

Scaling Large-Language-Model-based Multi-Agent Collaboration

Chen Qian^{†*} Zihao Xie^{†*} Yifei Wang^{†*} Wei Liu^{*} Yufan Dang^{*}
 Zhuoyun Du^{*} Weize Chen^{*} Cheng Yang[♣] Zhiyuan Liu^{*} Maosong Sun[✉]

^{*}Tsinghua University [♣]Beijing University of Posts and Telecommunications
 qianc62@gmail.com sms@tsinghua.edu.cn

Abstract

Pioneering advancements in large language model-powered agents have underscored the design pattern of multi-agent collaboration, demonstrating that collective intelligence can surpass the capabilities of each individual. Inspired by the neural scaling law, which posits that increasing neurons leads to emergent abilities, this study investigates whether a similar principle applies to increasing agents in multi-agent collaboration. Technically, we propose multi-agent collaboration networks (MACNET), which utilize directed acyclic graphs to organize agents and streamline their interactive reasoning via topological ordering, with solutions derived from their dialogues. Extensive experiments show that MACNET consistently outperforms baseline models, enabling effective agent collaboration across various network topologies and supporting cooperation among more than a thousand agents. Notably, we observed a *small-world collaboration phenomenon*, where topologies resembling small-world properties achieved superior performance. Additionally, we identified a *collaborative scaling law*, indicating that normalized solution quality follows a logistic growth pattern as scaling agents, with collaborative emergence occurring much earlier than previously observed instances of neural emergence. The code and data will be available at <https://github.com/OpenBMB/ChatDev>.

1 Introduction

In the rapidly advancing field of artificial intelligence, *large language models* (LLMs) have catalyzed transformative shifts across numerous domains due to their remarkable linguistic capacity to seamlessly integrate extensive world knowledge (Vaswani et al., 2017; Brown et al., 2020; Bubeck et al., 2023). Central to this breakthrough

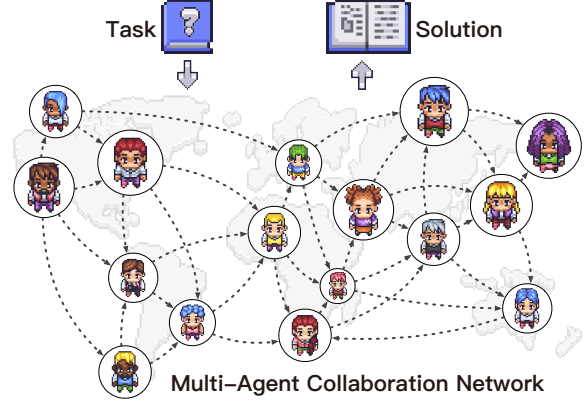


Figure 1: Given a task, multi-agent collaboration networks (MACNET) utilize directed acyclic graphs to organize diverse agents for collaborative interactions, with the final solution derived from their dialogues.

is the *neural scaling law* that fosters emergent capabilities, where well-trained neural networks often exhibit power-law scaling relations primarily with the number of neurons, alongside factors such as dataset size and training time (Kaplan et al., 2020; Schaeffer et al., 2024; Muennighoff et al., 2024).

Despite this, LLMs have inherent limitations in enclosed reasoning, particularly when addressing complex situations that extend beyond textual boundaries (Richards, 2023). To this end, subsequent studies have successfully transformed foundational LLMs into versatile *autonomous agents* by equipping them with advanced capabilities such as tool use (Schick et al., 2023), long-context memory (Park et al., 2023), and procedural planning (Wei et al., 2022b). Along this line, *multi-agent collaboration* has emerged as an effective paradigm to integrate the specialties of different agents (Park et al., 2023; Li et al., 2023a; Qian et al., 2024b). Through linguistic interaction, agents engage in instructive and responsive utterances to foster high-quality collaboration, leading to final

[†]Equal Contribution.

[✉]Corresponding Author.

solutions¹ derived from their dialogues (Qian et al., 2024b,a; Chen et al., 2024b).

Inspired by the neural scaling law, a natural question arises: *does increasing agents in multi-agent collaboration exhibit emergent capabilities?* Investigating the *cooperative scaling law* is essential for accurately estimating the relationship between computing resources and performance trends in multi-agent systems. This understanding enables the optimization of resource utilization and the minimization of unnecessary waste, ultimately leading to more scalable, practical, and resource-efficient agent systems (Kaplan et al., 2020). However, effective multi-agent collaboration transcends the mere aggregation of responses from different agents through majority voting (Chen et al., 2024a); instead, it constitutes an organically integrated system that requires task-oriented interactions and thoughtful decision-making (Hopfield, 1982).

In this paper, as illustrated in Figure 1, we envision multiple agents as a well-organized team composed of specialized agents, investigating their interdependent interactive reasoning and collective intelligence in autonomously solving complex problems. To further this goal, we design appropriate topologies and effective interaction mechanisms that align with both the static organizational structure and the dynamic reasoning process.

- To ensure generalizability, we design the topology as a directed acyclic graph where each edge is managed by a supervisory instructor issuing directional commands, and each node is supported by an executive assistant providing tailored solutions. This mechanism effectively fosters a division of labor among agents through functional dichotomy, seamlessly integrating a static topology with specialized agents to form a *multi-agent collaboration network* (MACNET).
- To facilitate agents’ interactive reasoning, the interaction sequence is orchestrated via topological ordering, ensuring orderly information transmission throughout the network. Within this arrangement, each interaction round involves two adjacent agents refining a previous solution, with only the refined solution, rather than the entire dialogue, being propagated to the next neighbors. This mechanism strategically avoids global broadcasts and significantly reduces the risk of

overly extended contexts, enabling scalable collaboration across nearly any large-scale network.

