AIS Transactions on Human-Computer Interaction

Volume 16 | Issue 3 Article 4

9-30-2024

Generative AI for Intelligence Augmentation: Categorization and Evaluation Frameworks for Large Language Model Adaptation

Jie Tao
Fairfield University, jtao@fairfield.edu

Lina Zhou
University of North Carolina at Charlotte, lzhou8@charlotte.edu

Xing Fang
College of Applied Science & Technology, Illinois State University, xfang13@ilstu.edu

Recommended Citation
Tao, J., Zhou, L., & Fang, X. (2024). Generative AI for Intelligence Augmentation: Categorization and Evaluation Frameworks for Large Language Model Adaptation. *AIS Transactions on Human-Computer Interaction, 16*(3), 364-387. https://doi.org/10.17705/1thci.00210
DOI: 10.17705/1thci.00210

Transactions on Human-Computer Interaction

Volume 16 Issue 3

9-2024

Generative AI for Intelligence Augmentation: Categorization and Evaluation Frameworks for Large Language Model Adaptation

Jie Tao

Fairfield University, jtao@fairfield.edu

Lina Zhou

The University of North Carolina - Charlotte, lzhou8@charlotte.edu

Xing Fang

Illinois State University, xfang13@ilstu.edu

Recommended Citation

Tao, J., Zhou, L., & Fang, X. (2024). Generative AI for intelligence augmentation: Categorization and evaluation frameworks for large language model adaptation. *AIS Transactions on Human-Computer Interaction*, *16*(3), pp. 364-387.

DOI: 10.17705/1thci.00210

Available at http://aisel.aisnet.org/thci/vol16/iss3/4

Transactions on Human-Computer Interaction

Research Commentary DOI: 10.17705/1thci.00210 ISSN: 1944-3900

Generative AI for Intelligence Augmentation: Categorization and Evaluation Frameworks for Large Language Model Adaptation

Jie Tao
Dolan School of business, Fairfield University
jtao@fairfield.edu

Lina Zhou
Belk College of Business, The University of North Carolina at Charlotte

Xing Fang
School of Information Technology, Illinois State University

Abstract:

Generative AI (GenAI) has transformed how businesses operate and innovate and how individuals learn, live, and work. Large language models (LLMs), a specific type of GenAI, focus on generating human-like text based on user instructions. Like other types of GenAI, LLMs have received wide recognition for their potential to augment human intelligence, but several challenges hinder efforts to realize their full potential in practice. Some notable challenges include not adequately exploring LLM applications beyond chatbots and/or text generation, the difficulty in categorizing various LLM adaptation strategies (particularly regarding human interactions), and the lack of a reference framework for evaluating and selecting LLM adaptation strategies from a human-centered perspective. To address these challenges, we propose a categorization framework for LLM adaptation that features two human-centered dimensions and stage LLM adaptation with respect to when it interacts with human intelligence. Additionally, we introduce an evaluation framework that incorporates a human-centered perspective that goes beyond the common machine-centered measures. Our empirical investigations, in which we use text classification as use cases, not only demonstrate the application of these frameworks but also compare various adaptation strategies. These artifacts and findings provide fresh insights and practical recommendations for selecting effective adaptation strategies to improve the efficacy of LLMs for intelligence augmentation. We further identify future research issues to address current limitations and suggest improvements for the proposed frameworks.

Keywords: Generative AI, Large Language Models, Intelligence Augmentation, Adaptation, Evaluation Framework, Text Classification

Fiona Nah was the accepting senior editor for this paper.

1 Introduction

Generative AI (GenAI) has generated significant buzz across various domains, transforming how businesses operate and innovate and how individuals learn, live, and work. GenAI constitutes a broad category of AI systems designed to generate seemingly new, meaningful content, such as text, images, and/or audio from training data (Feuerriegel et al., 2024). GenAI has made a significant impact and offers potential applications across various domains including business, education, and healthcare (Nah et al., 2023a, 2023b; Preiksaitis & Rose, 2023). Current GenAI and other technologies have the potential to automate work activities that absorb 60 to 70 percent of employees' time today (Chui et al., 2023). Moreover, GenAI will likely bring a fundamental change to the creative processes by which creators formulate ideas and put them into production (Epstein et al., 2023). Intelligence augmentation aims to "enhance and elevate human intelligence, capacity, performance, protection, and quality of life with support from information technology" (Zhou et al., 2023, p. 113). Therefore, GenAI has the potential to change the anatomy of work and augment the capabilities of individual workers. As the next productivity frontier (Chui et al., 2023), the generative capabilities of GenAI epitomize intelligence augmentation.

GenAI encompasses a range of models, with each designed to generate specific types of content. Large language models (LLMs) are a specific type of GenAI that specializes in generating human-like text. Technically, LLMs, such as Llama 3.1, GPT-4o, and Claude 3.5, constitute large-scale, advanced, statistical language models that leverage the decoder part in the transformer architecture that is based on neural networks to understand and generate text (Minaee et al., 2024; Zhao et al., 2023) in contrast to encoder-based models (e.g., BERT, RoBERTa). They are pre-trained on vast amounts of textual data, and can be fine-tuned using task-specific data, which enables them to perform a variety of natural language processing tasks. Yahoo! Finance (2024) recently estimated the global LLM market at US$6.4billion in 2024 and projected it to increase to US$36.1 billion by 2030. The projected rapid growth underscores the increasing adaptation of LLMs for applications ranging from customer service automation to complex data analyses to drive strategic decisions and enhance operational efficiency. Building on the widespread application of ChatGPT, we focus on LLMs in this paper and use them as a lens to examine adaptation strategies for GenAI in augmenting human intelligence (Karabacak & Margetis, 2023; Karanikolas et al., 2023).

In this paper, we first identify several key challenges associated with leveraging GenAI to augment human intelligence and describe them in Section 2. We then propose a categorization framework and an evaluation framework for addressing those challenges in Section 3 and Section 4, respectively. We choose text classification, an underexplored area in GenAI, as the use case to evaluate the proposed frameworks in Section 5. Finally, we discuss our research contributions and implications, offer recommendations, and explore future research issues in Section 6.

2 Human–(Gen)AI Interactions and Challenges

2.1 Human-AI Interactions

Human–AI interactions play a central role in the symbiotic relationship between human intelligence and machine intelligence such as AI or even GenAI (Zhou et al., 2021). Theoretically, we can characterize both human and machine intelligence along spectra with multiple dimensions, such as specialized versus general intelligence, computational depth versus breadth, repetitive versus non-routine/creative decisions, and experiential versus reflective intelligence (Zhou et al., 2021). Nevertheless, as technologies evolve, the positioning of machine intelligence on these spectra may shift. For instance, many believe that GenAI will fundamentally transform the role that machine intelligence plays in the creative process.

Human-AI interactions can be categorized along two dimensions: collaboration and creativity (Li et al., 2024). Collaboration refers to the human/AI involvement in the decision-making process, while creativity characterizes how innovative AI is in collaborations with humans. Accordingly, collaboration and creativity in combination results in four types of AI roles: literature processing tools (where AI processes data with minimal creativity), analysis assistant (where AI formulates opinions based on provided information), creative companion (where AI handles more complex tasks and has to select and use appropriate skill(s) from a multi-skill set), and processing agent (where AI leverages its generative capabilities to process tasks and make decisions). For example, in a recent study, Sayin et al. (2024) discuss AI as an analysis assistant and how they implemented pre-trained open-sourced models (e.g., LLaMA-2 and Mistral) to correct physicians' decisions on a binary classification problem. Their findings suggest that LLMs are sensitive to the prompt design and structure, particularly the context (e.g., examples) used in few-shot learning.

Similarly, researchers have used LLMs as intelligent agents. Sample applications include video games (Park et al., 2023), collaborative research (Hutson & Ratican, 2023), and driving/transportation simulations (Bhattacharyya et al., 2023).

In contrast to the spirit of collaboration, other researchers have examined human-AI interactions via the lens of adversarial learning. For instance, Chiang et al. (2024) recently examined how LLM-powered agents can facilitate collaborative decision making via group discussion where AI or its agents serve the role of a "devil's advocate". Additionally, another perspective on human-AI interactions involves leveraging existing approaches to enhance human intelligence in guiding AI or improving human–AI interactions (e.g., crowdsourcing to formulate prompting templates for better human–AI interactions). For instance, Vicuna-13B (Chiang et al., 2023), an open-source chatbot with 90 percent of the capabilities of ChatGPT, uses conversations between humans and AI trained on ShareGPT as training input.

2.2 Three Challenges in Adapting GenAI

Researchers have widely recognized GenAI for its role in augmenting human intelligence. However, we still face numerous challenges in achieving its full potential for intelligence augmentation. First, GenAI has demonstrated its superior capabilities in generating content and assisting people on creative tasks across various domains, such as textual writing, coding, music composition, and art. However, researchers have devoted far less attention to exploring its capability to understand or process content. Many people misconceive GenAI as merely a chatbot, which oversimplifies its capabilities and limits its perceived utility. Popular tools such as ChatGPT do not fully encompass generative AI's diverse capabilities in augmenting human intelligence across diverse tasks. Take LLMs as an example: besides text generation, LLMs can be applied to language understanding tasks, such as classification and personalization. However, the literature lacks studies that have systematically discussed and empirically investigated language understanding tasks because LLMs (i.e., decoder-based models) struggle to strictly follow human instructions and, thus, produce more "free-form" outputs compared to what encoder-based models produce. To the best of our knowledge, little research in the field of text classification has focused on adaptation strategies of LLMs. These adaptation strategies involve determining the stage and level of context at which human users should interact with LLMs.

