experiments are conducted with bfloat16 precision and gradient checkpointing enabled to reduce memory consumption. DeepSpeed is employed for distributed training with ZeRO-based optimizer state partitioning.

Task-Specific Fine-Tuning.

- **GSM8K**: The model is fine-tuned to generate step-by-step mathematical reasoning followed by a final answer marker (e.g., "#### 72"), consistent with standard GSM8K supervision.
- **ARC-Easy**: The model is fine-tuned on four-way multiple-choice science questions, where the output is restricted to a single option label in the format "Answer: A/B/C/D".
- **mCSQA**: The model is fine-tuned on five-way commonsense multiple-choice questions, with outputs formatted as "Answer: A/B/C/D/E".

All fine-tuned checkpoints are saved at the end of each epoch, and training loss curves are monitored throughout optimization.

B.2.3 Continual Pretraining Details

To obtain language-adapted models for multilingual transfer, we perform continual pretraining (CPT) on language-specific corpora covering 10 target languages.

Training Data. We use text from CulturaX (Nguyen et al., 2024) for Russian, German, Spanish, French, Japanese, Chinese, Swahili, Telugu, Thai, and Bengali. For Swahili, we use all 66,506 available instances. For each of the other nine languages, we use the first one million instances from the corresponding language corpus. The selected CPT corpora are shuffled with a fixed random seed of 42 before training. As described in Appendix B, English data is additionally mixed with each target-language corpus at a target-language-to-English ratio of 3:1.

Training Setup. All CPT experiments adopt the same training configuration. We set the maximum sequence length to 4096 tokens and train each model for one epoch. We use AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, a learning rate of 2.0×10^{-5}, and a weight decay of 0.0. A cosine learning-rate schedule with a warmup ratio of 0.1 is used throughout training.

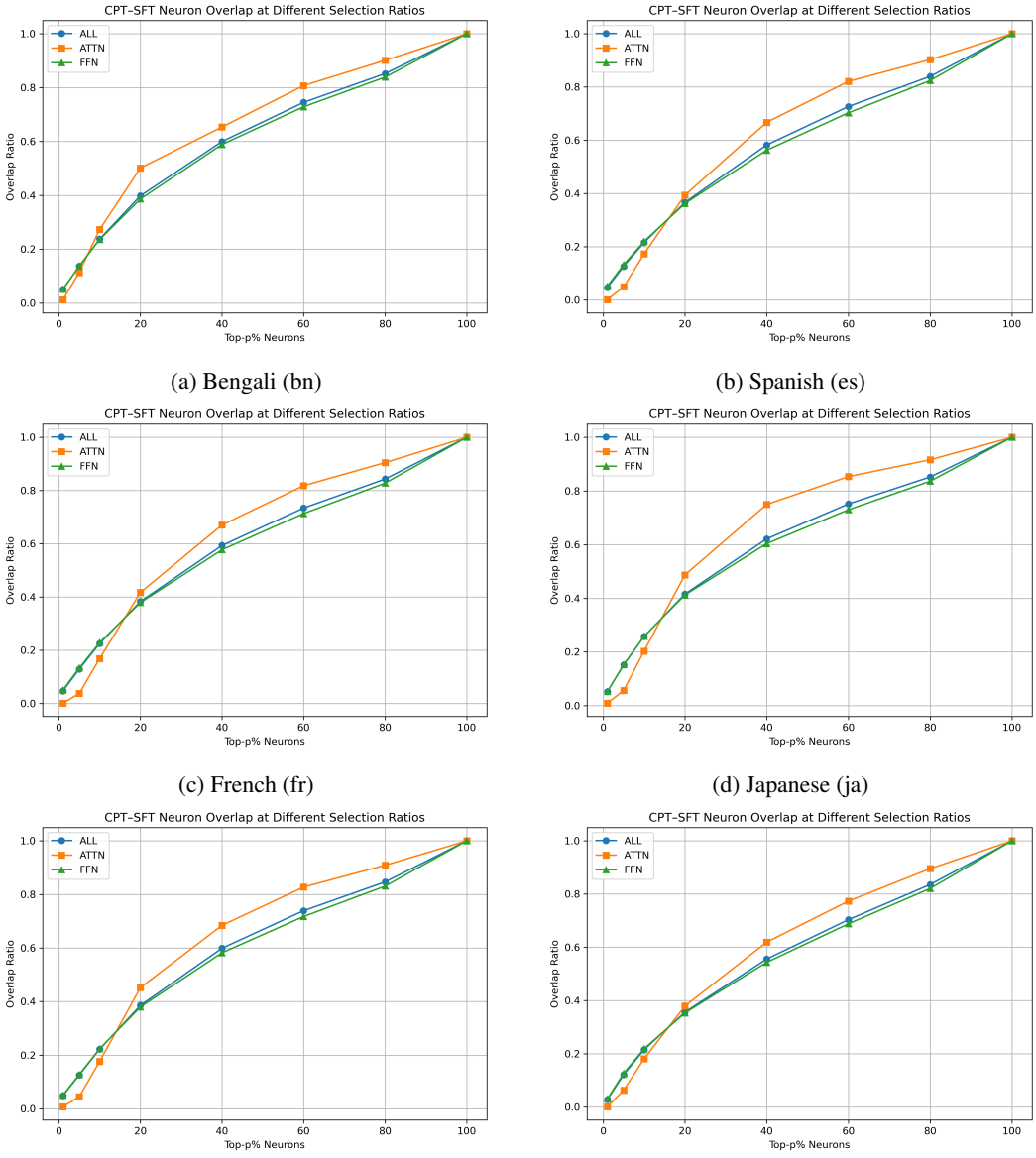
We use a per-device batch size of 2 with gradient accumulation over 8 steps. With two GPUs, this corresponds to an effective global batch size of 32. Training is conducted in bfloat16 precision with gradient checkpointing enabled, and gradient clipping with a maximum norm of 1.0 is applied for stability. All CPT experiments are conducted on a single node equipped with two NVIDIA A800 80GB GPUs using DeepSpeed ZeRO-3.

Training Steps. The exact number of optimization steps for each target language is reported in Table 6.

B.3 Merging Configurations

For all NLMR experiments, the merging weights are fixed to $w_{\text{CPT}} = w_{\text{SFT}} = 1.0$. The neuron selection ratios are selected from the same search space $p_{\text{CPT}}, p_{\text{SFT}} \in \{1\%, 20\%, 40\%, 60\%, 80\%, 99\%\}$ for all models, datasets, and target languages. While the merging weights and search space remain fixed throughout the experiments, the selected neuron ratios are determined separately for each dataset–language combination.

Table 7 reports the exact configurations used for the main multilingual reasoning results. Each entry denotes $(p_{\text{CPT}}, p_{\text{SFT}})$, where p_{CPT} and p_{SFT} specify the proportions of CPT- and SFT-induced neurons selected for merging, respectively.



(a) Bengali (bn)

(b) Spanish (es)

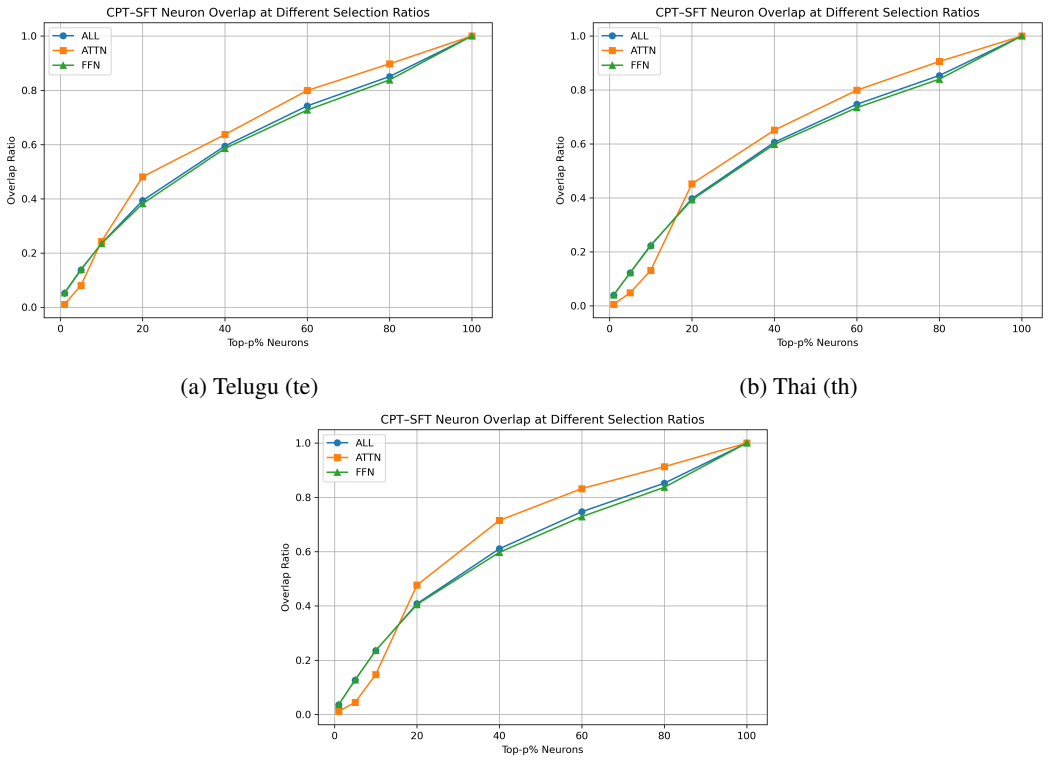
(c) French (fr)

(d) Japanese (ja)

(e) Russian (ru)

(f) Swahili (sw)

Figure 10: Module-wise neuron overlap between CPT and SFT for Bengali, Spanish, French, Japanese, Russian, and Swahili.



(a) Telugu (te)

(b) Thai (th)

(c) Chinese (zh)

Figure 11: Module-wise neuron overlap between CPT and SFT for Telugu, Thai, and Chinese.

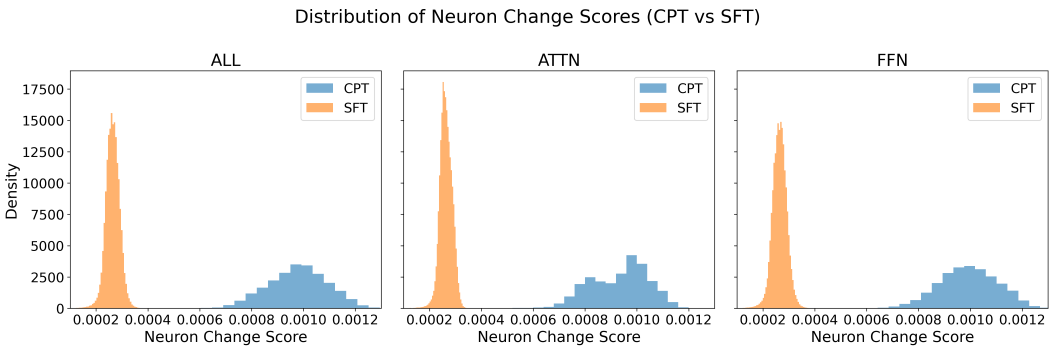


Figure 12: Distribution of neuron importance scores under CPT and SFT for Bengali (bn).

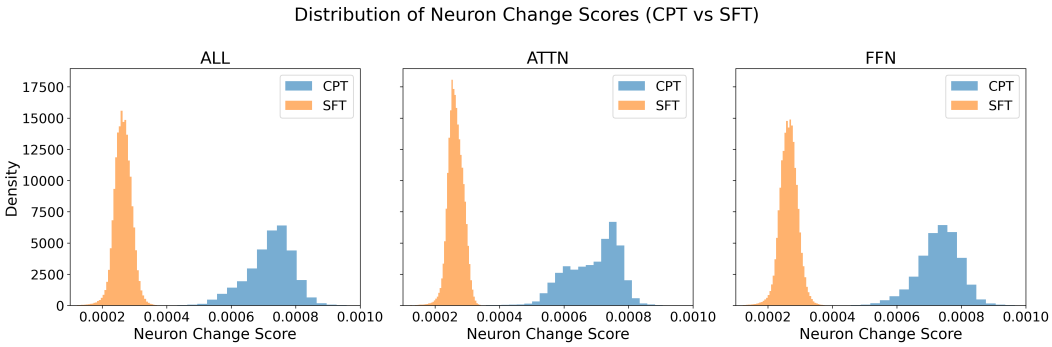


Figure 13: Distribution of neuron importance scores under CPT and SFT for Spanish (es).

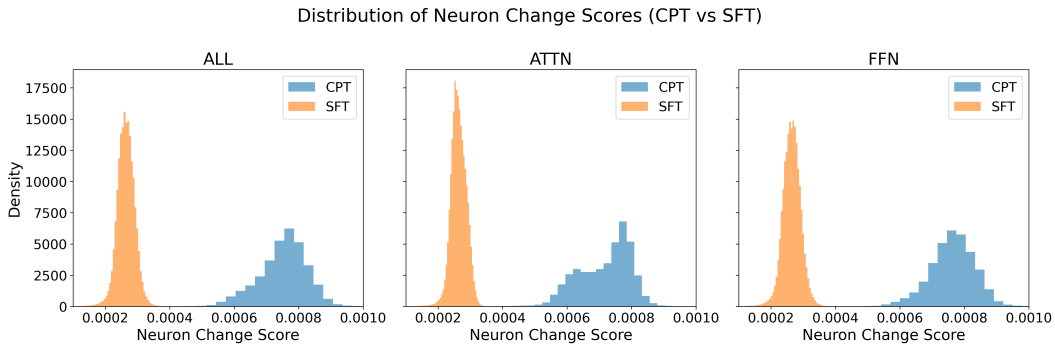


Figure 14: Distribution of neuron importance scores under CPT and SFT for French (fr).

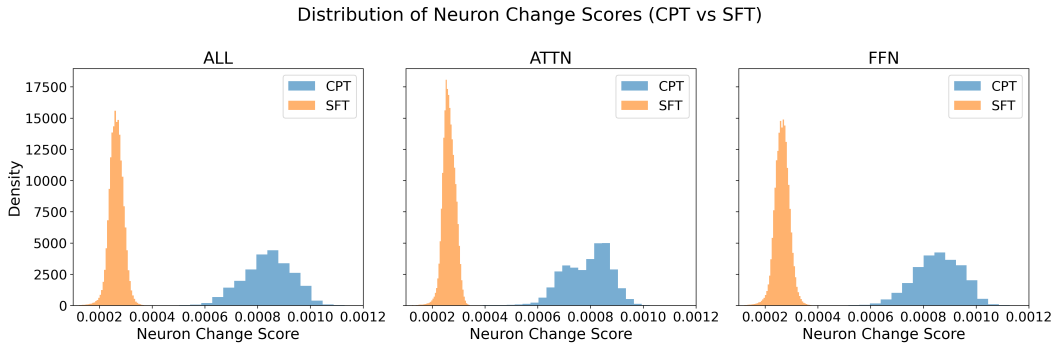


Figure 15: Distribution of neuron importance scores under CPT and SFT for Japanese (ja).

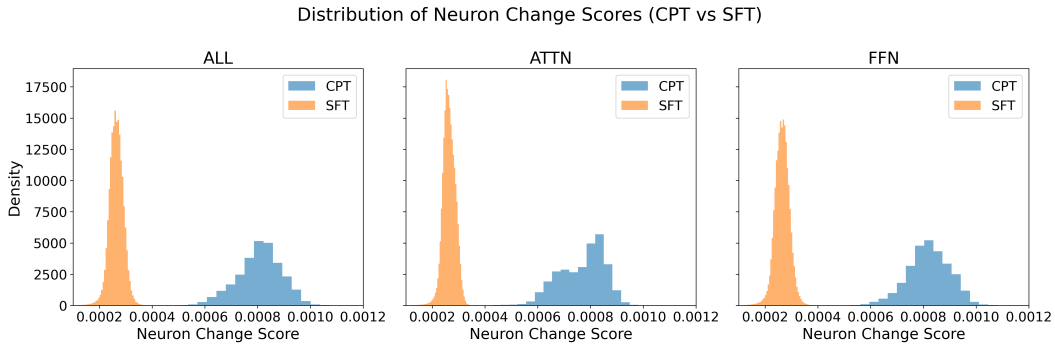


Figure 16: Distribution of neuron importance scores under CPT and SFT for Russian (ru).

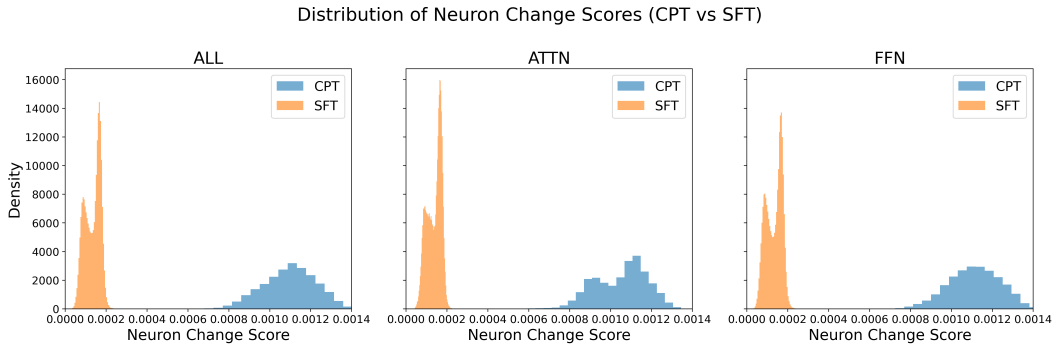


Figure 17: Distribution of neuron importance scores under CPT and SFT for Swahili (sw).

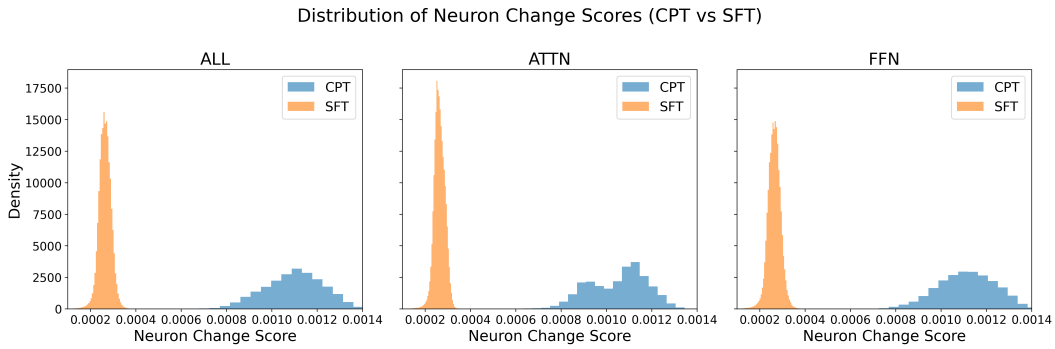


Figure 18: Distribution of neuron importance scores under CPT and SFT for Telugu (te).

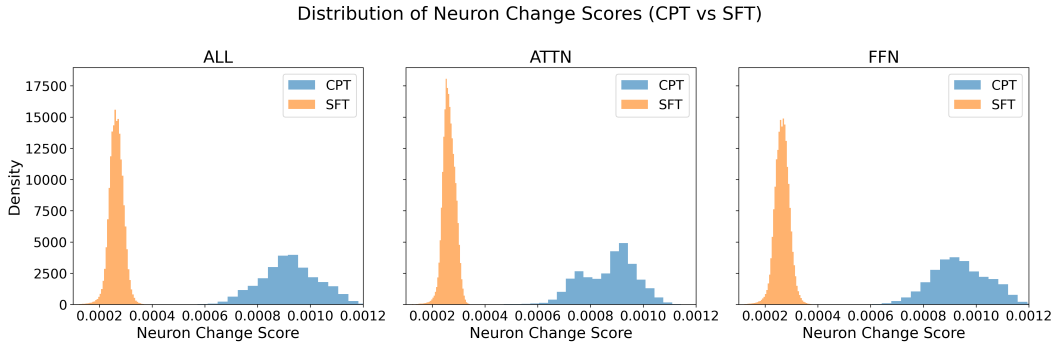


Figure 19: Distribution of neuron importance scores under CPT and SFT for Thai (th).

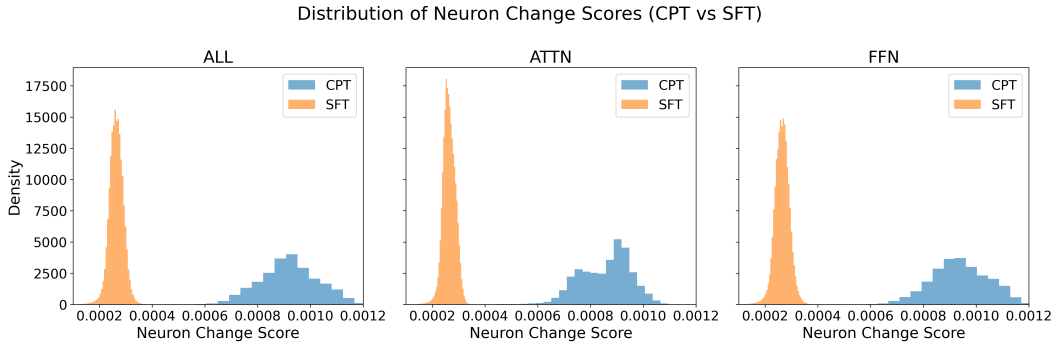


Figure 20: Distribution of neuron importance scores under CPT and SFT for Chinese (zh).

