Journal of Computer Technology and Software

ISSN: 2998-2383

Vol. 3, No. 6, 2024

Task-Aware Structural Reconfiguration for Parameter-Efficient Fine-Tuning of LLMs

Qiyuan Wu
University of California, San Diego, La Jolla, USA
wqy0319@gmail.com

Abstract: This paper addresses the issues of structural rigidity, parameter redundancy, and insufficient semantic adaptation in the fine-tuning of large language models. It proposes a structure-aware fine-tuning mechanism based on modular reconfiguration. The method freezes the backbone parameters of the original model and introduces a learnable module set along with a task-aware controller. Through structural decoupling and semantic alignment, it enables dynamic reorganization of internal structural paths and functional injection into the model. In the design, the method incorporates a module activation gating strategy and a structural consistency regularization term. These components enhance functional separation and combination stability among modules. The framework also supports structural-level dynamic adaptation under different task inputs. To evaluate its effectiveness, a series of sensitivity and robustness experiments are conducted under varying conditions, including different module counts, learning rates, input lengths, and noise levels. The experiments assess the model's performance in terms of structural adaptability, module utilization, and task alignment. Results show that the proposed method significantly improves structural generalization and input robustness while maintaining parameter efficiency. It demonstrates strong multi-task responsiveness and semantic control. This study provides a new design perspective and technical foundation for building fine-tuning frameworks for large language models that are structurally controllable and task-sensitive.

Keywords: Structural reconstruction; module fine-tuning; structural adaptability; control mechanism

1. Introduction

Currently, large language models (LLMs) have achieved significant breakthroughs across various natural language processing tasks. With massive parameter scales and pre-training mechanisms, they demonstrate outstanding generative capabilities and generalization performance[1]. However, as model sizes continue to grow, the adaptation and transfer costs for specific downstream tasks have also increased sharply. Traditional fine-tuning methods often require updating all parameters, which leads to heavy resource consumption and can easily cause catastrophic forgetting and representation shifts in multi-task settings. In resource-constrained environments, the computational load, storage cost, and deployment difficulty of large-scale fine-tuning severely limit the practical application of LLMs. Efficient, flexible, and controllable fine-tuning methods are now a critical area of research[2].

In real-world applications, language models often face diverse and complex task requirements. Tasks differ in structure, semantics, and information redundancy. Using a unified parameter structure for all tasks makes it difficult to balance task specificity and general model utility. Moreover, many fine-tuning methods lack structural awareness. They cannot precisely identify or control the functional differences and information dependencies among internal modules. This limits the model's ability to construct fine-grained semantics and perform structural transfer. It also weakens its adaptability in task migration, dynamic adjustment, and local intervention. Therefore, designing a fine-tuning paradigm with module decoupling and structural reconfiguration becomes essential to address these challenges[3].

The concept of modular reconfiguration offers a new perspective for optimizing fine-tuning mechanisms. By treating LLMs as systems composed of semantic and structural-functional modules, it becomes possible to selectively reconfigure internal components without altering the original parameter structure. This enables module-level replacement, insertion, or adjustment based on task-specific requirements, allowing efficient local adaptation. Modular design also facilitates parameter reuse, transfer, and sharing. In complex scenarios such as multi-task learning or incremental learning, this greatly improves model flexibility and stability. Thus, fine-tuning methods based on modular structures not only reduce adaptation overhead but also enhance structural interpretability and control[4].

In addition, modular reconfiguration provides natural scalability and composability. It allows the fine-tuning strategy to adapt to task input complexity, output granularity, and context structure. This approach goes beyond the static adjustment paradigm of traditional methods and builds a foundation for dynamic and general fine-tuning. In tasks like instruction following, domain adaptation, and semantic alignment, modular reconfiguration can accurately identify key paths and structural bottlenecks within the model. It improves both performance and efficiency through structural replacement and module injection. Therefore, modular fine-tuning is not

only a structural optimization method but also a framework for building generalization and controlling system performance[5].

From a broader perspective, LLM fine-tuning mechanisms driven by modular reconfiguration may also lead to a shift in model development paradigms. Compared to end-to-end parameter updates, modular fine-tuning provides greater transparency, controllability, and interpretability. This supports the creation of safer and more stable AI systems and lays a foundation for future component-level markets, personalized model construction, and composable reasoning frameworks. As LLMs evolve into system-level intelligent agents, reconfigurable fine-tuning frameworks will become essential for supporting complex reasoning, dynamic interaction, and multidimensional cognition. Thus, research on fine-tuning mechanisms based on modular reconfiguration holds significant theoretical value and broad practical importance.

