Exploring Translation Mechanism of Large Language Models

Anonymous ACL submission

Abstract

Large language models (LLMs) have succeeded remarkably in multilingual translation tasks. However, the inherent translation mechanisms of LLMs remain poorly understood, largely due to sophisticated architectures and vast parameter scales. In response to this issue, this study explores the translation mechanism of LLM from the perspective of computational components (e.g., attention heads and MLPs). Path patching is utilized to explore causal relationships between components, detecting those crucial for translation tasks and subsequently analyzing their behavioral patterns in human-interpretable terms. Comprehensive analysis reveals that translation is predominantly facilitated by a sparse subset of specialized attention heads (less than 5%), which extract source language, indicator, and positional features. MLPs subsequently integrate and process these features by transiting towards English-centric latent representations. Notably, building on the above findings, targeted fine-tuning of only 64 heads achieves translation improvement comparable to full-parameter tuning while preserving general capabilities. [1]

1 Introduction

Large language models (LLMs) have succeeded remarkably in multilingual translation tasks (Chen et al., 2024a,b; Zhu et al., 2024; Zhang et al., 2024), paving the way for a new paradigm in machine translation (Xu et al., 2024a; Alves et al., 2024). Recent advancements have continuously focused on enhancing translation capabilities, bringing them progressively closer to human-level translation (Xu et al., 2024c; Lu et al., 2024; Xu et al., 2024b). Despite the widespread adoption and recent advancements in LLMs, the internal mechanisms by which they perform translation tasks remain poorly understood and pose severe challenges. Prior analyses focused on surface-level emergent linguistic phenomena (e.g., neuron activation patterns (Mu et al., 2024; Tang et al., 2024) or intermediate representations (Wendler et al., 2024; Zhu et al., 2024)), remaining *observational* rather than elucidating the *computational mechanistic basis* underlying translation. A comprehensive understanding of these functional mechanisms is critical for achieving robust improvements in translation capability and advancing the development of controllable and interpretable LLMs (Wang et al., 2023; Zhang et al., 2025).

In this paper, we study the internal mechanism of LLM translation by progressively investigating the following research questions:

- *Which components of LLMs crucially contribute to performing translation?*
- *What behavioral patterns do these translation-crucial components exhibit?*
- *Can fine-tuning these translation-crucial components enhance LLM translation capability?*

To this end, we leverage path patching (Goldowsky-Dill et al., 2023) to examine the causal relationships between computational components (e.g., attention heads and MLP), detecting those crucial for translation tasks. For components judged as crucial, we then systematically analyze their behavioral patterns by (1) characterizing attention head's roles according to the attention contribution to lexical alignment and (2) measuring correlations between MLP representations and translation-relevant token embeddings. Our analysis reveals three distinct attention head functional roles: (i) *source heads* that focus on source-language tokens, (ii) *indicator heads* that track translation-initiating signals, and (iii) *positional heads* that maintain sequential coherence. Additionally, we demonstrate that MLPs dynamically integrate translation-related features

[1] Our code and data will be released once accepted.

from critical attention heads, iteratively transiting them into English-centric latent representations.

Building on these insights, we design a targeted optimization strategy based on supervised fine-tuning (SFT) (Ouyang et al., 2022) to selectively fine-tune translation-crucial components, thereby assessing whether fine-tuning these components improves translation performance. As a result, our findings are as follows:

- *Only a sparse subset of heads (less than 5%) are crucial for LLMs' translation.*
- *Crucial heads exhibit specialized functions to process translation-relevant features, with MLPs integrating these features and transiting to English-centric latent representations.*
- *Fine-tuning merely 64 heads achieves performance parity with full-parameter fine-tuning.*

2 Related Works

Mechanistic Interpretability. Mechanistic interpretability (MI) elucidates neural network mechanisms by seeking to reverse-engineer and decode their functioning (Meng et al., 2022; Lan et al., 2024; Zhao et al., 2024a; Rai et al., 2024). Within the broader MI landscape, two key techniques are foundational to this work: (i) **Path patching** (Goldowsky-Dill et al., 2023; Wang et al., 2023), derived from activation patching (Heimersheim and Nanda, 2024; Zhang and Nanda, 2024), probes causal relationships and analyzes interactions between components in neural networks by tracing effect propagation along network pathways via targeted activation interventions. (ii) **Embedding projection** (Geva et al., 2022; Dar et al., 2023) maps high-dimensional representations to human-interpretable spaces via dimensionality reduction approaches. Recent studies highlight the utility of *path patching* to gain insights into functioning behavior, such as identifying circuits for tasks like indirect object identification(Wang et al., 2023) and arithmetic calculations(Zhang et al., 2025).

Interpretability in Multilingual LLMs. Recent studies have delved deeper into *how* LLMs achieve multilingually by investigating linguistic phenomena emergent in multilingual context (Bhattacharya and Bojar, 2024; Peng and Søgaard, 2024; Ferrando and Costa-jussà, 2024; Dumas et al., 2024). Key findings indicate that (i) increased linguistic diversity in inputs leads to reduced neuron activations (Mu et al., 2024); (ii) LLMs exhibit language-specific functional regions (Tang et al., 2024); and (iii) English frequently functions as an implicit computational pivot (Wendler et al., 2024; Zhao et al., 2024b). Unlike prior research focusing on surface-level emergent linguistic phenomena rather than computational translation mechanisms, this work comprehensively analyzes the functional processes underlying LLMs' translation abilities. We present a novel method to systematically examine how LLMs execute translation tasks, improving interpretability and practical understanding of their translation functionalities.

3 Constructing Analysis Dataset

Our goal is to explore the translation mechanism of LLMs. Directly leveraging existing sentence-level parallel corpora is challenging due to the lack of one-to-one word alignment between source and target languages. This motivates us to investigate how LLMs perform word-level translation. Taking inspiration from the prompt design and word selection in Wendler et al. (2024), we construct a closed set of word translation analysis datasets across five widely used languages (e.g., English (En), Chinese (Zh), Russian (Ru), German (De), and French (Fr)). Taking word translation from English to Chinese as an example, a word translation prompt like "English: book - 中文: " ("中文" means "Chinese") might appear in the analysis datasets. In this study, we select the samples that LLMs can translate correctly, denoting these sentences successfully prompting LLMs to translate as reference data using the notation of X_f. More details of the construction of word translation datasets can be referred to Appendix A.1.

Moreover, to meet the demand for component activation perturbation, we constructed a supplementary dataset comprising counterfactual sentences that exclude translation logic, using the notation of X_{cf}. The counterfactual sentences are generated adhering to two core principles: (1) preserving grammatical structures from the original X_f sentences and (2) replacing several crucial words responsible for the translation logic with contextually irrelevant terms. For instance, a sentence from X_f like "English: cloud - 中文: _" is replaced with the corresponding counterfactual one "English: cloud - Nothing: _". This isolates model's impact on translation tasks

from sentence structural or syntactic variables, enabling precise analysis of how LLMs perform translation tasks. Various counterfactual templates are considered in this study and provided in Appendix A.2. The constructed analysis dataset (including X_f and X_{cf}) is utilized in subsequent sections, as illustrated in the examples below:

Reference Data (X_f)

```
English: "cloud"  - 中文: "
English: "flower" - 中文: "
English: "snow"   - 中文: "
```

Counterfactual Data (X_{cf})

```
English: "cloud"  - Nothing: "
English: "flower" - Nothing: "
English: "snow"   - Nothing: "
```

4 Overview of Interpretation

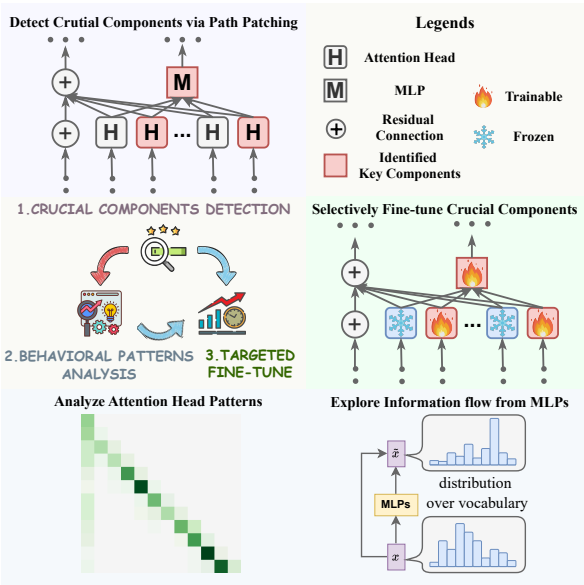


Figure 1: The overview of the interpretation method: (1) detect crucial components, (2) analyze their behavioral patterns, and (3) selectively fine-tune them.

