When One LLM Drools, Multi-LLM Collaboration Rules

Shangbin Feng[1], Wenxuan Ding[2], Alisa Liu[1], Zifeng Wang[3], Weijia Shi[1], Yike Wang[1], Shannon Zejiang Shen[4], Xiaochuang Han [1], Hunter Lang[4], Chen-Yu Lee[3], Tomas Pfister[3], Yejin Choi[5], Yulia Tsvetkov[1]

[1]University of Washington, [2]New York University, [3]Google, [4]Massachusetts Institute of Technology, [5]Stanford University

shangbin@cs.washington.edu

Abstract

This position paper argues that in many realistic (i.e., complex, contextualized, subjective) scenarios, one LLM is not enough to produce a reliable output. We challenge the status quo of relying solely on a single general-purpose LLM and argue for *multi-LLM collaboration* to better represent the extensive diversity of data, skills, and people. We first posit that a single LLM underrepresents real-world data distributions, heterogeneous skills, and pluralistic populations, and that such representation gaps cannot be trivially patched by further training a single LLM. We then organize existing multi-LLM collaboration methods into a hierarchy, based on the level of access and information exchange, ranging from API-level, text-level, logit-level, to weight-level collaboration. Based on these methods, we highlight how multi-LLM collaboration addresses challenges that a single LLM struggles with, such as reliability, democratization, and pluralism. Finally, we identify the limitations of existing multi-LLM methods and motivate future work. We envision multi-LLM collaboration as an essential path toward compositional intelligence and collaborative AI development.

1 Introduction

The successes of scaling models (Kaplan et al., 2020) and data (Hoffmann et al., 2022) have fueled the overly optimistic hope that simply building an ever-larger language model is a path to achieving human-like intelligent AI models. From research artifacts to user-facing products, the commercialization of LLM and AI technologies further reinforces this status quo: major players train a single general-purpose in-house LLM and compete by attempting to outrank other models (Henshall, 2024). This quest for the "best" single LLM—measured by leaderboard scores, user adoption, and profitability—has brought about the bloom of LLM research and development where new models emerge daily and the state-of-the-art is constantly reshaped (Liang et al., 2023a; Chiang et al., 2024b; Guo et al., 2025).

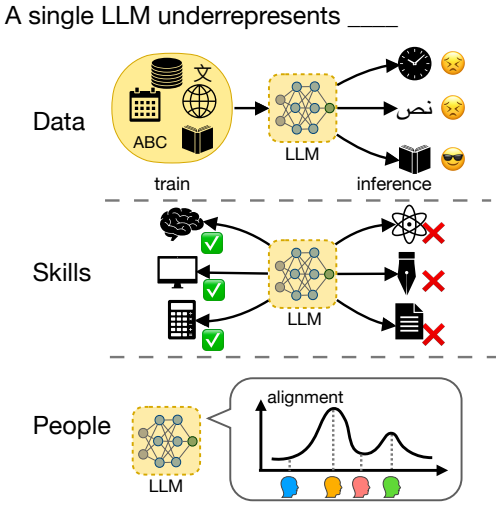

Figure 1: Despite the quest for general-purpose models, a single LLM suffers from underrepresentation of data (language varieties, domains, styles), skills (reasoning abilities, linguistic and communication skills, creative capacities, and technical competencies), and people (opinions, values, cultural norms).

We challenge the status quo by arguing that **one LLM is not enough** and advocate for **multi-LLM collaboration**, where multiple language models collaborate for compositional generative modeling. We first argue *why* one LLM is not enough (§2): despite being general-purpose, a single monolithic model struggles to reflect the intricate diversity of the real world and *underrepresents* the long tail of data types, model skills, and people. (Figure 1) We then propose a taxonomy of multi-LLM collaboration protocols (§3) in which LLMs collaborate, interact, and exchange information at the API-level, text-level, logit-level, and weight-level, offering diverse modes of collaboration compatible with all stages of the LLM lifecycle and usage types. We then argue that multi-LLM systems empowered by these protocols bring out benefits that a single LLM

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 17048–17063

struggles to reflect (§4): pluralism, democratization, efficiency, adaptability, and more. Together, these arguments demonstrate that multi-LLM collaboration is an important yet overlooked family of methods, and a promising approach to advance language technologies.

We identify some limitations of existing multi-LLM collaboration protocols and applications (§5), which motivate us to lay out an actionable roadmap for future work beyond monolithic models and towards advancing modular multi-LLM systems. We hope that this position will be a call-to-action for the research community to propose, evaluate, and promote collaboration strategies and communication protocols for multi-LLM collaboration.

2 One LLM Is Not Enough

From the early successes of scaling up model size (Kaplan et al., 2020) and training data (Hoffmann et al., 2022), language technologies have transitioned from task-specific systems to "general-purpose" language models (Brown et al., 2020): by pretraining on gigantic corpora and aligning with extensive instruction tuning and preference optimization, one LLM can be prompted to solve a diverse range of tasks and problems, leading some to believe that the future of language technologies is in figuring out the recipe for scaling and developing a single omnipotent LLM. Despite its promise, we argue that a single LLM, as designed today, is not enough to achieve a truly reliable system: even with the best effort to curate data, design architectures, and improve model inference, a single LLM suffers from *underrepresentation* on three fronts: data, skills, and people.

Underrepresentation of data. Despite extensive data curation efforts, a single LLM is ultimately trained on a static snapshot of what is available, and there are always elements in the real-world language distributions that are missing or down-weighted in this static snapshot (Lazaridou et al., 2021). For example, constant changes in the state of the world quickly make the parametric information of LLMs outdated (Dhingra et al., 2022; Kasai et al., 2024). Private and copyrighted texts would require careful consideration in LLM training, but are otherwise essential for personalization and context (Wei et al., 2024; Chen et al., 2024). Languages, dialects, and language varieties in the long tail of data distributions are easily outnumbered and overshadowed by the majority (Song et al., 2023; Faisal et al., 2024). Evolving trends, unspoken cultural and social norms essential for socially-aware LLM applications, commonsense and implicit knowledge are hard to pin down with static data snapshots (Rao et al., 2024; Shi et al., 2024c). The list goes on, and much of the real-world variation expressed through language will inevitably be lost when we solely rely on a single LLM with a static hodgepodge of training corpora.

Underrepresentation of skills. Earlier language technologies were defined by task-specific progress with specialized methods, models, and subcommunities of experts for tasks like machine translation, summarization, question answering, and natural language inference (Sun et al., 2022). LLMs broke from this trend by being seemingly "general-purpose" and it appears plausible that all we will need in the near future is a single omnipotent LLM that works best in any task and context.

