EDITOR: Can Wang, can.wang@griffith.edu.au

DEPARTMENT: AI AND BEHAVIOR

The Rise of Small Language Models

Qin Zhang and Ziqi Liu, *Shenzhen University, Guangdong, 518055, China*
Shirui Pan, *Griffith University, Gold Coast, Qld., 4215, Australia*

Large language models (LLMs), such as GPT and LLAMA, exhibit exceptional comprehension and reasoning capabilities across a wide range of tasks, which are a result of the extensive corpora and the enormous number of parameters in a model. However, their size can pose significant challenges for deployment, particularly on resource-constrained devices. For issues that degrade the user experience, such as efficiency, latency, safety, and privacy, small language models (SLMs) offer a solution. This article begins by outlining the key principles behind SLMs and the reasons for their importance in the field. Subsequently, we discuss the methods used to develop SLMs and explore the collaboration between SLMs and LLMs. By exploring the pathways for harnessing the unique capabilities of SLMs and optimizing their integration with LLMs, it contributes to the ongoing discussion on their application and collaboration in natural language processing and offers insights for advancement and innovation in the field.

Large language models (LLMs) have emerged as key drivers of progress in artificial intelligence, demonstrating significant proficiency in a wide range of tasks, including writing, coding, and many other text-based activities. Recent developments in LLMs have focused on increasing model sizes, with recent iterations incorporating billions of parameters.[1] This trend toward larger models enhances their capabilities, enabling effective and versatile application across a variety of downstream tasks.

Despite their impressive capabilities, LLMs face several significant challenges that require resolution to maintain innovation and broader adoption. First, the exponential increase in model sizes places considerable demands on essential resources such as computations, energy, and memory.[2] The costs associated with training and deploying these large-scale models make them financially challenging, particularly within resource-constrained settings such as academic institutions and the medical sector. Second, LLMs usually face latency issues in real-time applications,[3] leading to delays between user input and output generation, such as online hallucination detection tasks. Another critical concern is the safety of LLM-generated output with respect to their tendency to produce ungrounded responses,[3] ranging from factual inaccuracies to entirely fabricated information. This issue is compounded by the potential for LLMs to learn and replicate biases, inconsistencies, or false information embedded in their training data, further resulting in the generation of unreliable output. In addition, the widespread commercialization and deployment of LLMs in cloud environments have introduced challenges to user privacy and data safety.[4] For instance, when seeking customized or personalized services, it is often required to upload sensitive personal information or business data.

Digital Object Identifier 10.1109/MIS.2024.3517792
Date of current version 7 February 2025.

Towards these challenges and limitations, an increasing number of researchers are turning to small language models (SLMs), which represent a significant innovation and offer the potential to enhance environmental sustainability and operational adaptability. This article begins by explaining the fundamental principles behind SLMs and the rationale for their essentiality in the field. Subsequently, we discuss the methodologies employed for developing SLMs with efficient architectural designs and model compression techniques. Then we entail a comprehensive examination of how SLMs can be seamlessly integrated with LLMs to enhance data efficiency, computational efficiency, safety, and privacy safeguards within the artificial intelligence landscape. By exploring how to leverage the unique capabilities of SLMs and optimize their integration with LLMs, it contributes to the ongoing discussion on the utilization and integration of SLMs and LLMs and offers insights into the potential avenues for advancement and innovation in this domain.

WHY SLMs?

LLMs perform well in many Natural Language Processing (NLP) tasks, but they also face numerous challenges such as high computational and memory costs as well as high latency. Compared to LLMs like OpenAI's GPT-4, Anthropic's Claude 3.5 Sonnet, and Google's Gemini 1.5 Ultra, which have billions or even trillions of parameters, SLMs require significantly fewer computational resources and much lower energy consumption. They are designed for and widely integrated into small devices, such as the latest versions of iOS 18, iPadOS 18, and macOS Sequoia on iPhone, iPad, and MacBook include a personal intelligence system. This can work with private cloud computing and runs on Apple chip servers, efficiently, accurately, and responsibly executing specialized tasks. In summary, SLMs show outstanding advantages in terms of efficiency, accessibility and customization, privacy, and security.