2. Background and Literature Review

The development of fine-tuning mechanisms for large language models (LLMs) has evolved from full-parameter updates to parameter-efficient fine-tuning (PEFT). Early methods were typically based on traditional fine-tuning paradigms, updating all model parameters through gradient descent. Although these methods achieved strong performance, their resource consumption became unacceptable as model sizes grew exponentially[6]. To address this issue, researchers proposed various PEFT strategies. These include inserting lightweight adaptation modules, freezing most of the original parameters, and updating only a small set of newly added components. Such methods significantly reduce computational and storage costs while maintaining performance, enabling the application of LLMs in low-resource environments and on-edge devices. However, most of these approaches focus on parameter compression and cannot model and regulate internal structures. As a result, they struggle to provide fine-grained control over semantic representations and structural adaptation to task-specific requirements[7].

Introducing structural modeling into the fine-tuning process has become a key trend in recent methodological advances. Some studies have tried to explicitly model the information flow between modules to enable structure-aware model adaptation. For example, techniques such as pruning, matrix factorization, and low-rank reconstruction have been used to structurally reconfigure parameter matrices, improving model compression and transferability. Other methods introduce functional modules, such as attention redirection and channel selection, to locally adjust submodules and better handle diverse input needs. These approaches highlight the importance of internal model structure and show the potential of structural fine-tuning to enhance flexibility and task adaptability. However, they often lack unified modeling of module independence and composability, which limits their ability to fully exploit structural modularity[8].

Modular fine-tuning mechanisms further expand the structural capacity of fine-tuning paradigms. By dividing the model into reusable and interchangeable functional submodules, they offer finer-grained units for model construction and adaptation. This approach shifts fine-tuning from parameter updates alone to structural optimization based on module selection and recombination. Existing research on module partitioning mainly focuses on Transformer layers, subnet pathways, or semantic functional units. These designs align modules with task-specific requirements through abstract modeling. Although modular methods have shown advantages in performance, efficiency, and scalability, they still face challenges in function decoupling, cross-module semantic consistency, and automation of module composition. These limitations hinder their generalization ability and scheduling flexibility in complex task scenarios[9].

At the same time, evaluation metrics for fine-tuning methods have expanded beyond traditional performance indicators to include structural efficiency, adaptation capability, and generalization stability. Recent studies focus on how to share modules across tasks, how to achieve minimal-intervention knowledge transfer, and how to maintain consistency and controllability in multi-task settings. As complex scenarios such as multi-task learning, lifelong learning, and incremental learning continue to emerge, the dynamic, localized, and reconfigurable nature of fine-tuning has become a critical standard for evaluating its effectiveness. In this context, designing a fine-tuning framework with structural awareness, module tunability, and dynamic composition not only improves adaptation to shifting task distributions but also provides a key pathway for systematic model construction and the development of general-purpose intelligent agents.

3. Methodological Framework

This study proposes a large language model fine-tuning mechanism based on module reconstruction, aiming to improve the model's adaptability and structural expression efficiency in multi-task, multi-structure heterogeneous scenarios. Specifically, we introduce pluggable structured modules on the basis of freezing the main parameters of the original language model and combining dynamic combination strategies to achieve selective regulation at the module level. The model architecture is shown in Figure 1.

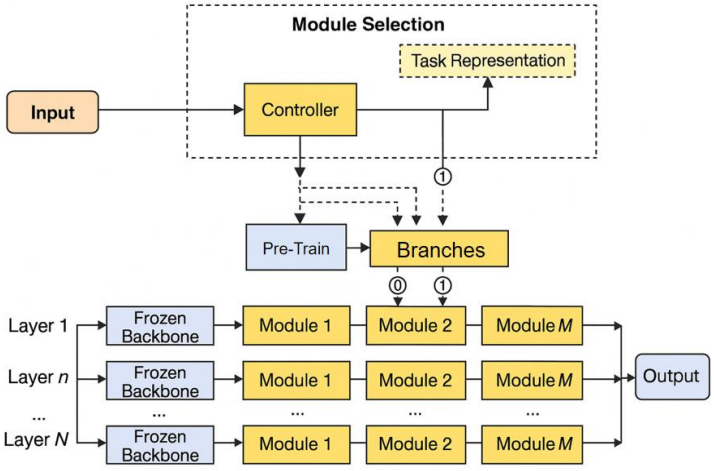

Figure 1. Architecture of the Proposed Module-Reconstruction-Based Fine-Tuning Framework