Second, even within the LLM realm, the wealth and diversity of models present challenges for human adaptation. We need to extend the current literature, which mostly makes inference on pre-trained "Swiss army knife"-like LLMs (Kocoń et al., 2023), so that users can weigh different options and trade-offs to make informed decisions. Existing studies have either focused on one specific model or model family (Ma et al., 2024a, 2024b; Rahman & Watanobe, 2023; Yeom et al., 2024) or on a specific category of adaptation strategies (Pourpanah et al., 2023; Sheng et al., 2024; Song et al., 2023). Although Liu et al. (2022a) compared few-shot parameter efficient fine-tuning (PEFT) with pure in-context learning, their analysis was limited to a closed source model (GPT-3). Moreover, different adaptation strategies can be used together rather than in isolation. However, there is a lack of guiding framework to help users integrate these strategies effectively to potentially improve model performance. Furthermore, despite the growing capabilities of GenAI, it still relies on human input and/or feedback. For instance, another common misconception about LLMs is that zero-shot learning alone (a machine learning technique where a model is trained to recognize and categorize objects or concepts without having seen any examples of those categories during training) suffice for most tasks, potentially eliminating the need for human intervention. The reality in practice is that zero-shot learning still depends on auxiliary information supplied by humans, such as textual descriptions and attributes. As a result, it often falls short in terms of contextual understanding, accuracy, and specificity. Despite the widespread use of in-context learning, LLMs often struggle to generate high-quality content for specific tasks when relying solely on this approach. As a result, we see a strong need to expand the existing literature to incorporate models from diverse families and, more importantly, to compare the various adaptation strategies of different model families.

Third, given the diverse array of GenAI technologies available, users face challenges in choosing the most appropriate adaptation strategies for specific problems. Existing research mainly focuses on either the technical or design aspects of LLM adaptation (e.g., Liu et al., 2022a), while few explore human aspects with respect to human–GenAI interactions in terms of intelligence augmentation (Karabacak & Margetis, 2023). Thus, users have limited guidance in selecting adaptation strategies. For instance, Feng et al. (2024) proposed CANVIL as a method to analyze the effect of incorporating design thinking about user experience factors into LLM-based products. However, the authors mainly addressed how to productively formulate system and user prompts when using LLMs. Ibrahim et al. (2024) proposed a framework for evaluating

human interaction with LLMs that specifically addresses potential harm and risks. This framework comprises three stages: risk identification, context characterization, and parameter selection. To the best of our knowledge, this study represents an early effort to incorporate human input into LLM evaluations. Nevertheless, the proposed guidelines primarily focus on harm and risk and may not address other human aspects of LLM adaptation. Based on an extensive survey on evaluating LLMs, Chang et al. (2024) emphasized the importance of benchmarks specific to different downstream tasks, human-in-the-loop evaluations, and successful and failed LLM adaptations. However, the survey focused on LLMs themselves rather than LLM adaptation and did not provide empirical results. In terms of inference on pre-trained models, Ahmed and Devanbu (2023) compared few-shot against zero- and one-shot learning on code summarization tasks. Their findings show that few-shot learning with carefully selected demonstrations (e.g., sample code from the same project) can outperform full fine-tuning on smaller models. The findings also suggest that the intrinsic homogeneity of the demonstration(s) and the target can improve model performance. Other researchers have exclusively focused on fine-tuned models. For instance, Ding et al. (2023) compared fine-tuning and four representative PEFT techniques (adapter based, low-rank adaptation or LoRA, prompt tuning, and prefix tuning) along the dimensions of performance, convergence, efficiency, combinability, and scalability in over 100 NLP tasks. Among the PEFT techniques, LoRA yielded the best performance, convergence, and efficiency. Only a handful of other recent studies have compared using pre-trained versus fine-tuned models. For instance, Weyssow et al. (2024) compared full fine-tuning on small models (e.g., CodeT5), PEFT on LLMs (e.g., LoRA on CodeLLaMA), and in-context learning on LLMs in the context of code generation. The findings indicate that PEFT, specifically LoRA, outperformed both full fine-tuning on small models and in-context learning on LLMs. Additionally, quantized LoRA (QLoRA) not only did not compromise performance but also outperformed LoRA in certain tasks with up to a two-fold reduction in resource usage. Another study (Patwa et al., 2024) compared zero-shot/in-context learning with LoRA on the Vicuna model for different text-classification tasks. Their results suggest that carefully curated in-context learning can approach the performance of LoRA.

In summary, our understanding of how humans can effectively interact with GenAI to augment human intelligence remains fragmented. By addressing these challenges and related misconceptions, we can improve the effectiveness and efficiency of intelligence augmentation across various domains.

3 A Categorization Framework and Stages of LLM Adaptation

Despite the increasing availability of pre-trained LLMs, it remains challenging for these models to consistently produce high-quality content for specific user tasks in a context that concerns augmenting human intelligence. As a result, user supervision is essential to facilitate effective human-LLM collaboration. To help users select strategies to adapt LLMs, it would be instrumental to propose a categorization framework and situate human interactions in the LLM pipeline stages.

3.1 A Framework for Categorizing LLM Adaptation: Human-centered Dimensions

We introduce two dimensions that focus on human intelligence for categorizing LLM adaptation strategies: customization and context (see Figure 1).

- *Customization* refers to tailoring a pre-trained LLM to better meet the specific needs or requirements of a particular application or user. To support customization, we need to obtain user requirements such as specific needs or preferences of human users, domain-specific data with sufficient size, and relevant domain knowledge. Depending on the degree of customization, human users can directly use pre-trained LLMs without fine-tuning or perform various degrees of fine-tuning to those models for their specific downstream tasks.
- *Context* refers to the information given to the LLM to guide its responses or outputs. Typical context information includes user intent or needs, relevant high-quality data, and background information or domain-specific knowledge. Like customization, context can vary from minimal to comprehensive and cover a spectrum from no context to full context. The framework that we propose highlights the collaboration or fusion of human intelligence and machine intelligence (e.g., LLMs).

The combination of these two dimensions, customization and context, can result in various adaptation strategies. Figure 1 illustrates each intersection with sample strategies, some of which we applied in the case study that we describe in Section 5. For instance, from the customization perspective, human users can determine whether to directly use a pre-trained model or fine-tune the model for specific downstream

tasks based on their technical knowledge and resources. From the context perspective, human users can choose to provide no context to full context, which may not only help address the intrinsic issues of LLMs such as hallucination but also assist with prompt engineering. We note that both dimensions indicate that human intelligence plays a role in LLM adaptation. For instance, the strategy in the bottom-left region, pre-trained + zero-shot (inference), requires the least human intelligence. In contrast, the strategy in the top-right region, full fine-tuning + RAG few-shot (inference), demands the most human intelligence, where RAG refers to Retrieval-Augmented Generation.

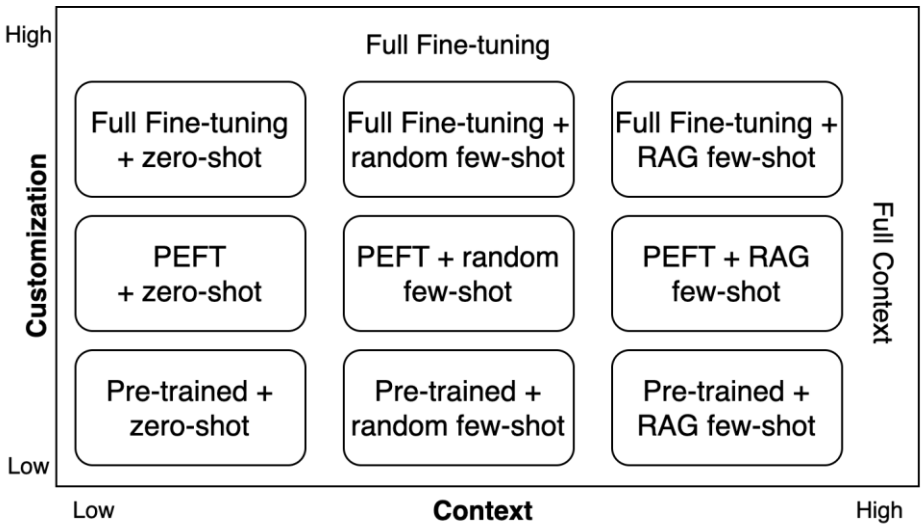


Figure 1. A Categorization Framework for LLM Adaptation

3.2 LLM-adaptation Stages: "When" to Interact with Human Intelligence

Based on the view that "when" represents a crucial aspect of intelligence augmentation (Zhou et al., 2023), we consider human-LLM interaction an important consideration in LLM adaptation. Specifically, as different levels of human intelligence play a role in the LLM-adaptation process, we segment it into three stages: training (or pre-training), customization, and inference. Among them, inference represents an essential stage because it directly generates the responses to user prompts; in contrast, the other two stages are optional. Technological powerhouses, such as Google, Meta, OpenAI and Nvidia, usually perform training on LLMs, which in turn becomes transparent to the individual users unless necessary. In addition to the technical knowledge and resources previously mentioned, customization depends on whether the model is open-sourced (i.e., its architecture and parameters are publicly available). Model adaptation comprises two key phases—the customization phase and the inferencing phase—and each phase requires different levels of human intelligence and involvement. The model customization phase requires substantial human intelligence to tailor a model to specific requirements and involves tasks such as data preparation, parameter adjustment, and algorithm selection. The inferencing phase also requires a significant level of human intelligence: besides the reasoning behind the conversational process and means of interactions via prompt engineering, users also need to decide how much context they should provide to a model. To provide users with concrete guidance on LLM adaptation, we follow the sequence of stages to introduce specific adaptation strategies in the remainder of this section.