Efficiency

Being sufficiently trained or fine-tuned on domain-specific data, SLMs have proven to be efficient solutions for target tasks. Compared to LLMs, the smaller number of parameters allows SLMs to analyze queries and generate responses more quickly. Shorter inference times are crucial in many real-time applications. For example, in online hallucination detection tasks, faster responses help maintain smooth interactions without frustrating delays.

Accessibility and Customization

For researchers or companies with limited resources, SLMs present a more feasible option for implementation due to their reduced hardware requirements. The introduction of SLMs can offer an optimal solution for research institutions and small- or medium-sized enterprises, as their accessibility promotes the exploration and experimentation with NLP technologies. This accessibility not only encourages innovation but also enables organizations to align with the advancements in modern technology. In addition, SLMs capitalize on their fewer parameters, allowing for swift retraining or fine-tuning to cater to specific tasks. This adaptability enables them to adeptly respond to evolving demands without incurring the substantial costs associated with LLMs.

Safety and Privacy

SLMs serve as a robust solution for mitigating data privacy and security challenges through the facilitation of localized processing, thus mitigating the necessity to transmit sensitive data across networks. The reduced hardware requirements of SLMs not only contribute to decreased data storage demands but also play a critical role in minimizing the susceptibility to data breaches. The adaptability of SLMs allows for fine-tuning on proprietary datasets, ensuring alignment with regulatory frameworks and bolstering the protection of confidential information. The strategic deployment of SLMs in data processing pipelines enables users to maintain a secure data environment by limiting the exposure of sensitive information during communication and storage. Additionally, the ability to fine-tune SLMs on proprietary datasets helps ensure compliance with data protection regulations and industry-specific standards.

Overall, SLMs possess several compelling characteristics that make them valuable assets in both academic and commercial settings, complementing the capabilities of LLMs in a synergistic and mutually reinforcing manner. By strategically combining SLMs and LLMs, researchers can unlock novel capabilities, pushing the boundaries of NLP while ensuring optimal resource utilization and practical deployment considerations.

HOW TO TRAIN SLMs?

SLMs are compact and streamlined language models that are designed to operate efficiently on devices with limited computational resources. They are particularly useful in applications where real-time processing or low power consumption is a priority, making them suitable for deployment on mobile devices, Internet of Things (IoT) devices, and other edge computing environments. By reducing the complexity and size of the model architecture, SLMs aim to achieve high performance while minimizing computational demands and memory requirements. The approaches used to obtain an SLM can be categorized into two main kinds of strategies: training SLMs from scratch using efficient transformers (illustrated in Figure 1) and deriving SLMs from LLMs through model compression (illustrated in Figure 2). In the following sections, we provide a detailed analysis of each of these strategies.

Training SLMs From Scratch: Efficient Transformer

Transformer-based LLMs harness the power of feed forward networks (FFNs) and attention mechanisms to compute intricate pairwise relationships across input sequences. The FFN in each layer increases resource requirements, especially in deep models, while

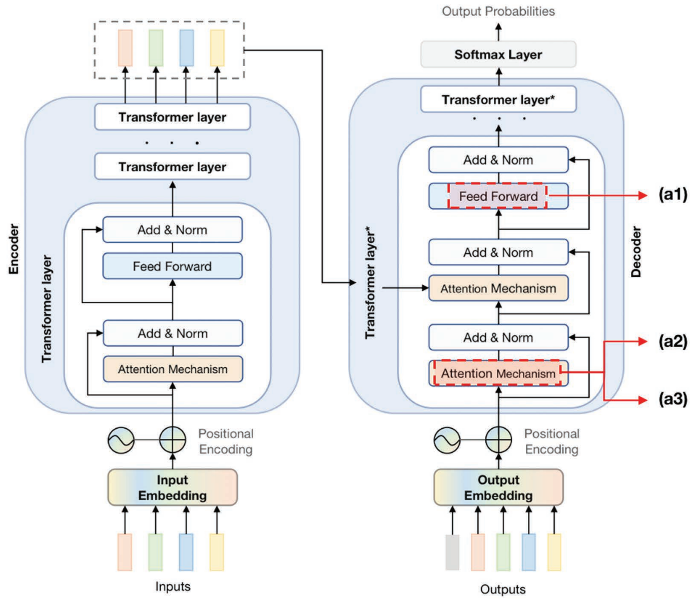

FIGURE 1. The three primary strategies of training SLMs from scratch with efficient transformers: (a1) Efficient FFN. (a2) Efficient Attention. (a3) Attention-free.