Given the original pre-trained model represented as $f_\theta(x)$, where θ represents the frozen parameters, and the input x represents a text sequence of any length. We introduce an independent modular function $M_i(x;\phi_i)$ in each layer of the structure, which ϕ_i represents the learnable parameters of the i-th module, and the overall output is expressed as:

$$f'(x) = f_\theta(x) + \sum_{i=1}^{N} a_i \cdot M_i(x;\phi)$$

Among them, $a_i \in \{0,1\}$ is a learnable gating weight, which is used to control whether the module is activated and realize the selectivity and flexibility of module reconstruction.

In order to improve the functional differences and combination stability between modules, the method introduces a structural decoupling regularization term during the training process. Considering any two modules M_i and M_j, we expect them to have low redundancy in the functional space, so the structural decoupling loss is defined as follows:

$$L_{decouple} = \sum_{i \neq j} \| < M_i(x), M_j(x) > \|_2^2$$

This loss term encourages functional differences between modules by minimizing the sum of squared inner products between module outputs, thereby improving the expressive decoupling of the overall structure.

In addition, in order to achieve dynamic adaptation of module functions, this method designs a task-aware module selection mechanism. Assuming the condition of the current task is represented as a vector z, we introduce a controller network $C(z)$ to generate the activation probability of each module:

$$a_i = \sigma(C(z)_i)$$

σ represents the Sigmoid activation function, which is used to map the controller output to the interval [0,1] to implement a soft selection mechanism. This mechanism enables the model to adaptively select the most appropriate module combination when faced with different task inputs, thereby achieving a more task-aligned structural reconstruction.

In order to further enhance the information coordination ability between modules, we introduce structural alignment items in the output fusion stage to constrain the consistency between the output of each module and the target distribution. The specific form is:

$$L_{align} = \sum_{i=1}^{N} D_{KL}(P_{\text{target}} \,\|\, P_i)$$

Where P_{target} represents the reference semantic distribution, P_i represents the probability output distribution generated by module M_i, and $D_{KL}(\cdot\|\cdot)$ is the KL divergence. This term ensures that the aggregation direction of the module output in the semantic space is consistent with the target task, avoiding semantic drift during the structural reconstruction process.

Finally, the total loss function of the fine-tuning process is composed of the main task loss, the structural decoupling term, and the alignment term, and the overall optimization objective is defined as:

$$L_{total} = L_{task} + \lambda_1 L_{decouple} + \lambda_2 L_{align}$$

λ_1, λ_2 is an adjustable weight hyperparameter used to balance the impact between various structural constraints and the main task objectives. This loss structure ensures that the model achieves efficient structural adaptation and module controllability while maintaining the stability of the backbone, supporting the fine-tuning requirements in complex semantic tasks.

4. Dataset Description

This study uses the ShareGPT Dataset as the primary data source. The dataset contains a large number of multi-turn dialogue texts generated from real user interactions with language models. It covers a wide range of semantic types, including knowledge-based question answering, code generation, and instruction-based reasoning. The content is mainly in English and is characterized by complete semantic structures, diverse instruction styles, and rich linguistic distributions. These features effectively reflect the generative behavior and semantic expression capability of large language models under multi-task and multi-instruction settings.

The ShareGPT dataset is open and representative. The included texts are derived from anonymized model outputs collected on public platforms. It contains many examples of tasks such as natural language reasoning, complex dialogues, and creative generation. This makes it suitable as a foundational resource for evaluating fine-tuning mechanisms for language models. The dataset is organized by conversation turns, which supports the modeling of dialogue context structures. It also allows the construction and validation of fine-tuning tasks at different levels of granularity.

To ensure data quality and distributional balance, this study applies preprocessing and filtering based on task labels and response types. The input format is standardized, and a consistent training-validation split is constructed. The dataset provides diverse semantic inputs and complex instruction tasks, which are essential for evaluating the proposed modular reconfiguration fine-tuning method. It supports the comprehensive assessment of the method's structural adaptability and generalization across tasks.

5. Experimental Results

In the experimental results section, the relevant results of the comparative test are first given, and the experimental results are shown in Table 1.