We conducted a comprehensive quantitative evaluation of three prevalent topologies—chain, tree, and graph—divided into six variants, across multiple heterogeneous downstream scenarios. Extensive experiments demonstrate that MACNET consistently outperforms all baseline, enabling effective agent collaboration even in fully-connected dense networks, supporting cooperation among more than a thousand agents. Notably, we observed a *small-world collaboration phenomenon*, where topologies resembling small-world properties demonstrated superior performance. Additionally, we identified a *collaborative scaling law*, revealing that normalized solution quality follows a logistic growth pattern as scaling agents. Meanwhile, collaborative emergence can be observed occurring significantly earlier compared to previous instances of neural emergence. We hope our findings offer valuable insights into resource prediction and optimization to enhance the efficiency and scalability of LLM systems (Kaplan et al., 2020).

2 Related Work

Trained on vast datasets and capable of manipulating billions of parameters, LLMs have become pivotal in natural language processing due to their seamless integration of extensive knowledge (Brown et al., 2020; Bubeck et al., 2023; Vaswani et al., 2017; Radford et al., 2019; Touvron et al., 2023; Wei et al., 2022a; Shanahan et al., 2023; Chen et al., 2021; Brants et al., 2007; Chen et al., 2021; Ouyang et al., 2022; Yang et al., 2024; Qin et al., 2023; Kaplan et al., 2020). Central to this breakthrough is the neural scaling law, which posits that loss scales as a power-law with model size, dataset size, and the amount of compute used for training (Kaplan et al., 2020; Smith et al., 2022; Muennighoff et al., 2024). The principle underscores that scaling up language models can lead to emergent abilities—where performance experiences a sudden leap as the model scales (Wei et al., 2022a; Schaeffer et al., 2024).

Despite these, LLMs have inherent limitations in enclosed reasoning, motivating subsequent studies to effectively equip LLMs with advanced capabilities such as role playing (Li et al., 2023a; Chan et al., 2024), tool use (Schick et al., 2023; Qin et al., 2024), long-context memory (Park et al., 2023; Wang et al., 2023), and procedural plan-

¹Solutions can range from a multiple-choice answer to repository-level code or a coherent narrative, among numerous other possibilities.

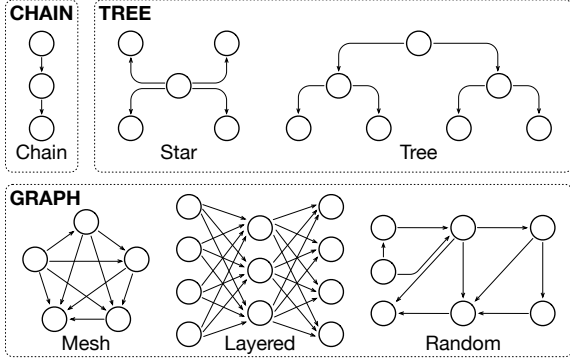


Figure 2: Representative topological structures.

ning (Wei et al., 2022b; Yao et al., 2023), thereby transforming fundamental LLMs into versatile autonomous agents (Richards, 2023; Shinn et al., 2024; Zhao et al., 2024). Along this line, multi-agent collaboration has emerged as an effective paradigm to integrate the specialities of different agents (Park et al., 2023; Zhou et al., 2023; Chen et al., 2024b; Chan et al., 2024; Chen et al., 2023; Cohen et al., 2023; Li et al., 2023b; Hua et al., 2023). A straightforward collaboration strategy is majority voting (Chen et al., 2024a), where individuals remain independent; however, more effective multi-agent collaboration should form an integrated system that fosters interdependent interactions and thoughtful decision-making (Li et al., 2024; Chen et al., 2024a; Piatti et al., 2024). Based on this, pioneering studies have dichotomized the functionality of agents into two distinct roles: instructors, who provide directional instructions, and assistants, who respond with tailored solutions; these agents engage in instructive and responsive utterances to foster an interaction chain, collaboratively arriving at final solutions derived from their dialogues (Qian et al., 2024b; Li et al., 2023a). This paradigm facilitates a well-orchestrated workflow for task-oriented interactions, significantly reducing the need for manual intervention while demonstrating promising quality. (Chen et al., 2024b; Chan et al., 2024).

3 Multi-Agent Collaboration Network

We aim to establish a scalable framework for multi-agent collaboration, comprising two key components: the design of a multi-agent collaboration network (MACNET) and collaborative reasoning.

3.1 Network Construction

To establish a organizational structure for multi-agent collaboration that is both efficient and scalable,

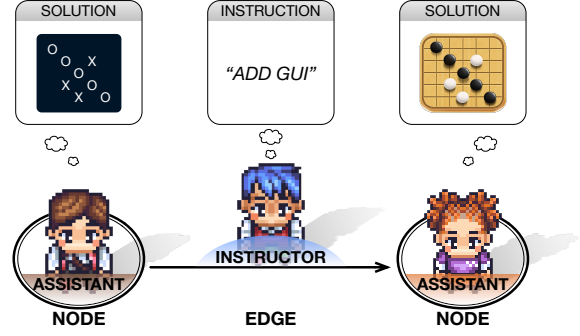


Figure 3: Assign different agents on nodes and edges.

drawing on the concept of graphs—a data structure that describes entities and their interrelations, we model the topology as a directed acyclic graph (DAG) (Nilsson et al., 2020) to organize interactions among collaborative agents (Qian et al., 2024a). Concretely, a feasible topology is denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$:

$$\mathcal{V} = \{v_i | i \in I\}, \mathcal{E} = \{\langle v_i, v_j \rangle | i, j \in I \wedge i \neq j\} \quad (1)$$

where \mathcal{V} denotes the set of nodes indexed by I , and \mathcal{E} denotes the set of edges, with each edge directed from one node to another and no cycles exist.