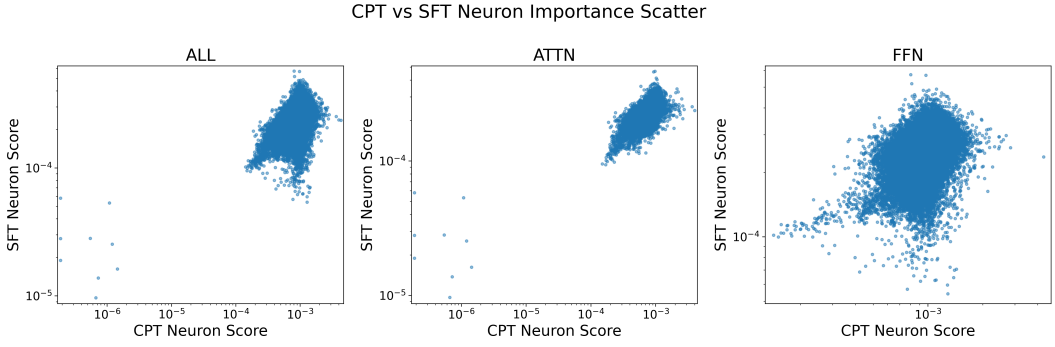


Figure 21: Neuron-level interaction between CPT- and SFT-induced subspaces for Bengali (bn).

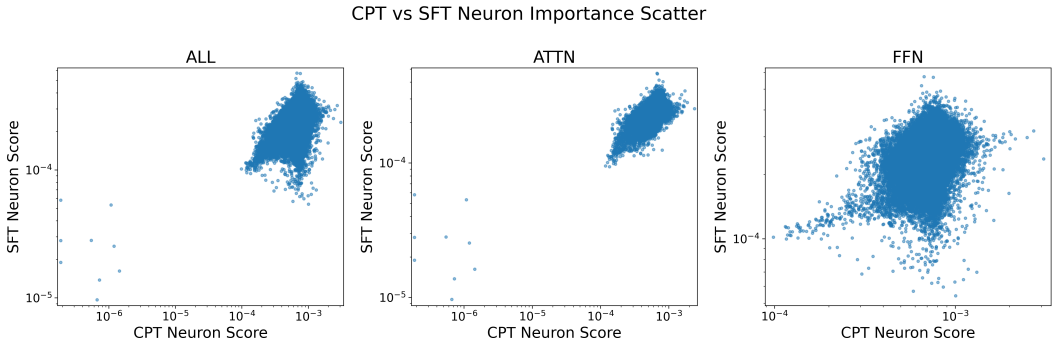


Figure 22: Neuron-level interaction between CPT- and SFT-induced subspaces for Spanish (es).

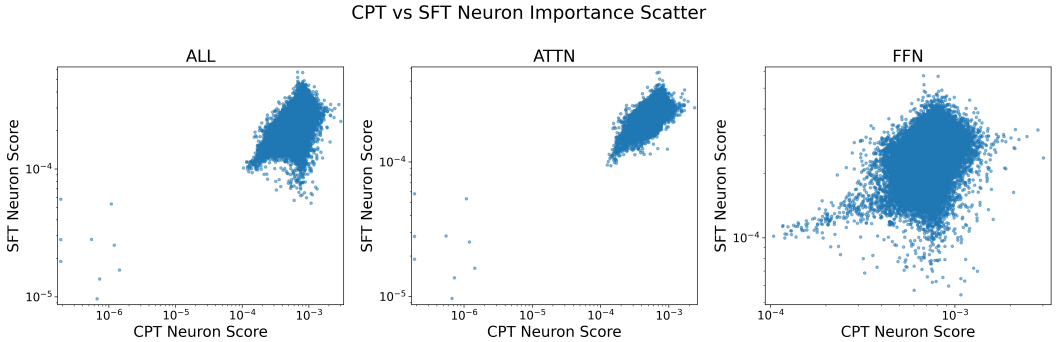


Figure 23: Neuron-level interaction between CPT- and SFT-induced subspaces for French (fr).

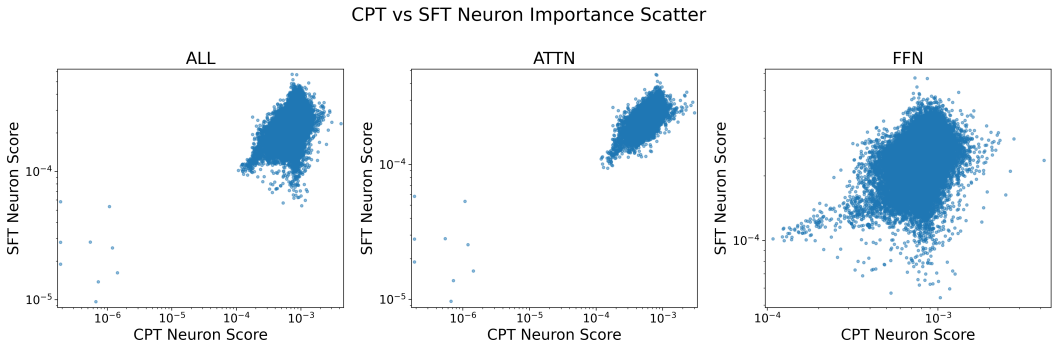


Figure 24: Neuron-level interaction between CPT- and SFT-induced subspaces for Japanese (ja).

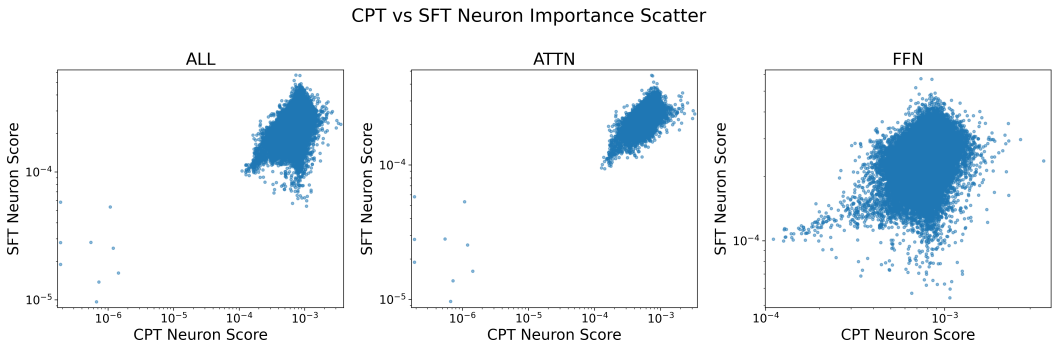


Figure 25: Neuron-level interaction between CPT- and SFT-induced subspaces for Russian (ru).

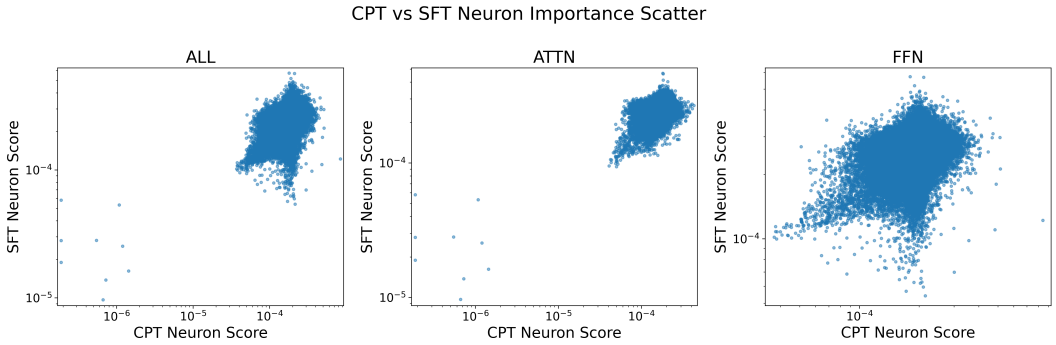


Figure 26: Neuron-level interaction between CPT- and SFT-induced subspaces for Swahili (sw).

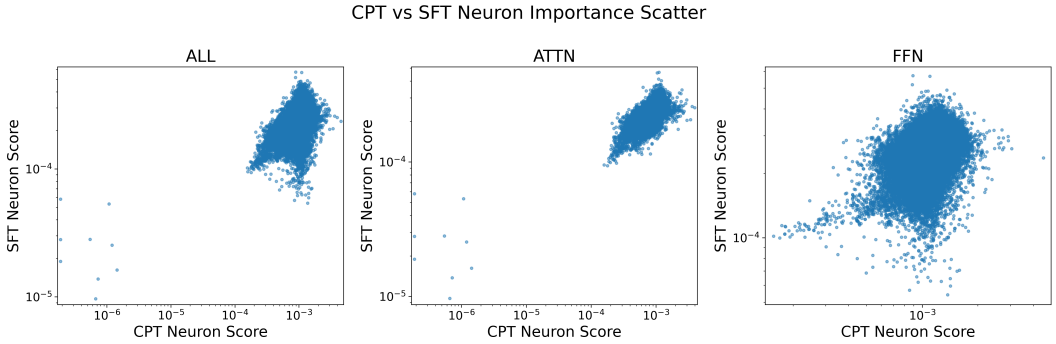


Figure 27: Neuron-level interaction between CPT- and SFT-induced subspaces for Telugu (te).

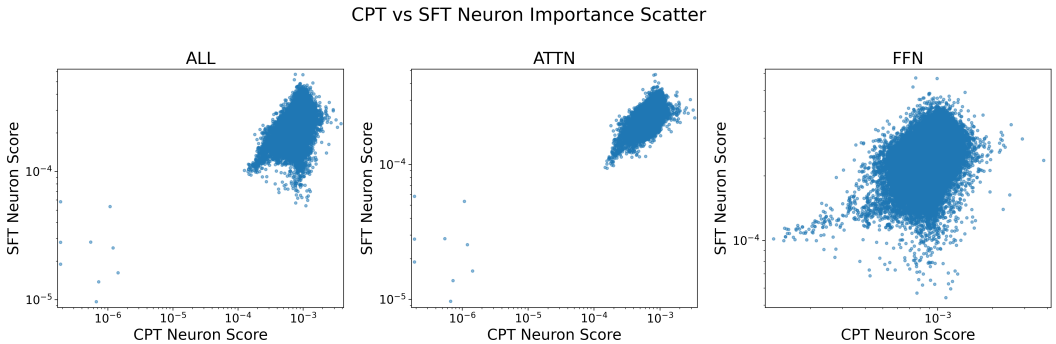


Figure 28: Neuron-level interaction between CPT- and SFT-induced subspaces for Thai (th).

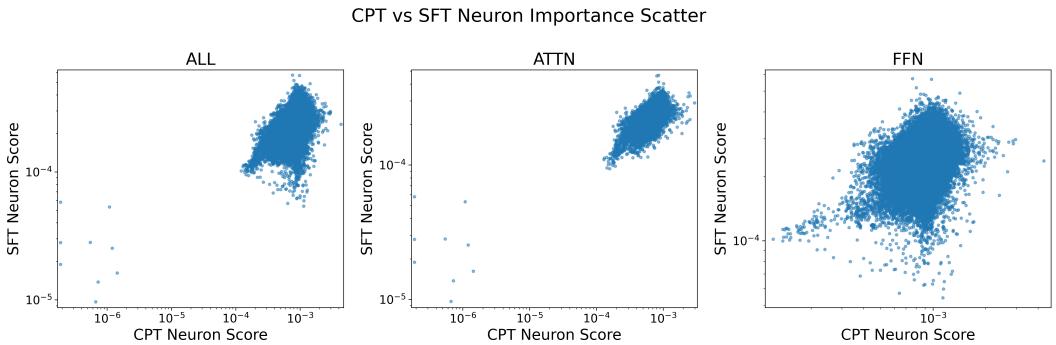


Figure 29: Neuron-level interaction between CPT- and SFT-induced subspaces for Chinese (zh).

Table 6: Number of CPT training steps for each target language.

Language	Steps	Language	Steps
Bengali	12,386	German	8,489
Spanish	8,160	French	8,476
Japanese	9,103	Russian	9,547
Swahili	437	Telugu	15,409
Thai	10,592	Chinese	10,462

Table 7: Neuron selection ratios used for the main NLMR results. Each entry denotes $(p_{\text{CPT}}, p_{\text{SFT}})$. The merging weights are fixed to $w_{\text{CPT}} = w_{\text{SFT}} = 1.0$.

Language	MGSM	ARC-Easy	mCSQA
Russian	(1%, 80%)	(60%, 20%)	(20%, 80%)
German	(1%, 80%)	(80%, 20%)	(60%, 99%)
Spanish	(1%, 80%)	(40%, 20%)	(80%, 80%)
French	(1%, 80%)	(20%, 1%)	(99%, 80%)
Japanese	(20%, 40%)	(80%, 20%)	(99%, 80%)
Chinese	(1%, 40%)	(20%, 20%)	(60%, 60%)
Swahili	(60%, 60%)	(80%, 20%)	(80%, 99%)
Telugu	(1%, 60%)	(80%, 1%)	(99%, 80%)
Thai	(40%, 80%)	(80%, 20%)	(99%, 80%)
Bengali	(1%, 40%)	(80%, 20%)	(99%, 80%)

B.4 Prompt Templates

One-shot Prompt Templates for MGSM For MGSM evaluation, we adopt one-shot Chain-of-Thought (CoT) prompting with a single in-context example in the target language. Figure 30 shows representative examples used during testing.

One-shot Prompt Template for ARC-Easy For ARC-Easy evaluation, we use a one-shot direct prompting strategy for multiple-choice questions. Each test instance is preceded by a single English example demonstrating the expected input format and answer style. Figure 31 presents a representative prompt used during evaluation.

One-shot Prompt Template for mCSQA For mCSQA evaluation, we adopt a one-shot direct prompting strategy for five-choice multiple-choice questions. Each test instance is preceded by a single English example illustrating the required question format and answer style. Figure 32 shows a representative prompt used in our experiments.

Training Prompt Formats Figure 33 summarizes the prompt formats used during supervised fine-tuning for GSM8K, ARC-Easy, and mCSQA. Each task adopts a unified instruction–input–output structure with task-specific answer formats.

C Case Study

We conduct a case study on 10 languages (bn, de, es, fr, ja, ru, sw, te, th, zh) across 3 datasets: MGSM, ARC-Easy, and mCSQA. The selected cases mainly focus on scenarios where our neuron-level merging method outperforms baseline methods. For each language–dataset pair, we present one representative example, and all predictions from different methods are shown for direct comparison. The case study results are illustrated in Figure 34 - 63, where Figure 34 – 43 correspond to MGSM, Figure 44 – 53 correspond to ARC-Easy, and Figure 54 – 63 correspond to mCSQA, covering all 10 languages.

D Qualitative Error Analysis

To better understand the remaining limitations of NLMR, we conduct a compact qualitative error analysis on representative failure cases across languages and tasks. Rather than reporting language-specific or task-specific error statistics, we group recurring failure cases by their dominant error patterns. For MGSM, where the model produces chain-of-thought rationales, we include the original CoT output in the corresponding figures. For ARC-Easy and mCSQA, which are multiple-choice tasks, we conduct an observable answer-level diagnosis based on the question, candidate options, model prediction, and gold answer.

Prompt for MGSM Evaluation

Prompt Type
One-shot CoT

Example Content
Example:
Question: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
Step-by-step answer: Roger starts with 5 tennis balls. 2 cans with 3 tennis balls each add 6 more tennis balls, so $5 + 6 = 11$. The answer is 11.
Answer: 11
Question:
(Question in the target language)
Expected Output:
Step-by-step answer: [English reasoning]
Answer: [number]

Figure 30: Representative one-shot prompt examples used for MGSM evaluation. The following elements are highlighted in distinct colors for clarity: MGSM, Prompt Type, Example Content, Example, Question, and Expected Output.

Prompt for ARC-Easy Evaluation

Prompt Type
One-shot Direct

Example Content
Example:
Question: Which factor will most likely cause a person to develop a fever?
A: a leg muscle relaxing after exercise
B: a bacterial population in the bloodstream
C: several viral particles on the skin
D: carbohydrates being digested in the stomach
Answer: B
Question:
(ARC-Easy test question)
A: (option A)
B: (option B)
C: (option C)
D: (option D)
Expected Output:
Answer: A / B / C / D

Figure 31: Representative one-shot direct prompt used for ARC-Easy evaluation. The following elements are highlighted in distinct colors for clarity: ARC-Easy, Prompt Type, Example Content, Example, Question, and Expected Output.

Figures 64, 65, 66, and 67 present representative NLMR failure cases. These examples do not contradict our main finding that NLMR better balances language adaptation and reasoning transfer. Instead, they illustrate the remaining limitations after neuron-level merging, including numerical reasoning errors, relational constraint decomposition errors, knowledge application errors, and fine-grained option grounding errors.

Prompt for mCSQA Evaluation

Prompt Type
One-shot Direct

Example Content
Example:
Question: What is the most likely reason someone would wear sunglasses on a sunny day?
A: To improve their hearing
B: To protect their eyes from bright light
C: To help them run faster
D: To keep their head warm
E: To make their shoes more comfortable
Answer: B
Question:
(mCSQA test question)
A: (option A)
B: (option B)
C: (option C)
D: (option D)
E: (option E)
Expected Output:
Answer: A / B / C / D / E

Figure 32: Representative one-shot direct prompt used for mCSQA evaluation. The following elements are highlighted in distinct colors for clarity: mCSQA, Prompt Type, Example Content, Example, Question, and Expected Output.

Prompt for SFT on GSM8K

Instruction
Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?
Output
Natalia sold 48/2 = 24 clips in May.
Natalia sold 48+24 = 72 clips altogether in April and May.
72

Prompt for SFT on ARC-Easy

Instruction
Answer the following multiple-choice question by selecting the correct option. Output the final answer as Answer: A", Answer: B", Answer: C", or Answer: D".
Input
Question: Which factor will most likely cause a person to develop a fever?
A: a leg muscle relaxing after exercise
B: a bacterial population in the bloodstream
C: several viral particles on the skin
D: carbohydrates being digested in the stomach
Output
Answer: B

Prompt for SFT on mCSQA

Instruction
Answer the following multiple-choice question by selecting the correct option. Output the final answer as Answer: A", Answer: B", Answer: C", Answer: D", or Answer: E".
Input
Question: What activity involves going to different places and experiencing new cultures and environments?
A: travel
B: sightseeing
C: departing and returning flights
D: visiting
E: exploring local cuisine
Output
Answer: A

Figure 33: Representative training prompt formats used for supervised fine-tuning on GSM8K, ARC-Easy, and mCSQA. The following elements are highlighted in distinct colors for clarity: GSM8K, ARC-Easy, mCSQA, Instruction, Input, and Output.