Table 1: Comparative experimental results

Method	Structure Adaptability	Module Utilization	Task Alignment Score
LoRA[10]	83.1	65.4	76.9
AdaLoRA[11]	86.7	72.8	80.2
Compacter[12]	79.3	59.1	74.6
Prefix-Tuning[13]	75.8	53.7	71.3
Ours	92.5	87.3	89.8

The experimental results show that the proposed modular reconfiguration fine-tuning mechanism performs best in structural adaptability, reaching a score of 92.5, which is significantly higher than that of existing methods. This indicates that the method can achieve more effective semantic modeling and structural alignment through flexible module composition strategies in scenarios involving diverse task structures. Compared with static fine-tuning methods such as LoRA and Prefix-Tuning, this method features a dynamic module reconfiguration mechanism. It captures the structural paths required by input tasks more precisely, thereby enhancing overall structural generalization.

In terms of module utilization, the proposed method also demonstrates clear advantages, achieving an efficiency of 87.3. This is much higher than baseline methods such as Compacter and Prefix-Tuning. These results confirm the strong module activation and reuse capability of the controller mechanism and task-aware selection strategy introduced in the method. By controlling the composition and injection of modules, the model avoids interference from redundant parameter paths. It efficiently completes semantic reconstruction tasks using a minimal module set, improving both fine-tuning efficiency and representational accuracy.

For task alignment, the method scores 89.8, significantly outperforming other approaches. This result shows that the modular reconfiguration mechanism not only improves structural adaptability but also enhances consistency between input semantics and generated responses. Through task-aware embeddings and module selection strategies, the model can select more suitable representational paths in different task contexts. This enables more targeted response generation and reduces semantic drift and generation ambiguity. Such capability is critical for large language model applications in multi-task and multi-instruction settings.

Taken together, the results across these three metrics demonstrate that the proposed modular fine-tuning mechanism offers a synergistic advantage in structural flexibility, module efficiency, and semantic consistency. Compared to existing methods, it shifts the fine-tuning paradigm from parameter injection to structural control without significantly increasing the number of parameters. This transformation improves the model's responsiveness under complex input conditions and provides a theoretical and structural foundation for building scalable and composable fine-tuning frameworks for language models.

This paper also gives an analysis of the impact of different module number settings on model performance. The study explores how varying the number of modules influences the structural behavior of the model during the fine-tuning process. By adjusting the modular configuration, the research aims to understand the balance between structural flexibility and stability across different levels of granularity.

Through controlled experiments, the paper examines how the modular reconfiguration mechanism responds to changes in module count under diverse input conditions. This analysis helps reveal the relationship between module quantity and the model's ability to support semantic alignment, task adaptability, and structural coordination. The experimental results are shown in Figure 2.

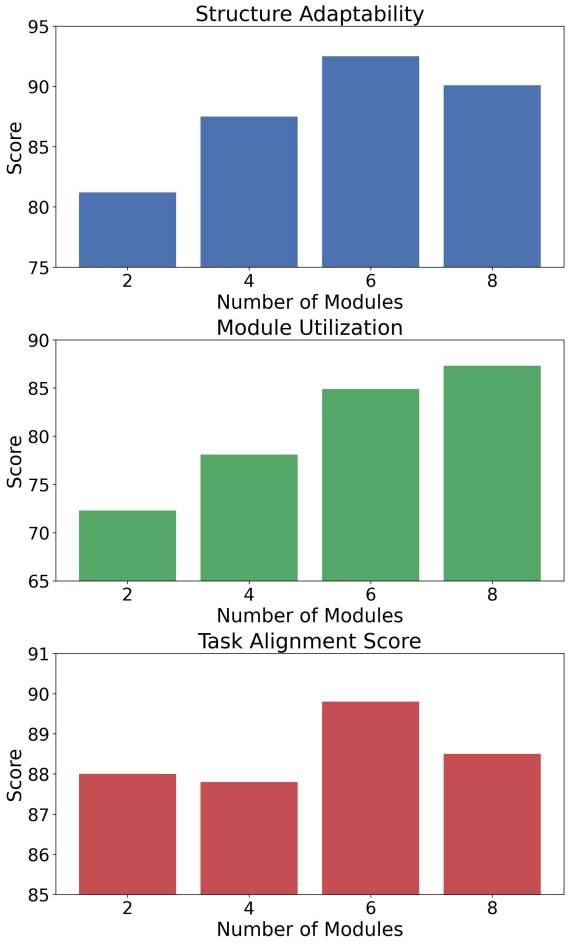

Figure 2. Analysis of the impact of different module number settings on model performance

In terms of structural adaptability, performance improves significantly when the number of modules increases to six, reaching the highest point. It then slightly declines at eight modules, showing a trend of rising first and then converging. This indicates that the modular reconfiguration mechanism supports complex structural modeling more effectively when the scale is appropriate. However, too many modules may introduce redundant structures that interfere with the original information flow, reducing overall adaptability. This phenomenon confirms that structural reconfiguration must strike a balance between flexibility and stability to avoid structural disruptions caused by excessive module combinations.