Given the impracticality of enumerating all possible topologies, our study focus on three prevalent types—chain, tree, and graph—further divided into six structures, as depicted in Figure 2. Chain topologies, resembling the waterfall model (Petersen et al., 2009), linearly structuring interactions along agents. Tree topologies enable agents to branch out, interacting in independent directions; further categorized into "wider" star-shaped and "deeper" tree-shaped structures. Graph topologies support arbitrary interaction dependencies, with nodes having multiple children and parents, forming either divergent and convergent interactions; further classified into fully-connected mesh structures, MLP-shaped layered structures, and irregular random structures. These representative topologies are extensively studied in complex network (Strogatz, 2001; Albert and Barabási, 2001) and LLM agent reasoning (Liu et al., 2023; Besta et al., 2024), ensuring a comprehensive examination of the most significant and practical structures in understanding multi-agent systems.

In the ecosystem of LLM-powered agents, a functional dichotomy (Li et al., 2023a)—consisting of supervisory instructors who issue directional instructions and executive assistants who provide tailored solutions—can effectively promote division

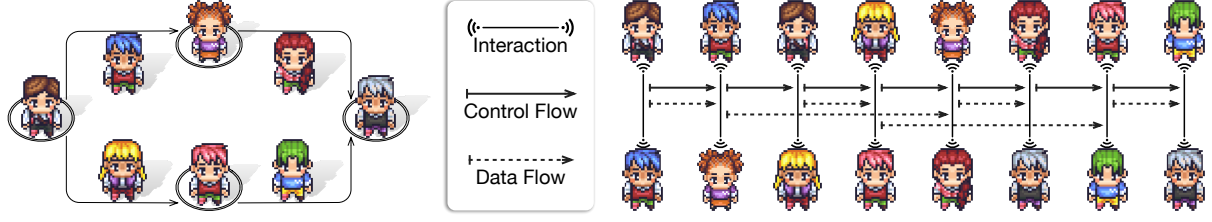


Figure 4: Streamlining the agents' reasoning process involves a series of dual-agent interactions. The topological order guides the interaction sequence, while the original connectivity governs the data flow.

of labor, stimulate functional behaviors, and facilitate efficient task resolution (Qian et al., 2024b,a). To integrate this strategy into the topology, as depicted in Figure 3, we strategically assign an instructor to each edge and an assistant to each node:

$$\begin{aligned} \mathbf{a}_i &= \rho(v_i), \forall v_i \in \mathcal{V} \\ \mathbf{a}_{ij} &= \rho(\langle v_i, v_j \rangle), \forall \langle v_i, v_j \rangle \in \mathcal{E} \end{aligned} \quad (2)$$

where $\rho(x)$ represents the agentization operation on an element x , achieved by equipping a foundation model with professional roles (Li et al., 2023a), external tools (Schick et al., 2023), and context-aware memory (Park et al., 2023); \mathbf{a}_i and \mathbf{a}_{ij} denote an assistant agent assigned to node v_i and an instructor agent assigned to edge v_{ij} , respectively.

This dichotomous design allows agents to specialize in their functions, driving task-oriented language interactions and facilitating efficient information transmission throughout the network. Additionally, the "directed" nature of the edges enables the orchestration of agent interactions, while the "acyclic" configuration prevents information propagation deadlocks.

3.2 Interactive Reasoning

In the process of completing complex tasks, interactive reasoning among agents within a static MACNET requires strategical traversal to establish an orderly interaction sequence. Our graph traversal strategy adheres to the principles of topological ordering (Bondy and Murty, 1976), a fundamental algorithm in graph theory, which ensures that each node is visited only after all its dependencies have been traversed (Gross et al., 2018). Formally, for a MACNET \mathcal{G} , its topological order is a linear arrangement of agents \mathbf{a}_i and \mathbf{a}_{ij} such that for every directed edge $\langle v_i, v_j \rangle \in \mathcal{E}$, the ordering satisfies:

$$\forall \langle v_i, v_j \rangle \in \mathcal{E}, \mathbb{I}(\mathbf{a}_i) < \mathbb{I}(\mathbf{a}_{ij}) < \mathbb{I}(\mathbf{a}_j) \quad (3)$$

where $\mathbb{I}(x)$ denotes the index of agent x in the topological sequence. This arrangement ensures that

each node-occupied agent \mathbf{a}_i precedes its corresponding edge-occupied agent \mathbf{a}_{ij} , and \mathbf{a}_{ij} precedes \mathbf{a}_j , thereby guaranteeing ensuring orderly information transmission throughout the network.