Our method investigates the internal mechanisms of LLM translation through three steps:

1. Detecting, validating translation-crucial components and examining their consistency across training phases (Section §5).
2. Analyzing the inherent patterns of these components to characterize their behavioral and distinctive features (Section §6).
3. Implementing a targeted SFT strategy to fine-tune essential components and improve translation performance (Section §7).

5 Crucial Components Detection

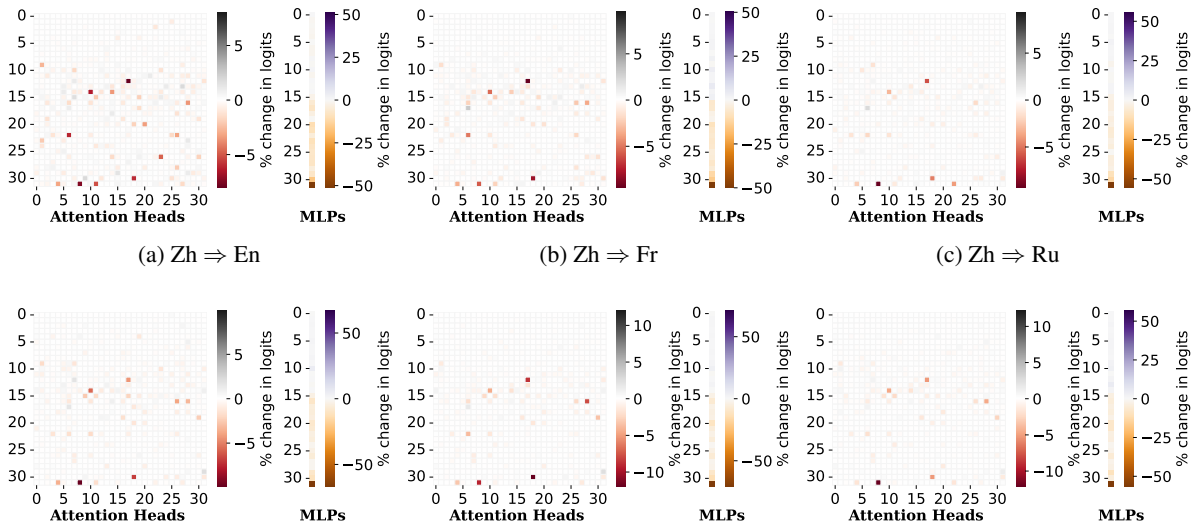
We begin by addressing the first research question: "*Which components crucially influence LLMs' translation capabilities?*" Using path patching, we detect components crucial for performing translation tasks (Section §5.1), subsequently validate their importance through knockout (Section §5.2), and further examine whether these heads exhibit consistency across training stages (Section §5.3).

5.1 Detecting Crucial Components for Translation Tasks via Path Patching

To determine the causal mechanisms behind the model's translation, we employ the *path patching* (Wang et al., 2023; Zhang et al., 2025). This method systematically analyzes causal relationships between two computation nodes (*Sender* → *Receiver*), evaluating whether the *Sender* causally influences the *Receiver*, and whether their connection is functionally crucial for translation tasks. We perturb specific activations using counterfactual data X_{cf}, while maintaining others at reference data X_f, measuring the counterfactual effect through output logit comparisons. Our method iteratively examines all components, isolates constituent circuits, and quantifies changes in ground-truth token logits. Appendix B provides more details of the method.

Detection results of crucial heads. We begin by examining the causal impact of logits from path patching each head across layers on LLaMA2-7B (Touvron et al., 2023). As a particularly clean case, we focus on two categories of translation directions: Zh ⇒ X, and X ⇒ Zh, which has many single-token words. We define "crucial heads" as those whose magnitude of logit change exceeds 1.0%. As illustrated in Figure 2, we highlight several key findings:

1. **Only a sparse subset of heads significantly influences translation performance.** For instance, patching the head at position $(31, 8)$ results in a substantial decrease in the target token's logit value, illustrating its critical role in the translation process.
2. **Impactful heads are concentrated in the middle and final layers.** Earlier layers lack heads directly influencing target token logits; instead, crucial heads cluster predominantly between layers 12 and 20 and in the final two



(a) Zh ⇒ En (b) Zh ⇒ Fr (c) Zh ⇒ Ru

(d) En ⇒ Zh (e) Fr ⇒ Zh (f) Ru ⇒ Zh

Figure 2: Importance of heads related to translation across different directions. Each square at position (x, y) refers to the x-th head in the y-th layer. Red (Brown) squares denote heads (MLPs) that have a positive impact on predicting the target token, while grey (purple) squares indicate heads (MLPs) with a negative effect.

layers. This pattern remains consistent across all translation directions.

3. **Crucial heads exhibit high transferability across translation directions.** A notable finding is the significant overlap of crucial heads across diverse language pairs. Analysis reveals that language pairs sharing the same source or target language exhibit a crucial attention head overlap exceeding 60%, while bidirectional translation pairs (e.g., Zh ⇔ En) surpass 70%. This overlap suggests these heads serve generalizable functions in translation, independent of translation directions. Their consistency across language pairs underscores their importance and transferability, indicating contributions to core translation mechanisms regardless of specific languages.

For robustness, we also conduct additional experiments on detecting crucial heads in other LLMs and other directions (e.g., En ⇒ X, and X ⇒ En). Details are provided in the Appendix C.

Detection results of crucial MLPs. Similar to crucial heads, most MLPs in earlier layers (0–14) exhibit negligible influence on output logits, with changes confined to approximately $\pm 0.0\%$. Crucial MLPs cluster predominantly after layer 15, exceeding 5.0% logit change, whereas the final layer MLP exhibits a substantial impact—reaching 70.0% on target token logit change. This strong correlation between later MLP layers and logit changes underscores their progressively critical role in shaping translations as processing advances.

5.2 Validating Crucial Heads Through Knockout

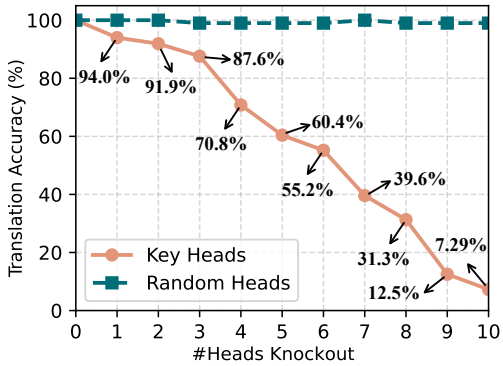
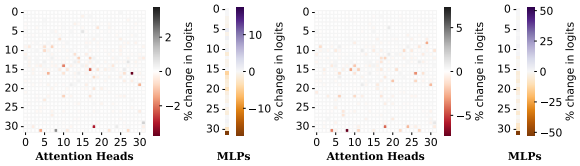


Figure 3: The influence on **En** ⇒ **Zh** translation accuracy in the analysis dataset when attention heads are progressively knocked out, sorted by their effect on logits ("key heads"), and randomly ("random heads")

Interpretive analyses of model components risk misleading or non-rigorous (Bolukbasi et al., 2021; Wiegreffe and Pinter, 2019). To ensure reliability, we validate the significance of detected crucial heads and test the irrelevance of non-crucial ones via *mean ablation* (Wang et al., 2023). This method replaces a component's activation with average activations across counterfactual data X_{cf}, thereby removing task-specific information. Performance decline confirms a component's importance for translation tasks, whereas no significant performance change indicates uncritical.

Validation results on the analysis dataset. We examine how incrementally knocking out En ⇒ Zh crucial heads affects LLM translation performance on the analysis dataset. As shown in Figure 3, disabling "*crucial heads*" leads to a significant decline in translation accuracy, whereas knocking out "*random heads*" causes minor fluctuations, with accuracy remaining stable within 2%. These results highlight the essential role of the detected key attention heads in sustaining the translation functionality of the LLM.

5.3 Examine Consistency of Crucial Heads Across Training



(a) CPT-LLM (b) SFT-LLM

Figure 4: Importance of heads related to **En⇒ Zh** translation across LLM after CPT or SFT.

To investigate whether crucial attention heads remain consistent across distinct training phases, we analyze (1) continued pre-training (CPT) (Xu et al., 2024a) on the LLaMA-2-7B base model on 1 billion tokens of OSCAR data (Ortiz Suárez et al., 2020) and (2) supervised fine-tuning (SFT) (Ouyang et al., 2022) on LLaMA-2-7B base model on the WMT17-22 validation dataset.