However, no single LLM is Pareto-optimal *empirically* and it is prohibitively expensive (if not impossible) to optimize for a single model that outperforms all other models on *all* skills. For example, Gemini (Team et al., 2023) currently ranks best on Chatbot Arena (Chiang et al., 2024b) focusing on instruction following, GPT-4o (Achiam et al., 2023) is best on the HELM leaderboard (Liang et al., 2023a) with an emphasis on QA and math reasoning, while a fine-tuned version of InternLM (Team, 2023) is best on textual and algorithmic tasks in Big-Bench Hard (Suzgun et al., 2023) on Open LLM Leaderboard (Fourrier et al., 2024).[1] Compared to multilingual LLMs, these models would perform poorly on GlobalBench (Song et al., 2023) and DialectBench (Faisal et al., 2024) compared to multilingual LLMs, where tasks include languages and language varieties not captured in the most popular leaderboards. This demonstrates that even the most advanced LLMs have major limitations in skills and task coverage, and that additional specialization of models is critical.

Underrepresentation of people. All LLMs are ultimately used by people with diverse needs, pluralistic values, and varying socio-cultural backgrounds. Despite the ever-increasing model size and benchmark scores, we witness a constant lack of representation of actual LLM users.

On one hand, a single LLM struggles to reflect pluralistic human values, cultures, and social contexts (Sorensen et al., 2024b; Feng et al., 2024d;

[1] Leaderboards accessed on Nov 24, 2024.

Leibo et al., 2024), in any language. LLM users are not homogeneous, bringing a wealth of perspectives and diversity that shapes our world: despite the potential diversity in data sources, even state-of-the-art LLMs cannot equitably serve the entire spectrum of users by reflecting such heterogeneity. For example, LLMs often feature a West-centric cultural persona (Naous et al., 2023) and struggle to adapt to cultural variation (Rao et al., 2024); a single LLM would most likely reinforce the majority class in training data and exhibit biases in opinions and perspectives (Santurkar et al., 2023; Feng et al., 2023a); user agency often remains overlooked since monolithic LLMs lack steerability and controllability in value-laden instructions and contexts (Sorensen et al., 2024a). Since LLMs are already trained on diverse texts from the web, representing populations would require solutions beyond scaling data for a general-purpose LLM.

Moreover, by solely relying on one single model we are also solely relying on only one team of model developers. With the increasing cost and opaqueness of developing an LLM, these teams are becoming highly homogeneous: big tech companies, researchers with advanced degrees, overrepresentation of certain demographic groups are common sketches of the teams behind state-of-the-art LLMs (EEOC, 2024). However, this leaves the vast majority of actual and underprivileged LLM users without a say in the decision making of model training and development, while they can only access these LLMs which might not have been developed with their needs and priorities in mind. An open and collaborative development approach that is the cornerstone of open-source communities (Johnson, 2006) is thus neglected in the over-focus on chasing the best single model, underrepresenting the voices and needs of actual LLM users that go beyond synthetic benchmark numbers.

Challenges to Improve One Model's Coverage A tempting solution to these problems is to further train the current best LLM to improve the representation of data, skills, and users. We argue that this band-aid approach is challenging at best:

When *data* is underrepresented, model developers can scrape from previously unseen domains for further tuning. However, it is costly to frequently re-train and update model versions with gigantic LLMs, while private and copyrighted data simply should not be included in training. Retrieval-augmented geneartion (Guu et al., 2020; Shi et al., 2023) could provide unseen data as context, but it is unclear whether LLMs would fully leverage the context (Shi et al., 2024b) and to what extent is this steerability reliable (Sprague et al., 2024).

When *skills* are underrepresented, model developers can derive targeted supervised fine-tuning data for continual learning (Zhang et al., 2023). However, tuning to patch a weakness in skills may lead to tradeoffs in other tasks and sometimes even catastrophic forgetting (Luo et al., 2023; Lin et al., 2024), as any specialization on the trained model might harm its general-purpose utility.

When *humans* are underrepresented, model developers can survey the needs of diverse populations and communities for LLMs and invite collaborative contributions (Feng et al., 2024a). However, there is little to no incentive for teams behind commercial state-of-the-art LLMs to take great strides towards equitable language technologies without obvious profitable gains.

It is important to note that these underrepresentation issues of a single LLM, especially with respect to data and skills, are grounded in *empirical evidence*, i.e., current state-of-the-art LLMs are suffering from these challenges. There might emerge future "perfect" algorithms/architectures/etc. that fully address these issues, but given that multi-LLM collaboration research is *already demonstrating empirical benefits* in addressing these issues, we advocate for multi-LLM collaboration as a promising and effective research avenue.

3 Types of Multi-LLM Collaboration

We categorize existing (often unrelated) methods into a conceptual family of collaboration strategies, organizing methods by (1) collaboration at different levels of access to an LLM, as illustrated in Figure 2, and (2) collaboration at different stages of an LLM's lifecycle.

3.1 Different levels of model access

API-Level As the name suggests, access to an LLM's API is sufficient to enable API-level collaboration between models. Such strategies focus on the dynamic selection of the most cost-efficient and high-performing model among a diverse pool of LLMs for different inputs. Intuitively, we should assign simpler requests to smaller (Tambon et al., 2024), more efficient LLMs for *reduced cost and latency*, and domain-specific requests to expert LLMs for *improved performance*. There are two mainstream lines of research on API-level LLM

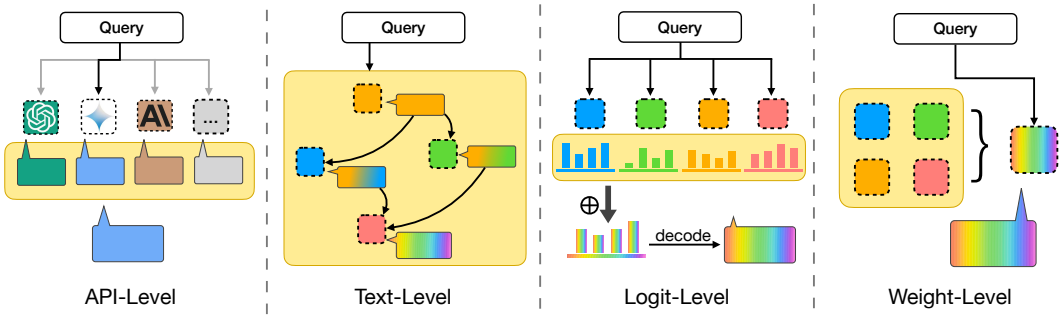

Figure 2: We propose a typology of multi-LLM collaboration approaches, focusing on different levels of access to LLMs, and survey existing methods that fall into each type.

collaboration: *Routing* (Hu et al., 2024) and *Cascading* (Chen et al., 2023).