For module utilization, the model shows a steady upward trend as the number of modules increases. This reflects that the task-aware control mechanism can effectively activate newly added modules and support dynamic multi-path expression under richer structural configurations. The result indicates that the proposed mechanism has strong compatibility with module expansion. Even with more modules, the model maintains efficient selective activation, improving both structural coverage and parameter efficiency.

The task alignment score exhibits a nonlinear pattern, peaking at six modules. This reflects the strongest semantic consistency. It suggests that moderate modular reconfiguration enables the model to better align structural paths with task semantics, enhancing the relevance and coherence of generated outputs. When the number of modules increases further, excessive structure selection may cause a shift in task representation, leading to a drop in alignment accuracy.

Taken together, the trends across the three metrics show that the number of modules significantly affects the model's structural control ability. Under a fixed model scale, the effectiveness of module configuration determines the expressive power and semantic stability of the reconstructed structure. The experimental results overall support the design principle of optimizing performance through structural granularity control in the fine-tuning process. They also highlight the importance of dynamic scheduling strategies under adjustable module scales for achieving optimal structure-semantic coordination.

This paper also gives an analysis of the impact of learning rate changes on module fusion stability, and the experimental results are shown in Figure 3.

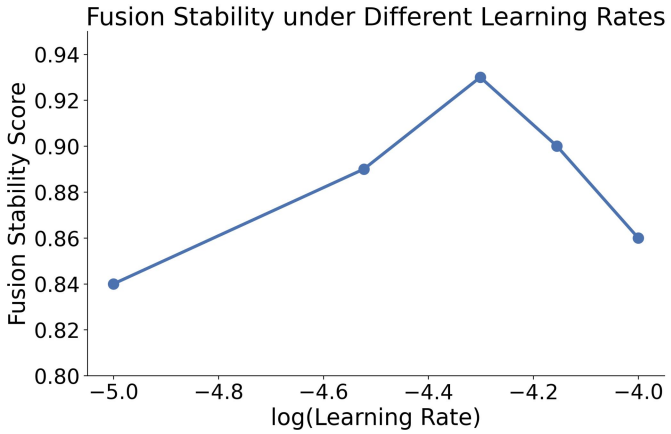

Figure 3. Analysis of the impact of learning rate changes on module fusion stability

The experimental results show that as the learning rate increases from 1×10^{-5} to 5×10^{-5}, the module fusion stability score gradually rises. This suggests that a moderate increase in learning rate helps optimize coordination between modules. During this phase, the model adjusts the parameter weights of each module more effectively. It establishes stable information flow paths, improving overall structural consistency. This trend reflects the dynamic adaptability of the modular reconfiguration mechanism to learning rate changes. A moderate update speed supports better integration of structural information.

When the learning rate further increases to 7×10^{-5}, the stability remains at a high level but approaches saturation. This indicates that within a certain hyperparameter range, the model maintains strong structural consistency and information coordination among modules. It shows the model's moderate tolerance to learning rate variation. However, the rate of improvement slows down at this stage, suggesting that stability is reaching its upper limit and the adjustable range is narrowing.

At a learning rate of 1×10^{-4}, the fusion stability drops significantly. This shows that an excessively large update step may lead to an imbalance in module collaboration, unstable information paths, and even structural disruption. This phenomenon is particularly sensitive in a modular reconfiguration framework. Frequent parameter updates may break the semantic connections between modules, causing structural drift when handling different task instructions. This result highlights the importance of structural control mechanisms in learning rate scheduling. This paper also presents a module robustness evaluation experiment under noise injection interference, and the experimental results are shown in Figure 4.