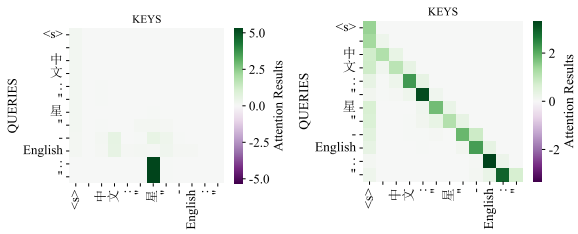
Detection results across different training phases. Following Section §5.1, we examine the causal impact of logits on different LLM training phases in En ⇒ Zh translation of analysis dataset. The results are illustrated in Figure 4, compared to the base LLM results in Figure 2d, LLMs after CPT exhibit significant distributional shifts in translation-crucial heads, whereas changes are minimal after SFT. This demonstrates that pre-training stage changes LLMs' core translation capabilities, while supervised fine-tuning primary focuses on localized parameter adjustments without altering their fundamental abilities.

6 Behavioral Patterns Analysis

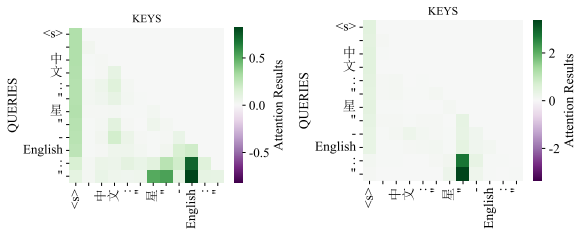
Motivated by the sparse distribution of crucial heads, we now turn to the second research question: "*What behavioral patterns do translation-crucial components exhibit?*" by systematically investigating their computational mechanisms through two interpretable diagnostic methods: (1) visualizing attention patterns to characterize the roles of crucial heads (Section §6.1), and (2) projecting MLP representation to measure correlations with translation-related token embeddings (Section §6.2).

6.1 Analysis of Attention Head

Following the findings of Kobayashi et al. (2020), who demonstrates that attention weights alone fail to explain model behavior, we inspect attention values $\mathbf{O}^{i,j} \in \mathbb{R}^{N\times N}$ (where N denotes the sequence length) to analyze significant token interactions. We compute $\mathbf{O}^{i,j} = \sum_{n=1}^{N} \mathbf{A}_n^{i,j}\mathbf{X}_f\mathbf{W}_V^{i,j}$ over reference data $\mathbf{X}_f$ for each analyzed head (i,j), where, $\mathbf{A}_{i,j}^{n}$ denotes attention weights and $\mathbf{W}_V^{i,j}$ value matrix. Each heads' role is determined by salient feature of $\mathbf{O}_{\{END\}}^{i,j} \in \mathbb{R}^{1\times N}$ between the END position's Query token and all Key tokens.



(a) Source Head $(17, 18)$ (b) Positional Head $(15, 11)$



(c) Indicator Head $(14, 27)$ (d) Indicator Head $(14, 4)$

Figure 5: The attention values visualization of the role-classified key heads, which show different characteristics of different crucial heads.

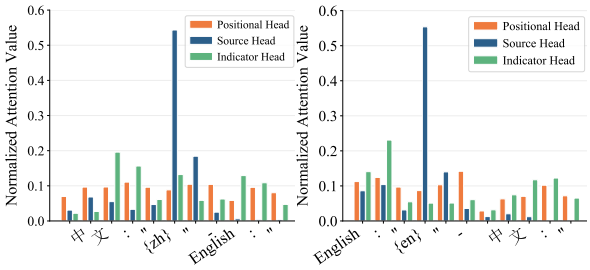
Characterizing heads. To better understand the "behavior" of the translation-crucial heads, we first gain an intuitive insight by visualizing their attention values as shown in the case in Figure 5. Our findings indicate that these heads exhibit distinct focus patterns across different types of input tokens. Building on these patterns and following Voita et al. (2019), we further categorize these heads into three distinct functional roles:

- **Source Heads** demonstrate concentrated attention on source-language tokens, specializing in cross-lingual alignment. These heads

facilitate lexical transfer by identifying source language tokens among the input sequence.

- **Indicator Heads** exhibit spike-shaped attention patterns on translation-specific indicators (e.g., language identifiers like "English" or "中文", and structural cues like colons), assisting translation mode recognition and syntactic boundary detection.

- **Positional Heads** predominately attend to adjacent tokens, managing contextual dependencies and resolving grammatical agreement.



(a) Zh ⇒ En (b) En ⇒ Zh

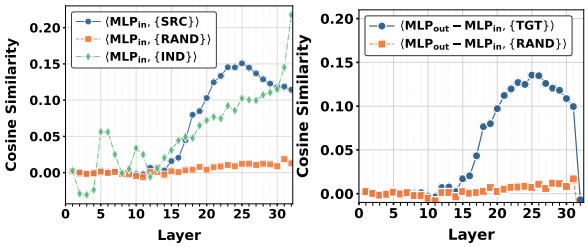
Figure 6: The attention value distribution of different roles of key heads across Zh ⇔ En translation tasks.

Distinct attention distribution across heads. To quantitatively analyze the distinct patterns of the crucial heads' roles, we randomly selected 100 samples from the analysis dataset and plotted the distribution of averaged attention values for the three key head roles across two translation directions (Zh ⇔ En). Figure 6 demonstrates that these heads exhibit distinct attention distributions, with minimal focus on tokens outside important input tokens. The source heads primarily attend to source input tokens, the positional heads distribute attention uniformly across the input context, and the indicator heads concentrate on translation task indicator tokens.

Overall, these analyses provide a clear, human-interpretable perspective of why deactivating crucial heads significantly impacts LLM translation.

6.2 Analysis of MLP

To analyze linguistic content encoded in the inputs (MLP_{in}) and outputs (MLP_{out}) of MLP layers, particularly for translation-relevant tokens: translation indicator (IND), source (SRC) and target-language (TGT), we employ the unembedding matrix W_U as a diagnostic probe and $W_U[*]$ denotes the unembedding vectors corresponding to a specific token. For each token TOK, we compute cosine similarities (denoted as $\langle MLP, \text{TOK}\rangle$) between MLP_{in}, MLP_{out}, and $W_U[\{TOK\}]$ to quantify linguistic information propagation through MLP layers. Following Geva et al. (2022), we isolate MLP contributions evaluating: $\frac{MLP_{out}-MLP_{in}}{\|MLP_{out}-MLP_{in}\|} \cdot \frac{W_U[\{TOK\}]}{\|W_U[\{TOK\}]\|}, TOK \in \{\text{IND}, \text{SRC}, \text{TGT}\}$.

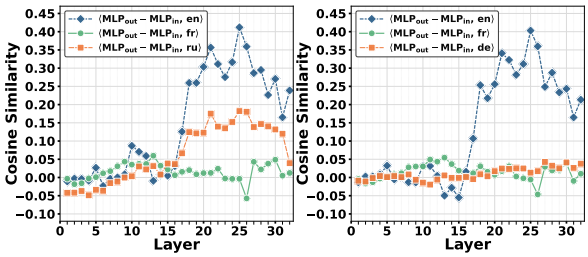


(a) {SRC}, {IND} Reception (b) Generation of {TGT}

Figure 7: We investigate the correlation between MLP input or output with translation-related tokens.

MLPs integrate and process translation-related features iteratively, yielding target translation. Figure 7 investigates MLP interactions with source and target tokens across 100 En ⇒ Zh samples. Figure 7a presents that in early layers (1–14), $\langle MLP_{in}, \text{SRC}\rangle$ values remain near-zero, indicating minimal source token encoding, consistent with the inactive region before layer 14 in Figure 2d. A surge in $\langle MLP_{in}, \text{SRC}\rangle$ occurs between layers 15–25, aligning with activation of key attention heads (e.g., 15.12 and 16.26), where source information is encoded into MLP representations for downstream processes. From layers 25–31, $\langle MLP_{in}, \text{SRC}\rangle$ declines, signaling a transition to target translation. Concurrently, ($\langle MLP_{in}, \text{IND}\rangle$) rises after layer 12 and peaks in the final layers, enabling coherent target-language generation. Critically, control comparisons with random English tokens ($\langle MLP_{in}, \text{RAND}\rangle$) remain near-zero throughout all layers, confirming the specificity of the observed patterns. As shown in Figure 7b, starting at layer 15, where MLPs start processing target token information, $\langle MLP_{out} - MLP_{in}, W_U[\text{TARGET}]\rangle$ sharply increases, while $\langle MLP_{out} - MLP_{in}, W_U[\text{RANDOM}]\rangle$ declines. This indicates that MLPs progressively execute translation across layers. Parallel trends in other LLMs (Appendix C) confirm their generality.