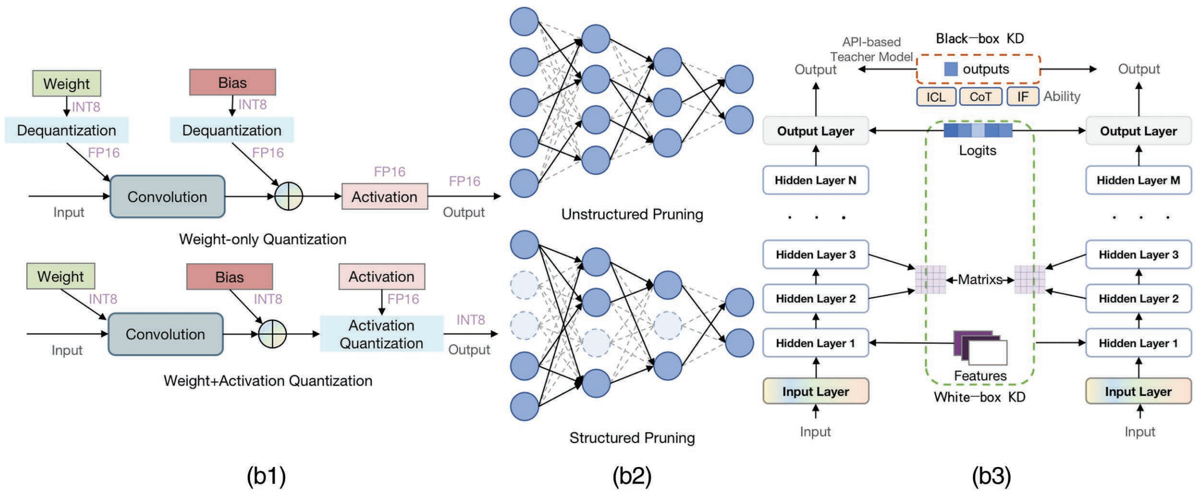

FIGURE 2. Obtaining SLMs from LLMs through model compression: Key approaches include (b1) quantization, (b2) pruning, and (b3) knowledge distillation.

the attention mechanism scales quadratically with sequence length, leading to high processing times and memory consumption. In response to this computational challenge, a range of techniques have been devised to streamline the calculation process and to improve the efficiency of the architectures. As illustrated in Figure 1, these efforts can be categorized into three groups: efficient FFN, efficient attention, and attention-free architecture.

Efficient FFN

Efforts on FFN architecture in transformer-based language models mainly entails a strategic shift through the transition from a dense FFN structure to a sparse FFN configuration utilizing a mixture-of-experts (MoE) framework. Within this paradigm, only a selected subset of "expert" networks is activated for processing each input sequence, reducing computational complexity and allowing for effective parameter scaling without total reliance on the entire network capacity.

A prevalent strategy in MoE-based LLM design involves the substitution of each dense FFN with multiple expert modules. During the inference phase, streamlining the expert weights or streamlining the complexity of these individual experts contributes significantly to the overall efficiency gains. Alternatively, a trainable routing mechanism is implemented to control the activation of specific expert modules, allowing the system to leverage the expertise of a subset of experts tailored to the input data characteristics. By strategically integrating these design principles, the efficiency and effectiveness of FFN in transformer-based LLMs can be markedly enhanced, paving the way for optimized performance and scalability in complex language processing tasks.

Efficient Attention

Transformer-based LLMs harness attention mechanisms to analyze intricate pairwise relations within input sequences, a process that traditionally results in quadratic complexity. In response to this computational burden, several strategies have been devised to streamline the attention calculation process by emphasizing and preserving critical relationships. One representative approach involves the implementation of fast or sparse attention mechanisms. These mechanisms optimize computational efficiency by converting dense 2-D attention matrices into more manageable 1-D structures or by filtering out near-zero attention scores, thereby allowing the model to concentrate computational resources on pivotal interactions.