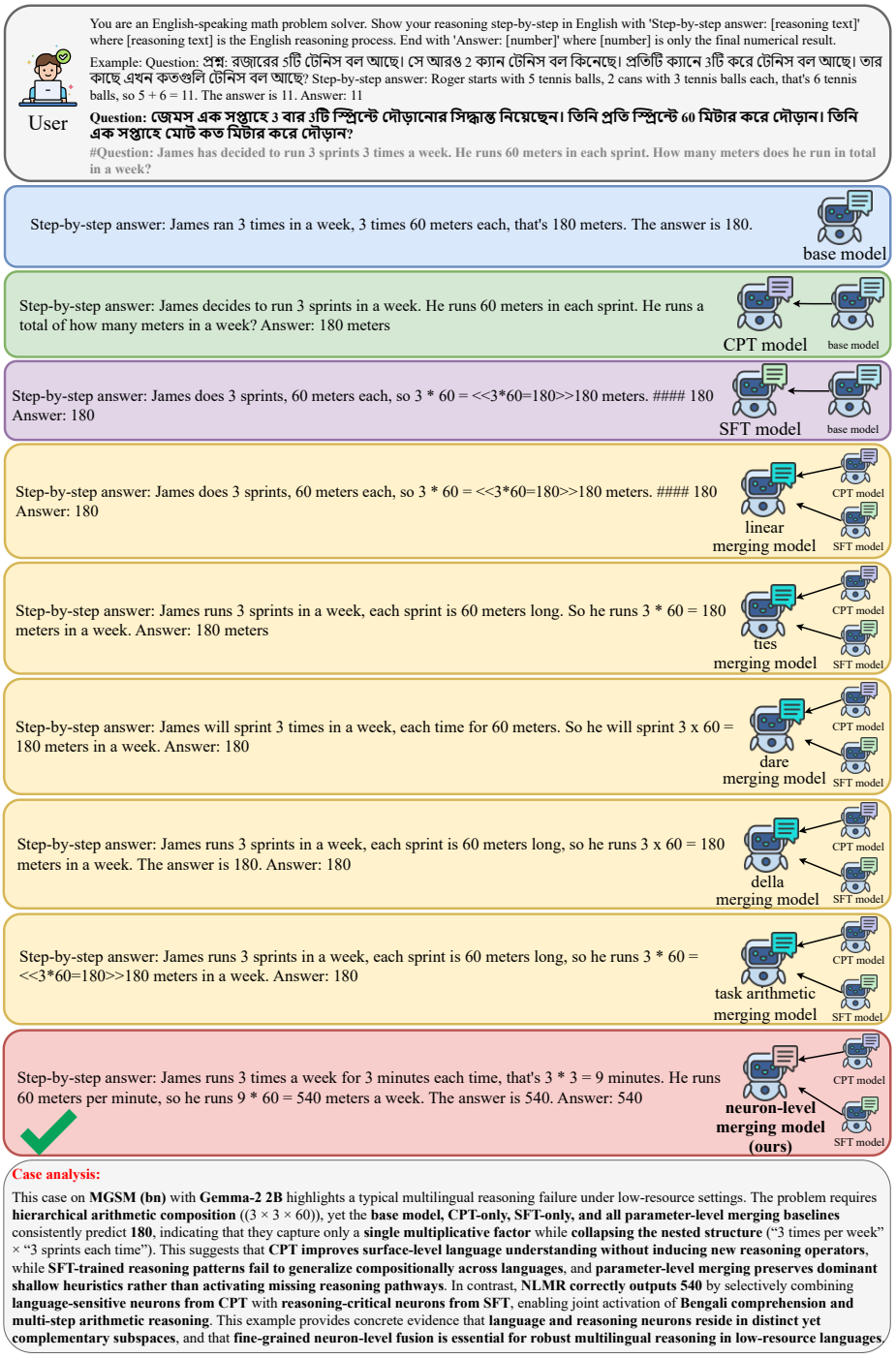


Figure 34: Case study on the MGSM dataset in Bengali (bn).

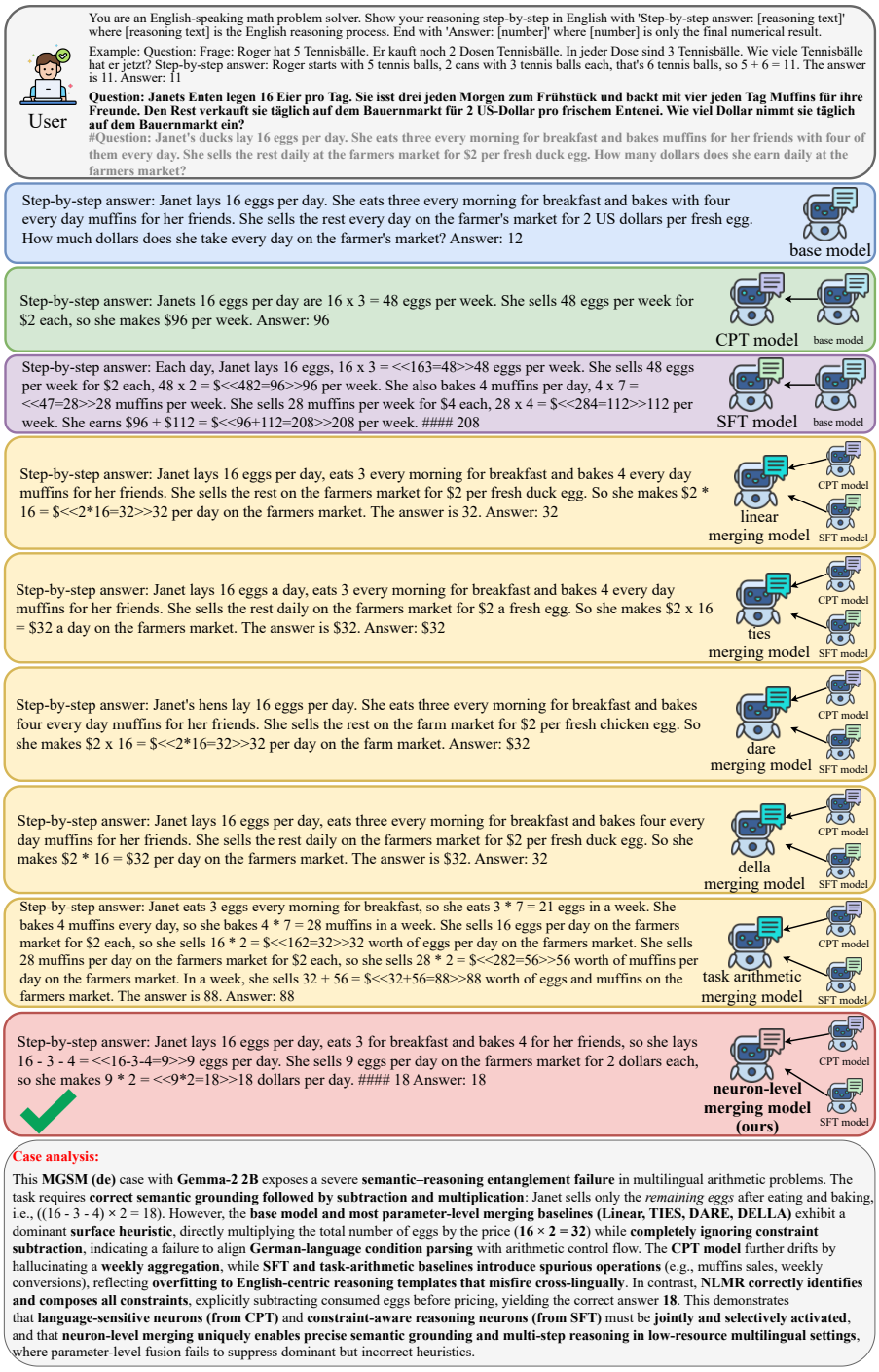
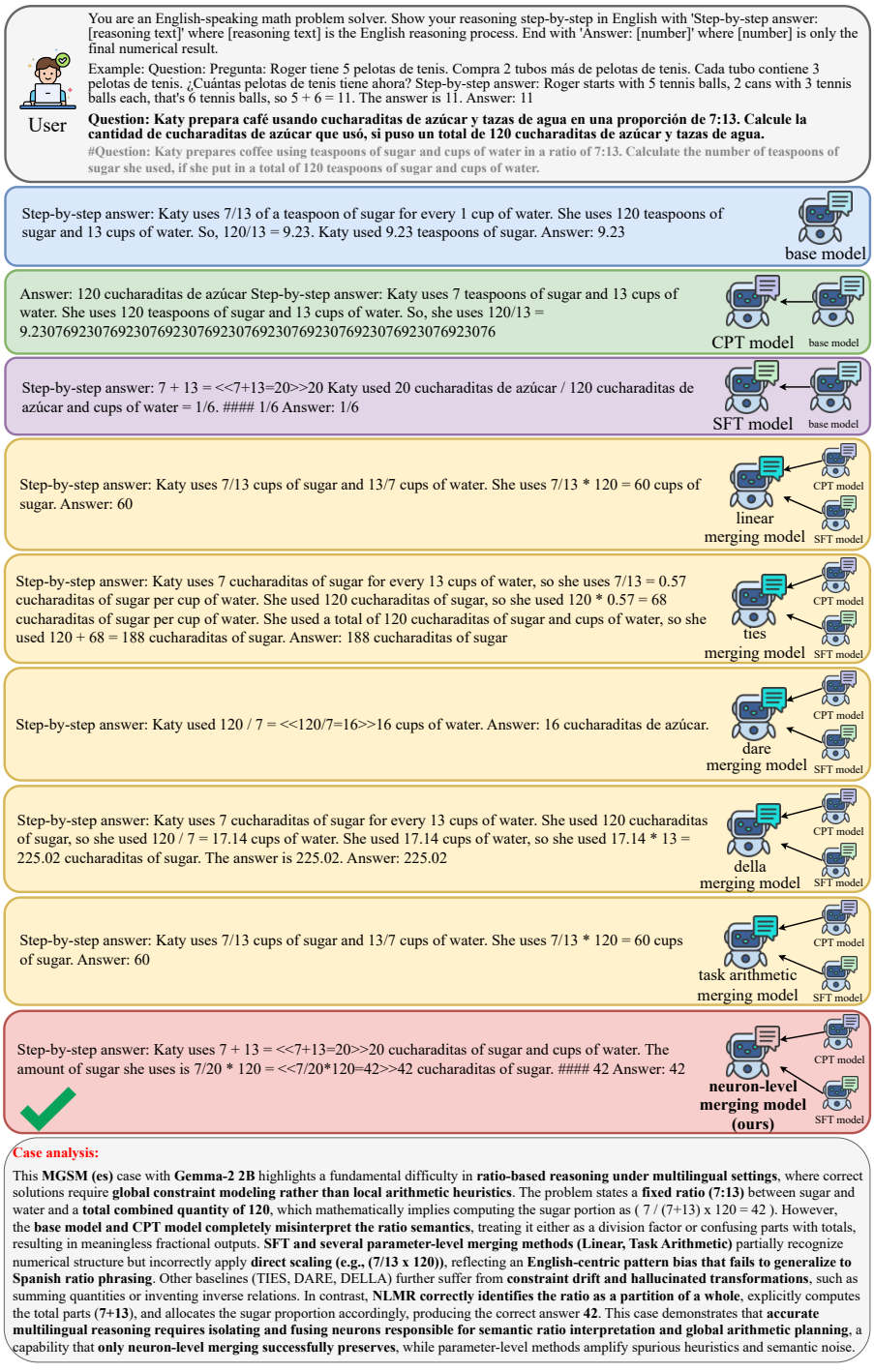


Figure 35: Case study on the MGSM dataset in German (de).


User
You are an English-speaking math problem solver. Show your reasoning step-by-step in English with 'Step-by-step answer: [reasoning text]' where [reasoning text] is the English reasoning process. End with 'Answer: [number]' where [number] is only the final numerical result.
Example: Question: Pregunta: Roger tiene 5 pelotas de tenis. Compra 2 tubos más de pelotas de tenis. Cada tubo contiene 3 pelotas de tenis. ¿Cuántas pelotas de tenis tiene ahora? Step-by-step answer: Roger starts with 5 tennis balls, 2 cans with 3 tennis balls each, that's 6 tennis balls, so 5 + 6 = 11. The answer is 11. Answer: 11
Question: Katy prepara café usando cucharaditas de azúcar y tazas de agua en una proporción de 7:13. Calcule la cantidad de cucharaditas de azúcar que usó, si puso un total de 120 cucharaditas de azúcar y tazas de agua.
#Question: Katy prepares coffee using teaspoons of sugar and cups of water in a ratio of 7:13. Calculate the number of teaspoons of sugar she used, if she put in a total of 120 teaspoons of sugar and cups of water.
Step-by-step answer: Katy uses 7/13 of a teaspoon of sugar for every 1 cup of water. She uses 120 teaspoons of sugar and 13 cups of water. So, 120/13 = 9.23. Katy used 9.23 teaspoons of sugar. Answer: 9.23
base model
Answer: 120 cucharaditas de azúcar Step-by-step answer: Katy uses 7 teaspoons of sugar and 13 cups of water. She uses 120 teaspoons of sugar and 13 cups of water. So, she uses 120/13 = 9.230769230769230769

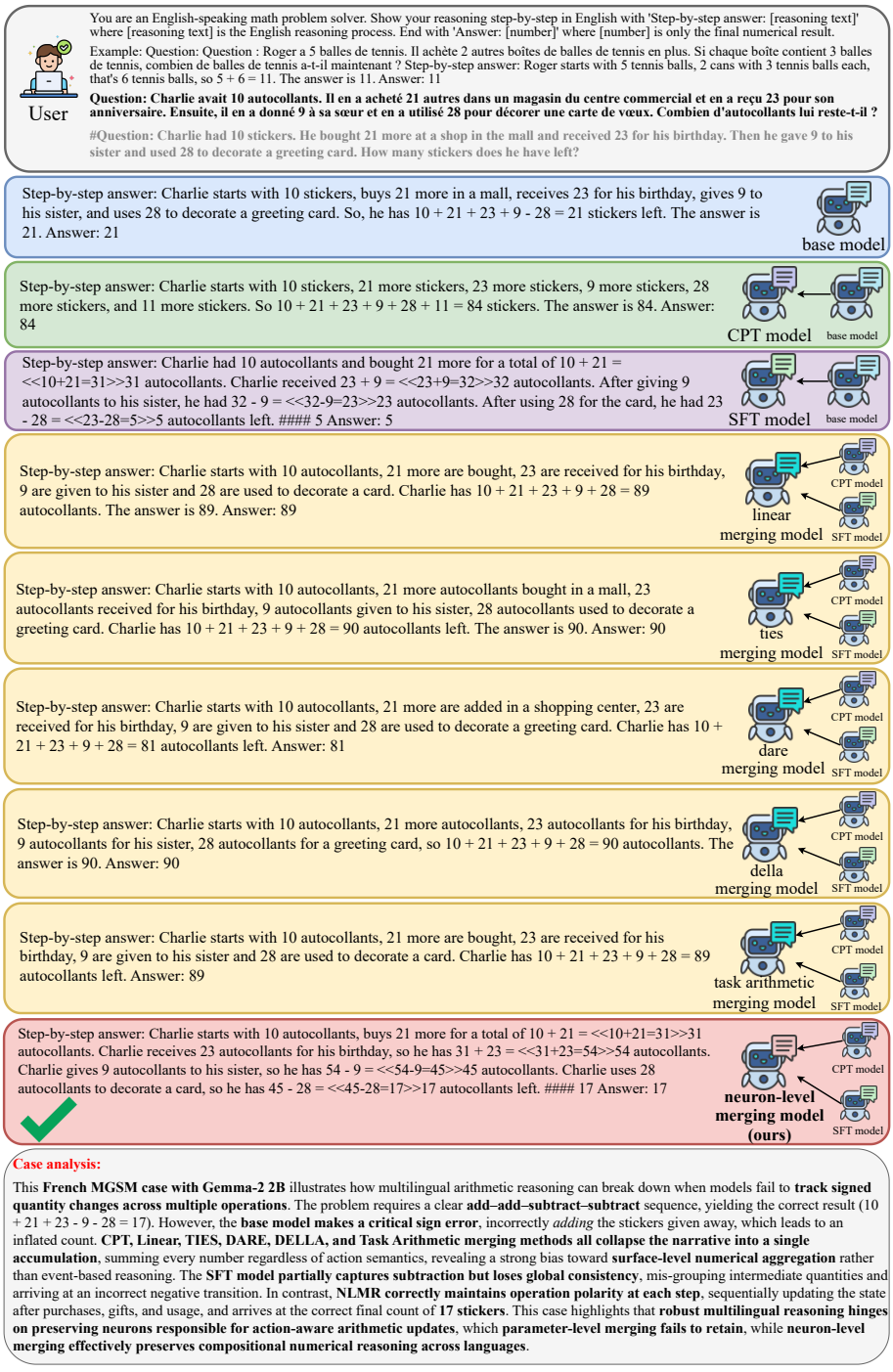


Figure 37: Case study on the MGSM dataset in French (fr).

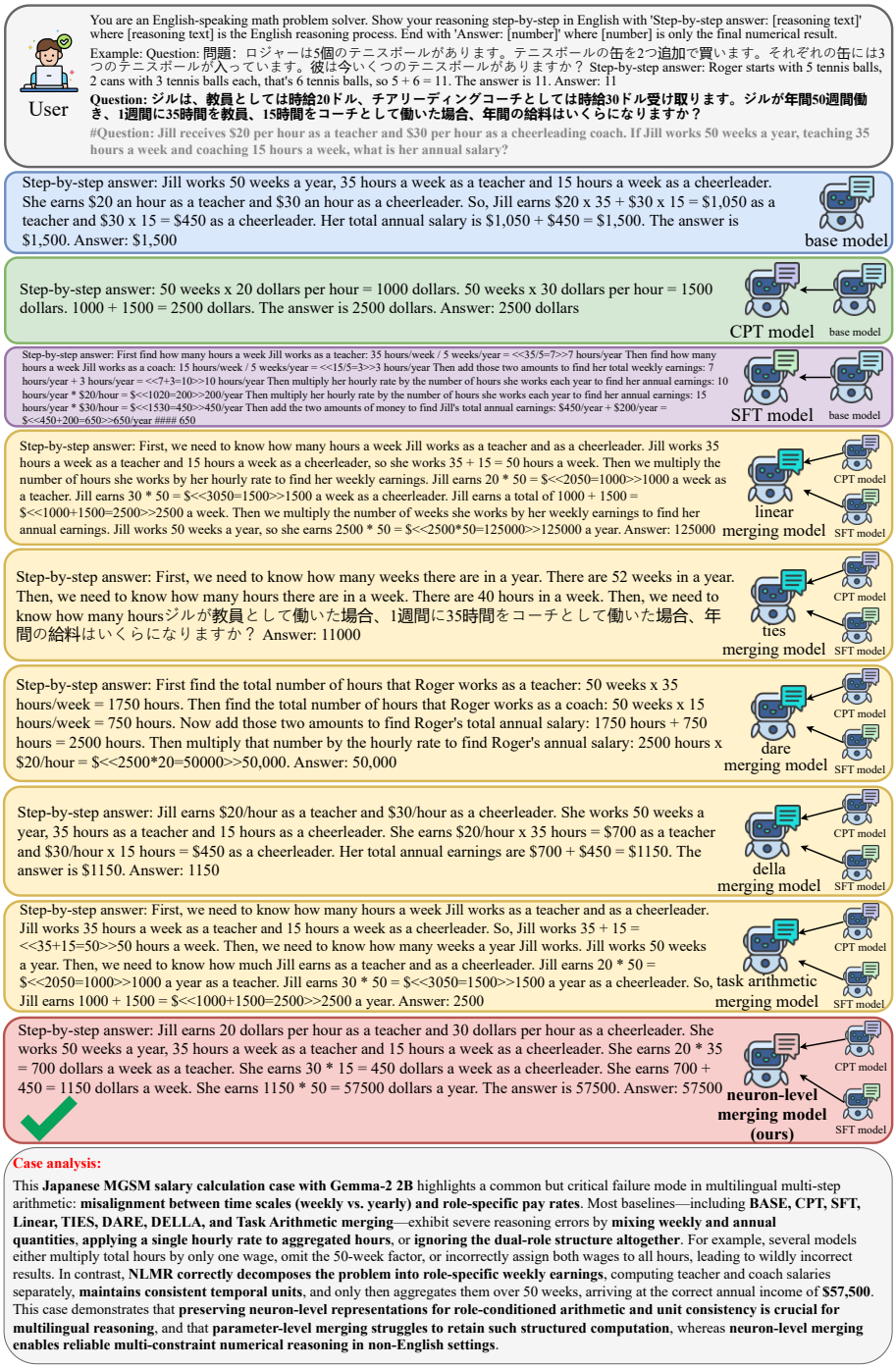


Figure 38: Case study on the MGSM dataset in Japanese (ja).