After establishing the global order, as illustrated in Figure 4, we enable each pair of edge-connected adjacent agents to interact and exchange information. For a topology \mathcal{G} , the design result in a total deployment of $|\mathcal{V}| + |\mathcal{E}|$ agents and require $2|\mathcal{E}|$ interaction rounds. Within each edge, the interaction pattern between assistants and instructors follows a multi-turn instruction-response sequence:

$$\begin{aligned} \tau(\mathbf{a}_i, \mathbf{a}_{ij}, \mathbf{a}_j) &= (\tau(\mathbf{a}_i, \mathbf{a}_{ij}), \tau(\mathbf{a}_{ij}, \mathbf{a}_j)) \\ \tau(\mathbf{a}_i, \mathbf{a}_{ij}) &= (\mathbf{a}_i \rightarrow \mathbf{a}_{ij}, \mathbf{a}_{ij} \rightsquigarrow \mathbf{a}_i)_{\circ} \\ \tau(\mathbf{a}_{ij}, \mathbf{a}_j) &= (\mathbf{a}_{ij} \rightarrow \mathbf{a}_j, \mathbf{a}_j \rightsquigarrow \mathbf{a}_{ij})_{\circ} \end{aligned} \quad (4)$$

where \rightarrow symbolizes the act of instructing, \rightsquigarrow indicates the corresponding responding, and \circ represents the iterative nature of the process. Specifically, \mathbf{a}_i requests feedback, \mathbf{a}_{ij} offers optimization suggestions and requests further refinement, and \mathbf{a}_j provides the refined solution. Thus, the agents associated with a single edge can effectively optimize a solution in one iteration.

Delving deeper, the topological ordering methodically unfolds agent interactions into an interaction sequence, outlining the control flow² within a multi-agent collaboration process. Concurrently, the data flow within this process is consistent with the original dependencies connected by edges, ensuring that the flow of interacted information aligns with the inherent dependencies outlined in the topology.

3.3 Memory Control

In a multi-agent collaboration system, unrestrained context information exchange can lead to excessively long contexts, ultimately limiting scalability.

²Note that although the interaction order is unfolded as a sequence for visualization purposes only, certain substructures (e.g., star-structured topology) inherently support parallel processing, which is essential in enhancing the reasoning efficiency of practical systems.

ity by supporting only a few agents. To address this, we adopt a heuristic mechanism (Qian et al., 2024b) to manage context visibility using short-term and long-term memory. Short-term memory captures the intra-interaction working memory during each dual-agent interaction, ensuring context-aware decision-making. Long-term memory maintains inter-interaction context continuity by transmitting only the final solutions derived from dialogues, not the entire conversational history. This approach ensures that the context of ancestor agents remains Markovian, with solutions propagated only from adjacent agents rather than from all previous dialogues. Consequently, it reduces the risk of context overload while preserving context continuity, thereby enabling scalable multi-agent collaboration across nearly any large-scale network.

Furthermore, an original solution propagating through the network undergoes continuous refinement, improving its quality over time. As solutions traverse the network, they either branch off at divergent nodes or aggregate at convergent nodes. Branching is achieved through parallel propagation, while merging from multiple nodes, akin to a non-linear perceptron, requires an effective aggregation mechanism. Technically, convergent agents assess the strengths and weaknesses of each solution, synthesizing their strengths and discarding weaknesses, which results in a strength-aggregated outcome from "non-linear" decision-making, rather than a simple combination of all solutions.

4 Experiments

Baselines We select different kinds of representative methods for quantitative comparison.

- CoT (Wei et al., 2022b) is a technically general and empirically powerful method that endows LLMs with the ability to generate a coherent series of intermediate reasoning steps, naturally leading to the final solution through thoughtful thinking and allowing reasoning abilities to emerge.
- AUTOGPT (Richards, 2023) is a versatile single-agent system that employs multi-step planning and tool-augmented reasoning to autonomously decompose complex tasks into chained subtasks and iteratively leverages external tools within an environment-feedback cycle to progressively develop effective solutions.
- GPTSWARM (Zhuge et al., 2024) formalizes a swarm of LLM agents as computa-

tional graphs, where nodes represent manually-customized functions and edges represent information flow, significantly surpassing the tree-of-thought method by optimizing node-level prompts and modifying graph connectivity.

- AGENTVERSE (Chen et al., 2024b) recruits and orchestrates a team of expert agents in either a horizontal or vertical topological structure, employing multi-agent linguistic interaction to autonomously refine solutions and demonstrating emergent performance compared to individual agents, serving as both general and powerful multi-agent framework.

Datasets and Metrics We adopt publicly available and logically challenging benchmarks to evaluate across heterogeneous downstream scenarios.

- MMLU (Hendrycks et al., 2020) provides a comprehensive set of logical reasoning assessments across diverse subjects and difficulties, utilizing multiple-option questions to measure general world knowledge and logical inference capabilities. We assess the quality of generated solutions via *accuracy*, reflecting the correctness of responses to multiple-choice questions.
- HumanEval (Chen et al., 2021), a widely recognized benchmark for function-level code generation, designed for measuring basic programming skills. We assess via *pass@k*, reflecting function correctness across multiple standard test cases.
- SRDD (Qian et al., 2024b) integrates complex textual software requirements from major real-world application platforms, designed for repository-level software development, including requirement comprehension, system design, and integration testing. We assess using *quality*, a comprehensive metric that integrates crucial factors including completeness, executability, and consistency.
- CommonGen-Hard (Madaan et al., 2023) requires models to generate coherent sentences incorporating discrete concepts, designed to test systems’ advanced commonsense reasoning, contextual understanding, and creative problem-solving. We assess using a comprehensive score that integrates crucial factors including grammar, fluency, context relevance, and logic consistency (Li et al., 2018; Chen et al., 2024b).