Besides, the exploration of attention operators integrated with approximation techniques, such as block-level attention computation, presents a promising avenue for enhancing efficiency. By adopting these innovative methodologies, parallel processing capabilities can be leveraged, leading to significant efficiency gains while maintaining performance standards. This strategic blend of attention optimization techniques not only improves computational efficiency but also empowers SLMs to scale effectively, ensuring their adaptability and robustness across a wide range of language processing tasks.

Attention-Free Architecture

While efficient attention structures offer some improvements, they do not alter the worst-case theoretical complexity. Consequently, attention-free architectures have emerged as viable solutions to mitigate the significant computational overhead of traditional attention mechanisms by introducing alternatives that avoid the quadratic attention matrix. These innovative approaches can be broadly categorized into two main types: those that substitute the attention mechanism with recurrent computation and those that employ discretized state-space representations.

Recurrent architectures, for example, leverage sequential processing techniques to manage input data, effectively capturing long-range dependencies without relying extensively on attention calculations. On the other hand, discretizing state-space representations allows models to operate within a reduced-dimensional framework, simplifying token interactions and diminishing computational burdens. By distinguishing between these different methodologies, researchers can explore diverse avenues to optimize computational efficiency and enhance the performance of language models in handling complex language processing tasks.

Training SLMs From LLMs: Model Compression

In the quest to acquire compact language models, besides training from scratch with efficient transformers, another approach is extracting SLMs from existing LLMs through model compression. Specifically, as illustrated in Figure 2, primary model compression approaches further involve quantization, pruning, and knowledge distillation.

Quantization

Model compression through quantization is a fundamental technique that reduces the precision of a model's weights and/or activations, thereby enhancing inference speed on hardware compatible with low-bit datatypes. Illustrated in Figure 2(b1), quantization is a widely accepted approach that converts high-bit representations to more efficient lower-bit datatypes.

One common way is decreasing the precision of weights while maintaining activations at their original level. In weight-only quantization, high-bit weight formats (e.g., 16-bit floating-point) are converted to lower-bit formats (e.g., 8-bit integers). These lower-bit weights are stored and need to be dequantized for computations, ensuring full precision for activations during inference. In this way, it optimizes memory usage, accelerates computations, and maintains accuracy.

Alternatively, weight and activation quantization involves converting both weights and activations to lower-bit representations, often with specialized treatment for activation outliers. This approach leverages faster low-bit arithmetic on specific hardware, eliminating dequantization overhead during inference. However, it may necessitate a larger bit-width for storing weights and activations to accommodate the reduced precision requirements. By exploring these quantization pathways, researchers can tailor model compression strategies to suit different hardware configurations and optimization objectives, striking a balance between speed, accuracy, and resource efficiency in deploying SLMs.

Pruning

As depicted in Figure 2(b2), pruning is a technique focused on optimizing the structure of the model by identifying and eliminating redundancy within the operators of LLMs. This is achieved by removing unnecessary weights or neurons to simplify the model's operations. Pruning can be further categorized into two types: unstructured pruning and structured pruning.

Unstructured pruning involves the selective removal of individual weights from a neural network based on predefined criteria, often emphasizing their magnitudes. This kind of method prunes weights with minimal contribution to model performance, typically those close to zero, resulting in a sparser weight matrix. While unstructured pruning can introduce significant sparsity with minimal accuracy loss, its practical impact on inference speed may be constrained by hardware limitations. In contrast, structured pruning focuses on removing entire groups of parameters, such as neurons, channels, or layers, to simplify the model architecture while preserving or even enhancing performance. This systematic approach to model compression poses challenges when applied to LLMs, particularly in addressing resource constraints and ensuring the availability of suitable datasets.