MLP intermediate features exhibit a transition to English-centric latent representation. We further investigate the detailed translation process between non-English pairs (e.g., German/Russian ⇒ Chinese) by analyzing the word "*book*". Quan-



(a) De ⇒ Zh (b) Ru ⇒ Zh

Figure 8: We investigate the correlation between intermediate representation with different languages tokens unembedding vector.

titative comparisons between $MLP_{out} - MLP_{in}$ representations and cross-lingual semantic embeddings (Figure 8) reveal: in layers 16–26, similarity with English embeddings surpasses other languages, declining in later layers (25–31). We hypothesize LLMs employ a "bridge-translation" paradigm—akin to humans using their native language as a mental intermediary—where source inputs are first processed into English-centric latent representations before generating target outputs. This aligns with prior work (Wendler et al., 2024; Zhao et al., 2024b), confirming English's latent intermediary role in multilingual LLM tasks.

Consolidating these findings, we conclude that LLMs employ attention heads to capture source language and translation indicator tokens, which are forwarded to downstream MLPs. MLPs integrate and process these features by transiting towards an English-centric latent representation, finally generating the target translation.

7 Targeted Fine-tune

Building on the insights given from two previous investigations, we aim to answer the final question: "*Can fine-tuning these translation-crucial components enhance LLM translation capability?*" To address this, we propose a method to fine-tune these components selectively, as detailed in Section §7.1. We then introduce our experimental setup in Section §7.2 and further carry out three sets of experiments (Section §7.3, §7.4, and §7.5) to comprehensively evaluate the proposed method.

7.1 Selectively Fine-tune Crucial Components

SFT is a common technique for improving translation performance in LLMs (Jiao et al., 2023; Xu et al., 2024a). Building on this, our method selectively updates parameters directly tied to translation tasks (those detected as crucial in Section §5.1) while preserving the remaining. This strategy aims to precisely improve the model's translation capabilities without compromising general functionality. Given crucial translation-related components Θ, our method computes gradients G for Θ rather than for the entire set of parameters and iteratively adjusts these parameters. Modifying only a subset of parameters reduces training duration and mitigates interference with the model's pre-existing capabilities.

7.2 Experimental Setup

We examine three approaches: (1) Full-parameter fine-tuning (Full SFT), (2) Targeted fine-tuning of translation-crucial components (Targeted SFT), and (3) Random-component fine-tuning (Random SFT), where random components match the parameter count of Targeted SFT. For training, we leverage human-parallel corpora (WMT17–WMT22, Flores-200 (Guzmán et al., 2019)) following Xu et al. (2024a), evaluating translation accuracy on WMT23/24 and general-domain benchmarks (MMLU (Hendrycks et al., 2021), ARC (Clark et al., 2018), SIQA (Sap et al., 2019)). Implementation details are in Appendix D.

Our experiments focus on two goals: (1) comparing Full, Targeted, and Random SFT on Llama-2-7B across six bidirectional translation tasks (English ⇔ Chinese, German, Russian), and (2) assessing generalization by fine-tuning English ⇒ Chinese crucial heads and testing performance on English ⇔ Japanese/Czech translation tasks.

7.3 Comparison Experimental Results

As shown in Tables 1 and 2, Targeted SFT demonstrates three key advantages: (1) **Translation performance improvement**—it significantly improves translation performance across all language directions, particularly in X ⇒ En, outperforming Full SFT and far exceeding Random SFT; (2) **General capabilities preservation**-unlike Full SFT, which degrades non-translation task performance, Targeted SFT maintains baseline generalization; (3) **Training efficiency**-it modifies fewer than 5% of parameters and halves training time compared to Full SFT, achieving substantial computational cost savings. Additional results for other LLMs are provided in Appendix E.

7.4 Generalization Evaluation Results

Table 3 demonstrates that translation-crucial attention heads exhibit cross-lingual generalization: fine-tuning only the **En** ⇒ **Zh** crucial heads

Models	Train Speed	Tuned Params.	Translation Tasks			Generic Tasks	
			Zh⇒En	De⇒En	Ru⇒En	MMLU	Commonsense Reasoning
			BLEU↑/COMET↑/BLEURT↑			Acc.	Acc.
LLaMA2-7B	-	-	15.6/73.1/56.6	24.8/76.8/62.1	20.2/73.8/60.3	45.9	55.3
+ Full SFT	17sam./sec.	6.7B	20.4/78.7/63.9	35.4/83.4/70.7	25.8/79.8/67.6	42.6	50.2
+ Targeted SFT	33sam./sec.	0.27B	21.3/79.1/64.3	37.1/83.7/71.4	27.8/80.3/68.4	46.0	55.7
+ Random SFT	33sam./sec.	0.27B	16.9/76.9/61.1	32.5/81.6/68.1	23.7/78.2/65.3	45.9	54.9

Table 1: The overall results of **X** ⇒ **En** translation on WMT'23/24 and generic tasks. Results surpassing Full SFT are highlighted in green, inferior outcomes in red, and comparable performance in blue.

Models	Train Speed	Tuned Params.	Translation Tasks			Generic Tasks	
			En⇒Zh	En⇒De	En⇒Ru	MMLU	Commonsense Reasoning
			BLEU↑/COMET↑/BLEURT↑			Acc.	Acc.
LLaMA2-7B	-	-	17.0/74.1/55.9	13.0/64.2/49.1	12.8/70.5/52.4	45.9	55.3
+ Full SFT	17sam./sec.	6.7B	30.3/80.7/62.9	27.9/78.3/63.7	19.5/80.0/63.2	40.2	50.0
+ Targeted SFT	33sam./sec.	0.27B	30.7/81.4/64.3	27.6/78.4/63.8	20.1/80.4/63.6	46.2	56.0
+ Random SFT	33sam./sec.	0.27B	26.4/79.3/61.6	22.7/76.2/60.3	15.8/77.9/60.7	46.1	55.2

Table 2: The overall results of **En** ⇒ **X** translation on WMT'23/24 and generic tasks.

in Llama-2-7B and evaluating them on other translation directions (*En* ⇔ *Cs (Czech)* and *En* ⇔ *Ja (Japanese)*) achieves performance gains comparable to full-parameter fine-tuning.

Models	En⇒Cs	En⇒Ja	Cs⇒En	Ja⇒En
	BLEU↑/COMET↑/BLEURT↑			
LLaMA2-7B	4.4/63.6/39.7	6.1/73.3/47.4	23.7/77.9/65.1	10.8/72.9/56.6
+ Full SFT	20.2/80.0/66.5	15.2/82.4/56.7	31.9/83.1/71.7	17.4/79.5/64.1
+ Targeted SFT	20.8/80.3/66.7	15.3/81.9/56.7	33.5/83.5/72.3	18.7/80.0/64.7
+ Random SFT	15.8/78.5/63.8	11.3/79.9/53.7	29.1/81.5/68.8	14.0/77.9/62.1

Table 3: WMT'23/24 *En* ⇔ *Cs* and *En* ⇔ *Ja* Results. Targeted SFT fine-tunes **En** ⇒ **Zh** crucial heads.

7.5 Ablation Study of Trainable Components

Ablating Attention Heads	Train Speed	Tuned Params.	Zh ⇒ En	MMLU
			BLEU/COMET/BLEURT	Acc.
top-8 heads	58sam./sec.	0.017B	18.7/78.1/63.0	46.1
top-16 heads	52sam./sec.	0.033B	20.0/78.4/63.5	45.9
top-32 heads	50sam./sec.	0.067B	20.4/78.6/63.8	45.8
top-64 heads	40sam./sec.	0.134B	**21.3/79.1/64.3**	45.9
top-96 heads	36sam./sec.	0.134B	21.0/79.0/64.2	45.7
top-128 heads	33sam./sec.	0.268B	21.1/79.1/64.4	45.5
top-160 heads	30sam./sec.	0.335B	21.3/79.1/64.4	45.3

Table 4: Ablative experiments on the number of heads. The most cost-effective setting is shown in green.

We conduct ablation studies in Zh ⇒ En translation on WMT'23/24 to examine how varying the number of fine-tuned attention heads and MLPs affects translation performance, generic capabilities, and training efficiency. As shown in Table 4, fine-tuning 64 attention heads achieves the optimal balance between performance and computational cost. Table 5 reveals that increasing MLPs enhances translation performance but more significantly degrades generic capabilities and training speed compared to tuning additional heads.