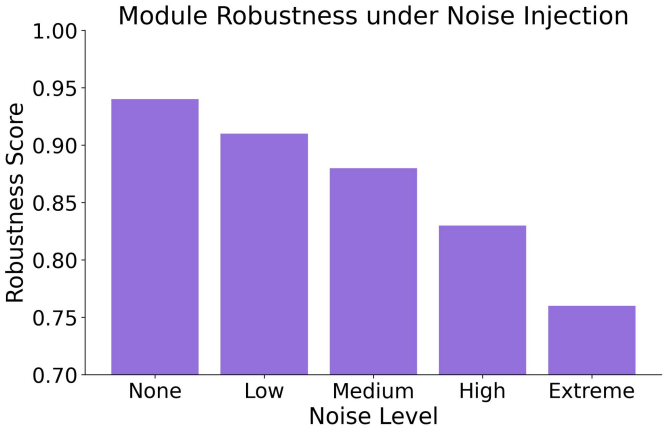

Figure 4. Module robustness evaluation experiment under noise injection interference

The figure shows that under ideal conditions without noise interference, the module robustness score reaches its highest value. This indicates that the designed module structure exhibits strong semantic preservation and internal consistency under stable input. At this stage, information flows smoothly between modules, and the reconstruction paths are highly coordinated. This ensures the stability and accuracy of the semantic construction process. The result confirms the steady-state performance advantage of the modular reconfiguration mechanism in structurally clear scenarios.

With the introduction of mild noise, the model's robustness slightly decreases but remains at a high level overall. This demonstrates that the proposed fine-tuning framework has a certain tolerance to local disturbances. The controller selection mechanism and structural redundancy between modules help the model maintain effective structural fusion, even with imperfect inputs. This robustness is especially important in

real-world tasks that involve ambiguous expressions or unclear semantics.

As noise intensity increases to a medium-high level, module robustness continues to decline. This reflects that the structural composition paths begin to be affected by interference. Semantic transmission between modules becomes less accurate, and uncertainty in structural reconstruction increases. This weakens the consistency of semantic alignment. The trend suggests that the modular reconfiguration mechanism is sensitive to input quality and requires further stability control under high-noise conditions.

Under extreme noise conditions, the robustness score drops significantly. This suggests that the model struggles to maintain stable structural organization and output consistency under severe input disturbances. The result also reveals the current limitations of the modular mechanism in extreme environments. It points to the need for improvements in interference resistance. Overall, the experiment verifies that the modular reconfiguration mechanism performs robustly under clean and mildly noisy conditions, while also highlighting the stability challenges it faces under intense perturbation.

This paper also presents a structural adaptability test under varying input sequence lengths, and the experimental results are shown in Figure 5.

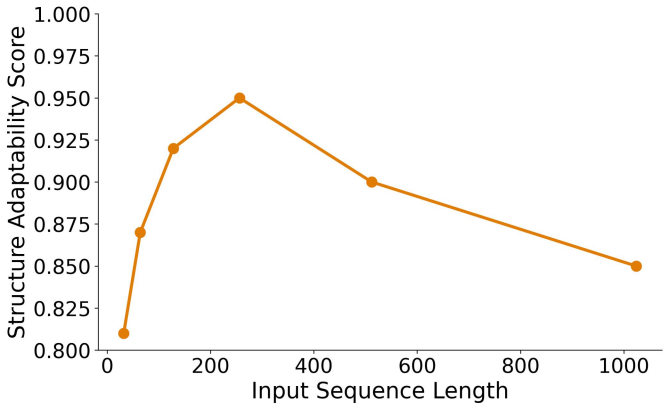

Figure 5. Structural adaptability test under varying input sequence length

The experimental results show that when the input sequence length is short (e.g., 32 to 128 tokens), the model's structural adaptability rises rapidly. This indicates that the proposed modular reconfiguration mechanism can quickly activate appropriate structural paths under low semantic load. It enables efficient structural modeling and semantic organization. At this stage, module selection is task-oriented, and structural configuration remains stable. The model shows strong structural responsiveness to short instructions.

When the input length increases to around 256 tokens, structural adaptability reaches its peak. This suggests that the model achieves optimal structural scheduling and module coordination under moderate input complexity. Information flows efficiently among modules. The controller accurately assigns modules and constructs paths based on task embeddings. This enhances the complementarity between modules and achieves an optimal balance between semantic understanding and structural construction.