Routing selects the most suitable model only based on the input, without performing inference on any LLM. A typical routing paradigm involves designing a separate router model that learns from human preferences. As a key step, preference labels (Ong et al., 2024) that represent the relative response quality of different LLMs are collected for each input. Prior work developed various router models to learn from input–preference pairs, including non-parametric routers like KNN-based router (Shnitzer et al., 2023), and parametric routers like MLP- (Hu et al., 2024), encoder- (Ding et al., 2024), and decoder-based (Ong et al., 2024) routers. Beyond preference data, additional information can be leveraged to assist in routing decision-making. To select the most suitable expert LLM, domain-specific routing strategies Lu et al. (2023); Stripelis et al. (2024) extract key information about the task and domain directly from the input. Feng et al. (2024f) further introduce a heterogeneous graph framework to leverage contextual interactions among tasks, queries, and LLMs.

Cascading defers the input to larger/more capable LMs when the response from the smaller LM is not satisfactory enough. The crux of cascading is the *deferral rule* to determine whether to return the prediction or to invoke the next LLM. The pioneering work Frugal GPT (Chen et al., 2023) trains a regression model that predicts a response quality score and establishes the deferral rule by thresholding the predicted score. Yue et al. (2023) presents a consistency-based approach that estimates the response confidence score, such that inputs with low response confidence are deferred to the next LLM. Gupta et al. (2024) further incorporate token-level uncertainty into deferral rules. Cascading strategies, while potentially improving overall quality by leveraging additional signal from smaller LLMs, often come with increased cost and latency due to the overhead of decoding intermediate responses.

Text-Level Text-level approaches enable multi-LLM collaboration through exchanges of generated texts, where one LLM's output becomes another LLM's input. They usually follow a conversational setting where LLMs can "cooperate" or "compete" with each other.

For cooperation, models can *divide and conquer* complex problems through multi-agent systems where each agent is seeded by different models/prompts (Wu et al., 2024a; Guo et al., 2024; Wu et al., 2025); specialized models can *augment* general-purpose LLMs to patch their gaps (Feng et al., 2024a; Shen et al., 2024); one LLM can generate *feedback* or perform verification for another LLM's outputs and consequently refine the generation (Burns et al., 2024; Feng et al., 2024b).

For *competition*, multiple LLMs can "debate" with each other to advance factuality and reasoning (Liang et al., 2023b; Du et al., 2024b). Recent research also explored employing a pool of diverse specialized LLMs to model social (Zhao et al., 2024a) and economic (Zhao et al., 2024b) behavior to simulate the real-world environment.

Much effort of multi-LLM collaboration research currently operates at the text-level, probably because such interaction allows for the use of APIs in closed models, the ease of engineering to redirect model outputs, and transparency through intermediate model outputs. However, text-level multi-LLM collaboration also faces challenges such as error propagation from outputs of individual models, the lack of generalization across tasks, and the costs of model inference for multiple LLMs.

Logit-Level LLMs may also collaborate by jointly contributing to each next-token prediction. In this case, the logit-level predictions of multiple LLMs are combined via arithmetic to create a single next-token logit distribution. This approach uses other LLMs as "experts" and/or "anti-experts", whose predictions are additively or negatively combined in the prediction (Liu et al., 2021).

Using an anti-expert achieves the effect of steering *away* from the preferences of that model, and is also known as *contrastive decoding* (Li et al., 2023). For instance, the anti-expert may be an LLM tuned explicitly to be toxic (Liu et al., 2021), to achieve safer generations, or a smaller LLM (Li et al., 2023), to avoid the pitfalls of weaker LMs for better open-ended generation. In fact, the anti-expert does not need to be a distinct LLM, and can instead be the result of ablating some part of the current LLM, e.g., by withholding necessary context (Pei et al., 2023; Sennrich et al., 2024; Leng et al., 2024; Shi et al., 2024b) or early-exiting from an earlier layer of the transformer model (Gera et al., 2023; Chuang et al., 2024).

On the other hand, using multiple expert LLMs combines their predictions in a product-of-experts fashion. Intuitively, this leads to next-token predictions that are high-probability under all LLMs. This has been used to achieve decoding-time adaptation of LLMs using small tuned experts with a large pretrained LLM (Liu et al., 2024; Mitchell et al., 2024), allowing for on-the-fly customization of the weights of multiple finetuning objectives (Shi et al., 2024a).

The weights assigned to experts and anti-experts may also be automatically determined at each step (Mavromatis et al., 2024; Fan et al., 2024; Du et al., 2024a). At the extreme, this takes the form of token-level routing (Shen et al., 2024).

Weight-Level The collaboration of multiple LLMs through parameter-level collaboration has been explored using paradigms such as mixtures of feed-forward layers (Sukhbaatar et al., 2024), adapters (Wang et al., 2022b; Pfeiffer et al., 2020), and low-rank adaptation (LoRA) experts (Wu et al., 2024b). In this paradigm, components like feed-forward layers and adapters are first trained independently on domain-specific or task-specific data to achieve specialization. Subsequently, in a combination stage, these independently trained modules are jointly optimized to collaborate effectively, creating a unified system that benefits from the specialized expertise of each component.

This framework supports collaboration across varying levels of input granularity the way experts are selected and aggregated. For example, some approaches dynamically select modules for individual *tokens* (Vaswani, 2017; Houlsby et al., 2019; Pfeiffer et al., 2020; Belofsky, 2023; Wu et al., 2024b; Sukhbaatar et al., 2024), enabling fine-grained expertise sharing. Others perform collaboration at the *sentence* level (Diao et al., 2023; Xu et al., 2023), where different input sentences activate different modules. At the *task* level, methods such as Chiang et al. (2024a) assign a single expert model to all examples from a particular task. Weight-level collaboration typically allows for deeper integration of experts by enabling routing decisions at each layer where modules are inserted, offering greater flexibility and adaptability to diverse tasks and data.

Another line of weight-level collaboration research is the merging/composition of model weights across multiple LLMs. These approaches mainly vary by *data dependency*, i.e., how much task-specific data is required to compose and adapt models. *Zero-shot* model composition approaches rely on heuristics about model weights (Yu et al., 2024; Yadav et al., 2024b) or task arithmetic (Ilharco et al., 2023) to produce composed models and advance generalization without access to task data. Given a small set of task data, *dynamic composition* approaches optimize the model composition based on performance and metrics on the task data (Huang et al., 2023) with perplexity heuristics (Mavromatis et al., 2024) and evolutionary algorithms (Feng et al., 2024e). If supervised data is abundant, *learn-to-fuse* approaches design trainable modules (Bansal et al., 2024), adapters (Wang et al., 2024), or even LLMs (Jiang et al., 2023a) to "glue" multiple LLMs together: the component LLMs are often kept frozen while the trainable parts go through supervised fine-tuning from scratch. Weight-level approaches offer a spectrum of solutions based on data availability, and the *many-to-one* setup offers reduced inference costs. However, weight-level approaches are less interpretable in how model expertise is composed and do not tap into the power of collaborative generation like text- or logit-level approaches.