Ablating MLPs	Train Speed	Tuned Params.	Zh ⇒ En	MMLU
			BLEU/COMET/BLEURT	Acc.
Top-64 heads	33sam./sec.	0.27B	21.3/79.1/64.3	45.8
+top-1 MLP	30sam./sec.	0.41B	21.8/79.1/64.5	45.7
+top-2 MLP	27sam./sec.	0.54B	21.8/79.1/64.5	45.6
+top-3 MLP	24sam./sec.	0.68B	21.9/79.1/64.5	45.3
+top-5 MLP	20sam./sec.	0.95B	22.1/79.2/64.6	44.2
+all MLP	18sam./sec.	4.62B	22.5/79.4/64.7	42.8

Table 5: Ablative experiments on the number of MLPs.

8 Conclusion

This study systematically explores the translation mechanism of LLM by progressively addressing three research questions. We begin by detecting components crucial for translation via path patching and find that only a sparse subset of components (less than 5%) are indispensable for translation. These heads exhibit specialized functions, extracting translation-related features, while MLPs integrate and process by transiting toward English-centric latent representations. Based on these findings, we found that targeted fine-tuning of merely 64 translation-crucial heads achieves performance parity with full-parameter tuning. These findings collectively advance the interpretability of the inner translation mechanism of LLMs.

Limitations

This study acknowledges two methodological considerations that guide future research directions. While the intentionally simplified lexical translation task provided crucial experimental control to isolate core mechanisms, extending these findings to more ecologically valid sentence-level contexts would strengthen their practical relevance. Furthermore, although our parameter-aware methodology proves effective across open-source architectures, its applicability to closed-source systems remains theoretically constrained—a limitation that simultaneously highlights the urgent need for developing model-agnostic analysis frameworks in this evolving research domain.

References

Duarte M. Alves, José Pombal, Nuno M. Guerreiro, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024. Tower: An open multilingual large language model for translation-related tasks. *Preprint*, arXiv:2402.17733.

Sunit Bhattacharya and Ondřej Bojar. 2024. Understanding the role of ffns in driving multilingual behaviour in llms. *Preprint*, arXiv:2404.13855.

Tolga Bolukbasi, Adam Pearce, Ann Yuan, Andy Coenen, Emily Reif, Fernanda Viégas, and Martin Wattenberg. 2021. An interpretability illusion for bert. *Preprint*, arXiv:2104.07143.

Andong Chen, Kehai Chen, Yang Xiang, Xuefeng Bai, Muyun Yang, Yang Feng, Tiejun Zhao, and Min zhang. 2024a. Llm-based translation inference with iterative bilingual understanding. *Preprint*, arXiv:2410.12543.

Andong Chen, Lianzhang Lou, Kehai Chen, Xuefeng Bai, Yang Xiang, Muyun Yang, Tiejun Zhao, and Min Zhang. 2024b. DUAL-REFLECT: Enhancing large language models for reflective translation through dual learning feedback mechanisms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 693–704, Bangkok, Thailand. Association for Computational Linguistics.

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

Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. 2023. Analyzing transformers in embedding space. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16124–16170, Toronto, Canada. Association for Computational Linguistics.

Clément Dumas, Veniamin Veselovsky, Giovanni Monea, Robert West, and Chris Wendler. 2024. How do llamas process multilingual text? a latent exploration through activation patching. In *ICML 2024 Workshop on Mechanistic Interpretability*.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2021/framework/index.html.

Javier Ferrando and Marta R. Costa-jussà. 2024. On the similarity of circuits across languages: a case study on the subject-verb agreement task. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10115–10125, Miami, Florida, USA. Association for Computational Linguistics.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. 2023. Localizing model behavior with path patching. *arXiv preprint arXiv:2304.05969*.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6098–6111, Hong Kong, China. Association for Computational Linguistics.

Stefan Heimersheim and Neel Nanda. 2024. How to use and interpret activation patching. *Preprint*, arXiv:2404.15255.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Wenxiang Jiao, Jen-tse Huang, Wenxuan Wang, Zhiwei He, Tian Liang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. ParroT: Translating during chat using large language models tuned with human translation and feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15009–15020, Singapore. Association for Computational Linguistics.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7057–7075, Online. Association for Computational Linguistics.

Michael Lan, Philip Torr, Austin Meek, Ashkan Khakzar, David Krueger, and Fazl Barez. 2024. Sparse autoencoders reveal universal feature spaces across large language models. *Preprint*, arXiv:2410.06981.

Yinquan Lu, Wenhao Zhu, Lei Li, Yu Qiao, and Fei Yuan. 2024. LLaMAX: Scaling linguistic horizons of LLM by enhancing translation capabilities beyond 100 languages. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10748–10772, Miami, Florida, USA. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*, volume 35, pages 17359–17372. Curran Associates, Inc.

Yongyu Mu, Peinan Feng, Zhiquan Cao, Yuzhang Wu, Bei Li, Chenglong Wang, Tong Xiao, Kai Song, Tongran Liu, Chunliang Zhang, and JingBo Zhu. 2024. Revealing the parallel multilingual learning within large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6976–6997, Miami, Florida, USA. Association for Computational Linguistics.

Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2020. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Qiwei Peng and Anders Søgaard. 2024. Concept space alignment in multilingual LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5511–5526, Miami, Florida, USA. Association for Computational Linguistics.

Daking Rai, Yilun Zhou, Shi Feng, Abulhair Saparov, and Ziyu Yao. 2024. A practical review of mechanistic interpretability for transformer-based language models. *Preprint*, arXiv:2407.02646.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China. Association for Computational Linguistics.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5701–5715, Bangkok, Thailand. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2023. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in English?

on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand. Association for Computational Linguistics.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 11–20, Hong Kong, China. Association for Computational Linguistics.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024a. A paradigm shift in machine translation: Boosting translation performance of large language models. In *The Twelfth International Conference on Learning Representations*.

Haoran Xu, Kenton Murray, Philipp Koehn, Hieu Hoang, Akiko Eriguchi, and Huda Khayrallah. 2024b. X-alma: Plug & play modules and adaptive rejection for quality translation at scale. *Preprint*, arXiv:2410.03115.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024c. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Forty-first International Conference on Machine Learning*.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2025. Language models are super mario: absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Fred Zhang and Neel Nanda. 2024. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations*.

Hongbin Zhang, Kehai Chen, Xuefeng Bai, Yang Xiang, and Min Zhang. 2024. Paying more attention to source context: Mitigating unfaithful translations from large language model. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13816–13836, Bangkok, Thailand. Association for Computational Linguistics.

Wei Zhang, Chaoqun Wan, Yonggang Zhang, Yiu-ming Cheung, Xinmei Tian, Xu Shen, and Jieping Ye. 2025. Interpreting and improving large language models in arithmetic calculation. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024a. Explainability for large language models: A survey. *ACM Trans. Intell. Syst. Technol.*, 15(2).

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024b. How do large language models handle multilingualism? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2765–2781, Mexico City, Mexico. Association for Computational Linguistics.

A Translation Task Templates and Examples

As a clear case study, we first focus on Chinese due to its prevalence of single-token words and lack of spacing. We analyze Llama-2's vocabulary to identify single-token Chinese words (primarily nouns) with direct single-token English translations. This enables direct comparison of the model's next-token probabilities for correct Chinese words and their English equivalents. For robustness, we replicate experiments in German, Russian, and French, compiling datasets of 139 Chinese, 120 German, 115 Russian, and 118 French words.

A.1 Dataset Construction

To ensure the next token is unambiguously inferable as a single token, we design translation prompts where x_{n+1} is uniquely determined by the preceding context $x_1 ... x_n$. Each prompt specifies the source language, word, and target language, requiring the model to predict the translated word. Taking English-to-Chinese as an example, a word translation like "English: flower - 中文: 花" ("中文" means "Chinese", "花" means "flower") might naturally appear in the pretraining corpus.

Such prompts explicitly guide Llama-2 to perform translation by leveraging its pretrained linguistic knowledge.

