# Length Representations in Large Language Models

**Sangjun Moon[1†], Dasom Choi[1†], [*]Jingun Kwon[1],**
**Hidetaka Kamigaito[2], Manabu Okumura[2],**
[1]Chungnam National University, [2]Nara Institute of Science and Technology (NAIST),
[3]Institute of Science Tokyo
{sangjunmoon,dasomchoi}@o.cnu.ac.kr
jingun.kwon@cnu.ac.kr
kamigaito.h@is.naist.jp
oku@pi.titech.ac.jp

## Abstract

Large language models (LLMs) have shown remarkable capabilities across various tasks, that are learned from massive amounts of text-based data. Although LLMs can control output sequence length, particularly in instruction-based settings, the internal mechanisms behind this control have been unexplored yet. In this study, we provide empirical evidence on how output sequence length information is encoded within the internal representations in LLMs. In particular, our findings show that multi-head attention mechanisms are critical in determining output sequence length, which can be adjusted in a disentangled manner. By scaling specific hidden units within the model, we can control the output sequence length without losing the informativeness of the generated text, thereby indicating that length information is partially disentangled from semantic information. Moreover, some hidden units become increasingly active as prompts become more length-specific, thus reflecting the model's internal awareness of this attribute. Our findings suggest that LLMs have learned robust and adaptable internal mechanisms for controlling output length without any external control.

## 1 Introduction

Large language models (LLMs) have gained considerable attention in recent years for their remarkable task-solving capabilities (Ouyang et al., 2022; Wei et al., 2022; Bubeck et al., 2023). LLMs are trained to predict the next token in a sequence. They can produce coherent and informative text, which demonstrates their implicit understanding of diverse linguistic structures (Tenney et al., 2019; Niu et al., 2022; Beguš et al., 2023). Furthermore, they also learn when to stop generating text to ensure that the output adheres to appropriate length constraints (Juseon-Do et al., 2024). Controlling output sequence length in LLMs is crucial for real-world applications, such as text summarization (Liu et al., 2018; Makino et al., 2019; Liu et al., 2022; Kwon et al., 2023), machine translation (Wu et al., 2016; Murray and Chiang, 2018; Zhuocheng et al., 2023), knowledge QA, and dialogue generation (Liu et al., 2020; Gupta et al., 2021), that necessitate fitting content within specified length limits without losing informativeness. Therefore, the number of studies attempting to improve length controllability has increased drastically (Shen et al., 2023; Jie et al., 2024; Yuan et al., 2024).

Based on advancements in instruction-based LLMs, it is observed that injecting constraints into prompts can further effectively control output length without requiring model modifications (Juseon-Do et al., 2024). However, these prompt engineering methods mainly focus on external controls, and it has not been explored yet how LLMs internally encode and constrain output sequence length. Understanding these internal mechanisms is critical for achieving precise length control, while enhancing the interpretability and robustness of LLMs in generation-based systems. Herein, we aim to investigate how output sequence length information is encoded within the internal representations of general transformer architectures. Specifically, we first investigate which components within LLM transformer layers contribute to length control. Our findings reveal that the outputs from multi-head attention mechanisms in the lower layers play a key role in determining and controlling output sequence length in a tunable and disentangled manner.

For this work, we utilize a sentence summarization task, which often requires adherence to desired summary lengths, and employ models from the Llama (Meta, 2024), Phi-3 (Abdin et al., 2024), and Qwen-2.5 (Qwen et al., 2025) families.

We empirically demonstrate, based on human

[†] Authors contributed equally. [*] Corresponding author.
Our code is available at https://github.com/Mcat00/gilab_length.



*Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 19775–19793
November 4-9, 2025 

evaluations, that we can adjust output length during generation without losing the informativeness of texts by scaling specific hidden units within the outputs from the lower layers of multi-head attention mechanisms. For instance, multiplying certain hidden units with negative numbers results in longer text, while multiplying them by positive numbers generates more concise texts without losing informativeness. Furthermore, certain hidden units related to length information show increasing activity as prompts become more specific regarding length constraints. These units appear to be directly involved in controlling output length, indicating that LLMs have learned to process length-related information as a distinct feature, partially disentangled from other semantic information. Moreover, we find that the same highly activated hidden units are consistently involved in length control even after fine-tuning, regardless of length constraints in prompts (Dai et al., 2023). Our code is available at https://github.com/Mcat00/gilab_length.

## 2 Related Work

**Large Language Models.** In recent years, LLMs have achieved considerable success due to their remarkable task-solving abilities, specifically in zero-shot settings (Radford et al., 2019; Brown et al., 2020). LLMs are broadly categorized into open and closed models. The open models, such as the Llama or Phi family, offer flexible access to modify their architectures, while the closed models, such as ChatGPT,[1] have demonstrated remarkable reasoning abilities in various natural language processing tasks (Jiao et al., 2023; Peng et al., 2023; Laskar et al., 2023; Ye et al., 2023; Xie et al., 2023, 2024). Recent studies have focused on finding better methods to prompt LLMs (Zhou et al., 2022; Kojima et al., 2023; Zhou et al., 2023).

**Mechanistic Interpretability.** Due to increasing interest in investigating the internal mechanisms of deep neural networks (Räuker et al., 2023), significant attempts have been made to understand LLMs with a focus on models like BERT (Tenney et al., 2019; Rogers et al., 2020; Niu et al., 2022), GPT (Hanna et al., 2023), and even multimodal models (Goh et al., 2021). For instance, (Gurnee and Tegmark, 2024) showed that, when handling various prompts, LLMs learn linear representations of space and time across multiple scales, that show robustness. They also showed that next token prediction can be changed simply by disentangling hidden units related to time. (Heinzerling and Inui, 2024) introduced directions that encode numeric properties in an interpretable manner; hence, by disentangling these representations, LLM prediction can change accordingly. There have been attempts to investigate how in-context learning with LLMs behaves similar to explicit fine-tuning for better understanding them (Dai et al., 2023). Early efforts to investigate how neural networks treat length information have focused on memory cell networks in LSTMs, as they recursively encode and decode sequences, though they failed to find single units related to length information (Shi et al., 2016).

**Length Controllable Summarization.** Text summarization aims to produce a concise summary from an original text by retaining informative contents (Liu et al., 2018; Takase and Okazaki, 2019; Li et al., 2020; He et al., 2022). As the summarization often requires additional constraints such as a desired summary length, previous studies have focused on learning length-specific parameters (Kikuchi et al., 2016; Schumann et al., 2020; Ghalandari et al., 2022), injecting direct constraints (Takase and Okazaki, 2019; Makino et al., 2019), or splitting the training dataset into specific length ranges (He et al., 2022). Recently, (Juseon-Do et al., 2024) considered in-context learning and demonstrated that LLMs can control output sequence length through "length priming". This method involves injecting more length-specific information into prompts, thereby allowing the model to adjust output sequence length without modifying model architectures or learning parameters. (Jie et al., 2024) considered length control types such as greater/smaller than a value with exhaustive model modifications by reinforcement learning.

To the best of our knowledge, this study is the first attempt to interpret how length information is encoded in LLMs and demonstrate how length-specific information is partially disentangled from semantic information. Furthermore, by comparing various length-specific prompts, we investigate how in-context learning and fine-tuning can influence the internal representations of LLMs. Finally, we demonstrate how disentangling length-specific hidden units can adjust output sequence length without losing informativeness.

[1] https://chat.openai.com/

| Constraint | Instruction |
|---|---|
| No-constraint | Sentence:{src}The sentence without the less important tokens would be: |
| Length | Sentence:{src}The sentence without the less important {del} tokens would be: |
| Priming | Sentence that consists of {src len} tokens:{src}The sentence that consists of {keep} tokens without the less important {del} tokens would be: |

Table 1: Instruction formats. "src" indicates the placeholder for a source sentence, "del" denotes the placeholder for the number of deleted tokens, and "keep" and "src len" denote additional length information.

## 3 Finding Length Representations

Our goal is to understand whether and how length representations are encoded in LLMs when using various length-constraint prompts. For this, we extracted outputs from different components and layers of transformer architectures during text generation. We then applied regression to predict the generation time steps from these hidden states.

**Summarization Dataset.** We used the **Google** sentence summarization dataset[2] (Filippova and Altun, 2013) in an instruction-based format, following previous work[3] (Juseon-Do et al., 2024) because recent studies on length control still faces challenges in managing it with LLMs (Jie et al., 2024; Yuan et al., 2024). Table 1 presents the instruction templates. As can be seen, in the **No-constraint** setting, the model summarizes a given sentence without considering a desired length, while in the **Length** setting, it summarizes the sentence with a specific desired length (Fetahu et al., 2023). The **Priming** setting further considers more specific length information, such as the length of the given sentence and the number of tokens to keep (Juseon-Do et al., 2024). The dataset includes 200k training, 1k validation, and 1k test pairs, where the average compression ratio in the test dataset is 0.45. Length-specific prompts use ground-truth summary lengths.