Implementation Details By default, our method utilizes the GPT-3.5-turbo model, chosen for its optimal balance of reasoning efficacy and efficiency.













Method	Paradigm	MMLU	HumanEval	SRDD	CommonGen	AVG.
CoT		0.3544 [†]	<u>0.6098</u> [†]	0.7222 [†]	0.6165	0.5757 [†]
AUTOGPT		0.4485 [†]	0.4809 [†]	0.7353 [†]	0.5972 [†]	0.5655 [†]
GPTSWARM		0.2368 [†]	0.4969	0.7096 [†]	0.6222 [†]	0.5163 [†]
AGENTVERSE		0.2977 [†]	0.7256 [†]	0.7587 [†]	0.5399 [†]	0.5805
MACNET-CHAIN		0.6632	0.3720	0.8056	0.5903	0.6078
MACNET-STAR		0.4456	0.5549	0.7679	<u>0.7382</u>	0.6267
MACNET-TREE		0.3421	0.4878	0.8044	0.7718	0.6015
MACNET-MESH		<u>0.6825</u>	0.5122	0.7792	0.5525	<u>0.6316</u>
MACNET-LAYERED		0.2780	0.4939	0.7623	0.7176	0.5629
MACNET-RANDOM		0.6877	0.5244	<u>0.8054</u>	0.5912	0.6522

Table 1: The overall performance of LLM-driven methods across various datasets, including both single-agent () and multi-agent () paradigms. For each dataset, the highest scores are highlighted in bold, while the second-highest scores are underlined. A dagger (†) denotes statistically significant differences ($p \leq 0.05$) between the baseline and our chain-structured setting.

We enhance the diversity of perspectives by leveraging GPT-4 to generate a pool of 4,000 profiles for assignment. These agents are equipped to autonomously use external tools (*e.g.*, Python compilers), and their temperatures decrease linearly from 1.0 to 0.0 according to topology depths. Topological sorting is implemented via Kahn’s algorithm (Kahn, 1962). During agent interactions, a maximum of three rounds of utterances is allowed. To ensure fairness, all baselines adhere to identical hyperparameters and settings in the evaluation. All code and data will be publicly available.

4.1 Does Our Method Lead to Superior Performance?

We first employ the simplest topology—chain—as the default setting for our comparative analysis. As shown in Table 1, the chain-structure method consistently outperforms all baseline methods across most metrics, demonstrating a significant margin of improvement. The primary advantage of MACNET, compared to a single agent providing answers from a specific perspective, lies in its facilitation of a sequential process where solutions are continuously refined. This enables autonomous and incremental optimization, effectively alleviating previously imperfect solutions or false hallucinations (Qian et al., 2024b,a; Chen et al., 2024b; Chan et al., 2024). Moreover, we observe that CoT exhibits strong performance on certain datasets, even surpassing some multi-agent methods in specific cases. This is primarily because the underlying knowledge of widely-researched benchmarks is largely embed-

ded in foundational models, giving single agents a notable capability in these relatively "simple" tasks. Although GPTSWARM self-organizes agents through dynamic optimization of nodes and edges, it still requires extensive task-specific customization for all agents and their behaviors, making it challenging to seamlessly transfer to heterogeneous downstream tasks. Given the increasing need for highly performant and automatic real-world systems, it is unrealistic to expect that all preparatory knowledge can be fully pre-encoded in foundation models, nor can specific adaptations be pre-made for all unforeseen complex tasks. Luckily, MACNET addresses this challenge by automatically generating various networks through simple hyperparameters (*e.g.*, topology type and scale), without requiring additional specific adaptations, which represents a more promising paradigm for enhancing autonomy, scalability, and generalizability.

In addition, we ablate agents’ profiles and temperature, which is equivalent to graph-of-thought method—graph-guided reasoning thoughts by a single agent who lacks a profile and has a temperature set to 0. We find that ablating these mechanisms results in significant performance degradation across all topologies, with an average decrease of 2.69%. This highlights the superior collective intelligence over any form of reasoning by a single agent, as the latter corresponds to a feature dimension reduction of the high-dimensional multi-agent combination space, which solidifies reasoning ability due to the lack of flexibility to explore a better configuration.

4.2 How Do Different Topologies Perform Against Each Other?

To understand the topological properties, we conducted extensive experiments by altering MACNET’s topologies. The results in Table 1 demonstrate that different topologies exhibit varying levels of effectiveness for distinct tasks. For instance, a chain topology is more suitable for software development, while a mesh topology excels in logical selection. No single topology consistently delivers optimal results across all tasks. Further observation reveals that topologies approaching the small-world property (Watts and Strogatz, 1998)—characterized by a small *average path length*³—tend to exhibit superior performance, which we refer to as the "*small-world collaboration phenomenon*". Concretely, as each edge in MACNET triggers agent interactions, the graph’s density naturally represents the agents’ interaction density. Empirically, higher interaction density is associated with improved performance among the three coarse-grained topological types.⁴ This performance discrepancy can be attributed to the fact that a higher graph density generally correlates with a higher *clustering coefficient*⁵. This increase in clustering coefficient results in more adjacent node pairs, decreasing the average path length; consequently, the likelihood of long-distance solution invisibility is correspondingly decreased. Along this reason, we also discover that irregular random structures outperform regular mesh structures. This advantage can be attributed to random edge connections, which, in analogy to social networks, potentially link "unacquainted" agents via a direct shortcut, making them adjacent "acquaintances" and implicitly reducing the average path length, thus resembling small-world properties. Meanwhile, unlike mesh topology, which exhibit the highest interaction density, random topology achieve an optimal balance between reduced arrangement depth and enhanced reasoning efficiency, making it a more suitable tradeoff in practice.