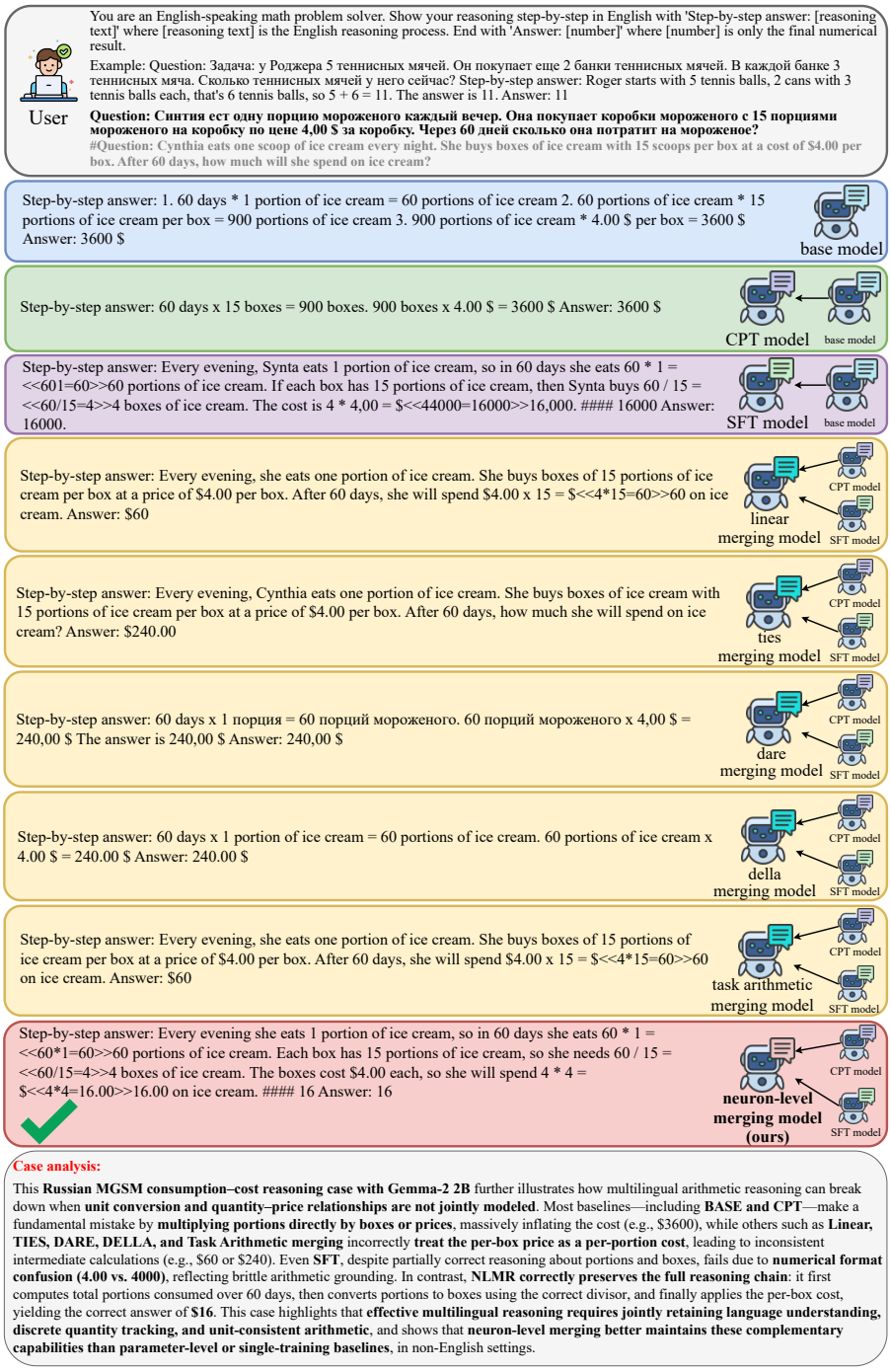


Figure 39: Case study on the MGSM dataset in Russian (ru).

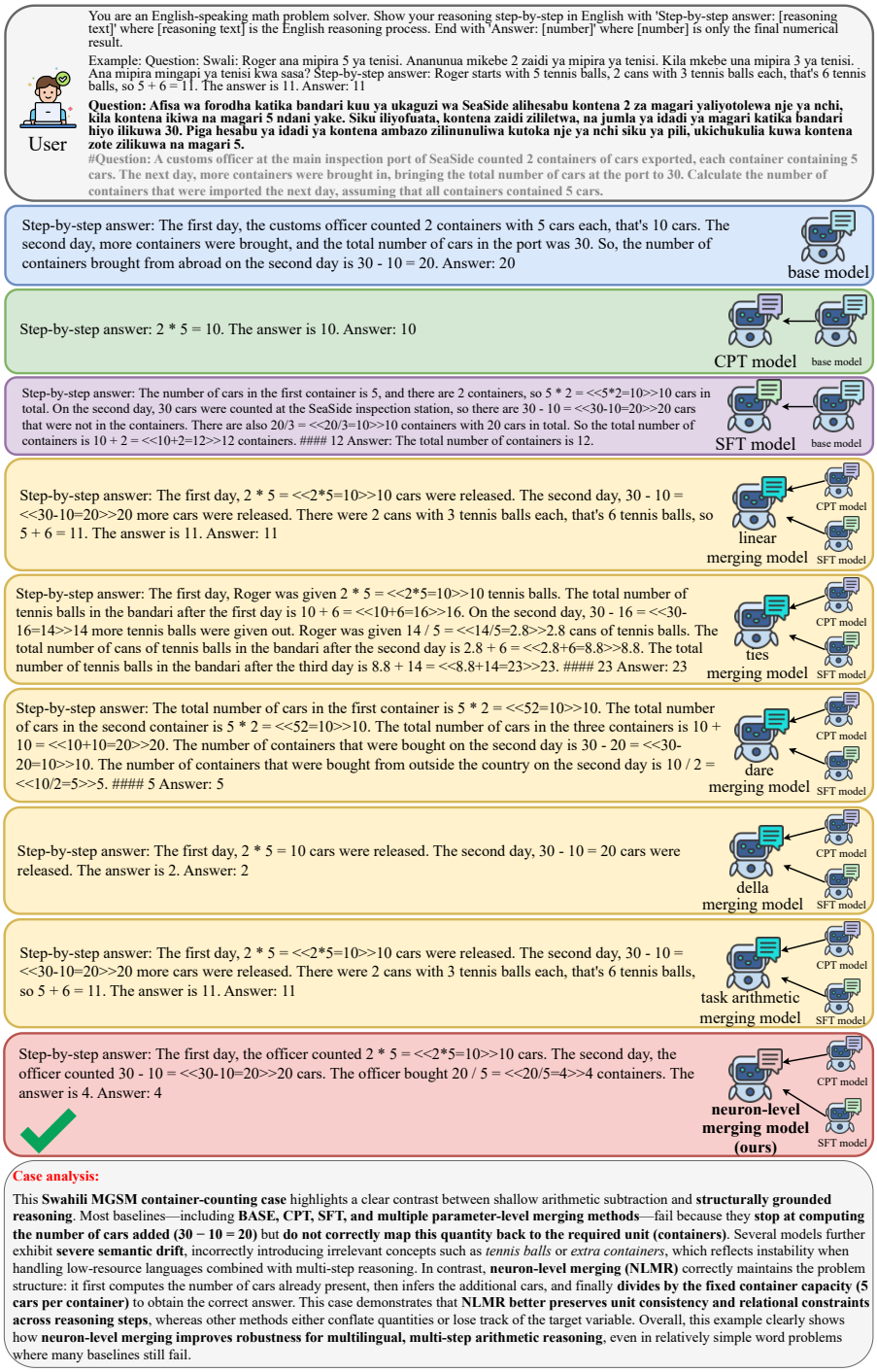


Figure 40: Case study on the MGSM dataset in Swahili (sw).

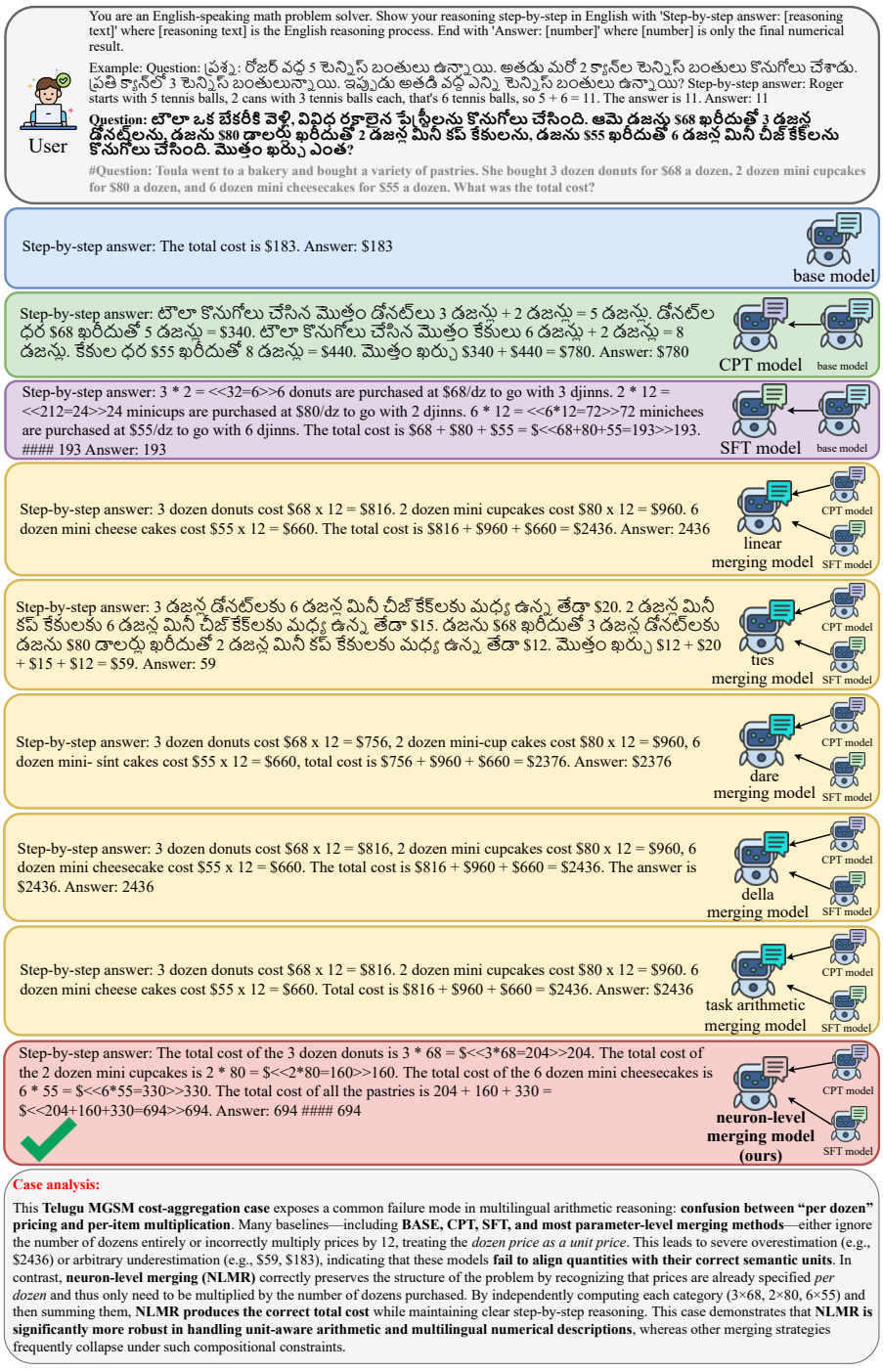


Figure 41: Case study on the MGSM dataset in Telugu (te).

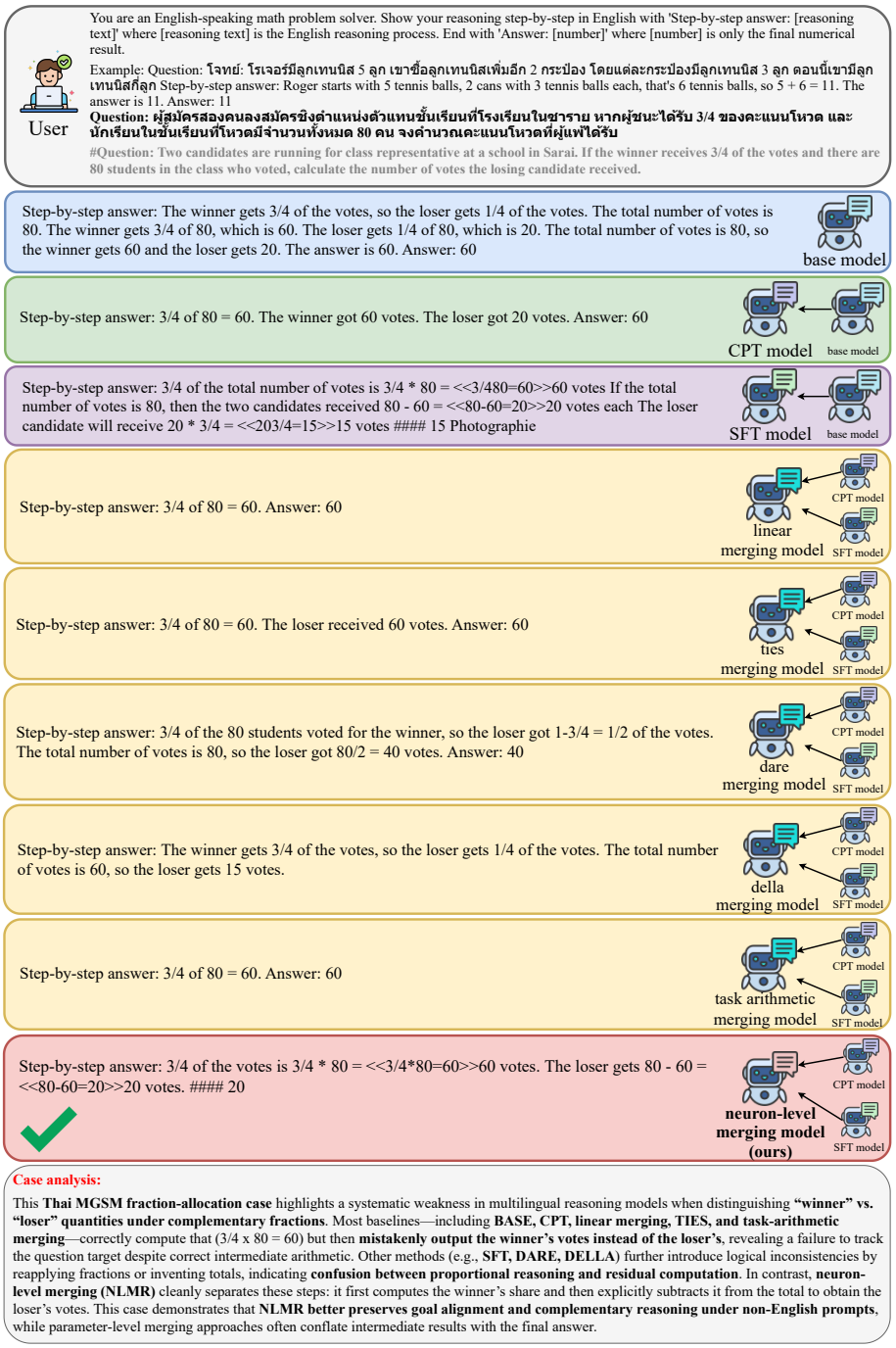


Figure 42: Case study on the MGSM dataset in Thai (th).

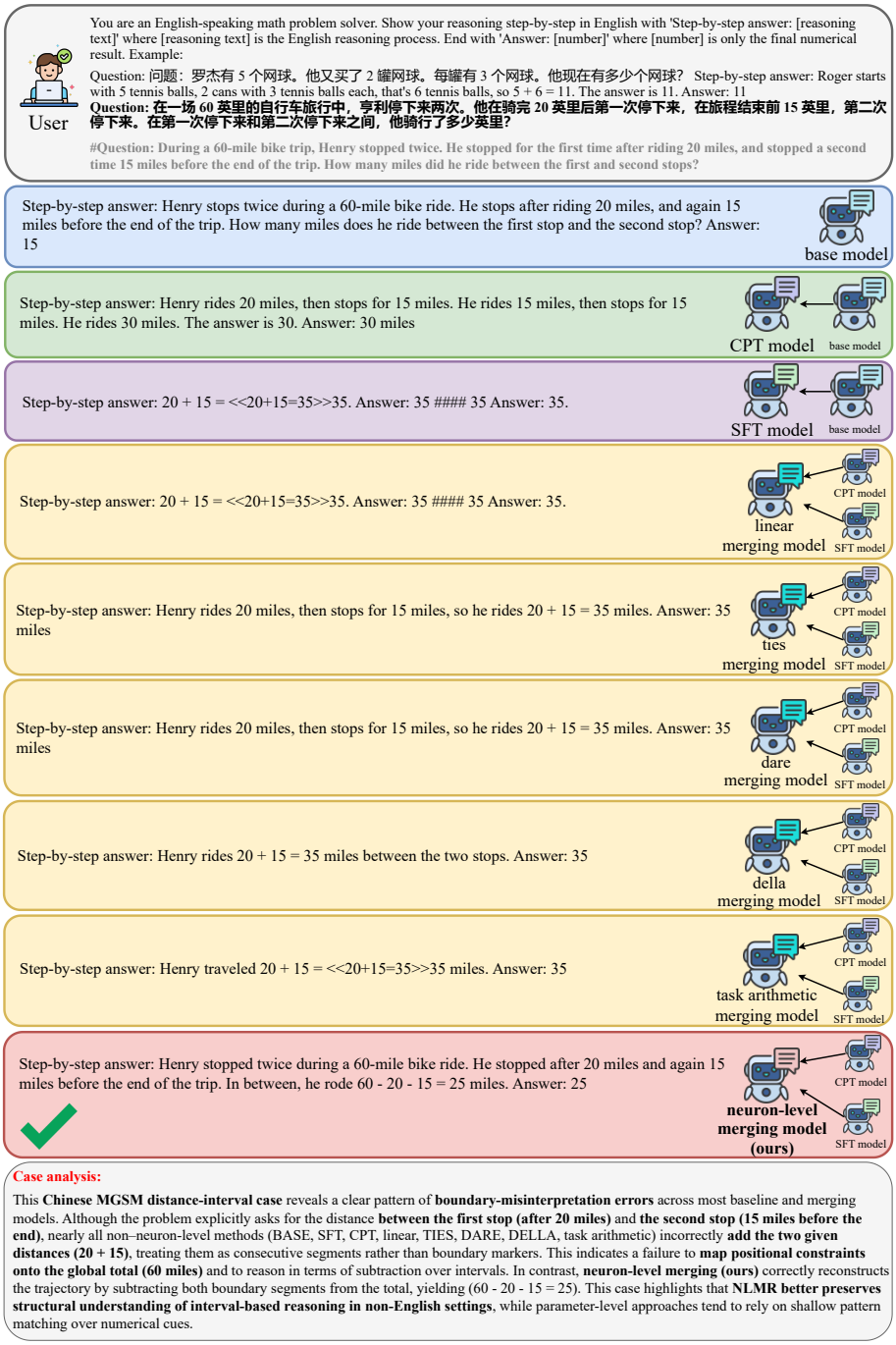


Figure 43: Case study on the MGSM dataset in Chinese (zh).

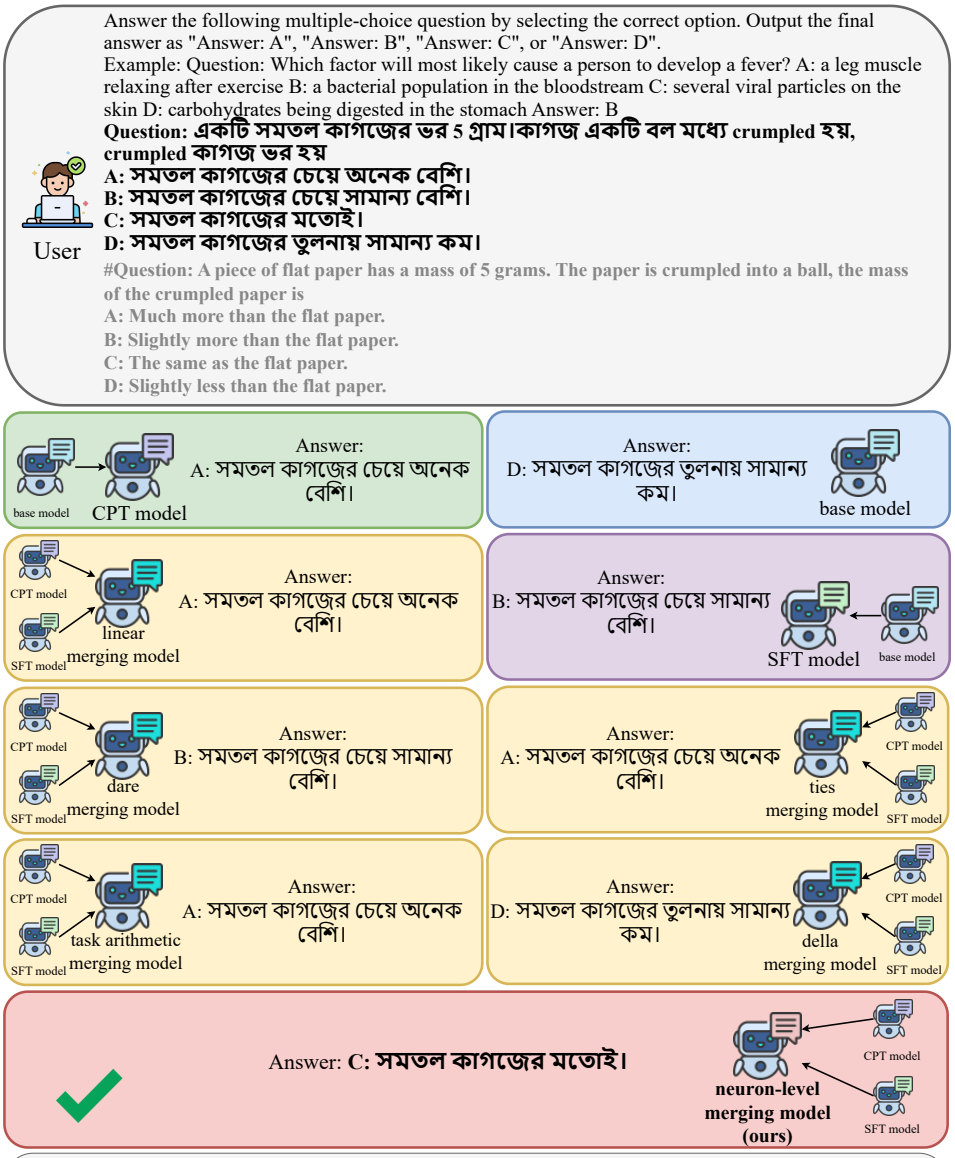


Figure 44: Case study on the ARC-Easy dataset in Bengali (bn).