3.2 Different stages of LLM development

We can also categorize multi-LLM collaboration approaches by the three stages of the LLM lifecycle: *(pre)training, post-training, and inference*. Pretraining-time approaches focus on partitioning LLM training data (Gururangan et al., 2023) and training multiple specialized LLMs separately (Li et al., 2022) or at the same time (Devvrit et al., 2024). Post-training approaches explore collaborative alignment with modular reward models (Jang et al., 2023), multi-LLM self-alignment (Feng et al.,

2024d), or constructing synthetic supervised fine-tuning data through multi-LLM debate (Subramaniam et al., 2024, 2025). The vast majority of multi-LLM collaboration approaches currently operate at inference time, offering diverse ways of reusing existing models spanning all four collaboration levels (Hu et al., 2024; Du et al., 2024b; Liu et al., 2024). In general, weight-level methods often require more (pre)training and post-training efforts, while API-level/logit-level collaborations focus more on inference-time solutions.

By conceptually structuring and organizing these (originally unrelated) methods into a family of approaches, we argue that multi-LLM collaboration research offers flexible methodologies for any level of model access across all stages in the LLM lifecycle, providing an alternative and promising school of thought to advance language technologies.

4 The Promise of Multi-LLM Collaboration

Multi-LLM collaboration systems offer unique advantages over single general-purpose models: we summarize the methodological and empirical benefits of existing multi-LLM proposals in this section.

Factuality and reliability Despite prior efforts (Shi et al., 2023; Press et al., 2022; Feng et al., 2023b) to expand the knowledge of LLMs, knowledge gaps—missing or outdated information in LLMs—may persist due to the evolving nature of knowledge, presenting challenges to reliability. Self-reflection (Wang et al., 2022a; Xu et al., 2024; Shinn et al., 2024; Madaan et al., 2024), where a single LLM critically evaluates its own generations, is used in decoding, confidence calibration, and inference to improve factual accuracy and mitigate hallucinations. However, this suffers from confirmation biases (Ji et al., 2023) and relies on the assumption that LLMs can generate novel reflections from their initial outputs (Liang et al., 2023b). Recent studies address these issues by promoting collaboration among multiple LLMs. With distinct knowledge coverage, LLMs evaluate and reflect on each other's outputs, collaboratively probing and identifying the knowledge gaps of each other. Specifically, Feng et al. (2024c) enable robust LLM abstention through multi-LLM collaboration to reflect on generated text in cooperative or competitive settings. Cohen et al. (2023) employs cross-examination to detect errors in LLM generations. Other studies (Liang et al., 2023b; Du et al., 2024b) suggest that multiple LLMs could propose and debate their individual responses and reasoning processes over multiple rounds to arrive at a common final answer, and LLMs with comparable abilities have been shown to demonstrate such collaborative spirit (Xiong et al., 2023). Given its superior performance in various settings, we believe that multi-LLM collaboration offers a promising way to further improve the factual validity of generated context and reduce fallacious answers and hallucinations that contemporary models are prone to.

Alignment and pluralism State-of-the-art LLMs are documented with all kinds of cultural (Naous et al., 2023), political (Santurkar et al., 2023), and broadly social biases (Kumar et al., 2023). This comes with the fact that these models have already seen "diverse" web data that should serve as a decentralized representation of real-world diversity. Much research attributes this to LLMs learning disproportionately from and hence reinforcing the majority in training data (Feng et al., 2023a; Gallegos et al., 2024), thus scaling data diversity used in training a single LLM is not an effective solution. We see an increasing line of work focused on *modular multi-LLM systems* to alleviate these biases, including modular plug-ins (Feng et al., 2024d), multi-LLM as a judge (Zhao et al., 2024a), and employing multiple and compositional reward models (Jang et al., 2023). Together with data-side modularity spanning diverse communities (Kumar et al., 2024; Kirk et al., 2024) we believe multi-LLM collaboration offers a modular and flexible solution to addressing the fairness and pluralism challenges of LLMs.

Efficiency The most capable LLMs at the moment often feature gargantuan sizes and prohibitively high inference costs. However, not all queries require such computation overhead: by employing multi-LLM collaboration across sizes/expertise the largest model doesn't need to be called every single time. MatFormer (Devvrit et al., 2024) simultaneously trains modules of varying sizes in a nested architecture and could be selectively activated to result in LLMs of varying sizes given the compute budget. Instead of training an LLM on *all* the data, approaches such as Branch-Train-Merge (Li et al., 2022) leverage the modularity of data provenance to train a pool of LLM experts and dynamically aggregated for inference. A growing line of research also investigates *defer* and *backoff* behavior between models of varying

sizes and/or specialization (Shen et al., 2024; Jung et al., 2024). These approaches highlight multi-LLM collaboration as a promising direction to balance utility and training/inference efficiency.

Adaptation Training a gigantic LLM and repurposing it with prompt engineering is the most popular status quo of LLM research and applications. However, one gigantic model is prohibitively expensive to re-train and update, while the effectiveness of prompt-based adaptation is limited and brittle (Sprague et al., 2024). Multi-LLM collaboration offers strategies for adapting language models that are lightweight and flexible: Token-level methods pair a general-purpose LLM with specialized peers for collaborative generation (Shen et al., 2024); logit-level approaches mix the logit distributions of multiple LLMs for collaborative decoding (Liu et al., 2024); weight-level approaches flexibly reuse and adapt existing models through weight arithmetic (Ilharco et al., 2023; Yadav et al., 2024b; Feng et al., 2024e). Multi-LLM collaboration offers diverse and flexible solutions for adaptation spanning varying levels of model access.

Privacy Despite the extensive effort to curate (pre)training data, private and copyrighted data will need to be left out for privacy, compliance, and ethics concerns (Karamolegkou et al., 2023; Yao et al., 2024). These data sets are nonetheless helpful in highly specialized or personalized contexts. Multi-LLM offers preliminary solutions where private/copyrighted data could be employed in a local model at the data provenance, then interact with a larger general-purpose model (Zhang et al., 2024). Though it might be possible to extract private data from the model (Carlini et al., 2021), we envision future work on augmenting the "private" LLM with contextual integrity guardrails (Mireshghallah et al., 2024) for controllable and context-aware information sharing (Ye et al., 2024a,b).

Democratization and collaborative development A single LLM is often trained by only a team of researchers and engineers, struggling to reflect the diversity of real-world LLM users. The priorities of long-tail and underprivileged users are often not incorporated when making decisions about model training and alignment. On the contrary, multi-LLM collaboration uniquely enables decentralized and collaborative development: all stakeholders in LLM development and applications could contribute models based on their needs, priorities, and compute budgets, then composed through various levels of multi-LLM collaboration protocols (§3). In this way, we democratize language technologies through participatory and collaborative development where everyone is welcome.

5 Limitations and Future Directions for Multi-LLM Collaboration

We identify various limitations of existing collaboration systems and motivate future work.