Knowledge Distillation

As illustrated in Figure 2(b3), knowledge distillation normally utilizes supervisory signals from a larger and more sophisticated "teacher" model to train a compact "student" model. The student model learns to replicate the teacher's outputs, focusing on specific domains. Further, knowledge distillation can be categorized into black-box distillation and white-box distillation.

Black-box distillation is employed when contemporary closed-source large models provide limited internal information, only offering predictions to users. These methods involve transmitting knowledge exclusively through the teacher model's predictions. It normally incorporates three key techniques: in-context learning, chain-of-thought, and instruction following. In-context learning enables large models to generate accurate outputs based on examples, without the need for parameter updates. The chain-of-thought technique helps LLMs enhance their responses by reasoning following the guidance of prompts, thereby improving response details and quality. Instruction following guides LLMs to generate outputs based on specific instructions, facilitating task completion effectively.

In contrast, white-box knowledge distillation allows the student model to access the internal mechanisms of the teacher model during training, enabling it to observe not only the final outputs but also intermediate representations. A representative approach is utilizing outputs from both the final and intermediate layers to guide the student's training and explores the interrelationships between different layers to enhance the student's learning process. By leveraging both black-box and white-box distillation techniques, researchers can effectively transfer knowledge from large models to compact models, facilitating advancements in model compression and deployment.

HOW DO SLMs AND LLMs COLLABORATE?

LLMs have demonstrated impressive performance; however, they continue to face several persistent challenges. For example, LLM training requires large amounts of high-quality data, leading to substantial human and financial resource requirements for data collection and annotation. Furthermore, increasing data volume, model complexity, and parameter count amplify the demands for computational power and time resources. In addition, the practical implementation of LLMs is plagued by issues such as privacy and safety concerns in the need of customization.

In addition to using SLMs alone, the collaboration of SLMs and LLMs can achieve better LLM training, fine-tuning, and adaptation. For example, SLMs can offer the capability to generate sizable, top-quality training datasets, reducing the reliance on manual annotation during the training and fine-tuning phases of LLMs. SLMs can mitigate the necessity for direct fine-tuning of LLMs, resulting in significant reductions in computational resource requirements. Through fostering collaboration between LLMs and SLMs, it can also contribute to addressing the issues such as privacy and safety.

By Improving Data Efficiency

LLMs exhibit advanced reasoning capabilities through training on diverse and extensive corpora like C4, RefinedWeb, and The Pile. However, these datasets, primarily sourced from web scraping, often contain substantial amounts of noisy and low-quality text, leading to potential performance drawbacks for LLMs. In response, there is a growing consensus within the research community that prioritizing data quality over quantity is paramount. Accordingly, there is an increasing emphasis on leveraging data selection and pruning techniques to optimize the performance of large models. While rule-based heuristics are commonly used for data filtering, these hand-crafted filters fall short in providing a reliable measure of quality for individual training examples due to the absence of established best practices in this domain.

The utilization of SLMs as evaluation tools for data curation and selection presents a viable solution to address these challenges by pinpointing high-quality subsets within datasets. Figure 3(c1) illustrates the integration potential of small models in data quality assessment processes. For instance, leveraging perplexity scores calculated by a proxy language model can aid in selecting data with higher quality potential[5]; Data Selection with Importance Resampling (DSIR),[6] utilizes an SLM to evaluate instruction data based on criteria such as quality, coverage, and necessity.

In conclusion, the synergistic collaboration between SLMs and LLMs represents a cutting-edge strategy to enhance data efficiency and bolster model performance. By incorporating small models into data curation processes and leveraging their capabilities for precise data selection and optimization, researchers can unlock novel pathways to boost the quality and generalization capabilities of LLMs. This strategic integration underscores a promising direction in advancing the field of language processing systems.