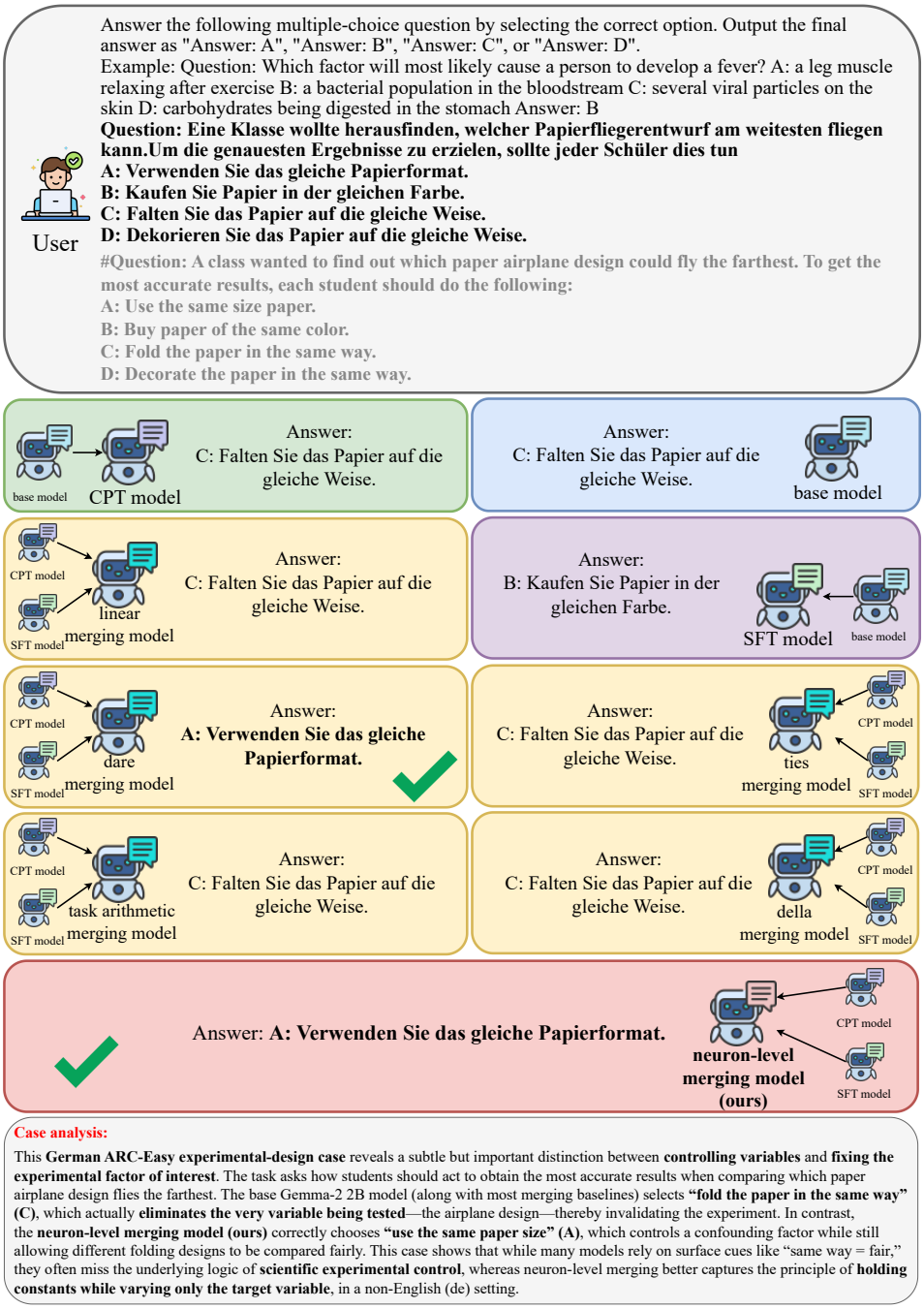


Figure 45: Case study on the ARC-Easy dataset in German (de).

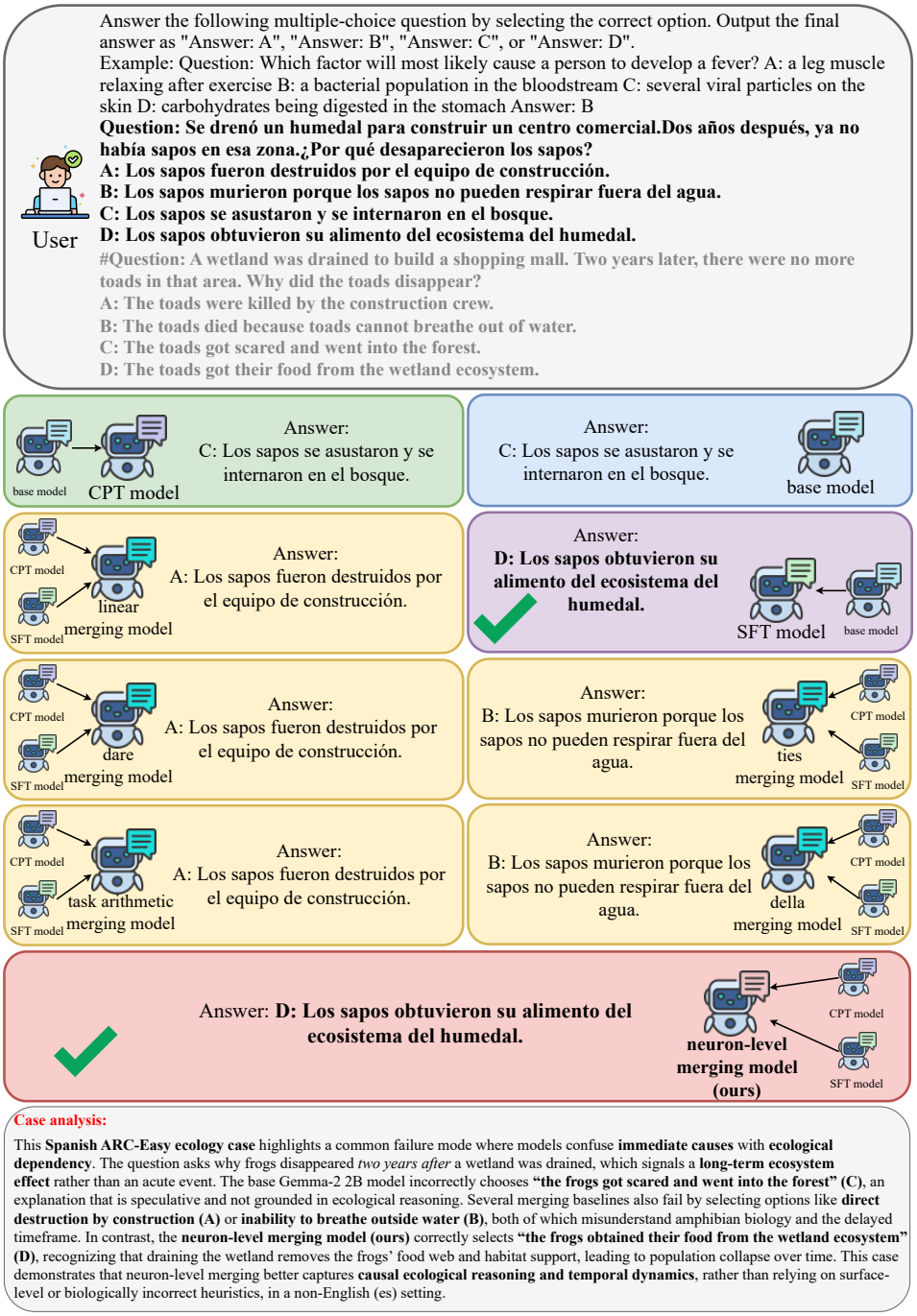


Figure 46: Case study on the ARC-Easy dataset in Spanish (es).

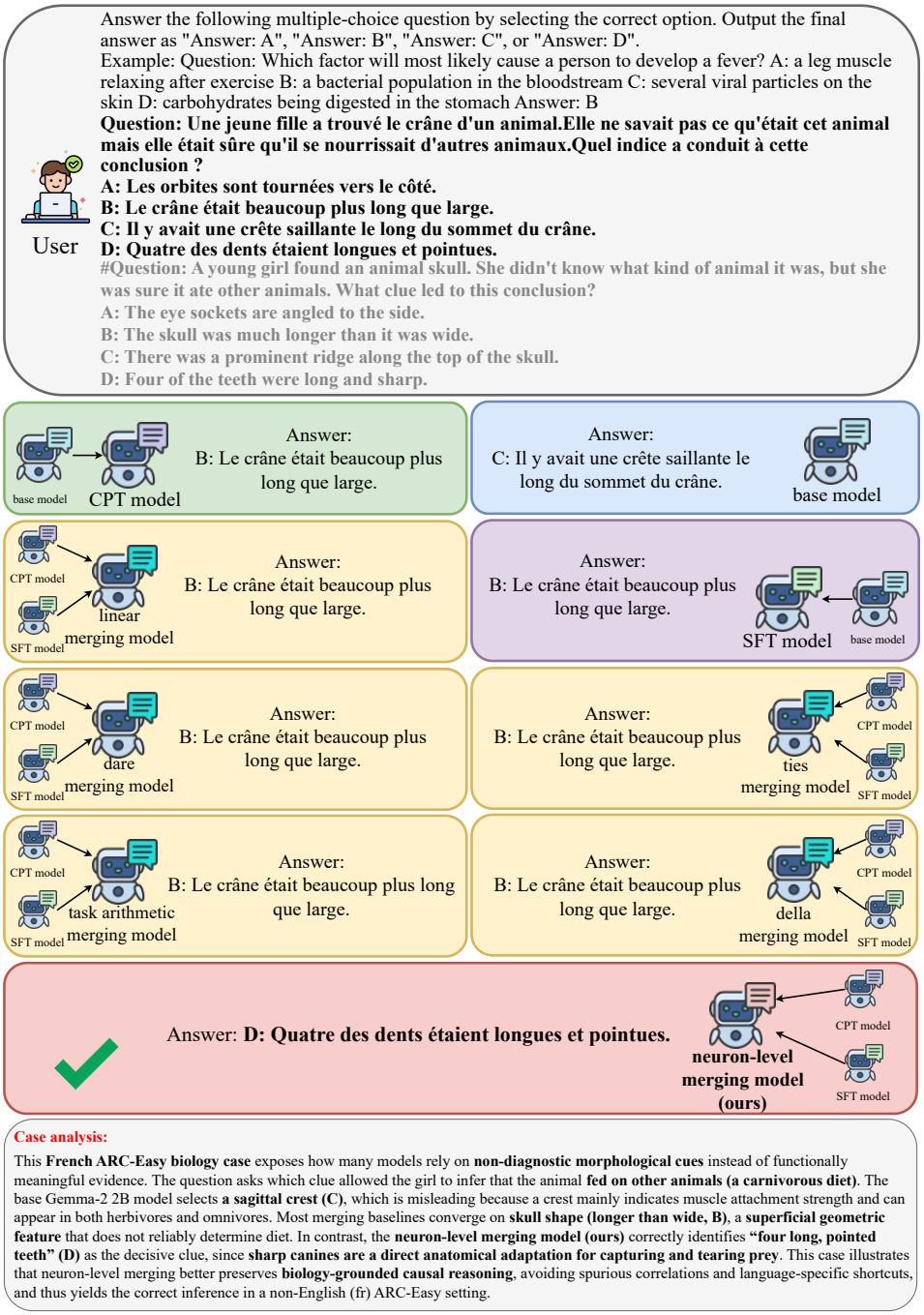


Figure 47: Case study on the ARC-Easy dataset in French (fr).

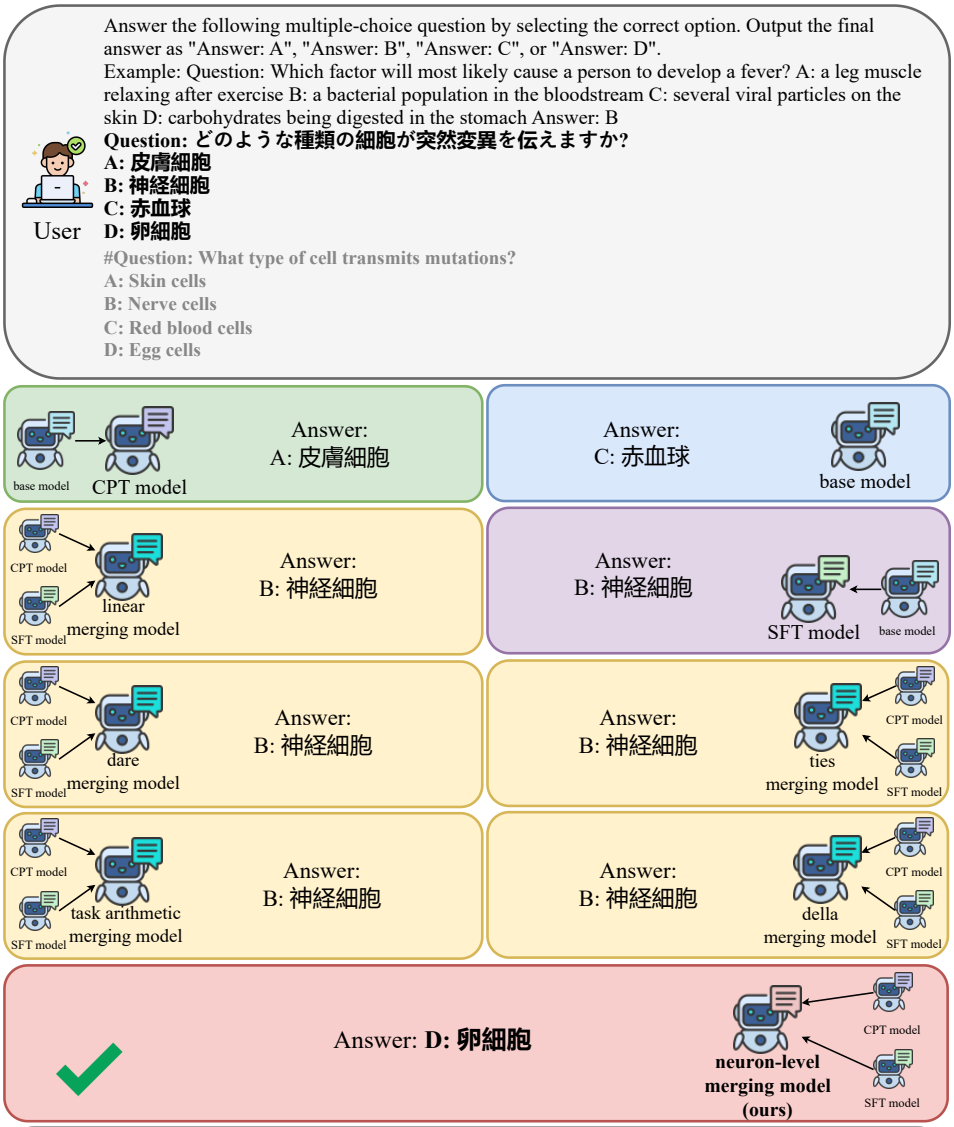


Figure 48: Case study on the ARC-Easy dataset in Japanese (ja).

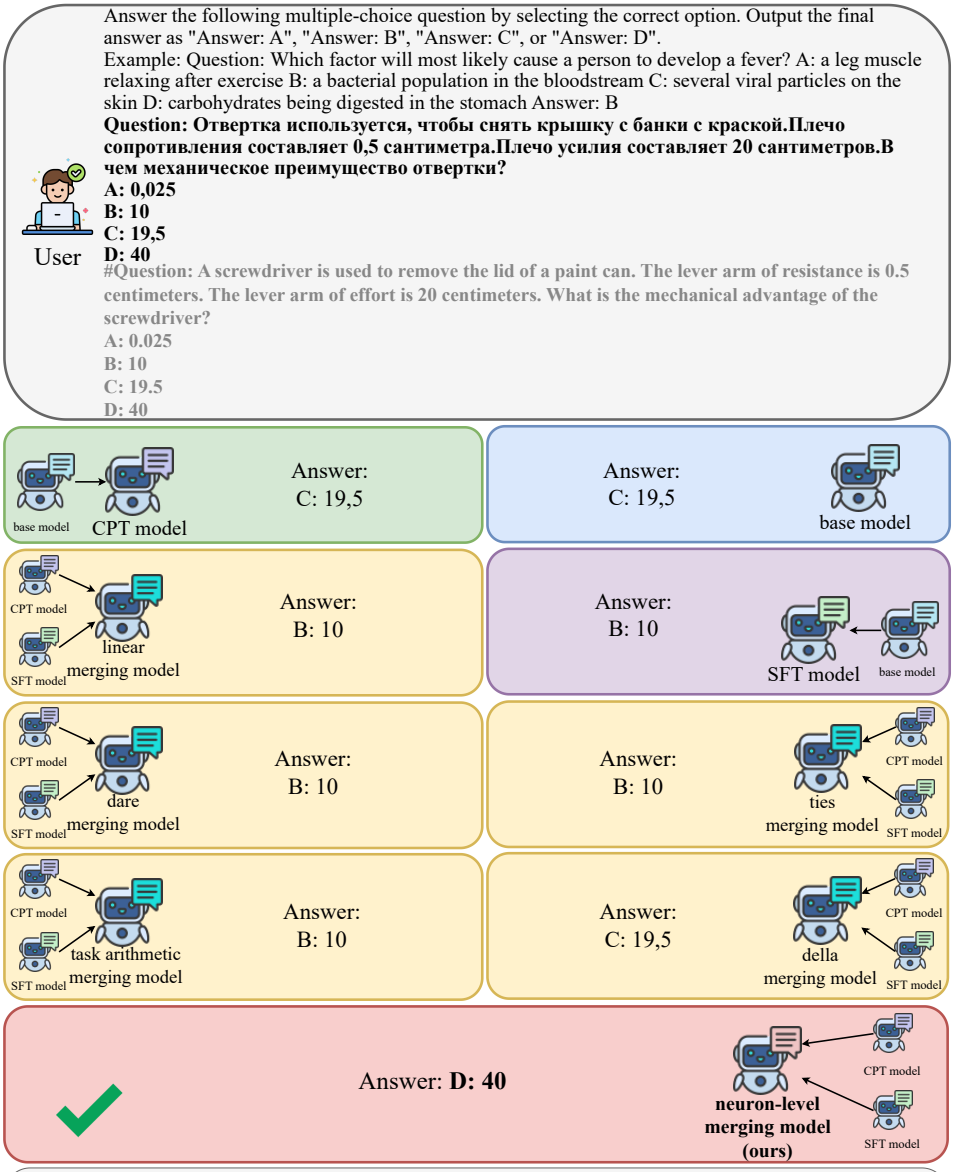


Figure 49: Case study on the ARC-Easy dataset in Russian (ru).