Theories of human communication While the status quo focuses on developing a single general-purpose LLM, there is no "general-purpose" human, but specialized individuals collaborating through communication protocols for collective intelligence (Hutchins, 2000). We argue that future multi-LLM collaboration research could benefit from cognitive science and communications theories, designing social science-inspired protocols for multiple LLMs to compose and collaborate.

Encapsulation and handoff Another interesting challenge in multi-LLM collaboration is the absence of clear handoff boundaries. In software engineering, *encapsulation* serves as a cornerstone of collaborative development by establishing well-defined interfaces between components: modifications to one part of the codebase do not propagate unexpected changes to others. However, cleanly separating and containing the expertise of different models remains an open challenge. While recent work has demonstrated progress in developing modularized model components (Pfeiffer et al., 2020; Hu et al., 2021; Yadav et al., 2024a), modifications to base model weights can still introduce unpredictable changes beyond the intended training objectives (for example, catastrophic forgetting (McCloskey and Cohen, 1989; Kirkpatrick et al., 2017)). Developing reliable encapsulation mechanisms can ensure robust and predictable model composition, and could be a critical step to achieve the vision for "building LMs like open-source software" (Raffel, 2021).

Compatibility with the status quo Despite the active research in multi-LLM collaboration, there is limited uptake in large-scale and industry settings beyond academic papers. One reason could be that many existing approaches require the training/development of extra modules such as gates and routers (Jiang et al., 2023a; Muqeeth et al.,

2024), while most open-source activities only feature the sharing of model weights. We thus argue that future protocols should be compatible with the status quo of model sharing by employing no extra step beyond using model checkpoints.

Interpretability insights Interpretability techniques unveil the mechanisms underlying LMs for reasoning (Stolfo et al., 2023), factual association (Meng et al., 2022), and more (Nanda et al., 2023). The interpretability insights enable localized manipulation of sub-modules for efficient enhancement (Yin et al., 2024), thereby facilitating the potential for lightweight model collaboration. Moreover, while diverse LMs may exhibit similar or distinct mechanisms for comparable tasks, the reliability of their capability beyond memorization varies (Yang et al., 2024). Interpretability tools offer insights into determining the fitting contribution of each component model in collaboration.

Cost The cost of model collaboration varies across collaboration levels (Figure 2). Generally, the inference cost follows text-level > logit-level > API-level > weight-level. Text-level collaboration is the most expensive since multiple models generate text depending on the output of each other: this increases latency and memory usage, but the collaboration of distinct models is very helpful in agent/compositional problem solving. Weight-level collaboration is the least expensive since multiple models are usually composed into one output model for deployment and inference. Multi-LLM collaboration offers a wide spectrum of technical approaches to choose from based on computing constraints, while we envision future work on reducing the cost of model collaboration.

Evaluating multi-LLM collaboration Research on modular and multi-LLM systems has not yet devised an agreed-upon evaluation methodology. Most of the existing work resorts to evaluation with tasks and datasets typical for a single LLM. Future work could explore specifically evaluating multi-LLM collaboration, designing tasks and datasets where multiple models are evaluated in collaboration, e.g., ablating by withholding copyright data (Min et al., 2024), or evaluating multi-agent collaboration where multiple models divide and conquer complex problems (Guo et al., 2024).

Democratizing ways of contribution While we hope that collaborative and participatory contributions to multi-LLM systems could alleviate the underrepresentation of people, *not everyone knows how to train an LLM*. This is especially true for the already underrepresented and underprivileged (Kirk et al., 2024), thus the benefits of collaboration will not reach them if we expect users to train and contribute models on their own. Thus, we argue that we should lower the barrier of contribution: for example, by designing an agent framework that solicits user feedback in natural language, fetches data, trains models, generates synthetic data to evaluate, and finally pushes the model and contributes. In this way, users only need to provide a few sentences of feedback about the gaps in existing LLMs, and a new component LLM could be developed and contributed on their behalf.

6 Related Work

Two recent position papers discuss related topics.

Yadav et al. (2024a) present a taxonomy of model merging and MoE approaches, arguing for reusing and routing of existing expert models. They focus primarily on weight-level collaboration approaches, while we aggregate a broader family of methods with a broader definition of *multi-LLM collaboration* where models could collaborate through four different levels of information exchange.

Du and Kaelbling (2024) present a position paper arguing for compositional generative modeling, discussing the benefits of combining multiple modules across computer vision, reinforcement learning, robotics, and a brief mention of language. We focus on language models and dive deep into LLM-specific arguments, methods, and future research.

7 Conclusion

We argue that one LLM is not enough and advocate for multi-LLM collaboration to better represent diverse data distributions, heterogeneous skills, and pluralistic populations. We propose a hierarchy of existing multi-LLM collaboration approaches based on information exchange levels, spanning API-level, text-level, logit-level, and weight-level collaboration. We then summarize the benefits of existing multi-LLM systems over a single model and discuss the limitations of existing methods to motivate future work. We envision multi-LLM collaboration as a viable path to compositional intelligence and an important initiative toward collaborative AI development.

Limitations

As a position paper, we discuss the limitations of existing multi-LLM collaboration approaches in Section 5 and motivate future work that addresses these limitations. We also identify two alternative positions to ours and discuss them in Appendix A.

Acknowledgements

This research was developed in part with funding from the Defense Advanced Research Projects Agency's (DARPA) SciFy program (Agreement No. HR00112520300). The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. This research was supported by the Coefficient Giving and Amazon Health. Shangbin Feng would like to thank the support of the IBM PhD Fellowship, the Jane Street Graduate Research Fellowship, and the NVIDIA Graduate Fellowship.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Rachit Bansal, Bidisha Samanta, Siddharth Dalmia, Nitish Gupta, Sriram Ganapathy, Abhishek Bapna, Prateek Jain, and Partha Talukdar. 2024. Llm augmented llms: Expanding capabilities through composition. In *The Twelfth International Conference on Learning Representations*.

Joshua Belofsky. 2023. Token-level adaptation of lora adapters for downstream task generalization. In *Proceedings of the 2023 6th Artificial Intelligence and Cloud Computing Conference*, pages 168–172.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *ArXiv*, abs/2005.14165.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, and 1 others. 2024. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. In *Forty-first International Conference on Machine Learning*.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, and 1 others. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Lingjiao Chen, Matei Zaharia, and James Zou. 2023. Frugalgpt: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*.

Tong Chen, Akari Asai, Niloofar Mireshghallah, Sewon Min, James Grimmelmann, Yejin Choi, Hannaneh Hajishirzi, Luke Zettlemoyer, and Pang Wei Koh. 2024. Copybench: Measuring literal and non-literal reproduction of copyright-protected text in language model generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15134–15158.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, and 1 others. 2024a. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, and 1 others. 2024b. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R. Glass, and Pengcheng He. 2024. Dola: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*.

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. 2023. Lm vs lm: Detecting factual errors via cross examination. *arXiv preprint arXiv:2305.13281*.