By Improving Computational Efficiency

LLMs excel in generalization but often lack domain-specific knowledge essential for specialized tasks like legal or medical domains. This deficiency necessitates

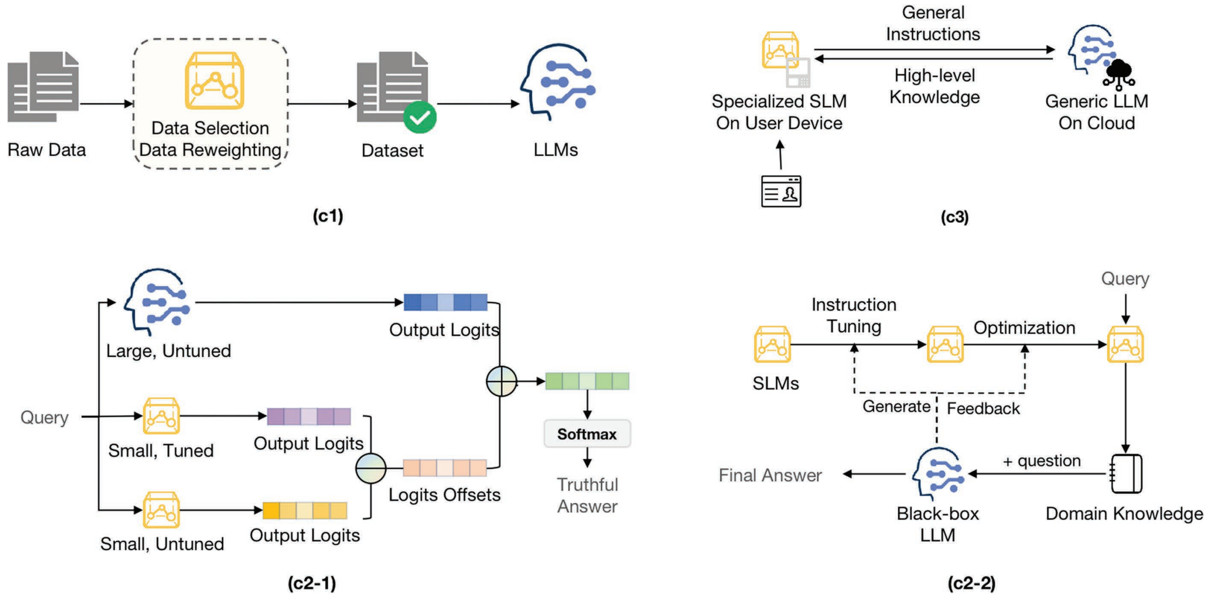

FIGURE 3. Collaboration between SLMs and LLMs: Key benefits include (c1) improving data efficiency, (c2) improving computational efficiency, and (c3) enhancing privacy and safety.

additional fine-tuning or even retraining of LLMs for optimal performance in specific downstream tasks. However, these processes are resource-intensive and may prove unreliable in practical applications, especially with closed-source LLMs like ChatGPT and GPT-4, where model access and modification are restricted.

Recent research has emphasized leveraging SLMs to avoid direct fine-tuning of larger models, thereby reducing computational resources required for retraining. For example, depicted in Figure 3(c2-1), we can utilize a smaller, fine-tuned "expert" model to adjust the output logits of a larger, untuned "base model" during decoding.[7] This approach relies solely on output vocabulary predictions, obviating the need to access the internal weights of the larger model. Additionally, as illustrated in Figure 3(c2-2), we can use less capable SLMs to fine-tune more potent LLMs,[8] enabling broader generalization beyond the limitations of weaker supervisors.

In conclusion, the collaboration between SLMs and LLMs for retraining or tuning offers a solution to resource constraints and the unpredictability associated with conventional methods. By fostering synergy between general-purpose LLMs and specialized SLMs, this approach presents an efficient and cost-effective solution to optimize model performance and address domain-specific requirements.

By Enhancing Privacy and Safety

Many large commercial language models are closed-source and deployed in the cloud, presenting limited privacy protection methods for application programming interface-based services and increasing the risk of data leakage. By using SLMs to facilitate client deployment,[4] the system can access private data and activity logs on devices while processing custom instructions, leveraging the capabilities of large models in the cloud for general instructions [as illustrated in Figure 3(c3)]. This approach eliminates the need to upload user context information, effectively preventing privacy leaks and balancing context privacy with performance.

In addition, the closed-source nature of LLMs, such as ChatGPT and GPT-4, limits the reproducibility of the results. Enabling SLMs to function as agents[2,9] and dynamically select appropriate tools to detect various types of hallucinations, including those in text, code, and mathematical expressions, can improve the capability of hallucination detection significantly.