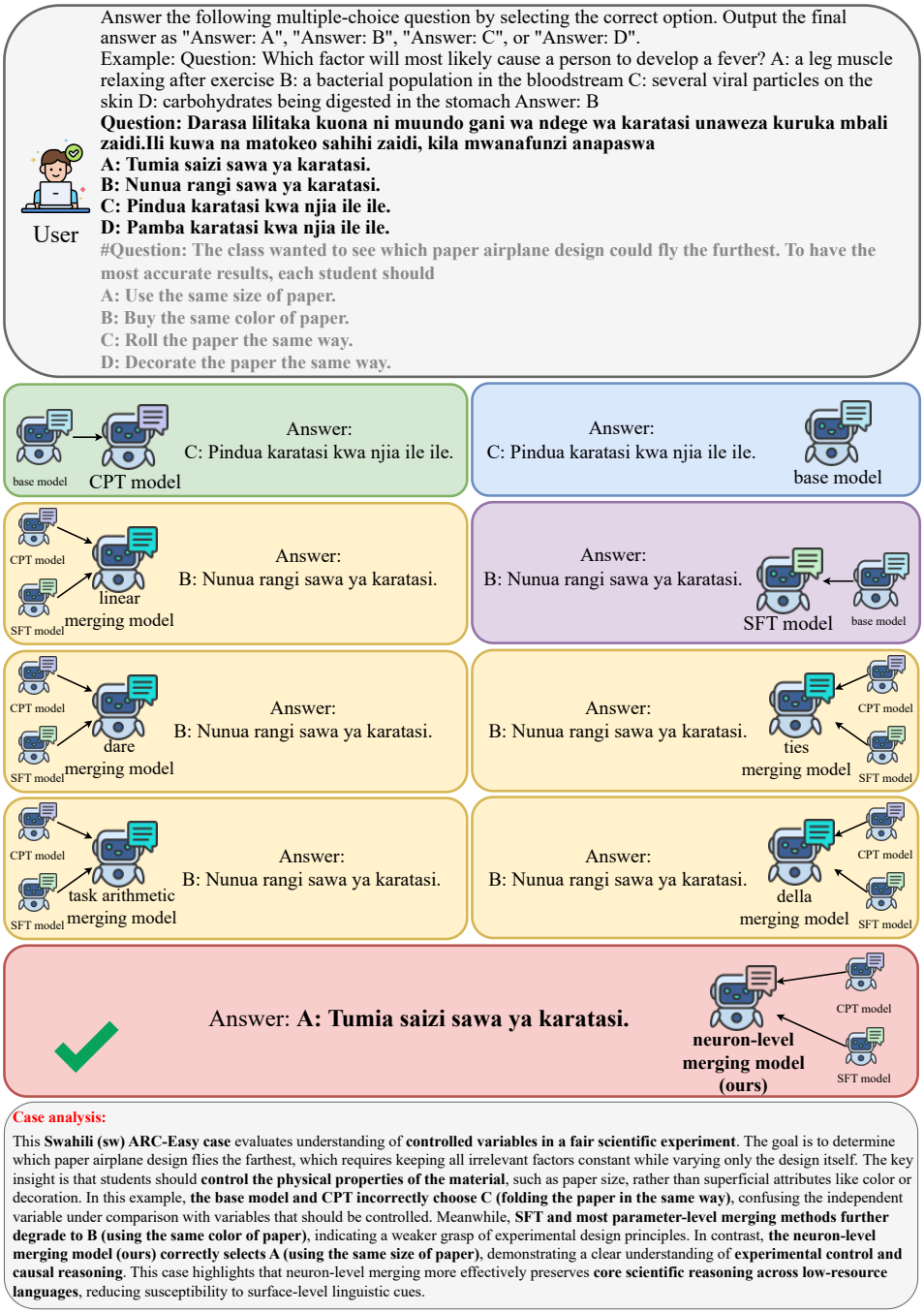


Figure 50: Case study on the ARC-Easy dataset in Swahili (sw).

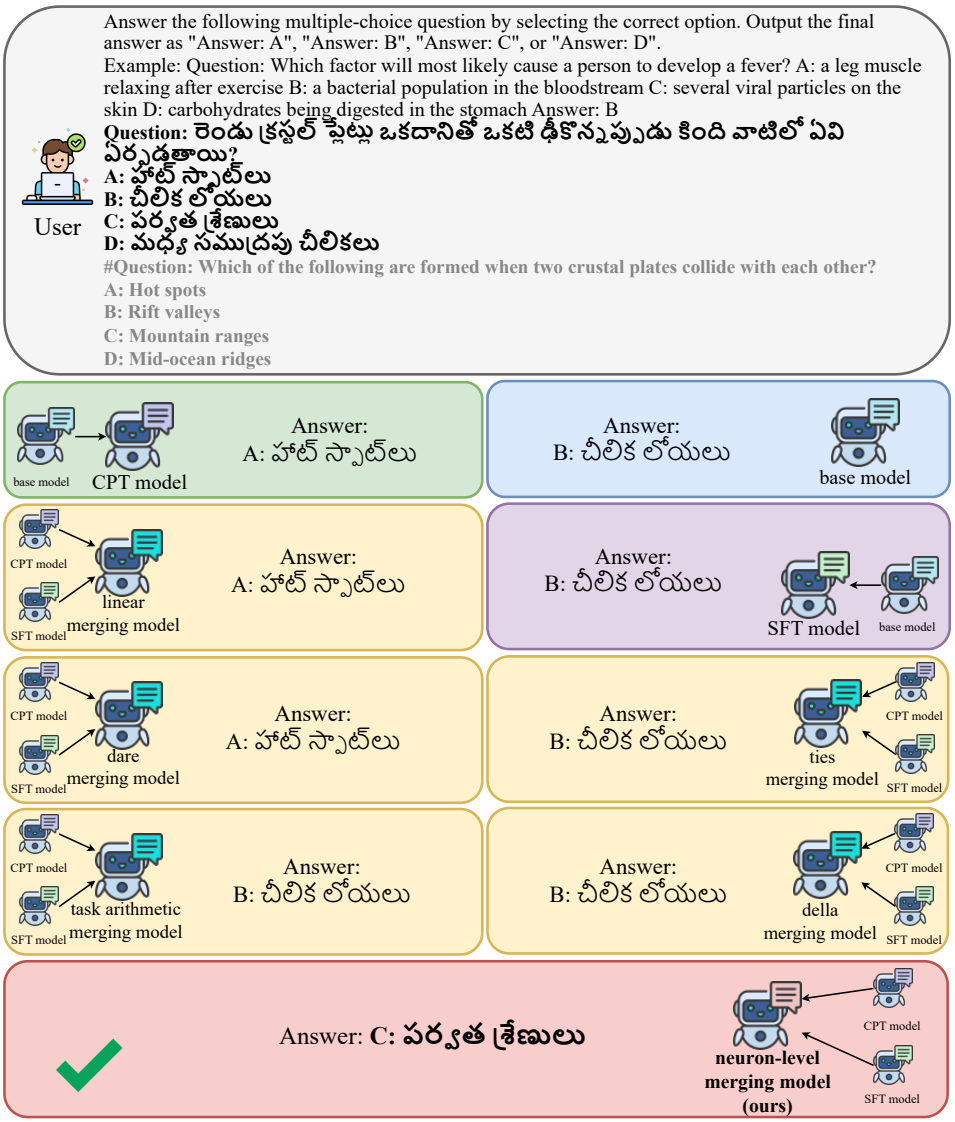


Figure 51: Case study on the ARC-Easy dataset in Telugu (te).

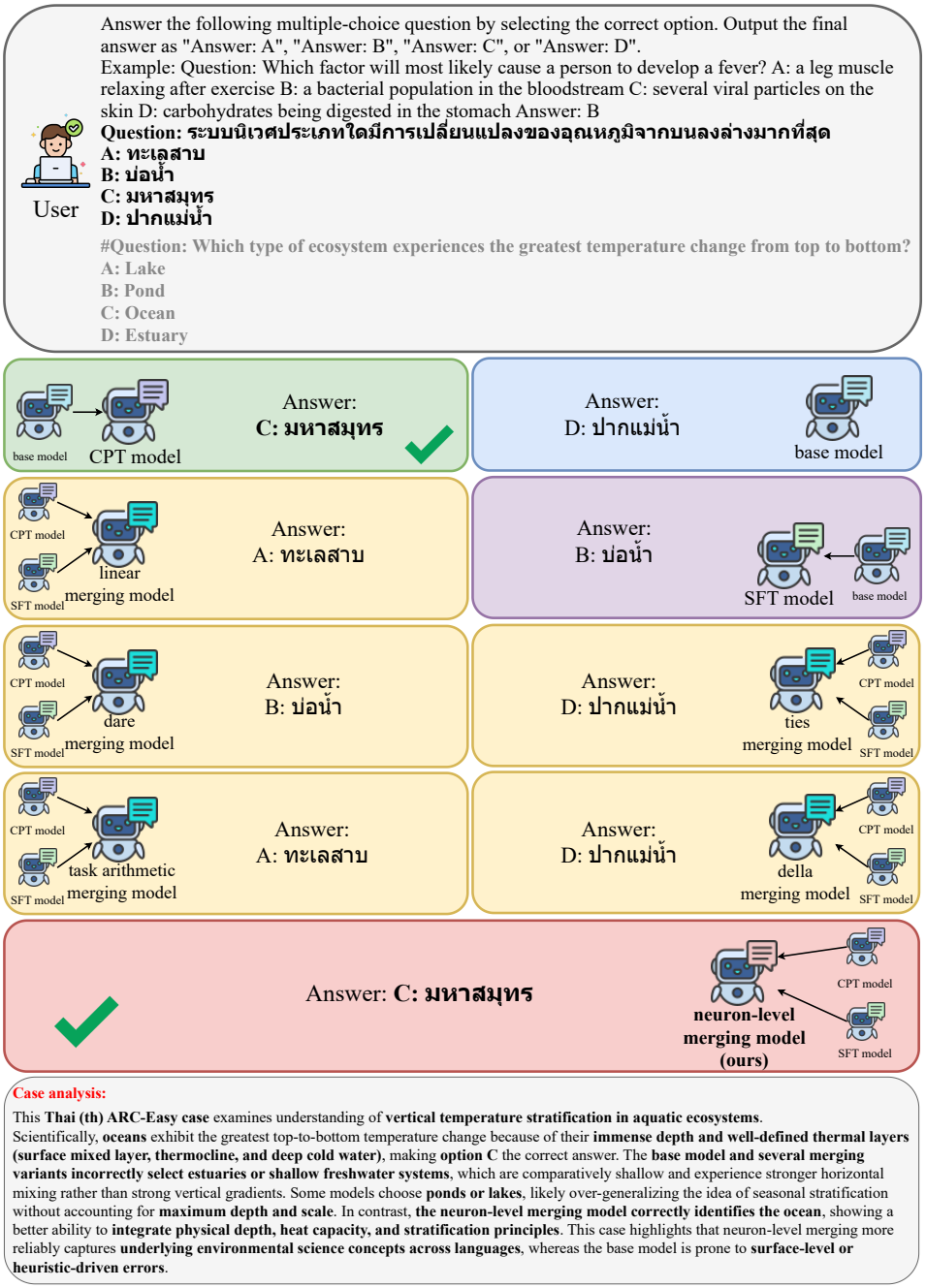


Figure 52: Case study on the ARC-Easy dataset in Thai (th).

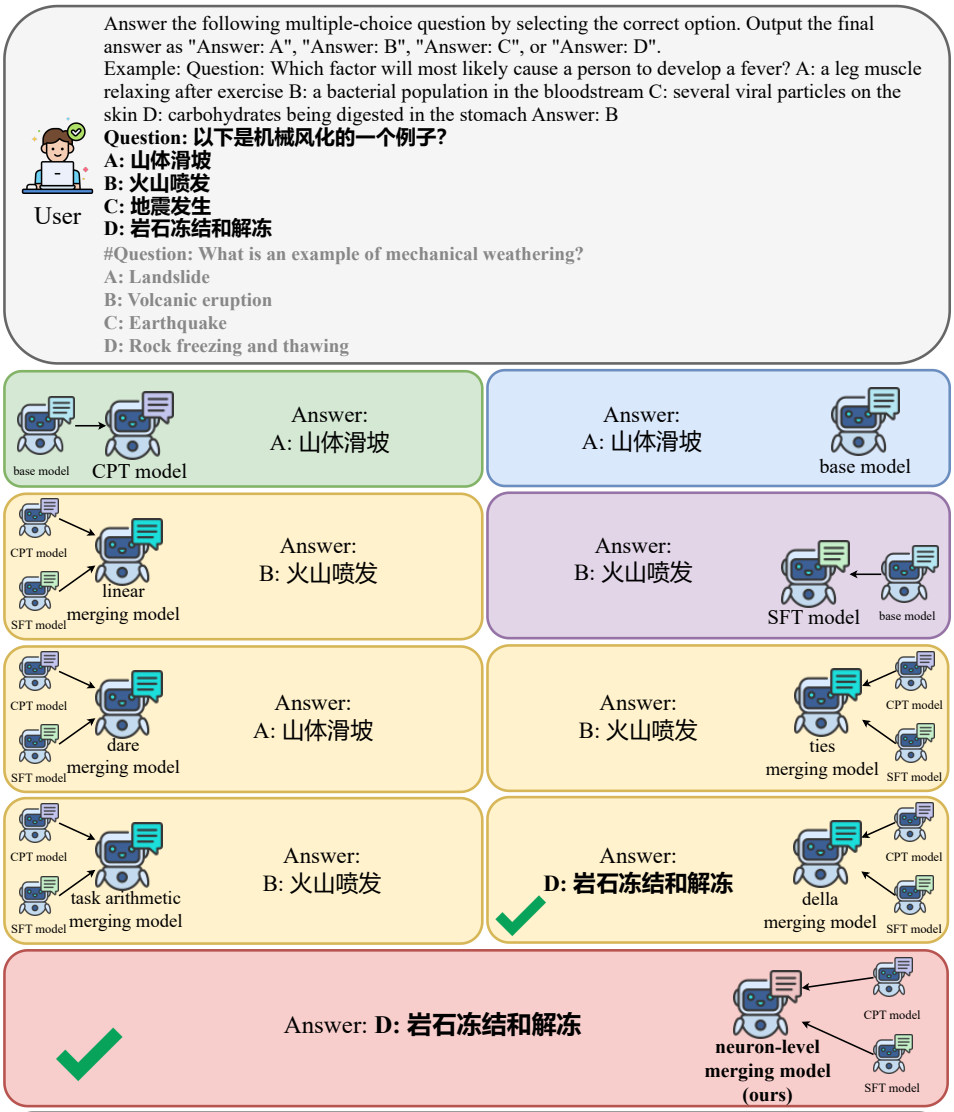


Figure 53: Case study on the ARC-Easy dataset in Chinese (zh).

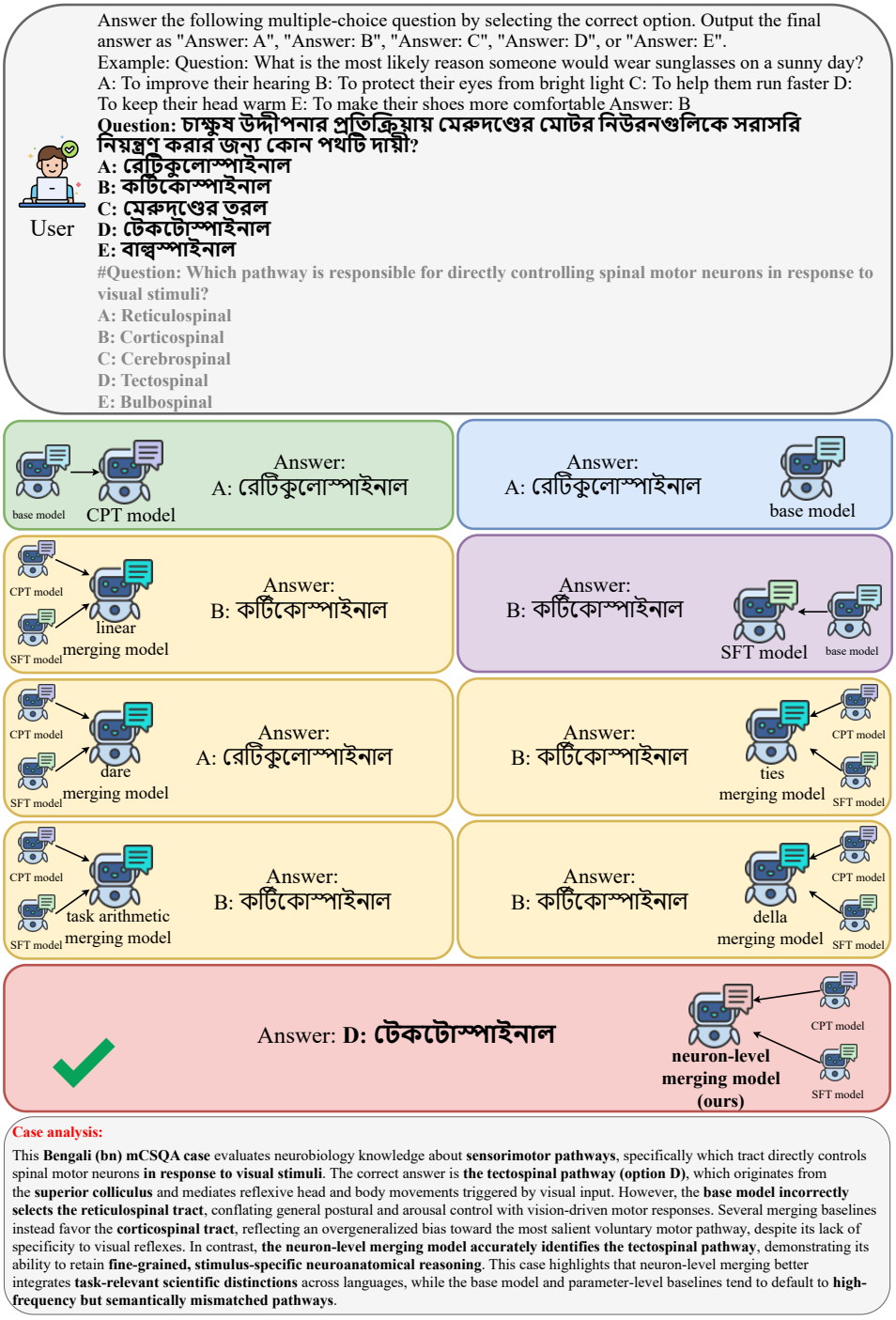


Figure 54: Case study on the mCSQA dataset in Bengali (bn).

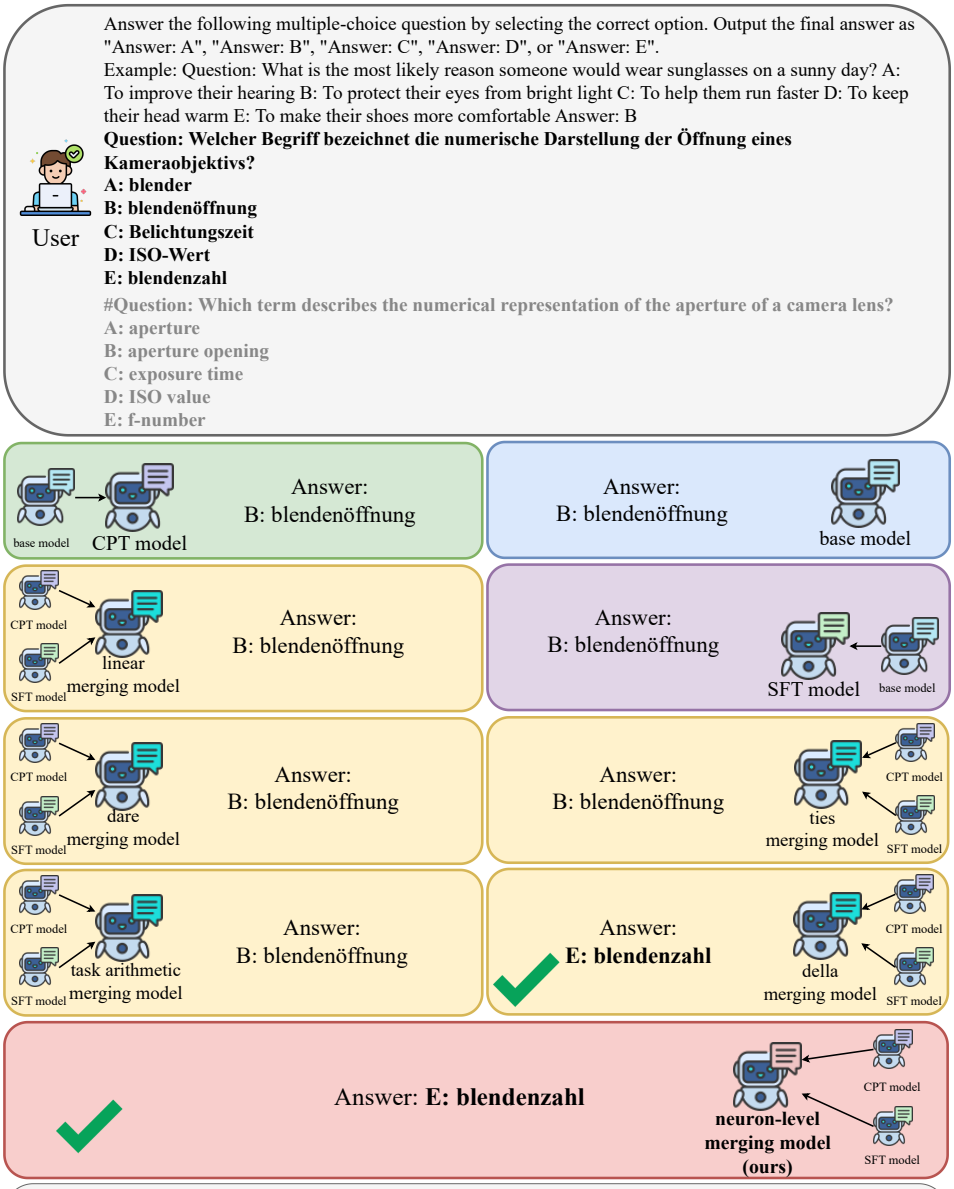


Figure 55: Case study on the mCSQA dataset in German (de).