Additionally, it is observed that, given the same density, "wider" star-shaped topologies generally

³Average path length (Albert and Barabási, 2001) is the average number of steps along the shortest paths for all possible pairs of network nodes, which is a measure of the efficiency of information transport on a network.

⁴For example, the densely connected mesh topology outperforms the moderately dense tree topology, which in turn outperforms the sparsely connected chain topology.

⁵The clustering coefficient measures how densely connected a node’s neighbors are to each other (Strogatz, 2001).

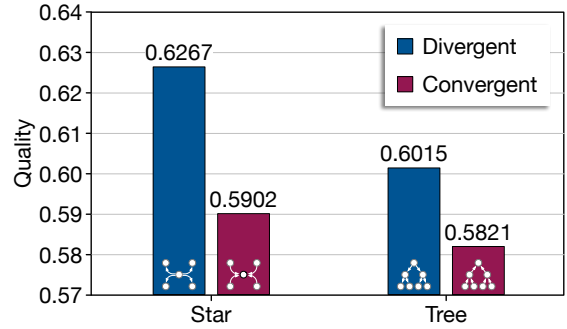


Figure 5: The average performance of the divergent topology (default) and its convergent counterpart.

outperform "deeper" tree-shaped ones. This is primarily attributable to our solution propagation mechanism, which inhibits the propagation of excessively long contextual reasoning processes throughout the entire network. As a result, deeper topologies may cause agents to lose sight of farther contexts, potentially leading to version roll-back—solutions revert to earlier or similar versions. The same principle applies to graph structures, in which mesh topologies, compared to layered ones, enable direct reasoning between agents through direct edges, thereby implicitly reducing network depth and enhancing performance.

In addition to the structural point of view, the directional characteristics of some topologies, which exhibit inherent asymmetry—reversing the edges results in an entirely unequal one—motivated us to explore reverse topologies. As shown in Figure 5, merely altering the symmetry topologies’ orientation leads to significant performance degradation. Typically, divergent structures (those with more child nodes than parent nodes) significantly outperform convergent counterparts. Intuitively, solution flow smoothly diverges, allowing each agent to propose solutions from varied perspectives concurrently; conversely, converging the solutions of multiple agents at a single point poses a greater challenge, illustrating the complexity involved in integrating diverse perspectives into a cohesive strategy.

4.3 Does a Collaborative Scaling Law Exist?

Recall that the neural scaling law fosters emergent capabilities (Kaplan et al., 2020; Schaeffer et al., 2024; Muennighoff et al., 2024), where the synergy among numerous neurons enables a continuous trend of performance improvement. To investigate the *collaborative scaling law*—the potential predictable relationship between agent scale and

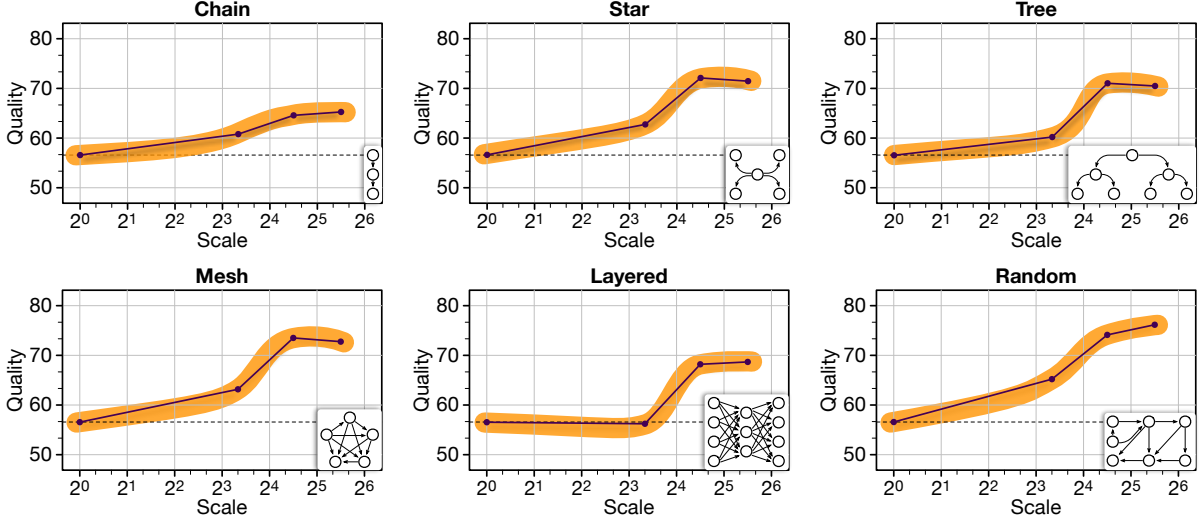


Figure 6: Scaling performance of multi-agent collaboration under different topologies.

performance, considering the associated time and economic costs—we scaled different topologies by exponentially increasing the number of nodes from 1 (regressing to a single-agent method) to 50 (corresponding to 1,275 agents on a mesh setting). As shown in Figure 6, our results confirm the small-world collaboration phenomenon, where optimal outcomes are achieved in high-density networks. Additionally, a reverse degradation phenomenon can be also observed, where certain configurations led to an overall quality reduction ranging from 2.27% to 6.24%.