### 3.1 Models and Methods

**Models.** We performed our experiments using the Llama family of pre-trained instruction-based LLMs, which range from 1B to 70B parameters (Touvron et al., 2023; Grattafiori et al., 2024), the Phi-3 family of mini (Abdin et al., 2024), and the Qwen-2.5 family (Qwen et al., 2025). Additionally, we considered how 4- and 8-bit quantizations influence length representations in LLMs. Furthermore, we fine-tuned an LLM on **Google** using QLoRA (Dettmers et al., 2023) to study how fine-tuning with length constraint prompts affect length-related internal representations. We followed previous work that incorporates such prompts to enhance output length control (Juseon-Do et al., 2024). We used greedy decoding in all experiments to eliminate randomness in generation.

**Gathering Model States.** In the transformer, an input sentence $\mathbf{S} = \{s_1, s_2, \ldots, s_n\}$ was first converted into vector embeddings, after which learned positional embeddings were added to form $\mathbf{S_{emb}} = \{s_1^e, s_2^e, \ldots, s_n^e\}$. These embeddings were then normalized using layer normalization, expressed as $S_{\text{norm}} = \text{LN}(\mathbf{S}_{\text{emb}})$. Then, they were computed through query ($\mathbf{W_Q}$), key ($\mathbf{W_K}$), and value ($\mathbf{W_V}$) matrices, and were fed into the transformer layers as follows:

$$\text{MH}(Q, K, V) = \text{Concat}(\text{h}_1, \ldots, \text{h}_h)W_O, \quad (1)$$

$$S_{\text{attn}} = \mathbf{S}_{\text{emb}} + \text{MH}(S^Q_{\text{norm}}, S^K_{\text{norm}}, S^V_{\text{norm}}), \quad (2)$$

$$S_{\text{ffn}} = \text{ReLU}(\text{LN}(S_{\text{attn}})W_1 + b_1)W_2 + b_2, \quad (3)$$

$$S_{\text{out}} = S_{\text{attn}} + S_{\text{ffn}}, \quad (4)$$

where each $\text{h}_i = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$ indicates a self-attention operation.

We considered four outputs from the transformer layers: (1) multi-head attention, (2) multi-head attention combined with the original embeddings, (3) the outputs of feed-forward networks, and (4) an integration of (2) and (3). Each output represents a distinct level of encoded information derived from the original input sentence $\mathbf{S}$. We conducted sentence summarization using prompts in three different settings: No-constraint, Length, and Priming. For each setting, we investigated these four outputs for each layer. During token generation, we saved each output with its corresponding numeric time step value, excluding the input token prompts (Kaplan et al., 2024). For instance, we saved $n$ with its corresponding output when the model generated

[2] https://github.com/google-research-datasets/sentence-compression.git

[3] https://github.com/JuseonDo/InstructCMP

the $n$-th token. Appendix A provides further details of data for predicting time steps from hidden states.

**Neural Network Regression.** To find evidence of length representations in LLMs, we applied a standard technique to predict a target label associated with labeled input data (Shi et al., 2016), specifically, $\mathbf{X} \in \mathbb{R}^{m \times d_{\text{model}}}$, where $m$ refers to the number of data, $d_{\text{model}}$ is the dimensionality of a model's hidden states, and $\mathbf{Y}$ is a target that contains the generation time step as a numeric value for each corresponding $\mathbf{X}$. We used a two-layer neural network with a hidden layer of 100 neurons to predict $\hat{\mathbf{Y}} = \mathbf{W}_2(\text{ReLU}(\mathbf{W}_1\mathbf{X} + \mathbf{b}_1)) + \mathbf{b}_2$. By investigating how well the model can predict the generation time step, we can gain insights into how length representations are encoded within the LLM's hidden states. To assess how well the time step can be predicted from its corresponding hidden state in LLMs, we considered the coefficient determination, $R^2$, as a standard regression metric to evaluate the overall performance. Appendix B provides details of hyper-parameters and settings.

## 4 Length Representations in LLMs

We explored which transformer layers and outputs contain length information, how length-specific prompts and quantization affect length representations, and the impact of fine-tuning on LLMs.

**Layer-wise Analysis for Length Representations.** Figure 1 shows the variation of $R^2$ for outputs from a transformer layer corresponding to Equations (1), (2), (3), and (4). In the second layer, the outputs of Equation (1), which indicates the attention mechanism, show a stronger correlation with the length representations than the outputs from Equations (2), (3), and (4) for all prompts. Length representations decrease through LLM layers during token generation but increase in the final layer based on the attention outputs of Equation (1). This indicates that the LLM captures length representations in the early stages, similar to how they capture semantic representations (Niu et al., 2022). As such, the increase in length representations in the final layer indicates that the model may revisit this information to reinforce positional context.

**Influence of Length-specific Prompts.** Table 2 shows the results of $R^2$ for outputs, which include Llama and Phi LLMs with a 4-bit quantization setting. The results reveal that the attention output consistently has higher $R^2$ scores than the other outputs, particularly in the second layer for the Llama- and Phi-3 families, regardless of model sizes. However, we observed a notable decrease in performance in the first layer, particularly in the attention residual. This indicates that the initial input sequence embeddings do not effectively contain length information; however, these representations progressively accumulate it through the layers. Although the length-specific prompting method (Priming) can precisely control output sequence length (Juseon-Do et al., 2024), it does not increase the $R^2$ when using all hidden units for prediction. However, when we fine-tuned the models, we found that for every model, regardless of the prompts used, the $R^2$ scores were improved.

**Quantization on Length Representations.** Table 3 shows the results with 8-bit and full-precision settings. The results are similar to those obtained with 4-bit quantization, wherein length representations are more prominently encoded in the attention outputs from the second layer than the other outputs. This indicates that whether 4- or 8-bit quantization is applied does not significantly affect the LLMs' capabilities to encode length representations. Therefore, the attention mechanism of the second layer consistently captures length representations across different precision levels even for different models with varying sizes.

## 5 Disentangling Length Representations

The previous section explored which components and layers contain length representations for output sequence length with varying prompts. While the second attention layer has a strong correlation with length representations, this does not indicate which hidden units are actually responsible for controlling the output sequence length. Thus, the specific hidden units must be identified for a better understanding of LLMs' length control.

**Do Length-specific Prompts Affect Inner Length Representations?** We additionally trained separate neural network regression models on each single hidden unit from the second layer of the attention outputs in Llama-2-13B-Chat, which has a total of 5,120 hidden units. Table 4 shows the results for hidden units of the top-5 highest $R^2$ scores. Compared to the No-constraint and Length prompting methods, length-related hidden units become more active in representing length information when we used more length-specific prompts Priming. In the zero-shot setting, No-constraint and Length prompts share similar top-5 hidden units for the

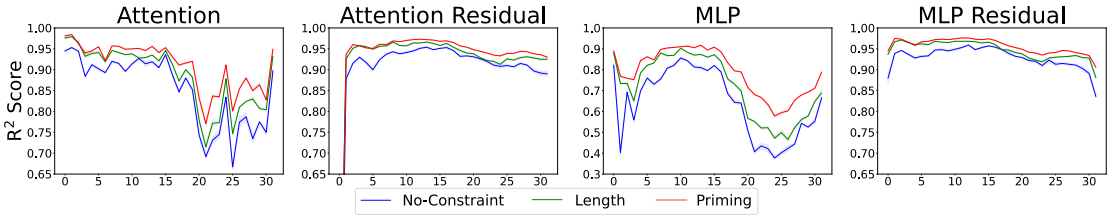


Figure 1: Average $R^2$ scores and standard errors with five runs for outputs of four different types of transformer layers using Llama-2-7B-Chat.

length representation, while the Priming prompt activates different units, showing a shift in length capture and stronger activation in top-$k$ units. After fine-tuning, the hidden units for the length representation became nearly identical across prompting methods, because the model learned the precise length control. Interestingly, the same top-3 hidden units are activated with the Priming prompt in the zero-shot and fine-tuning settings. This finding shows that specific length-related units consistently activate during Priming, thus guiding LLMs in output length control and revealing in-context learning as implicit fine-tuning (Dai et al., 2023).

**Does Scaling Length Representations Affect Model-Generated Text?** Since identified length-related units do not guarantee their actual involvement in length representations within LLMs (Sajjad et al., 2022; Belinkov, 2022), we investigate the effect of scaling these representations on model-generated text. Specifically, we disentangled the top-$k$ and smallest-$k$ activated hidden units in the second layer's multi-head attention by scaling them with positive or negative values. The scaling was applied to all output token positions except the input prompts.[4] This approach demonstrates that the identified units contribute to length representations in LLMs and they are partially disentangled from semantic representations.