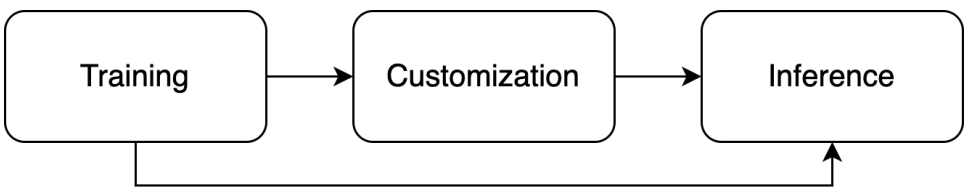


Figure 2. LLM-adaptation Stages

3.2.1 Training

Training, or pre-training, a LLM involves several key steps. First, we need to collect a diverse and extensive dataset from various sources such as the Web to ensure that it covers a wide range of topics for comprehensive learning. Second, we need to clean and prepare the data to remove noise and irrelevant content, which is crucial for maintaining the quality of the training data. Tokenization breaks down text into manageable units for the model to process effectively. Third, we need to choose an appropriate architecture, often a transformer-decoder based neural network, for scalability and to ensure the LLM can handle long-range dependencies in text embeddings. Fourth, a well-defined objective function guides training by maximizing the likelihood that the model will predict the next token accurately. Lastly, during pre-training, the model learns to predict tokens across varied contexts through multiple epochs. Evaluation measures for its predictability are crucial and they include perplexity, coverage, and diversity.

Instruction-tuning, also known as instruction fine-tuning, constitutes a supervised fine-tuning technique that re-trains pre-trained LLMs by following instructions. Typically, the model is trained using input-output pairs, which enables it to learn the specific task that the input (instruction) entails. The key difference between instruction tuning in the training phase and different fine-tuning techniques in the model-customization phase is that instruction tuning usually involves widely varied tasks and widely varied instructions, whereas fine-tuning in model customization typically trains the model for a specific downstream task. Thus, the instructions used in model customization are typically the same or identical for a given downstream task.

Another method to train LLMs such as ChatGPT includes reinforcement learning with human feedback (RLHF), which incorporates direct feedback from human users. In this approach, the model interacts with users who provide feedback on the model's responses. This feedback can include ratings (e.g., thumbs up/down), corrections, or specific instructions to improve the quality or relevance of responses. RLHF can improve a model's performance iteratively by reinforcing behaviors that receive positive feedback and adjusting those that are less or not well received. Unlike traditional reinforcement learning, which often relies solely on simulated environments or predefined rewards, RLHF leverages real-time human interaction to guide learning. This method helps refine a LLM's ability to generate contextually appropriate responses across widely varied conversational scenarios, which makes it more responsive and adaptable to user needs and preferences in real-world applications. By continuously learning from human feedback, an LLM evolves to better understand nuances in language and context, enhancing its overall effectiveness as a conversational agent.

Pre-training and instruction tuning leverage human intelligence from the data used in these stages. During the respective stages, the model is trained to extract and learn patterns from the training datasets, and then uses these patterns to improve its reasoning capabilities in various tasks. In the RLHF stage, the model output, and implicitly its behaviors, is aligned with expectations from human experts and regular users.

3.2.2 Model Customization

The next stage in LLM adaptation, model customization, includes full fine-tuning and PEFT. Full fine-tuning refers to updating all model weights for a specific downstream task, which has yielded superior (sometimes state-of-the-art) performances with smaller models. However, full fine-tuning becomes very resource intensive with larger models. Consequently, many studies have called for efficient fine-tuning mechanisms to transform general purposed LLMs to specific downstream tasks. One can further categorize fine-tuning into full fine-tuning and PEFT with the latter being more efficient than the former without much compromise in performance. For instance, Raiaan et al. (2024) suggest there is a high demand in the field of education for effectively fine-tuning models to acquire new skills rather than simply relying on pre-trained models.

We can further categorize the four techniques in the PEFT paradigm (i.e., adapter based, LoRA, prompt tuning, and prefix tuning) into four main categories; namely, addition based, specification based, reparameterization based, and hybrid methods (Ding et al., 2023). Addition-based methods include adapter-based and prompt-based tuning. Although both have been proven effective and parameter efficient, adapter-based tuning has difficulty sharing and reusing adapters, while prompt-based tuning converges slowly. Specification-based methods target a subset of parameters that are "mission critical" to the downstream task rather than introduce new parameters during the fine-tuning process. However, they usually work well on smaller models (<1 billion parameters). Reparameterization-based methods, particularly the LoRA family of methods, introduce low-rank adaptation of model parameters by decomposing the weight matrix into two low-rank matrices.

According to a recent study that comprehensively compared model-customization techniques, LoRA emerged as the superior option (compared to other PEFT techniques) with respect to performance, convergence, and other dimensions (Ding et al., 2023). However, we can further improve fine-tuning efficiency by reducing the memory footprint and training time using techniques such as QLoRA (Dettmers et al., 2023). Compared to LoRA, QLoRA saves memory footprint of fine-tuning via the 4-bit NormalFloat (nf4) data type, double quantization on the frozen model parameters, and paged optimizers while preserving the performance from a 16-bit LoRA model. QLoRA not only enables users to fine-tune larger models on smaller GPUs but also enables smaller models to perform on par with larger ones. To demonstrate the proposed framework, we focus on LoRA and QLoRA as the main model-customization techniques in this study.

During the model-customization phase, models should acquire new reasoning skill(s) for specific respective downstream tasks as they do in the pre-training and instruction tuning stages. In contrast to the pre-training stage, however, human users need to decide if they need to customize a model and, if so, determine the most appropriate model-customization technique for a certain downstream task. Our proposed adaptation and evaluation framework can provide some guidance on this issue.

3.2.3 Inference

In general, inference refers to entering prompts as instructions for certain tasks. LLMs generate responses by drawing conclusions or making predictions based on the instructions and the knowledge learned from pre-training and/or model customization. Human users can make inferences on both pre-trained and fine-tuned models using zero-shot and/or few-shot learning. Specifically, the “shot” (or “demonstrations”) refers to the context or examples that the model can refer to or reason on. Thus, some people refer to few-shot learning as in-context learning.

The most popular inference type is zero-shot inference, where human users enter the instruction only, without any context or demonstrations. But, given the very limited context, models usually perform at an inferior level. Prompt engineering, which requires human users to provide carefully curated instructions, represents one possible way to combat such an issue. With respect to prompt engineering, some studies (e.g., Navigli et al., 2023) have suggested that including both examples (i.e., few-shot learning) and guard rails (i.e., restrictions on the model output/generated contents) can effectively decrease the intrinsic biases associated with respective models. In addition, researchers have used crowdsourcing (e.g., ShareGPT) to formulate effective prompt templates as well.

In-context learning for LLMs involves the capability to adapt and refine their responses based on an ongoing dialogue with users. Different from traditional static models that generate responses independently of previous interactions, models that employ in-context learning can memorize conversation histories and use them to generate more relevant and coherent replies. This approach allows the models to understand and respond contextually by considering not only the immediate input but also the broader conversation context. In-context learning is achieved with techniques such as context windowing, where the model retains a limited history of previous interactions, or more sophisticated methods that dynamically update context representations over longer dialogues. This ability enables LLMs to offer more personalized and coherent responses, which can improve user engagement and satisfaction by fostering a more natural and continuous interaction. Additionally, in-context learning supports applications that require sustained dialogue comprehension, such as customer service chatbots or educational assistants, where maintaining continuity and relevance across multiple turns of conversation is essential for effective communication and task completion. Prompt design, which includes demonstration organization and instruction organization, is a critical factor for success of in-context learning. Nearest neighbor-based methods on semantic similarity have been prevalent in selecting demonstrations due to their unsupervised nature (Liu et al., 2022b). The order of demonstrations also matters. Researchers have suggested that positioning the closest example in the rightmost demonstration helps with model generation. Selecting context or demonstrations in in-context learning poses no easy task. Studies (e.g., Zhao et al., 2024) have indicated that merely increasing the number of demonstrations or examples in a prompt does not necessarily improve a model’s performance. Additionally, even with carefully selected demonstrations, in-context learning cannot enable models to match instructions as it follows the behaviors of full fine-tuning.

RAG optimizes efforts to generate LLMs by referencing an authoritative knowledge base beyond the training data (Lewis et al., 2020). It typically incorporates a non-parametric (i.e., retriever) and a parametric (i.e., generator) component. The retriever encodes user queries (e.g., model inputs and instructions) and the external knowledge base, and the generator is a generative model (e.g., seq2seq model or LLMs).

Sometimes, the encoded knowledge base is stored in a vector database for computational efficiency. RAG has been proven to be efficient, particularly in knowledge-intensive tasks where human intelligence is insufficient without access to external knowledge sources. Even though RAG seemingly focuses on the retriever side, the literature suggests that techniques such as prompt engineering and generator fine-tuning would further improve performance (Nashid et al., 2023). To demonstrate our proposed framework, we focus on zero-shot, in-context learning, and RAG as the inference techniques.

Chain-of-thoughts (CoTs) represents a relatively new prompting technique that works with LLMs to elicit multi-step reasoning from them (Wei et al., 2022). CoTs allows LLMs to maintain coherence and context across multiple interactions or narrative segments. Unlike simpler models, which may respond independently to each input, models that employ chain-of-thoughts learning retain and build on the context established in previous conversation turns or prompts. Techniques such as context chaining and memory mechanisms facilitate this capability and enable the model to recall and integrate relevant information to produce more coherent and contextually relevant responses. Traditionally, CoTs require multi-step answer examples to achieve state-of-the-art performance in difficult mathematics and symbolic-reasoning problems. However, researchers have argued that LLMs represent decent zero-shot reasoners where the zero-shot inference performance can be improved by inserting a “let’s think step by step” statement into the prompts (Kojima et al., 2022).

Similar to model customization, selecting the most appropriate inference technique for a specific problem represents a tall requirement for human users. We again believe our framework can shed some light on it. Additionally, prompt engineering represents a critical phase to infuse human intelligence in LLMs to solve problems. RAG requires models to use external related knowledge bases, which humans typically generate, to solve certain problems, while CoTs let human intelligence and LLMs contribute in a collaborative fashion to enhance the models’ reasoning capabilities.