A.2 Templates

We formalize counterfactual prompt generation through systematic grammatical preservation and semantic disruption, operating under two core design principles:

- **Structural Isomorphism**: Maintain original syntactic patterns (interrogative formats, placeholder positions, punctuation) while altering semantic content
- **Targeted Lexical Substitution**: Replace critical components through four operation classes

Perturbation Taxonomy The perturbation strategies fall into four principal categories, as detailed in Table 6:

Validation Protocol The constructed templates undergo rigorous verification:

1. *Grammatical Integrity Check*: Measure template fluency via language model perplexity scores (threshold: $\leq$15% deviation from originals)

Operation Type	Implementation Mechanism
Target Nullification	Replace language identifiers with non-linguistic concepts ({tgt_lang} → "Void"/"Null")
Action Distortion	Substitute translation verbs with irrelevant actions ("translate" → "eat"/"delete")
Semantic Obfuscation	Alter task-specific nouns to disrupt functionality ("translation" → "color"/"flavor")
Paradox Insertion	Introduce self-contradictory modifiers ("into {tgt_lang}" → "into a silent rock")

Table 6: Taxonomy of Counterfactual Perturbation Operations

2. *Task Disruption Test*: Verify semantic shift through human annotation (success criterion: $\geq$90% agreement on functionality removal)

Implementation Advantages Our methodology provides three key benefits:

- Controlled isolation of template components affecting model behavior
- Cross-lingual consistency through placeholder-based design
- Reproducible taxonomy enabling systematic ablation studies

The counterfactual prompts we used are shown in Table 7

B Path Patching for Detecting Components Crucial for LLM Translation

Algorithm 1 Critical Component Detection via Path Patching

Require: Dataset $\mathcal{D}$ containing factual/counterfactual pairs (X_f, X_{cf}), model $\mathcal{F}$ with components C
Ensure: Node importance scores $\Delta = \delta_1, ..., \delta_m$

for each data pair $(X_f^{(i)}, X_{cf}^{(i)}) \in \mathcal{D}$ **do**
  Compute reference activations $\mathbf{H}_f \leftarrow \mathcal{F}(X_f^{(i)})$
  Compute contrastive activations $\mathbf{H}_{cf} \leftarrow \mathcal{F}(X_{cf}^{(i)})$
  for each component $c^{(j)} \in C$ **do**
    Create hybrid activation map $\widetilde{\mathbf{H}}_f$ where:
    $$\widetilde{H}_f \leftarrow \begin{cases} H_{cf}^k & \text{if } k = c^{(j)} \\ H_f^k & \text{otherwise} \end{cases}$$
    Obtain original logit $y_f \leftarrow \mathcal{F}(X_f; \mathbf{H}_f)$
    Obtain patched logit $\widetilde{y}_f \leftarrow \mathcal{F}(X_f; \widetilde{\mathbf{H}}_f)$
    Calculate patched effect: $\delta_j^{(i)} \leftarrow \frac{\widetilde{y}_f - y_f}{y_f + \epsilon}$
  end for
end for
for each importance score $\delta_i \in \Delta$ **do**
  Aggregate across dataset: $\delta_i \leftarrow \frac{1}{|\mathcal{D}|} \sum_{j=1}^{|\mathcal{D}|} \delta_i^{(j)}$
end for
return Node importance scores Δ

The computation of large language models (LLMs) can be formalized as a directed acyclic

Normal Prompt	Counterfactual Prompt	Perturbation Type
`{src_lang}: "{src_word}" - {tgt_lang}: "{tgt_word}`	`{src_lang}: "{src_word}" - There is nothing: "{tgt_word}`	Target Nullification
`Translate "{src_word}" into {tgt_lang}: "`	`Translate "{src_word}" into Nothing: "`	Target Nullification
`Translate the {src_lang} word "{src_word}" to {tgt_lang}: "`	`Translate the {src_lang} word "{src_word}" to Null: "`	Target Nullification
`From {src_lang}: "{src_word}" to {tgt_lang}: "`	`From {src_lang}: "{src_word}" to Nowhere: "`	Target Nullification
`Provide the translation of "{src_word}" from {src_lang} to {tgt_lang}: "`	`Provide the color of "{src_word}" from {src_lang} to {tgt_lang}: "`	Action Distortion
`Q: How do you say "{src_word}" in {tgt_lang}? A: "`	`Q: How do you eat "{src_word}" in {tgt_lang}? A: "`	Action Distortion
`Q: What is the {tgt_lang} translation "{src_word}"? A: "`	`Q: What is the {tgt_lang} flavor "{src_word}"? A: "`	Semantic Obfuscation
`Translate "{src_word}" into {tgt_lang}: "`	`Translate "{src_word}" into a silent rock: "`	Paradox Insertion
`Q: What is "{src_word}" translated into {tgt_lang}? A: "`	`Q: What is "{src_word}" erased into {tgt_lang}? A: "`	Action Distortion
`From {src_lang}: "{src_word}" - {tgt_lang}: "{tgt_word}`	`From {src_lang}: "{src_word}" - Disabled: "{tgt_word}`	Action Distortion

Note: All placeholders ({src_lang}, {src_word}, etc.) follow actual implementation syntax. Counterfactual perturbations preserve original grammatical structures while altering translation semantics through targeted substitutions.

Table 7: Examples of some regular translation prompt templates and counterfactual prompt templates.

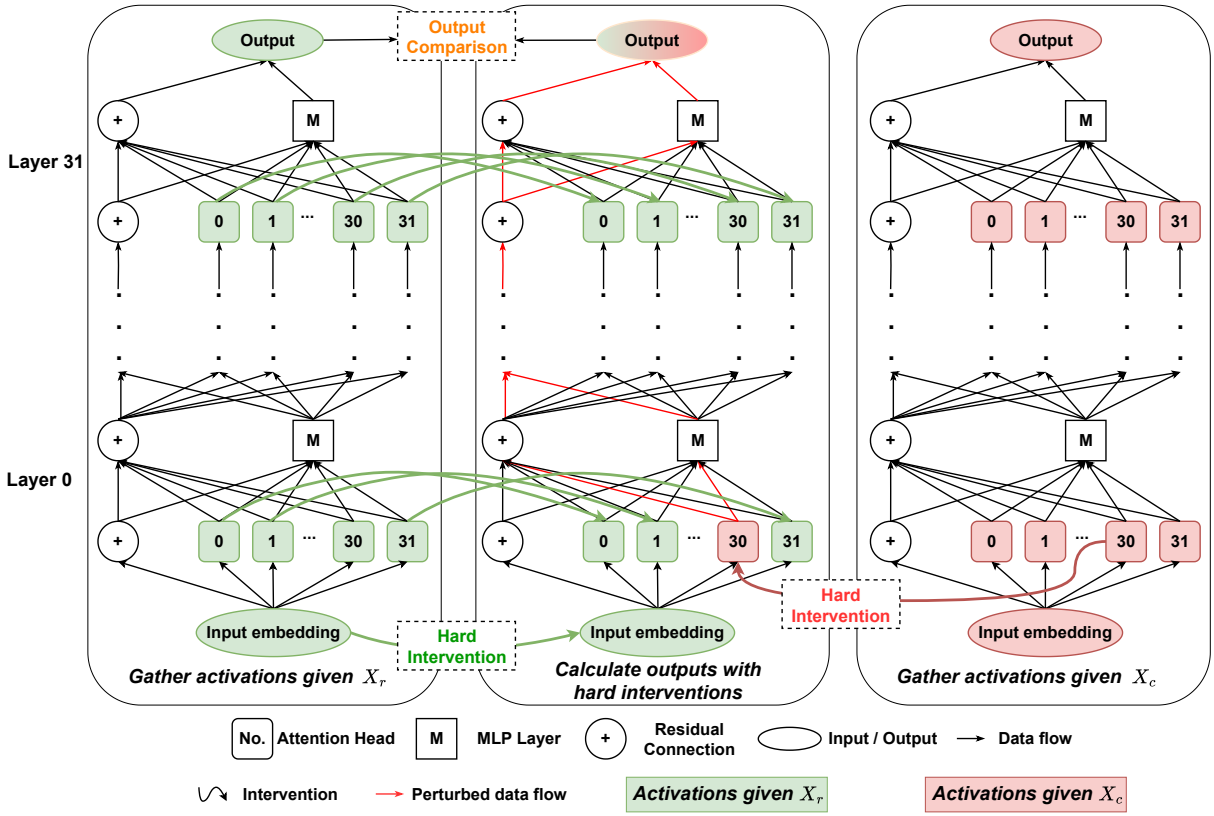
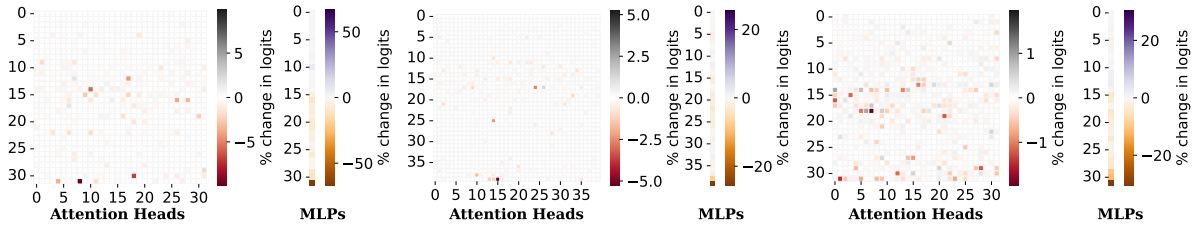
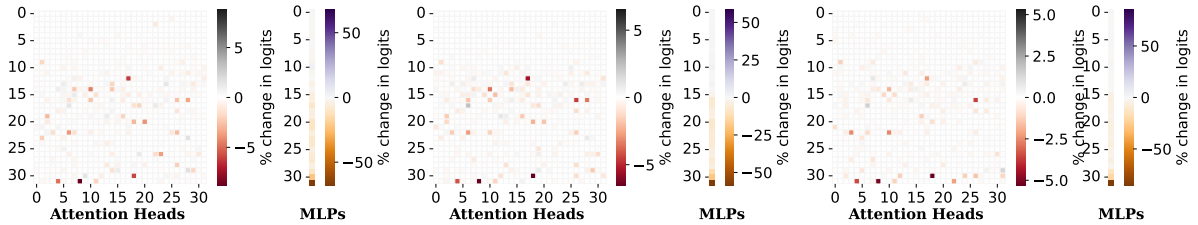


Figure 9: Illustration of the method "path patching". It measures the importance of the selected circuit (*i.e.*, the red lines that originate from Head 30 in Layer 0 to Output) for the transformer in completing the task on reference data.