**Evaluation Metrics.** We used Rouge-L (R-L) (Lin, 2004) to evaluate the informativeness of the summarized sentences. To evaluate length control performance, we used $\Delta$CR, the arithmetical difference between model-generated and gold compression ratios. The compression ratio is the number of summary tokens divided by the number of source tokens. A $\Delta$CR close to zero indicates that the generated summaries have a compression ratio similar to the gold summaries, with higher values indicating longer summaries and lower values indicating shorter ones. Thus, deviations in $\Delta$CR from zero often lead to lower R-L scores due to reduced alignment with the gold summary (Makino et al., 2019).

**Results in Zero-Shot Settings.** Figure 2 (a) presents the results of applying negative or positive scaling factors to the top-$k$ and smallest-$k$ activated hidden units in zero-shot settings. When using more length-specific prompts of Priming, we observed more consistent changes of $\Delta$CR with modifying only the top-1 hidden unit. We think this is because Priming contains more highly activated hidden units related to length representations than No-constraint and Length. Thus, the highly activated length-related units provide better length representations and length controllability, as shown in Table 4. Additionally, the output sequence length changes according to increases in the scaling factor. While multiplying positive and negative values enables the LLM to produce shorter and longer summaries, respectively, than the original hidden units, particularly in the Priming prompt, in the No-constraint and Length prompts, the LLM does not generate shorter summaries even when positive scaling values were applied.

As for R-L scores, disentangling the top-$k$ units improves performance, particularly in the Priming prompt due to improved length alignment with the gold summary. This finding indicates that adjusting the most highly activated length-related units not only controls length but also enhances the informativeness of the generated text. However, when we applied large scaling factors, such as -10 or 10, the R-L scores slightly decrease when the No-constraint and Length prompts were used. In comparison, for Priming, which is more length-specific prompts, continues to improve performance even when we applied a large scaling factor of 10. Disentangling the smallest-$k$ units does not lead to significant changes in output sequence length, thus indicating that these units are less involved in encoding length information. For selected smallest-$k$ units, the individual $R^2$ scores are nearly 0.

**Results in Fine-Tuning Settings.** Figure 2 (b)

[4]We also explored other scaling methods, such as applying scaling only to the first output token, but found they were ineffective at controlling output lengths.

| Model | Prom. | Layer type | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Attn Out | | | Attn Residual | | | MLP Out | | | MLP Residual | | |
| | | F | S | L | F | S | L | F | S | L | F | S | L |
| Llama-2 -7B | [1] | 0.94 | **0.95** | 0.88 | 0.00 | 0.90 | 0.89 | 0.84 | 0.70 | 0.67 | 0.90 | 0.94 | 0.85 |
| | [2] | 0.98 | **0.99** | 0.93 | 0.11 | 0.94 | 0.93 | 0.89 | 0.77 | 0.70 | 0.95 | 0.97 | 0.89 |
| | [3] | 0.98 | **0.99** | 0.95 | 0.11 | 0.94 | 0.94 | 0.89 | 0.77 | 0.78 | 0.95 | 0.98 | 0.92 |
| Llama-2 -13B | [1] | 0.95 | **0.96** | 0.93 | 0.08 | 0.93 | 0.92 | 0.90 | 0.83 | 0.74 | 0.93 | 0.95 | 0.89 |
| | [2] | **0.94** | **0.94** | 0.92 | 0.10 | 0.92 | 0.92 | 0.89 | 0.81 | 0.75 | 0.91 | **0.94** | 0.91 |
| | [3] | **0.99** | **0.99** | 0.91 | 0.17 | 0.96 | 0.92 | 0.92 | 0.81 | 0.72 | 0.97 | 0.98 | 0.89 |
| Llama-2 -13B (finetuned) | [1] | **0.99** | **0.99** | 0.88 | 0.17 | 0.97 | 0.92 | 0.93 | 0.81 | 0.74 | 0.98 | 0.98 | 0.91 |
| | [2] | **0.99** | **0.99** | 0.87 | 0.21 | 0.97 | 0.93 | 0.92 | 0.83 | 0.78 | 0.98 | 0.98 | 0.91 |
| | [3] | **0.99** | **0.99** | 0.90 | 0.16 | 0.96 | 0.93 | 0.92 | 0.85 | 0.83 | 0.97 | 0.98 | 0.92 |
| Llama-2 -70B | [1] | 0.97 | **0.99** | 0.95 | 0.16 | 0.93 | 0.92 | 0.83 | 0.81 | 0.82 | 0.95 | 0.98 | 0.92 |
| | [2] | 0.97 | **0.99** | 0.94 | 0.17 | 0.92 | 0.93 | 0.87 | 0.84 | 0.80 | 0.95 | 0.98 | 0.92 |
| | [3] | **0.98** | 0.97 | 0.91 | 0.18 | 0.91 | 0.89 | 0.82 | 0.76 | 0.78 | 0.94 | 0.95 | 0.88 |
| Llama-3 -8B | [1] | 0.96 | **0.98** | 0.91 | 0.20 | 0.86 | 0.91 | 0.70 | 0.74 | 0.78 | 0.88 | 0.95 | 0.88 |
| | [2] | 0.96 | **0.97** | 0.93 | 0.16 | 0.88 | 0.93 | 0.72 | 0.75 | 0.79 | 0.90 | 0.96 | 0.89 |
| | [3] | 0.97 | **0.98** | 0.94 | 0.24 | 0.87 | 0.94 | 0.73 | 0.76 | 0.87 | 0.89 | 0.95 | 0.92 |
| Phi-3 -mini -4k | [1] | 0.93 | **0.97** | 0.91 | 0.07 | 0.80 | 0.91 | 0.61 | 0.66 | 0.55 | 0.84 | 0.95 | 0.86 |
| | [2] | 0.94 | **0.97** | 0.92 | 0.04 | 0.80 | 0.92 | 0.65 | 0.67 | 0.56 | 0.82 | 0.94 | 0.86 |
| | [3] | 0.93 | **0.97** | 0.89 | 0.07 | 0.77 | 0.90 | 0.48 | 0.63 | 0.58 | 0.80 | 0.95 | 0.84 |
| Phi-3.5 -mini | [1] | 0.90 | **0.96** | 0.90 | 0.06 | 0.71 | 0.91 | 0.53 | 0.58 | 0.63 | 0.75 | 0.93 | 0.83 |
| | [2] | 0.90 | **0.96** | 0.89 | 0.07 | 0.71 | 0.90 | 0.56 | 0.59 | 0.62 | 0.74 | 0.93 | 0.84 |
| | [3] | 0.63 | **0.73** | 0.55 | 0.05 | 0.40 | 0.63 | 0.30 | 0.44 | 0.44 | 0.45 | **0.73** | 0.60 |
| Qwen-2.5 -3B | [1] | 0.86 | **0.98** | 0.75 | 0.15 | 0.68 | 0.77 | 0.45 | 0.87 | 0.46 | 0.85 | 0.95 | 0.71 |
| | [2] | 0.84 | **0.97** | 0.77 | 0.15 | 0.64 | 0.78 | 0.42 | 0.84 | 0.57 | 0.83 | 0.95 | 0.75 |
| | [3] | 0.82 | **0.96** | 0.72 | 0.17 | 0.65 | 0.73 | 0.39 | 0.85 | 0.47 | 0.82 | 0.93 | 0.67 |
| Qwen-2.5 -7B | [1] | 0.92 | **0.99** | 0.82 | 0.14 | 0.84 | 0.88 | 0.65 | 0.79 | 0.60 | 0.92 | 0.98 | 0.83 |
| | [2] | 0.93 | **0.99** | 0.81 | 0.15 | 0.85 | 0.86 | 0.69 | 0.80 | 0.60 | 0.92 | 0.98 | 0.82 |
| | [3] | 0.96 | **0.99** | 0.80 | 0.32 | 0.92 | 0.87 | 0.76 | 0.85 | 0.66 | 0.96 | 0.98 | 0.83 |

Table 2: Average $R^2$ scores with five runs for different models with constraint prompt types from the first (F), second (S), and last (L) layers in LLMs. The standard errors are nearly zero. Prom. indicates the prompting method used, and [1], [2], and [3] indicate No-constraint, Length, and Priming prompts.

shows the results of applying negative or positive scaling factors to the top-*k* and smallest-*k* activated hidden units in fine-tuning settings. In contrast to the previous zero-shot settings, we obtained more stable results for all prompts when we disentangled the hidden units. While multiplying positive scaling values results in generating shorter summaries, multiplying negative values produces longer summaries. This is because fine-tuning has strengthened the LLM's reliance on the top-*k* length-related units for precise length control. While large scaling factors lead to greater changes in $\Delta$CR and R-L for the Priming prompt, higher overall R-L scores are maintained. The decrease in R-L occurs because the Priming prompt in fine-tuning settings already achieves precise control, resulting in a $\Delta$CR close to zero. Applying large scaling factors causes deviations from zero and leads to a decline in R-L scores. Disentangling the smallest-*k* has minimal impact on sequence length among all prompts. Specifically, there are no significant changes in output sequence length when the smallest-*k* hidden units were modified.