4 An Evaluation Framework for LLM Adaptation

To address the challenges in choosing adaptation strategies that meet users’ needs or preferences, we propose an evaluation framework for LLM adaptation. More importantly, given the need to involve human intelligence in adapting LLMs (see Figure 1), we categorize evaluation measures for LLM adaptation into two dimensions (human centered and machine centered) and propose evaluation measures for each dimension. These two dimensions complement each other.

4.1 Human-centered Dimension

This dimension refers to an LLM adaptation’s design, process, or outcome that prioritizes human needs, experience, and wellbeing. In particular, the evaluation dimension focuses on human-model interaction rather than the impact of models on individual users or broader society effects, as is often discussed in the literature on augmented intelligence. We introduce the following measures for this dimension:

- **Human control**: This measure refers to how much control human users have over the generated contents or, conversely, how much effort users need to expend to post-process generated content to meet their needs and expectations. Due to the autoregressive nature of LLMs, human users have limited control over the generation process, beyond adjusting a few hyperparameters (e.g., temperature, number of beams). Therefore, human users need to conduct a post-hoc evaluation to determine how well the generated content aligns with their instructions when assessing human-AI interactions. Additionally, due to the hallucination and performance issues in the machine-centered dimension (see Section 4.2), one may deem manual post-processing necessary. For instance, post-processing to mitigate hallucination may include extracting implicit information from the generated content (e.g., labels) and mapping it to the appropriate solution space (e.g., predefined labels) to ensure alignment with human intelligence.
- **Ease of use**: This measure refers to how easy users find it to interact with a model or whether users can interact with it without extensive training. Indeed, the technical knowledge and skills that one needs to perform inferencing on and/or modify models represent a major hindrance to LLM adaptation. Such knowledge and skills include understanding LLMs, familiarity with model architecture and functioning, hyperparameter tuning, prompt design, experience with frameworks and libraries, and programming proficiency.

- **Prompt complexity**: This measure falls under ease of use; however, we consider it separately because LLMs can handle a range of user-formulated prompts, and the quality of engineered prompts can significantly impact model performance. Different models have varying requirements for prompt complexity. Thus, prompt complexity remains an important factor to consider for evaluation even though an in-depth discussion on prompt engineering falls beyond the scope here. In this study, we operationalize prompt complexity as the number of examples in a prompt (0 versus 3).
- **Catastrophic forgetting resistance**: This measure refers to the situation where a model, after a user fully fine-tunes it for a specific downstream task, might lose some capabilities that it acquired earlier (e.g., during pre-training). This measure represents a crucial one for maintaining the continuity and reliability of the LLM's performance over time, as human users may need to task a customized LLM for a different purpose that it has been customized for. LLMs are susceptible to catastrophic forgetting, whereas one of the key motivations for developing PEFT techniques is to prevent catastrophic forgetting. Catastrophic forgetting resistance is typically measured by applying a fine-tuned LLM on a task different from what it has been fine-tuned for, then comparing the performance before and after fine-tuning.

4.2 Machine-centered Dimension

This dimension focuses on the technical and functional aspects of an LLM, specially whether the model produces accurate and reliable results for its intended tasks, in alignment with the user's needs and context. We highlight the following quantitative measures within this dimension:

- **Performance:** Performance is the key dimension for comparing LLM outputs across various adaptation strategies. It assesses how well the LLM-generated content matches human expectations, knowledge, behavior, and preferences. For example, in a text-classification problem, performance is measured by how well the generated labels match the actual or ground-truth labels. Since we selected text classification as the use case for this study, we adopted commonly used metrics (e.g., accuracy) to measure performance.
- **Computational efficiency**: This dimension captures the fine-tuning and inference time (duration required to fine-tune and inference specific models) as well as the memory footprint of fine-tuning (VRAM needed on GPU). It is another important evaluation measure for choosing adaptation strategies given that implementing these strategies may place significant demands on hardware and other resources. For instance, it is more convenient to fine-tune a full-sized model on a single GPU if the model and required data can fit onto the GPU's VRAM rather than distribute them across different GPUs. We operationalize computational efficiency as the inference time (minutes) only, while treating fine-tuning time and required VRAM as the computational overhead. We make this design decision because inference time applies to all adaptation strategies and plays an essential role in using LLMs, whereas computational overhead is relevant only during fine-tuning.
- **Robustness to hallucinations**: With respect to LLMs, hallucinations refer to instances when they generate factually incorrect or nonsensical output even though it may sound plausible for a given problem. In addition to generating nonsensical or inaccurate content, LLMs can generate content in different variations, or even in an implicit form that requires post-hoc deduction. For instance, LLMs may output classification labels as a sentence, a different spelling variant, or an encoding (e.g., 1 for "positive"); they may even produce variations across different instances. In our study, we operationalize hallucinations as the model generating contents that do not contain any of the predefined labels in the given problem.

5 Use Cases: Text Classification

We conducted an empirical investigation into LLM adaptation for different use cases, focusing on three main objectives: 1) to showcase how to address the challenges associated with adapting LLMs (see Section 2), 2) to apply the evaluation framework that we propose in Section 4 for comparing LLM adaptation strategies, and 3) to provide fresh insights and actionable recommendations for selecting effective adaptation strategies based on the categorization framework that we introduce in Section 3.

One can view LLM adaptation as a decision-making problem. In this study, we adopt Herbert A. Simon's (1976) decision-making model, which provides a systematic approach to the process. The model comprises

three key stages: 1) intelligence (focusing on identifying the problem or opportunity and gathering relevant data), 2) design (generating and evaluating various potential solutions to the problem), and 3) choice (selecting the best solution from the alternatives generated during the design phase based on specific evaluation measures).

5.1 Intelligence: Identifying the Problem

5.1.1 Problem Identification

Text classification represents a prevalent task in NLP. However, few studies have used LLMs for text-classification tasks. We can partly attribute this scarcity to the difficulty associated with achieving state-of-the-art performance with encoder-based models. Among the studies that have used LLMs for text classification, some have focused on in-context learning with close-sourced LLMs. For instance, Sun et al. (2023) introduced clue and reasoning prompting (CARP) framework for addressing various text-classification problems using a GPT-3 model in both zero- and few-shot settings. We adopt a similar strategy for selecting demonstrations/examples in our few-shot setting. Krugmann and Hartmann (2024) investigated sentiment analysis, a specific type of text-classification problem, using both close-source and open-source LLMs (e.g., GPT-3.5/4 and LLaMA-2) and observed that LLaMA-2 delivered exceptional classification performance. Consequently, we believe it is essential to include at least one model from the LLaMA family in this study.

Binary classification is the most prevalent type of text-classification. One well-studied example is sentiment analysis. To enhance the generalizability of our findings, we chose a different classification problem — content moderation- as well as a multiclass sentiment analysis problem as use cases for our experiments.

5.1.2 Data Selection

We selected two datasets that correspond to the above text-classification types. The first dataset consists of financial news articles with sentiment labels (Malo et al., 2014). We selected this dataset because it contained a high ratio of one-sentence contents, which LLMs often find difficult to classify (Krugmann & Hartmann, 2024). The dataset contained 5,842 finance news articles along with their corresponding sentiment labels (i.e., positive, neutral, and negative). On average, each news article in the dataset comprised 32.44 tokens. We first performed random stratified sampling to reserve 50 percent of the data for random sample selection and RAG. We reserved 40 percent for model fine-tuning and 10 percent for testing.

The second dataset included social media posts, their associated metadata, and information indicating whether each post had been moderated (Wang et al., 2023). Content moderation represents a common process for intervening in user-generated content on social media platforms. Wang et al. (2023) collected the data in the dataset from 40 subreddits on Reddit each day for over two months across four different domains. We selected this dataset because it addresses LLMs often face challenges in classifying informal social media content (Krugmann & Hartmann, 2024). The dataset contained 17,022 posts evenly split between moderated and unmoderated categories. We first concatenated each post's title and body and then performed random stratified sampling given the dataset's large size. Specifically, we reserved 75 percent of the data for the sample selection, 20 percent for fine-tuning, and five percent for testing. On average, each post in the dataset contained 91.86 tokens.

5.2 Design: Generating and Evaluating Alternatives

5.2.1 Model Selection

The findings from the related literature (see Section 2) support our decision to focus on the LoRA family in this study. Moreover, unlike earlier work (Ding et al., 2023), we emphasize more up-to-date and larger models (Phi-3/TinyLlama versus RoBERTa/T5). Furthermore, we examined in-context learning beyond zero-shot inference.

LLMs require significant GPU VRAM resources for fine-tuning and demand substantial resources to achieve a high token-per-second rate during inferencing. For instance, the latest LLaMA-3-70B model requires Nvidia A100/H100 with 40 or 80 GB VRAM for fine-tuning. However, few have access to data center-level GPUs or multi-GPU setups. As a result, users can benefit more from smaller LLMs which one can tune on consumer-level GPUs with no more than 24 GB VRAM. For feasibility and efficiency, we chose two smaller

LLMs (namely, Phi-3-mini and TinyLlama) that one can fine-tune on consumer-level GPUs without the need for quantization. We selected these two specific models for two additional reasons:

- Launched by Microsoft (Abdin et al., 2024), Phi-3-mini soon has become a popular model (with over one million downloads on the Hugging Face model repository (Hugging Face, 2024). Microsoft claims that Phi-3-mini outperforms models twice its size (i.e., the most popular LLaMA-3-8B) on various open-ended generation tasks (Bilenko, 2024), such as reasoning and language understanding tasks.
- TinyLlama (Zhang et al., 2024) represents the smallest model in the LLaMA model family. Apart from architectural differences (encoder- versus decoder-based), we believe one billion parameters serves as a suitable threshold since many decoder-based models contain billions of parameters. Additionally, since many other LLMs, including models in the Bloom family (1.1B and 1.7B variants), have a similar number of parameters, we believe that TinyLlama is representative of this category.