CONCLUSION AND FUTURE DIRECTION

While SLMs offer the advantage of having fewer parameters and are more suitable for deployment on various end devices, they inherently possess a limited information processing capacity compared to LLMs. This limitation often results in challenges when analyzing and generating responses to complex queries that demand a deeper understanding of context and linguistic nuances. Additionally, SLMs tend to exhibit lower precision relative to their larger counterparts. Their reduced parameter count restricts their ability to capture intricate patterns and subtleties in language data. Consequently, this can lead to responses that are less accurate or consistent, particularly when addressing complex queries.

SLMs should not be perceived merely as substitutes for LLMs. Instead, they should be recognized as complementary tools that have the potential to synergistically enhance the capabilities of their larger counterparts. In considering future directions, it is important to adopt a strategic approach to resource allocation by investing in a diverse range of specialized SLMs tailored to specific needs and objectives. This strategic shift moves away from a reliance on a single, general-purpose LLM for all tasks. By embracing a diversified approach to model selection, we can capitalize on the unique strengths of each model type while simultaneously improving overall efficiency, flexibility, and adaptability.

Another avenue for future research involves leveraging the inherent disparities between SLMs and LLMs to devise sophisticated and resource-conscious fine-tuning methodologies. Rather than directly fine-tuning the computationally intensive LLMs for specific tasks, researchers are exploring alternative strategies that entail fine-tuning the more lightweight SLMs. In this way, SLMs can be strategically fine-tuned to act as effective proxies or intermediaries, indirectly influencing the behavior of their larger counterparts. This innovative technique harnesses the complementary strengths of SLMs and LLMs, leveraging the computational efficiency and adaptability of the former while retaining the vast knowledge and reasoning capabilities of the latter.

ACKNOWLEDGMENTS

The work of Qin Zhang and Ziqi Liu was supported by National Natural Science Foundation of China (62206179).

REFERENCES

1. A. Kundu, Y. C. Fabian Lim, A. Chew, L. Wynter, P. Chong, and R. Lee, "Efficiently distilling LLMs for edge applications," in *Proc. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol.*, 2024, pp. 52–62.

2. X. Cheng et al., “Small agent can also rock! Empowering small language models as hallucination detector,” 2024, *arXiv:2406.11277*.
3. M. Hu et al., “SLM meets LLM: Balancing latency, interpretability and consistency in hallucination detection,” 2024, *arXiv:2408.12748*.
4. K. Zhang, J. Wang, E. Hua, B. Qi, N. Ding, and B. Zhou, “Cogenesis: A framework collaborating large and small language models for secure context-aware instruction following,” in *Proc. 62th Annu. Meeting Assoc. Comput. Linguistic*, 2024, pp. 4295–4312.
5. M. Marion et al., “When less is more: Investigating data pruning for pretraining LLMS at scale,” 2023, *arXiv:2309.04564*.
6. S. M. Xie et al., “Data selection for language models via importance resampling,” in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2023, pp. 34201–34227.
7. A. Liu, X Han, Y. Wang, Y. Tsvetkov, Y. Choi, and N. Smith, “Tuning language models by proxy,” 2024, *arXiv:2401.08565*.
8. C. Burns et al., “Weak-to-strong generalization: Eliciting strong capabilities with weak supervision,” in *Proc. Int. Conf. Mach. Learn.*, 2024, pp. 4971–5012.
9. I. Ong et al., “RouteLLM: Learning to route LLMs with preference data,” 2024, *arXiv:2406.18665*.

QIN ZHANG is an assistant professor at Shenzhen University, Shenzhen, 518055, China. Contact her at qinzhang@szu.edu.cn.

ZIQI LIU is a Master candidate in natural language processing and graph learning at Shenzhen University, Shenzhen, 518055, China. Contact her at liuziqi2022@email.szu.edu.cn.

SHIRUI PAN is a professor and an ARC Future Fellow with the School of Information and Communication Technology, Griffith University, Gold Coast, Qld., 4215, Australia. Contact him at s.pan@griffith.edu.au.