**Results Using Different LLMs.** Figure 3 shows the results using different LLMs. While Llama-2-7B-Chat and Llama-2-13B-Chat used standard multi-head attentions during their pre-training steps, other LLMs employed grouped-query attentions (Grattafiori et al., 2024; Abdin et al., 2024). When we scaled the top-*k* hidden units by multiplying scaling factors in the LLMs which employed the multi-head attentions, we observed variations in output length under zero-shot settings while scaling the smallest-*k* hidden units did not impact length control during generation. In contrast, scaling the top-*k* hidden units in the LLMs that employed the grouped-query attentions did not effectively control output length. However, after fine-tuning these LLMs, we found that disentangling the top-*k* hidden units effectively controls output length. Addi-

| Model | Prom. | Layer type | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Attn Out | | | Attn Residual | | | MLP Out | | | MLP Residual | | |
| | | First | Second | Last | First | Second | Last | First | Second | Last | First | Second | Last |
| Llama-2 -13B | [1] | 0.58/0.55 | 0.70/0.68 | **0.76/0.74** | 0.01/0.04 | 0.58/0.56 | 0.73/0.72 | 0.59/0.58 | 0.56/0.57 | 0.61/0.59 | 0.58/0.55 | 0.66/0.64 | 0.70/0.69 |
| | [2] | **0.99/0.99** | **0.99/0.99** | 0.94/0.94 | 0.11/0.11 | 0.96/0.97 | 0.94/0.94 | 0.92/0.93 | 0.83/0.83 | 0.76/0.76 | 0.96/0.96 | 0.98/0.98 | 0.92/0.92 |
| | [3] | **0.99/0.99** | **0.99/0.99** | 0.92/0.92 | 0.19/0.19 | 0.96/0.96 | 0.92/0.91 | 0.92/0.91 | 0.80/0.81 | 0.75/0.76 | 0.96/0.97 | 0.98/0.98 | 0.90/0.89 |
| Llama-2 -13B (finetuned) | [1] | **0.99/0.99** | 0.98/0.98 | 0.87/0.87 | 0.19/0.22 | 0.96/0.97 | 0.92/0.92 | 0.91/0.92 | 0.80/0.80 | 0.75/0.77 | 0.97/0.97 | 0.99/0.98 | 0.90/0.90 |
| | [2] | **0.99/0.99** | 0.98/**0.99** | 0.87/0.87 | 0.19/0.20 | 0.96/0.97 | 0.92/0.93 | 0.91/0.91 | 0.81/0.83 | 0.78/0.78 | 0.97/0.97 | 0.98/0.98 | 0.91/0.92 |
| | [3] | **0.99/0.99** | **0.99**/0.98 | 0.90/0.90 | 0.19/0.19 | 0.95/0.96 | 0.93/0.93 | 0.91/0.92 | 0.86/0.86 | 0.82/0.82 | 0.96/0.97 | 0.98/0.98 | 0.92/0.92 |
| Llama-3 -8B | [1] | 0.96/0.93 | **0.97/0.96** | 0.92/0.92 | 0.18/0.18 | 0.86/0.82 | 0.91/0.91 | 0.69/0.69 | 0.76/0.75 | 0.79/0.80 | 0.87/0.83 | 0.95/0.93 | 0.88/0.88 |
| | [2] | 0.96/0.95 | **0.98/0.97** | 0.93/0.93 | 0.15/0.15 | 0.87/0.86 | 0.92/0.93 | 0.70/0.72 | 0.78/0.76 | 0.78/0.79 | 0.89/0.87 | 0.96/0.95 | 0.89/0.89 |
| | [3] | 0.88/**0.86** | **0.91**/0.85 | 0.90/**0.86** | 0.16/0.14 | 0.79/0.64 | 0.84/0.82 | 0.64/0.53 | 0.60/0.55 | 0.59/0.61 | 0.79/0.70 | 0.87/0.79 | 0.75/0.73 |
| Phi-3 -mini -4k | [1] | 0.94/0.94 | **0.97/0.97** | 0.92/0.92 | 0.07/0.07 | 0.80/0.82 | 0.93/0.93 | 0.61/0.64 | 0.66/0.67 | 0.52/0.53 | 0.84/0.84 | 0.95/0.96 | 0.87/0.87 |
| | [2] | 0.94/0.94 | **0.97/0.97** | 0.93/0.93 | 0.06/0.05 | 0.82/0.82 | 0.93/0.92 | 0.61/0.63 | 0.67/0.65 | 0.53/0.52 | 0.84/0.83 | 0.95/0.96 | 0.87/0.86 |
| | [3] | 0.92/0.93 | **0.97/0.98** | 0.90/0.90 | 0.10/0.12 | 0.74/0.75 | 0.90/0.90 | 0.48/0.46 | 0.61/0.66 | 0.58/0.60 | 0.78/0.78 | 0.94/0.96 | 0.84/0.85 |
| Qwen-2.5 -1.5B | [1] | 0.81/0.82 | **0.96/0.96** | 0.51/0.53 | 0.12/0.13 | 0.67/0.69 | 0.82/0.83 | 0.55/0.59 | 0.69/0.67 | 0.63/0.65 | 0.81/0.81 | 0.93/0.93 | 0.77/0.75 |
| | [2] | 0.83/0.82 | **0.96/0.97** | 0.54/0.56 | 0.12/0.12 | 0.66/0.66 | 0.87/0.86 | 0.58/0.58 | 0.71/0.69 | 0.64/0.61 | 0.82/0.81 | 0.95/0.95 | 0.80/0.77 |
| | [3] | 0.67/0.80 | **0.92/0.97** | 0.23/0.38 | 0.10/0.15 | 0.54/0.65 | 0.55/0.85 | 0.50/0.52 | 0.56/0.65 | 0.41/0.50 | 0.66/0.78 | 0.82/0.94 | 0.49/0.77 |

Table 3: Average $R^2$ scores with 8-bit and full-precision settings based on five runs. In each cell, x/y represents the 8-bit quantization and full-precision scores. The standard errors are nearly zero.

| Setting | Prompting | $1^{st}$ | $2^{nd}$ | $3^{rd}$ | $4^{th}$ | $5^{th}$ | Avg 30 |
|---|---|---|---|---|---|---|---|
| Zero-shot | No-constraint | 0.11 (2,100) | 0.10 (110) | 0.09 (435) | 0.07 (3,499) | 0.07 (190) | 0.06 |
| | Length | 0.14 (2,100) | 0.10 (110) | 0.06 (435) | 0.05 (321) | 0.05 (1,411) | 0.05 |
| | Priming | **0.38** (371) | **0.32** (2,741) | **0.23** (1,380) | **0.19** (4,698) | **0.18** (4,554) | **0.08** |
| Fine-tuning | No-constraint | **0.42** (2,741) | 0.35 (1,380) | **0.34** (371) | 0.28 (4,698) | **0.26** (2,282) | 0.19 |
| | Length | 0.39 (1,380) | 0.38 (371) | 0.37 (2,741) | 0.28 (4,372) | 0.25 (4,698) | 0.20 |
| | Priming | 0.40 (371) | **0.39** (2,741) | **0.34** (1,380) | **0.31** (4,372) | **0.26** (1,419) | **0.21** |

Table 4: $R^2$ scores for individual hidden unit. The numbers in parentheses indicate an index of hidden units from the second layer of the attention mechanisms.

| | Zero-shot | | Fine-tuning | |
|---|---|---|---|---|
| Scale | Conc. | Infor. | Conc. | Infor. |
| -10 | 3.56 | **3.71**$^{\dagger}$ | 3.33 | **3.34**$^{\dagger}$ |
| 1 | 3.59 | 3.70 | 3.46 | 3.31 |
| Gold | 3.58 | 3.68 | 3.45 | 3.28 |
| 10 | 3.59 | 3.63 | **3.47**$^{\dagger}$ | 3.19 |

Table 5: The results of human evaluations using the Priming prompt with Llama-2-13B-Chat. † indicates the improvement for scales between 10 and -10 is significant using paired-bootstrap-resampling with 100,000 random samples ($p<0.05$) (Koehn, 2004).

tional experimental results using beam and top-$k$ sampling decoding strategies are in Appendix C. We also experimented with machine translation and story generation tasks using WMT16 (Bojar et al., 2016) and ROCStories (Mostafazadeh et al., 2016) test datasets. We disentangled identified length-related units from the summarization prompts. Figure 6 and 7 show the results. We observed that length-specific units are globally shared regardless of tasks. The details are in Appendix D.