Table 1. Model Hyperparameters

Adaptation strategy	Phi-3-mini	TinyLlama
Customization	Training epochs: 2 batch size: 4 Learning rate: 2e-5 Weight decay: 0.01 Alpha: 16 R: 16 Dropout: 0 Warm up ratio: 0.01 Max_length: 4096 Quantization: 4bit (nf4) LR scheduler: linear Optimizer: paged AdamW 8bit	Training epochs: 2 batch size: 4 Learning rate: 2e-5 Weight decay: 0.1 Alpha: 32 R: 32 Dropout: 0 Warm up ratio: 0.03 Max_length: 4096 Quantization: 4bit (nf4) LR scheduler: linear Optimizer: paged AdamW 8bit
Inference	Task: text generation Max new tokens: 3/12	
1) The respective original tech reports suggested all parameters 2) We set max_new_tokens as 3 for the finance sentiment analysis experiment since the model only needed to output the label (positive/negative/neutral) but as 12 since we expected the model to output following the template “the social media post is {} by human”.		

We performed fine-tuning and inferencing on Google Colaboratory with a Nvidia T4 GPU (16GB VRAM). We believe this setup better reflects the computational environment of typical users, ensuring the reported computational efficiency is relevant. We detail the hyperparameters for fine-tuning and inferencing in Table 1.

5.2.2 Evaluation

We recruited three human raters (all graduate students) to independently assess the measures in the human-centered dimension. Two raters had advanced technical expertise, including advanced Python programming and deep learning, while the other had minimal technical knowledge such as basic programming skills. We tasked the raters with ranking the selected models with respect to each dataset (1 = the highest rank (best) and 4 = the lowest rank (worst)). We defined the evaluation measures and instructed the raters on how to apply them, along with providing general rating guidelines to mitigate any pre-existing biases. The inter-rater reliability (Fleiss' kappa (Falotico & Quatto, 2015)) was 0.84.

We developed automated methods for the measures in the machine-centered dimension. Since we considered different inference strategies (zero-shot and random 3-shot and RAG 3-shot as in context learning), we averaged the respective measurements for all zero- and few-shot variants to represent each corresponding adaptation strategy. For example, we calculated the average accuracy of the phi-3-mini model under LoRA for the multiclass sentiment analysis problem as (0.8034 + 0.7812 + 0.7350) / 3 = 0.7732, ranking it the highest among all strategies.

We also assigned an overall ranking to each strategy based on its average score across all measures. For instance, we calculated zero-shot learning’s overall rank in human-centered dimension as (4 + 2 + 1 + 3) /

4 = 2.375, which ranked it the lowest among all four strategies. The raw results from the case studies appear in an online appendix (see https://github.com/DrJieTao/LLM_research/blob/main/Online%20Appendix.pdf).

5.3 Choice: Choosing an Alternative

Based on the evaluation results, we present our findings and provide recommendations for selecting strategies to adapt LLMs.

5.3.1 Human-centered Dimension Results

Since we did not observe any variations in the measures of the human-centered dimension across different model-dataset combinations, we report the overall ranking of each measure in Table 2 and summarize the results below.

Table 2. Ranking Adaptation Strategies in the Human-centered Dimension

Evaluation measure	Zero-shot learning	In-context learning	LoRA	QLoRA
Human control	4	3	**1**	**1**
Prompt complexity	**1**	4	**1**	**1**
Ease of use	**1**	2	3	3
Catastrophic forgetting resistance	**1**	**1**	**1**	**1**
Overall ranking	3	4	**1**	**1**

- **Human control**: for customized models (both LoRA and QLoRA), we found they effectively followed human instructions. For example, we could use regular expressions containing the training templates to extract labels from their outputs easily. In contrast, the outputs from pre-trained models varied in how well they followed human instructions, as labels could differ in spelling ("pos" instead of "positive" or "moderation" instead of "moderated"), be encoded differently (using 1 instead of "positive"), or contain extra information ("82.63 percent positive"). In some cases, the model provided incomplete reasoning (limited by the max_new_tokens), making the label obscure. Customized clean-up and mapping functions are needed to effectively extract labels from the pre-trained models. We also observe that the Phi-3-mini model could follow human instructions better than the TinyLlama model, particularly in quantized cases, regardless of whether one customized it or not.
- **Ease of use**: users need a deeper level of technological knowledge to fine-tune models compared to simply using pre-trained models. Loading a quantized model for QLoRA requires some additional technical knowledge, compared to using full-sized LoRA. Fortunately, this knowledge requirement becomes minimal with the help of software tools (e.g., Unsloth AI (Hanchen, 2024)). However, as we discuss earlier, users also need understand how to (programmatically) extract labels from content generated by pre-trained models.
- **Prompt complexity**: we note that few-shot learning, both random few-shot and RAG, increased inference time without necessarily improving performance on the larger phi-3-mini model. Three explanations for this finding are as follows. First, due to the model's default context window size, adding more examples to the prompt may exceed this limit, resulting in an "incomplete" (truncated) prompt that could negatively impact performance. To test this explanation, we performed 2-shot learning in both random and RAG fashion and obtained slightly better results. Second, prompts used in PEFT do not contain any examples nor placeholders for examples. Consequently, after fine-tuning, the newly learned parameters may not adapt to the prompt's altered structure (with added examples), which could explain the observed decrease in performance in fine-tuned models with few-shot learning. Third, the phi-3-mini model was already performing at a cutting-edge level. The smaller TinyLlama model showed improved performance with additional examples in the prompt, and RAG outperformed random 3-shot inferencing. This could be because, given its limited parameters, the TinyLlama model benefits from the additional information in the examples, and RAG provides richer information compared to random 3-shot. Additionally, compared to sentiment analysis, human moderation-detection in social media is less common during the pre-training process. As a result, in this use case, in-

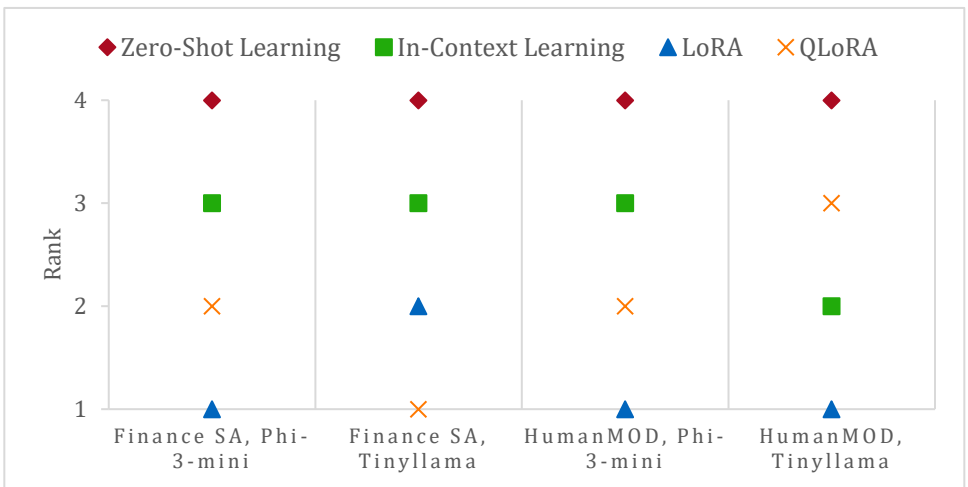
context learning or few-shot learning, improved both model performance and robustness against hallucinations.

- **Catastrophic forgetting resistance**: given that both LoRA and QLoRA introduce additional parameters to the models, and only these new parameters are updated during backpropagation, removing them restores the model to its original pre-trained state, allowing it to perform tasks as it did prior to fine-tuning. We tested the fine-tuned models, after removing the additional parameters introduced by LoRA/QLoRA, on non-text classification tasks (such as text summarization and Q&A generation) and found that they performed similarly to the pre-trained versions.

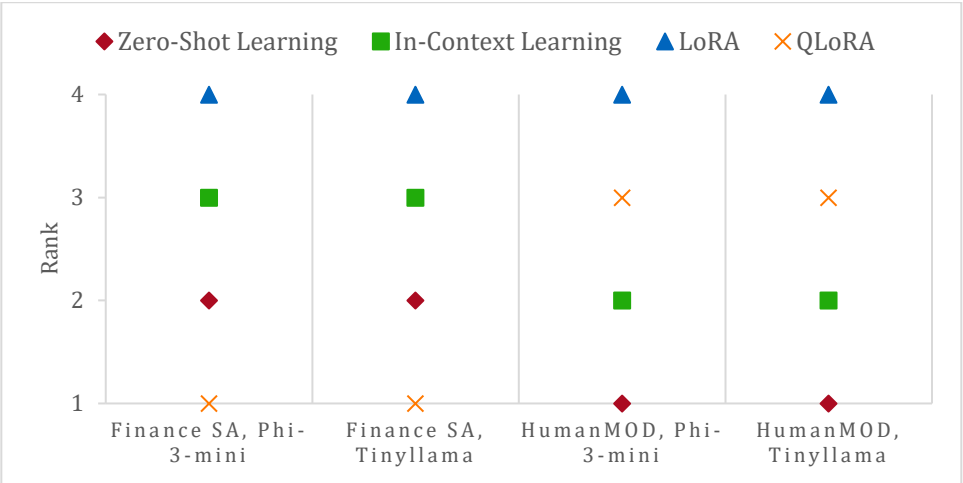
5.3.2 Machine-centered Dimension Results

To compare rankings for each measure across different model-dataset combinations, we calculated their simple averages. We present the rankings of each evaluation measure in the machine-centered dimension individually in Figure 3, and collectively in Tables 3 and Table 4.