As the topology scales, the quality of solutions produced by the multi-agent system initially rises rapidly before reaching a saturation point (or slightly declining), which can be approximated by a sigmoid-shaped function:

$$f(x) = \frac{\alpha}{1 + e^{-\beta(x-\gamma)}} + \delta \quad (5)$$

where α , β , γ and δ being real numbers specific to a topology. It is important to emphasize that this is only an average characterization based on scale; a more precise multi-agent system should consider additional factors (*e.g.*, foundation models, profile, and tool spaces). Notably, neural scaling laws typically require a million-fold increase in neurons to reveal significant trends around a scale of 10^{18} to 10^{24} (Schaeffer et al., 2024). In contrast, most topologies in MACNET exhibit performance saturation around a scale of 2^4 to 2^5 . This collaborative emergence occurs more rapidly compared to neural emergence and is observable at smaller scales. The underlying reason is that neuron coordination,

relying on from-scratch training in latent space via matrix operations, requires a vast scale to incorporate extensive world knowledge and develop learning capabilities. In contrast, agent coordination, based on the implicit knowledge of pretrained LLMs, leverages the understanding and refinement of textual information through linguistic interactions, often circumventing the extensive scaling needed by neuronal coordination. Combining these two scaling mechanisms at different levels holds promise for producing higher-quality outcomes.

5 Conclusion

We have introduced MACNET, which leverages DAGs to structure the agents’ cooperative topologies and streamline their interactive reasoning through topological ordering, with solutions derived from their dialogues. Extensive experiments demonstrate that MACNET consistently outperforms all baseline models, enabling effective agent collaboration across various topologies. Notably, we observed a *small-world collaboration phenomenon*, where topologies resembling small-world properties demonstrated superior performance. Additionally, we identified a *collaborative scaling law*, revealing that normalized solution quality follows a logistic growth pattern as scaling agents. Meanwhile, collaborative emergence can be observed occurring significantly earlier compared to previous instances of neural emergence. We hope our findings offer valuable insights into resource prediction and optimization to enhance the efficiency and scalability of LLM systems.

References

- Réka Albert and Albert-László Barabási. 2001. [Statistical Mechanics of Complex Networks](#). volume cond-mat/0106096.
- Maciej Besta, Florim Memedi, Zhenyu Zhang, Robert Gerstenberger, Guangyuan Piao, Nils Blach, Piotr Nyczyk, Marcin Copik, Grzegorz Kwaśniewski, Jürgen Müller, Lukas Gianinazzi, Ales Kubicek, Hubert Niewiadomski, Aidan O’Mahony, Onur Mutlu, and Torsten Hoeffler. 2024. [Demystifying Chains, Trees, and Graphs of Thoughts](#). In *arXiv preprint arXiv:2401.14295*.
- J. A. Bondy and U.R. Murty. 1976. [Graph Theory with Applications](#). In *London: Macmillan*.
- Thorsten Brants, Ashok C. Popat, Peng Xu, Franz J. Och, and Jeffrey Dean. 2007. [Large Language Models in Machine Translation](#). In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 858–867.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language Models are Few-Shot Learners](#). In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 1877–1901.
- Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. [Sparks of Artificial General Intelligence: Early Experiments with GPT-4](#). In *arXiv preprint arXiv:2303.12712*.
- Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2024. [Chateval: Towards Better LLM-based Evaluators through Multi-agent Debate](#). In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Dake Chen, Hanbin Wang, Yunhao Huo, Yuzhao Li, and Haoyang Zhang. 2023. [GameGPT: Multi-agent Collaborative Framework for Game Development](#). In *arXiv preprint arXiv:2310.08067*.
- Lingjiao Chen, Jared Quincy Davis, Boris Hanin, Peter Bailis, Ion Stoica, Matei Zaharia, and James Zou. 2024a. [Are More LLM Calls All You Need? Towards Scaling Laws of Compound Inference Systems](#). In *arXiv preprint arXiv:2403.02419*.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. [Evaluating Large Language Models Trained on Code](#). In *arXiv preprint arXiv:2107.03374*.
- Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, et al. 2024b. [AgentVerse: Facilitating Multi-agent Collaboration and Exploring Emergent Behaviors in Agents](#). In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. 2023. [LM vs LM: Detecting Factual Errors via Cross Examination](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 12621–12640.
- Jonathan L. Gross, Jay Yellen, and Mark Anderson. 2018. [Graph Theory and Its Applications](#). In *Chapman and Hall/CRC*.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Xiaodong Song, and Jacob Steinhardt. 2020. [Measuring Massive Multitask Language Understanding](#). In *arXiv preprint arXiv:2009.03300*.
- J J Hopfield. 1982. [Neural Networks and Physical Systems with Emergent Collective Computational Abilities](#). In *Proceedings Of The National Academy Of Sciences (PNAS)*.
- Wenyue Hua, Lizhou Fan, Lingyao Li, Kai Mei, Jianchao Ji, Yingqiang Ge, Libby Hemphill, and Yongfeng Zhang. 2023. [War and Peace \(WarAgent\): Large Language Model-based Multi-Agent Simulation of World Wars](#). In *arXiv preprint arXiv:2311.17227*.
- A. B. Kahn. 1962. [Topological Sorting of Large Networks](#). *Communications of the ACM*, 5(11):558–562.
- 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](#). In *arXiv preprint arXiv:2001.08361*.
- Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023a. [CAMEL: Communicative Agents for “Mind” Exploration of Large Language Model Society](#). In *Thirty-seventh Conference on Neural Information Processing Systems (NeurIPS)*.
- Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. 2024. [More Agents is All You Need](#). *arXiv preprint arXiv:2402.05120*.
- Yuan Li, Yixuan Zhang, and Lichao Sun. 2023b. [Metaagents: Simulating Interactions of Human Behaviors](#)