Fnu Devvrit, Sneha Kudugunta, Aditya Kusupati, Tim Dettmers, Kaifeng Chen, Inderjit S Dhillon, Yulia Tsvetkov, Hannaneh Hajishirzi, Sham M Kakade, Ali Farhadi, , and Prateek Jain. 2024. Matformer: Nested transformer for elastic inference. In *NeurIPS*.

Bhuwan Dhingra, Jeremy R Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Shizhe Diao, Tianyang Xu, Ruijia Xu, Jiawei Wang, and Tong Zhang. 2023. Mixture-of-domain-adapters: Decoupling and injecting domain knowledge to pretrained language models memories. *arXiv preprint arXiv:2306.05406*.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Ruhle, Laks VS Lakshmanan, and Ahmed Hassan Awadallah. 2024. Hybrid llm: Cost-efficient and quality-aware query routing. *arXiv preprint arXiv:2404.14618*.

Yanrui Du, Sendong Zhao, Danyang Zhao, Ming Ma, Yuhan Chen, Liangyu Huo, Qing Yang, Dongliang Xu, and Bing Qin. 2024a. MoGU: A framework for enhancing safety of LLMs while preserving their usability. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yilun Du and Leslie Pack Kaelbling. 2024. Position: Compositional generative modeling: A single model is not all you need. In *Forty-first International Conference on Machine Learning*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2024b. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning*.

EEOC. 2024. High tech, low inclusion: Diversity in the high tech workforce and sector 2014 - 2022.

Fahim Faisal, Orevaoghene Ahia, Aarohi Srivastava, Kabir Ahuja, David Chiang, Yulia Tsvetkov, and Antonios Anastasopoulos. 2024. DIALECTBENCH: An NLP benchmark for dialects, varieties, and closely-related languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Chenghao Fan, Zhenyi Lu, Wei Wei, Jie Tian, Xiaoye Qu, Dangyang Chen, and Yu Cheng. 2024. On giant's shoulders: Effortless weak to strong by dynamic logits fusion. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Shangbin Feng, Chan Young Park, Yuhan Liu, and Yulia Tsvetkov. 2023a. From pretraining data to language models to downstream tasks: Tracking the trails of political biases leading to unfair nlp models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11737–11762.

Shangbin Feng, Weijia Shi, Yuyang Bai, Vidhisha Balachandran, Tianxing He, and Yulia Tsvetkov. 2023b. Cook: Empowering general-purpose language models with modular and collaborative knowledge. *arXiv preprint arXiv:2305.09955*.

Shangbin Feng, Weijia Shi, Yuyang Bai, Vidhisha Balachandran, Tianxing He, and Yulia Tsvetkov. 2024a. Knowledge card: Filling llms' knowledge gaps with plug-in specialized language models. In *The Twelfth International Conference on Learning Representations*.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024b. Don't hallucinate, abstain: Identifying LLM knowledge gaps via multi-LLM collaboration. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024c. Don't hallucinate, abstain: Identifying llm knowledge gaps via multi-llm collaboration. *arXiv preprint arXiv:2402.00367*.

Shangbin Feng, Taylor Sorensen, Yuhan Liu, Jillian Fisher, Chan Young Park, Yejin Choi, and Yulia Tsvetkov. 2024d. Modular pluralism: Pluralistic alignment via multi-llm collaboration. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4151–4171.

Shangbin Feng, Zifeng Wang, Yike Wang, Sayna Ebrahimi, Hamid Palangi, Lesly Miculicich, Achin Kulshrestha, Nathalie Rauschmayr, Yejin Choi, Yulia Tsvetkov, and 1 others. 2024e. Model swarms: Collaborative search to adapt llm experts via swarm intelligence. *arXiv preprint arXiv:2410.11163*.

Tao Feng, Yanzhen Shen, and Jiaxuan You. 2024f. Graphrouter: A graph-based router for llm selections. *arXiv preprint arXiv:2410.03834*.

Clémentine Fourrier, Nathan Habib, Alina Lozovskaya, Konrad Szafer, and Thomas Wolf. 2024. Open llm leaderboard v2.

Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, pages 1–79.

Ariel Gera, Roni Friedman, Ofir Arviv, Chulaka Gunasekara, Benjamin Sznajder, Noam Slonim, and Eyal Shnarch. 2023. The benefits of bad advice: Autocontrastive decoding across model layers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. Large language model based multi-agents: A survey of progress and challenges. *arXiv preprint arXiv:2402.01680*.

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. 2024. Language model cascades:

Token-level uncertainty and beyond. *arXiv preprint arXiv:2404.10136*.

Suchin Gururangan, Margaret Li, Mike Lewis, Weijia Shi, Tim Althoff, Noah A Smith, and Luke Zettlemoyer. 2023. Scaling expert language models with unsupervised domain discovery. *arXiv preprint arXiv:2303.14177*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Will Henshall. 2024. Big tech companies were investors in smaller ai labs. now they're rivals. https://time.com/6977424/ai-competition-openai-anthropic-microsoft-amazon/.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, and 1 others. 2022. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pages 30016–30030.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Qitian Jason Hu, Jacob Bieker, Xiuyu Li, Nan Jiang, Benjamin Keigwin, Gaurav Ranganath, Kurt Keutzer, and Shriyash Kaustubh Upadhyay. 2024. Routerbench: A benchmark for multi-llm routing system. *arXiv preprint arXiv:2403.12031*.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. 2023. Lorahub: Efficient cross-task generalization via dynamic lora composition. *arXiv preprint arXiv:2307.13269*.

Edwin Hutchins. 2000. Distributed cognition. *International Encyclopedia of the Social and Behavioral Sciences. Elsevier Science*, 138:1–10.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. 2023. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. *arXiv preprint arXiv:2310.11564*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023a. Llm-blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14165–14178.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023b. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992.

Justin P Johnson. 2006. Collaboration, peer review and open source software. *Information Economics and Policy*, 18(4):477–497.

Jaehun Jung, Faeze Brahman, and Yejin Choi. 2024. Trust or escalate: Llm judges with provable guarantees for human agreement. *arXiv preprint arXiv:2407.18370*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Antonia Karamolegkou, Jiaang Li, Li Zhou, and Anders Søgaard. 2023. Copyright violations and large language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Jungo Kasai, Keisuke Sakaguchi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A Smith, Yejin Choi, Kentaro Inui, and 1 others. 2024. Realtime qa: what's the answer right now? *Advances in Neural Information Processing Systems*, 36.

Hannah Rose Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, and 1 others. 2024. The prism alignment project: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. *arXiv preprint arXiv:2404.16019*.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, and 1 others. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

Sachin Kumar, Vidhisha Balachandran, Lucille Njoo, Antonios Anastasopoulos, and Yulia Tsvetkov. 2023. Language generation models can cause harm: So what can we do about it? an actionable survey. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3299–3321.