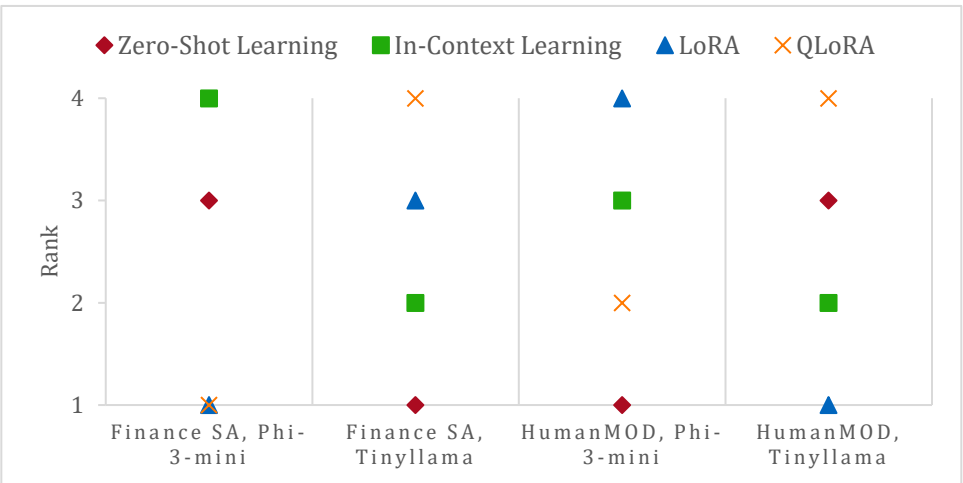
- **Performance**: LoRA achieved the best performance in three model-dataset combinations and placed second in another. Thus, it emerged as the most effective adaptation strategy in terms of performance, followed by QLoRA, in-context learning, and zero-shot learning. We also note that employing RAG did not improve performance compared to random 3-shot learning. We also observed slightly lower performance in the HumanMOD results compared to the Finance SA counterparts even though the latter involves less complexity (multi-class classification vs. binary). As the literature has suggested (Laban et al., 2023), we can attribute this difference in performance to the fact that the data that people use to train models on typically includes human moderation-detection problems less frequently than sentiment information.
- **Computational efficiency**: intuitively, zero-shot learning was the most efficient on the HumanMOD dataset, while QLoRA demonstrated the highest efficiency on the Finance SA dataset. Additionally, we identified LoRA as the least computationally efficient adaptation strategy. Thus, even though zero-shot learning achieved the best overall ranking on computational efficiency, one should consider QLoRA.
- **Robustness to hallucination**: zero-shot learning received the highest overall ranking followed by LoRA and then a tie between in-context learning and QLoRA. Upon further investigation, QLoRA ranked last when we employed the TinyLlama model. This may be because quantization on such a small model increased the likelihood of hallucination, which the high hallucination ratios on the 4-bit quantized pre-trained model evidence. Thus, for small-sized models, heavy quantization might not be a viable option with respect to resisting hallucination.



(a) Performance



(b) Computational Efficiency



(c) Robustness to Hallucination

Figure 3. Adaptation Strategy Ranking by Individual Measures in the Machine-centered Dimension

Table 3. Overall Ranking of Adaptation Strategies by the Machine-centered Dimension

Evaluation measures	Zero-shot learning	In-context learning	LoRA	QLoRA
Performance	4	3	**1**	2
Computational efficiency	**1**	3	4	2
Robustness to hallucination	**1**	3	2	3

Understanding each evaluation dimension enables users to choose the most appropriate adaptation for specific applications. However, combining different dimensions may lead to trade-offs. For instance, one might need to balance performance with computational efficiency or balance training time with inferencing time. Moreover, the measures to be chosen often depend on the specific requirements of the task to be performed and the data to be used. Therefore, we briefly compare the various combinations of evaluation dimensions. We also assume that performance is a key factor in selecting adaptation strategies, so we consider the following combinations that include performance.

- **Performance and computational efficiency (P + CE)**: when considering both performance and computational efficiency, QLoRA significantly outperformed the other adaptation strategies, and zero-shot learning ranked last (it fell not far behind LoRA and in-context learning). These findings disprove the misconception that “zero-shot learning suffices for most tasks”. Additionally, the tie

between LoRA and in-context learning gives users some flexibility: they can make the decision based on other factors, such as the measures under the human-centered dimension.

- **Performance and robustness to hallucination (P + RH)**: when considering both performance and robustness to hallucination, LoRA ranked first followed by QLoRA, in-context learning, and finally zero-shot learning. This observation suggests that zero-shot learning's decent ranking in robustness to hallucination cannot offset its low performance. Additionally, the higher ranking on QLoRA compared to in-context learning suggests that model customization contributed more to the models.
- **Performance, computational efficiency, and robustness to hallucination (P + CE + RH)**: when considering all three measures together, QLoRA received the highest overall ranking followed by a tie between zero-shot learning and LoRA and lastly in-context learning. This observation suggests that, if users are looking for an all-around adaptation strategy for LLM when considering performance, computational efficiency, and robustness to hallucination, QLoRA should be their top choice. It also implies that in-context learning struggles to achieve an optimal state, which explains the attention to prompt engineering among both the academics and practitioners.

Table 4. Adaptation Strategy Ranking by Machine-centered Evaluation Measure Combinations

Measure combination	Zero-shot learning	In-context learning	LoRA	QLoRA
P + CE	3	2	2	**1**
P + RH	4	3	**1**	2
P + CE + RH	2	3	2	**1**

6 Discussion

6.1 Research Contributions and Practical Implications

This study makes several research contributions. First, to address the challenges associated with augmenting human intelligence with LLMs, we first introduce a framework that categorizes LLMs based on the dimensions of customization and context and showcase representative adaptation strategies for the two dimensions. Then we align LLM adaptations with the stages of adaptation (training, customization, and inference), focusing specifically on the various types and levels of human intelligence at each stage. The categorization framework allows users to gain a comprehensive understanding of the different types of LLM adaptations.

Second, we introduce an evaluation framework that outlines various dimensions, measures, and methodologies for assessing an LLM. Extended from the extant literature, which leans heavily toward machine-centered evaluations, we propose a human-centered dimension. The human-centered dimension contains human control (the effort required to align the generated content with human expected outcome), prompt complexity (the complexity of human intelligence that feeds into LLMs), ease of use (the technical knowledge required for human-LLM interactions), and catastrophic forgetting resistance (LLMs' capability to retain skills beyond the specific task they are fine-tuned for). To the best of our knowledge, our study marks the first to propose human-centered evaluation measures for LLM adaptation. The machine-centered dimension encompasses measures such as performance (alignment between generated content and specific downstream tasks), computational efficiency (resources required for LLMs to enhance human intelligence), and robustness to hallucination (alignment between generated content and human intelligence). By providing a more comprehensive and organized structure beyond simply listing evaluation measures, we not only aids users in selecting among different LLM types according to their performance measures but also establish a theoretical foundation and guides future research in the field.

Third, to offer specific guidance on selecting adaptation strategies, we chose representative LLMs from the two pivotal model-development stages (model customization and inference) as case studies and assessed them using our evaluation framework. Specifically, we chose full fine-tuning and PEFT (e.g., LoRA and QLoRA) from the model-customization stage and zero-shot learning and in-context learning from the inferencing stage. We significantly expand the literature by employing text classification as the target task and selecting two classification problems to assess the different LLM types with respect to the proposed evaluation framework. We empirically tested the models using real-world datasets. Based on our key

observations from the results, we assigned rankings to the various selected LLM types according to our evaluation framework. The framework and rankings can serve as general guidelines for making an informed decision in selecting LLMs to enhance human intelligence. Furthermore, we found that the widely adopted zero-shot learning strategy often proves insufficient for knowledge-intensive tasks. Beyond the results from our use cases that support our proposed frameworks, we also made several observations that can serve as guidelines for future human–LLM interactions:

- PEFT outperform in-context learning strategies in text-classification tasks, though QLoRA can achieve comparable performance with significantly higher computational efficiency.
- RAG can greatly enhance models' robustness to hallucination, especially for full-sized models, compared to their quantized counterparts.
- Selecting demonstrations (e.g., examples in few-shot learning) constitutes an art more than a science. Selecting semantically similar examples cannot guarantee an enhancement in performance but will prolong the inference time.
- PEFT strategies, including both LoRA and QLoRA, improve model alignment with human control and, thus, reduce the need for post-processing to achieve the desired outcome. However, they are more sensitive to knowledge requirements compared to zero-shot learning and in-context learning.
- Model size does matter. Although we cannot guarantee that larger models will yield superior results compared to smaller ones, we suggest that users need to carefully select models based on their downstream tasks. In addition to carefully evaluating which foundational models are best suited for specific downstream tasks, users should consider training separate adapters for each task using various PEFT strategies and swapping them as needed. Another approach involves using quantized models to reduce the computational resource requirements while maintaining performance.

The proposed frameworks can significantly influence human–LLM interactions in several ways, such as:

- Both the categorization and evaluation frameworks can guide developers in creating more robust and versatile LLMs, improving their capacity to understand and respond to a wide range of human inputs. The frameworks enable users to continuously refine and adapt LLMs, leading to better alignment with user needs, increased reliability, and enhanced performance.
- The categorization framework for LLM adaptation provides a structured approach for selecting and implementing LLM adaptation strategies. In this way, it helps to ensure that the chosen methods align with one's specific goals and constraints.
- The evaluation framework helps to balance performance, computational efficiency, and robustness to hallucination. As such, it provides valuable guidance for selecting appropriate strategies to adapt LLMs for specific scenarios.

By integrating these elements, the proposed adaptation and evaluation frameworks can foster more effective, efficient, and trustworthy human–AI interactions.

6.2 Limitations and Future Research Issues

This study has several limitations. Firstly, selecting additional models from different families can help improve the generalizability of the findings. Secondly, a more quantifiable measure for human control, such as edit distance, can improve its reliability. Third, both researchers and practitioners can benefit from a fact-based validation of the robustness to hallucination measure.