### 5.1 Human Evaluation and Case Study

**Human Evaluation.** We conducted human evaluations to further assess the effect of disentangling length-related units. Note that we separately evaluated the zero-shot and fine-tuning settings; thus, their scales might be different. We sampled 100 instances for each setting from the Google test dataset. Using Amazon Mechanical Turk, we assigned a total of 80 evaluators who held both US high school and bachelor's degrees for grading the results, with scores from 1 to 5 (5 is the best), in terms of conciseness (Conc) and informativeness (Info). Table 5 shows the results. In the zero-shot and fine-tuning settings, adjusting the length-related hidden units with positive scaling factors generally enhances conciseness but slightly decreases informativeness because of generating shorter summaries. In some cases, generated summaries are already short (Juseon-Do et al., 2024), and so conciseness scores are slightly higher when positive scaling was applied than those from the Base scale 1. In contrast, negative scaling improves informativeness but slightly decreases conciseness due to the production of longer summaries, which can be an inherent trade-off between conciseness and informativeness when controlling output sequence length in summarization (Kikuchi et al., 2016; Makino et al., 2019).

**Case Study.** We conducted a detailed case study to analyze the effects of disentangling length-related hidden units by comparing the generated outputs for different scaling factors with the source and the gold summary. Table 6 and 9 present examples. We observed changes in the generated sum-

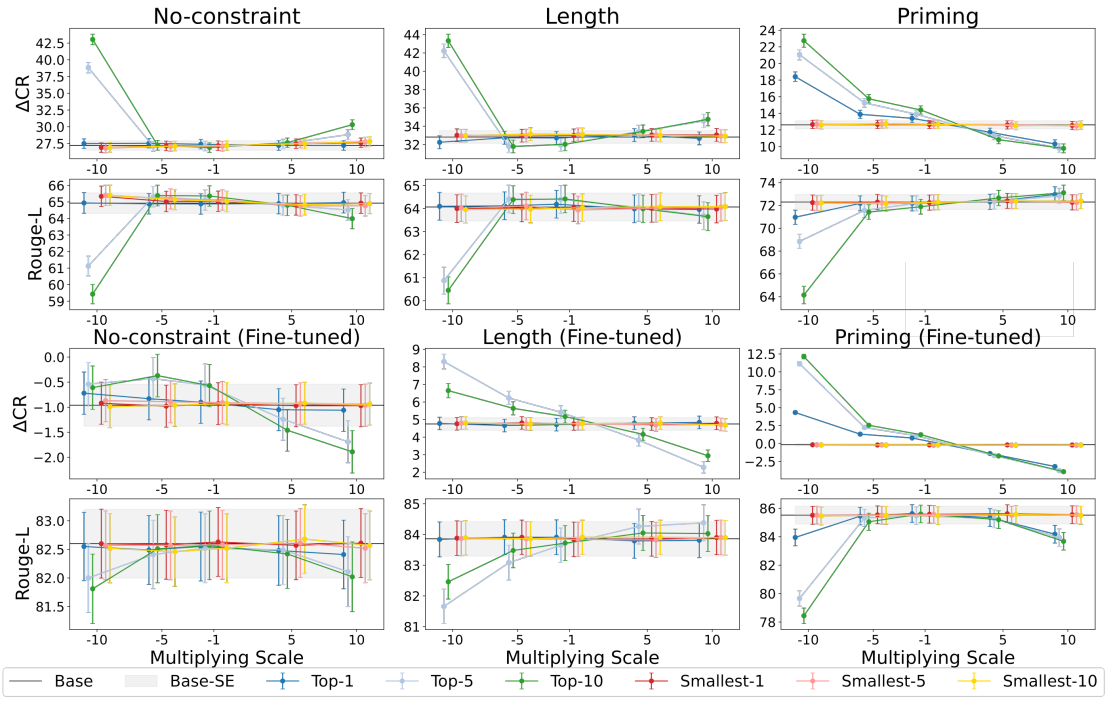


Figure 2: ΔCR and Rouge-L scores change with standard errors by multiplying scale in the Llama-2-13B-Chat. (a) and (b) mean zero-shot and fine-tuning settings, respectively. Base means original scores without scale modification (i.e., the multiplying scale is 1). The gray color represents the standard errors of the Base.

| Type | | Text | Length (#word) |
|---|---|---|---|
| Source | | Armenian national's midfielder Aras Ozbiliz may miss the friendly match against Russia, technical director Vardan Minasyan told reporters ahead of the match. | 22 |
| Gold | | Aras Ozbiliz may miss the friendly match against Russia. | 9 |
| **Top-10** | Scale 5 | Armenian midfielder may miss Russia against match. | 6 (-1) |
| | Scale 10 | Armenian midfielder may miss Russia against match. | 6 (-1) |
| | Scale -5 | Armenian midfielder Aras Ozbiliz may miss the match against Russia. | 10 (+3) |
| | Scale -10 | Aras Ozbiliz may miss the friendly match against Russia, technical director Vardan Minasyan told reporters ahead of the match. | 19 (+12) |
| **Base (Scale 1)** | | Armenian midfielder may miss match against Russia. | 7 |
| **Smallest -10** | Scale 5 | Armenian midfielder may miss match against Russia. | 7 |
| | Scale 10 | Armenian midfielder may miss match against Russia. | 7 |
| | Scale -5 | Armenian midfielder may miss match against Russia. | 7 |
| | Scale -10 | Armenian midfielder may miss match against Russia. | 7 |

Table 6: Summarization example by scaling with Llama-2-13B-Chat in zero-shot Priming. The highlighted part represents the changed part from the Base text. The gray and red tokens indicate deleted and added tokens, respectively, while the blue token represents tokens that have changed their positions.

maries based on different scaling factors. In particular, when negative scaling was applied, the generated summaries became longer than the Base summary by incorporating redundant information from the source. In comparison, applying positive scaling values leads to shorter summaries by focusing on important content similar to the gold summary. When we disentangled the smallest-$k$ hidden units, the generated summaries remained unchanged, regardless of the scaling factors, consistently producing the same summary as the Base.

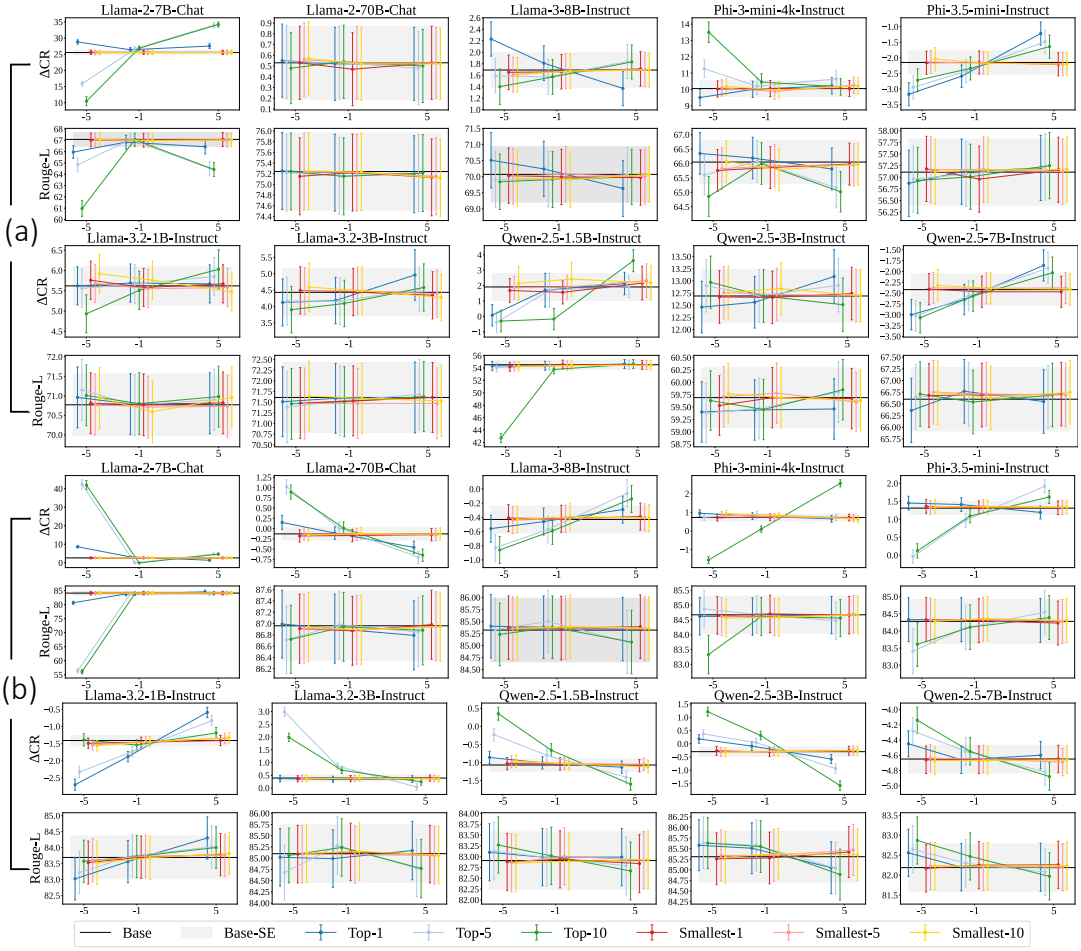


Figure 3: Results in (a) zero-shot settings and (b) fine-tuning settings with the Priming prompt.