- for LLM-based Task-oriented Coordination via Collaborative Generative Agents. In *arXiv preprint arXiv:2310.06500*.
- Zhongyang Li, Xiao Ding, and Ting Liu. 2018. Generating Reasonable and Diversified Story Ending using Sequence to Sequence Model with Adversarial Training. In *the International Conference on Computational Linguistics (COLING)*, pages 1033–1043.
- Zijun Liu, Yanzhe Zhang, Peng Li, Yang Liu, and Diyi Yang. 2023. Dynamic LLM-Agent Network: An LLM-agent Collaboration Framework with Agent Team Optimization. In *arXiv preprint arXiv:2310.02170*.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-Refine: Iterative Refinement with Self-Feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. 2024. Scaling Data-Constrained Language Models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.
- Anton Nilsson, Carl Bonander, Ulf Strömberg, and Jonas Björk. 2020. A Directed Acyclic Graph for Interactions. In *International Journal of Epidemiology*.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training Language Models to Follow Instructions with Human Feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 27730–27744. Curran Associates, Inc.
- Joon Sung Park, Joseph O’Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative Agents: Interactive Simulacra of Human Behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*, pages 1–22.
- Kai Petersen, Claes Wohlin, and Dejan Baca. 2009. The Waterfall Model in Large-Scale Development. In *Product-Focused Software Process Improvement*, pages 386–400.
- Giorgio Piatti, Zhijing Jin, Max Kleiman-Weiner, Bernhard Schölkopf, Mrinmaya Sachan, and Rada Mihalcea. 2024. Cooperate or Collapse: Emergence of Sustainability Behaviors in a Society of LLM Agents. *arXiv preprint arXiv:2404.16698*.
- Chen Qian, Yufan Dang, Jiahao Li, Wei Liu, Zihao Xie, Yifei Wang, Weize Chen, Cheng Yang, Xin Cong, Xiaoyin Che, Zhiyuan Liu, and Maosong Sun. 2024a. Experiential Co-Learning of Software-Developing Agents. In *the Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2024b. ChatDev: Communicative Agents for Software Development. In *the Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. 2024. ToolLLM: Facilitating Large Language Models to Master 16000+ Real-World APIs. In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2023. Large Language Models are Effective Text Rankers with Pairwise Ranking Prompting. In *arXiv preprint arXiv:2306.17563*.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language Models are Unsupervised Multitask Learners. In *OpenAI blog*, volume 1, page 9.
- Toran Bruce Richards. 2023. AutoGPT. In <https://github.com/Significant-Gravitas/AutoGPT>.
- Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. 2024. Are Emergent Abilities of Large Language Models a Mirage? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.
- Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language Models Can Teach Themselves to Use Tools. In *arXiv preprint arXiv:2302.04761*.
- Murray Shanahan, Kyle McDonell, and Laria Reynolds. 2023. Role Play with Large Language Models. In *Nature*, volume 623, pages 493–498.
- 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.
- Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zhang, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. 2022. Using DeepSpeed and Megatron to

- Train Megatron-Turing NLG 530B, A Large-Scale Generative Language Model. In *arXiv preprint arXiv:2201.11990*.
- Steven H. Strogatz. 2001. [Exploring Complex Networks](#). In *Nature*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. [Llama: Open and Efficient Foundation Language Models](#). In *arXiv preprint arXiv:2302.13971*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is All You Need](#). In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30.
- Haotian Wang, Xiyuan Du, Weijiang Yu, Qianglong Chen, Kun Zhu, Zheng Chu, Lian Yan, and Yi Guan. 2023. [Apollo’s Oracle: Retrieval-Augmented Reasoning in Multi-Agent Debates](#). In *arXiv preprint arXiv:2312.04854*.
- Duncan J. Watts and Steven H. Strogatz. 1998. [Collective Dynamics of Small-World Networks](#). *Nature*, 393:440–442.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022a. [Emergent Abilities of Large Language Models](#). In *Transactions on Machine Learning Research*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022b. [Chain-of-thought Prompting Elicits Reasoning in Large Language Models](#). In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 24824–24837.
- Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. 2024. [Large Language Models as Optimizers](#). In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. [Tree of Thoughts: Deliberate Problem Solving with Large Language Models](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. [Expel: LLM Agents are Experiential Learners](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19632–19642.
- Wangchunshu Zhou, Yuchen Eleanor Jiang, Long Li, Jialong Wu, Tiannan Wang, Shi Qiu, Jintian Zhang, Jing Chen, Ruipu Wu, Shuai Wang, Shiding Zhu, Jiyu Chen, Wentao Zhang, Ningyu Zhang, Huajun Chen, Peng Cui, and Mrinmaya Sachan. 2023. [Agents: An Open-source Framework for Autonomous Language Agents](#). In *arXiv preprint arXiv:2309.07870*.
- Mingchen Zhuge, Wenyi Wang, Louis Kirsch, Francesco Faccio, Dmitrii Khizbullin, and Jurgen Schmidhuber. 2024. [Language Agents as Optimizable Graphs](#). *arXiv preprint arXiv:2402.16823*.