Additionally, we employed text classification as the use case. Future research can validate our proposed framework across various use cases (e.g., text summarization) and customization techniques, and then extensively evaluate the models against adversarial conditions to better understand and mitigate hallucinations. For instance, we selected LoRA and QLoRA as representative PEFT techniques. As we discuss Section 3, researchers could explore additional PEFT techniques using our proposed evaluation framework, such as prefix tuning, prompt tuning, and decomposed LoRA. Additionally, we limited the quantization level to 4-bit; however, researchers could investigate heavier quantized techniques (including 1-bit quantization (Ma et al., 2024a)) so that users can fine-tune bigger models on consumer-level hardware.

In the proposed evaluation framework, particularly the human-centered dimension, we focused on those measures that pertain specifically to LLMs, such as ease of use and prompt complexity. Researchers should

consider other evaluation measures that play an important role in GenAI as well, such as scalability (how well an approach scales with model size and task complexity), resistance to adversarial attacks, transferability (the extent to which fine-tuning on one task can improve performance on related tasks), interpretability (the ease with which human users can understand and reason about a model's decisions), and fairness (the extent to which LLMs make unbiased predictions across different groups or individuals). For instance, transparency represents an important principle for AI (Zhou et al., 2021). It would be challenging to encourage users to adopt a model if it remained a “black box”. In addition to the traditional eXplainable AI (XAI) approaches and tools (e.g., LIME, SHAP (Salih et al., 2024)), another perspective would involve exposing LLMs’ reasoning process to human users. Strategies such as CoTs can illustrate how LLMs reach conclusions on specific problems in a human-in-the-loop manner. This knowledge would allow users to engage with the reasoning process, which would enhance their understanding and enable them to intervene if the reasoning goes wrong. This knowledge also highlights the when-aspect of intelligence augmentation (Zhou et al., 2023).

Autonomy constitutes another important aspect related to transparency (Andreoni et al., 2024) and refers to the degree to which models can act in a self-contained manner in the reasoning process. Take tool use (Zhuang et al., 2023) as an application example. The code snippets that LLMs generate can create exceptions that range from halt execution to critical damage to the related systems. It might be helpful to grant autonomy to the LLMs in a sandbox environment where the prompts embed error messages from the exceptions in an iterative and incremental manner so that the generated contents cannot leave the sandbox until they pass the pre-defined tests by human users.

To address the insufficiency of labeled data for text classification, researchers have employed LLMs to augment data by either annotating unlabeled data or generating synthetic data. For instance, one study used close-sourced LLMs (GPT-3.5-Turbo) in zero-shot and few-shot settings to generate synthetic labeled data to train downstream classification models (e.g., BERT/RoBERTa). Another example would involve using LLMs for labeling or co-annotation. Alternatively, one can use LLMs to generate label definitions that one then inserts into the subsequent prompts to assist classifiers. Furthermore, one can also use them to provide detailed, multi-step reasoning for previous/exemplar decisions. However, data-contamination concerns, poor understanding of low-resource cultures and languages, and human control over generations represent the main challenges when labeling and co-annotating training data for text classification (Li et al., 2023). These interesting issues require future research.

Future research could also explore using external tools to facilitate LLMs interacting with external entities, such as programming environments or database management systems, to perform specific downstream tasks (Zhuang et al., 2023). External tools could prove particularly useful for complex, knowledge-intensive problems that require rich, multi-modal knowledge to address. During the reasoning process, LLMs could temporarily pause to generate and execute computer code (e.g., SQL queries or Python snippets) and then integrate the results into the prompts before resuming the main reasoning process.

Acknowledgments

We thank the Editor-in-Chief, Dr. Fiona Nah, for her invaluable guidance and feedback. We are also grateful to Kanlun Wang for sharing the HumanMOD dataset and Adam LeBrocq for his thorough and careful edits.

References

Abdin, M., Jacobs, S. A., Awan, A. A., Aneja, J., Awadallah, A., Awadalla, H., Bach, N., Bahree, A., Bakhtiari, A., Bao, J., Behl, H., Benhaim, A., Bilenko, M., Bjorck, J., Bubeck, S., Cai, Q., Cai, M., Mendes, C. C. T., Chen, W., Chaudhary, V., Chen, D., Chen, D., Chen, Y.-C., Chen, Y.-L., Chopra, P., Dai, X., Del Giorno, A., de Rosa, G/. Dixon, M., Eldan, R., Fragoso, V., Iter, D., Gao, M., Gao, M., Gao, J., Garg, A., Goswami, A., Gunasekar, S., Haider, E., Hao, J., Hewett, R. J. Huynh, J., Javaheripi, M., Jin, X., Kauffmann, P., Karampatziakis, N., Kim, D., Khademi, M., Kurilenko, L., Lee, J. R., Lee, Y. T., Li, Y., Li, Y., Liang, C., Liden, L., Liu, C., Liu, M., Liu, W., Lin, E., Lin, Z., Luo, C., Madan, P., Mazzola, M., Mitra, A., Modi, H., Nguyen, A., Norick, B., Patra, B., Perez-Becker, D., Portet, T., Pryzant, R., Qin, H., Radmilac, M., Rosset, C., Roy, S., Ruwase, O., Saarikivi, O., Saied, A., Salim, A., Santacroce, M., Shah, S., Shang, N., Sharma, H., Shukla, S., Song, X., Tanaka, M., Tupini, A., Wang, X., Wang, L., Wang, C., Wang, Y., Ward, R., Wang, G., Witte, P., Wu, H., Wyatt, M., Xiao, B., Xu, C., Xu, J., Xu, W., Yadav, S., Yang, F., Yang, J., Yang, Z., Yang, Y., Yu, D., Yuan, L., Zhang, C., Zhang, C., Zhang, J., Zhang, L. L., Zhang, Y., Zhang, Y., Zhang, Y., Zhou, X. (2024). Phi-3 technical report: A highly capable language model locally on your phone. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2404.14219

Ahmed, T., & Devanbu, P. (2023). Few-shot training LLMs for project-specific code-summarization. In *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering.*

Andreoni, M., Lunardi, W. T., Lawton, G., & Thakkar, S. (2024). Enhancing autonomous system security and resilience with generative AI: A comprehensive survey. IEEE Access, *12*, 109470-109493.

Bhattacharyya, R., Wulfe, B., Phillips, D. J., Kuefler, A., Morton, J., Senanayake, R., & Kochenderfer, M. J. (2023). Modeling human driving behavior through generative adversarial imitation learning. *IEEE Transactions on Intelligent Transportation Systems*, *24*(3), 2874-2887.

Bilenko, M. (2024). Introducing Phi-3: Redefining what's possible with SLMs. *Microsoft Azure Blog*. Retrieved from https://azure.microsoft.com/en-us/blog/introducing-phi-3-redefining-whats-possible-with-slms/

Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., Ye, W., Zhang, Y., Chang, Y., Yu, P. S., Yang, Q., & Xie, X. (2024). A Survey on Evaluation of Large Language Models. *ACM Transactions on Intelligent Systems and Technology*, *15*(3), 1-45.

Chiang, C.-W., Lu, Z., Li, Z., & Yin, M. (2024). Enhancing AI-assisted group decision making through LLM-powered devil's advocate. In *Proceedings of the 29th International Conference on Intelligent User Interfaces.*

Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., & Gonzalez, J. E. (2023). *Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality*. Large Model Systems Organization. Retrieved from https://lmsys.org/blog/2023-03-30-vicuna/

Chui, M., Hazan, E., Roberts, R., Singla, A., & Smaje, K. (2023). *The economic potential of generative AI.* McKinsey & Company. Retrieved from http://dln.jaipuria.ac.in:8080/jspui/bitstream/123456789/14313/1/The-economic-potential-of-generative-ai-the-next-productivity-frontier.pdf

Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023). QLoRA: Efficient finetuning of quantized LLMs. *Advances in Neural Information Processing Systems*, *36*, 10088-10115.

Ding, N., Qin, Y., Yang, G., Wei, F., Yang, Z., Su, Y., Hu, S., Chen, Y., Chan, C.-M., Chen, W., Yi, J., Zhao, W., Wang, X., Liu, Z., Zheng, H.-T., Chen, J., Liu, Y., Tang, J., Li, J., & Sun, M. (2023). Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3), 220-235.

Epstein, Z., Hertzmann, A., & the Investigators of Human Creativity. (2023). Art and the science of generative AI. *Science*, *380*(6650), 1110-1111.

Falotico, R., & Quatto, P. (2015). Fleiss' kappa statistic without paradoxes. *Quality & Quantity*, *49*(2), 463-470.

Feng, K. J. K., Liao, Q. V., Xiao, Z., Vaughan, J. W., Zhang, A. X., & McDonald, D. W. (2024). Canvil: designerly adaptation for LLM-powered user experiences. arXiv. Retrieved from https://doi.org/10.48550/arXiv.2401.09051

Feuerriegel, S., Hartmann, J., Janiesch, C., & Zschech, P. (2024). Generative AI. *Business & Information Systems Engineering*, *66*(1), 111-126.

Hanchen, D. (2024). *Unslothai/unsloth*. Retrieved from https://github.com/unslothai/unsloth

Hutson, J., & Ratican, J. (2023). Leveraging generative agents: Autonomous AI with simulated personas for interactive simulacra and collaborative research. *Journal of Innovation and Technology*, *2023*(15).

Ibrahim, L., Huang, S., Ahmad, L., & Anderljung, M. (2024). Beyond static AI evaluations: Advancing human interaction evaluations for LLM harms and risks. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2405.10632

Karabacak, M., & Margetis, K. (2023). Embracing large language models for medical applications: Opportunities and challenges. *Cureus*, *15*(5).