## 6 Discussion and Conclusion

We examined how LLMs encode output sequence length in their internal representations. Our findings empirically demonstrated that the outputs from the second layer's attention mechanisms showed a strong correlation with the generation time step, thus indicating that length representations were captured early in the process. We also found that this pattern is consistent with different models with different sizes and continues to be robust even when 4- and 8-bit quantizations were applied. Furthermore, we analyzed individual hidden unit from the second layer attention outputs and found that certain hidden units are highly activated and directly contribute to the process of representing length information. Moreover, these units became more active when length-specific prompts such as Priming were used. This finding indicates that LLMs adjust their internal representations based on the input prompts. Furthermore, by scaling these length-related hidden units, we effectively controlled the output sequence length without losing informativeness, that indicates that length information is partially disentangled from semantic representations within LLMs. Finally, our results revealed that fine-tuning further improves the LLMs' capabilities by reinforcing reliance on the top-$k$ length-related units. We also found the same activation of specific hidden units in the Priming prompt are shared between zero-shot and fine-tuning settings, that indicates LLMs have constructed robust internal mechanisms for controlling output sequence length, and in-context learning performs similarly to implicit fine-tuning (Dai et al., 2023).

## Limitations

Our findings have important implications for the interpretability and controllability of LLMs in natural language generation tasks. Understanding how length information is internally encoded allows for more precise length control over generated outputs, which is crucial in applications, such as summarization and machine translation, where adhering to length constraints is often required. We focused

on a summarization dataset because summarization is a widely used task for controlling output length (Kikuchi et al., 2016; Takase and Okazaki, 2019), and recent studies on the summarization task still face challenges when using LLMs for length control (Juseon-Do et al., 2024; Yuan et al., 2024; Wang et al., 2024). Thus, we considered other tasks such as machine translation and story generation and discussed them in Appendix D.

While we used neural networks to identify length representations in LLMs, there are inherent limitations, particularly due to their ability to decode functionally irrelevant information from model representations (Sajjad et al., 2022; Belinkov, 2022). To validate that the identified hidden units are involved in length control in LLMs, we disentangled their top-$k$ and smallest-$k$ length-related units. However, our findings face a limitation in their application to LLMs that use grouped-query attentions in zero-shot settings. Despite disentangling the top-$k$ units, the proposed methods do not effectively control or influence the model's internal representations for length control. Moreover, whether our findings can extend to models employing Mixture of Experts (MoE) architectures or other types of LLMs remains an open question. In the future, we will extend our approach for the models that use grouped-query attentions and will investigate the MoE models as well.

## Ethics Statement

We recruited annotators using Amazon Mechanical Turk for human evaluations. Because the LLM-generated summaries may contain inappropriate or sensitive language, we reviewed the summaries beforehand and found no problematic samples.

## Acknowledgments

We would like to gratefully acknowledge the anonymous reviewers for their helpful comments and feedbacks. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2022-00155857, Artificial Intelligence Convergence Innovation Human Resources Development (Chungnam National University)).

## References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Gasper Beguš, Maksymilian Dąbkowski, and Ryan Rhodes. 2023. Large linguistic models: Analyzing theoretical linguistic abilities of llms. In *arXiv*.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 131–198, Berlin, Germany. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child,

Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. In *arXiv*.

Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. 2023. Why can GPT learn in-context? language models secretly perform gradient descent as meta-optimizers. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4005–4019, Toronto, Canada. Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *arXiv preprint arXiv:2305.14314*.

Besnik Fetahu, Zhiyu Chen, Oleg Rokhlenko, and Shervin Malmasi. 2023. InstructPTS: Instruction-tuning LLMs for product title summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 663–674, Singapore. Association for Computational Linguistics.

Katja Filippova and Yasemin Altun. 2013. Overcoming the lack of parallel data in sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1481–1491, Seattle, Washington, USA. Association for Computational Linguistics.

Demian Ghalandari, Chris Hokamp, and Georgiana Ifrim. 2022. Efficient unsupervised sentence compression by fine-tuning transformers with reinforcement learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1267–1280, Dublin, Ireland. Association for Computational Linguistics.

Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, , and Chris Olah. 2021. Multimodal neurons in artificial neural networks. *Distill*, (6(3):e30).

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xin-

feng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Prakhar Gupta, Jeffrey Bigham, Yulia Tsvetkov, and Amy Pavel. 2021. Controlling dialogue generation with semantic exemplars. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3018–3029, Online. Association for Computational Linguistics.

Wes Gurnee and Max Tegmark. 2024. Language models represent space and time. In *The Twelfth International Conference on Learning Representations*.

Michael Hanna, Ollie Liu, and Alexandre Variengien. 2023. How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. In *arXiv*.

Junxian He, Wojciech Kryscinski, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2022. CTRLsum: Towards generic controllable text summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5879–5915, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Benjamin Heinzerling and Kentaro Inui. 2024. Monotonic representation of numeric attributes in language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 175–195, Bangkok, Thailand. Association for Computational Linguistics.

Wenxiang Jiao, Wenxuan Wang, Jen tse Huang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. Is chatgpt a good translator? yes with gpt-4 as the engine. In *arXiv*.

Renlong Jie, Xiaojun Meng, Lifeng Shang, Xin Jiang, and Qun Liu. 2024. Prompt-based length controlled generation with multiple control types. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1067–1085, Bangkok, Thailand. Association for Computational Linguistics.

Juseon-Do Juseon-Do, Hidetaka Kamigaito, Manabu Okumura, and Jingun Kwon. 2024. InstructCMP: Length control in sentence compression through instruction-based large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8980–8996, Bangkok, Thailand. Association for Computational Linguistics.

Guy Kaplan, Matanel Oren, Yuval Reif, and Roy Schwartz. 2024. From tokens to words: On the inner lexicon of llms. *Preprint*, arXiv:2410.05864.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2016. Controlling output length in neural encoder-decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1328–1338, Austin, Texas. Association for Computational Linguistics.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large language models are zero-shot reasoners. In *arXiv*.

Jingun Kwon, Hidetaka Kamigaito, and Manabu Okumura. 2023. Abstractive document summarization with summary-length prediction. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 618–624, Dubrovnik, Croatia. Association for Computational Linguistics.

Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Huang. 2023. A systematic study and comprehensive evaluation of ChatGPT on benchmark datasets. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 431–469, Toronto, Canada. Association for Computational Linguistics.

Haoran Li, Junnan Zhu, Jiajun Zhang, Chengqing Zong, and Xiaodong He. 2020. Keywords-guided abstractive sentence summarization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8196–8203.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Bang Liu, Haojie Wei, Di Niu, Haolan Chen, and Yancheng He. 2020. Asking questions the human way: Scalable question-answer generation from text corpus. In *Proceedings of The Web Conference 2020*, WWW '20, page 2032–2043, New York, NY, USA. Association for Computing Machinery.

Yizhu Liu, Qi Jia, and Kenny Zhu. 2022. Length control in abstractive summarization by pretraining information selection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6885–6895, Dublin, Ireland. Association for Computational Linguistics.

Yizhu Liu, Zhiyi Luo, and Kenny Zhu. 2018. Controlling length in abstractive summarization using a convolutional neural network. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4110–4119, Brussels, Belgium. Association for Computational Linguistics.

Takuya Makino, Tomoya Iwakura, Hiroya Takamura, and Manabu Okumura. 2019. Global optimization under length constraint for neural text summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1039–1048, Florence, Italy. Association for Computational Linguistics.