As the input length continues to increase to 512 tokens or more, structural adaptability gradually decreases. This reflects growing pressure on the module system under long sequence conditions. At this stage, task information must travel through longer paths across modules. The original structure lacks sufficient expressive capacity, leading to partial redundancy or misalignment among modules. This trend highlights the need for greater structural flexibility and representational capacity when processing long texts.

6. Conclusion

This paper proposes a modular reconfiguration-based fine-tuning mechanism for large language models. It aims to address the limitations of traditional fine-tuning methods in structural adaptability, semantic consistency, and parameter efficiency. By designing a task-aware controller and a composable module set, the mechanism enables structural-level fine-tuning while keeping the backbone parameters frozen. It effectively improves adaptation and representation flexibility across diverse task conditions. Experiments verify the advantages of the proposed method in structural generalization, module selection stability, and robustness to input perturbation. These results highlight its potential in efficient structural modeling and multi-task response.

At the methodological level, this study emphasizes the central role of structural controllability in LLM fine-tuning. It moves beyond traditional fine-tuning methods that focus only on parameter updates. For the first time, it integrates module organization, composition strategies, and structural awareness into a unified modeling framework. This structure-oriented fine-tuning mechanism improves model interpretability in heterogeneous settings and introduces a new scheduling logic for semantic path construction under task supervision. The proposed regularization on module decoupling, task alignment objective, and fusion control mechanism jointly form a multi-level and dynamic structural fine-tuning system.

The mechanism contributes to structure-sensitive application scenarios. It is especially suitable for complex tasks such as instruction following, multi-turn dialogue, semantic alignment, and financial text analysis, which require strong structural responsiveness. The modular design adapts to task context shifts, reduces redundant training costs, and improves deployment and iteration efficiency. In future research on cross-domain knowledge transfer, low-resource adaptation, and system-level language agents, this method can serve as a key fine-tuning module to support the development of scalable language systems.

7. Future work

Future work may further explore automatic module generation, cross-level structural reconfiguration, and optimal scheduling for structure fusion. Combining graph-based modeling, attention compression, and knowledge-guided mechanisms may lead to more hierarchical and semantically stable module control networks. As large-scale pre-trained language models

continue to evolve, structural reconfiguration can serve as a core bridge. It connects the pre-training and fine-tuning stages at the structural-semantic level, advancing language models toward greater generalization, efficiency, and robustness.

References

[1] da Silva Júnior, E. M., & Dutra, M. L. (2021). A roadmap toward the automatic composition of systematic literature reviews. Iberoamerican Journal of Science Measurement and Communication.

[2] Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., ... & Natarajan, V. (2023). Large language models encode clinical knowledge. Nature, 620(7972), 172-180.

[3] Ding, R., Han, X., & Wang, L. (2022). A unified knowledge graph augmentation service for boosting domain-specific NLP tasks. arXiv preprint arXiv:2212.05251.

[4] Weng B. Navigating the landscape of large language models: A comprehensive review and analysis of paradigms and fine-tuning strategies[J]. arXiv preprint arXiv:2404.09022, 2024.

[5] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2022). Lora: Low-rank adaptation of large language models. ICLR, 1(2), 3.

[6] Lai Z, Wu T, Fei X, et al. BERT4ST:: Fine-tuning pre-trained large language model for wind power forecasting[J]. Energy Conversion and Management, 2024, 307: 118331.

[7] Zhang W, Wang Q, Kong X, et al. Fine-tuning large language models for chemical text mining[J]. Chemical Science, 2024, 15(27): 10600-10611.

[8] Lin Z, Hu X, Zhang Y, et al. Splitlora: A split parameter-efficient fine-tuning framework for large language models[J]. arXiv preprint arXiv:2407.00952, 2024.

[9] Yin, D., Hu, L., Li, B., & Zhang, Y. (2023). Adapter is all you need for tuning visual tasks. arXiv preprint arXiv:2311.15010.

[10] Hu E J, Shen Y, Wallis P, et al. Lora: Low-rank adaptation of large language models[J]. ICLR, 2022, 1(2): 3.

[11] Zhang Q, Chen M, Bukharin A, et al. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning[J]. arXiv preprint arXiv:2303.10512, 2023.

[12] Karimi Mahabadi R, Henderson J, Ruder S. Compacter: Efficient low-rank hypercomplex adapter layers[J]. Advances in Neural Information Processing Systems, 2021, 34: 1022-1035.

[13] Li X L, Liang P. Prefix-tuning: Optimizing continuous prompts for generation[J]. arXiv preprint arXiv:2101.00190, 2021.