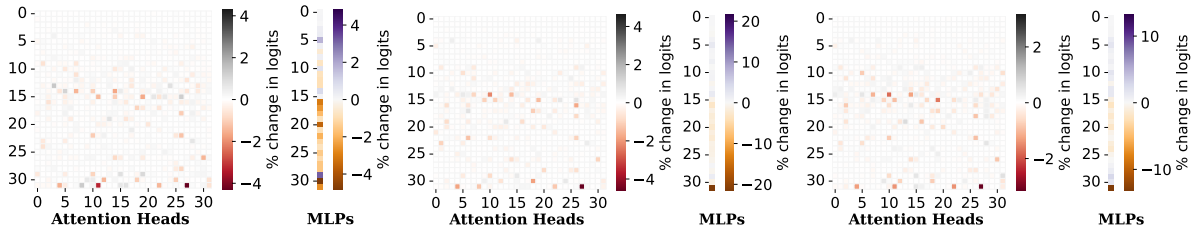


(a) LLaMA2-7B (b) LLaMA2-13B (c) Mistral-7B

Figure 10: Comparison of the results of path patching experiments on LLaMA2-7B, LLaMA2-13B, and Mistral-7B (Jiang et al., 2023) across Zh ⇒ En translation task. Each square at position (x, y) refers to the xth-head in the y-th layer. Red (Brown) squares denote heads (mlps) that have a positive impact on predicting the target token, while grey (purple) squares indicate heads (mlps) with a negative effect. For each head/MLP, a darker color indicates a larger logit difference from the original model before patching.

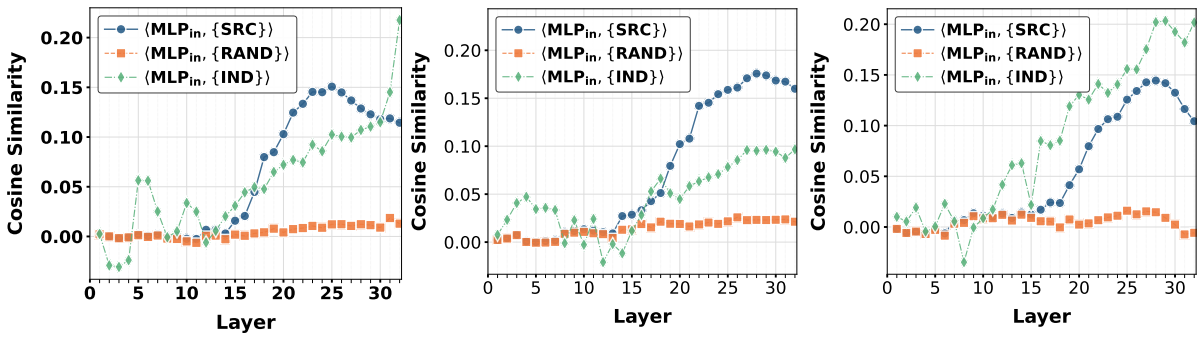


(a) En ⇒ De (b) En ⇒ Fr (c) En ⇒ Ru



(d) De ⇒ En (e) Fr ⇒ En (f) Ru ⇒ En

Figure 11: Importance of heads related to translation across different directions. Each square at position (x, y) refers to the x-th head in the y-th layer. Red (Brown) squares denote heads (MLPs) that have a positive impact on predicting the target token, while grey (purple) squares indicate heads (MLPs) with a negative effect.



(a) LLaMA2-7B (b) LLaMA2-13B (c) Mistral-7B

Figure 12: We investigate the projection of each MLP layer input (MLP_{in}) along the direction of the source language, indicator, and random English tokens ({SRC},{IND}, and {RAND}), respectively.

graph (DAG) (Wang et al., 2023), where nodes represent computational components (e.g., attention heads, MLP layers) and edges denote directional data flow between them. Mechanistic interpretability seeks to reverse-engineer neural networks into interpretable algorithms, leveraging computational circuits as a framework. A computational circuit is a subgraph of the model's computational graph M, comprising nodes (e.g., embeddings, attention heads) and edges (e.g., residual connections, projections) that collectively implement specific tasks, such as translation.

To analyze causal relationships within these circuits, we employ *path patching* (Goldowsky-Dill et al., 2023; Wang et al., 2023; Zhang et al., 2025). Algorithm 1 formalizes path patching as follows: for each component $c^{(j)}$, we (1) compute reference and counterfactual activations (H_f, H_{cf}), (2) create hybrid activations by replacing $c^{(j)}$'s activations with H_{cf} while keeping others at H_f, (3) compute logit differences (δ_j) between original and patched outputs, and (4) aggregate δ_j across the dataset to quantify $c^{(j)}$'s task-critical importance. This method isolates the causal effect between a *Sender* node (e.g., Head 30 in Layer 0) and a *Receiver* node (e.g., the output layer) by perturbing the Sender's activations with X_{cf} while freezing other nodes with X_f. . As illustrated in Figure 9, activations from all nodes are first recorded. A hard intervention replaces the Sender's activations with those from X_{cf}, , propagating the effect through paths $\mathcal{P}$ (residual connections and MLPs). Concurrently, other attention heads are frozen to X_f to isolate the Sender's impact. The resulting logits are compared to quantify the Sender's causal contribution: significant changes indicate critical paths for task execution.

Since residual streams and MLPs process tokens independently (Elhage et al., 2021), perturbing activations at the END token position suffices to measure effects on next-token prediction.

C More Analysis of Other LLMs and Translation Directions

Crucial Component Detection. Figure 10 extends key component identification to LLaMA2-13B and Mistral-7B. All three models exhibit sparse localization of translation-critical attention heads (e.g., 17.24, 16.0) in middle layers, despite architectural differences (e.g., LLaMA2-13B's 40 layers with 40 heads per layer).

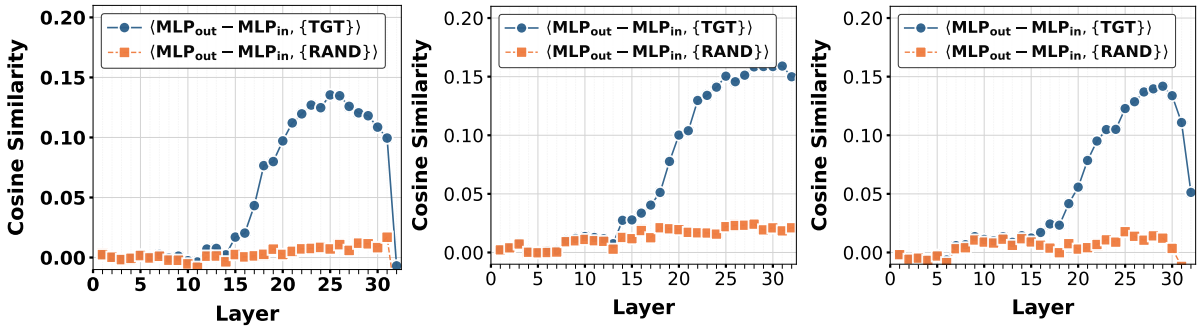
Figure 11 illustrates the detection results for bidirectional translation directions (En ⇒ X and X ⇒ En). While the multi-token nature of English tokens results in fewer prominent detection instances, the findings remain consistent with the earlier analysis in Section §5.1. Together, these observations support the conclusion that translation mechanisms utilize a sparse subset of attention heads, which are language-agnostic, thereby underscoring their generalization capacity.