Karanikolas, N., Manga, E., Samaridi, N., Tousidou, E., & Vassilakopoulos, M. (2023). Large language models versus natural language understanding and generation. In *Proceedings of the 27th Pan-Hellenic Conference on Progress in Computing and Informatics.*

Kocoń, J., Cichecki, I., Kaszyca, O., Kochanek, M., Szydło, D., Baran, J., Bielaniewicz, J., Gruza, M., Janz, A., Kanclerz, K., Kocoń, A., Koptyra, B., Mieleszczenko-Kowszewicz, W., Miłkowski, P., Oleksy, M., Piasecki, M., Radliński, Ł., Wojtasik, K., Woźniak, S., & Kazienko, P. (2023). ChatGPT: Jack of all trades, master of none. *Information Fusion*, *99*, 1-37.

Kojima, T., Gu, S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems*, *35*, 22199-22213.

Krugmann, J. O., & Hartmann, J. (2024). Sentiment analysis in the age of generative AI. *Customer Needs and Solutions*, *11*, 1-19.

Laban, P., Kryscinski, W., Agarwal, D., Fabbri, A., Xiong, C., Joty, S., & Wu, C.-S. (2023). SummEdits: measuring LLM ability at factual reasoning through the lens of summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.*

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *arXiv.* Retrieved from https://doi.org/10.48550/arXiv.2005.11401

Li, J., Li, J., & Su, Y. (2024). A map of exploring human interaction patterns with LLM: Insights into collaboration and creativity. In H. Degen & S. Ntoa (Eds.), *Artificial intelligence in HCI* (pp. 60–85). Springer.

Li, Z., Zhu, H., Lu, Z., & Yin, M. (2023). Synthetic data generation with large language models for text classification: Potential and limitations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.*

Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M., & Raffel, C. (2022a). Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35, 1950-1965.

Liu, J., Shen, D., Zhang, Y., Dolan, B., Carin, L., & Chen, W. (2022b). What makes good in-context examples for GPT-3? In *Proceedings of the 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures.*

Ma, S., Wang, H., Ma, L., Wang, L., Wang, W., Huang, S., Dong, L., Wang, R., Xue, J., & Wei, F. (2024a). The era of 1-bit LLMs: All large language models are in 1.58 Bits. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2402.17764

Ma, X., Wang, L., Yang, N., Wei, F., & Lin, J. (2024b). Fine-tuning LLaMA for multi-stage text retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval.*

Malo, P., Sinha, A., Korhonen, P., Wallenius, J., & Takala, P. (2014). Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, *65*(4), 782-796.

Hugging face. (2024). *Models*. Retrieved from https://huggingface.co/models

Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., & Gao, J. (2024). Large language models: A survey. *arXiv*. Retrieved from http://arxiv.org/abs/2402.06196

Nah, F., Cai, J., Zheng, R., & Pang, N. (2023a). An activity system-based perspective of generative AI: Challenges and research directions. *AIS Transactions on Human-Computer Interaction*, *15*(3), 247-267.

Nah, F., Zheng, R., Cai, J., Siau, K., & Chen, L. (2023b). Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. *Journal of Information Technology Case and Application Research*, *25*(3), 277-304.

Nashid, N., Sintaha, M., & Mesbah, A. (2023). Retrieval-based prompt selection for code-related few-shot learning. In *Proceedings of the 45th International Conference on Software Engineering.*

Navigli, R., Conia, S., & Ross, B. (2023). Biases in large language models: Origins, inventory, and discussion. *ACM Journal of Data and Information Quality*, *15*(2), 1-21.

Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*.

Patwa, P., Filice, S., Chen, Z., Castellucci, G., Rokhlenko, O., & Malmasi, S. (2024). Enhancing low-resource LLMs classification with PEFT and synthetic data. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2404.02422

Pourpanah, F., Abdar, M., Luo, Y., Zhou, X., Wang, R., Lim, C. P., Wang, X.-Z., & Wu, Q. M. J. (2023). A review of generalized zero-shot learning methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(4), 4051-4070.

Preiksaitis, C., & Rose, C. (2023). Opportunities, challenges, and future directions of generative artificial intelligence in medical education: Scoping review. *JMIR Medical Education*, *9*(1), 1-13.

Rahman, M. M., & Watanobe, Y. (2023). ChatGPT for education and research: Opportunities, threats, and strategies. *Applied Sciences*, *13*(9), 1-21.

Raiaan, M. A. K., Mukta, Md. S. H., Fatema, K., Fahad, N. M., Sakib, S., Mim, M. M. J., Ahmad, J., Ali, M. E., & Azam, S. (2024). A review on large language models: Architectures, applications, taxonomies, open issues and challenges. *IEEE Access*, *12*, 26839-26874.

Salih, A. M., Raisi-Estabragh, Z., Galazzo, I. B., Radeva, P., Petersen, S. E., Lekadir, K., & Menegaz, G. (2024). A perspective on explainable artificial intelligence methods: SHAP and LIME. *Advanced Intelligent Systems*, 1-8.

Sayin, B., Minervini, P., Staiano, J., & Passerini, A. (2024). Can LLMs correct physicians, yet? Investigating effective interaction methods in the medical domain. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2403.20288

Sheng, Y., Cao, S., Li, D., Hooper, C., Lee, N., Yang, S., Chou, C., Zhu, B., Zheng, L., Keutzer, K., Gonzalez, J., & Stoica, I. (2024). SLoRA: Scalable serving of thousands of LoRA adapters. *arXiv.* Retrieved from https://arxiv.org/abs/2311.03285v3

Simon, H. A. (1976). *Administrative behavior: A study of decision-making processes in administrative organization* (3rd ed.). Free Press.

Song, Y., Wang, T., Cai, P., Mondal, S. K., & Sahoo, J. P. (2023). A comprehensive survey of few-shot Learning: evolution, applications, challenges, and opportunities. *ACM Computing Surveys*, *55*(13s), 1-40.

Sun, X., Li, X., Li, J., Wu, F., Guo, S., Zhang, T., & Wang, G. (2023). Text classification via large language models. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2305.08377

Wang, K., Fu, Z., Zhou, L., & Zhang, D. (2023). How does user engagement support content moderation? A deep learning-based comparative study. In *Proceedings of the Americas Conference on Information Systems.*

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, *35*, 24824-24837.

Weyssow, M., Zhou, X., Kim, K., Lo, D., & Sahraoui, H. (2024). Exploring parameter-efficient fine-tuning techniques for code generation with large language models. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2308.10462

Yahoo! Finance. (2024). *Large language model (LLM) market research 2024-2034: Continued push towards larger and more capable models, increasing integration into business applications.* Retrieved from https://finance.yahoo.com/news/large-language-model-llm-market-094600872.html

Yeom, J., Lee, H., Byun, H., Kim, Y., Byun, J., Choi, Y., Kim, S., & Song, K. (2024). Tc-llama 2: Fine-tuning LLM for technology and commercialization applications. *Journal of Big Data*, *11*, 1-31.

Zhang, P., Zeng, G., Wang, T., & Lu, W. (2024). TinyLlama: An open-source small language model. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2401.02385

Zhao, H., Andriushchenko, M., Croce, F., & Flammarion, N. (2024). Is in-context learning sufficient for instruction following in LLMs? *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2405.19874

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.-Y., & Wen, J.-R. (2023). A survey of large language models. *arXiv*. Retrieved from https://doi.org/10.48550/arXiv.2303.18223

Zhou, L., Paul, S., Demirkan, H., Yuan, L., Spohrer, J., Zhou, M., & Basu, J. (2021). Intelligence augmentation: Towards building human-machine symbiotic relationship. *AIS Transactions on Human-Computer Interaction*, 13(2), 243-264.

Zhou, L., Rudin, C., Gombolay, M., Spohrer, J., Zhou, M., & Paul, S. (2023). From artificial intelligence (AI) to intelligence augmentation (IA): Design principles, potential risks, and emerging issues. *AIS Transactions on Human-Computer Interaction*, *15*(1), 111-135.

Zhuang, Y., Yu, Y., Wang, K., Sun, H., & Zhang, C. (2023). ToolQA: A dataset for LLM question answering with external tools. *Advances in Neural Information Processing Systems*, *36*, 50117-50143.

About the Authors

Jie Tao is an Associate Professor of Analytics in the Charles F. Dolan School of Business at Fairfield University. Dr. Tao received a Doctoral degree in Information Systems from Dakota State University. His recent research interests majorly include *deep learning and natural language processing,* as well as *eXplainable AI*. He published several papers in prestigious Information Systems journals, such ACM Transactions on Management Information Systems, IEEE Transactions on Engineering Management, Information Systems Frontiers, and Information Systems Research. He also received the best paper award at Hawaii International Conference on System Sciences (HICSS-52' 2020). Dr. Tao also received the Association for Information Systems (AIS) ATLAS award for his services to AIS in 2013. He served as the faculty advisor for one of the top 10 student teams in IBM WAGC 2016. Dr. Tao is currently a certified instructor for the NVidia Deep Learning Institute.

Lina Zhou is a Professor of Management Information Systems at the University of North Carolina at Charlotte. Her research focuses on improving human decision-making and knowledge management through both the design and development of intelligent systems and the understanding of human behavior. She has published in journals, including MIS Quarterly, Information Systems Research, Journal of Management Information Systems, ACM and IEEE Transactions, Information & Management, Decision Support Systems, and AIS Transactions on Human-Computer Interaction. Her research has received funding from different federal and state agencies.

Xing Fang is an Associate Professor in Computer Science at the School of Information Technology in Illinois State University. Dr. Fang received a Ph.D. in Computer Science from North Carolina A&T State University. He also holds a Master's degree in Computer Science and a Master's degree in Information Assurance from North Carolina A&T State University and Dakota State University, respectively. Dr. Fang published in several journals, such Journal of Big Data, Engineering Applications of AI, and IEEE Access. His recent research interests majorly include *deep learning and natural language processing*.

Transactions on Human - Computer Interaction

https://aisel.aisnet.org/thci/