Meta. 2024. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Kenton Murray and David Chiang. 2018. Correcting length bias in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 212–223, Brussels, Belgium. Association for Computational Linguistics.

Jingcheng Niu, Wenjie Lu, and Gerald Penn. 2022. Does BERT rediscover a classical NLP pipeline? In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3143–3153, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. 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 Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Keqin Peng, Liang Ding, Qihuang Zhong, Li Shen, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2023. Towards making the most of ChatGPT for machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5622–5633, Singapore. Association for Computational Linguistics.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. page 9. OpenAI.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Tilman Räuker, Anson Ho, Stephen Casper, and Dylan Hadfield-Menell. 2023. Toward transparent ai: A survey on interpreting the inner structures of deep neural networks. In *arXiv*.

Hassan Sajjad, Nadir Durrani, and Fahim Dalvi. 2022. Neuron-level interpretation of deep nlp models: A survey. *Transactions of the Association for Computational Linguistics*, 10:1285–1303.

Raphael Schumann, Lili Mou, Yao Lu, Olga Vechtomova, and Katja Markert. 2020. Discrete optimization for unsupervised sentence summarization with word-level extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5032–5042, Online. Association for Computational Linguistics.

Wei Shen, Rui Zheng, Wenyu Zhan, Jun Zhao, Shihan Dou, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. Loose lips sink ships: Mitigating length bias in reinforcement learning from human feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2859–2873, Singapore. Association for Computational Linguistics.

Xing Shi, Kevin Knight, and Deniz Yuret. 2016. Why neural translations are the right length. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2278–2282, Austin, Texas. Association for Computational Linguistics.

Sho Takase and Naoaki Okazaki. 2019. Positional encoding to control output sequence length. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3999–4004, Minneapolis, Minnesota. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. In *arXiv*.

Noah Wang, Feiyu Duan, Yibo Zhang, Wangchunshu Zhou, Ke Xu, Wenhao Huang, and Jie Fu. 2024. PositionID: LLMs can control lengths, copy and paste with explicit positional awareness. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16877–16915, Miami, Florida, USA. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing

Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. In *arXiv*.

Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2023. Empirical study of zero-shot NER with ChatGPT. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7935–7956, Singapore. Association for Computational Linguistics.

Tingyu Xie, Qi Li, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2024. Self-improving for zero-shot named entity recognition with large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 583–593, Mexico City, Mexico. Association for Computational Linguistics.

Xi Ye, Srinivasan Iyer, Asli Celikyilmaz, Veselin Stoyanov, Greg Durrett, and Ramakanth Pasunuru. 2023. Complementary explanations for effective in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4469–4484, Toronto, Canada. Association for Computational Linguistics.

Weizhe Yuan, Ilia Kulikov, Ping Yu, Kyunghyun Cho, Sainbayar Sukhbaatar, Jason Weston, and Jing Xu. 2024. Following length constraints in instructions. In *arXiv*.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *arXiv*.

Hattie Zhou, Azade Nova, Hugo Larochelle, Aaron Courville, Behnam Neyshabur, and Hanie Sedghi. 2022. Teaching algorithmic reasoning via in-context learning. In *arXiv*.

Zhang Zhuocheng, Shuhao Gu, Min Zhang, and Yang Feng. 2023. Addressing the length bias challenge in document-level neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11545–11556, Singapore. Association for Computational Linguistics.

## A Dataset Details

For the neural network regression, we used the dataset generated from each sequence of summaries, excluding input token prompts. We randomly divided the datasets into 90% for training and 10% for validation. Table 7 shows statistics.

| Model | No-constraint | Length | Priming |
|---|---|---|---|
| Llama-2-7B-Chat | 21,385 | 24,121 | 25,394 |
| Llama-2-13B-Chat | 26,526 | 28,590 | 20,291 |
| Llama-2-13B-Chat(fine-tuned) | 14,826 | 16,373 | 14,884 |
| Llama-2-70B-Chat | 20,885 | 15,707 | 19,870 |
| Llama-3-8B-Instruct | 17,952 | 22,853 | 13,366 |
| Phi3-mini-4k-Instruct | 30,552 | 18,500 | 25,160 |
| Phi3-small-8k-Instruct | 25,578 | 30,938 | 18,841 |

Table 7: Dataset Statistics.

## B Experimental Details

| Parameter | Value |
|---|---|
| Epochs | 1,000 |
| Batch size | 32, 64 |
| Learning rate | 1e-3 |
| Dropout rate | 0.1 |
| Patience | 5, 10 |
| Loss | MSE |
| Activation | ReLU |

Table 8: Hyperparameters.

**Computing Interfaces.** We used the following GPUs:

- NVIDIA A100 GPU for Llama-2-70B-Chat
- NVIDIA A6000 GPU for other LLMs

**Hyperparameters.** Table 8 shows the hyperparameters used in our experiments. For the neural network regression to predict the generation time step from all hidden unit (Table 2, Table 3, and Figure 1), the batch size was set to 32 with an early stopping patience of 10 epochs. For the neural network regression to predict the generation time step from each individual hidden unit (Table 4, Figure 2 and Figure 3), the batch size was set to 64 with an early stopping patience of 5 epochs.

## C Other Decoding Strategies

Figure 4 and 5 show additional experimental results for disentangling the top- and smallest-$k$ hidden units using beam and top-$k$ sampling methods. The beam size was set to three, and top-$k$ sampling was set to 10. We observed consistent results with the greedy decoding method.

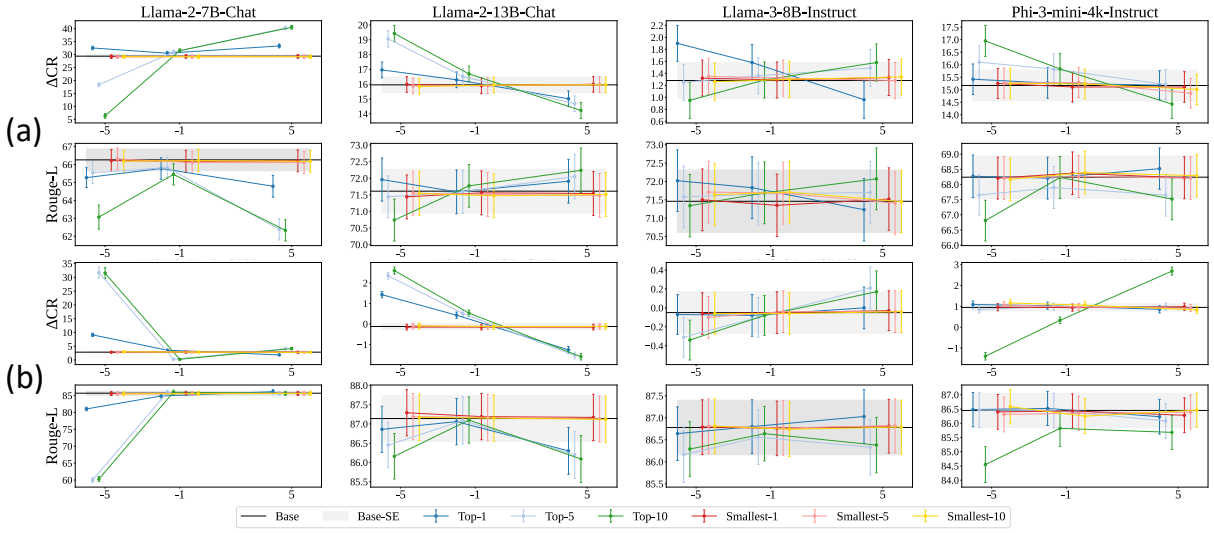


Figure 4: Results in (a) zero-shot settings and (b) fine-tuning settings using the Priming prompt with Beam decoding.