Analysis of Crucial MLPs. Figures 12 and 13 reveal consistent MLP dynamics across models. For MLP input/{SRC},{IND} similarities, trends follow ascending-descending phases with inflection points at layers (13-18-28) for LLaMA2-7B, (13-18-35) for LLaMA2-13B, and (13-20-28) for Mistral-7B. Similarly, $MLP_{out} - MLP_{in}$ and target token {TGT} similarities show stabilization-to-increase patterns with identical inflection layers. This synchronization across models indicates a shared computation mechanism: attention heads initiate translation processing, which MLPs subsequently refine. These results demonstrate robustness across architectures and scales.

Cross-Lingual Bridge Translation. We extend our analysis to non-English pairs (e.g., French/Japanese Chinese) by examining token-level dynamics. As shown in Figure 14, similarity trends between $MLP_{out} - MLP_{in}$ representations and cross-lingual embeddings align with the bridge-translation hypothesis: in layers 15–24, English-centric latent representations dominate across LLaMA2-13B and Mistral-7B, with similarity declining sharply in layers 25–32. This reinforces the observed paradigm where LLMs internally map source languages to English-like representations before generating target outputs, corroborating findings in multilingual latent alignment studies (Wendler et al., 2024; Zhao et al., 2024b). The consistency across both architectures underscores the generality of English's intermediary role.

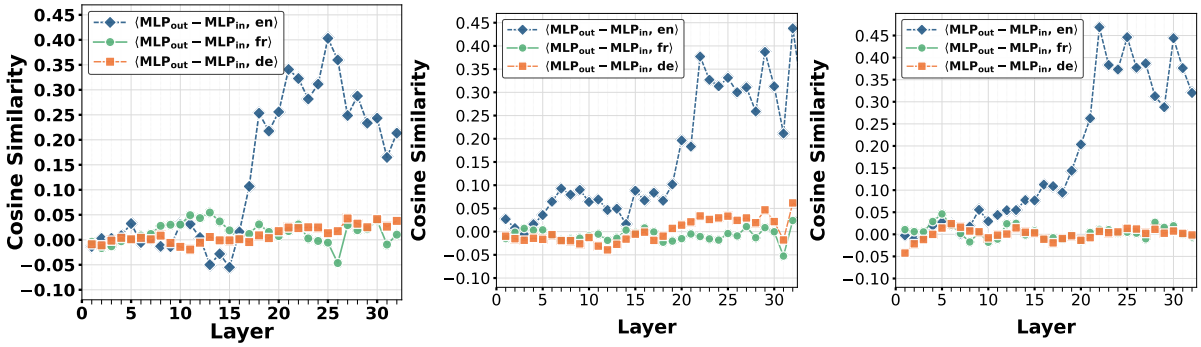
D Experimental Setup Details

Following the gradient rescaling method proposed by (Yu et al., 2025), gradients are adjusted by a factor of $\frac{H}{h}$, where H is the total number of attention heads in a layer and h represents the updated heads in the same layer. For model fine-tuning, we use Llama2-7B and Llama2-13B with



(a) LLaMA2-7B (b) LLaMA2-13B (c) Mistral-7B

Figure 13: We investigate the projection of each MLP layer ($MLP_{out} - MLP_{in}$) along the direction of the target language, and random English tokens ({TGT} (i.e., right translation), and {RAND} (i.e., wrong translation)), respectively.



(a) LLaMA2-7B (b) LLaMA2-13B (c) Mistral-7B

Figure 14: We investigate the projection of each MLP layer ($MLP_{out} - MLP_{in}$) along the direction of the different languages.

a learning rate of 2×10^{-5}, a batch size of 128, and train for 2 epochs. The warm-up ratio is set to 0.02, and weight decay is configured at 0.1. All experiments are conducted on a cluster of 8 NVIDIA A100 80 GB GPUs.

E Comparison Experimental Results on More LLMs

We investigate whether our method generalizes to larger LLMs (Llama-2-13B) and diverse architectures (Mistral-7B). As shown in Tables 8 and 9, Targeted SFT exhibits three consistent advantages across LLMs: (1) Enhanced translation performance, particularly in X En, surpassing Full SFT and significantly outperforming Random SFT; (2) Generalization preservation, maintaining baseline non-translation task performance unlike Full SFT; (3) Training efficiency, modifying fewer than 5% of parameters and reducing training time by 50% compared to Full SFT.

Models	Train Speed	Tuned Params.	Translation Tasks			Generic Tasks	
			En⇒Zh	En⇒De	En⇒Ru	MMLU	Commonsense Reasoning
			BLEU↑/COMET↑/BLEURT↑			Acc.	Acc.
LLaMA2-7B	-	-	17.0/74.1/55.9	13.0/64.2/49.1	12.8/70.5/52.4	45.9	55.3
+ Full SFT	17sam./sec.	6.7B	30.3/80.7/62.9	27.9/78.3/63.7	19.5/80.0/63.2	40.2	50.0
+ Targeted SFT	33sam./sec.	0.27B	27.6/80.0/62.5	27.6/78.4/63.8	20.1/80.4/63.6	46.2	56.0
+ Random SFT	33sam./sec.	0.27B	26.4/79.3/61.6	22.7/76.2/60.3	15.8/77.9/60.7	46.1	55.2
LLaMA2-13B	-	-	23.0/77.5/59.1	17.1/67.7/52.8	15.6/72.9/55.1	55.1	58.4
+ Full SFT	12sam./sec.	13.0B	32.8/81.8/64.4	29.8/80.0/65.8	20.7/81.6/65.0	53.7	56.4
+ Targeted SFT	28sam./sec.	0.32B	33.4/82.2/64.8	30.1/80.1/65.9	21.3/81.8/65.3	54.9	58.1
+ Random SFT	28sam./sec.	0.32B	28.8/80.6/63.3	24.6/78.3/62.9	17.3/80.0/62.8	55.0	58.2
Mistral-7B	-	-	13.7/68.0/49.6	15.6/63.1/49.3	11.2/65.1/48.1	62.7	59.2
+ Full SFT	17sam./sec.	6.7B	31.1/80.6/63.4	26.5/77.4/62.8	19.6/79.5/62.5	43.0	40.8
+ Targeted SFT	33sam./sec.	0.27B	31.9/82.0/65.1	26.3/78.0/63.2	20.5/79.9/63.1	62.5	59.1
+ Random SFT	33sam./sec.	0.27B	27.5/79.5/61.6	22.2/75.5/59.8	15.6/77.4/60.5	62.4	59.2

Table 8: The evaluation results of **En⇒X** translation (average WMT23 and WMT24 evaluation results) and generic tasks of different SFT strategies.

Models	Train Speed	Tuned Params.	Translation Tasks			Generic Tasks	
			En⇒Zh	En⇒De	En⇒Ru	MMLU	Commonsense Reasoning
			BLEU↑/COMET↑/BLEURT↑			Acc.	Acc.
LLaMA2-7B	-	-	15.6/73.1/56.6	24.8/76.8/62.1	20.2/73.8/60.3	45.9	55.3
+ Full SFT	17sam./sec.	6.7B	20.4/78.7/63.9	35.4/83.4/70.7	25.8/79.8/67.6	42.6	50.2
+ Targeted SFT	33sam./sec.	0.27B	21.7/79.1/64.4	37.1/83.7/71.4	27.8/80.3/68.4	46.0	55.7
+ Random SFT	33sam./sec.	0.27B	16.9/76.9/61.1	32.5/81.6/68.1	23.7/78.2/65.3	45.9	54.9
LLaMA2-13B	-	-	17.3/74.0/57.8	27.0/78.0/63.8	22.2/74.9/61.5	55.1	58.4
+ Full SFT	12sam./sec.	13.0B	22.4/79.5/65.3	36.9/84.0/71.6	27.8/80.8/68.9	50.0	55.3
+ Targeted SFT	28sam./sec.	0.32B	23.6/80.5/66.5	38.3/84.7/72.7	29.7/81.5/69.3	54.9	58.1
+ Random SFT	28sam./sec.	0.32B	19.0/78.1/63.1	34.2/81.8/68.9	25.3/79.3/66.6	55.5	58.8
Mistral-7B	-	-	16.9/74.3/58.1	26.6/77.9/63.9	22.6/75.3/62.5	62.7	59.2
+ Full SFT	17sam./sec.	6.7B	19.7/78.4/63.1	32.0/82.2/69.0	24.0/78.7/66.2	40.3	50.3
+ Targeted SFT	33sam./sec.	0.27B	21.2/79.2/64.3	33.7/83.0/70.2	26.4/79.6/66.4	62.9	59.1
+ Random SFT	33sam./sec.	0.27B	16.8/77.1/61.1	29.3/80.6/66.8	21.4/77.1/63.9	62.5	59.3

Table 9: The evaluation results of **X⇒En** translation (average WMT23 and WMT24 evaluation results) and generic tasks of different SFT strategies.