Sachin Kumar, Chan Young Park, Yulia Tsvetkov, Noah A Smith, and Hannaneh Hajishirzi. 2024. Compo: Community preferences for language model personalization. *arXiv preprint arXiv:2410.16027*.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Sebastian Ruder, Dani Yogatama, and 1 others. 2021. Pitfalls of static language modelling. *arXiv preprint arXiv:2102.01951*.

Joel Z Leibo, Alexander Sasha Vezhnevets, Manfred Diaz, John P Agapiou, William A Cunningham, Peter Sunehag, Julia Haas, Raphael Koster, Edgar A Duéñez-Guzmán, William S Isaac, and 1 others. 2024. A theory of appropriateness with applications to generative artificial intelligence. *arXiv preprint arXiv:2412.19010*.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. 2024. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13872–13882.

Margaret Li, Suchin Gururangan, Tim Dettmers, Mike Lewis, Tim Althoff, Noah A Smith, and Luke Zettlemoyer. 2022. Branch-train-merge: Embarrassingly parallel training of expert language models. *arXiv preprint arXiv:2208.03306*.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2023a. Holistic evaluation of language models. *Transactions on Machine Learning Research*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2023b. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*.

Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, and 1 others. 2024. Mitigating the alignment tax of rlhf. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 580–606.

Alisa Liu, Xiaochuang Han, Yizhong Wang, Yulia Tsvetkov, Yejin Choi, and Noah A. Smith. 2024. Tuning language models by proxy. In *First Conference on Language Modeling*.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. DExperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Keming Lu, Hongyi Yuan, Runji Lin, Junyang Lin, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2023. Routing to the expert: Efficient reward-guided ensemble of large language models. *arXiv preprint arXiv:2311.08692*.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2023. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, and 1 others. 2024. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36.

Costas Mavromatis, Petros Karypis, and George Karypis. 2024. Pack of LLMs: Model fusion at test-time via perplexity optimization. In *First Conference on Language Modeling*.

Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 36. ArXiv:2202.05262.

Sewon Min, Suchin Gururangan, Eric Wallace, Weijia Shi, Hannaneh Hajishirzi, Noah A Smith, and Luke Zettlemoyer. 2024. Silo language models: Isolating legal risk in a nonparametric datastore. In *The Twelfth International Conference on Learning Representations*.

Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. 2024. Can llms keep a secret? testing privacy implications of language models via contextual integrity theory. In *The Twelfth International Conference on Learning Representations*.

Eric Mitchell, Rafael Rafailov, Archit Sharma, Chelsea Finn, and Christopher D Manning. 2024. An emulator for fine-tuning large language models using small language models. In *The Twelfth International Conference on Learning Representations*.

Mohammed Muqeeth, Haokun Liu, Yufan Liu, and Colin Raffel. 2024. Learning to route among specialized experts for zero-shot generalization. In *Forty-first International Conference on Machine Learning*.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations*.

Tarek Naous, Michael J Ryan, Alan Ritter, and Wei Xu. 2023. Having beer after prayer? measuring cultural bias in large language models. *arXiv preprint arXiv:2305.14456*.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E Gonzalez, M Waleed Kadous, and Ion Stoica. 2024. Routellm: Learning to route llms with preference data. *arXiv preprint arXiv:2406.18665*.

Jonathan Pei, Kevin Yang, and Dan Klein. 2023. PREADD: Prefix-adaptive decoding for controlled text generation. In *Findings of the Association for Computational Linguistics: ACL 2023*.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2020. Adapterfusion: Non-destructive task composition for transfer learning. *arXiv preprint arXiv:2005.00247*.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *ArXiv*, abs/2210.03350.

Colin Raffel. 2021. A call to build models like we build open-source software. Accessed: 2025-01-15.

Abhinav Rao, Akhila Yerukola, Vishwa Shah, Katharina Reinecke, and Maarten Sap. 2024. Normad: A benchmark for measuring the cultural adaptability of large language models. *arXiv preprint arXiv:2404.12464*.

Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto. 2023. Whose opinions do language models reflect? In *International Conference on Machine Learning*, pages 29971–30004. PMLR.

Rico Sennrich, Jannis Vamvas, and Alireza Mohammadshahi. 2024. Mitigating hallucinations and off-target machine translation with source-contrastive and language-contrastive decoding. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Zejiang Shen, Hunter Lang, Bailin Wang, Yoon Kim, and David Sontag. 2024. Learning to decode collaboratively with multiple language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Ruizhe Shi, Yifang Chen, Yushi Hu, Alisa Liu, Hannaneh Hajishirzi, Noah A. Smith, and Simon Shaolei Du. 2024a. Decoding-time language model alignment with multiple objectives. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. 2024b. Trusting your evidence: Hallucinate less with context-aware decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*.

Weiyan Shi, Ryan Li, Yutong Zhang, Caleb Ziems, Raya Horesh, Rogério Abreu de Paula, Diyi Yang, and 1 others. 2024c. Culturebank: An online community-driven knowledge base towards culturally aware language technologies. *arXiv preprint arXiv:2404.15238*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2024. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36.

Tal Shnitzer, Anthony Ou, Mírian Silva, Kate Soule, Yuekai Sun, Justin Solomon, Neil Thompson, and Mikhail Yurochkin. 2023. Large language model routing with benchmark datasets. *arXiv preprint arXiv:2309.15789*.

Yueqi Song, Simran Khanuja, Pengfei Liu, Fahim Faisal, Alissa Ostapenko, Genta Winata, Alham Aji, Samuel Cahyawijaya, Yulia Tsvetkov, Antonios Anastasopoulos, and 1 others. 2023. Globalbench: A benchmark for global progress in natural language processing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14157–14171.

Taylor Sorensen, Liwei Jiang, Jena D Hwang, Sydney Levine, Valentina Pyatkin, Peter West, Nouha Dziri, Ximing Lu, Kavel Rao, Chandra Bhagavatula, and 1 others. 2024a. Value kaleidoscope: Engaging ai with pluralistic human values, rights, and duties. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19937–19947.

Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell L Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, and 1 others. 2024b. Position: A roadmap to pluralistic alignment. In *Forty-first International Conference on Machine Learning*.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2024. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. *arXiv preprint arXiv:2409.12183*.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. 2023. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Dimitris Stripelis, Zijian Hu, Jipeng Zhang, Zhaozhuo Xu, Alay Dilipbhai Shah, Han Jin, Yuhang Yao, Salman Avestimehr, and Chaoyang He. 2024. Tensoropera router: A multi-model router for efficient llm inference. *arXiv preprint arXiv:2408.12320*.

Vighnesh Subramaniam, Yilun Du, Joshua B Tenenbaum, Antonio Torralba, Shuang Li, and Igor Mordatch. 2025. Multiagent finetuning: Self improvement with diverse reasoning chains. *arXiv preprint arXiv:2501.05707*.

Vighnesh Subramaniam, Antonio Torralba, and Shuang Li. 2024. Debategpt: Fine-tuning large language models with multi-agent debate supervision.