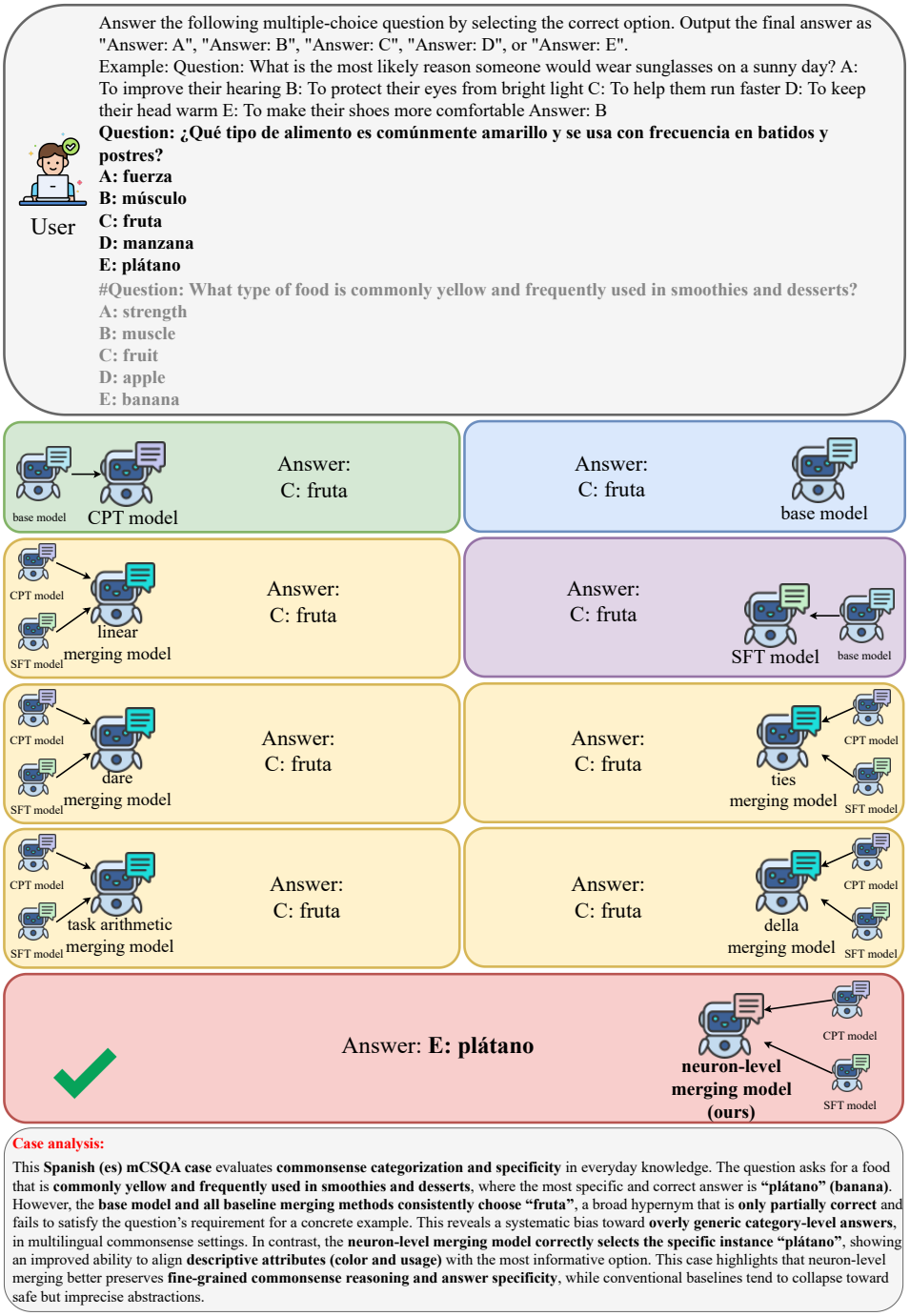


Figure 56: Case study on the mCSQA dataset in Spanish (es).

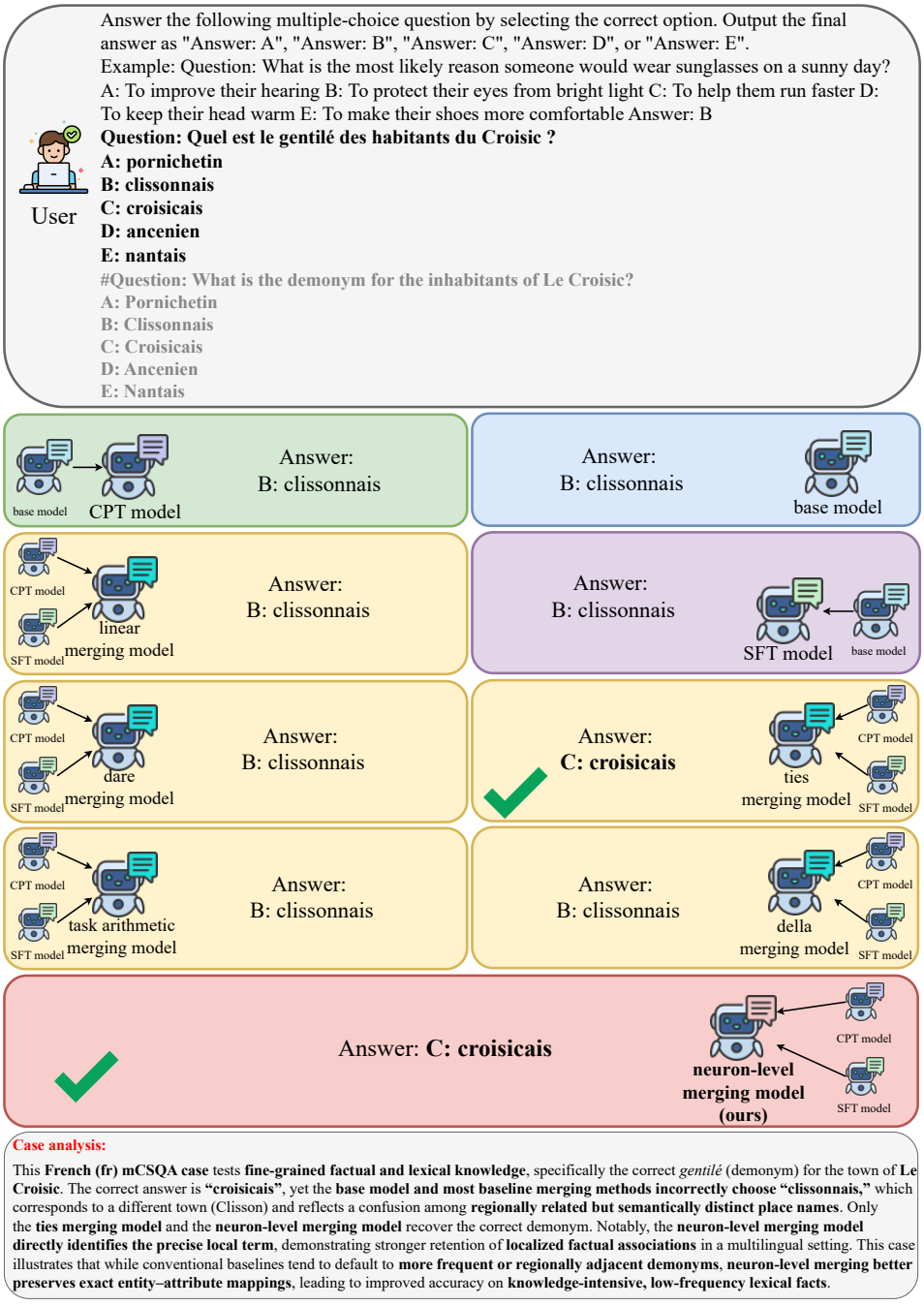
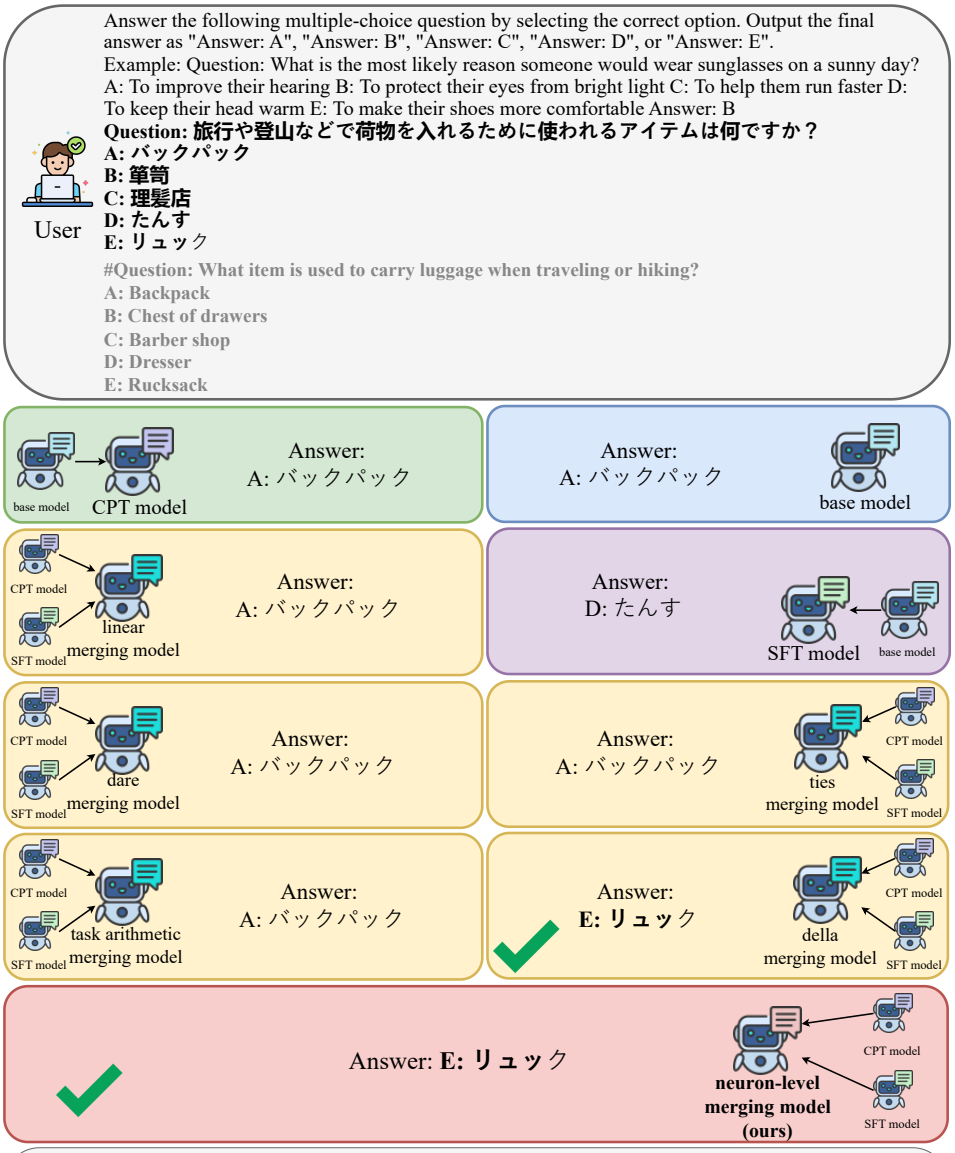


Figure 57: Case study on the mCSQA dataset in French (fr).



Case analysis:

This case from **mCSQA (Japanese)** highlights a subtle but important distinction between **surface lexical overlap** and **contextually appropriate reasoning**. The question asks which item is used to carry luggage for travel or hiking. While many baselines (BASE, CPT, Linear, TIES, DARE, Task Arithmetic) answer **「バックパック」**, and the SFT model even makes a clear error by selecting **「たんす」** (a chest of drawers), these models largely rely on **direct lexical association or English-biased mappings**. In contrast, our neuron-level merging model (NLMR) correctly predicts **「リュック」**, which is the **most natural and commonly used Japanese term** for a backpack in everyday contexts such as travel and hiking. Notably, **「バックパック」 and 「リュック」 are semantically related**, but 「リュック」 is more pragmatically appropriate in Japanese usage. This demonstrates that **NLMR better integrates language-specific CPT knowledge with task reasoning from SFT**, enabling it to capture **fine-grained, culturally grounded lexical preferences** rather than defaulting to English-influenced transliterations. This case illustrates how neuron-level selective merging improves **context-aware multilingual reasoning**, even when multiple candidate answers are superficially correct.

Figure 58: Case study on the mCSQA dataset in Japanese (ja).

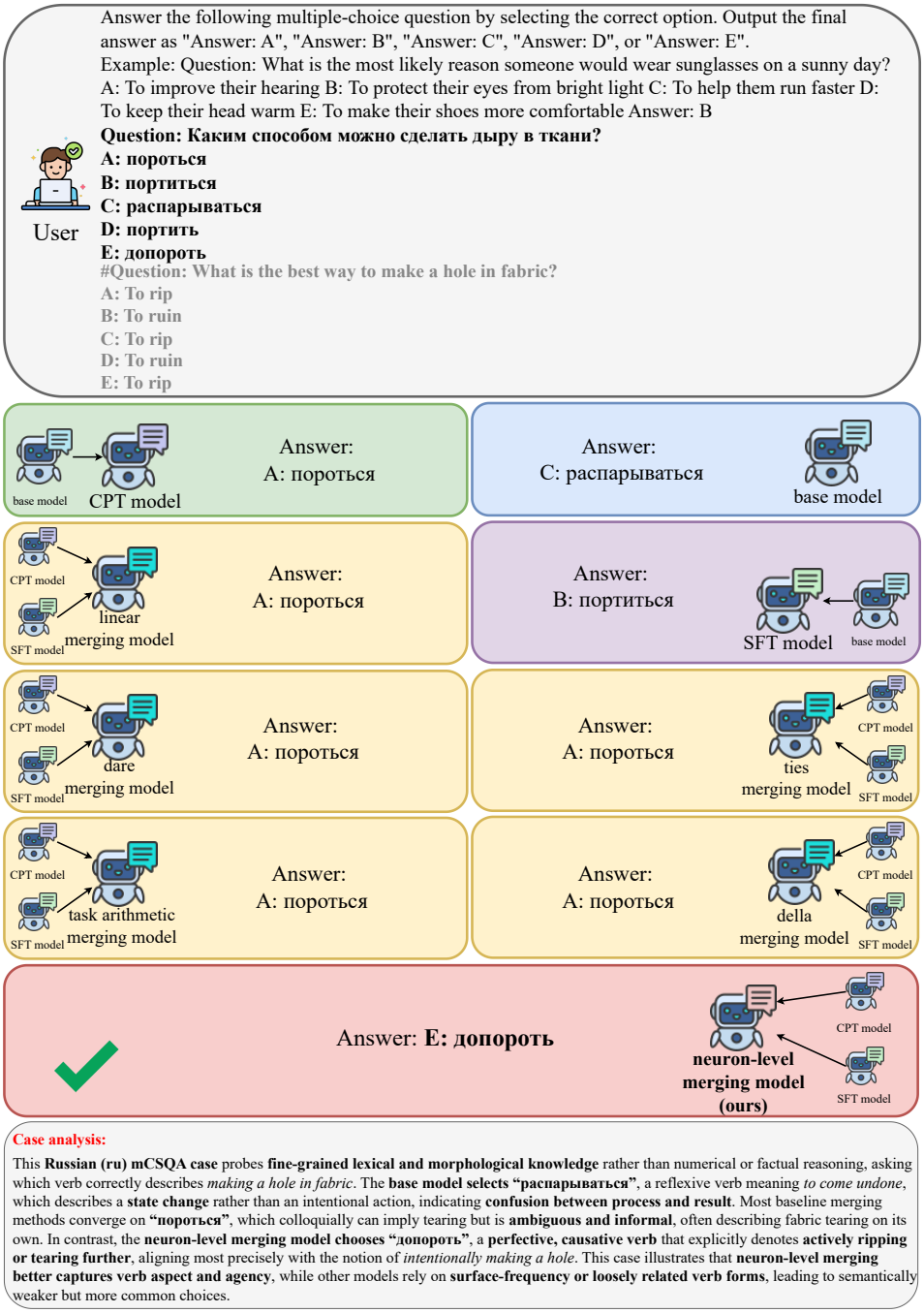


Figure 59: Case study on the mCSQA dataset in Russian (ru).

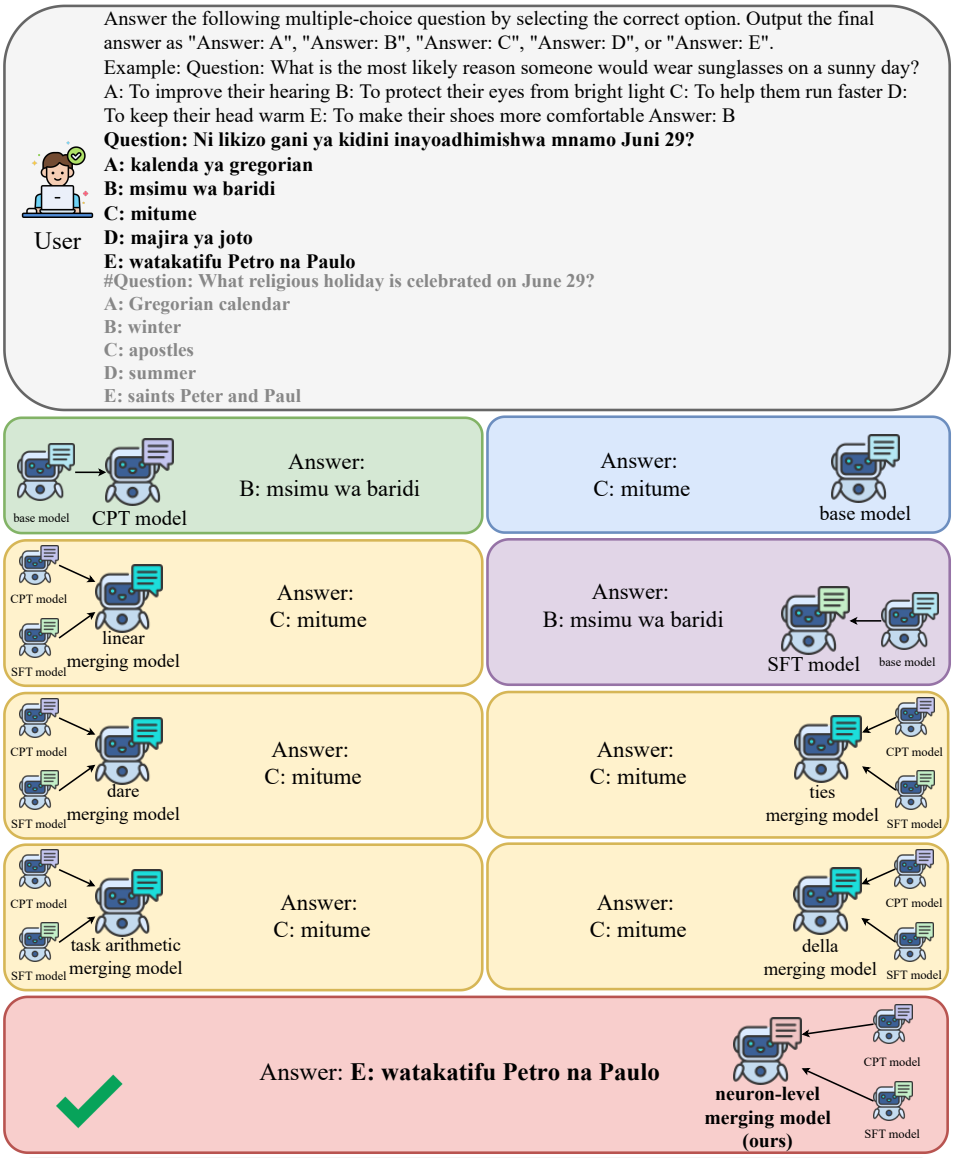


Figure 60: Case study on the mCSQA dataset in Swahili (sw).