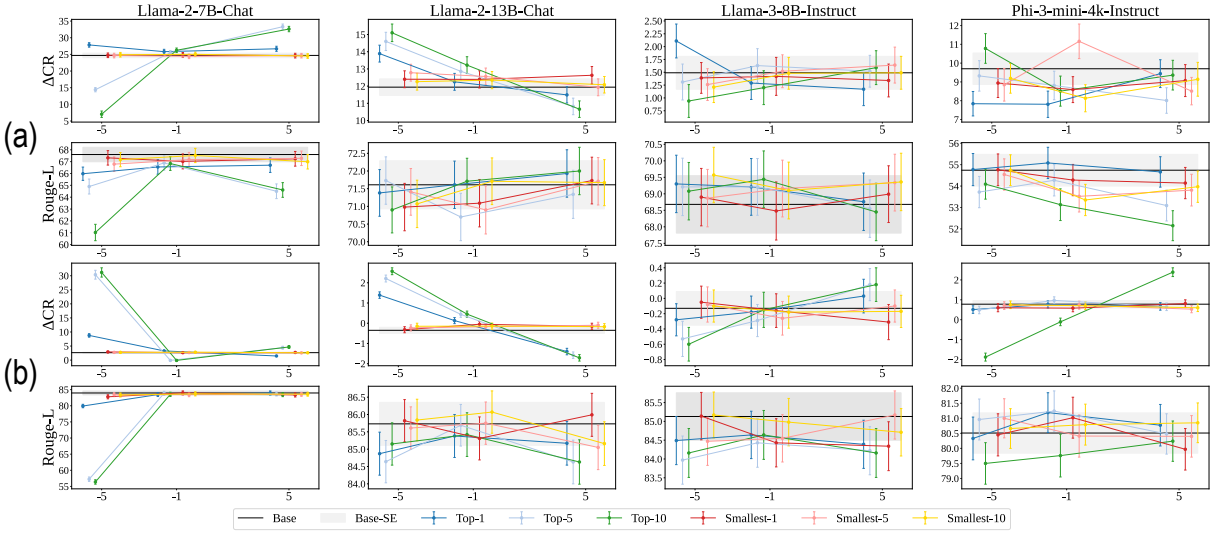


Figure 5: Results in (a) zero-shot settings and (b) fine-tuning settings using the Priming prompt with top-*k* sampling.

## D Machine Translation and Story Generation

We conducted additional experiments on two tasks: machine translation and story generation. For machine translation, we randomly sampled 500 instances from the WMT16 test dataset (Bojar et al., 2016). For story generation, we used the ROCStories test dataset comprising 1,571 instances (Mostafazadeh et al., 2016). We evaluated these tasks in zero-shot settings, disentangling length-related units identified from summarization prompts. Figures 6 and 7 show the results. Interestingly, we found that length-specific units are globally shared across tasks and can be adjusted without sacrificing informativeness.

**Machine translation prompt that considers "priming"**

Sentence that consists of {len(en)} tokens:
{en}
The sentence translated into German that consists of {len(de)} tokens with {len(de)-len(en)} additional tokens would be:
Sentence that consists of {len(en)} tokens:
en
The sentence translated into German that consists of {len(de)} tokens without {len(en)-len(de)} tokens would be:

**Story generation Prompt**

You are given the first four sentences of a short story. Please write a coherent fifth sentence that naturally concludes the story.
Story:
1) {sentence 1}
2) {sentence 2}
3) {sentence 3}
4) {sentence 4}
Now, write the fifth sentence:

## E Other case study

Table 9 shows case studies. We found that the generated summaries ended abnormally early or that tokens were generated without spaces when extreme numeric values, such as -10, were used. This resulted in cases where the R-L scores significantly decreased with extreme scaling factors.

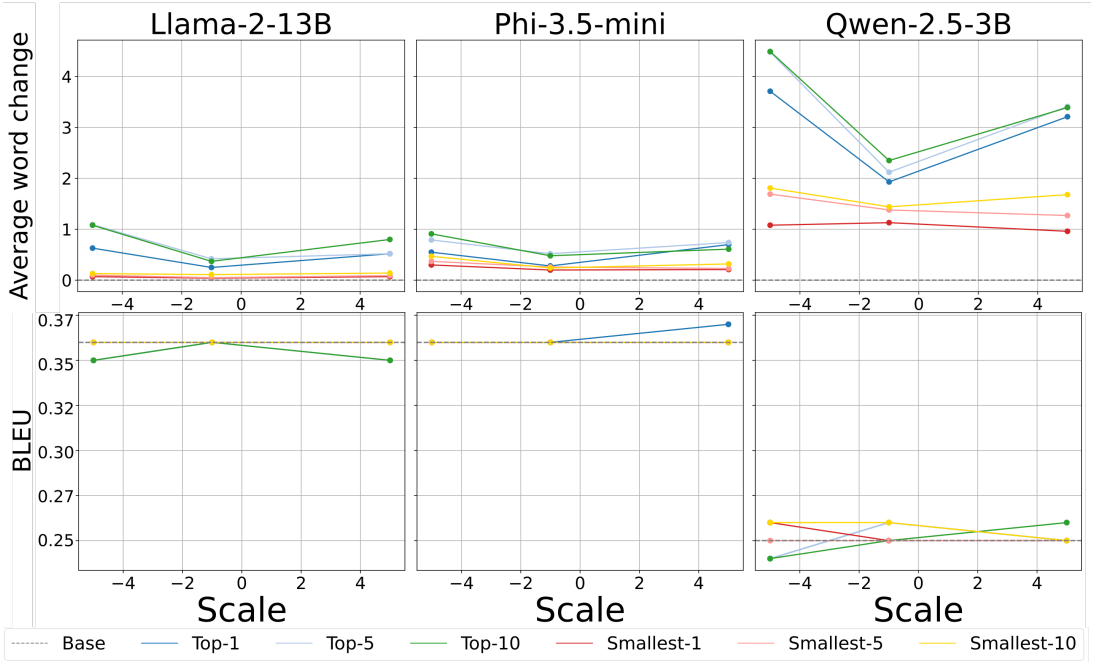


Figure 6: Experimental results on the WMT16 test dataset. We evaluated translation quality using the average of BLEU-1 and BLEU-2 scores.

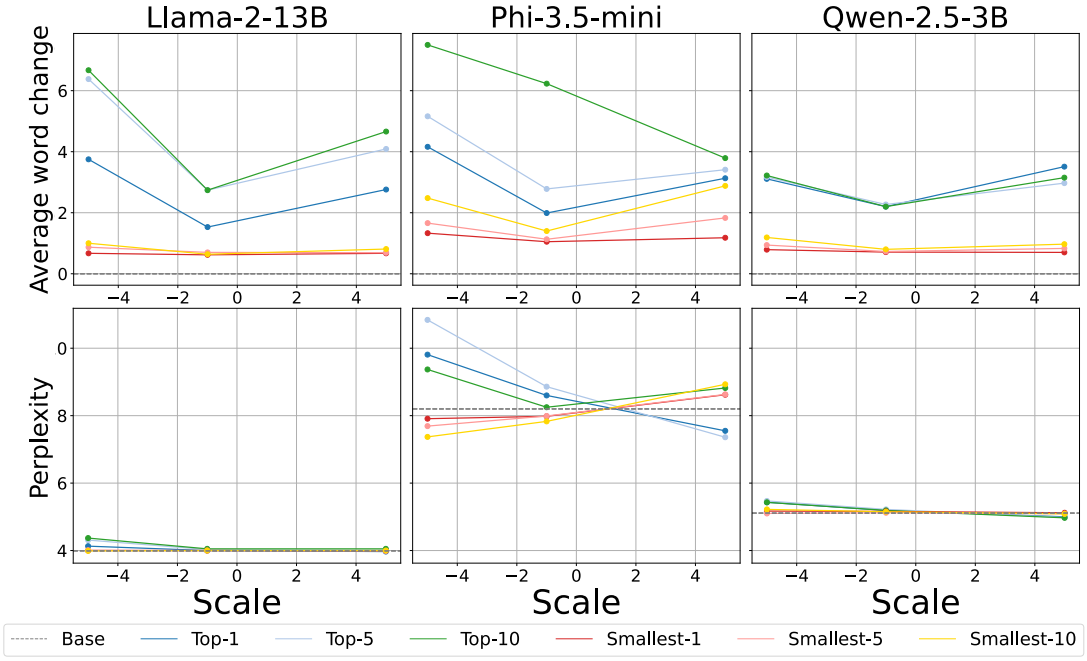


Figure 7: Experimental results on the ROCStories test dataset. We used the Qwen2.5-7B-Instruct model to evaluate perplexity (PPL). Lower PPL scores indicate better text quality.

| Type | | Text | Length (#word) |
|---|---|---|---|
| Source | | South African captain Graeme Smith hailed "an incredible win" for his team after they clinched an emphatic ten-wicket victory on the fifth day of the second and final Test against India at Kingsmead on Monday. | 35 |
| Gold | | Graeme Smith hailed an incredible win. | 6 |
| **Top-10** | Scale -10 | S. | 1 (-8) |
| **Base (Scale 1)** | | South African captain Graeme Smith hailed an incredible win. | 9 |
| Source | | Unknown assailants blew up a natural gas pipeline in Egypt, a security source said. | 14 |
| Gold | | Assailants blew up a gas pipeline in Egypt. | 8 |
| **Top-10** | Scale -10 | AssBlewUpNatGasPipEgy. | 1 (-8) |
| **Base (Scale 1)** | | Assailants blew up a natural gas pipeline in Egypt. | 9 |

Table 9: Case studies by scaling factors using Llama-2-13B-Chat with zero-shot priming.