Sainbayar Sukhbaatar, Olga Golovneva, Vasu Sharma, Hu Xu, Xi Victoria Lin, Baptiste Rozière, Jacob Kahn, Daniel Li, Wen-tau Yih, Jason Weston, and 1 others. 2024. Branch-train-mix: Mixing expert llms into a mixture-of-experts llm. *arXiv preprint arXiv:2403.07816*.

Tian-Xiang Sun, Xiang-Yang Liu, Xi-Peng Qiu, and Xuan-Jing Huang. 2022. Paradigm shift in natural language processing. *Machine Intelligence Research*, 19(3):169–183.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and 1 others. 2023. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051.

Florian Tambon, Amin Nikanjam, Foutse Khomh, and Giuliano Antoniol. 2024. Assessing programming task difficulty for efficient evaluation of large language models. *arXiv preprint arXiv:2407.21227*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

InternLM Team. 2023. Internlm: A multilingual language model with progressively enhanced capabilities.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Hongyi Wang, Felipe Maia Polo, Yuekai Sun, Souvik Kundu, Eric Xing, and Mikhail Yurochkin. 2024. Fusing models with complementary expertise. In *The Twelfth International Conference on Learning Representations*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022a. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Yaqing Wang, Sahaj Agarwal, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Hassan Awadallah, and Jianfeng Gao. 2022b. Adamix: Mixture-of-adaptations for parameter-efficient model tuning. *Preprint*, arXiv:2205.12410.

Boyi Wei, Weijia Shi, Yangsibo Huang, Noah A Smith, Chiyuan Zhang, Luke Zettlemoyer, Kai Li, and Peter Henderson. 2024. Evaluating copyright takedown methods for language models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, and 1 others. 2024a. Autogen: Enabling next-gen llm applications via multi-agent conversation. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*.

Shirley Wu, Michel Galley, Baolin Peng, Hao Cheng, Gavin Li, Yao Dou, Weixin Cai, James Zou, Jure Leskovec, and Jianfeng Gao. 2025. Collabllm: From passive responders to active collaborators. *arXiv preprint arXiv:2502.00640*.

Xun Wu, Shaohan Huang, and Furu Wei. 2024b. Mixture of lora experts. *arXiv preprint arXiv:2404.13628*.

Kai Xiong, Xiao Ding, Yixin Cao, Ting Liu, and Bing Qin. 2023. Examining inter-consistency of large language models collaboration: An in-depth analysis via debate. *arXiv preprint arXiv:2305.11595*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Tianyang Xu, Shujin Wu, Shizhe Diao, Xiaoze Liu, Xingyao Wang, Yangyi Chen, and Jing Gao. 2024. Sayself: Teaching llms to express confidence with self-reflective rationales. *arXiv preprint arXiv:2405.20974*.

Prateek Yadav, Colin Raffel, Mohammed Muqeeth, Lucas Caccia, Haokun Liu, Tianlong Chen, Mohit Bansal, Leshem Choshen, and Alessandro Sordoni. 2024a. A survey on model moerging: Recycling and routing among specialized experts for collaborative learning. *arXiv preprint arXiv:2408.07057*.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. 2024b. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. 2024. Do large language models latently perform multi-hop reasoning? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Zhibo Sun, and Yue Zhang. 2024. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing*, page 100211.

Rui Ye, Rui Ge, Xinyu Zhu, Jingyi Chai, Du Yaxin, Yang Liu, Yanfeng Wang, and Siheng Chen. 2024a. Fedllm-bench: Realistic benchmarks for federated learning of large language models. *Advances in Neural Information Processing Systems*, 37:111106–111130.

Rui Ye, Wenhao Wang, Jingyi Chai, Dihan Li, Zexi Li, Yinda Xu, Yaxin Du, Yanfeng Wang, and Siheng Chen. 2024b. Openfedllm: Training large language models on decentralized private data via federated learning. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 6137–6147.

Fangcong Yin, Xi Ye, and Greg Durrett. 2024. Lofit: Localized fine-tuning on llm representations. *arXiv preprint arXiv:2406.01563*.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*.

Murong Yue, Jie Zhao, Min Zhang, Liang Du, and Ziyu Yao. 2023. Large language model cascades with mixture of thoughts representations for cost-efficient reasoning. *arXiv preprint arXiv:2310.03094*.

Kaiyan Zhang, Jianyu Wang, Ermo Hua, Biqing Qi, Ning Ding, and Bowen Zhou. 2024. Cogenesis: A framework collaborating large and small language models for secure context-aware instruction following. *arXiv preprint arXiv:2403.03129*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and 1 others. 2023. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Justin Zhao, Flor Miriam Plaza-del Arco, and Amanda Cercas Curry. 2024a. Language model council: Benchmarking foundation models on highly subjective tasks by consensus. *arXiv preprint arXiv:2406.08598*.

Qinlin Zhao, Jindong Wang, Yixuan Zhang, Yiqiao Jin, Kaijie Zhu, Hao Chen, and Xing Xie. 2024b. Competeai: Understanding the competition dynamics of large language model-based agents. In *Forty-first International Conference on Machine Learning*.

A Alternative Views

We identify two alternative views to our position.

We could patch the underrepresentations of data, skills, and people by further augmenting a single model. While existing band-aid approaches such as LoRA fine-tuning (Hu et al., 2021) or retrieval augmented generation (RAG) (Shi et al., 2023; Jiang et al., 2023b; Asai et al., 2024) patch the gaps in data and skills to some extent, we present empirical evidence of their limitations in Section 2, suffering from challenges such as privacy and copyright, catastrophic forgetting, lack of participation, and more. Further fine-tuning with LoRA could patch the gap of skills, but it risks jeopardizing the general-purposeness and leads to trade-offs of existing skills (Kirkpatrick et al., 2017); retrieval could provide new information and data to improve reliability, but there is no guarantee that LLMs would fully leverage the retrieved context (Shi et al., 2024b). While it is not impossible that with future progress a single LLM could offer perfect representations, we argue that multi-LLM collaboration offers a more concrete and actionable roadmap to advance language technologies, and a more efficient one, as it reuses developments made so far.

We could enable collaboration through a single model. It is theoretically possible to collaborate in the development lifecycle of a single model. Different communities could contribute heterogeneous data to be combined for training a single model; different engineering teams could train part of the model architecture for later merging; different users could annotate diverse alignment preferences to jointly align an LLM. We argue that while they are all possible, it is more natural to collaborate on the level of models since 1) model sharing is the

default open-source activity, 2) there are already 2,134,229 LLMs[2] openly available for collaboration, and 3) the companies that have the resource to carry out these protocols are incentivized to not go open about development of their LLM for competitive advantage. We envision multi-LLM collaboration as a promising path to reuse existing models, promote collaborative development, and advance compositional intelligence.

[2]Huggingface accessed on Oct 4, 2025.