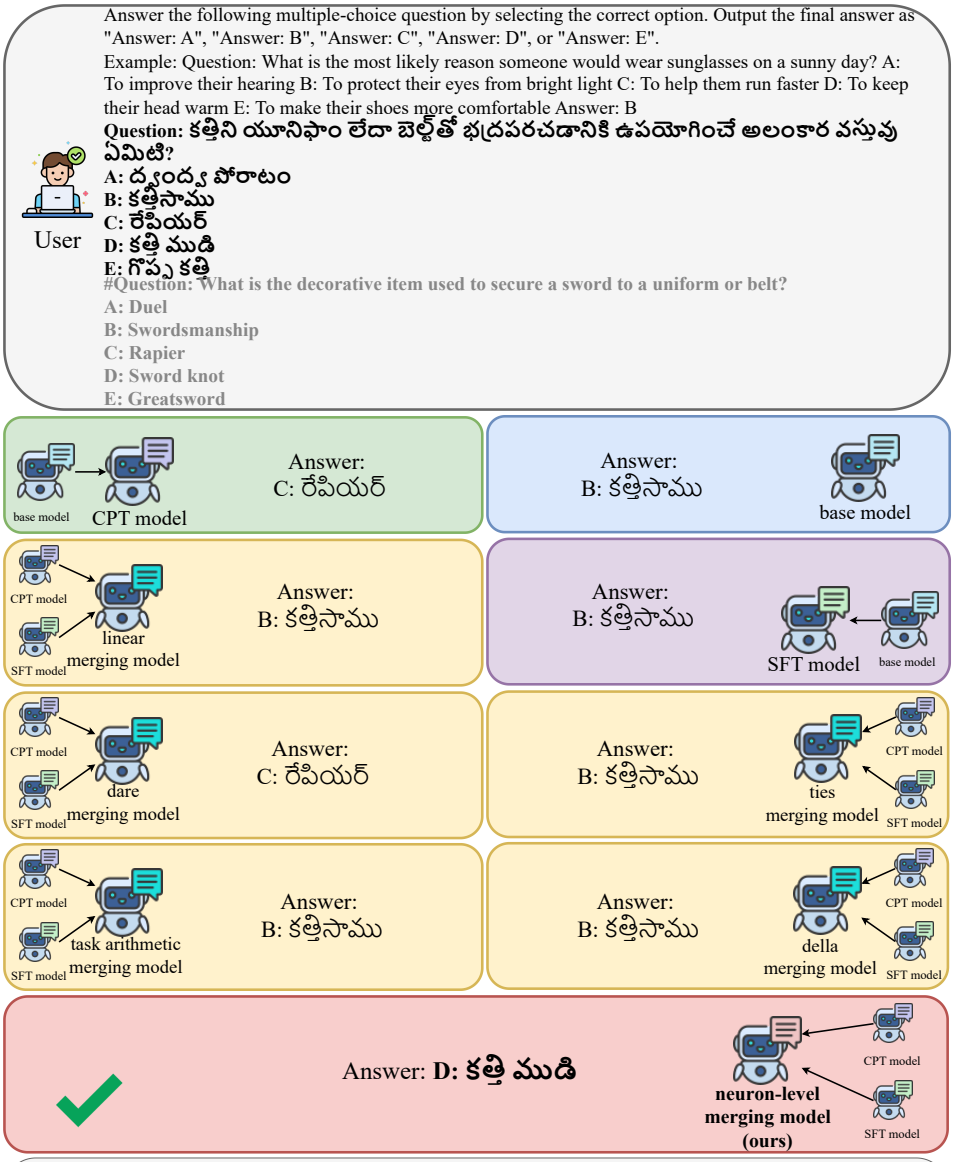
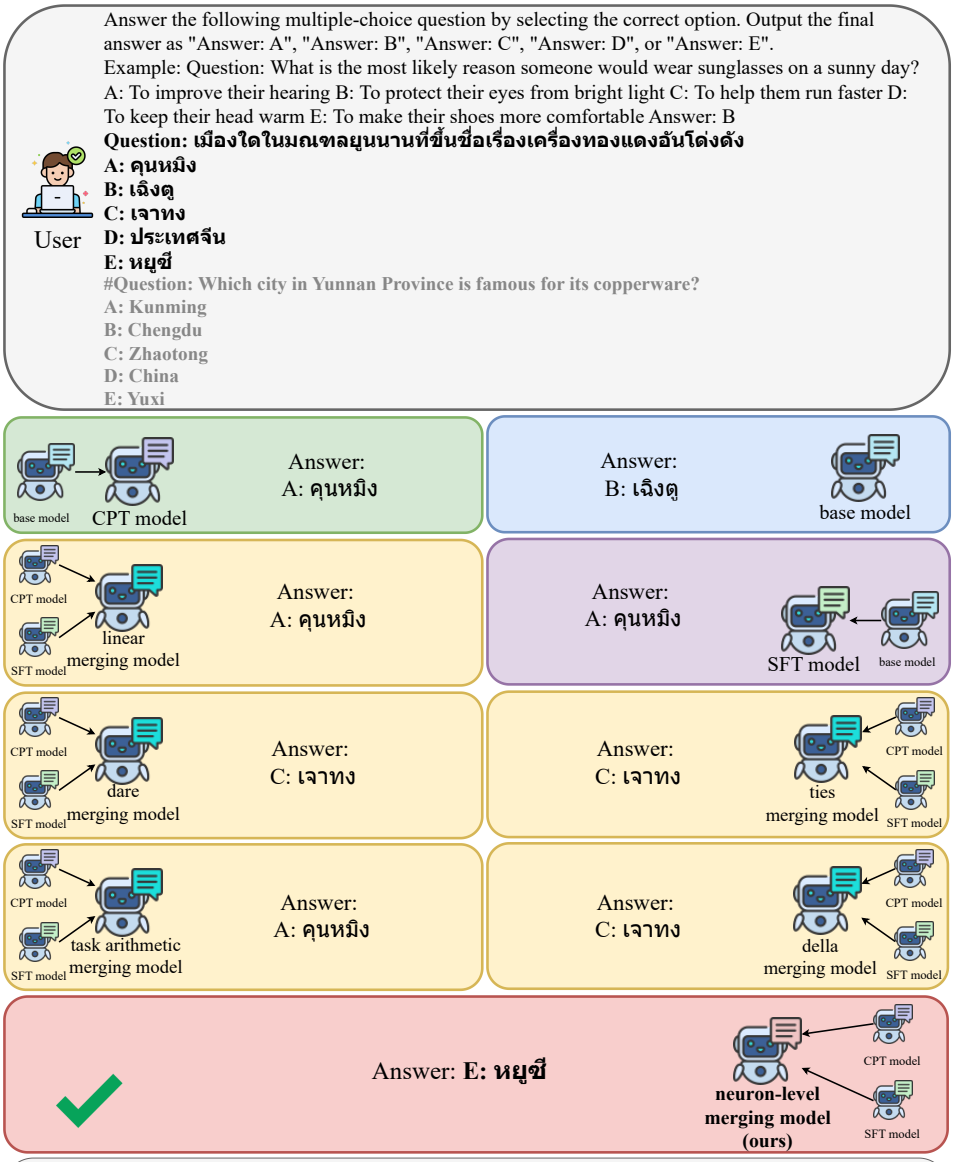


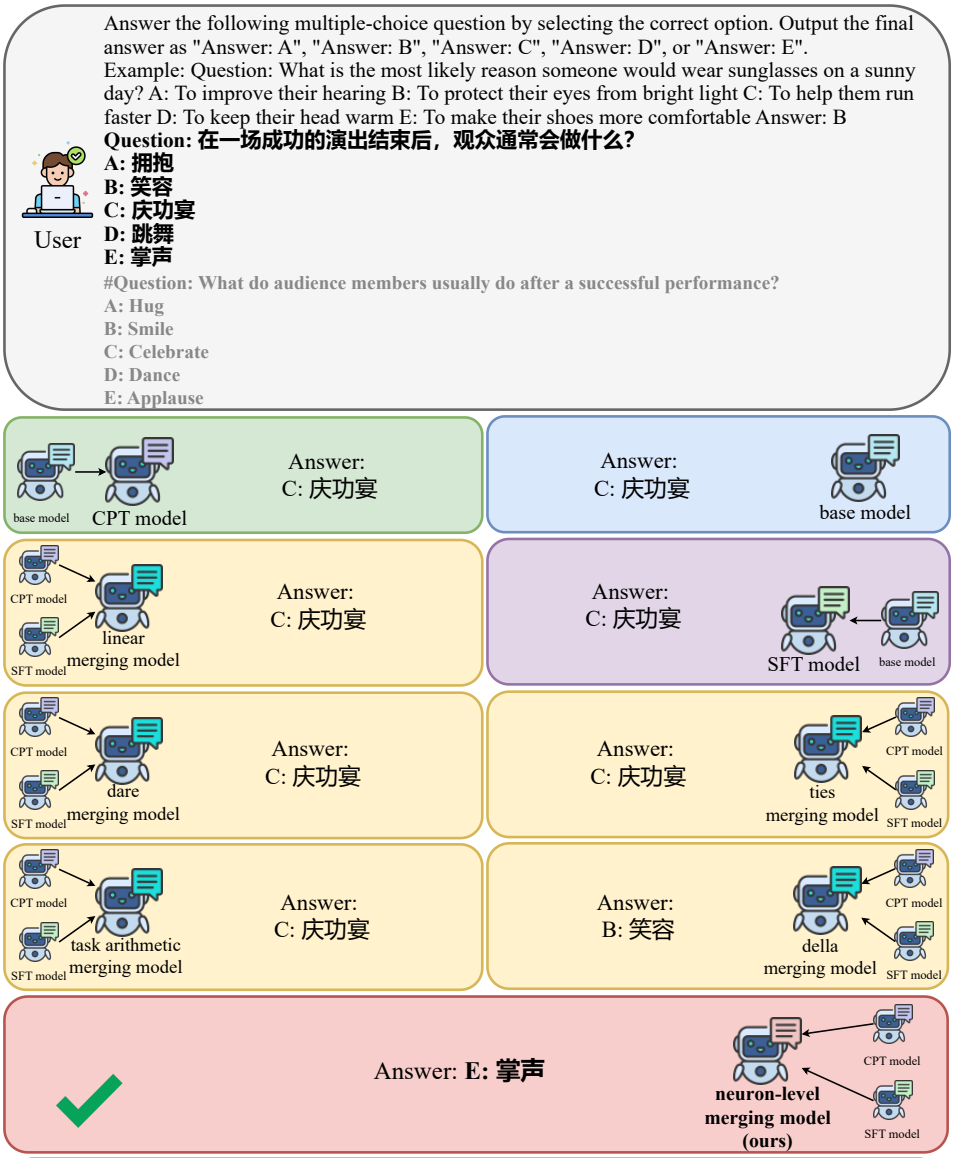
Figure 61: Case study on the mCSQA dataset in Telugu (te).



Case analysis:

This **Thai (th) mCSQA case** evaluates **fine-grained geographic and cultural knowledge** about regional specialties in China. The question asks which city in **Yunnan Province** is famous for its **copper craftsmanship**, where the correct answer is **Yuxi (E)**. The **base model and most baseline or parameter-level merging methods incorrectly favor more globally salient cities such as Kunming or Chengdu**, reflecting a strong **popularity bias** toward well-known administrative or metropolitan centers, even when they are not associated with the specific cultural product mentioned. Other merging baselines diverge toward **Zhaotong**, suggesting unstable reasoning when faced with unfamiliar regional facts. In contrast, the **neuron-level merging model correctly identifies Yuxi**, indicating its ability to **retrieve less prominent but more precise regional associations**. This case highlights that neuron-level merging can **mitigate salience-driven errors** and better preserve **localized world knowledge** in multilingual question answering scenarios.

Figure 62: Case study on the mCSQA dataset in Thai (th).



Case analysis:

This **Chinese (zh) mCSQA example** tests **everyday commonsense reasoning about social behavior**. The question asks what audiences usually do after a successful performance, where the correct and most salient response is **applause (E)**. However, the **base model and almost all parameter-level merging baselines incorrectly select "celebration banquet (庆功宴)"**, which is an event typically associated with performers or organizers rather than the audience, revealing a **role confusion error** and an over-reliance on loosely related celebratory concepts. One baseline shifts to "smile," which is plausible but still underspecified. In contrast, the **neuron-level merging model correctly produces "applause,"** demonstrating a stronger ability to **align the actor (audience) with the appropriate action** and to capture **high-frequency, culturally invariant commonsense associations**. This case illustrates how neuron-level merging can **suppress spurious semantic correlations** and recover **precise commonsense knowledge** in multilingual settings.

Figure 63: Case study on the mCSQA dataset in Chinese (zh).

Case	Case 1
Error Type	Reasoning / Calculation Error
Task	MGSM
Language	Russian
Original Question	Карла загружает файл размером 200 ГБ. Обычно она может загружать 2 ГБ в минуту, но на 40 % загрузки Windows делает принудительную перезагрузку для установки обновлений, которая занимает 20 минут. После этого Карле приходится перезапустить загрузку сначала. Сколько времени занимает загрузка файла?
Original CoT	Step-by-step answer: First find the total time to load the file without the forced Windows update: 200 GBY 2 G/min = <<200*2=400>>400 min. Then find the time to load the file after the update: 400 min + 20 min = <<400+20=420>>420 min. Then add the time to load the file without the update to find the total time: 420 min + 400 min = <<420+400=820>>820 min. Answer: 820.
English Translation	Carla is downloading a 200 GB file. She can normally download 2 GB per minute, but when 40% of the download is complete, Windows forces a reboot for updates that takes 20 minutes. After that, she must restart the download from the beginning. How long does the download take?
Gold Answer	160
Model Prediction	820
Diagnosis	The model miscomputes the download time by multiplying 200 by 2 instead of dividing 200 by 2, and then carries that mistake through a multi-step rate calculation. This is a numerical reasoning error rather than a simple language failure.

Case	Case 2
Error Type	Reasoning / Calculation Error
Task	MGSM
Language	Swahili
Original Question	Bata wa Janet hutaga mayai 16 kila siku. Huwa anakula matatu wakati wa staftahi kila asubuhi na huokea marafiki zake mafini kila siku kwa mayai manne. Huwa anauza yaliyosalia katika soko la wakulima kila siku kwa $2 kwa kila yai la bata. Huwa anatengeneza pesa ngapi katika dola kila siku katika soko la wakulima?
Original CoT	Step-by-step answer: Janet's hens lay 16 eggs per day. She eats 3 eggs per day for breakfast and sells 4 eggs to her friends. She sells 16 - 3 - 4 = <<16-3-4=11>>11 eggs in the farmer's market each day. She earns $22 per day in the farmer's market because 11 x 2 = <<11*2=22>>22. Answer: 22.
English Translation	Janet's ducks lay 16 eggs per day. She eats 3 for breakfast and uses 4 for muffins. She sells the rest at the farmers' market for $2 per egg. How many dollars does she make each day?
Gold Answer	18
Model Prediction	22
Diagnosis	The model follows the intended chain-of-thought format but computes 16 - 3 - 4 as 11 instead of 9. The wrong intermediate result directly leads to the incorrect final answer.

Figure 64: Representative **reasoning/calculation errors** of NLMR. Both cases are from MGSM and include the model's original chain-of-thought outputs. The examples show that NLMR can still make arithmetic or rate-calculation mistakes even when it follows a plausible reasoning format.

Case	Case 1
Error Type	Constraint / Relation Decomposition Error
Task	MGSM
Language	Russian
Original Question	Кайлар пошел в магазин, чтобы купить стаканы для своей новой квартиры. Один стакан стоит 5 $, но каждый второй стакан стоит всего лишь 60 % от цены. Кайлар хочет купить 16 стаканов. Сколько он должен за них заплатить?
Original CoT	Step-by-step answer: The price of one stack is 5 $, but the price of each second stack is only 60/100 * 5 = $<<5*60/100=3>>3. So for 16 stacks, Кайлар will pay 16 * 5 = $<<16*5=80>>80 for the first 16 stacks. For the next 16 stacks, he will pay 16 * 3 = $<<16*3=48>>48. In total Кайлар will pay 80 + 48 = $<<80+48=128>>128. Answer: 128.
English Translation	Kaylar went to buy glasses for a new apartment. One glass costs $5, but every second glass costs only 60% of that price. Kaylar wants to buy 16 glasses. How much should he pay?
Gold Answer	64
Model Prediction	128
Diagnosis	The model misreads the alternating discount rule and effectively counts 16 full-price glasses plus 16 discounted glasses, instead of 8 full-price and 8 discounted glasses. The failure comes from decomposing the relation structure incorrectly.

Case	Case 2
Error Type	Constraint / Relation Decomposition Error
Task	MGSM
Language	Swahili
Original Question	Toulouse lina kondoo mara mbili ya idadi ya kondoo katika Charleston. Charleston lina kondoo mara 4 ya idadi ya kondoo katika Seattle. Majiji ya Toulouse, Charleston, na Seattle yana jumla ya kondoo wangapi kwa pamoja iwapo Seattle lina kondoo 20?
Original CoT	Step-by-step answer: Toulouse has twice as many as Charleston, so 20 * 2 = <<20*2=40>>40. Charleston has 4 times as many as Seattle, so 20 * 4 = <<20*4=80>>80. Toulouse, Charleston, and Seattle have a total of 40 + 80 + 20 = <<40+80+20=140>>140 kondoo. Answer: 140.
English Translation	Toulouse has twice as many sheep as Charleston. Charleston has 4 times as many sheep as Seattle. If Seattle has 20 sheep, how many sheep do Toulouse, Charleston, and Seattle have altogether?
Gold Answer	260
Model Prediction	140
Diagnosis	The model fails to compose the relations correctly. It computes Charleston from Seattle, but then computes Toulouse directly from Seattle as 40, instead of using Charleston to get Toulouse = 2 × 80 = 160.

Figure 65: Representative **constraint/relation decomposition errors** of NLMR. These cases show that some remaining MGSM failures arise from incorrectly decomposing nested, alternating, or compositional relations, rather than from simple target-language misunderstanding.

Case	Case 1
Error Type	Knowledge Application Error
Task	ARC-Easy
Language	Thai
Original Question	ต้องใช้แรง 5 N เพื่อเพิ่มความเร็วของกล่องจากอัตรา 1.0 m/s เป็น 3.0 m/s ภายใน 5 วินาที ตามแนวพื้นผิวการเปลี่ยนแปลงใดน่าจะต้องใช้กำลังเพิ่มเติมเพื่อให้ได้ผลลัพธ์เดียวกัน Options: A: ลดมวลของกล่อง; B: เพิ่มมวลของกล่อง; C: ทำให้พื้นผิวของกล่องเรียบ; D: ทำให้พื้นผิวเรียบ
English Translation	A force of 5 N is needed to increase a box's speed from 1.0 m/s to 3.0 m/s in 5 seconds. Which change would most likely require additional force to produce the same result?
Gold Answer	B (เพิ่มมวลของกล่อง / increase the mass of the box)
Model Prediction	D (ทำให้พื้นผิวเรียบ / make the surface smooth)
Diagnosis	The model fails to apply the force-mass-acceleration relation. Increasing mass requires more force to produce the same acceleration. This is a knowledge-application failure.

Case	Case 2
Error Type	Knowledge Application Error
Task	ARC-Easy
Language	Thai
Original Question	ขณะเดินป่าผ่าน Granville State Forest นักเรียนคนหนึ่งพบสิ่งมีชีวิตคล้ายพืชที่ผิดปกติซึ่งดูเหมือนจะขาดคลอโรฟิลล์ เมื่อนักเรียนตรวจสอบตัวอย่างโดยใช้กล้องจุลทรรศน์ เขามองเห็นเซลล์จำนวนมากที่มีผนังเซลล์และไม่มีคลอโรพลาสต์ สิ่งมีชีวิตนี้น่าจะเป็นสมาชิกของอาณาจักรใด Options: A: สัตว์; B: ยูแบคทีเรีย; C: เชื้อรา; D: โปรติสต้า
English Translation	While hiking in Granville State Forest, a student found an unusual plant-like organism that seemed to lack chlorophyll. Under a microscope, the student saw many cells with cell walls and no chloroplasts. Which kingdom is this organism most likely a member of?
Gold Answer	C (เชื้อรา / fungi)
Model Prediction	B (ยูแบคทีเรีย / eubacteria)
Diagnosis	The model does not correctly apply the biological concept that organisms with cell walls but no chloroplasts are consistent with fungi rather than eubacteria. This is a scientific knowledge-application error.

Figure 66: Representative **knowledge application errors** of NLMR. These ARC-Easy cases show that NLMR may still fail when the question requires applying scientific or factual knowledge, even when the target-language input and answer options are largely interpretable.

Case	Case 1
Error Type	Option Grounding / Discrimination Error
Task	mCSQA
Language	Spanish
Original Question	¿Cuál es una tarea doméstica común que implica mantener a las mascotas limpias y saludables?
Options	A: limpieza de alfombras B: aseo de perros C: aspirar suelos D: lavar mascotas E: quitarle las pulgas a los perros
English Translation	What is a common household task that involves keeping pets clean and healthy?
Gold Answer	D (lavar mascotas / washing pets)
Model Prediction	B (aseo de perros / dog grooming)
Diagnosis	The predicted option is semantically related, but the model fails to ground the question to the most direct and appropriate answer choice. This is a fine-grained option discrimination error.

Case	Case 2
Error Type	Option Grounding / Discrimination Error
Task	ARC-Easy
Language	Chinese
Original Question	一名学生列出了 5 名学生在 10 分钟内跑完的距离。最好地组织这些数据的另一种方法是使用
Options	A: 饼图。 B: 条形图。 C: 折线图。 D: 象形文字。
English Translation	A student listed the distances that 5 students ran in 10 minutes. Another best way to organize these data is to use:
Gold Answer	B (条形图 / bar graph)
Model Prediction	C (折线图 / line graph)
Diagnosis	The model chooses a plausible but less appropriate visualization. A bar graph is better for comparing values across individual students, so the error reflects option grounding and fine-grained discrimination among related choices.

Figure 67: Representative **option grounding/discrimination errors** of NLMR. These cases indicate that some remaining multiple-choice failures involve selecting semantically related distractors or failing to identify the most appropriate candidate option.